INGENIERIA DE CONTROL DIGITAL DE PROCESOS Y APLICACIONES
(Octubre 1977)

Fecha	Duración	Tema	Profesor
Oct. 7	17 a 21 h	Introducción y Control Supervisorio	Dr. Víctor Gerez Greiser
Oct. 8	9 a 13 h	Hardware	M. en C. Marcial Portilla Robertson
	14 a 18 h	Software	M. en C. Marcial Portilla Robertson
Oct. 14	17 a 21 h	Sistemas Muestreados	M. en C. Mauricio Mier Muth
Oct. 15	9 a 13 h	Consideraciones en el Dominio de la Frecuencia	M. en C. Mauricio Mier Muth
	14 a 18 h	Técnicas de Identificación	Ing. Rafael López López
Oct. 21	17 a 21 h	Algorítmos de Control	Dr. Víctor Gerez Greiser
Oct. 22	9 a 13 h	Algorítmos de Control	Dr. Víctor Gerez Greiser
	14 a 18 h	Técnicas Avanzadas de Control	Dr. Víctor Gerez Greiser
Oct. 28	17 a 21 h	Control Optimo	Ing. Rafael López López
Oct. 29	14 a 18 h	Aplicaciones y Simulación	M. en C. Marcial Portilla Robertson
		Clausura	

PROFESORES DEL CURSO INGENIERIA DE CONTROL DIGITAL DE PROCESOS Y APLICACIONES

DR. VICTOR GEREZ GREISER
Profesor Titular
Ingeniería Mecánica y Eléctrica
Facultad de Ingeniería
UNAM
Tel.: 550.52.15 E. 3746

M.EN C. MAURICIO MIER MUTH
Investigador
División de Sistemas de Potencia
Shakespeare 6-3°
México 5, D.F.
Tel.: 525.64.52

M. EN C. MARCIAL PORTILLA ROBERTSON
Jefe de la Sección de Computación
Edificio de Ingeniería Mecánica y Eléctrica
Facultad de Ingeniería
UNAM
Tel.: 550.52.15 E.3750 y 3746

'edcs. 6, X, 1977.

centro de educación continua
división de estudios superiores
facultad de ingeniería, unam

INGENIERIA DE CONTROL DIGITAL DE PROCESOS Y APLICACIONES

INTRODUCCION Y CONTROL SUPERVISORIO

DR. VICTOR GEREZ GREISER

NOVIEMBRE, 1978.

~~EL~~ PROBLEMA DEL CONTROL DE PROCESOS

Ejemplo: SISTEMA ELECT. DE POT.

Función del Sistema: Suministrar:

Potencia Activa P } que de-
Potencia Reactiva Q } mandan los usuarios.

con Restricciones sobre:

Voltage V

frecuencia f

Problema de operación:

Modelar el sistema

Desarrollar estrategias de operación optima

Implementarlas (Control)

EL PROBLEMA DEL CONTROL DE PROCESOS

①

Ejemplo: <u>SISTEMA ELECT. DE POT.</u>

Función del Sistema: <u>Suministrar</u>

Potencia Activa P } que demandan los usuarios.
Potencia Reactiva Q }

~~con~~ <u>Restricciones</u> sobre:

Voltage V

frecuencia f

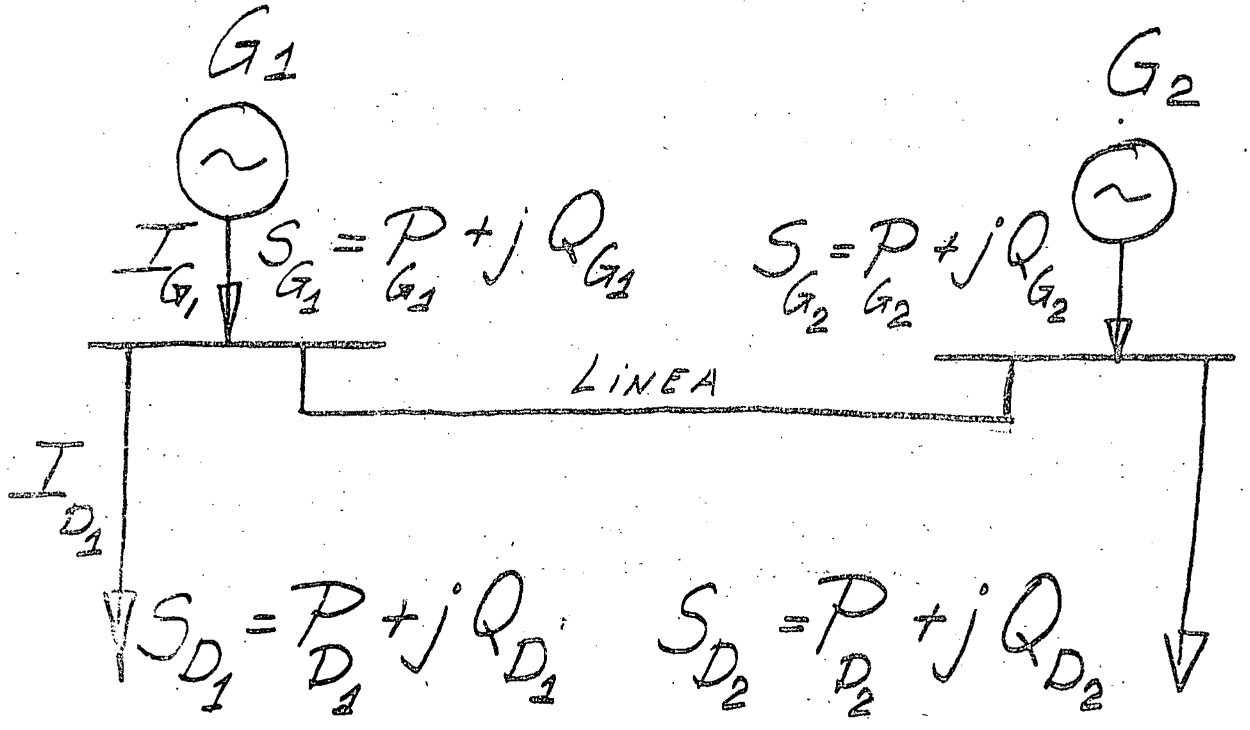
Problema de operación:

Modelar el sistema

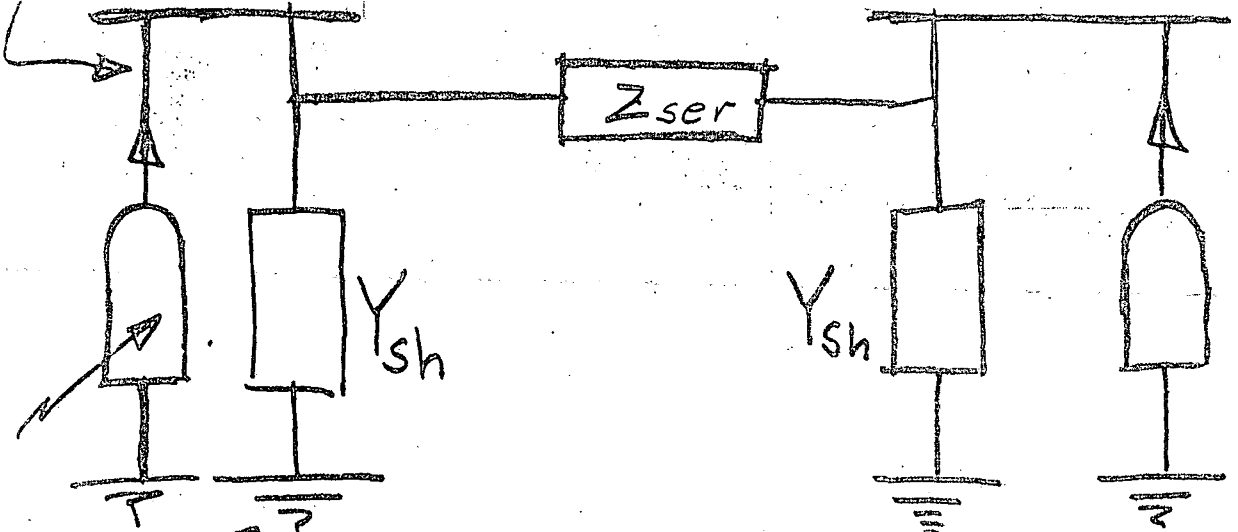
Desarrollar estrategias de operación optima

Implementarlas (Control)

MODELADO DEL SISTEMA:



$$J_1 = I_{G_1} - I_{D_1}$$



$$S_1 = S_{G_1} - S_{D_1}$$

Net bus power:

$$S_1 = P_1 + jQ_1$$

$$S_2 = P_2 + jQ_2$$

$$\begin{aligned} S_1 &\triangleq P_{G_1} - P_{D_1} + j(Q_{G_1} - Q_{D_1}) \\ S_2 &\triangleq P_{G_2} - P_{D_2} + j(Q_{G_2} - Q_{D_2}) \end{aligned} \qquad (1)$$

Potencia Real Generada =
Consumo + Pérdidas
$(F = \text{cst.})$

Potencia Reactiva Generada =
Consumo + Pérdidas
$(V = \text{cst.})$

$$S = V I^*$$

$$S^* = V^* I$$

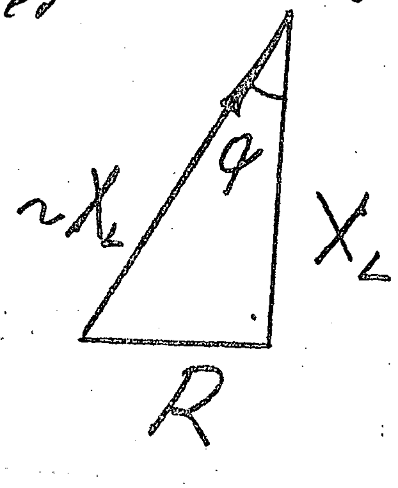
$$\frac{S_1^*}{V^*} = V_1 Y_{sh} + \frac{V_2 - V_1}{Z_{ser}} \quad (2)$$

otra para bus (2)

Simplificaciones:

$$Y_{sh} = \frac{j}{X_c} \quad \text{capacitivo serie} \quad (3)$$

$$Z_{ser} = R + j X_L \quad (4)$$



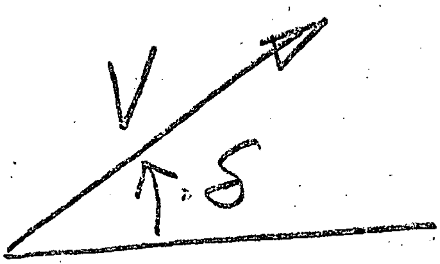
$$X_L \gg R$$

$$Z_{ser} \cong X_L \angle \frac{\pi}{2} - \varphi$$

Tensiones de Bus: ⑤

$$V_1 = |V_1| \angle \underline{\delta_1}$$

$$V_2 = |V_2| \angle \underline{\delta_2} \qquad (5)$$



Sustituyendo (1); (3); (4) y (5) en (2) ⟹

Modelo:

$$P_{G1} - P_{D1} - \frac{|V_1|^2}{X_L}\sin\alpha + \frac{|V_1|\,|V_2|}{X_L}\sin[\alpha - (\delta_1 - \delta_2)] = 0$$

$$P_{G2} - P_{D2} - \frac{|V_2|^2}{X_L}\sin\alpha + \frac{|V_1|\,|V_2|}{X_L}\sin[\alpha + (\delta_1 - \delta_2)] = 0$$

$$Q_{G1} - Q_{D1} + \frac{|V_1|^2}{X_c} - \frac{|V_1|^2}{X_L}\cos\alpha + \frac{|V_1|\,|V_2|}{X_L}\cos[\alpha - (\delta_1 - \delta_2)] = 0 \qquad (7\text{-}6)$$

$$Q_{G2} - Q_{D2} + \frac{|V_2|^2}{X_c} - \frac{|V_2|^2}{X_L}\cos\alpha + \frac{|V_1|\,|V_2|}{X_L}\cos[\alpha + (\delta_1 - \delta_2)] = 0$$

Características:

1) Ecs. algebraicas
2) No lineales (Comp. digital)
3) Relacionan tensiones con potencia
4) No aparece f (Estado estable)
5) Siempre aparece

$$\delta_1 - \delta_2 \doteq \delta_{12} = \delta$$

6) Doce variables

Cuatro ecuaciones

$\Downarrow$

Hay que pedefinir 8

Clasificacion de variables

a) Fuera de control ó disturbios

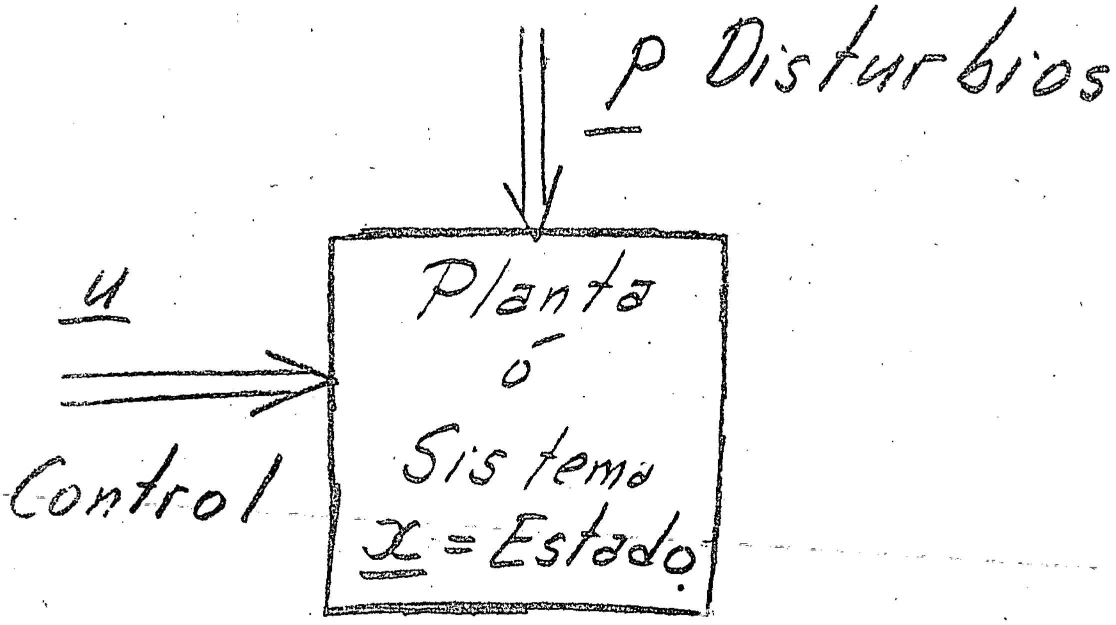
$$\underline{p} = \begin{bmatrix} P_{D_1} \\ Q_{D_1} \\ P_{D_2} \\ Q_{D_2} \end{bmatrix}$$

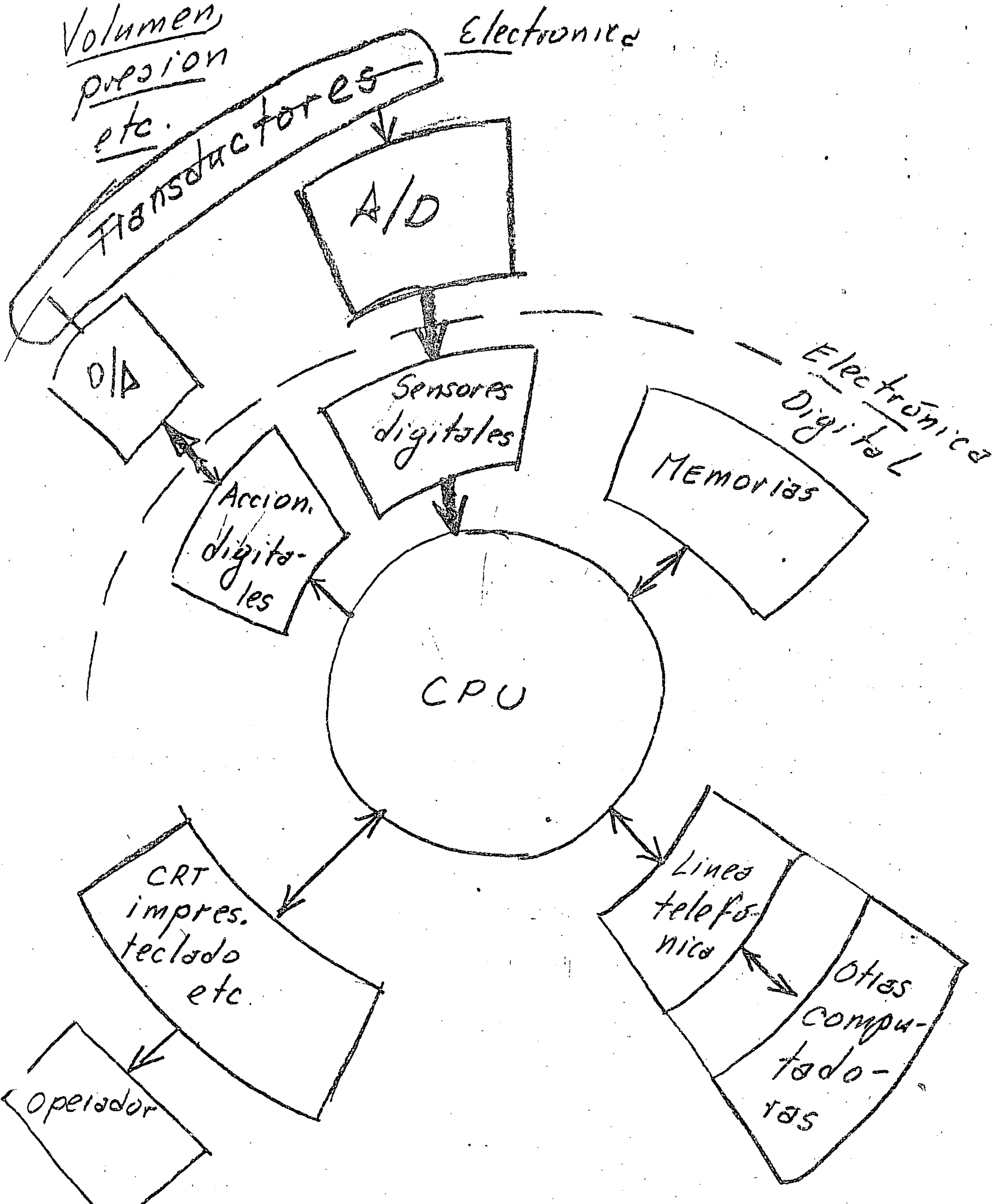
b) Control o Manipuladas

$$\underline{u} = \begin{bmatrix} P_{G_1} \\ Q_{G_1} \\ P_{G_2} \\ Q_{G_2} \end{bmatrix}$$

c) De estado

$$\underline{x} = \left[\begin{matrix} \delta_1 \\ |V_1| \\ \delta_2 \\ |V_2| \end{matrix}\right.$$




Volumen
presion
etc.
Electronica
Transductores
A/D
D/A
Electrónica
Digital
Sensores
digitales
Memorias
Accion.
digita-
les
CPU
CRT
impres.
teclado
etc
Operador
Linea
telefo-
nica
Otras
compu-
tado-
ras

centro de educación continua
división de estudios superiores
facultad de ingeniería, unam

INGENIERIA DE CONTROL DIGITAL DE PROCESOS Y APLICACIONES

HARDWARE SOFTWARE

DR. VICTOR GEREZ GREISER
M. EN C. MARCIAL PORTILLA ROBERTSON

EL PAPEL DE LA COMPUTADORA EN EL CONTROL DIGITAL DIRECTO DE PROCESOS.

Existe consenso que durante el cuarto de siglo precedente la computadora fue el principal avance tecnológico que ha tenido impacto en todas las ramas de la ingeniería y muy particular en el control de procesos.

En este campo existe todavía una amplia posibilidad de aplicar conceptos teóricos de control a aplicaciónes reales.

Desde luego que al considerar posibles aplicaciones es necesario tomar en cuenta tanto los equipos de control (Hardware) como la programación necesaria para implementar las funciones de control (software).

EL PROBLEMA DEL CONTROL DE PROCESOS.

Como se ilustra en el problema del sistema eléctrico de potencia que aparece en el apéndice de este capítulo, es posible distinguir diversas variables al analizar un proceso de control.

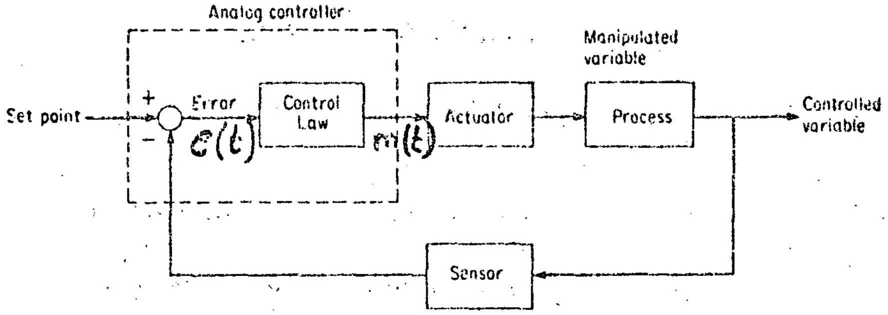
1. Variables de control o controlables. Son aquellas cuyos valores pueden ajustarse como son en el caso del sistema eléctrico, la corriente de excitación del generador y el par aplicado a la turbina.

2. Distrubios. Estas variables desde luego afectan a la operación del proceso o del sistema pero no pueden ser sujetas a ajustes. En el sistema eléctrico la potencia real y reactiva que demandan los consumidores está fuera del control del sistema.
3. Variables controladas. Estas variables son las que determinan la operación de la planta. Son aquellas para las cuales se diseña una extrategia de control con objeto de mantenerlas dentro de ciertos límites. En nuestro ejemplo de sistema eléctrico de potencia son éstas la tensión y la frecuencia.
4. Variables intermedias. En diferentes puntos del proceso apa-
   re recen otras variables que en caso de ser observable el sistema pueden emplearse para obtener información sobre su estado de operación.

Como ilustra claramente el ejemplo de sistemas de potencia uno de los problemas mas difíciles de resolver es la determinación del modelo matemático adecuado para controlar el proceso. En procesos o sistemas grandes el número de variables que hay que medir y en función de las cuales hay que determinar una estrategia de control es enorme. La computadora digital con su habilidad de colectar una gran cantidad de información, analizarla y tomar decisiónes lógicas basadas en estos resultados resulta la herramienta ideal para este tipo de aplicaciones.

Sistemas de control analógico (convencional).

Como muestra la figura la parte más importante y característica de un sistema de control es la realimentación.



SISTEMA REALIMENTADO.

La señal de entrada marca el valor que debe tener la variable de salida o controlable. En el llamado punto de suma se comparan ambas señales y se genera el error que sirve como señal de entrada al controlador.

Este dispositivo genera una señal que en el caso más general en este tipo de controles es proporcional al error, a su integral y a su derivada. Como muestra la fórmula siguiente:

$$m(t) = K \left\{ e(t) + \frac{1}{T_i}\int_0^t e(\tau)\, d\tau + T_d \frac{d\,e(t)}{dt} \right\} + m_R$$

En esta fórmula las variables son las siguientes:

K_C = ganancia proporcional

T_i = tiempo de reposición o integral

T_d = constante de derivación

M_r = valor de referencia al cual se inicia la acción de control.

Si bien es posible en teoría ajustar los tres parámetros de la acción de control en la mayoría de las aplicaciones se trabaja exclusivamente con control proporcional e integral.

En la mayoría de los casos este tipo de controladores han sido neumáticos por ser estos sumamente confiables y no presentar peligros en atmósferas explosivas. En fechas recientes sin embargo los avances en la electrónica han permitido construir controladores electrónicos con características equivalentes.

Estos controles adolecen de un problema, son sumamente inflexibles y debe existir una correspondencia uno a uno entre las funciones del lazo de control y el equipo que las implementa. La posibilidad de realizar estrategias complejas con este tipo de elementos analógicos es muy limitada.

A continuación se resumen las principales aplicaciones de las computadoras en el control de proceso.

REGISTRADORAS DE DATOS.

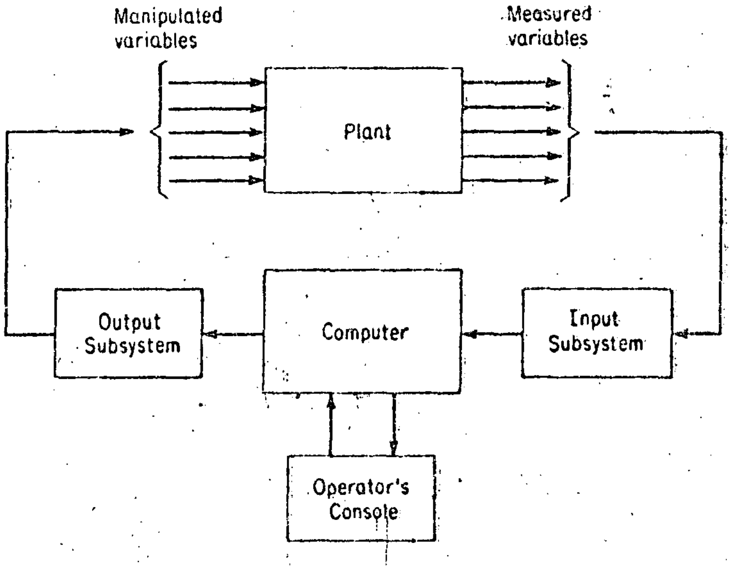
La aplicación más sencilla de una computadora es simplemente como un dispositivo para registrar datos generalmente con alguna lógica sencilla que permita imprimir un mensaje cuando alguna de las variables alcanza valores fuera de sus límites normales.

Los registros que genera la computadora son sin embargo importantes para el diseñador de un sistema de control de procesos ya que pueden emplearse si se han recabado con una estrategia adecuada para construir el modelo.

CONTROL DIGITAL DIRECTO.

En este tipo de esquema de control la computadora calcula el valore de las variables manipuladas directamente del valor de los puntos de ajuste, y de las variables que se miden durante el proceso.

La figura muestra el equema básico de un control digital directo



CONTROL DIGITAL DIRECTO.

En su aplicación más sencilla puede implementarse digitalmente el algoritmo de control proporcional, diferencia e integral cuya versión en este caso esta dada por las fórmulas siguientes:

$$m_n = K_c e_n + \frac{K_c T}{T_i} \sum_{i=0}^{n} e_i + \frac{T_d K_c}{T} (e_n - e_{n-1}) + m_R$$

T = tiempo de muestreo (se explica en el anexo 2).

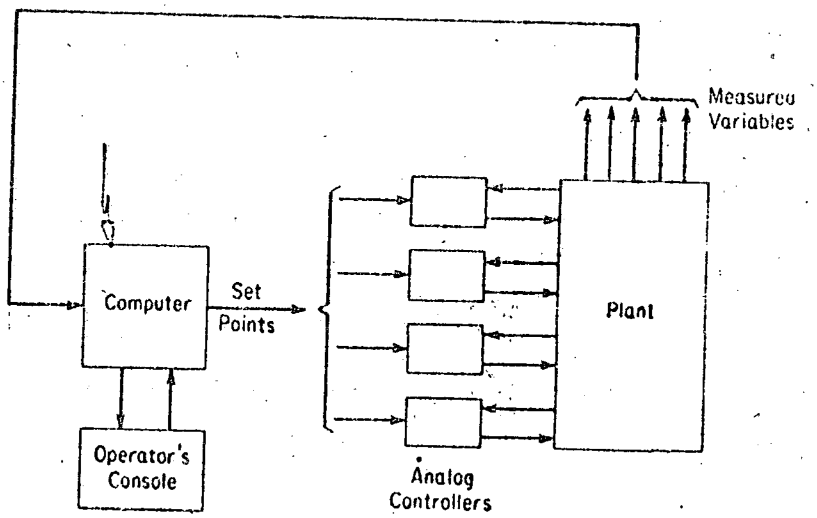
Generalmente no puede justificarse la adquisición de un equipo digital para hacer las mismas funciones que podría hacer un equipo analógico. Es necesario como se verá mas adelante aprovechar plenamente las capacidades del equipo digital implementando control óptimo.

Si puede justificarse la adquisición del equipo digital por razones adicionales a las de implementación de una ley de control proporcional, integral y diferencial, debemos mencionar que empleando técnicas digitales es posible obtener con el algoritmo anterior mejor respuesta que con su vérsión analógica.

CONTROL SUPERVISOR.

Una aplicación muy frecuente de la computadora digital se encuentra en el llamado control supervisorio. Es esta una solución híbrida donde se combina a la computadora con los controladores analógicos. Como muestra la figura, estos últimos realizan directamente la función de control. La computadora en función de variables medidas y de instrucciones que le da el operador a través de la consola e incluyendo generalmente criterios de carácter económico calcula que valor deben tener las diversas acciones de control (proporcional, derivativo e integral) que deben tomar los controles analógicos.

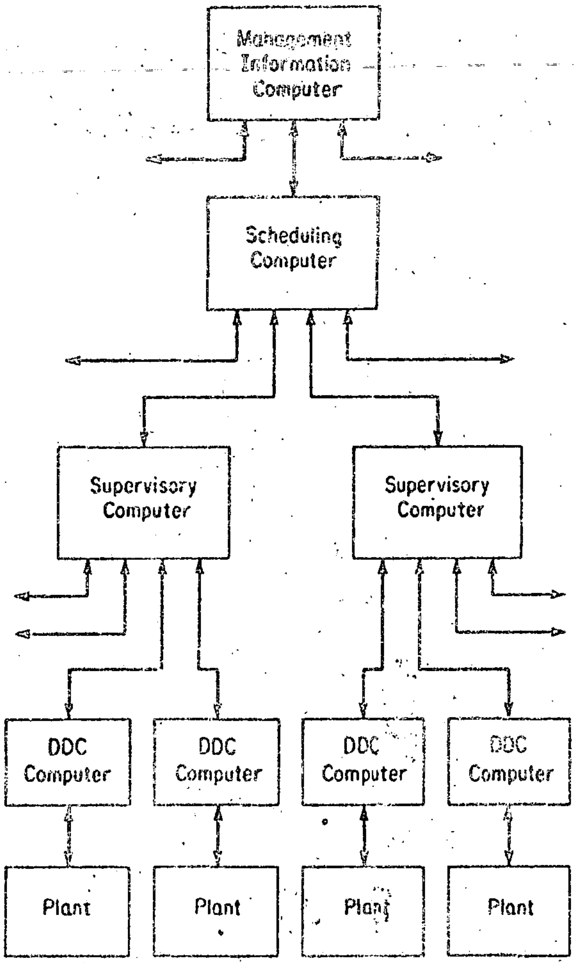
Debemos hacer incapié que la limitación principal para implementar este tipo de control es la disponibilidad de un buen modelo matemático de la planta o sistema que se desea controlar.



CONTROL SUPERVISORIO.

CONTROL JERARQUICO.

En general en grandes sistemas se recurre a un control de carácter jerárquico que es una combinación de control supervisorio, control digital directo y control analógico.



CONTROL JERARQUICO.

En el anexo 2 se explica con mayor detalle este concepto.

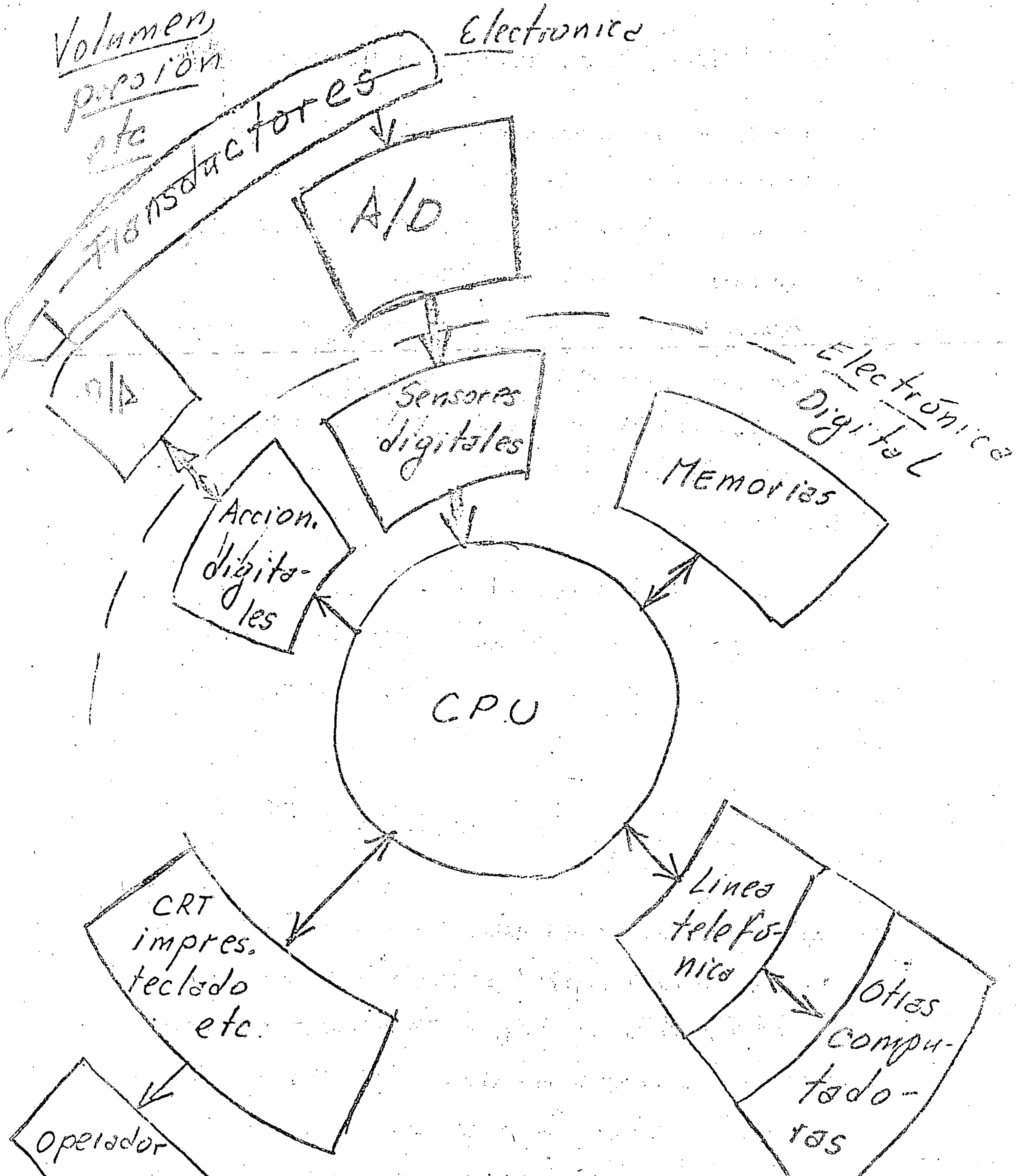
LA COMPUTADORA DIGITAL COMO ELEMENTO DE CONTROL.

Como se muestra en la figura la parte medular de este dispositivo es la unidad central de procesamiento. Los transductores convierten a las señales de carácter analógico en señales eléctricas de igual tipo. Convertidores analógicos digitales los convierten en señales digitales que ya pueden ser procesadas. La información que genera la computadora es también digital y antes de poder ser implementadas estas órdenes tienen en general que convertirse con ayuda de un convertidor digital analógico en una señal analógica.

A pesar del alto grado de automatismo que se logra conestas instalaciones es necesario preveer una interfase con un operador humano a través de tubos de rayos catódicos (CRT) impresoras teclados, etc.

Igualmente importantes son las memorias donde se almacena la información.

Frecuentemente, como en el sistema eléctrico de potencia que cubre una gran extensión territorial es necesario hacer llegar a la máquina información que se genera muy lejos y esta tiene que mandar señales de mando a lugares igualmente distantes, además es necesario que varias computadoras trabajen de manera coordinada todo ello requiere de una compleja red de comunicaciones

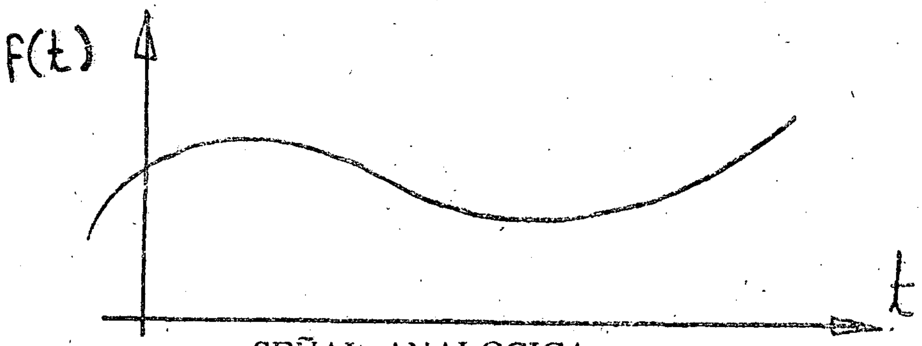


ESTRUCTURA DE UNA COMPUTADORA

que puede ser telefónica, de micro-ondas, por onda portadora sobrepuesta a líneas de transmisión.

SEÑALES ANALOGICAS Y SEÑALES DIGITALES.

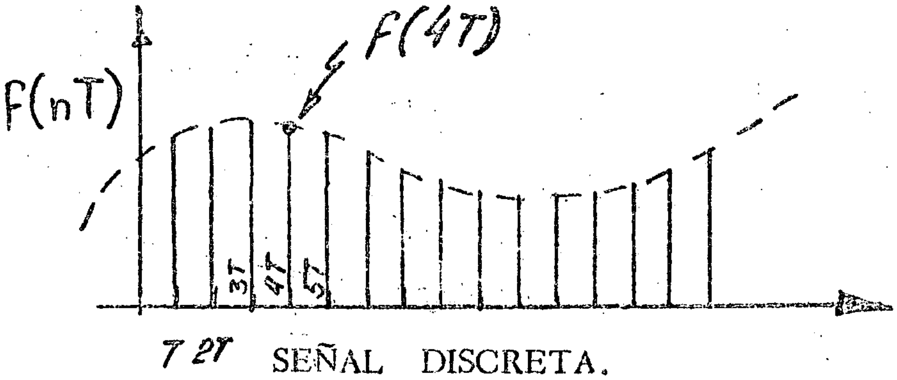
La figura muestra una señal analógica que exceptuando momentos de conexión o desconexión generalmente es contínua.



SEÑAL ANALOGICA

La computadora digital no trabaja con este tipo de señales. Dependiendo del tipo de proceso, en particular de la llamada constante de tiempo o sea de la velocidad con que puede variarse una variable un dispositivo llamado muestreador toma cada T segundos una medición.

De manera de obtener una serie de valores discretos, tal como; muestra la figura.



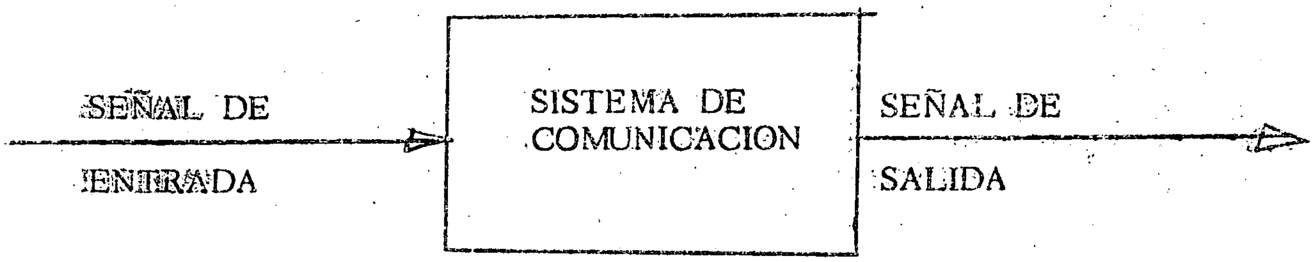
SEÑAL DISCRETA.

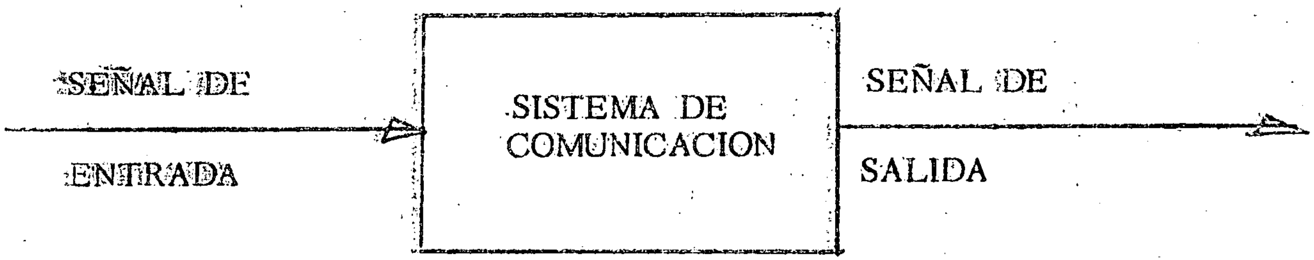
Un convertidor analógico digital transforma estos valores discretos en valores binarios, octales o de alguna otra base según el sistema digital que se estuviese empleando.

Las señales digitalizadas y expresadas en forma binaria o octal tienen en primer lugar la ventaja de ser las que procesan la máquina en segundo lugar, presentan ventajas desde el punto de vista de las comunicaciones. Como muestra la figura debido a la distorsión que se produce en un sistema de comunicaciones es posible que dos señales de entrada diferentes produzca en la salida o recepción señales casi iguales que resulta dificil o imposible de identificar. observando la señal de salida, cuál fué la señal que se transmitió?

Si se transmiten señales binarias por ejemplo, secuencias de ceros y unos (oo100100) el problema de identificación de la señal transmitida se simplifica enormemente ya que en el recepctor basta detectar si hay señal o no. En el primer caso, se concluye que se transmitio un uno mientras que en el segundo caso se decide que se transtió un cero.

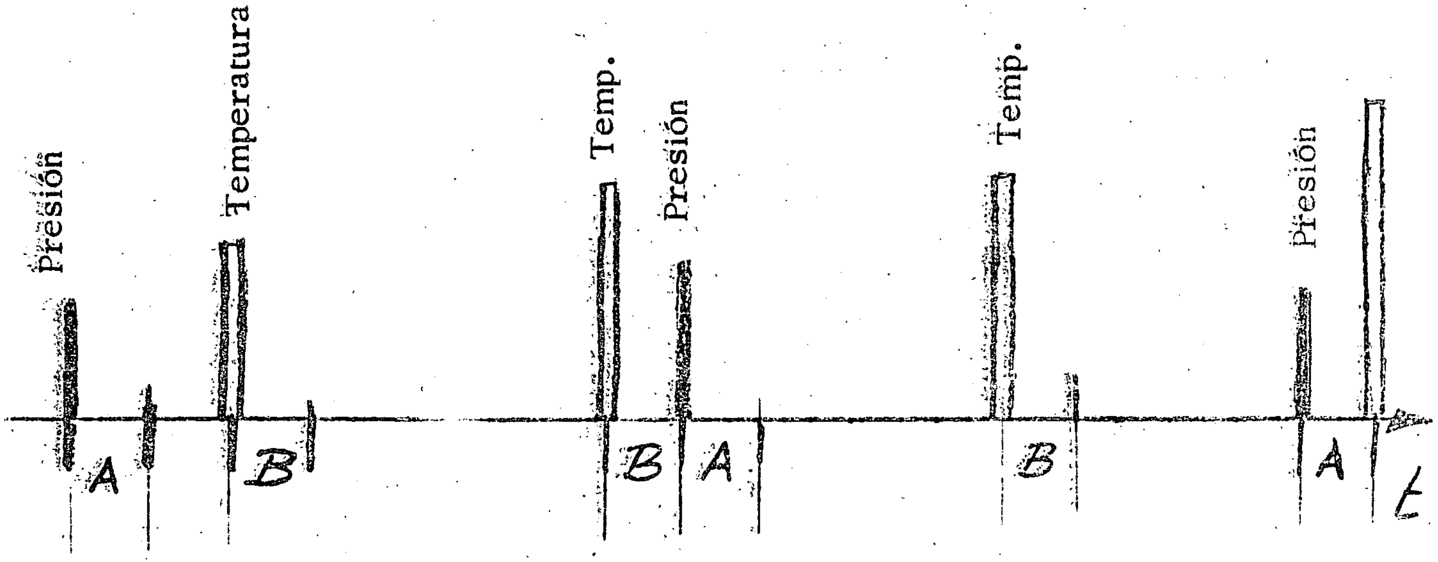
Como en la computadora además se trabaja a una enorme velocidad muy superior a la de muestreo entre operación de muestreo existe tiempo para realizar cálculos e inclusive muestrear otras cantidades.





TRANSMISION DE SEÑALES ANALOGICAS.

Debido a la distorsión producida por el sistema de comunicaciones dos señales de entrada diferentes producen señales de salida casi iguales dificultando o imposibilitando determinar observando la señal transmitida, qué señal se envio?



PROCESAMIENTO DE SEÑALES DIGITALES.

A = Intervalo de tiempo que se requiere para procesar la información sobre presión y enviar una señal de telecomando

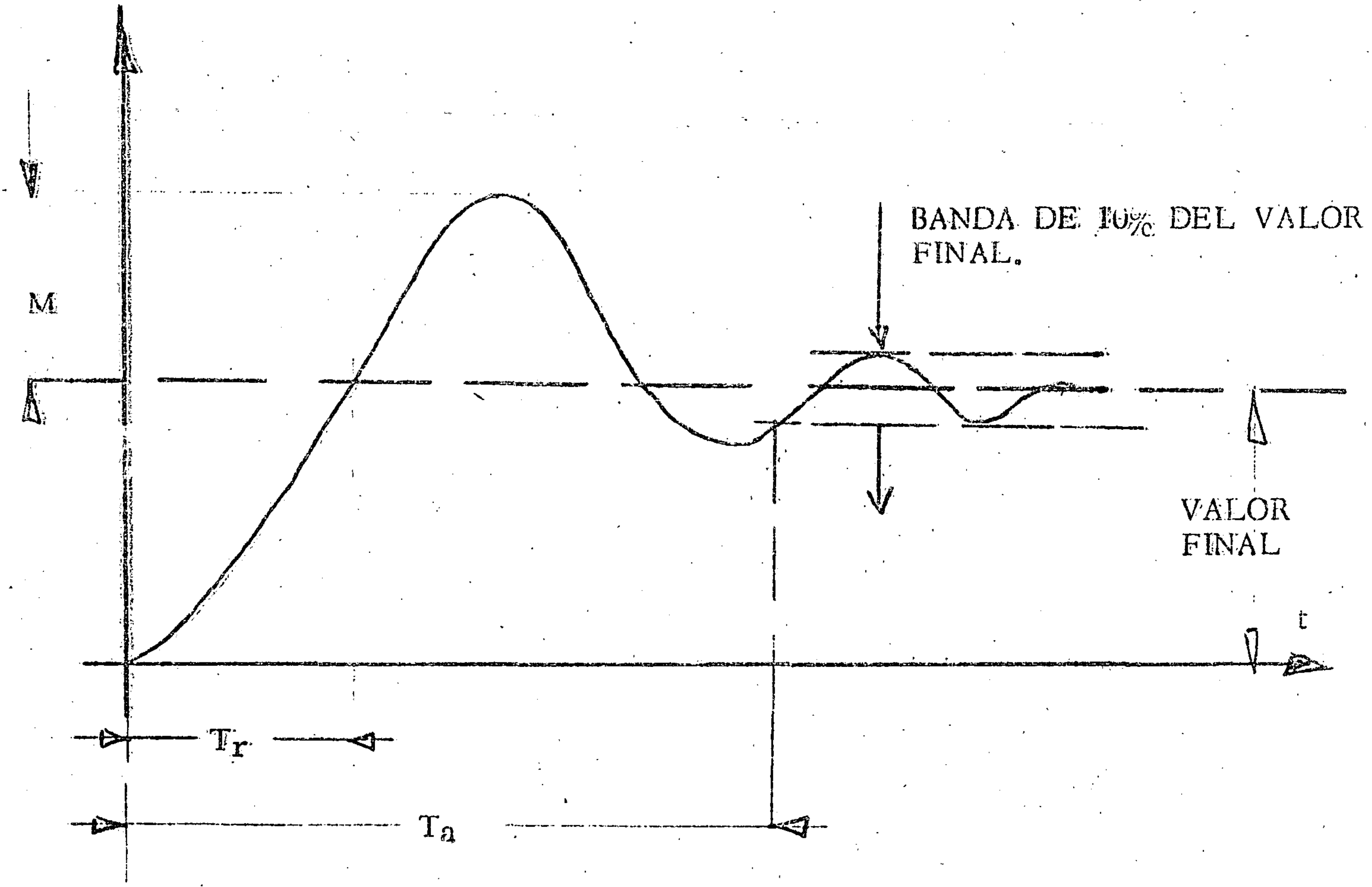
B = Intervalo correspondiente a la señal de temperatura.

DIFERENCIA ENTRE CONTROL CONVENCIONAL Y CONTROL OPTIMO.

La figura muestra la respuesta de un sistema dinámico típico a una señal de escalón. Este sistema puede ser por ejemplo un censor. La situación ideal sería aquella en que la señal de salida tuviese la misma forma que la señal de entrada. Sin embargo, esto no es posible teniendo en general la señal de salida un carácter oscilatorio. Dentro de ciertos límites es posible con controles analógicos del tipo proporcinal integral y diferencial ajustando las ganancias de los diferentes efectos lograr que parámetros de la respuesta como el sobretiro, el tiempo de respuesta y el tiempo de asentamiento tengan determinados valores de diseño. Es muy dificil implementar sin embargo, controladores analógicos que permitan tomar en cuenta criterios de optimalidad como los siguientes: mínimo consumo de energía, mínimo tiempo de respuesta, etc. Este tipo de algoritmos de control óptimo es sin embargo posible implementarlos usando sistemas digitales.

SELECCION ENTRE CONTROL ANALOGICO Y DIGITAL.

Una decisión de este tipo debe pasarse en: el costo de las funciones de control que se realizan, su confiabilidad y facilidad de mantenimiento. Como a esto tres factores se les puede dar un valor económico, en resumen el problema se reduce a seleccionar el sistema más económico.



RESPUESTA TIPICA DE UN SISTEMA DE SEGUNDO ORDEN.

M sobre tiro
T_r = tiempo de respuesta
T_a = tiempo de asentamiento

EN general la diferencia de costos del sistema de censores y actuadores no permite decidir entre un sistema analógico o digital. Es necesario tomar en cuenta las capacidades de uno y otro sistema.

El argumento original de que los sistemas digitales ahorrarían mano de obra resulto ser falaz en general plantas de proceso ya operaban aún antes de la introducción del control digital con mínimo personal. En general puede decirse que la justificación de un control digital debe basarse en consideraciones de la confiabilidad que le da la operación del sistema y al ahorro económico que puede obtenerse empleando esquemas de control óptimo que toman en cuenta factores económicos permitiendo reducir o inclusive minimizar los costos de operación.

El sistema eléctrico de potencia debido a su complejidad no podría operarse sin esta tecnología. Operándolo convencionalmente su confiabilidad no es adecuada y además no se obtienen los beneficios de un control óptimo. En resumen podemos decir que en los siguientes casos se justifica la instalación de un sistema digital:

1. Plantas muy complejas. En estas plantas resulta imposible que el personal leyendo ópticamente las variables tome las decisiones de control adecuadas, debido a su enorme número y a las muy complejas relaciones causa-efecto entre variables y acciones de control. Desde luego se hace indispensable contar con un modelo matemático adecuado para implementar estos esquemas de control.
2. Plantas con muy altos niveles de producción. En estas instalaciones cualquier ahorro por muy pequeño que resulte en el consumo de energía o en el desperdicio de material al cambiar especificaciones en un proceso contínuo representa fuertes sumas de dinero que justifican la instalación de un sis

tema de este tipo.

3. Plantas sujetas a disturbios frecuentes. Estos disturbios pueden ser físicos, como el cambio de demanda en el sistema eléctrico o pueden ser económicos como el cambio de precio en el combustible. En general el control de planta puede compensar por varios de estos disturbios pero resulta necesario cambiar los objetivos de operación empleando la computadora digital.

4. Procesos de manufactura completos. Una aplicaciónde creciente importancia es el control de procesos de manufactura donde el producto tiene que mantenerse dentro de estrechos límites de tolerancia maquinándose además a muy alta velocidad como en una planta de papel o en un tren de laminación. También aquí la computadora digital es un auxiliar indispensable.

Como nota final es necesario hacer incapié en que no debe olvidarse los costos de programación al evaluar un sistema de control digital directo.

APENDICE 1.

EJEMPLO DE LA CONSTRUCCION DEL MODELO DE UN PROCESO

A continuación se ilustra la construcción del modelo estático de un sistema de potencia muy simplificado.

Ejemplo: Sistema Elect. de Pot.

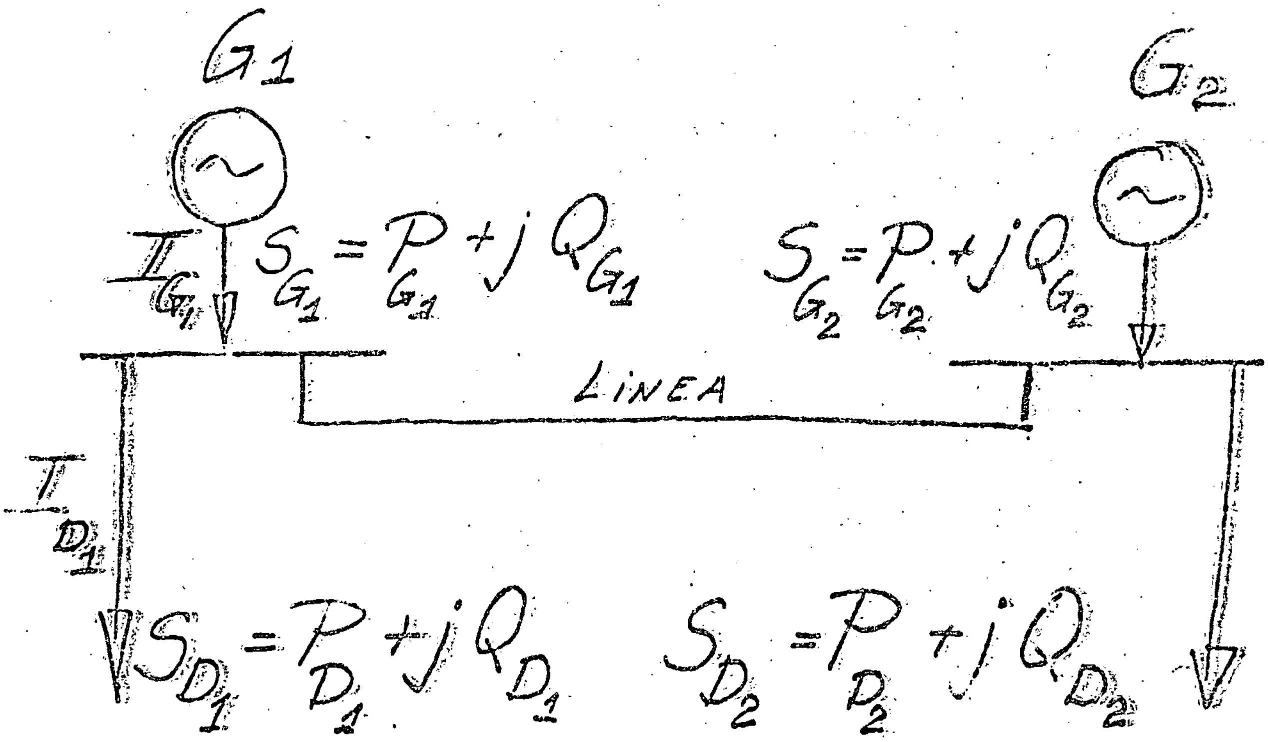
Función del Sistema: Suministrar

Potencia Activa P } que demandan
Potencia Reactiva Q } los usuarios
con Restricciones sobre:
Voltage V
frecuencia f

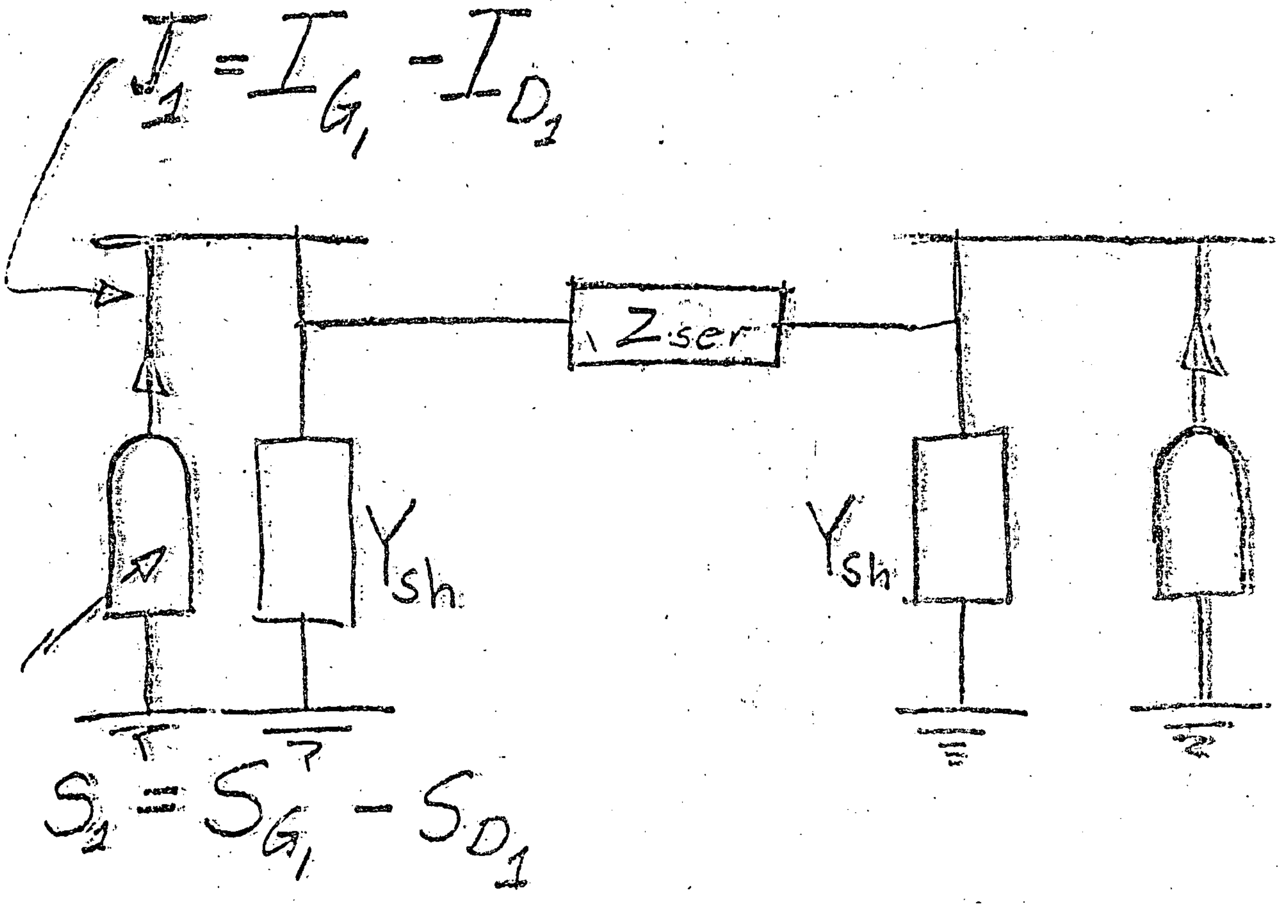
Problema de operación:
Modelar el sistema
Desarrollar estrategias de operación optima
Implementarlas (Control)

Con objeto de mantener el modelo lo más simple posible consideraremos que el sistema tiene dos generadores, dos consumidores conectados por

MODELADO DEL SISTEMA:



La línea se simula como una impedancia en serie con dos elementos en paralelo.



Para construir el modelo matemático correspondiente al modelo físico anterior es necesario incluir y definir diversas variables. Se define como potencia neta del bus (lugar donde se interconectan generadores cargas y líneas.

$$S_1 = P_1 + jQ_1$$

$$S_2 = P_2 + jQ_2$$

$$S_1 \triangleq P_{G_1} - P_{D_1} + j(Q_{G_1} - Q_{D_1})$$

$$S_2 \triangleq P_{G_2} - P_{D_2} + j(Q_{G_2} - Q_{D_2}) \quad (1)$$

Potencia Real Generada = Consumo + Pérdidas
(f = cst.)

Potencia Reactiva Generada = Consumo + Pérdidas
(V = cst.)

POTENCIA COMPLEJA:

$$S = V I^*$$

$$S^* = V^* I$$

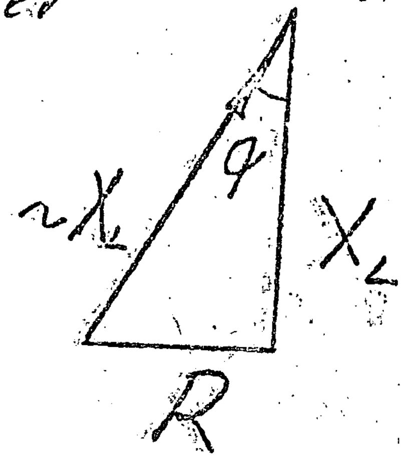
$$\frac{S_1^*}{V_1^*} = V_1 Y_{sh} + \frac{V_2 - V_1}{Z_{ser}} \quad (2)$$

otra para bus (2)

Simplificaciones:

$$Y_{sh} = \frac{j}{X_c}$$ capacitivo serie

$$Z_{ser} = R + j X_L \quad (4)$$

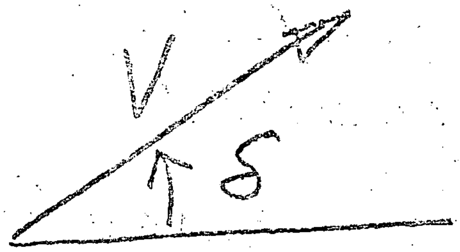


$$X_L \gg R$$

$$Z_{ser} \cong X_L \angle \frac{\pi}{2} - \varphi$$

Tensiones de Bus:

$$V_1 = |V_1| \angle \delta_1$$
$$V_2 = |V_2| \angle \delta_2 \qquad (5)$$



Sustituyendo (1); (3); (4) y (5) en (2) $\Longrightarrow$

Modelo

$$P_{G1} - P_{D1} - \frac{|V_1|^2}{X_L}\sin\alpha + \frac{|V_1|\,|V_2|}{X_L}\sin[\alpha - (\delta_1 - \delta_2)] = 0$$
$$P_{G2} - P_{D2} - \frac{|V_2|^2}{X_L}\sin\alpha + \frac{|V_1|\,|V_2|}{X_L}\sin[\alpha + (\delta_1 - \delta_2)] = 0$$
$$Q_{G1} - Q_{D1} + \frac{|V_1|^2}{X_c} - \frac{|V_1|^2}{X_L}\cos\alpha + \frac{|V_1|\,|V_2|}{X_L}\cos[\alpha - (\delta_1 - \delta_2)] = 0 \qquad (7\text{-}6)$$
$$Q_{G2} - Q_{D2} + \frac{|V_2|^2}{X_c} - \frac{|V_2|^2}{X_L}\cos\alpha + \frac{|V_1|\,|V_2|}{X_L}\cos[\alpha + (\delta_1 - \delta_2)] = 0$$

Características:

1) Ecs. algebraicas.

2) No lineales (Comp. digital)

3) Relacionan tensiones con potencia

4) No aparece f (Estado estable)

5) Siempre aparece

$$\delta_1 - \delta_2 = \delta_{12} = \delta$$

6) Doce variables

Cuatro ecuaciones

$$\Downarrow$$

Hay que pedefinir 8

Clasificacion de variables

a) Fuera de control ó disturbios

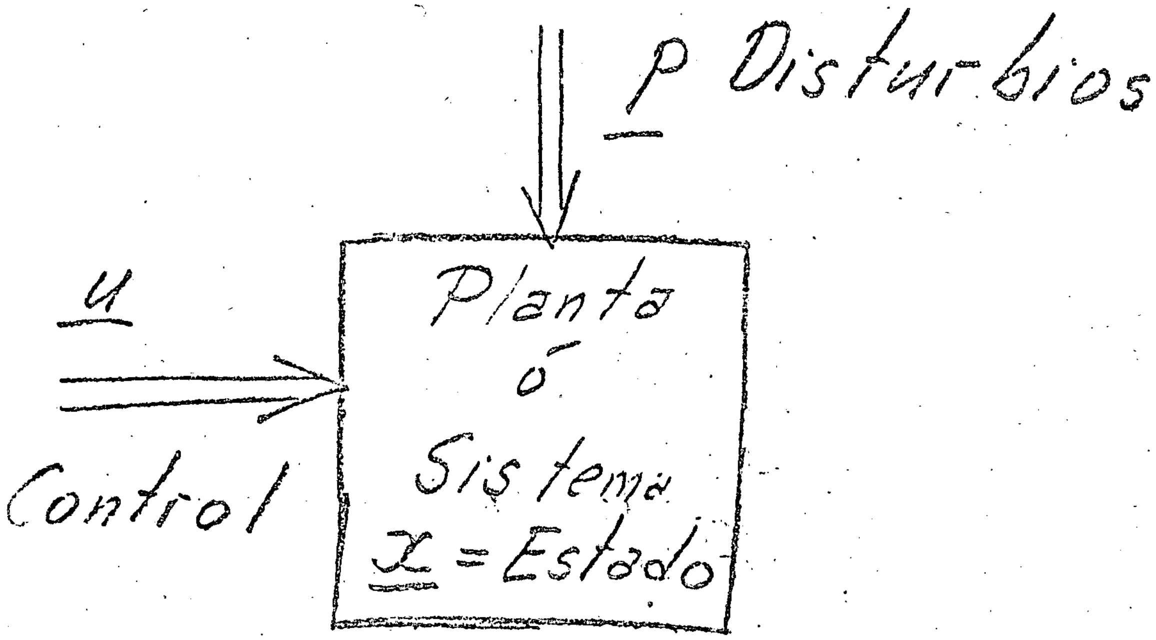
$$\underline{p} = \left[\begin{array}{l} P_{D_1} \\ Q_{D_1} \\ P_{D_2} \\ Q_{D_2} \end{array}\right.$$

b) Control o Manipuladas

$$\underline{u} = \left[\begin{array}{l} P_{G_1} \\ Q_{G_1} \\ P_{G_2} \\ Q_{G_2} \end{array}\right.$$

c) De Estado

$$\underline{x} = \left[\begin{matrix} \delta_1 \\ |V_1| \\ \delta_2 \\ |V_2| \end{matrix}\right.$$

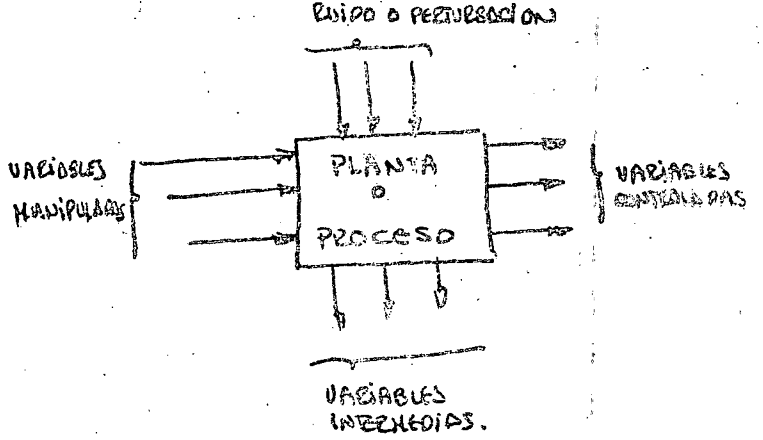


CONTROL DIGITAL DIRECTO

EXISTE UNA GRAN DISCREPANCIA ENTRE LA TEORIA DE CONTROL Y LAS APLICACIONES DE ESTA TEORIA. EN ESTA ULTIMA PARTE DEL CURSO NOS AVOCAREMOS AL PROBLEMA DE INCORPORAR LA TEORIA AL SISTEMA FISICO, UTILIZANDO COMO ELEMENTO DE CONTROL UNA COMPUTADORA DIGITAL.

1.- EL PROBLEMA DE CONTROL DE PROCESOS.

EXISTEN UNA INFINIDAD DE CASOS EN LOS CUALES EL PROCESO DE CONTROL SE LLEVA ACABO CON ENTO ① y ②. EN TODO PROCESO EXISTEN LOS SIGUIENTES TIPOS DE VARIABLES



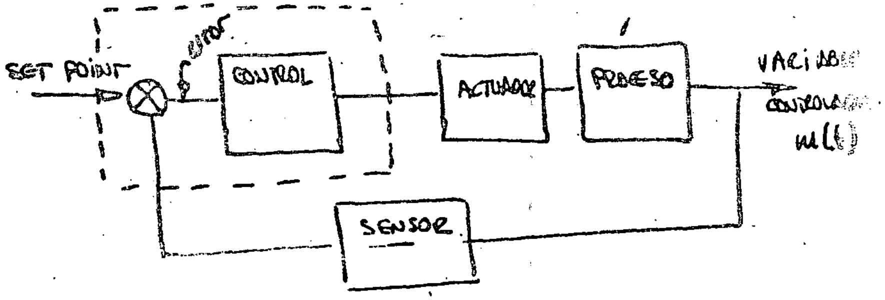
1.- VARIABLES MANIPULADAS.— ESTAS VARIABLES SON LA ENTRADA A NUESTRO PROCESO: VAPOR, AGUA, MATERIA PRIMA ETC. Y CUYOS VALORES DE ENTRADA NECESITAN SER MANIPULADOS

2.- RUIDO O PERTURBACION.— ESTOS VARIABLES AFFECTAN LA OPERACION DEL PROCESO Y NO ESTAN SUJETOS A CONTROL ie = ~~TEMPERATURA~~ VARIA ~~DE TEMPERATURA~~, IMPUREZA EN EL MATERIA DE ENTRADA

3.- VARIABLES CONTROLADAS.— ESTAS VARIABLES DEBEN MANTENERSE DENTRO DE UN RANGO, BLANCO, ETC, ALGUNAS VECES LLAMADO "SET POINT", EL PROBLEMA DE CONTROL CONSISTE EN MANTENER ESTAS VARIABLES DENTRO DE SU SET POINT O RANGO DE ACCION

4.- VARIABLES INTERMEDIAS.— ESTAS VARIABLES APARECEN EN ALGUN PUNTO INTERMEDIO EN EL PROCESO, SON UTILIZADAS PARA DETERMINAR FUTURAS ACCIONES DE CONTROL.

SISTEMAS DE CONTROL CONVENCIONALES.



① KORN MINICOMPUTERS FOR SCIENTIST AND ENGINEERS MCGRAW HILL PAG 216 — 220.

② F. COURY A PRACTICAL GUIDE TO MINICOMPUTER APLICATIONS IEEE (COMO EL LIBRO)

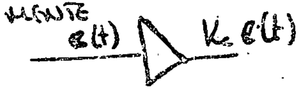
COMO YA SE VIO EN CAPITULOS PASADOS EXISTEN TRES ACCIONES BASICAS DE CONTROL (Y SUS COMBINACIONES), ESTAS ACCIONES SON:

CONTROL PROPORCIONAL

$$m(t) = K_c\, e(t)$$

K_c = GANANCIA PROPORCIONAL

QUE USUALMENTE SE UTILIZA UN AMPLIFICADOR PARA CONTROLAR PROPORCIONALMENTE

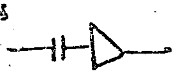


CONTROL DERIVATIVO

$$m(t) = K_c T_d \frac{de(t)}{dt}$$

T_d = TIEMPO DE DERIVACION

ESTE CONTROL SE CONOCE TAMBIEN COMO CONTROL "ANTICIPATIVO". POR SI SOLO CASI NUNCA SE UTILIZA PUES CAUSA INESTABILIDAD EN LOS SISTEMAS



CONTROL INTEGRAL

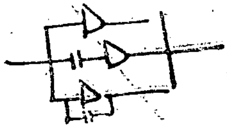
$$m(t) = \frac{K_c}{T_i}\int_0^t e(\sigma)\, d\sigma$$

T_i = TIEMPO DE INTEGRACION O "RESET"

CONTROL PID

$$m(t) = K_c\left\{ e(t) + \frac{1}{T_i}\int_0^t e(\sigma) + \frac{T_d\, de(t)}{dt} \right\} + m_r$$

LOS AJUSTES K_c, T_i, T_d, Y m_r QUE ES EL VALOR DE REFERENCIA SE ENCUENTRAN EN EL CONTROL Y ~~PUEDE~~. LOS VALOR DE K_c T_i, T_d ~~[illegible]~~ SON AJUSTADOS POR PRUEBA Y ERROR



EN GENERAL EL 75% DE LAS APLICACION UTILIZAN CONTROLES PI DEBIDO A LA DIFICULTAD DE AJUSTAR ~~[illegible]~~ EL CONTROL PID

EN PLANTAS EXISTEN DESDE UNOS CUANTOS ~~[illegible]~~ CONTROLES PI, HASTA MILES DE ESTOS. DESPUES DE LOS 50 LOS CONTROLES NEUMATICOS FUERON CAMBIADOS POR CONTROLES ELECTRICOS = [III] ELECTRONICOS, Y EN LOS 70 POR MINICOMPUTADORAS.

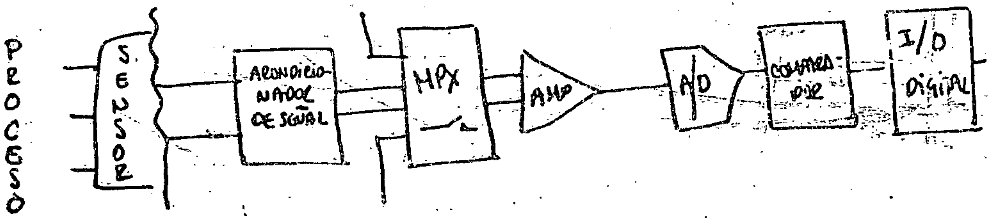
HAY QUE HACER NOTAR QUE EL EQUIPO AUXILIAR PUEDE LLEGAR A ~~EXCEDER~~ SER HASTA EL 80% DEL COSTO TOTAL DEL SISTEMA.

INTERFASES

PARA QUE EL PROCESO FUNCIONE CORRECTAMENTE LA COMPUTADORA DEBE RECIBIR DATOS DEL PROCESO Y MANDAR "OTROS DATOS" AL PROCESO, ESTOS DATOS PUEDEN SER:

1.- SEÑALES CONTINUAS (DATOS ANALOGICOS)
2.- DATOS DISCRETOS EN 2 NIVELES (ON-OFF)
3.- PULSOS

LAS SEÑALES CONTINUAS TIENEN EL SIGUIENTE ARREGLO TIPICO



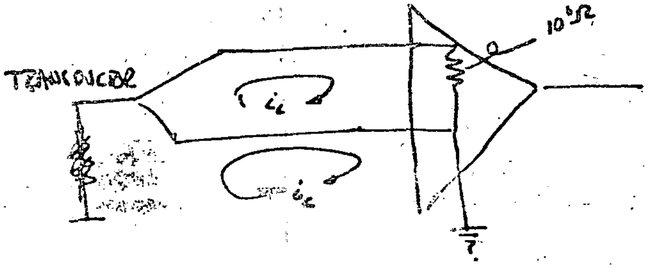
LAS SEÑALES SE CLASIFICAN COMO SIGUE:

1.- DE "BAJO NIVEL" (≤ 100 MICROVOLTS μV) ESTAS SEÑALES SE RECIBEN DE TERMOPARES, TERMÓMETROS DE RESISTENCIA, MEDIDORES DE COMPRESIÓN O TENSIÓN ETC.

2.- DE "ALTO NIVEL" (≥ 100 μV) ESTAS SEÑALES PROVIENEN DE TRANSDUCTORES A LOS CUALES SE TIENEN UN AMPLIFICADOR

CABE HACER NOTAR QUE LOS TERMOPARES SON EL ELEMENTO MAS POPULAR EN EL CONTROL DE PROCESOS, ESTOS ELEMENTOS MANDAN SEÑALES DE BAJO NIVEL (80 μV APROX); Y ESTAN SUJETOS A TODO TIPO DE RUIDO O DISTORSIÓN Y REQUIEREN ESPECIAL CUIDADO. LAS PUNTAS DEL TERMO-PAR EN GENERAL ESTAR RETORCIDAS Y PROTEGIDAS, ESTAS NO DEBEN ESTAR CERCA DE CIRCUITOS A-C O MOTORES Y GENERADORES GRANDES. SE DEBE TENER ESPECIAL CUIDADO EL EL ATERRIZAJE ~~PE~~ SE RECOMIENDA HACER EL ATERRIZAJE EN UN PUNTO Y ESTE ES LA COMPUTADORA

AUNQUE LA IMPEDANCIA EN LOS AMPLIFICADORES ES MUY GRANDE ($10^6 \Omega$), MUY POCA O NADA CORRIENTE RECIRCULADA, SIN EMBARGO ES CONVENIENTE HACER 2 ATERRIZAJE UNO EN LA COMPUTADORA Y OTRO EN EL TRANSDUCTOR



ACONDICIONAMIENTO DE LA SEÑAL .— CUANDO LA SEÑAL DEL TRANSDUCTOR ES UNA SEÑAL DE VOLTAJE, EL ACONDICIONADOR DE SEÑALES ~~ES~~ ES UN FILTRO RC. SI LA SEÑAL ES OTRA ES DE ACONDICIONAR. EN GENERAL SE TRANSFORMA EN UN VOLTAJE ANTES DE ENTRAR AL MULTIPLEXOR.

MULTIPLEXOR .— EL MULTIPLEXOR ES UN MECANISMO EL CUAL CONECTA UNA DE DE VARIAS SEÑALES DEL CONVERTIDOR A/D A LA COMPUTADORA (O VICEVERSA). PARA LAS SEÑALES DE ALTO NIVEL SE UTILIZAN MULTIPLEXORES ELECTRONICOS AUNQUE CON MUESTREOS MAYORES DE 10,000 PUNTOS POR SEGUNDO. PARA ~~LAS~~ SEÑALES BAJAS SE UTILIZAN RELAYS DE MERCURIO, PUES LA DISTORSION ~~DEL~~ EFEC DE CAMPO DE LOS TRANSISTORES NO PUEDE SER TOLERADO, LA RELACION DE ~~MUESTREO~~ DE ESTOS ULTIMOS ES DE APROXIMADAMENTE 200 PUNTOS/SEG.

LOS MULTIPLEXORES TIENEN DE 32 A 2024 PUNTOS O PUERTOS, Y SU ~~ESTO~~ MUESTREO ES SECUENCIAL. (ES MEJOR QUE EL MPX MANDE EL MUESTREO QUE EL CPU).

AMPLIFICADORES. LOS AMPLIFICADORES "ESCALAN" LA SEÑAL DEL PROCESO (+ o -) CON LA DEL CONVERTIDOR A/D (TIPICAMENTE 15 VOLTS.)

CONVERTIDOR A/D .— TRANSFORMAN UNA SEÑAL CONTINUA (ANALOGICA) EN UNA SEÑAL DIGITAL (DISCRETA). LA RESOLUCION DE UN CONVERTIDOR A/D ESTA RELACIONADA CON EL NUMERO "N" DE BITS DE LA COMPUTADORA DIGITAL

$$\text{RESOLUCION} = \frac{1}{2^{n} \; 1}$$

PARA N=11 BITS LA RESOLUCION ES APROX ≈0.05%
LO CUAL ES MUY ACEPTABLE PARA CASI TODAS LAS APLICACIONES

EL TIEMPO PARA QUE LA SALIDA DIGITAL DEL CONVERTIDOR A/D ALCANSE UN VALOR CTE DESPUES DE QUE FUE APLICADA UNA NUEVA SEÑAL SE LLAMA "TIEMPO DE ASENTAMIENTO" LOS CONVERTIDORES A/D ELECTRONICOS TIENEN UN TIEMPO DE ASENTAMIENTO ≤ 40 μseg.

<u>COMPARADOR</u>.– EL COMPARADOR LE QUITA CHAMBA AL CPU EVITANDO QUE SE DISTRAIGA HACIENDO TAREAS QUE PUEDEN SER EJECUTADAS POR FUERA. COMPARA LA SEÑAL DE ENTRADA CON UNA SEÑAL LIMITE (HIGH, Ó LOW).

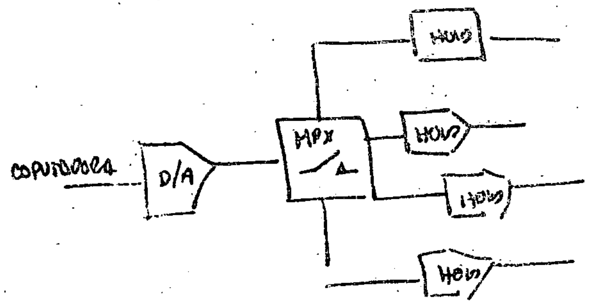
LOS COMPARADORES RESULTAN MUY UTILES EN SISTEMAS QUE HACEN MUESTREOS SECUENCIALES Y QUE TIENEN ~~PRE~~ CANALES DE ACSESO (?) DIRECTO DE MEMORIA, PUES GUARDAN LOS DATOS EN LOCALIDADES PRE ASIGNADAS DE MEMORIA.

SI LA SEÑAL SALE DE LOS LIMITES (HIGH Ó LOW) SE LE LLAMA LA ATENCION AL CPU ATRAVEZ DE UNA INTERRUPCION.

LA SALIDA DE DISPOSITIVOS TALES COMO TACOMETROS, Y VEL EN TURBINAS ES USUALMENTE EN FORMA DE PULSOS. AUNQUE LA COMPUTADORA PUEDE CONTAR PULSOS, ESTO CONSUMIRIA TIEMPO DE CPU, POR LO QUE SE UTILIZAN CONTADORES DE PULSOS EXTERNOS. EL CPU CARGA ALGUN REGISTRO CON EL NUMERO DE PULSOS CONTADOS POR EL CONTADOR.

LA SALIDA DE LA COMPUTADORA AL PROCESO

1.– CONVERTIDOR D/A.– ESTE APARATO CONVIERTE LA SEÑAL DIGITAL EN UNA SEÑAL ANALOGICA. USUALMENTE SE UTILIZA UN MPX PARA OBTENER VARIAS SALIDAS DE UN CONVERTIDOR D/A. UTILIZA UN "HOLD" PARA RETENER EL VALOR ENTRE MUESTRAS



2.– GENERADORES DE PULSOS, LOS CUALES GENERAN UN NUMERO ~~ESPECIFICO~~ DE PULSOS ESPECIFICADO POR LA COMPUTADORA. ES PULSOS TIENES AMPLITUD Y DURACION PREDETERMINADA. LA SALIDA DE ESTOS GENERADORES DE PULSOS ES USUALMENTE UTILIZADA COMO MANDO EN SERVOMOTORES

3.– CONTACTOS ON-OFF ~~(PARA~~ SON UTILIZADOS PARA ARRANCAR O PARAR BOMBAS, MOTORES ETC, ADEMAS SIRVEN PARA OBTENER PULSOS DE DURACION VARIABLE.

LOS USOS DE LOS CONVERTIDORES A/D Y LOS GENERADORES DE PULSOS SE PUEDEN APRECIAR EN LAS SIGUIENTES 3 FIGURAS

INTERRUPCIONES. EL PROPOSITO DE UNA INTERRUPCION ES EL ALTERAR EL FLUJO NORMAL DE INSTRUCCIONES EN UNA COMPUTADORA, PARA ATENDER ALGUNA FUNCION URGENTE DE MAYOR PRIORIDAD

1.- INTERRUPCIONES DEL SISTEMA.— ESTE TIPO DE INTERRUPCIONES SON CAUSADAS POR EL MISMO SISTEMA, POR EJEMPLO FALLA DE PAGINA, ~~ETC~~ PEDIR MAS DATOS PARA IMPRESION, ETC.

2.- INTERRUPCIONES DEL RELOJ. ~~ESTE TIPO DE OPERACIONES~~ ~~SINCRONIZAN~~ CUANDO SE REQUIERE UNA SINCRONIA ENTRE LAS OPERACIONES DEL SISTEMA Y EL MUNDO EXTERIOR. LAS INTERRUPCIONES DEL RELOJ SE DAN A ~~TIEMPOS~~ INTERVALOS DE TIEMPOS REGULARES. (IE CALCULO DE ALGUN ALGORITMO EN INTERVALO DE TIEMPO IGUALES, MUESTREO DE ALGUN DISPOSITIVO, ETC).

3.- INTERRUPCIONES DEL PROCESO — ESTAS SE ORIGINAN EN EL PROCESO, BAJO CONDICIONES ANORMALES O DE ALARMA Y REQUIEREN DISTRAER INMEDIATAMENTE LA ATENCION DE CPU, PARA LLEVAR ACABO ALGUNA TAREA ESPECIFICA EN CONDICIONES ESPECIALES

EN LOS ~~ESE~~ SISTEMAS EN QUE SE UTILIZA CDD EL SISTEMA DE INTERRUPCIONES JUEGA UN PAPEL IMPORTANTISIMO EN EL PROCESO.

CUANDO UNA INTERRUPCION OCURRE LOS SIGUIENTES EVENTOS SUCEDEN.

1.— OCURRE (SE DISPARA) UNA INTERRUPCION

2.- NO SE EJUTA LA SIGUIENTE INSTRUCCION, SINO QUE EL CONTROL ES DESVIADO A ALGUNA LOCALIDAD EN LA MEMORIA CENTRAL, Y LA INSTRUCCION AHI [illegible] ES EJECUTADA.

3.- SI SE REQUIERE LA EJECUCION DE VARIAS INSTRUCCIONES, LA INSTRUCCION EJECUTADA DEBIDO A LA INTERRUPCION ES UNA INSTRUCCION ESPECIAL, QUE GUARDA EL CONTENIDO DEL REGISTRO DE DIRECCION Y CARGA EN EL REGISTRO DE DIRECCION LA SIGUIENTE INSTRUCCION A SER EJECUTADA. LA INSTRUCCION LOCALIZADA EN EL REG. DE DIRECCION ES LA PRIMERA INSTRUCC. DE UNA INSTRUCCION LLAMADA "RUTINA DE SERVICIO DE INTERRUPCION" ~~SE~~ (ISR), UNA VEZ SATISFECHA LA INTERRUPCION SE, SE RESTAURAN LOS CONTENIDOS DE LOS REGISTROS DE TRABAJO (LOS) Y EL CONTENIDO DEL REGISTRO DE DIRECCION (RD) AL VALOR QUE TENIAN CUANDO LA INTERRUPCION SUCEDIO.

2.1 Introducción

*Los complejos sistemas de interés para el analista de sistemas están formados por múltiples partes o subsistemas. □Además por muy grande y complejo que sea el sistema en estudio, éste a su vez forma parte de otro sistema todavía más grande y de mayor complejidad. Todo análisis de sistemas debe tomar en cuenta cuál es la posición del subsistema dentro del sistema que lo incluye y cuáles son las partes que lo forman. *Estas relaciones entre subsistemas con un sistema más amplio que los incluye, frecuentemente son de una naturaleza jerárquica. En esta sección se estudian diversos tópicos relacionados con este tema.

***Todo sistema está formado por partes o subsistemas.**

□Todo sistema es parte de un sistema mayor.

***Entre los subsistemas de un sistema hay relaciones jerárquicas.**

La *configuración* estructural conocida con el nombre de *jerárquica* o de nivel múltiple es muy importante en sistemas de diversa índole, como pueden ser por ejemplo los de organización o los de maquinaria y equipo.

*Resulta importante determinar la estructura y jerarquía de un sistema y los niveles dentro de la jerarquía que corresponden a cada parte integrante del mismo, ya que las variables asociadas a cada subsistema y las funciones que realiza, que fijan sus características de operación que trata de analizar o determinar el analista, dependen de su *nivel jerárquico* dentro del sistema general como se señala posteriormente. *Además la operación de un sistema depende en forma importante de la coordinación que existe en el funcionamiento de las partes. *Esta coordinación entre las partes, que se basa en la información que recibe la unidad de coordinación o control, depende también de la estructura jerárquica de todo el sistema y del nivel que ocupa dentro de esa jerarquía el sistema en estudio.

***Determinar:**
Estructura
Niveles

***La operación conjunta de un sistema depende de la coordinación entre los subsistemas**

***La coordinación entre subsistemas se basa en la información.**

*En resumen, resulta imposible analizar un número importante de sistemas si se desconoce su estructura jerárquica y la estructura jerárquica del sistema mayor del que éste a su vez forma parte.

***La jerarquización es indispensable en el análisis de ciertos sistemas.**

*A continuación se describirá la estructura jerárquica de la industria eléctrica de servicio público. El objetivo de esta descripción es ilustrar el concepto de estructura jerárquica y señalar la relación que existe entre los niveles jerárquicos a que corresponde un subsistema y la naturaleza de la información que maneja.

***Ejemplo de estructura jerárquica: industria eléctrica.**

Así mismo se ilustra la forma del control y la naturaleza de la información que debe manejarse para poder controlar y coordinar entre sí los diversos subsistemas de una estructura jerárquica.

*La industria eléctrica, como toda industria, tiene una estructura piramidal en la que es posible identificar un proceso físico y una función de control tal como muestra la fig. 2.1.1.

*Industria:
proceso físico + controlador

*La *función de control* manipula el proceso con el fin de alcanzar los objetivos de la industria, que en este caso son: obtener máxima confiabilidad, minimizar los gastos de operación y maximizar la generación.

*El controlador manipula al proceso con el fin de que la industria alcance sus objetivos.

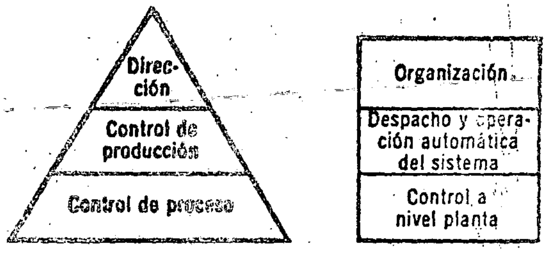


Fig. 2.1.1 Estructura jerárquica del control.

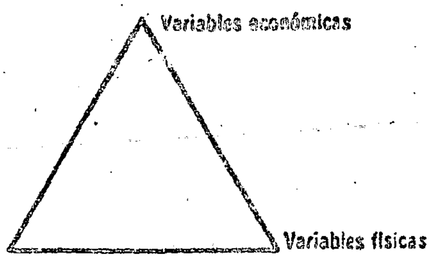
*Pueden distinguirse, en general, tres funciones de control a diferentes niveles. En el primer nivel están aquellas funciones asociadas con el control de las unidades de manufactura, que en el caso de la industria eléctrica corresponden a las plantas generadoras. En el segundo nivel, las funciones de control guían las actividades de producción mediante despacho de carga, operaciones de conexión, etc. En el último nivel, las funciones de control corresponden a la dirección empresarial e incluyen el establecimiento de objetivos para ser alcanzados con las restricciones del sistema.

*Tres funciones de control:
Dirección
Control de producción
Control de proceso

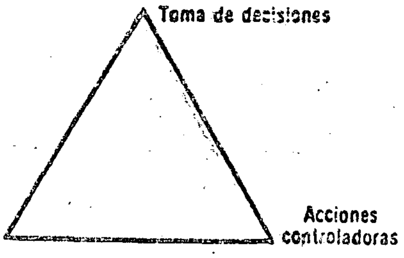
*Paralelamente a las jerarquías señaladas en el nivel de control, al ir hacia el vértice de la pirámide se puede identificar una jerarquía de funciones de control: regulación, optimización, adaptación y organización automática.

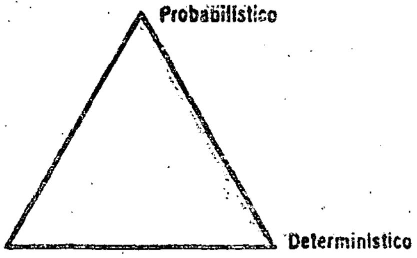
*Jerarquización de las funciones de control: regulación, optimización, adaptación y organización automática.

*Puede observarse que, a medida que se avanza hacia la cúspide, el énfasis en las variables físicas disminuye, y aumenta la importancia de las variables económicas en el proceso de toma de decisiones o funciones de control. El control de las unidades generadoras mediante gobernadores y reguladores se basa, exclusivamente, en variables físicas, mientras que al nivel de control de producción, el despacho económico se realiza en función de variables físicas y económicas.



*Otra característica del control de sistemas es la decreciente frecuencia de las acciones controladoras y la creciente complejidad del proceso de toma de decisiones al ascender a través de la jerarquía de control. En la industria eléctrica, dentro del primer nivel de control, los reguladores y generadores operan en forma continua y basan su acción, fundamentalmente, en mediciones de tensión y velocidad. En el segundo nivel, las acciones de control se realizan bajo crecientes condiciones de incertidumbre. *Debe anotarse también que, dentro del primer nivel, los problemas de control son determinísticos, mientras que se vuelven crecientemente probabilísticos al ascender a través de la jerarquía del sistema de control.





*Todos estos controles, ya sean máquinas o seres humanos, son procesadores de información. Reciben información sobre el estado del sistema y, en función de ésta y del conocimiento de los objetivos del sistema y sus restricciones, ejecutan acciones controladoras. *Como se ha señalado en los párrafos anteriores el tipo de acción de control que debe ejercerse depende del nivel jerárquico al que se encuentra el subsistema en estudio. También depende del nivel jerárquico, la naturaleza de la información (probabilística o determinística) que manejan los controladores de sistema.

***Los controladores procesan información.**

***La acción de control depende del nivel jerárquico, así como la naturaleza de la información.**

La descripción anterior ha servido para introducir al lector al problema de la jerarquización de un sistema y señalar su importancia.

En la siguiente sección se describen diversas clases de jerarquización: de nivel, tiempo y modo. Posteriormente se introduce un algoritmo para estudiar problemas de jerarquización.

El capítulo termina señalando la coordinación de información que debe existir entre los elementos de un sistema, con objeto de que todas sus partes operen en forma coordinada para alcanzar los objetivos operacionales del sistema.

2.2. Clases de subdivisiones en la jerarquización de sistemas

Siempre que se analice un sistema es necesario tener presente que éste es a su vez, parte de un sistema mayor. *De ahí que el propósito de la jerarquización es el de ayudar a determinar qué

***Todo sistema es, a su vez, parte de un sistema mayor.**

relación guarda un sistema con aquellos con los que interacciona. Es decir, saber cuál elemento o subsistema está subordinado a otros, y cómo.

*La forma de jerarquizar los sistemas puede ser muy variada, por lo que en esta sección únicamente se discutirán tres clases de subdivisiones: *de nivel, de tiempo y de modo*. Puede considerarse que éstas son las más importantes en sistemas de gran tamaño.

*Jerarquización { de Nivel
de Tiempo
de Modo

2.2.1 Subdivisiones jerárquicas de nivel

Estas subdivisiones usualmente se basan en consideraciones geográficas, de espacio, por lo general implican descentralización, o *conservación de la autonomía hasta donde sea posible. Considérese, al respecto, el ejemplo de un sistema eléctrico de potencia subdividido en tres niveles:

Nivel 1 Plantas generadoras

Nivel 2 Sistemas individuales

Nivel 3 Sistema interconectado

*Subdivisión de nivel
Consideraciones:
.geográficas
.de espacio
.de autonomía

La fig. 2.2.1 muestra la subdivisión del sistema eléctrico de México (nivel 3). *El cual se halla constituido por seis sistemas mayores (nivel 2);

Sistemas Mayores
I) Sonora Sinaloa
II) Torreón Chihuahua
III) Falcón Monterrey
IV) Occidental
V) Central
VI) Oriental

y *dos sistemas menores

**Sistemas Menores*
a) Baja California
b) Yucatán

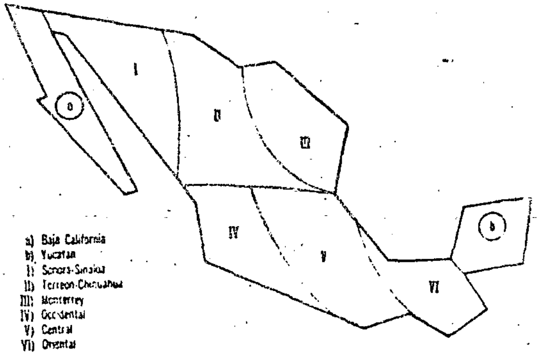


Fig. 2.2.1 Subdivisión, en sistemas regionales, de República Mexicana.

El conjunto de los sistemas mayores constituye el sistema eléctrico nacional interconectado que se esquematiza en la fig. 2.2.2.

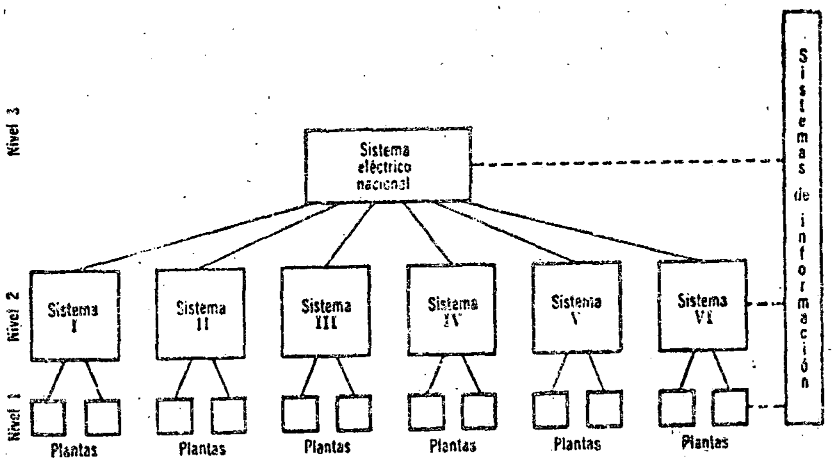


Fig. 2.2.2 Jerarquización del sistema eléctrico nacional por plantas, sistemas regionales y sistema interconectado.

En todos los sistemas existen plantas generadoras (nivel 1), termoeléctricas e hidroeléctricas, pudiendo contar cada una con una o varias unidades. Los seis sistemas mayores se encuentran débilmente interconectados, aun cuando hay planes para fortalecer los lazos de unión entre todos.

*Subdivisión del nivel por tensiones de transmisión.

*Otra subdivisión posible de nivel en los sistemas eléctricos de potencia, puede hacerse tomando como base el voltaje de transmisión (fig. 2.2.3). Por ejemplo una red con más de 230 kv, a la vez que interconecta los sistemas, conduce energía de las grandes plantas hidroeléctricas (que se encuentran por razones geográficas muy lejanas) a los centros de consumo.

Una serie de redes de distribución mayor (con voltaje entre 115 y 230 Kv), se utiliza para efectuar la distribución primaria de grandes cantidades de energía eléctrica, e integrar anillos de reparto de carga alrededor de grandes zonas urbanas. Por último, se emplean redes con voltajes menores de 115 Kv para la distribución final de la energía a los pequeños y medianos consumidores.

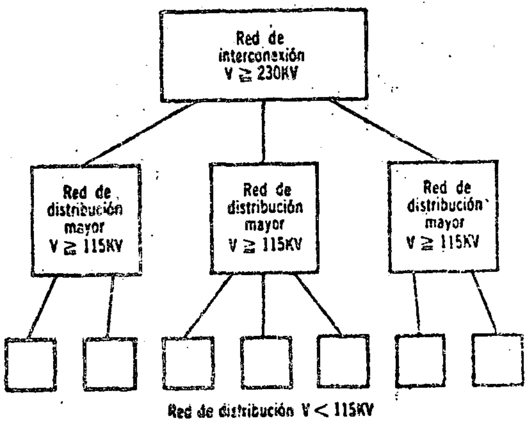


Fig. 2.2.3 Jerarquización del sistema eléctrico nacional por niveles de tensiones de transmisión

*Las subdivisiones de nivel no son exclusivas para los sistemas eléctricos de potencia, sino también son comunes a los sistemas educativos. Para representar estas subdivisiones jerárquicas es posible emplear figuras semejantes a las que se emplearon para

*Sistemas educativos.

sistemas eléctricos. En este caso, estas subdivisiones pueden hacerse tanto por razones geográficas (fig. 2.2.4) como de grado (fig. 2.2.5). Sin embargo, en estas figuras, se les ha representado con una estructura piramidal a fin de ilustrar la dependencia jerárquica que usualmente se representa en esta forma.

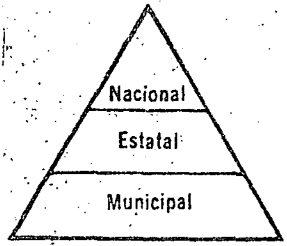


Fig. 2.2.4 Pirámide jerárquica administrativa de la educación

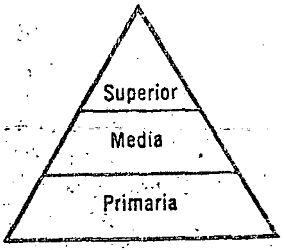


Fig. 2.2.5 Pirámide jerárquica de grado de la educación.

2.2.2. Subdivisiones jerárquicas de tiempo

*Estas subdivisiones surgen del amplio rango de *tiempos de respuesta* inherentes a muchos sistemas. Por ejemplo, en los de tipo educativo se nota una gran diferencia entre: el tiempo de respuesta del sistema a los cambios en la estructura social de un país (usualmente es de muchos años), el tiempo de respuesta a las diferencias entre el grado de educación de los diferentes niveles (usualmente de unos pocos años) y el tiempo de respuesta a las diferencias entre el grado de educación obtenido año con año (usualmente un año).

***Los tiempos de respuesta tienen diversos órdenes de magnitud.**

Como ejemplo de la subdivisión de tiempo en los sistemas eléctricos de potencia, considérense diversas funciones propias de estos sistemas, que junto con los tiempos en que se realizan, se muestran a continuación.

Planeación

*Consiste en determinar las necesidades del sistema durante los próximos años y tomar las medidas necesarias para satisfacerlas. Su escala de tiempo, es del orden de años.

*Años = Orden de tiempo de la planeación.

Despacho de unidades

*Asigna las unidades que estarán en operación durante las siguientes x horas ($x = 24$ horas), a fin de satisfacer de manera

*Días = Orden de tiempo del despacho de unidades.

apropiada la demanda. Su escala de tiempo es del orden de horas.

Despacho económico

*Señala qué parte de la generación comprende a cada unidad, de tal manera que el costo de generación sea mínimo. Su escala de tiempo es del orden de minutos.

*Minutos = Orden de tiempo del despacho económico.

Control frecuencia-carga

*Mantiene la frecuencia de generación del sistema lo más cerca posible de la frecuencia nominal de operación, con lo cual se logra armonizar la producción con el consumo. Su escala de tiempo es del orden de segundos.

*Segundos = Orden de tiempo del control de frecuencia-carga.

La fig. 2.2.6 muestra cómo dichas funciones se realizan en diferentes escalas de tiempo.

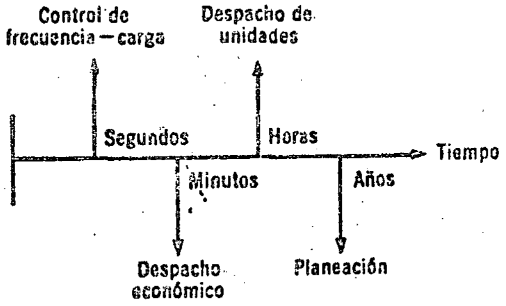


Fig. 2.2.6 Jerarquización por tiempo.

*La subdivisión de tiempo prácticamente tiene por objeto dividir el problema general (concerniente a todo el sistema) en problemas menores más fácilmente tratables.

*Subdivisión de tiempo: Simplifica el problema

La subdivisión de tiempo puede realizarse paralelamente con la subdivisión de nivel. En el caso de los sistemas eléctricos de potencia se tiene, por ejemplo, que el despacho económico se lleva a cabo en los sistemas individuales, la planeación se lleva a cabo en el sistema nacional, y el control de frecuencia-carga en las plantas.

2.2.3 Subdivisiones jerárquicas de modo

Tanto los sistemas educativos como los eléctricos de potencia deben ser capaces de trabajar bajo una gran variedad de condiciones: unas normales, otras de emergencia y otras preventivas.

Por ejemplo, en los sistemas eléctricos de potencia se presentan frecuentemente los siguientes modos de operación.

Modo normal

*Cuando el sistema se encuentra en estas condiciones, las necesidades de todos los clientes se satisfacen con la frecuencia y voltaje normales. Los objetivos que deben lograrse en el modo normal de operación son:

*Objetivo del modo normal

a) Mantener frecuencia
b) Mantener intercambios
c) Minimizar costos

a) Mantener la frecuencia igual a la frecuencia nominal;

b) Mantener los intercambios de energía con los sistemas vecinos dentro de los límites establecidos;

c) Efectuar la generación con el mínimo costo.

Modo preventivo

*La diferencia entre este modo de operación y el anterior es sutil. En principio, ambos son el mismo, y sólo cambia el valor esperado de la ocurrencia de una falla. En el modo normal, el valor esperado de ocurrencia de una falla es pequeño; en cambio, en el preventivo, es grande. El propósito de este modo de operación es tratar de evitar, mediante ciertas medidas preventivas, que el sistema tenga que pasar al modo de emergencia.

*Objetivos del modo preventivo

a) Mantener frecuencia
b) Mantener intercambios
c) Mantener reserva

Los objetivos que persigue el presente modo de operación son:

a) Mantener la frecuencia igual a la frecuencia nominal;

b) Mantener los intercambios de energía con los sistemas vecinos dentro de los límites establecidos;

c) Mantener cierta cantidad mínima de reserva rodante.

Modo de emergencia

*En este modo opera un sistema eléctrico de potencia cuando ha ocurrido una falla mayor y no es posible satisfacer la demanda de todos los clientes. En estos casos, los objetivos que se busca lograr son:

*Objetivos del modo de emergencia

a) Mantener frecuencia
b) Minimizar apagones

a) Mantener la frecuencia igual a la nominal;

b) Tratar de proveer a la mayor cantidad posible de clientes.

En comparación con el modo normal, el preventivo sacrifica parte de la economía por mantener una reserva rodante adecuada; y en el de emergencia, dicho sacrificio en economía es mayor y se hace para lograr satisfacer el número máximo de clientes.

Modo restaurativo

*Cuando el sistema ha tenido una falla grave (que ha obligado a emplear el modo de emergencia), es necesario reparar la falla e inmediatamente después, llevar al sistema otra vez al modo normal de operación. Los objetivos del modo restaurativo son:

a) Mantener la frecuencia igual a la nominal;

b) Llevar con la mayor rapidez posible el sistema a un estado tal, que satisfaga la demanda de todos los clientes.

La fig. 2.2.7 muestra los cuatro modos de operación de los sistemas eléctricos de potencia mencionados, así como la manera de efectuar las transiciones entre los diferentes modos.

*Objetivos del modo restaurativo.

a) Mantener frecuencia

b) Maximizar la velocidad de restauración

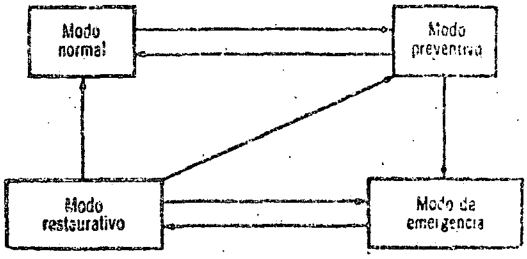


Fig. 2.2.7 Jerarquización de los modos de operación de un sistema eléctrico de potencia.

2.4. Coordinación e intercambio de información entre los elementos de un sistema

Una vez que un sistema se ha descompuesto en varios subsistemas es necesario para que el sistema opere coordinadamente que cada uno de estos subsistemas tenga cierta información relativa a los otros. La presente sección, trata sobre este intercambio de información, y de las fuentes de la información. Se señala además cómo ayuda este intercambio a la coordinación en la operación y de las implicaciones que tiene en la estructura general del sistema.

2.4.1 Fuentes y formas de información

*Existen, básicamente, dos *tipos de información*:

a) Numérica

b) De estructura.

*Tipos de información { Numérica
De estructura

*Por información numérica se entienden los valores de parámetros y variables de estado, y por de estructura, el conocimiento de la forma e interconexiones del sistema. *Por ejemplo, en un sistema eléctrico de potencia pueden considerarse como parámetros: la inercia de los generadores, la impedancia de las líneas de transmisión, el precio del combustible, etc. En sistemas educativos los parámetros pueden ser: la localización de los centros de educación, el presupuesto anual disponible, etc.

*Información numérica { Parámetros
variables de estado.

*Parámetros de sistemas eléctricos:
inercia de generadores
impedancia de líneas, etc.

*Como variables de estado en sistemas eléctricos de potencia se pueden citar: voltaje en los nodos, corrientes en las líneas, potencias generadas, pérdidas, etc. En los sistemas educativos entre las variables de estado pueden anotarse: número de alumnos de cada grado, número de profesores disponibles, deserción y admisión.

*Variables de estado en sistemas eléctricos: voltaje de nodos, corriente en las líneas, potencias, etc.

*Ejemplo de información de estructura en sistemas eléctricos de potencia serían: la topología de la red, estructura del sistema de control, mapa de carga, etc. *Como ejemplos de información de estructura en sistemas educativos se tiene: mecanismo de transferencia de alumnos de un grado a otro, la distribución geográfica de la demanda, etc.

*Estructura en sistemas eléctricos:
Topología
Mapa de carga

*Estructura en sistemas educativos:
Transferencia de grado
Distribución geográfica

En el cuadro de la figura 2.4.1 se sumarizan los diferentes tipos de información.

TIPOS DE INFORMACION	NUMERICA	Variables de estado
		Parámetros
	DE ESTRUCTURA	

Fig. 2.4.1 Tipos de información.

*De acuerdo con la forma, la información puede clasificarse en:

a) Inherente

b) Disponible de inmediato.

*Formas de información

Inherente
Disponible

A continuación se analizan estos tipos de información. *Si se cuenta con la información pero ésta no se puede usar de inmediato ésta recibe el nombre de inherente. Por ejemplo, supóngase que en un sistema eléctrico de potencia se han colocado medidores de corriente y voltaje en ciertas líneas, y que la configuración del sistema es tal que puede calcularse, a partir de los valores medidos, la potencia en las líneas restantes. Esta información es de tipo inherente, ya que es necesario realizar cálculos para poder obtenerla.

*La información inherente debe procesarse antes de usarse.

Cuando en un sistema educativo se conoce el volumen de nuevos ingresos y el de transferencias entre los diferentes grados, es posible conocer los índices de deserción, los cuales constituyen ejemplos de información inherente, ya que no se encuentran inmediatamente disponibles, hay que calcularlos.

La técnica conocida con el nombre de "estimación de estado" (ref. 5) es útil en el proceso de transformar información inherente a forma disponible inmediata; también lo son en el mismo proceso el filtrado estadístico de datos y las técnicas de estimación en general.

*En un sistema subdividido por una jerarquización la información asociada a cada subsistema puede provenir de dos fuentes:

a) Directamente del propio subsistema por mediciones o estimaciones en él.

b) De otros subsistemas. (Entre los diferentes subsistemas se transfiere información mediante una red de comunicaciones).

*Fuentes de Información
Medida en el sistema
Provenientes de otros sistemas

Cabe aclarar que aun cuando no existiera una red dedicada expresamente a la comunicación entre los diferentes subsistemas, uno de ellos puede obtener información inherente de los otros por mediciones internas. Recuérdese que, a menos que los subsistemas se encuentren completamente desconectados (independientes), siempre existe una dependencia mutua.

2.4.2 Información e incertidumbre

Es razonable pensar que sólo se tiene cierto grado de certidumbre sobre la información. Por ejemplo, ¿hasta qué punto puede confiarse en las lecturas obtenidas con los sensores?, ¿cuál es el grado de error que introducen los canales de telemetría?, ¿qué tan confiables son los censos y las encuestas?.

*Para designar el inverso de la cantidad de información se utiliza la palabra *incertidumbre*.

*Incertidumbre: antítesis de información

*Existen dos maneras básicas para expresar la incertidumbre:
a) mediante fronteras
b) probabilísticamente

*Expresión de ↕ Incertidumbre { Fronteras / Probabilidad

*Se dice que la incertidumbre se expresa mediante fronteras, cuando se desconocen los valores exactos de ciertas variables, pero se sabe que deben estar entre ciertos límites (fronteras). Por ejemplo, no se sabe con exactitud el número de alumnos que demandarán admisión en una escuela, pero sí que serán entre 8 000 y 10 000.

*Incertidumbre por fronteras → límites en los valores

*Expresar la incertidumbre por medios probabilísticos se utiliza cuando no se conoce el valor de una variable, pero se sabe que tiene una cierta función de densidad de probabilidad**. Por ejemplo, se ignora la demanda de energía de un sistema, pero se sabe que tiene una distribución gausiana con media 240 MW y desviación estándar de 5 MW.

*Incertidumbre por probabilidad → Probabilidad de los valores

En la fig. 2.4.2 se muestra un resumen de los medios para expresar la incertidumbre.

Incertidumbre en parámetros y Var. de Edo.	Incertidumbre en estructura
Los parámetros y variables de estado pueden tomar cualquier valor entre ciertos límites.	Una serie de modelos con ciertos modelos como casos extremos.
Los parámetros y variables de estado son variables aleatorias con cierta distribución.	Modelos con características y probabilísticas.

Fig. 2.4.2 Medios de expresar la incertidumbre.

*Se señaló anteriormente que con frecuencia es necesario realizar ciertos cálculos con la información para convertirla de inherente a disponible. Estos cálculos pueden reducir el nivel de incertidumbre de la información.

*Cálculos pueden reducir el nivel de incertidumbre.

**Estos conceptos se definen en el apéndice B, sección B.2

2.4.3 Información, coordinación y control.

*Cuando se toman una serie de medidas para que un sistema alcance ciertos objetivos, se dice que se le está controlando.

*Controlar para alcanzar objetivo.

El propósito de esta sección es establecer la relación que existe entre el control y la coordinación de la información. Considere al respecto, un sistema compuesto por dos escuelas fig. 2.4.3.

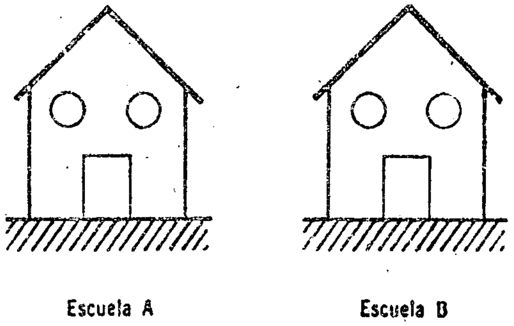


Fig. 2.4.3 Dos escuelas.

El costo por alumno para cada escuela se muestra en la fig. 2.4.4.

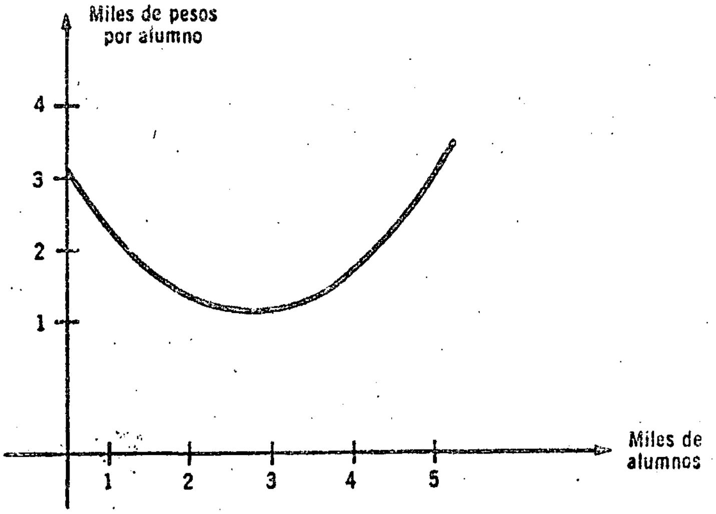


Fig. 2.4.4 Costo por alumno como función de la población.

*Supóngase que las escuelas A y B se encuentran en el mismo vecindario y que en el plantel A se inscriben 2 000 alumnos y al B 4 000. Si ambas no coordinan información, operarán con un costo total de:

*Plantel A 2 000 alumnos
Planta B 4 000 alumnos
Costo si no hay intercambio de información:

2 000 × 1 250 + 4 000 × 1 500 = 8 500 000

Si intercambian información y deciden que la escuela con 4 000 alumnos transfiera 1 000 a la que tiene menos, ambas operarán con un costo de:

3 000 × 1 000 + 3 000 × 1 000 = 6 000 000

*Como se ve en el ejemplo anterior, cuando los diferentes subsistemas tienen el mismo fin, es conveniente que exista una gran

*Coordinación
Aumenta eficiencia
(Disminuye costos)

coordinación entre ellos. Esta coordinación se basa en el intercambio de información, y aumenta la eficiencia del sistema.

*Hay dos maneras de coordinar sistemas, las cuales se muestran en las figs. 2.4.5 y 2.4.6 apreciándose la diferencia entre el método de intercambio de información directa y mediante el *centro de información,* respectivamente.

*2 formas de coordinar sistemas

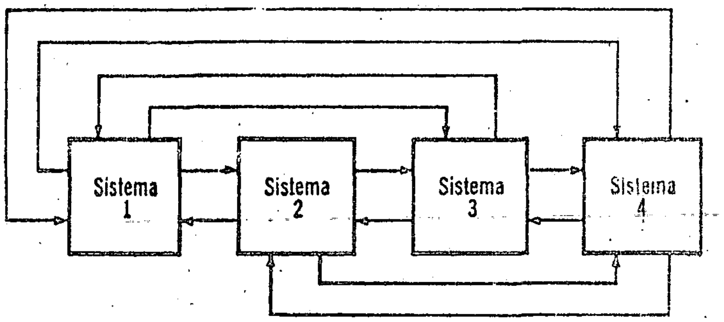


Fig. 2.4.5 Intercambio de información directa.

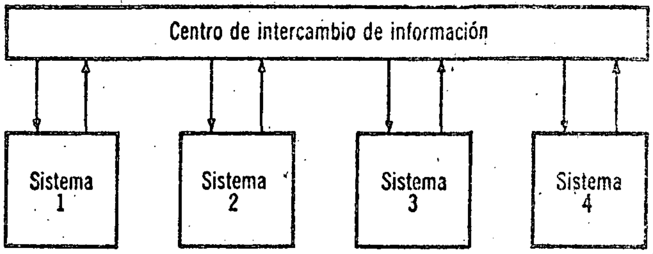


Fig. 4.2.6 Intercambio de información mediante centro de información.

*El método de intercambio de información directa consiste en contar con una red de comunicaciones que conecta, uno con otro, todos los subsistemas.

*Intercambio directo
Una red comunica todos los subsistemas.

*El método de centro de intercambio de información consiste en crear un subsistema que está comunicado con todos los demás y se encarga de coordinar los intercambios de información.

*Con el centro de intercambio un subsistema se encarga de la comunicación.

*Dicho método suele ser más apropiado para sistemas jerarquizados, ya que usualmente el sistema de mayor jerarquía toma a su vez el papel de centro de intercambio de información. Sin embargo, esto no es necesariamente cierto; puede existir una jerarquía en el sistema de coordinación de información, y ésta ser completamente independiente de la del sistema principal.

*El método de intercambio información
se utiliza principalmente en sistemas jerarquizados.

El grado de coordinación de un sistema puede variar desde sistemas no coordinados a sistemas completamente coordinados.

*En un sistema no coordinado, la falla en uno de los subsistemas no implica falla alguna en los demás, ya que en este caso los subsistemas están desconectados. Estos sistemas son muy confiables, pero poco eficientes.

*Poca coordinación
gran confiabilidad
baja eficiencia

chapter 2

The Computer Control System

The objective of this chapter is to briefly discuss the hardware (both computer and computer/process interface) and software generally found in a process computer configuration. Since other texts are available, we shall not give a very detailed discussion of subjects such as how a digital computer works. Furthermore, computer hardware has historically changed very rapidly, so any discussion is likely to become obsolete very quickly. In this chapter our main objective is to try to show the relationship of various hardware and software features to the capability of the computing system to perform in a process control environment.

Discussion of specific systems is intentionally avoided.

2-1 NUMBER SYSTEMS

The smallest storage unit in a digital computer is called a *bit*, a contraction of "*b*inary dig*it*." It can assume only two states—on or off—and thus can represent only the numbers zero and one. The base two or binary number system is most conveniently and efficiently used in such computers, which are frequently referred to as *binary machines.*

While the machine may conveniently work with binary numbers, programmers do not find this representation especially convenient. A casual examination of the first column of Table 2-1 should reveal the reason: too many ones and zeros leads to confusion. Unfortunately, conversion to the common decimal or base 10 number system is not especially easy. Instead, conversion to the octal (base 8) or hexadecimal (base 16) system is quite direct. For example, to convert from binary to octal, simply group the binary digits in groups of three from the right, and convert each group to octal. The binary number 100110111010 is converted as follows:

TABLE 2-1
Number Systems

Binary (base 2)	Octal (base 8)	Decimal (base 10)	Hexadecimal (base 16)
0	0	0	0
1	1	1	1
10	2	2	2
11	3	3	3
100	4	4	4
101	5	5	5
110	6	6	6
111	7	7	7
1000	10	8	8
1001	11	9	9
1010	12	10	A
1011	13	11	B
1100	14	12	C
1101	15	13	D
1110	16	14	E
1111	17	15	F
10000	20	16	10

100	110	111	010
4	6	7	2

Similarly, it is converted to hexadecimal as follows:

1001	1011	1010
9	B	A.

Conversion from octal or hexadecimal to binary is equally as easy. For the beginner, Table 2-1 is a useful aide, but it becomes unnecessary with a little practice.

Another characteristic that should be noted about the binary number system is the largest decimal number that can be represented by a given number of bits, which is given in Table 2-2 for up to sixteen bits. The first four entries can be verified from Table 2-1. The other entries can be computed as follows:

$$\text{Largest decimal number} = 2^n - 1$$

TABLE 2-2
Number of States per Number of Bits

Number of Bits	Largest Decimal Number	Number of States
1	1	2
2	3	4
3	7	8
4	15	16
5	31	32
6	63	64
7	127	128
8	255	256
9	511	512
10	1,023	1,024
11	2,047	2,048
12	4,095	4,096
13	8,191	8,192
14	16,383	16,384
15	32,767	32,768
16	65,535	65,536

where *n* is the number of bits. For example, computers that store data as one entry per sixteen bits are common. Reserving one bit for the sign, the largest number that can be stored in the remaining fifteen bits is 32,767. Another way of looking at this is to say that the maximum resolution of this data is one part in 32,767, or 0.003 percent.

In other applications, the number of states that can be represented by *n* binary bits is of importance, which is also given in Table 2-2. This is simply one more than the largest decimal number.

In the second generation computing machines (IBM 7094 and similar series), six bits were sufficient to represent the character set (letters of the alphabet, the ten digits, and special symbols such as the decimal point, comma, parentheses, etc.). Two octal digits could represent the six bits, and the use of the octal number system was common. With the introduction of the next generation of computers (IBM 360 and similar series), the character set was expanded, requiring eight bits for representation. The term "byte" arose to refer to a group of eight bits, and such computers were often referred to as byte-oriented machines. As two hexadecimal digits are required to represent the eight bits in a byte, this number system began to be used in place of the octal system. Not all manufacturers adopted the expanded character set, so the octal system still enjoys some use. Actually, the expanded character set is not necessary for most process control systems, but it is convenient for compatibility with the larger data-processing machines.

2-2 CENTRAL PROCESSING UNIT

The central processing unit, often designated CPU for short, is the heart of the computer, as illustrated in Fig. 2-1. Among its primary functions are the following:

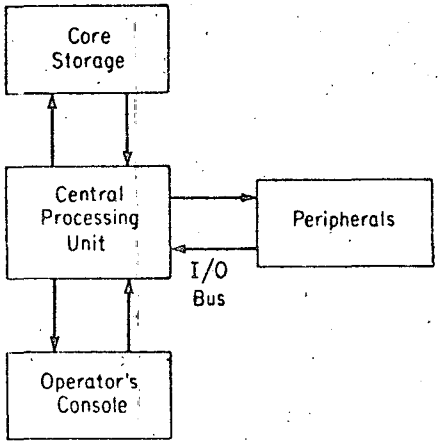


FIG. 2-1. Schematic representation of a computer.

1. Keeps track of the current location in the sequence of instructions via the *instruction address register*, which generally contains the address of the next instruction to be executed.
2. Retrieves instructions from core storage, decodes them, and executes them. The CPU contains hard-wired logic to perform a certain number of operations, which comprise the instruction set for the computer. These instructions might entail storage or retrieval of data from core storage, arithmetic operations, logic operations, or shift operations.
3. In simpler machines the CPU is responsible for the transfer of data between core storage and the peripheral units. In more sophisticated machines the CPU only directs these operations, a point we shall examine more closely in a subsequent section.

The word length of the computer generally corresponds to the number of bits which the processor stores in or retrieves from core storage in one read/write operation. Word lengths vary from machine to machine, with 8-bit, 12-bit, 16-bit, and 24-bit word lengths commonly used in process control computers. Of these, the 16-bit word length is most common.

The address of a word designates its location in core storage. Given the address, the CPU can retrieve its contents from core storage. However, the contents of the word generally give no clue as to the address from which it came.

The cycle time of the machine is the time required for the CPU to read one word from core storage and restore the contents. The cycle time is basically determined by the size of the ferrite rings used, in the core storage on current computers. The smaller these rings the faster the machine. But as the rings become smaller, the energy required to energize or de-energize them becomes smaller, and thus faster core is more subject to noise-induced errors. Cycle times on current machines range from slightly less than one microsecond (μsec) to about 4 μsec.

As we shall see, the cycle time is not the sole determinant of how fast the computer will execute a given set of code. For example, not all instructions can be executed in one cycle time. Furthermore, the instruction sets differ considerably from one machine to the next. Therefore a task that one machine could accomplish by executing one or two instructions might require four or five on another machine. Even though the second machine might have a shorter cycle time, it may not perform the desired operation as fast as the first machine.

To assist in performing various operations, the CPU has a number of registers, one of which, the instruction address register, has been mentioned already. Earlier machines had separate registers for different purposes, such as an accumulator to store the results of arithmetic operations, index registers for modifying addresses, and other registers for various purposes. Current machines tend to have general-purpose registers which can be used for practically any purpose with few restrictions. In this way, the registers are of more general utility and enable the programmer to prepare a more efficient program. All other things being equal, a computer with more registers will generally perform a given task faster than a machine with fewer registers.

Preferably, the registers are implemented as flip-flops in the CPU itself. An alternative is to reserve a few storage locations in the lower part of core storage for use as registers. This leads to a less expensive CPU but also to slower execution speeds. When a register is part of core, one memory cycle time is required to retrieve its contents, whereas considerably less time (on the order of 200 nanosec (0.2 μsec) or less) is required when the registers are part of the CPU.

A feature now enjoying considerable popularity is the read-only memory (ROM), a medium in which information is stored in permanent (nonerasable) form. This type of storage offers three advantages over read/write core.

1. Faster by a factor of about 10.
2. Less expensive.
3. Stored information is permanently protected from erasure by a "run-away" program.

Current practice is for the ROM to be prepared at the factory with field modification virtually impossible, but field-programmable ROM's are expected.

As an example of an application of an ROM, a commonly used routine such as the square root could be implemented in ROM to take advantage of the increased speed of execution. In other applications, special mathematical routines such as the fast Fourier transform could be implemented via ROM.

Microprogramming is another feature that increases the flexibility and decreases the costs of the central processor, making it quite popular for use in small computers. In this approach, a microprogram is prepared giving the elementary sequence of steps required to perform the same instruction that otherwise would have been implemented as a hard-wired instruction. In this approach, microprograms could be prepared to enable one machine to execute the instructions of another machine (i.e., to *emulate* the second machine). Use of an ROM in which to code these instructions is certainly advantageous.

2-3 RELATIONSHIP OF WORD LENGTH TO PERFORMANCE

When selecting a computer, the user can choose between various machines with different word lengths. For process control, the 12-, 16-, 18-, or 24-bit word lengths are all frequently used. The word length has a definite impact on the performance of the computer, and thus becomes an important factor in machine selection.

As either a data entry or an instruction can be stored in a word of memory, consideration must be given to both. We shall first consider data storage, then the instructions.

Process data generally enters the computing system in integer or fixed-point format. For example, suppose the input is a voltage signal in the 0 to 5 volt d-c range. If we use an 11-bit A/D converter, an input of 0 volts would correspond to all bits being set at zero; an input of 5 volts would correspond to all bits being set to 1, giving the binary representation of the decimal number 2047 (refer to Table 2-2). Since the resolution of this arrangement is 1 part in 2047 or slightly better than 0.05 percent, this is entirely adequate for most process transducers, whose accuracy is usually about 0.1 percent. Adding a bit for the sign gives a total of 12, and therefore a 12-bit word length would be adequate for storing most process data in integer format. Use of a longer word length would be wasteful.

When working with process data, it is generally more convenient to first convert it to engineering units. The integer or fixed-point representation is not especially convenient for this purpose, the real (floating-point or exponential) format being much more attractive. In this approach, a certain number of bits are reserved to represent the characteristic (including sign) and a certain number of bits are reserved to represent an integer exponent (including sign). The minimum workable combination is to reserve about 18 bits for the characteristic (giving from four to five decimal digits of precision) and about 6 bits for the exponent (which is sufficient to represent numbers between approximately 10^{-9} and 10^{+9}). This requires a total of 24 bits.

Although four digits is generally sufficient to represent the raw process data, this relatively low precision coupled with the round-off characteristics of binary machines often leads to numerical problems even in relatively simple mathematical procedures. Using a total of 32 bits, giving seven or eight digits of precision, to represent a real number circumvents these problems in most process control applications.

Insofar as process control applications are concerned, the following general statements apply to the selection of the word length in light of the data storage aspect.

12-bit word. Since two or three words would be required to represent a floating-point number, virtually all data must be stored in integer form. In fact, floating-point operations should be avoided. Therefore, machines in this category could be considered only for those applications in which little or no floating-point operations are expected.

16- or 18-bit word. In these machines, the use of two words to store a floating-point number makes their use a bit inconvenient but yet quite feasible. Storage of data in integer format reduces the words of core storage required by a factor of two. Manipulations of floating-point data will also be inherently slower because two memory cycles are required to retrieve a floating-point number from core storage as compared to one cycle to retrieve an integer number.

24-bit w In these machines there is no penalty for storing data in floating-point format. However, the relatively low precision of the floating-point number may require the use of double precision in some operations.

Of course, the cost to performance ratio is really the number of importance. Currently (1971), core storage costs about one dollar per byte (8 bits). Naturally, the 24-bit word length is the more expensive.

Virtually all process control computers in use today employ some variation of the single-address instruction format. As illustrated in Fig. 2-2, the instruction is divided into three fields, the operation code, the address mode, and the address itself. The purpose of these fields are as follows:

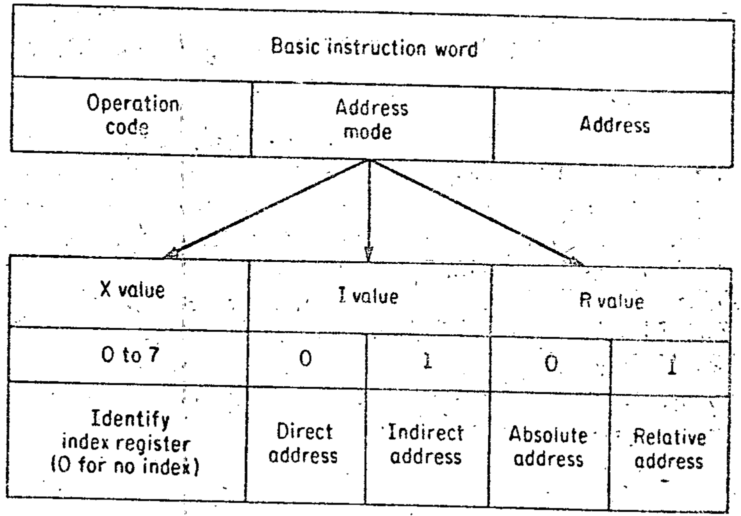


FIG. 2-2. Format of a single-address instruction. (Reproduced by permission from Ref. 1.)

Operation code. This field specifies the operation to be performed.

Address. This field contains the address utilized in executing the instruction.

Address mode. This field designates what modifications are to be made in the address contained in the address field before the instruction is executed.

Disregarding the address modifications for the moment, consider the following examples of instructions:

Left-Shift or *Right-Shift.* This instruction causes the contents of the accumulator to be shifted to the left by one bit o the right by one bit. The address field is not used.

Store-Word. This instruction stores the contents of the accumulator into the word whose address is in the address field. The converse of this operation is "load word."

Unconditional Transfer. After execution of this instruction, the next instruction executed is the one whose address is in the address field of the transfer instruction. Execution of the transfer instruction simply requires placing the contents of the address field into the instruction address register.

Load Immediate. Some instructions treat the contents of the address field as if it were data. For example, the "load immediate" instruction transfers the contents of the address field into the accumulator.

This last instruction illustrates an example of the effect of the instruction set on the machine's performance. On machines with an abbreviated instruction set not containing the "load immediate" instruction, a word of core storage must be reserved for the data and a "load word" instruction used instead. This "wastes" a word of core storage.

In direct addressing, the address field contains the actual address of the information to be accessed. In process control computers, three common approaches to modifying this address are used:

1. *Relative Addressing.* The contents of the address field are added to the contents of the program-location register to obtain the address to be used. In computers without this feature, a program is written (or compiled) to be executed from a predetermined location in core storage. Incorporating this feature permits a program to be loaded into any position in core storage and executed, a feature called *dynamic allocation of core storage.* As we shall see in a later section, this can be done only with the aid of a mass-storage device such as a disk or drum. Therefore, this feature is of little value on all-core machines.
2. *Indirect Addressing.* In its simplest form, the address field contains the address of a word in core storage that contains the address to be used in executing the instruction. This is known as *single-level* indirect addressing. This procedure can be nested to give *multilevel* indirect addressing. An extra memory cycle is required for each level of indirect addressing.
3. *Indexed Addressing.* The contents of an index register are added to the contents of the address field to obtain the address to be used in executing the instruction. If the index register is implemented as a word in core storage, a memory cycle is required to retrieve its contents. Implementing the index register as flip-flops in the CPU saves this time.

All of these types of address modifications may be used simultaneously.

To illustrate the effect of word length on the computer's performance, suppose we are considering a 16-bit machine with three index registers and the capability to perform relative and indirect addressing. This means that the address mode field must contain four bits—two bits to designate the index registers, one for relative addressing, and one for indirect addressing. This leaves twelve bits for the other two fields.

Furthermore, suppose four bits are reserved for the operation code. Table 2-2 indicates that four bits can designate only 16 different instructions, a rather paltry number. However, ingenious schemes have been devised to circumvent this problem. For example, all instructions not utilizing the address field are given the same operation code. Then the contents of the address field are used to specify the specific operation to be performed.

Reserving four bits for the operation code and four bits for the address mode leaves eight bits for the address field. Table 2-2 indicates that eight bits would be sufficient to direct address only 256 words of core storage. This fact indicates that indirect addressing must be used extensively on these machines, thereby reducing the effective speed with which they can execute a program.

On 24-bit machines the address field is sufficient to direct-access about 16K (K = 1,024) words of core storage. Thus indirect addressing is used less frequently. On 32-bit machines, the address field is generally sufficient to direct-access all of core storage.

On machines with word lengths shorter than 16 bits, double-word instructions must frequently be used, thereby offsetting the advantages of using the shorter word.

As the final point in this section, it should be noted that the word length essentially fixes the maximum core storage available on a 16-bit machine. As the maximum address that can be represented by 16 bits is 65,535, the maximum core available on most 16-bit machines is 64K.

2-4 CPU OPTIONS

In this section, we shall define a CPU option as any feature of the CPU that is optional on some (not all) computers that are frequently considered for process control. That is, some of our "options" are standard features on some computers.

Hardware Multiply/Divide (Also Called Fixed-Point Arithmetic)

Virtually all CPU's have an instruction to add the contents of a memory location to the contents of the accumulator (i.e., a fixed-point add instruction). While multiplication of two fixed-point numbers can be accomplished by successive additions and shifting operations, this entails two penalties:

1. Execution speed is reduced due to the large number of operations required.
2. The instructions required in this procedure must be stored at least once (usually as a subroutine) in core storage.

Division can be accomplished in a similar fashion, and the software routines for this purpose are commonly referred to as fixed-point software.

An alternative procedure is to implement hardware to perform fixed-point multiplications and divisions. This eliminates the need for the software and also increases execution speeds significantly, the order of magnitude being as follows:

	Hardware	Software
Multiply	10 μsec	200 μsec
Divide	20 μsec	500 μsec

As the cost is also reasonably low (about $2,000 in 1971 prices), this feature is found in most process control computers. However, in computers used for other purposes (e.g., in communications networks), this feature is not so important.

Hardware Floating-Point Arithmetic

In the minimal configuration, few CPU's have the capability to perform any floating-point operation. Just as in the case of fixed-point multiply/divide, either software routines may be used or additional hardware can be purchased. In either case, the functions that must be supplied include addition, subtraction, multiplication, division, and other floating-point manipulations. Orders-of-magnitude comparison of execution speeds of hardware vs. software are as follows:

	Hardware	Software
Add and Subtract	15 μsec	400 μsec
Multiply	20 μsec	400 μsec
Divide	30 μsec	1000 μsec

This feature is not commonly found on process computers because 1) the price is substantial (about $20,000 or more in 1971 figures), and 2) floating-point operations can be avoided to a large extent on process control computers.

Storage Protect

In process control computers, it is frequently desirable to protect a certain segment of the programs from being accidentally written over by a runaway program outside this segment of programs. One approach to implement this is by including a protect bit with each word of core storage. In this way a protected location of core storage can be written into only by an instruction whose protect bit is on. This feature in some form is found on most process control computers.

Because of the expense of adding a bit to each memory location, some manufacturers have adopted the paging concept for storage protect. In this approach, a single protect bit is provided for a segment of core storage generally consisting of about 256 or 512 words, otherwise known as a page.

Parity ?

In order to provide some error-detection and correction capability, a parity bit can be added to each word of core storage and to words of information transferred between peripheral devices. To illustrate the functioning of parity, suppose the parity bit is set "on" when the number of "on" bits in the word is odd. If an even number of bits are "on," the parity bit is set "off." Then including the parity bit, the number of bits that are "on" should always be even. If an error is made involving any one bit, the number of "on" bits would be odd, indicating an error. If two errors are made they would not be detected, but the probability of this happening is extremely remote.

Several manufacturers, contending that their core storage is so reliable that parity checking is not needed, do not even offer it as an option. However, peripherals are not so reliable, and data transferred to and from peripherals should always be accompanied with a parity bit.

Real-Time Clock

Virtually all process control computers require a real-time clock in order to coordinate the computer's operation with the real world's time schedule.

Power Fail-Safe

In the event of loss of power to the computer, this option provides the capability of executing a set number of instructions before the machine becomes inoperable. These instructions may generally be used for whatever the specific application requires.

Automatic Restart

With loss and resumption of power, the contents of core storage are not altered. However, the contents of the working registers implemented as flip-flops in the CPU are lost. But if some of the instructions available from the power fail-safe option are used to store the contents of the working registers, program execution can proceed when power is resumed. The function of the automatic restart option is to reload the working registers with their contents at the time of loss of power and resume program execution.

Watchdog Timer or Operations Monitor

If for any reason a program became "hung up" in a never-ending loop, the process control computer would effectively cease to perform all needed functions. To provide protection against this, the watchdog timer must be reset within a certain allotted time period (e.g., 15 sec) by whatever program or programs are being executed. Failure to do this serves as an indication of a problem somewhere in the software.

2-5 I/O STRUCTURE

As indicated previously, input/output (I/O) operations in earlier computers were accomplished via the CPU. In this way the CPU was committed to the I/O operation while it was in progress, and therefore was not available for other functions.

The I/O performance was improved by adding an I/O processor which operated independently but yet through the CPU on a cycle-stealing basis. That is, the CPU instructed the I/O processor as to what operations were needed, and these were performed by "stealing" memory cycles from the CPU as the peripheral device could receive or transmit information. This frees the CPU so that the remaining memory cycles can be used for computational purposes.

By using a multiple port to memory or direct memory access channel as illustrated in Fig. 2-3, the CPU is completely free of the major I/O functions. The direct memory access channel essentially consists of a satellite CPU whose functions are basically limited to I/O operations. When high data-transfer rates are expected, this approach is extremely attractive.

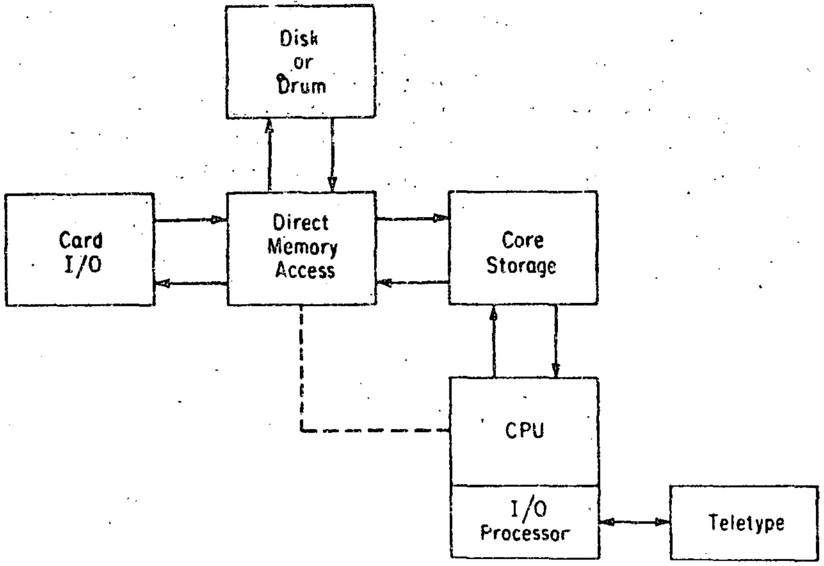


FIG. 2-3. Direct memory access channel.

The use of multiple ports to memory can produce a variety of computer configurations, even involving multiple processors as illustrated in Fig. 2-4. Each CPU has its own private memory in addition to the shared memory, which enables the two processors to communicate with each other quite readily. Peripherals with or without a direct memory access channel can be added to each CPU.

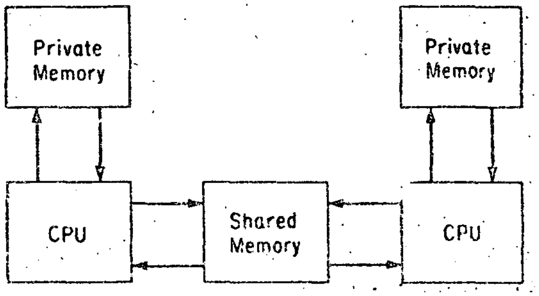


FIG. 2-4. Multiple processor configuration.

2-6 PERIPHERAL DEVICES

In this section we will be concerned only with the classical data-processing peripherals—teletype, paper tape, and similar devices. Process-oriented I/O devices are discussed in a later section.

Teletype

Virtually all computers have a teletype or typer in the computer room for communications with the computer operator. In addition, many process control computers have additional teletypes or typers out in the field for operator communications. These devices are rather low speed (10 to 15 characters per second), but their low cost makes them quite attractive where the output volume is low.

CRT Display Units

The low-speed output from the teletype detracts from its utility for operator communications. When a hard copy is not necessary, the cathode-ray tube (CRT) display units can accept a rather high data rate, and therefore are becoming quite popular for operator communications. One approach is to display information to the operator via the CRT, obtaining a hard copy of the desirable information via the teletype or line printer in the computer room. The alphameric CRT's are reasonably cost-competitive with the teletypes. Vector-drawing CRT's are considerably more expensive and therefore used more sparingly.

Paper Tape Read/Punch

While a slow-speed (10 characters/sec) paper tape read/punch can be added to a teletype for a nominal expense, the input/output speeds are too slow for all but a few applications. A high-speed paper tape unit (200 characters/sec reader; 100 characters/sec punch) has sufficient speed for normal program preparation, program debugging, and system maintenance. This unit is substantially less expensive than an equivalent card read/punch, but is not nearly as convenient for program preparation and debugging.

Card Read/Punch

While the high-speed paper tape unit was rather standard on early process control computers, the card read/punch has replaced it on practically all systems on which a significant program development effort is anticipated. Typical speeds for card read/punch units on process control computers are 200 card/min reading, 80 card/min punching. A card I/O unit generally costs at least twice that of a comparable paper tape unit.

Line Printer

The volume of printed output from a process control computer is seldom sufficient to justify the cost of a line printer. But for systems on which a large program development effort is expected, consideration should be given to renting a line printer during the initial programming stages when the volume of output is high.

Drum

A drum is a mass-storage device on which information is stored on the magnetized surface of a rotating drum. This surface is divided into tracks with a read/write head over each track. The rotating speed of the drum is such that one revolution is made every 33 millisec. If the item of information to be read from the drum has just passed under the read/write head, the computer must wait 33 millisec for the drum to make a complete revolution. This is the worst possible case, and is known as the *maximum access time*. On the average, the computer would have to wait for the drum to make one half revolution or 17 millisec, which is referred to as the *average access time*. The read/write circuitry is fast enough so that words can be read from or written onto the drum sequentially as it rotates.

The advantages and disadvantages of a drum relative to a disk are discussed in the next section.

Disk

A disk is similar to a drum except for two aspects. First, the magnetic coating is on the surface of a flat, circular plate which rotates at about the same speed as the drum. Second, most disks are equipped with a single head that can be moved from track to track to obtain the desired information. The average access time of the disk is essentially dictated by the speed of the positioner. Disks with very slow mechanical positioners have an average access time of about 500 millisec and a maximum access time of about twice this. Disks with the very best positioners have average access times of around 100 millisec. Recently disks have appeared with a read/write head per track. With an average access time of about 17 millisec, these disks are virtually equivalent to drums, and are often referred to as *drisks*.

The relative advantages and disadvantages of a movable head disk over a drum or drisk are

1. Since the read/write heads are quite expensive, the disk is

generally less expensive than a drum of the same storage capacity.

2. As evidenced by the average access times listed previously, the drum is faster.
3. Mechanical components have historically been the least reliable portion of a computer system. Thus by eliminating the mechanical positioner the drum is generally more reliable.
4. Many disks permit disk surfaces to be interchanged, which permits a copy of the information on the disk to be stored off-line as a backup. This is not possible with drums or drisks.

Maximum storage capacity is generally not a factor, since very large disks and very large drums are available. For process control, a minimum of a million words is generally required.

Magnetic Tapes

Due to the comparatively long access time of the magnetic tape, these units are rarely found on process control computers.

2-7 TYPICAL CONFIGURATIONS

Process control computers come in a wide variety of configurations, depending heavily on the application. In this section we shall give typical configurations of three classes of computers. For each of these we shall give an approximate cost breakdown based on 1971 prices.

Minicomputer

Usually installed as a dedicated computer to perform a relatively simple task, the configuration, as illustrated in Fig. 2-5, is practically the absolute minimum. These machines are generally programmed in assembly language on a once-for-all basis. For this to be practical, the task the computer is to perform must be well-defined beforehand.

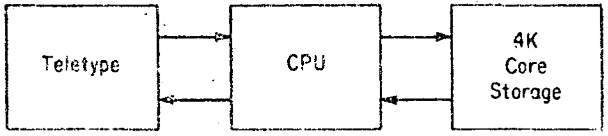


FIG. 2-5. Configuration of a minicomputer.

Most writers tend to define the minicomputer in terms of its cost (2). A typical definition of a minicomputer is one costing less than $25,000, again in 1971 prices. The configuration in Fig. 2-5 could be purchased in 1971 for less than $15,000 even with a 16 bit word length.

Direct Digital Control

Figure 2-6 illustrates a typical configuration of a computer used for direct digital control. Because fast response is generally the basic requirement of a DDC system, an all-core (no disk or drum) machine is illustrated. Programming would generally be in assembly language, although several standard DDC packages are available. Since relatively little programming effort is anticipated after the system once becomes operational, a paper tape I/O is frequently used on these machines.

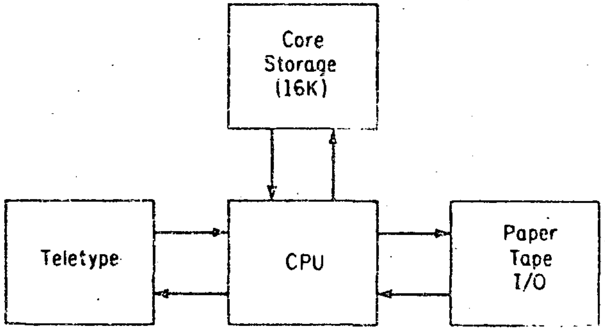


FIG. 2-6. Typical configuration of a direct digital control computer.

Based on 1971 prices, the cost of the configuration in Fig. 2-6 is approximately as follows:

CPU (16 bit) with hardware multiply/ divide, storage protect, real-time clock, power fail-safe	$20,000
Core storage	32,000
Teletype	3,000
Paper tape I/O	8,000
	$63,000

A machine of this configuration would probably be adequate for no more than 100 loops with a reasonable complement of feedforward, cascade, and other advanced control strategies.

Supervisory Systems

On the configuration of the supervisory system illustrated in Fig. 2-7, most of the programming could be done in a compiler level

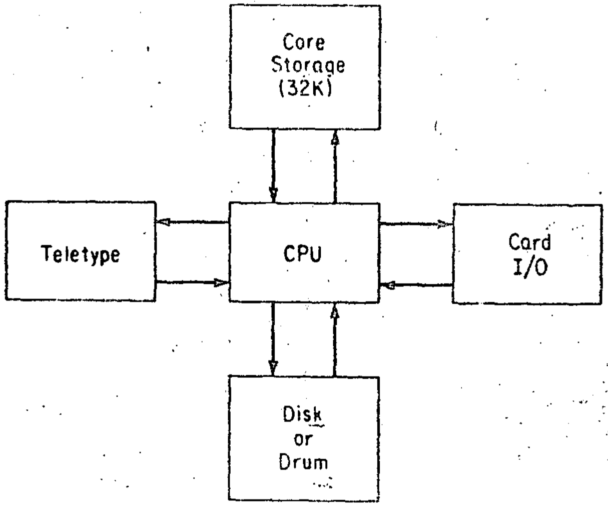


FIG. 2-7. Supervisory system.

e such as Fortran. Program development and compiling could on-line. The operating system would transfer programs from k or drum to core storage for execution. Since program de-ent and subsequent system improvement is likely to continue ie time, card I/O is preferred.

ed on 1971 prices, the cost of the configuration in Fig. 2-7 is imately as follows:

CPU (16 bit) with hardware multiply/ divide, storage protect, real time clock, power fail-safe	$50,000
Core storage	64,000
Card I/O	20,000
Teletype	3,000
Disk	35,000
	$172,000

a line printer would cost another $30,000, the main reason is usually omitted.

tal Project Cost

hough the computer system is generally a significant element total project cost, many other factors must be considered to it a total proj cost, which includes the computer system and iliary equipn. , programming, process analysis, installation, aining. Table 2-3 gives the cost breakdown on eight selected

TABLE 2-3

Costs of Selected Process Computer Systems in Survey Plants with Purchased Computers. (Reprinted from "Outlook for Computer Process Control", U.S. Department of Labor Bulletin 158, 1970.)

Type of Application	Total cost	Computer and Auxiliary Equipment[1]		Programing and Systems Analysis[2]		Installation and Additional Instrumentation[3]		Training[4]	
		Amount	Percent of total system cost	Amount	Percent of total system cost	Amount	Percent of total system cost	Amount	Percent of total system cost
Multicomputer system controlling all major processes in large chemical plant	$1,500,000	$1,125,000	75.0	$225,000[5]	15.0	$150,000	10.0	—	—
Complex system for control of an electric generating station	850,000	400,000	47.1	190,000	22.4	250,000	29.4	10,000	1.2
Operator guide control over a major process in a steel plant	810,000	290,000	35.8	300,000	37.0	200,000	24.7	20,000	2.5
Operator guide control of electric generating station	720,000	300,000	41.7	140,000	19.4	275,000	38.2	5,000	0.7
Direct digital control of a chemical process	500,000	275,000	55.0	75,000	15.0	150,000	30.0	—	—
Control over a key portion of a chemical process (early installation)	453,000	258,000	57.0	75,000	16.6	110,000	24.3	10,000	2.2
Control of analytical instruments in chemical plant laboratory	235,290	160,000	68.0	58,820	25.0	16,470	7.0	—	—
Experimental direct digital control system using 2 computers in a chemical plant	222,000	157,000	70.7	50,000	22.5	10,000	4.5	5,000	2.3

---

[1] Central processor, auxiliary memory, analog/digital signal converters, and input/output equipment such as operator console, typewriters, and tape equipment.

[2] Analysis of process, preparation of process model, programing for process control, and system operation.

[3] New instrumentation needed for process control installation of computer equipment, and instrumentation including site preparation.

[4] Instructing employees in programing, computer technology, maintenance, and system operation.

[5] Includes training.

process computer systems. Note that the cost of the computer and auxiliary equipment varies from 35.8 to 75 percent of the total system cost. As in all aspects of today's economy, the trend in process control systems is that the hardware costs are tending to decline and the people-related costs are tending to rise.

2-3 PROCESS INTERFACE

In order to function properly, the computer must receive certain data from the process and transmit other data to the process. The computer/process-interface, often called the analog front end, must somehow accomplish these functions. The data involved generally fall into one of the following three categories:

1. Continuous or analog data.
2. Discrete data involving only two levels (i.e., on-off type information).
3. Pulse data.

These catagories apply to both input and output data.

Figure 2-8 illustrates the typical arrangement for reading analog values from the process. These signals can be classified as follows:

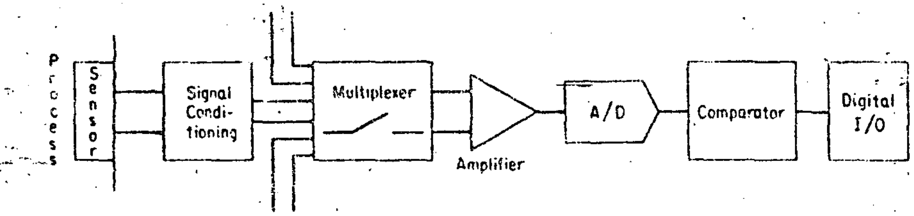


FIG. 2-8. Analog input system.

1. Low-level signals, generally considered to be those whose voltage level is less than 100 microvolts (μv), include the outputs of thermocouples, strain gauges, resistance thermometers, and similar transducers.
2. High-level signals, generally considered to be those whose voltage level is greater than 100 μv, emanate from transducers with a built-in amplifier of some type.

Due to the popularity of thermocouples for measuring temperatures, low-level signals are commonly encountered in process control systems. Naturally, these signals are most succeptible to distortion, thus requiring special precautions. The leads generally consist of a twisted, shielded pair. The leads should not be carried in the same tray as a-c power circuits, and in general should not come in close proximity of large electrical motors or generators.

Improper grounding can also be a potential source of distortion of low-level signals. In general, the circuit should be grounded at only one point, preferably at the computer. Figure 2-9 illustrates a circuit grounded at two points, one at the computer (specifically, at the amplifier) and the other at the transducer. The impedance of the amplifier is very large (on the order of 10^6 ohms), so negligible current flows around the loop. However, the two grounds are likely to be a considerable distance apart, and therefore probably at slightly different potentials. Therefore, a current i_c, called the *common-mode current*, flows in one of the leads and not the other (due to the impedance of the amplifier). The voltage drop due to this current causes a bias (which may vary with time) to appear in the reading. This bias is referred to as the *common-mode noise*. As this noise cannot be removed by filtering, steps should be taken to avoid it The easiest way is to avoid grounding at the transducer.

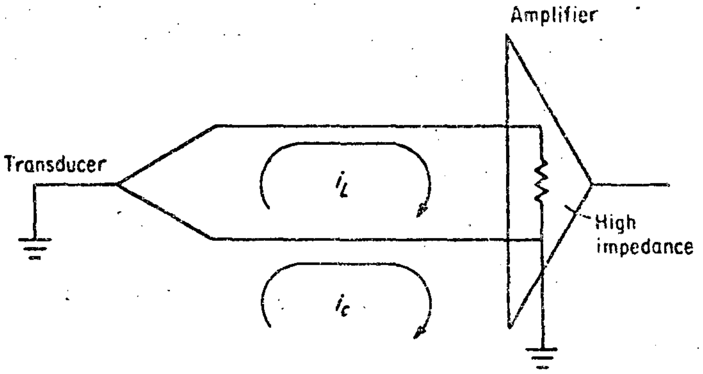


FIG. 2-9. Circuit susceptible to common-mode noise.

As rather detailed discussion of good wiring practices are available, they will not be repeated here (3).

The function of each of the elements in Fig. 2.8 is as follows.

Signal Conditioning. This may encompass a variety of element depending upon the sensor itself. When the output of the transduce is a voltage signal, the signal conditioner generally consists of only a *RC* filter. But if the output of the transducer is other than a voltag signal, the signal conditioner generally transforms it to a voltag signal prior to the multiplexer. For example, if the output of th transducer is a current signal, the signal conditioner generally contains a resistor across which the voltage input to the multiplexer taken.

Multiplexer. The multiplexer provides the mechanism by whic one of several signals is connected to the A/D converter through the amplifier. For high-level signals, solid-state electronics (field-effect transistors) are used in the switching circuits. Sampling rates of 10,000 points per second and higher are readily accomplished. For low-level signals, the distortion of the field-effect transistors cannot be tolerated. Reed- or mercury-wetted relays must be used, resulting in a much slower sampling rate (about 200 points per second). Some systems contain two distinct multiplexers—one for high-level signals and one for low-level signals.

Multiplexers range in size from about 32 input points up to 2,048 input points or more. The sampling sequence on some multiplexers is fixed to a certain sequence, yielding what is called a *sequential scan.* Other multiplexers permit selection of the point to be read, enabling the points to be read in random order. Of course, this latter multiplexer is more expensive.

Both types of multiplexers are found in process control systems. When the computer controls the analog scan, the multiplexer must be capable of reading the points in random order. On other systems, however, control of the scan may reside largely outside of the CPU. Using a sequential scan and a direct memory access channel to store the data in preassigned storage locations relieves the CPU of the burden of supervising the analog scan.

Amplifiers. The function of the amplifier is to scale the process signal either upward or downward so that the resulting range matches that of the A/D converter, typically 15 volts. Some systems utilize a fixed-gain amplifier, in which case voltage-divider circuits often appear in the signal conditioner. In other systems, a programmable-gain amplifier permits the computer to specify which one of several available gains is to be used. This latter alternative provides more flexibility, but the amplifier is more expensive and also requires some output data (i.e., the value of the gain) from the computer.

A/D Converter. Conversion of the signal from analog (continuous) form to digital (discrete) form is accomplished by the A/D converter. The resolution of the A/D converter is related to the number of bits in the digital output by the equation

$$\text{Resolution} = \frac{1}{2^n - 1}$$

where *n* is the number of bits. For process control, an 11-bit converter is entirely adequate, giving a resolution of about 0.05 percent. For some applications an eight bit converter with a resolution of about 0.4 percent is acceptable.

The time required for the digital output of the A/D converter to reach a constant value after a new input is applied is known as the *settling time.* For solid-state A/D converters, the settling time is 40 μsec or less, which becomes significant only at high data-transfer rates.

Comparator. In order to relieve the CPU of some of its burden, the input data can be compared to high and low limits outside the CPU. This feature is very attractive on systems using the sequential scan coupled with a direct memory access channel to store the input data in preassigned storage locations. The high and low limits are retrieved via the direct memory access channel from preassigned locations in core storage. If either limit is violated, an interrupt is generated, calling for the CPU's attention. Thus the input scan proceeds independently of the CPU until a limit is violated.

Although inputs which can assume only two states could be entered via the route described in the above paragraphs, this condition places an undue burden on the analog input system. Most process control systems permit the states of inputs of this type, known as *discretes*, to be read in groups. Normally each discrete is assigned to a bit in a word. In one cycle time, most computers can read a word containing the status of a number of discretes equal to the word length. The capability to manipulate the bits in a word in order to ascertain which bits are on or off becomes extremely important.

Discretes are commonly used to indicate the status of relays, which may be found in anything from electrical switches to high-pressure alarms. In the convential operator's console, the position of the thumbwheel switches and other devices for data entry is indicated via a bit pattern entered into the computer via discretes.

The output of certain measuring devices such as tachometers or turbine meters is often in the form of pulses. Although the computer can be readily programmed to count pulses for a given length of time, this tends to consume too much of the CPU's time. Instead, external *pulse counters* are generally preferred. In these devices, the CPU loads a register in the pulse counter with the number of pulses to be counted. With the receipt of each pulse, this register is "down-counted" (i.e., one is subtracted) until the register reaches zero, at which time an interrupt to the CPU is generated). To determine the time required for the given number of pulses to occur, the CPU needs only to subtract the time when the pulse counter was initialized from the current time. Thus the CPU has little to do.

The output of data to the process is generally by one of the following three means:

1. Digital-to-analog (D/A) converter, which converts a digital

value (in integer format) to an analog signal. A multiplexer could be used to obtain several outputs from a single D/A converter as illustrated in Fig. 2-10. But with the addition of the hold circuits to maintain the value between samples, the economics tend to favor individual D/A converters.

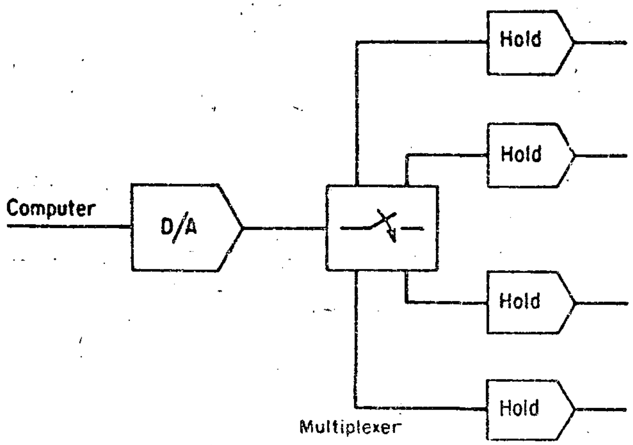


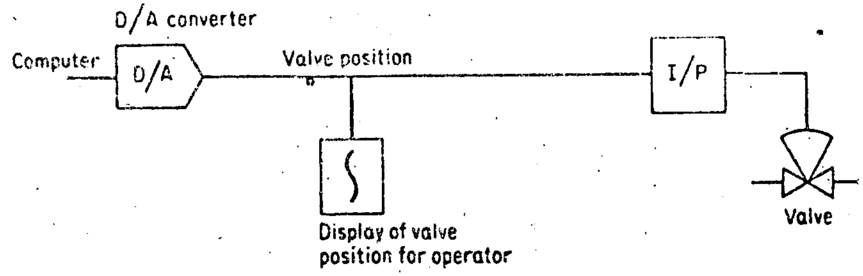
FIG. 2-10. Multiplexing the output of a D/A converter

2. Pulse generators, which generate the number of pulses specified by the computer. In most systems the pulses are of predetermined amplitude and duration and with a predetermined time between pulses. The outputs of pulse generators are commonly used to drive stepper motors.
3. Contact closures, which can assume only two states—on or off. In addition to simple applications such as turning pumps or lights on or off, a contact can be closed (or opened) for a period of time to obtain a pulse output of variable duration.

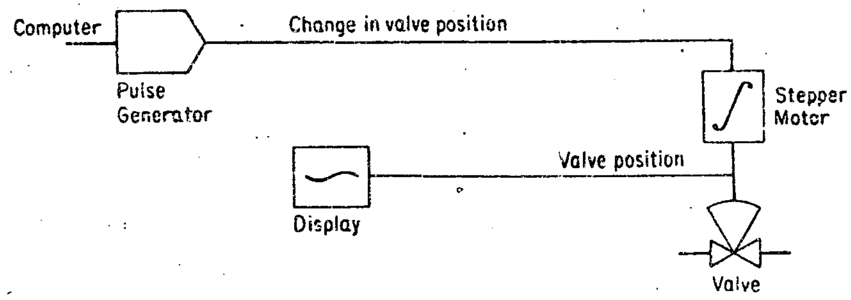
To illustrate the use of these devices, consider the output of a quantity such as a valve position or set point for an analog controller. Perhaps the most direct approach is to use a D/A converter as illustrated in Fig. 2-11a. Pertinent points are:

1. Since most valves are pneumatic, a current-to-pneumatic (I/P) transducer is required.
2. The output of the D/A converter can be displayed so that the operator can readily ascertain the valve position.
3. As the output is the actual valve position, some mechanism must be provided so that the computer can read the initial valve position.

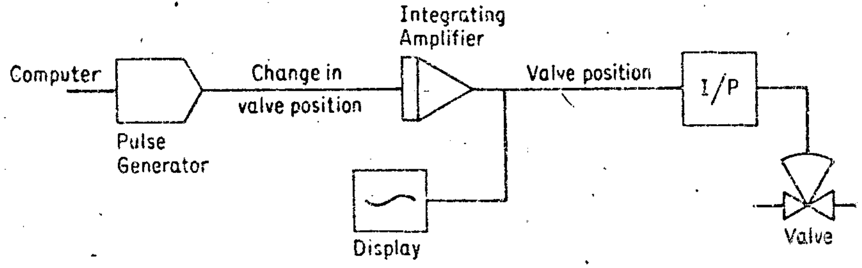
Alternatively, a pulse generator can be used in the configuration in



(a) D/A converters



(b) Pulse generator with stepper motor



(c) Pulse generator with integrating amplifier

FIG. 2-11. Uses of D/A converters and pulse generators.

Fig. 2-11b. Relevant points are:

1. Current/pneumatic transducer is replaced by the step motor, which inherently integrates the input.
2. As the computer output is a change in valve position, it is necessary that the computer be able to ascertain the va position (unless it must be verified that the desired change actually made).

3. In order that the operator be able to readily ascertain the valve position, a signal must be transmitted to the control room, thus entailing another signal lead.

This approach is commonly used for set points of analog controllers.

Another alternative is to use an integrating amplifier located in the control room as illustrated in Fig. 2-11c. This is similar to the configuration in Fig. 2-11b except for the following:

1. The valve position can be readily displayed to the operator.
2. An I/P transducer is required, although this could be incorporated into the integrating amplifier.
3. The saturation limit of the integrating amplifier may not exactly correspond to the valve full-open or full-closed, which may present some problems when using the velocity control algorithm.

Although a pulse generator is illustrated in Fig. 2-11c, a contact closure maintained for variable duration could be used instead.

2.9 SOFTWARE

In one sense, the computer control system can be considered as composed of two classes of elements. The first of these is called the *hardware*, which has been described up to this point. The second is the *software*, which can be defined as everything over and above the hardware required in order for the computer control system to function. This is perhaps the most encompassing definition, more restrictive definitions being available.

Basically there are two sources of software. The computer manufacturer generally supplies certain program packages with the computer system. Some of these are generally included in the basic price of the system. Others may be purchased at the option of the user. In either case, this software is termed *vendor-supplied software*.

Whereas the vendor-supplied software is generally usable in a relatively wide class of applications, each user will require certain programs specifically for his own installation. He has the option of either writing them himself or retaining an outside firm to write them for a negotiated fee. Software in this category is generally termed *user-supplied software*. Of course, the user would like to minimize the amount of software he must develop.

For a computer control system, the software required can be categorized as follows:

1. The operating system, monitor, or executive. This software supervises or directs the operation of the computer control system, scheduling programs for execution, transferring programs from disk to core, etc. This package is generally available from the computer manufacturer.
2. Supporting software packages, including compilers, loaders, disk editors, diagnostic routines, etc. Most of these are available from the vendor.
3. Applications programs (i.e., those directly concerned with implementing the selected control strategy). Most of these must be supplied by the user, although some parts such as operator's console service routines, thermocouple conversion routines, and the like may be available from the vendor.

With this overview of the software for a control computer, a few of the individual elements will be considered more closely.

2-10 THE ASSEMBLER

Earlier in this chapter we discussed the basic machine language, and indicated how certain operations could be obtained with the appropriate instructions. At this level programming is very tedious, and the programmer must remember the binary codes for each instruction as well as the addresses where each piece of information is stored. Programming in assembly language offers two advantages:

1. Mnemonics are used to indicate the instruction to be performed. For example, STW may indicate the "store word" instruction.
2. Variables are used in the place of absolute storage locations. The assembler collects the names of all variables used in the program and assigns storage locations to them, in much the same manner as the well-known Fortran compiler.

For example, the instruction

```
STW X
```

may instruct the machine to store the contents of the accumulator in the storage location corresponding to variable X.

In most basic assemblers, there is a one-to-one correspondence between assembly language statements and machine language instructions. These assemblers will frequently run on as small a system as one with only 4K words of core. Many manufacturers offer an advanced assembler which permits the use of "macros," which are certain assembly language statements or "instructions" that require the execution of more than one machine language instruction. A more expanded system is generally required for assemblers of this type.

As this is not a text on programming per se, we will not delve into the details of assembly language programming. Besides, an assembly language is generally specific to a specific machine, differing from one model to another even if made by the same manufacturer.

We shall defer discussion of the advantages and disadvantages of programming in assembly language until after our introduction to Fortran.

2-11 PROBLEM-ORIENTED LANGUAGES

Although modifications of other problem-oriented languages such as BASIC have been used for programming process control computers, Fortran is currently the most common problem-oriented language used in process control. As we shall enumerate shortly, the fact that Fortran has its shortcomings has given rise to some interest to abandoning the use of Fortran in process control. However, there is considerable inertia in the general use of Fortran, probably because most current technical graduates have been exposed to it. Consequently, we shall base our discussion in this section around Fortran, pointing out its advantages and limitations.

Fortran entirely abandons the one-to-one correspondence of statements to machine-language instructions. Instead, syntax is used to indicate procedures to be executed using desired information. For example, the statement

$$C = A + B.$$

indicates that A is to be added to B and the results stored in C. This statement would be equivalent to the following assembly language statements:

```
LDW   B   (load B into the accumulator)
ADD   A   (add A to the contents of the accumulator)
STW   C   (store contents of the accumulator in C)
```

As most readers are certainly familiar with Fortran, there is no need to go into the details of Fortran programming.

It should, however, be pointed out that the Fortran available on process control computers does not generally have as many features available as the Fortran available on the typical data processing machine. Notable exceptions include the absence of logical variables (including logical IF) and the ability to selectively define the precision used in the calculations. In general, single precision or double precision is used throughout the program, not just in selected places where it is needed. The same applies to variables. For example, all integer variables are stored one per word or all are stored one per two words (double precision).

Since the Fortran available on process control computers is really just a carryover of the Fortran available on data-processing machines, several needed features are not generally available. A prime example is the ability to directly perform bit manipulations. The status of process equipment is often indicated by the state of contact closures, which are read into the machine one word at a time. That is, in a 16-bit machine, the status of 16 contact closures would be indicated by one word. Therefore it is often necessary to determine if a certain bit is "on" or "off." This is readily accomplished in assembly language, but not in Fortran.

To provide capabilities like this, the usual approach has been to resort to subroutine calls to assembly-language subprograms that perform the needed manipulations. In addition to bit manipulations, most real-time functions such as initiating A/D conversions, initiating D/A conversions, generating pulse outputs, and the like are handled in this manner. This results in a certain amount of overhead in transferring control to and from the subprogram.

One approach to circumvent this drawback is to permit the insertion of assembly statements into a Fortran source program, a feature called *in-line assembly*. Now the programmer has direct access to the basic machine capability whenever the needed operation cannot be readily accomplished with Fortran.

Basically, the decision to use assembly or Fortran involves a decision of which resource is scarcer—man hours or machine capacity. Certainly a Fortran program can be prepared quicker than can an assembly-language program. However, the assembly-language program will run faster and will require less core storage. Thus a somewhat larger machine will usually be needed in order that the bulk of the programming can be done in Fortran. Other pertinent factors are outlined in Table 2-4.

2-12 FILL-IN-THE-FORMS SYSTEMS

Whether using assembly language or Fortran, the programming burden on the user is substantial. One approach to reducing this burden on the user is via fill-in-the-forms packages, where the user designates by data cards what functions are to be performed. In essence, the master program makes available to the user a number of functions. Via the input data deck, he prescribes what operations are to be performed on designated inputs to produce designated outputs.

TABLE 2-4
Assembly Versus Compiler Languages
(Reproduced by permission from Ref. 5)

Language	Advantage	Explanation
Assembly	Fast object code	Fewer instructions to convert into machine code decreases execution time
	Efficient memory utilization	Assembly code can take advantage of memory-conserving features of modern control computers
	Control over program and data location	Assembly code offers more flexibility in specifying program layout and data storage
	Access to all computer functions and instructions	Programmer can take advantage of his detailed computer knowledge to write more effective control programs
	Efficient program linkage	Calling up subroutines and shifting control parameters is simpler
	Ability to use different classes of codes	Reentrant routines for servicing priority interrupt are facilitated
Compiler	Machine independent and standardized	A limited advantage
	Self-documenting	Yes, but must be supplemented
	Easier to learn	Yes, for a scientist or engineer
	Quicker, less tedious to write or modify	Yes, provided the program writer knows when to provide control alternatives
	Easier to debug—self-checking	Prevents some programmer errors

The functions normally covered by languages of this type include the input scan routine, alarm scanning, conversion of input data to engineering units, three-mode control calculations, feedforward control calculations, cascade control, and similar functions. In general, all of the basic functions common to most control systems are provided.

The fill-in-the-forms system runs in what is called the *interpretive mode.* The input data is stored somewhere in the system, and the fill-in-the-forms system searches through the input data to ascertain what functions are to be performed. This entails considerable overhead as compared to either assembly or Fortran programs written to accomplish the same task, which necessitates a more expanded computer tem to perform the same task. The fill-in-the-forms language is not a compiler.

As it is unreasonable to expect any fill-in-the-forms system to provide all the functions required of a computer control system, provision is generally made in these systems for the user to add routines as necessary to augment the system.

2-13 DOCUMENTATION

No matter which programming language is used, preparation o
adequate documentation requires considerable effort, but is a tas
that must be undertaken while preparing the programs themselves
That is, it is not feasible to delay preparation of documentation unt
the programming task is completed.

In preparing documentation, the objective should be to enab
someone who is totally unfamiliar with the program to quickly an
easily understand its purpose and how it works. The following item
are essential:

1. A written statement of the function this
   form, as well as the details of the approach
   the desired function.
2. A flowchart of the program.
3. An up-to-date program listing.
4. Definitions of all variables used in the program.

This should be augmented as necessary to provide complete coverag

In regard to defining variables, a standard naming convention f
all variables in the various programs in the system has some meri
In this approach, each character in the variable name designates son
thing about its meaning. This method should lead to more consiste
variable naming, but a list of variable definitions for each program
still desirable.

2-14 FOREGROUND/BACKGROUND OPERATION

The programs executed by a typical process computer are ger
ally divided into two types: *foreground tasks* and *background tas*
The foreground tasks are generally those directly involved in cont
ling the process. The background tasks include many of the ta
required to support the computer control system. For example,
program used to compile programs, whether foreground or ba
ground, is run as a background task. Therefore if it is desirable to
able to compile while the computer is on-line (i.e., controlling
process), the operating system or executive must be capable of sir
taneously supporting foreground and background tasks on the s
machine. This does not mean that both tasks are executed si
taneously. Instead, the background task is executed only while

computer has no foreground task to perform. Systems that cannot support background tasks while on-line are commonly referred to as *dedicated systems.*

Basically, three different arrangements could be proposed to accomplish all the tasks necessary in the operation and support of a computer control system:

1. *Foreground/background on the same machine.* This dual function places an added burden on the monitor or executive system, thereby increasing the overhead. A rather expanded system is required to support both functions. Also, some consideration must be given to the possibility of a background program going astray and interferring with the control programs operating in the foreground. This requires some form of protection, either hardware or software.
2. *Foreground/background on separate machines.* If two machines are purchased from the same manufacturer, one can be dedicated to the control functions while the second is used off-line for program development. Each of the two systems will be of smaller configurations than a machine on which both functions are implemented. Separation of the two functions also eliminates the possibility of background programs interfering with foreground programs. Since one background computer can support several dedicated control computers, this approach can be very attractive when more than one control computer is involved.
3. *Off-line support by data-processing machines.* The general idea behind this approach is that a central computer or time-sharing system can be used to provide Fortran compilations and similar functions. As the central computer is most likely of a different make than the control computer, its compilers are not usable. Instead, a compiler is required that runs on the central computer yet produces code executable by the control computer. In the case of Fortran, this removes the restrictions on core made available by the control system to the compiler, and could conceivably permit the development of more efficient compilers that also provide some extra functions needed in process control. As for assemblers, an assembler written in Fortran could be run on virtually any data-processing machine.

2-15 INTERRUPTS

The purpose of an interrupt is to permit the normal flow of execution of instructions to be altered to permit the computer to attend to some urgent or higher priority function. Interrupts are basically of three types:

1. *System interrupts.* These interrupts originate within the computer system itself and play an integral part of the functioning of the system. An example is where the output typer signals the system that it has finished typing the previous character and is ready for another.
2. *Timer interrupts.* These synchronize the operations of the system with the real world. Timer interrupts are generated at prescribed intervals of time, and their occurrence can be used to initiate the execution of control programs (such as algorithm calculations) at regular intervals of time.
3. *Process interrupts.* These originate from the process and either signal alarm conditions, request that some function be performed by the computer, indicate completion of some task within the process, or similar purpose. For example, a high-pressure limit switch could be tied into the interrupt system to indicate alarm conditions in some part of the process equipment. The "request" button on the operator's console is tied to the interrupt system, thereby permitting him to request the computer to perform certain functions. On-stream analyzers often indicate completion of the analysis via an interrupt.

In most process control systems the interrupts play a most important role in the operation of the system.

The interrupt structure varies considerably from one computer system to the next. The sequence of events associated with the occurrence of an interrupt is typically as follows.

1. The interrupt occurs.
2. Instead of executing the very next instruction in sequence, control is transferred to a designated location in core storage and the instruction contained therein is executed. If the interrupt can be serviced by this one instruction, control then reverts back to the program being executed at the time the interrupt occurred.
3. If execution of several instructions is required, the instruction executed due to the interrupt is generally a special instruction that stores the current contents of the address register and loads into the address register the location of the next instruction to be executed.
4. The instruction located at the address now in the address register is the first instruction in a program called the *interrupt service routine.* However, the information in the working registers pertains to the program in execution when

the interrupt occurred. In order to resume execution of that program, their contents must be stored. The initial instructions of the interrupt service routine must accomplish this task.

5. The instructions to accomplish the function relative to servicing the interrupt are executed.
6. The contents of the working registers are restored to their values at the time the interrupt occurred.
7. The contents of the address register is restored to its value at the time the interrupt occurred.

After the last step, the program in progress when the interrupt occurred is resumed from the point at which it was interrupted.

Process computers come in several different "styles" in regard to their interrupt structure. In one style, there is essentially only one interrupt priority. Upon initiation of the servicing of any interrupt, all other interrupts are "inhibited" (i.e., servicing is not permitted until the one currently being processed is completed). In this type of system the interrupt service routines must generally be short.

In another variation, interrupts are grouped into levels of different priority, with several interrupts being tied into each level. In this system, interrupts occurring on high-priority levels will interrupt the servicing of interrupts on lower-priority levels. However, an interrupt will not interrupt the servicing of another interrupt on the same level.

In yet another variation each interrupt is provided its own distinct priority, and interrupts the servicing of interrupts of lower priority.

Some degree of program control is provided by inhibit commands which prohibit the recognition of all or selected interrupts until the machine is returned, under program control, to the uninhibited state.

2-16 THE EXECUTIVE

The operation of the process control computer is under the supervision of the executive, which is alternatively referred to as the *operating system* or *monitor.* One of its primary functions is to schedule the execution of control programs. Somewhere within the system is located all control programs which can be executed by the computer. Some of these may be located in core at all times, and are termed *core resident.* Others may be located on the disk or drum, if available. In this case, the monitor must supervise the transfer of the programs from the disk or drum to core storage.

The scheduling of execution of control programs is accomplished

with the aid of a table called QUEUE, which contains the name of all programs whose execution has been requested but not fulfilled. Along with each program is an associated priority, which is assigned under program control at the time the program's name is placed in QUEUE. Program names are placed into QUEUE mainly by one of the following ways:

1. A control program may place the name of another program into QUEUE, thereby permitting a train of successive programs to accomplish a given task rather than one large program.
2. An interrupt service routine may place the name of a program into QUEUE. In many cases, this is the only function of the interrupt service routine.

Once a program's name is placed into QUEUE, it is removed only when the program is executed. Highest-priority programs are executed before low-priority programs. Programs having the same priority are executed on a first-in, first-out basis.

On systems operating with a disk or drum, the layout of core storage is as illustrated in Fig. 2-12. The executive generally resides

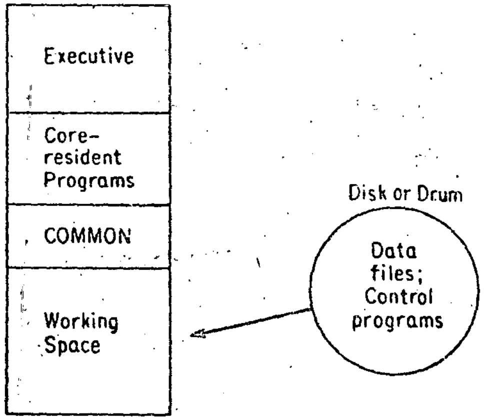


FIG. 2-12. Disk-oriented operating system.

in the lower portion of core storage. An area of core storage calle COMMON is reserved for the storage of frequently used data. Cor resident routines remain permanently in core storage. The remaind of core storage is called *working core*. It is into this area that t programs residing on the disk are loaded for execution.

When the execution of a control program residing on disk scheduled, the program is copied from the disk into working core the executive. Note the word copied—the original version on the disk is not altered. After execution of this program has been completed, the next program is copied into the same area of working core (i.e., it overlaps the original program). This means that the program is not returned to disk after execution is completed. Therefore, the same program is executed each time, only the data being different. Since the completed program is not copied back onto the disk, any data that may be needed next time the program is to be executed must be stored either in COMMON or in a file on the disk.

Depending upon the executive, the working core area may contain either only one program at a time, a specified maximum number of programs, or however many can be accommodated in the space available. In systems that can accommodate only one program at any given time in working core, the procedure is as follows:

1. QUEUE is consulted to determine which control program is to be executed.
2. The control program is loaded.
3. The control program is executed.
4. Return to step 1.

That is, QUEUE is consulted only at the completion of execution of a program. But as interrupts can be serviced while the control program is being executed, it is conceivable that an interrupt service routine could place the name of a control program into QUEUE whose priority exceeds that of the program now being executed. In most cases this program would not be loaded until execution of the program currently in working core has been completed.

The capability of multiple programs residing in working core storage at any one time is referred to as *multiprogramming*. When these programs may reside only at certain locations in core, this operation is said to be using fixed partitions, as illustrated in Fig. 2-13. Control programs are generally assigned to a particular partition and will only be executed in this partition. The term *dynamic storage allocation* is applied to the case when the program may reside in any area of working core. As illustrated in Fig. 2-13, this leads to a more efficient utilization of working core, but is more demanding on the executive and also requires some supporting hardware features (program location register) in the CPU for efficient implementation.

Although more than one program may reside in working core at any one time, only one program is actually being executed. The others are said to be in the *suspended state*.

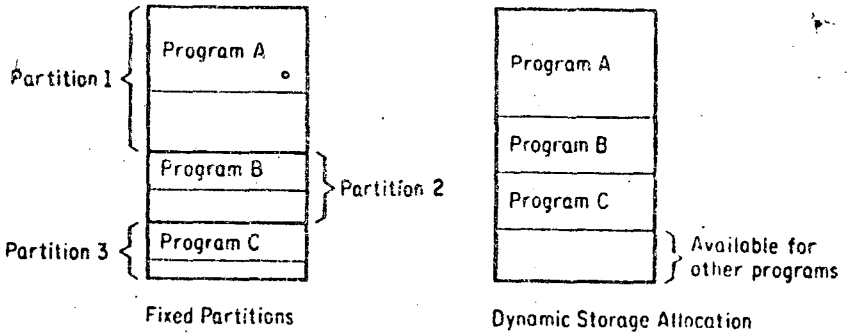


FIG. 2-13. Core allocation in multiprogramming systems.

Multiprogramming systems generally check QUEUE both upon completing execution of a control program and upon completion of an interrupt service routine. Thus, if a high-priority program has been entered into QUEUE, the executive loads it for execution provided space is available. In fixed partition systems, this generally means if the partition assigned to the program to be executed is not currently in use. For executives using dynamic storage allocation, this means if the unused area of working core is large enough to accommodate the program.

Some systems using dynamic storage allocation will remove low-priority programs to make room for high-priority ones. This is a rather ambitious undertaking. One approach is to not remove the program, but to store on disk the address in the program at which execution was terminated, the contents of the working registers, and the current values of all data used in the program. The program itself is then overlaid. When space is available for resumption of execution, a fresh version of the program is copied into working core, the working registers and data values are restored, and execution resumes.

2-17 FIRMWARE

The executives described in the previous section have one property in common—they all contain "bugs." Even with considerable effort on the part of both vendor and user, a few bugs still show up from time to time. In addition, the software executives also entail a certain amount of computational overhead to perform the desired duties. The executives also require considerable core storage, often as much as 50 percent of the available core.

One approach to circumventing these drawbacks is via a firmware executive, i.e., one that is hardware-implemented rather than software-implemented. This approach, however, has the disadvantage of generally being inflexible.

centro de educación continua
división de estudios superiores
facultad de ingeniería, unam

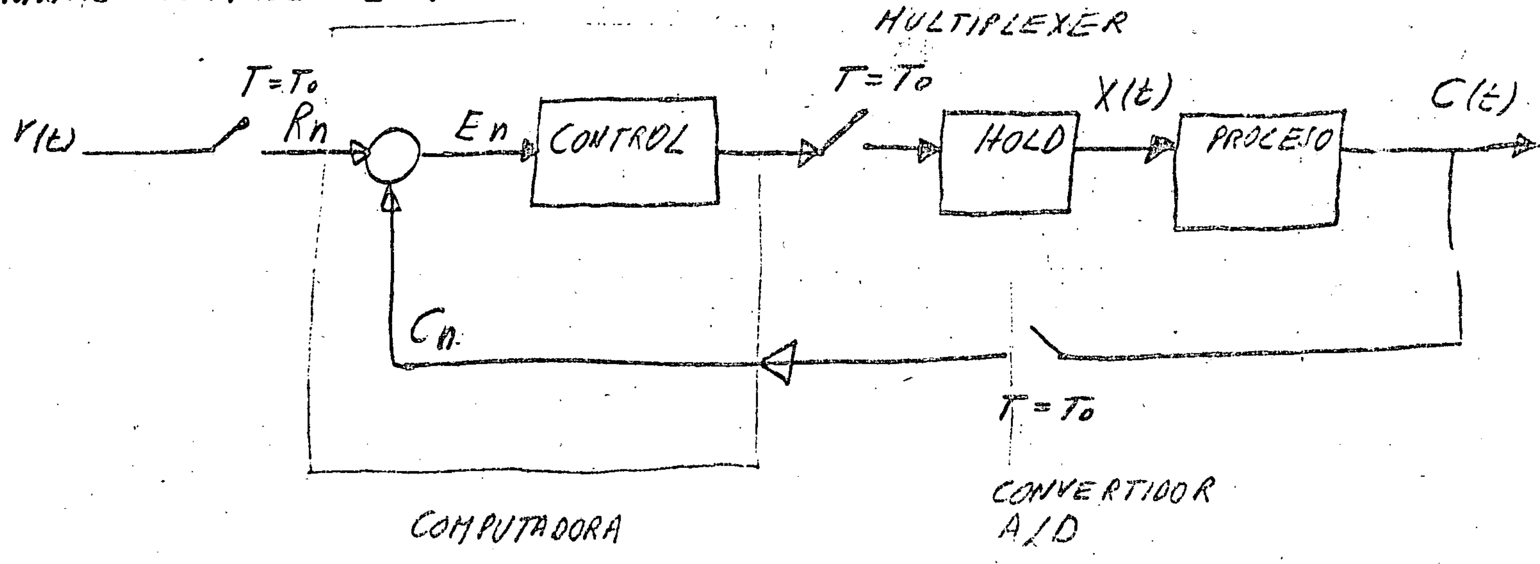
INGENIERIA DE CONTROL DIGITAL DE PROCESOS Y APLICACIONES

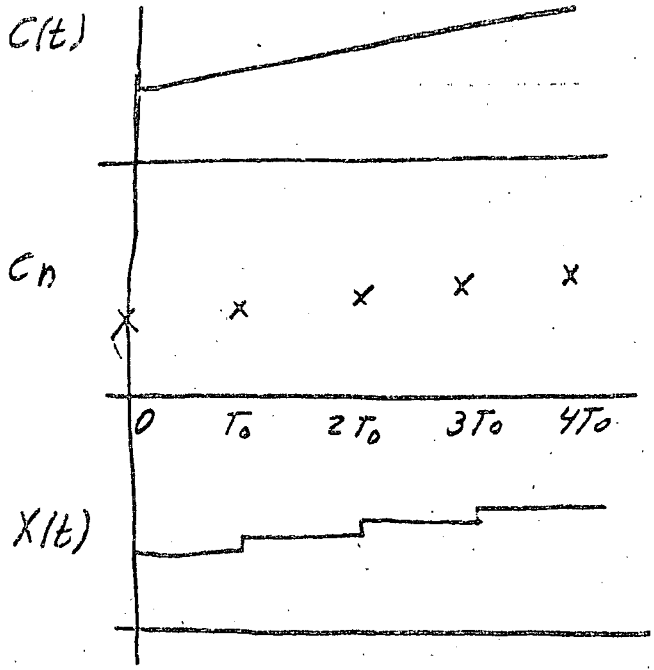
CONSIDERACIONES EN EL DOMINIO DE LA FRECUENCIA

M. EN C. ALBERTO MAURICIO MIER MUTH

NOVIEMBRE, 1978.

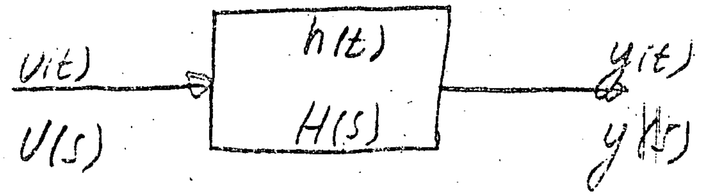
DIGITAL CONTROL LOOP



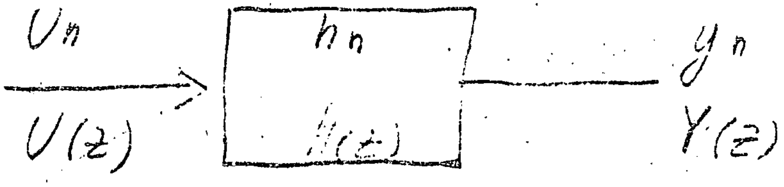


$$E_n = R_n - C_n$$

ANALISIS MATEMATICO DE DIGITAL CONTROL LOOPS



CONTINUOS



DISCRETOS

S. CONTINUOS

$$X(s) \triangleq \mathcal{L}\{X(t)\} = \int_0^{\infty} X(t)\, e^{-st}\, dt$$ T. de L.

$$y(t) = h(t) * u(t) = \int_0^{t} h(t-\sigma)\, u(\sigma)\, d\sigma$$

$$Y(s) = H(s) \cdot U(s)$$

S. DISCRETOS   TRANSFORMADA Z

$$X(z) \triangleq \mathcal{Z}\{\overset{*}{X}(t)\} \triangleq \mathcal{Z}\left\{X(t) \sum_{n=-\infty}^{\infty} \delta(t-nT)\right\}$$

$$\triangleq \sum_{i=0}^{\infty} X_i z^{-i}$$

$$= X_0 + X_1 Z^{-1} + X_2 Z^{-2} + \dots$$

$$= X(0) + X(T) Z^{-1} + X(2T) Z^{-2} + \dots$$

$$y_n = h_n * U_n = \sum_{i=0}^{n} h(n-i)\, U(i)$$

$$Y(z) = H(z) \cdot U(z)$$

donde Z = VARIABLE COMPLEJA

PROPIEDADES

i) TRANSFORMACION LINEAL

$$\mathcal{Z}\{af(t) + bg(t)\} = \sum_{n=0}^{\infty} (a f_n + b g_n) z$$

$$= a \sum_{n=0}^{\infty} f_n z^{-n} + b \sum_{n=0}^{\infty} g_n z$$

$$= a F(z) + b G(z)$$

ii) UNIT STEP

$$\mathcal{Z}(U(t)) = F_1(z)$$
$$= \sum_{n=0}^{\infty} U(nT)\, z^{-n}$$
$$= \sum_{n=0}^{\infty} z^{-n}$$
$$= 1 + z^{-1} + z^{-2} + z^{-3}$$

$$F_1(z) = \frac{1}{1-z^{-1}} \qquad \text{si } |z^{-1}| < 1$$

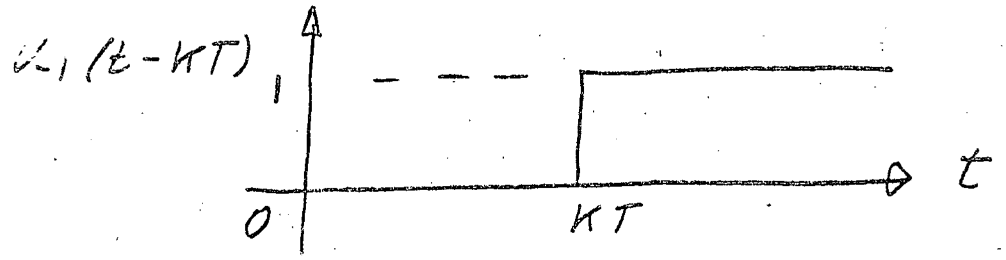
iii) CAMBIO DE ESCALA

$$\mathcal{Z}\left(a^n f(t)\right) = \sum_{n=0}^{\infty} a^n f(t)\, z^{-n}$$
$$= \sum_{n=0}^{\infty} f(t) \left(\frac{z}{a}\right)^{-n}$$
$$= F(z/a)$$

iV) MULT. POR UNA EXPONENCIAL

$$\mathcal{Z}\left\{e^{-at} U(t)\right\} = \sum_{n=0}^{\infty} e^{-anT} z^{-n}\, U(nT)$$
$$= \sum_{n=0}^{\infty} \left(e^{aT} z\right)^{-n} U(nT)$$
$$= F_1(e^{aT} z)$$
$$= \frac{1}{1-(z\,e^{aT})^{-1}}$$
$$= \frac{1}{1-e^{-aT} z^{-1}} \qquad |z^{-1}| < e^{aT}$$

V) RETRASOS EN EL TIEMPO



$$\mathcal{Z}\left\{ f(t-KT)\ U_{-1}(t-KT) \right\} = \sum_{n=0}^{\infty} f(nT-KT)\ U_{-1}(nT-KT)\, z$$

SI $m = n-K$

$$= \sum_{m=-K}^{\infty} f(mT)\ U(mT)\ z^{-(m+K)}$$

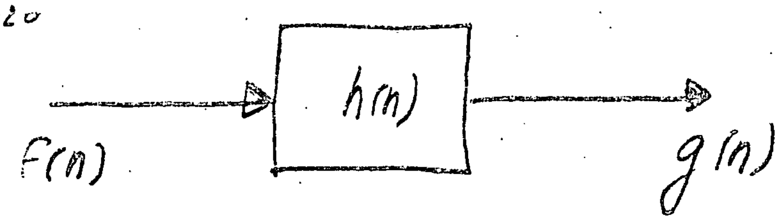
$$= z^{-K} \sum_{m=-K}^{\infty} f(mT)\ U(mT)\ z^{-m}$$

$$= z^{-K} \sum_{m=0}^{\infty} f(mT)\ z^{-m}$$

$$= z^{-K}\ F(z)$$

$$\Longrightarrow \quad \mathcal{Z}(f(t-KT) = z^{-K}\ F(z)$$

Ejemplo



$$g(n) = F(n) * h(n)$$
$$G(z) = F(z) \quad H(z)$$

$$F(n) = \begin{cases} \left(\frac{1}{2}\right)^n & n \geq 0 \\ 0 & n < 0 \end{cases}$$

$$h(n) = \begin{cases} \left(\frac{1}{3}\right)^n & n \geq 0 \\ 0 & n < 0 \end{cases}$$

a) $G(z) = F(z) \cdot H(z)$

$$G(z) = \frac{1}{1-\frac{1}{2}z^{-1}} \quad \frac{1}{1-\frac{1}{3}z^{-1}}$$

$$G(z) = \frac{3}{1-\frac{1}{2}z^{-1}} + \frac{-2}{1-\frac{1}{3}z^{-1}}$$

$$\Longrightarrow g_n = 3\left(\frac{1}{2}\right)^n - 2\left(\frac{1}{3}\right)^n \qquad n \geq 0$$
$$= 0 \qquad n < 0$$

b) $g_n = F_n * h_n$

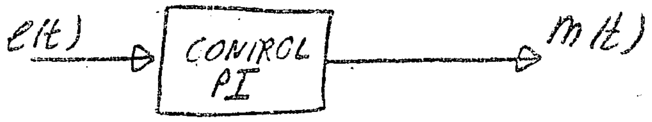
$$= \sum_{k=-\infty}^{\infty} F(n-k)\, h(k)$$

$$= \sum_{k=0}^{n} \left(\frac{1}{2}\right)^{n-k} \left(\frac{1}{3}\right)^k = \left(\frac{1}{2}\right)^n \sum_{k=0}^{n} 2^k \left(\frac{1}{3}\right)^k$$

$$= \left(\frac{1}{2}\right)^n \sum_{k=0}^{n} \left(\frac{2}{3}\right)^k = \left(\frac{1}{2}\right)^n \left[\frac{1-\left(\frac{2}{3}\right)^{n+1}}{1-\frac{2}{3}}\right]$$

$$= 3\left(\frac{1}{2}\right)^n - 2\left(\frac{1}{3}\right)^n \qquad n \geq 0$$

Z TRANSFORM OF DIFFERENCE EQUATIONS

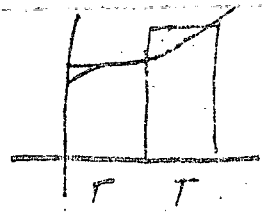


$$m(t) = K_c \left[e(t) + \frac{1}{T_i} \int e(t)\, dt \right] + M_0$$

KC = GANANCIA

M_0 = VALOR INICIAL DE M_0

T_i = RESET TIME



$$m_n = K_c \left[e_n + \frac{T}{T_i} \sum_{K=1}^{n} e_K \right] + M_0$$

$$m_{n-1} = K_c \left[e_{n-1} + \frac{T}{T_i} \sum_{K=1}^{n-1} e_K \right] + M_0$$

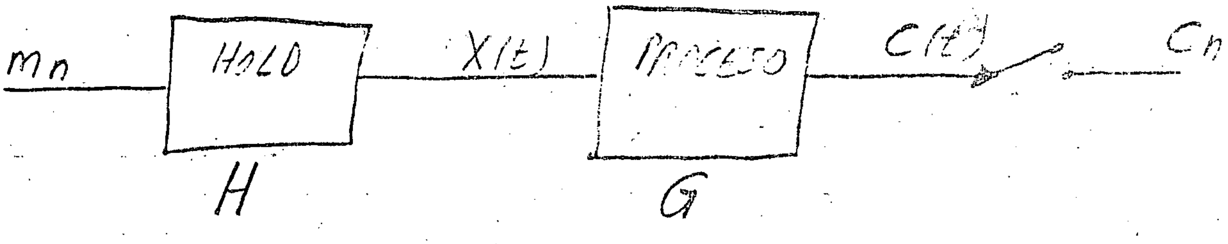
$$\Longrightarrow \quad m_n - m_{n-1} = K_c \left[e_n - e_{n-1} \right] + \frac{K_c T}{T_i} e_n$$

$$M(z) - z^{-1} M(z) = K_c \left[E(z) - z^{-1} E(z) \right] + \frac{K_c T}{T_i} E(z)$$

$$\frac{M(z)}{E(z)} = K_c \left[1 + \frac{T}{T_i} (1 - z^{-1})^{-1} \right]$$

$$\frac{M(s)}{E(s)} = K_c \left[1 + \frac{1}{T_i S} \right]$$

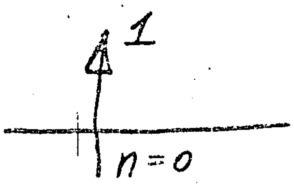
PULSE TRANSER FUNCTIONS



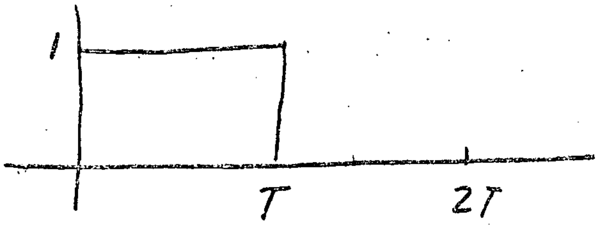
$$C(z) = [HG](z) \cdot M(z)$$

$$HG(z) = \mathcal{Z}\left[\mathcal{L}^{-1}(H(s)\, G(s))\right\}$$

51 $$p_n = \begin{cases} 1 & n=0 \\ 0 & n \neq 0 \end{cases}$$



ZERO ORDER HOLD



$$h(t) = U_{-1}(t) - U_{-1}(t-T)$$

$$H(s) = \frac{1}{s} - \frac{1}{s} e^{-sT}$$

51 $$G(s) = \frac{1}{s+1}$$

$$H(s)\, G(s) = \frac{1}{s+1}\left(\frac{1-e^{-sT}}{s}\right)$$

$$HG(z) = \mathcal{Z}\left[\mathcal{L}^{-1}\left(\frac{1-e^{-sT}}{s(s+1)}\right)\right]$$

$$= \mathcal{Z}\left[\mathcal{L}^{-1}\left(\frac{1}{s(s+1)}\right) - \mathcal{L}^{-1}\left(\frac{e^{-sT}}{s(s+1)}\right)\right]$$

$$= \mathcal{Z}\left[\mathcal{L}^{-1}\left(\frac{A}{s} + \frac{B}{s+1}\right) - \mathcal{L}^{-1}\left(\frac{e^{-sT}}{s(s+1)}\right)\right]$$

$$= \mathcal{Z}\left[\mathcal{L}^{-1}\left(\frac{1}{s} - \frac{1}{s+1}\right) - \mathcal{L}^{-1}\left(e^{-sT}\right)\left(\frac{1}{s} - \frac{1}{s+1}\right)\right]$$

$$= \mathcal{Z}\left[U_{-1}(t) - e^{-t} U_{-1}(t) - U_{-1}(t-T) - e^{-(t-T)} U_{-1}(t-T)\right.$$

$$= \frac{1}{1-z^{-1}} - \frac{1}{1-e^{-T}z^{-1}} - \frac{z^{-1}}{1-z^{-1}} - \frac{z}{1-e^{-1}z^{-1}}$$

$$= (1-z^{-1})\left[\frac{1}{1-z^{-1}} - \frac{1}{1-e^{-T}z^{-1}}\right]$$

$$= \cancel{(1-z^{-1})}\left[\frac{\cancel{1}-e^{-T}z^{-1} \cancel{-1}+z^{-1}}{\cancel{(1-z^{-1})}(1-e^{-T}z^{-1})}\right]$$

$$\frac{C(z)}{M(z)} = \frac{z^{-1}(1-e^{-T})}{(1-e^{-T}z^{-1})}$$

$$\Longrightarrow \quad C(z)\left[1-e^{-T}z^{-1}\right] = M(z)\, z^{-1}(1-e^{-T})$$

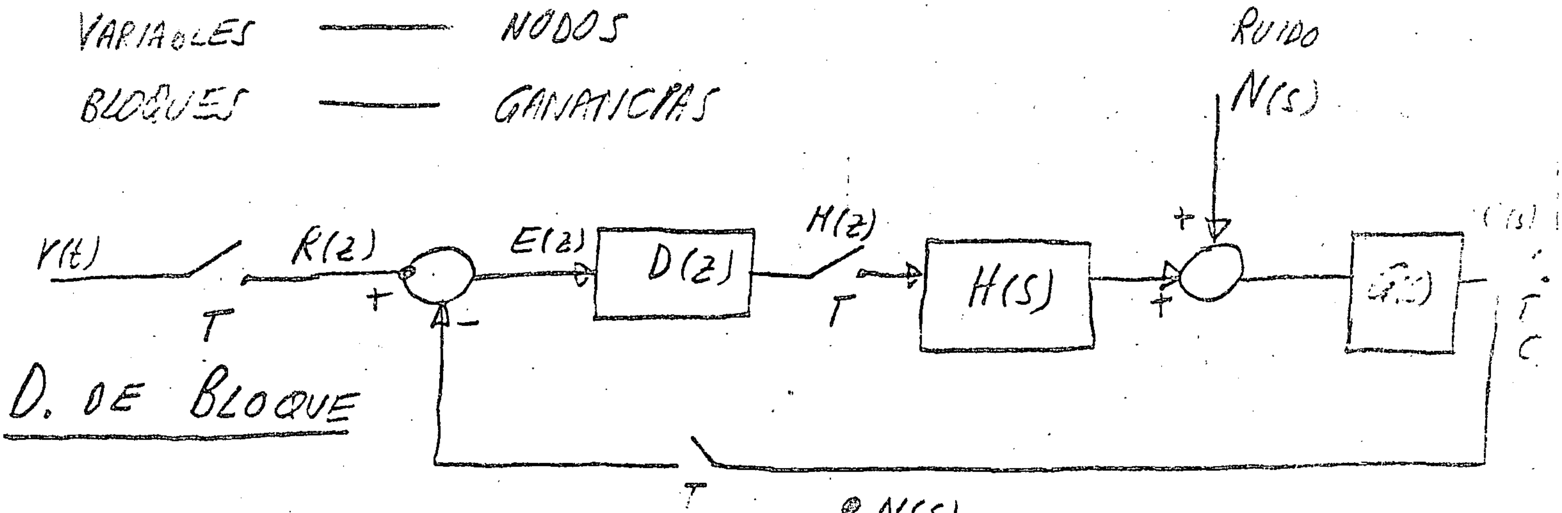
$$c_n - e^{-T}c_{n-1} = m_{n-1} - e^{-T}m_{n-1}$$

$$c_n - e^{-T}c_{n-1} = m_{n-1}(1-e^{-T})$$

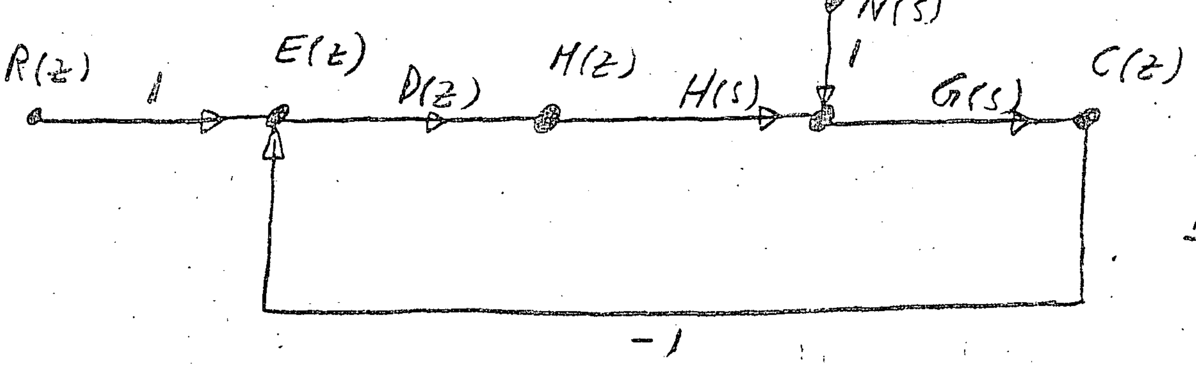
ANALYSIS DE DIAGRAMAS DE BLOQUE.

VARIABLES ——— NODOS

BLOQUES ——— GANANCIAS



D. DE BLOQUE



REOGRAMA

PARA SISTEMAS LINEALES

$\Longrightarrow$ SUPERPOSICION ES APLICABLE

$$C(z) = G_1(R(z), N(s))$$

NODO — 1 VARIABLE DEL SISTEMA

RAMA (ARISTA) — REPRESENTA RELACION CAUSA-EFECTO

NODO FUENTE

NODO POZO

TRANSMISION — VALOR DE LA OPERACION A EFECTAR SOBRE LA VARIABLE CAUSA PARA OBTENER LA VARIABLE EFECTO

REDGRAMA — NOTACION GRAFICA PARA DESCRIBIR CONJUNTOS DE RELACIONES LINEALES

TRAYECTORIA — SUBGRAFICA DE UN REDGRAMA FORMADA POR RAMAS CONECTADAS CON UNA MISMA DIRECCION (CADA NODO APARECE EN UNA SOLA OCASION)

MALLA SIMPLE — TRAYECTORIA CERRADA (CADA NODO APARECE EN UNA SOLA OCASION POR CICLO)

MALLA MULTIPLE DE ORDEN K — K MALLAS MULTIPLES QUE NO TIENEN NODO COMÚN

GANANCIA DE LA TRAYECTORIA TK — π DE LAS TRANSMISIONES DE LA TRAYECTORIA

GANANCIA DE LA MALLA — π DE LAS TRANSMISIONES DE LA MALLA

Δ DETERMINANTE $= 1 - \Sigma GM1 + \Sigma GM2 -$

Δ_K COFACTOR DE TK $=$ Δ DEL REDGRAMA QUE QUEDA AL ELIMINAR TK

TRAYECTORIA

T_K GANANCIA DE LA TRAYECTORIA

MALLA SIMPLE

GM_K MALLA DE ORDEN K

DETERMINANTE DEL REDGRAMA $= 1 - \Sigma GM1 + \Sigma GM2$

Δ_K COFACTOR DE TRAYECTORIA $=$ DET.

$$\frac{X_{\text{NO FUENTE}}}{X_{\text{FUENTE}}} = \frac{\sum_K T_K \Delta_K}{\Delta}$$

$$\frac{C(z)}{R(z)} = \frac{D(z)\,\mathcal{Z}(H(s)\,G(s)) \cdot 1}{1 + D(z)\,\mathcal{Z}(H(s)\,G(s))}$$

$$C(z) = \frac{\mathcal{Z}(N(s)\,G(s))}{1 + D(z)\,\mathcal{Z}(H(s)\,G(s))}$$

$$\Longrightarrow C(z) = \frac{D(z)\,\mathcal{Z}(H(s)\,G(s))\cdot R(z) + \mathcal{Z}(N(s)\,G(s))}{1 + D(z)\,\mathcal{Z}(H(s) \cdot G(s))}$$

EXAMPLE :

$D(z) = K$ CONTROLADOR PROPORCIONAL

$H(s) = \dfrac{1 - e^{-sT}}{s}$ ZERO ORDER HOLD

$G(s) = \dfrac{1}{s}$ INTEGRADOR

$N(s) = 0$

$r(t) = U_{-1}(t)$ ESCALÓN

$$\mathcal{Z}\left(\frac{1 - e^{-sT}}{s^2}\right) = \mathcal{Z}\left(U_{-2}(t) - U_{-2}(t-T)\right)$$

$$= \mathcal{Z}\left(t\,U_{-1}(t) - (t-T)\,U_{-1}(t-T)\right)$$

$$f_{nT} = nT \qquad nT \geq 0$$

$$= 0 \qquad nT < 0$$

$$= F(z) - z^{-1}F(z) = (1 - z^{-1})\,F(z)$$

$$F(z) = Tz^{-1} + 2Tz^{-2} + 3Tz^{-3} + \ldots$$

$$= Tz\,(z^{-2} + 2z^{-3} + 3z^{-4} + \ldots$$

$$= -zT\,\frac{d}{dz}\,(z^{-1} + z^{-2} + z^{-3} + \ldots$$

$$= -Tz\,\frac{d}{dz}\,(z^{-1}(1 + z^{-1} + z^{-2} + \ldots$$

$$= -Tz\,\frac{d}{dz}\,\frac{z^{-1}}{1-z^{-1}}$$

$$= -Tz\left[\frac{(1-z^{-1})z^{-2} - z^{-1}(z^{-2})}{(1-z^{-1})^2}\right]$$

$$= \frac{-Tz\,(-z^{-2})}{(1-z^{-1})^2} = \frac{z^{-1}T}{(1-z^{-1})^2}$$

$$\Longrightarrow \quad \frac{C(z)}{R(z)} = \frac{\frac{K\,z^{-1}T}{(1-z^{-1})}}{1 + \frac{(K)(z^{-1}T)}{(1-z^{-1})}} = \frac{KTz^{-1}}{(1-z^{-1}) + Kz^{-1}T}$$

$$\frac{C(z)}{R(z)} = \frac{KTz^{-1}}{1 + z^{-1}(KT-1)}$$

$$\Longrightarrow \quad C(z)(1 + z^{-1}(KT-1)) = R(z)\,KTz^{-1}$$

$$C_n + C_{n-1}KT - C_{n-1} = KT\,r(n-1)$$

$$C_n = KT\,r(n-1) + C_{n-1}(1-KT) \quad \text{———} \quad ①$$

TRANSFORMADAS INVERSAS.

$$F(z) = \frac{a_0 + a_1 z + a_2 z^2 + \ldots + a_m z^m}{b_0 + b_1 z + b_2 z^2 + \ldots\; b_n z^n}$$

$$F(z) = C_0 + C_1 z^{-1} + C_2 z^{-2} + \ldots$$

$$F(t) = f(0) + f(T)\,z^{-1} + f(2T)\,z^{-2} + \ldots$$

EXAMPLE

$$F(z) = \frac{3}{(1-z^{-1})^{2}(1-0.5z^{-1})}$$

$$F(z) = \frac{A}{(1-z^{-1})^{2}} + \frac{B}{1-z^{-1}} + \frac{C}{1-0.5z^{-1}}$$

$$A_K = \frac{1}{(K-1)!}\left[\frac{d^{K-1}}{dz^{K-1}} F(z)\right]_{z=z_K}$$

$$\Rightarrow \quad A = \frac{3}{(1-0.5z^{-1})}\bigg|_{z^{-1}=1} = 6$$

$$B = \frac{d}{dz}\,\frac{3}{1-0.5z^{-1}}\bigg|_{z^{-1}=1} = \frac{-3(0.5z^{-2})}{(1-0.5z^{-1})^{2}} = \frac{-3/2}{1/4}$$

$$= -6$$

$$C = \frac{3}{(1-z^{-1})^{2}}\bigg|_{z^{-1}=2} = 3$$

$$F(z) = \frac{6}{(1-z^{-1})^{2}} - \frac{6}{1-z^{-1}} + \frac{3}{1-0.5z^{-1}} =$$

$$F(z) = \frac{6z^{-1}}{(1-z^{-1})^{2}} + \frac{3}{1-0.5z^{-1}}$$

$$\Rightarrow \quad f(t) = 6t\,u_{-1}(t) + 3e^{-\alpha t}u_{-1}(t)$$

$$e^{-\alpha} = 0.5$$

$$-\alpha = \ln 0.5$$

$$\alpha = 0.694$$

ESTABILIDAD

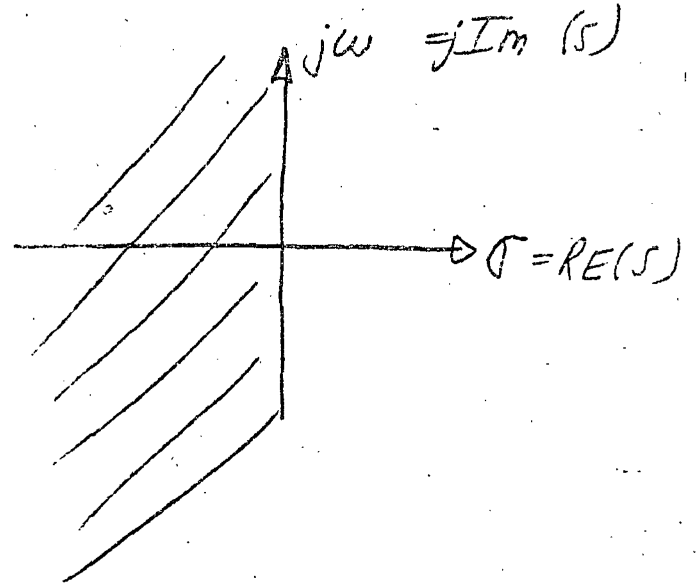
$$z = e^{Ts}$$

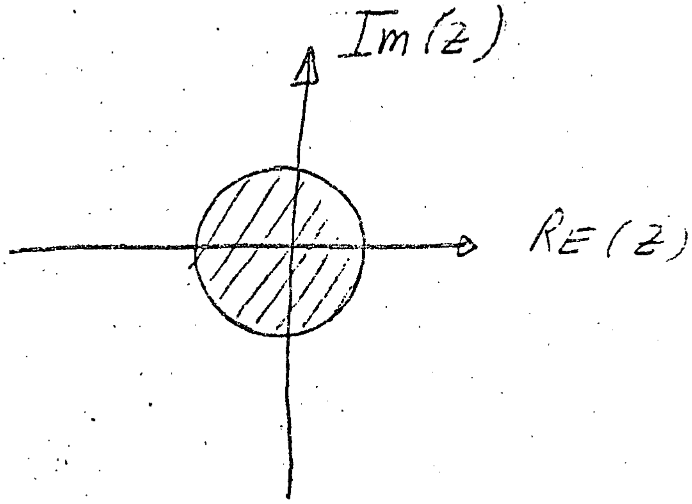
$$\ln z = Ts$$

$$\frac{1}{T}\ln z = s$$

$\Longrightarrow$ SI PARA ESTABILIDAD $RE(s) < 0$

$\Longrightarrow$ $|z| < 1$



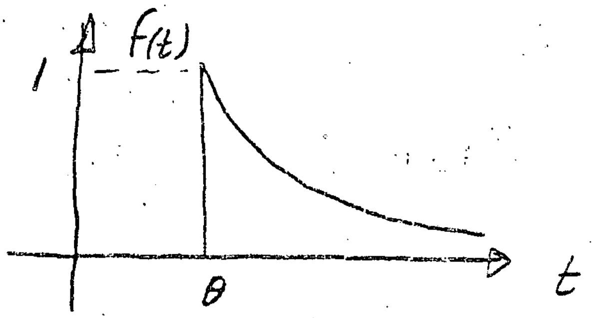


SYSTEMS WITH DEAD TIME

$$\mathcal{Z}\left[G(s)\,e^{-\theta s}\right] = \mathcal{Z}_m\left[G(s)\right] = G(z,m)$$

donde $m = 1 - \theta/T$

$mT = T - \theta$



$$\mathcal{Z}_m\left[e^{-at}\right] = \mathcal{Z}\left[\frac{e^{-\theta s}}{s+a}\right] = \mathcal{Z}\left[e^{-a(t-\theta)}\,u_{-1}(t-\right.$$

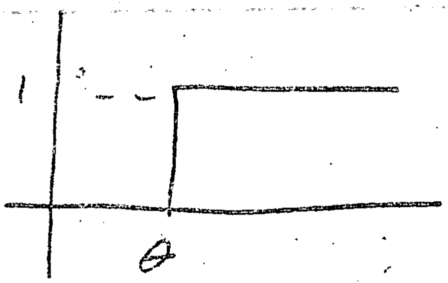
$$F(z) = \sum_{n=0}^{\infty} e^{+a(nT-\theta)} U_{-1}(nT-\theta) \; z^{-n}$$

$$= e^{-a(T-\theta)} z^{-1} + e^{-a(2T-\theta)} z^{-2} + \ldots$$

$$= e^{-amT} z^{-1} + e^{-amT} e^{-aT} z^{-2}$$

$$= e^{-amT} \left[z^{-1} + e^{-aT} z^{-2} + e^{-2aT} z^{-3} \right.$$

$$= e^{-amT} z^{-1} \left[\frac{1}{1 - e^{-aT} z^{-1}} \right]$$



$$\mathcal{Z}m \left\{ U_{-1}(t) \right\} = \mathcal{Z}\left[U_{-1}(t-\theta) \right] = \mathcal{Z}\left[\frac{e^{-\theta s}}{s} \right]$$

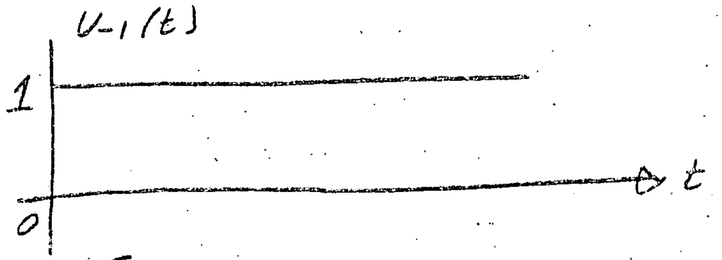
$$= \sum_{n=0}^{\infty} U_{-1}(nT-\theta) \; z^{-n}$$

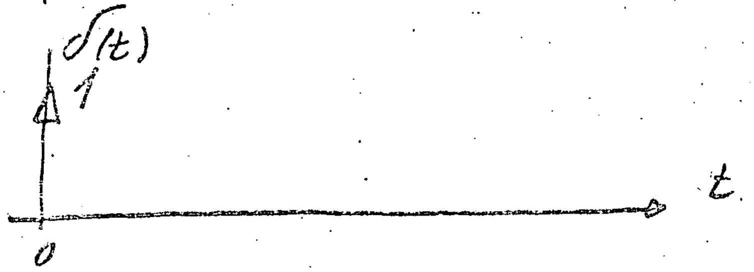
$$= z^{-1} + z^{-2} + z^{-3}$$

$$= \frac{z^{-1}}{1 - z^{-1}}$$

VER EJEMPLOS LIBRO 103-107

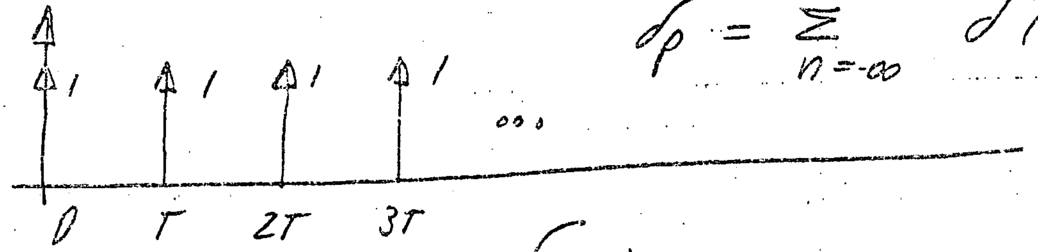
FREQUENCY SPECTRUM



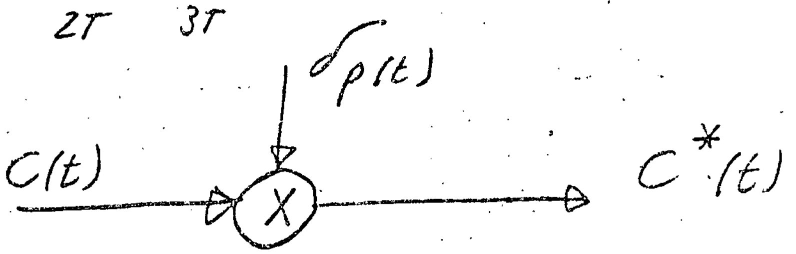


$$\delta(t) = \frac{d}{dt} u_{-1}(t)$$

$$u_{-1}(t) = \int_0^{\infty} \delta(t)\, dt = 1 \quad \forall\ t > 0$$

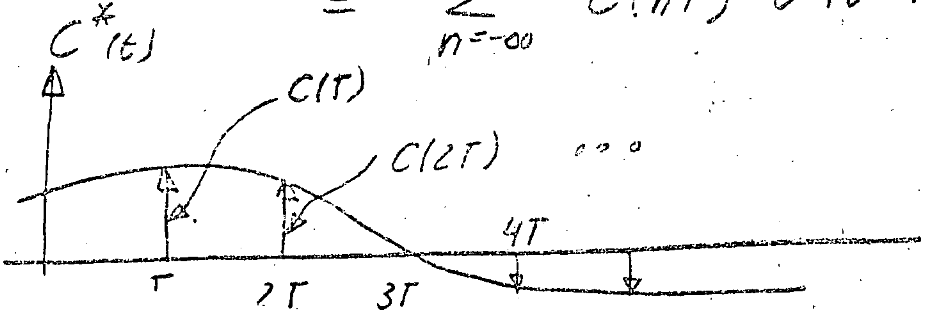


$$\delta_p = \sum_{n=-\infty}^{\infty} \delta(t-nT)$$



$$\therefore\ c^*(t) = c(t) \sum_{n=-\infty}^{\infty} \delta(t-nT)$$

$$= \sum_{n=-\infty}^{\infty} c(nT)\, \delta(t-nT)$$



$$\mathcal{L}\{c^*(t)\} = C^*(s) = \int_0^{\infty}\left[\sum_{n=-\infty}^{\infty} c(nT)\,\delta(t-nT)\right]e^{-st}\,dt$$

$$= \sum_{n=0}^{\infty} C(nT)\, e^{-nTs}$$

$$= \sum_{n=0}^{\infty} C(nT)\, z^{-n}$$

$$SI \quad z = e^{Ts}$$

$$= \mathcal{Z}(C(t))$$

$$c^*(t) = \delta_p(t)\cdot C(t)$$

$$\delta_p(t) = \sum_{n=-\infty}^{\infty} C_n\, e^{+jn\,2\pi F_0 t}$$ SERIE DE FOURIER

$$C_n = \frac{1}{T}\int_0^T \delta_p(t)\, e^{-jn\,2\pi F_0 t}\,dt$$

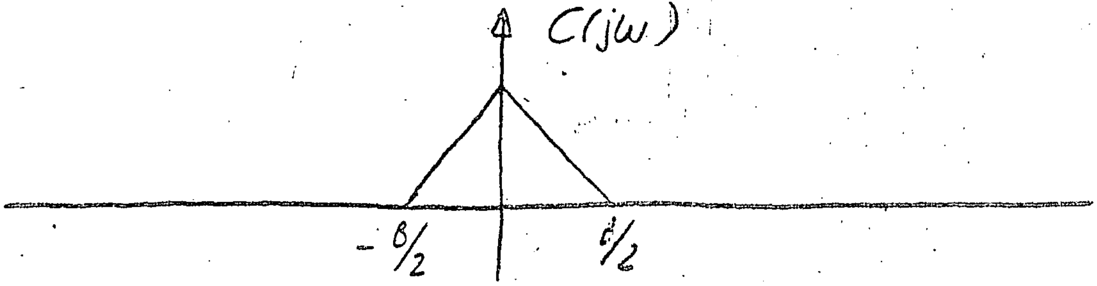
$$C_n = \frac{1}{T}$$

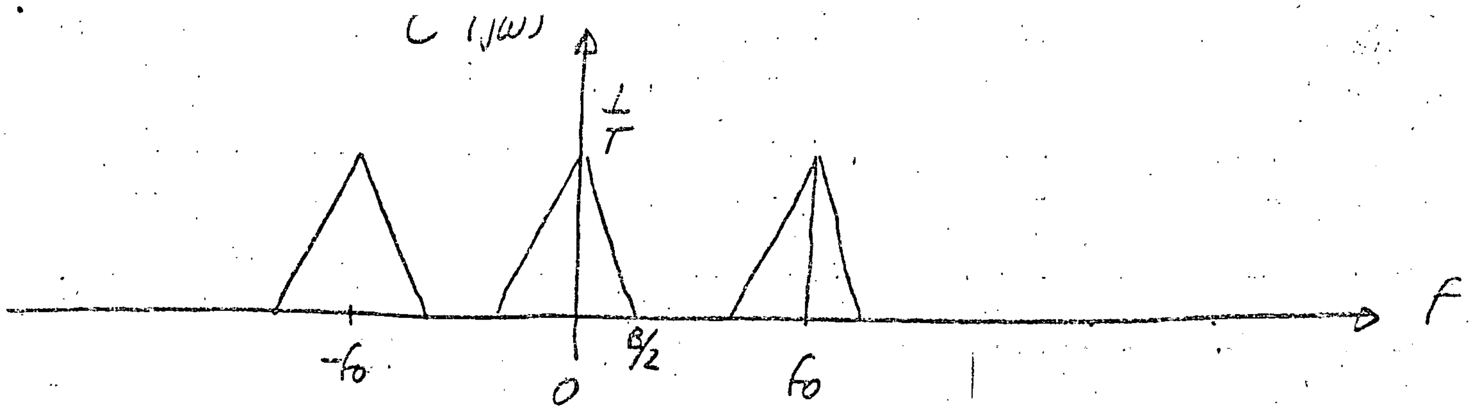
$$\Rightarrow \delta_p(t) = \sum_{n=-\infty}^{\infty} \frac{1}{T}\, e^{+j2\pi n F_0 t}$$

$$\Rightarrow c^*(t) = \sum_{n=-\infty}^{\infty} \frac{1}{T}\, e^{j2\pi n F_0 t}\cdot C(t)$$

$$C^*(j\omega) = \frac{1}{T}\sum_{n=-\infty}^{\infty} \mathcal{F}\{e^{j2\pi n F_0 t}\, C(t)\}$$

$$= \frac{1}{T}\sum_{n=-\infty}^{\infty} C(j\omega - j2\pi n F_0)$$





$\Longrightarrow$ $f_0 > B$ PARA RECUPERAR $C(jw)$

$$f_0 > 2\left(\frac{B}{2}\right)$$

DATA HOLDS

RECONSTRUYE UNA SEÑAL CONTÍNUA $m(t)$ A PARTIR DE LA SECUENCIA DE NUMEROS m_n

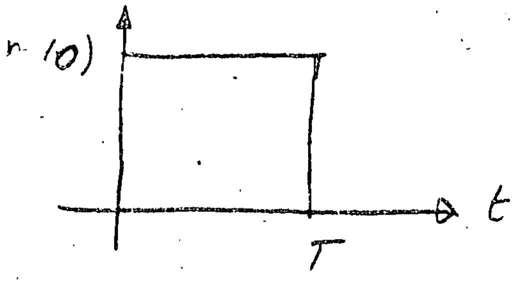
1) FACILIDAD PARA RECONSTRUIR LA SEÑAL
2) EFECTO QUE PRODUCE SOBRE LA MALLA DE CONTROL
3) ASPECTOS PRACTICOS SOBRE SU CONSTRUCCION

$m(t)$ EXPANDIDA EN SERIE DE TAYLOR :

$$m(t) = m(nT)$$
$$+ \frac{m(nT) - m(n-1)T}{T}\,(t - nT)$$
$$+ \;ooo$$

ZERO ORDER HOLD

$m(t) = m(nT)$ $nT \leq t\ (n+1)\,T$



$m(t) = U_{-1}(t) - U_{-1}(t-T)$

$M(s) = \frac{1 - e^{-sT}}{s}$

$M(j\omega) = \frac{1 - e^{-j\omega T}}{j\omega}$

$M(j\omega) = \frac{2e^{-j\omega T/2}}{2j\omega}\left[e^{+j\omega T/2} - e^{-j\omega T/2}\right]$

$M(j\omega) = \frac{2e^{-j\omega T/2}}{\omega}\left[\frac{\cancel{\cos} + j\,\text{sen} - \cancel{\cos\theta} + j\,\text{sen}\,(\theta)}{2j}\right]$

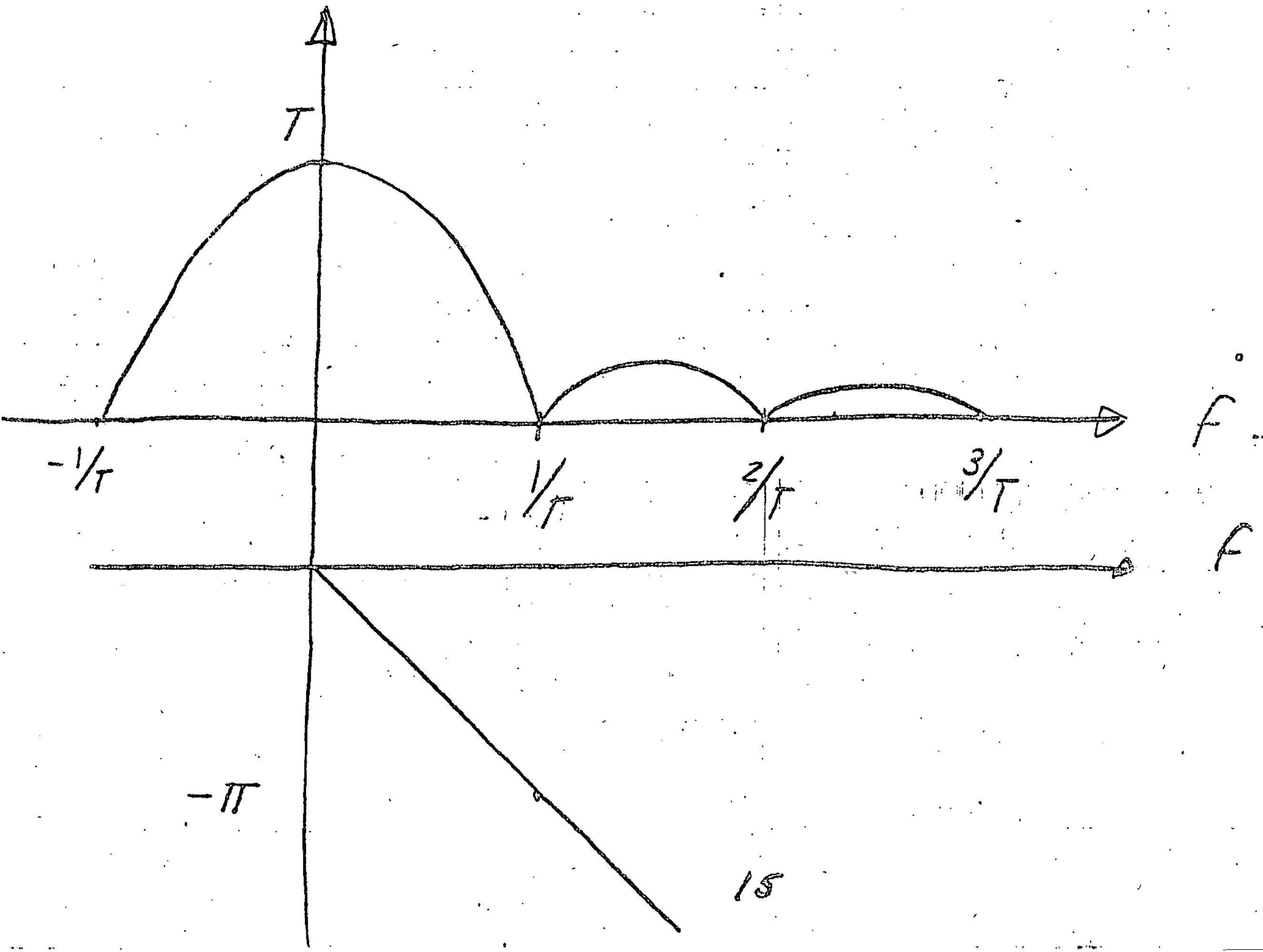
$M(j\omega) = Te^{-j\omega T/2}\;\frac{\text{sen}\;\omega T/2}{\omega T/2}$

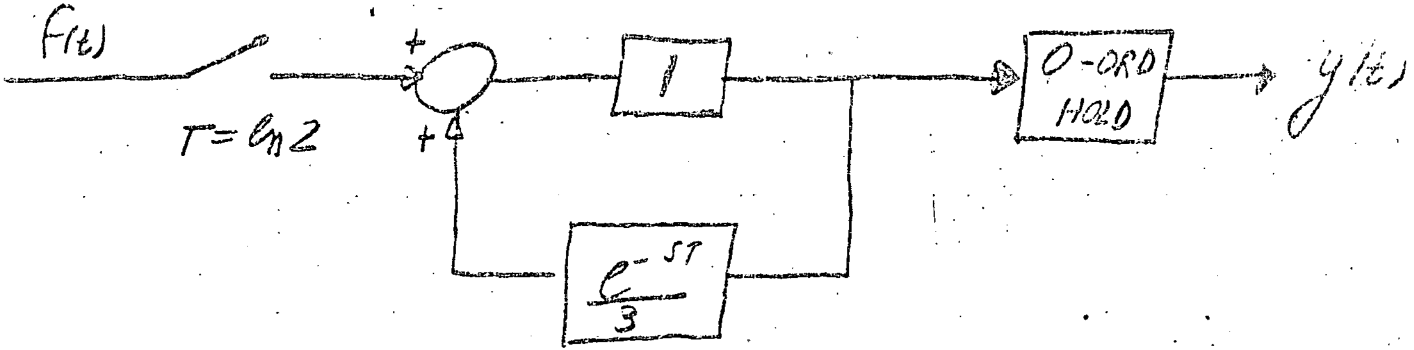
$M(f) = T\,\frac{\text{sen}\,\frac{2\pi f T}{2}}{\frac{2\pi f T}{2}}\,e^{-j\omega T/2} = T\,\text{sinc}\;fT\,e^{-j\pi Tf}$

$M(f) = M\angle\phi$

$\Rightarrow\; M = 1\cdot T\,\text{sinc}\;Tf$

$\phi = -\pi T f$





$$f(t) = e^{-t}$$

$$f^*(t) = e^{-t} \sum_{n=0}^{\infty} \delta(t-nT)$$

$$= \sum_{n=0}^{\infty} e^{-nT} \delta(t-nT)$$

$$= 1 + e^{-\ln 2} z^{-1} + e^{-2\ln 2} z^{-2} + \cdots$$

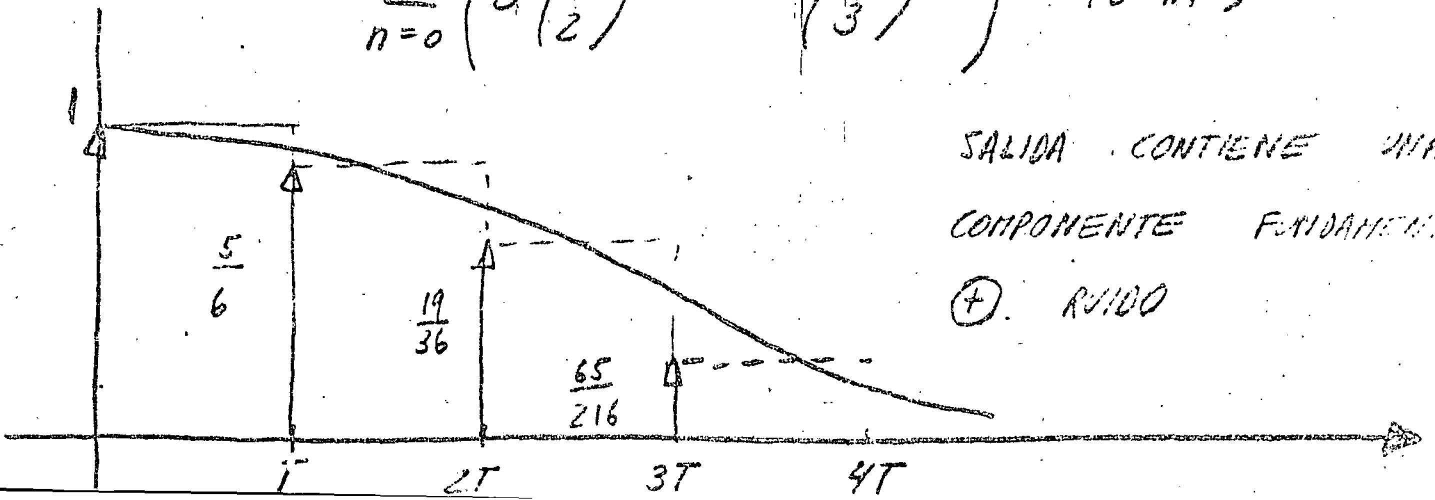
$$= \frac{1}{1-\frac{1}{2}z^{-1}}$$

INPUT TO ZERO ORDER HOLD

$$\left(\frac{1}{1-\frac{1}{2}z^{-1}}\right) \mathcal{Z}\left(\frac{1}{1-\frac{e^{-sT}}{3}}\right) = \left(\frac{1}{1-\frac{1}{2}z^{-1}}\right)\left(\frac{1}{1-\frac{z^{-1}}{3}}\right)$$

$$= \frac{3}{1-\frac{1}{2}z^{-1}} + \frac{-2}{1-\frac{1}{3}z^{-1}}$$

TIEMPO $\sum_{n=0}^{\infty} \left(3\left(\frac{1}{2}\right)^n - 2\left(\frac{1}{3}\right)^n\right) \delta(t-nT)$



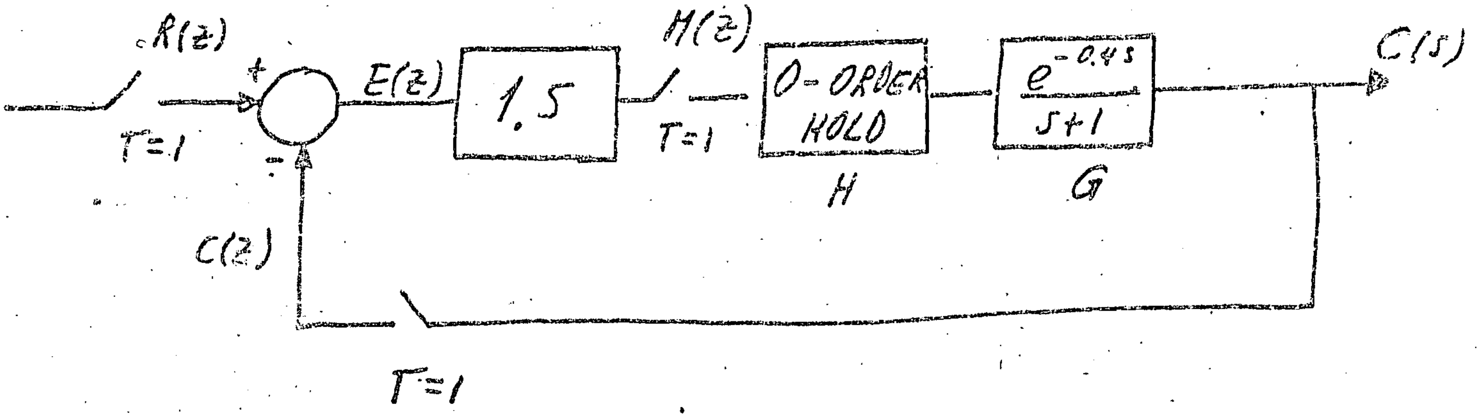
SALIDA CONTIENE UNA COMPONENTE FUNDAMENTAL ⊕ RUIDO

BIBLIOGRAFIA :

1) DIGITAL SIGNAL PROCESSING
ALAN V. OPPENHEIM / RONALD W. SCHAFER
PRENTICE HALL
2) TRANSFORM AND STATE VARIABLE METHODS IN LINEAR SYSTEMS
S. C. GUPTA
JOHN WILEY AND SONS. INC.
3) MATHEMATICS OF SAMPLED DATA SYSTEMS

EJEMPLO

TRANSFORMADA Z MODIFICADA.



$$\frac{C(z)}{R(z)} = \frac{1.5\ HG(z)}{1+1.5\ HG(z)}$$

SI $H(s) = \frac{1-e^{-sT}}{s}$

$$G(s) = \frac{e^{-0.4s}}{s+1}$$

$$HG(z) = \mathcal{Z}\left[\frac{1-e^{-sT}}{s}\cdot\frac{e^{-0.4s}}{s+1}\right]$$

$$HG(z) = (1-z^{-1})\,\mathcal{Z}\left[\frac{e^{-0.4s}}{s(s+1)}\right]$$

$$HG(z) = (1-z^{-1})\,\mathcal{Z}m\left[\frac{1}{s(s+1)}\right]$$

DONDE $m = 1-\theta/T = 1-0.4/1 = 0.6$

$$\Rightarrow HG(z) = (1-z^{-1})\,\mathcal{Z}m\left[\frac{1}{s}-\frac{1}{s+1}\right]$$

$$HG(z) = (1-z^{-1})\left[\frac{z^{-1}}{1-z^{-1}} - \frac{e^{-mT}z^{-1}}{1-e^{-T}z^{-1}}\right]$$

$$HG(z) = z^{-1}\left[\frac{(1-e^{-mT}) + z^{-1}(e^{-mT}-e^{-T})}{1-e^{-T}z^{-1}}\right]$$

$$HG(z) = \frac{z^{-1}(0.45 + 0.182 z^{-1})}{1 - 0.368 z^{-1}}$$

$$\Longrightarrow \quad \frac{C(z)}{R(z)} = \frac{1.5 z^{-1}(0.45 + 0.182 z^{-1})}{1 - 0.368 z^{-1} + 1.5 z^{-1}(0.45}$$

$$\frac{C(z)}{R(z)} = \frac{z^{-1}(1.125 + 0.455 z^{-1})}{1 + 0.757 z^{-1} + 0.455 z^{-2}}$$

$$\Longrightarrow \quad C_n + 0.757 C_{n-1} + 0.455 C_{n-2}$$

$$= R_{n-1} 1.125 + 0.455 R_{n-2}$$

centro de educación continua
división de estudios superiores
facultad de ingeniería, unam

INGENIERIA DE CONTROL DIGITAL DE PROCESOS Y APLICACIONES

TECNICAS DE IDENTIFICACION

M. EN C. RAFAEL LOPEZ

NOVIEMBRE 1978.

Palacio de Minería Calle de Tacuba 5, primer piso. México 1, D. F.

TECNICAS DE IDENTIFICACION

PROBLEMA : Encontrar un modelo matemático válido para un proceso dado.

OBJETIVO : Estimar los parámetros que intervienen en una estructura propuesta, en base a mediciones de entradas y salidas del proceso en cuestión.

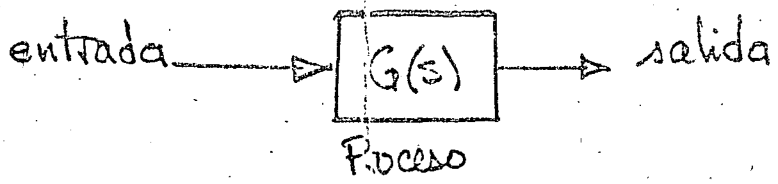
INTRODUCCION

El estudio y comprensión de un sistema particular se facilita en gran medida si se cuenta con un modelo matemático "válido" que lo describa. Asimismo, la mayoría de los esquemas de control se basan en un modelo del sistema que, aunque simplificado, ofrece una descripción adecuada para los fines que se persiguen. De ahí la importancia de disponer de un buen modelo. Conviene distinguir dos partes principales de un modelo:

$$\text{MODELO}\begin{cases}\text{- Estructura}\\ \text{- Valor de los parámetros}\end{cases}$$

EJEMPLO

De un proceso dado se puede suponer que una descripción mediante un sistema de primer orden más un retraso es adecuada.



Tenemos entonces :

Estructura : $G(s) = \dfrac{k e^{-\theta s}}{\tau s + 1}$

Parámetros :

K	ganancia
θ	tiempo de retraso
τ	constante de tiempo

Supongamos que en alguna forma se ha determinado que :

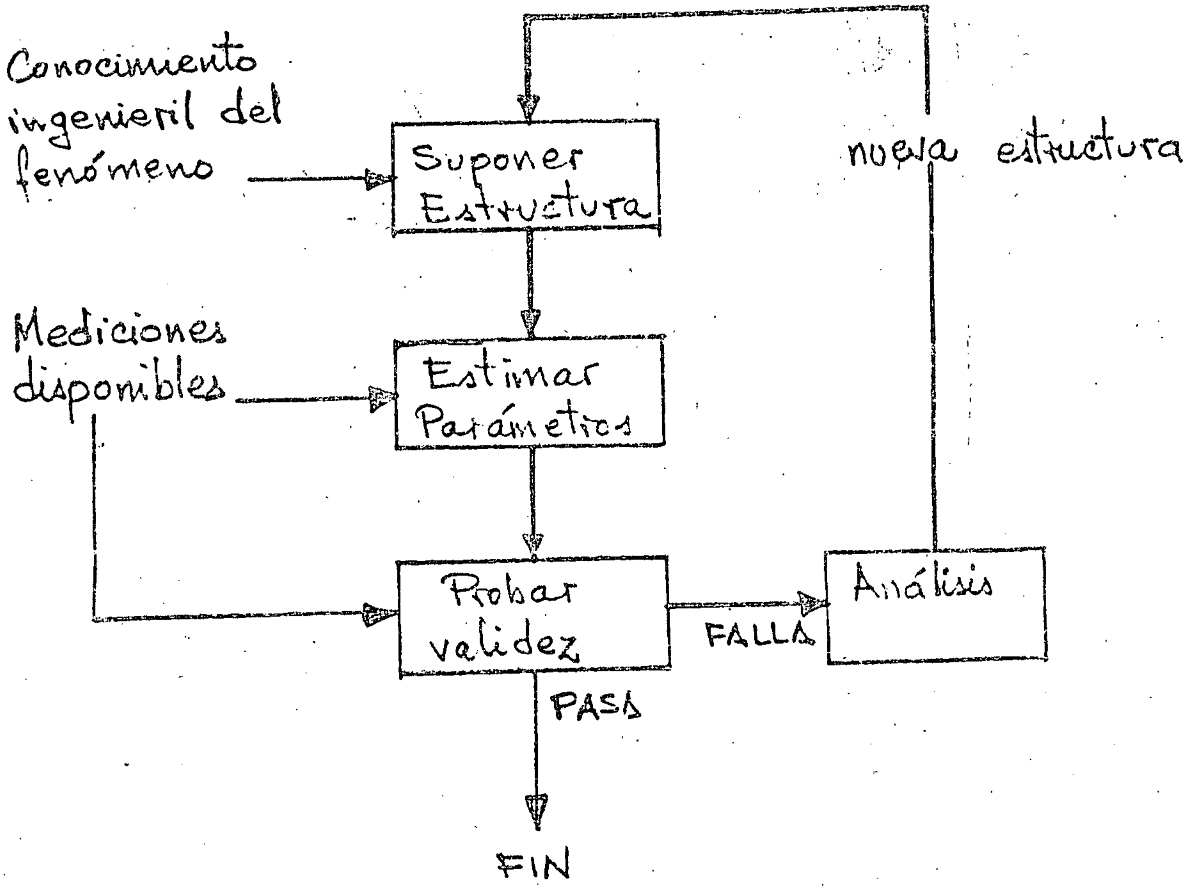
$$K = 3$$
$$\theta = 1.5 \ \text{seg}$$
$$\tau = 2 \ \text{seg}^{-1}$$

El **MODELO** del proceso es entonces

$$G(s) = \frac{3 e^{-1.5 s}}{2s + 1}$$

MODELO = ESTRUCTURA + VALORES DE PARAMETROS

En base a lo anterior podemos describir la forma general del problema de IDENTIFICACION DE SISTEMAS, en la siguiente figura :



== IDENTIFICACION DE SISTEMAS ==
(4 ETAPAS)

En esta sesión nos ocuparemos de la segunda etapa del problema, que puede enunciarse como:

Dado una supuesta estructura matemática, estimar los parámetros que intervienen en ella, en base a mediciones de entrada - salida del proceso, en forma tal que el "error sea mínimo".

Si bien es cierto que las técnicas en el dominio de la frecuencia (por ejemplo diagramas de Bode) hacen más énfasis en la determinación de un modelo adecuado, éstas no son fáciles de implementar en línea y por ello nos enfocaremos al estudio en el dominio del tiempo (figura de la pág. 3). Aquí es mas factible el uso de técnicas en línea, aunque existe el problema de que no se puede mejorar con facilidad la estructura propuesta, en caso de resultar inaceptable.

Se hará énfasis en la utilidad de los modelos discretos, que proporcionan métodos rápidos y eficientes en línea.

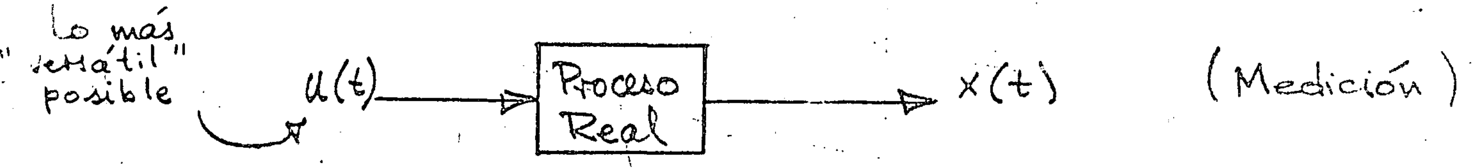
Se verá también que el problema se reduce a uno de regresión no lineal (o lineal en casos especiales)

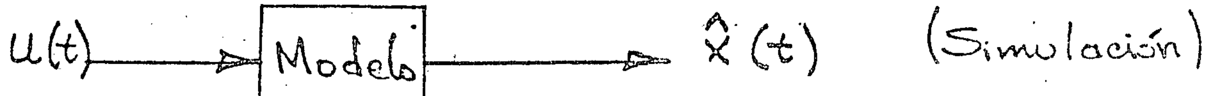
METODO

Se supone que se ha propuesto una estructura en base al conocimiento físico del fenómeno. Esta deberá incluir, en lo posible, retrasos, no linealidades, orden del sistema etc.

Se dispone además de un <u>PROCESO DE PRUEBA</u>

consistente en medir la entrada y la salida del proceso durante cierto tiempo.





Problema

Encontrar valores de los parámetros en forma tal que una <u>función del error</u>

$$e(t) \triangleq x(t) - \hat{x}(t)$$

<u>sea mínima</u>.

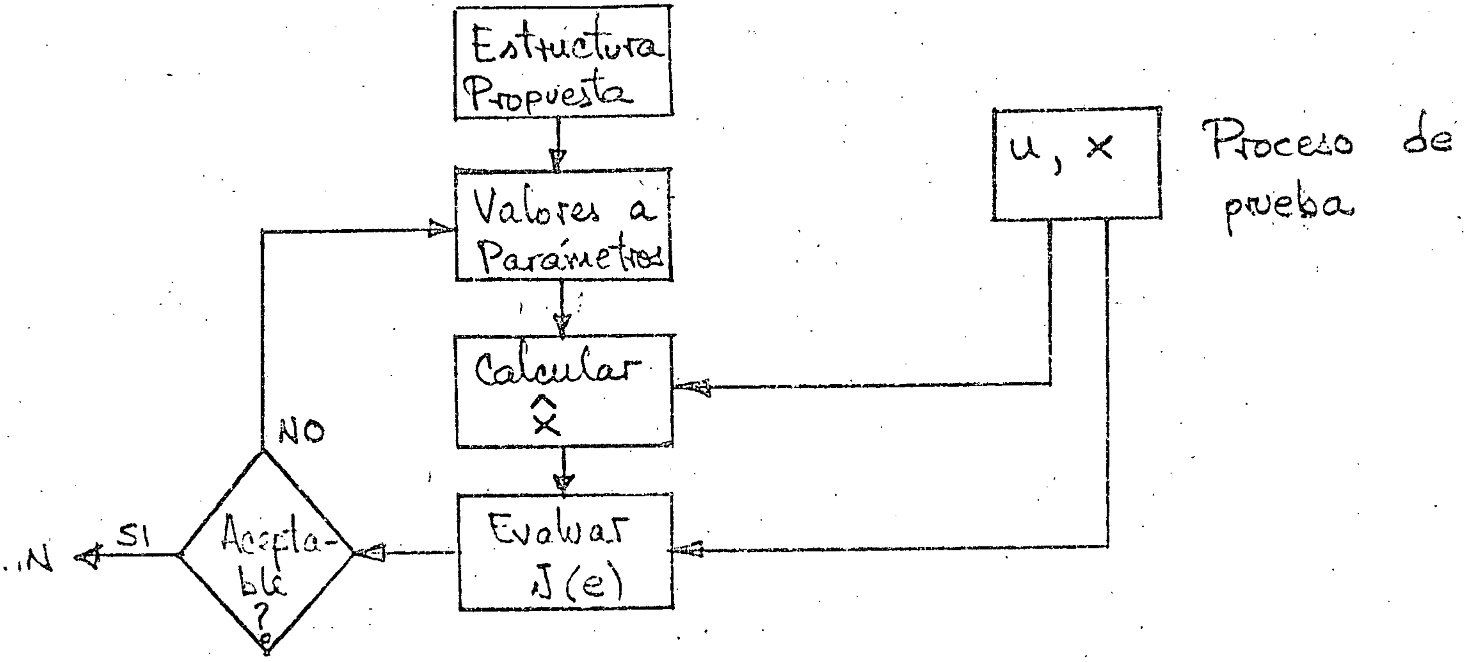
En tiempo discreto se tendrían secuencias x_i, $\hat{x}_i$ y el error

$$e_i = x_i - \hat{x}_i(t)$$

Matemáticamente: el problema es encontrar

$$\min J(e_i) \qquad J = \text{función del error}$$

En la práctica no se obtiene el mínimo exactamente y lo que se busca es que la función $J(e_i)$ sea menor que un valor fijado de antemano, o bien estar en una vecindad pequeña del mínimo.



ESTIMACION DE PARAMETROS

Criterios de error (forma de la función J

1- Mínimos cuadrados (mayor peso a errores grandes)

$$J = \int e^2(t)\,dt \qquad \text{(continuo)}$$

$$J = \sum e_i^2 \qquad \text{(discreto)}$$

2- Valor absoluto (igual peso a todas los errores)

$$J = \int |e(t)|\,dt$$

$$J = \sum |e_i|$$

3- Minimax (basado en el error máximo)

$$J = \min\left[\max_t \{e(t)\}\right]$$

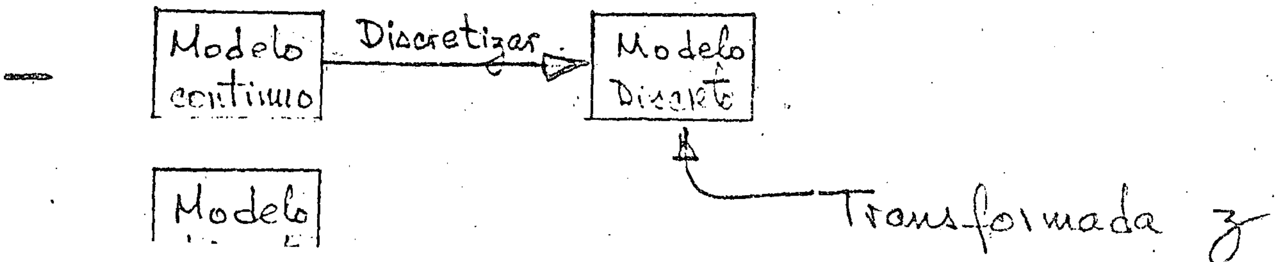
$$J = \min\left[\max_i \{e_i\}\right]$$

Utilizaremos MINIMOS CUADRADOS.

Note que el valor final de los parámetros dependerá del criterio empleado.

MODELOS DISCRETOS

Dos posibilidades al plantear un modelo :



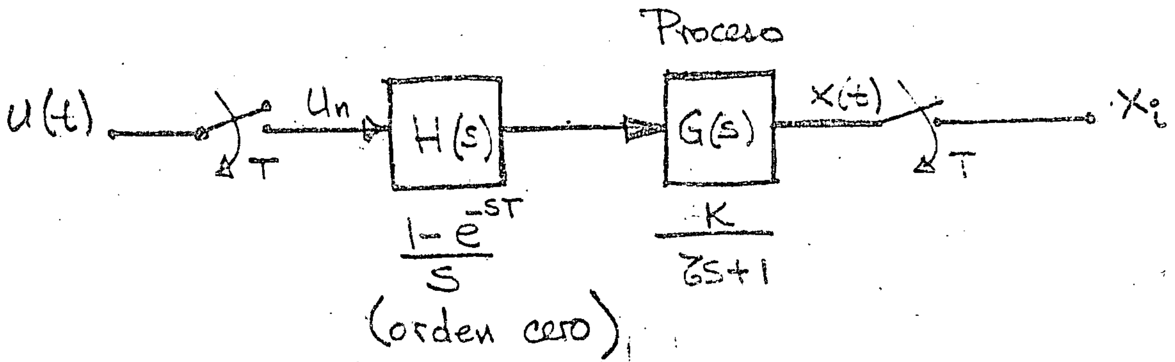
Ventajas del modelo discreto

- Más fácil de resolver ($\hat{x}_n$)
- En algunos casos permite usar regresión lineal

ESTRATEGIA GENERAL

En general se acostumbra plantear un modelo lineal alrededor de algún punto de equilibrio. En consecuencia el modelo (estructura más parámetros) varía si la operación del sistema se mueve a otro punto de equilibrio. De ahí que sea importante usar técnicas de identificación en línea

EJEMPLO



$$\frac{X(z)}{U(z)} = \frac{K\left(1-e^{-T/\tau}\right)z^{-1}}{1-z^{-1}e^{-T/\tau}}$$

ESTRUCTURA $x_{i+1} = e^{-T/\tau}x_i + K\left(1-e^{-T/\tau}\right)u_i + D$

$$x_{i+1} = a x_i + b u_i + D$$

sesgo (bias)

Problema de identificación (estimación):

Encontrar K, θ y τ óptimos

$$\text{Alternativas}\left\langle\begin{array}{l} \hat{x}_{i+1} = a x_i + b u_i + D \\ \\ \hat{x}_{i+1} = a \hat{x}_i + b u_i + D \quad \text{(independiente del proceso real } x \text{)} \end{array}\right.$$

Usaremos $\hat{x}_{i+1} = a x_i + b u_i + D$

porque produce una regresión lineal.

Mínimos cuadrados:

$$\min_{a,b,D} \sum e_i^2 = \min \sum \left(x_{i+1} - \hat{x}_{i+1}\right)^2$$

$$= \min_{a,b,D} \sum \left(x_{i+1} - a x_i - b u_i - D\right)^2$$

Note que es intuitivamente correcto.

Tomando parciales con respecto a a, b y D e igualando a cero:

$$\text{Regresión lineal}\left\{\begin{array}{l} a \sum x_i^2 + b \sum x_i u_i + D \sum x_i = \sum x_{i+1} x_i \\ a \sum x_i u_i + b \sum u_i^2 + D \sum u_i = \sum x_{i+1} u_i \\ a \sum x_i + b \sum u_i + N D = \sum x_{i+1} \end{array}\right.$$

N = numero de puntos

Tres ecuaciones con

= Otra forma de obtener el mismo modelo:

DIFERENCIAS FINITAS:

$$\tau \dot{x} + x = Ku + D' \Longrightarrow \tau \frac{x_{i+1} - x_i}{T} + x_i = Ku_i + D'$$

$$x_{i+1} = \left(1 + \frac{T}{\tau}\right) x_i + \frac{KT}{\tau} u_i + \frac{D'T}{\tau}$$

o $$x_{i+1} = a x_i + b u_i + D$$

(Válido si T es pequeño.)

= Simplificación (no estimar D)

$$(\hat{x}_{i+1} - \hat{x}_i) = a(x_i - x_{i-1}) + b(u_i - u_{i-1})$$

= Otra alternativa:

$$\overline{X}_{i+1} = a \overline{X}_i + b \overline{U}_i + D$$

donde $$\overline{x}_{i+1} = \frac{1}{i} \sum_{j=1}^{i+1} x_j$$

$$\overline{X}_i = \frac{1}{i} \sum_{j=0}^{i} x_j \quad ; \quad \overline{U}_i = \frac{1}{i} \sum_{j=0}^{i} u_j$$

Esto se basa fundamentalmente en la integral de los datos $\int x(t)\,dt$.

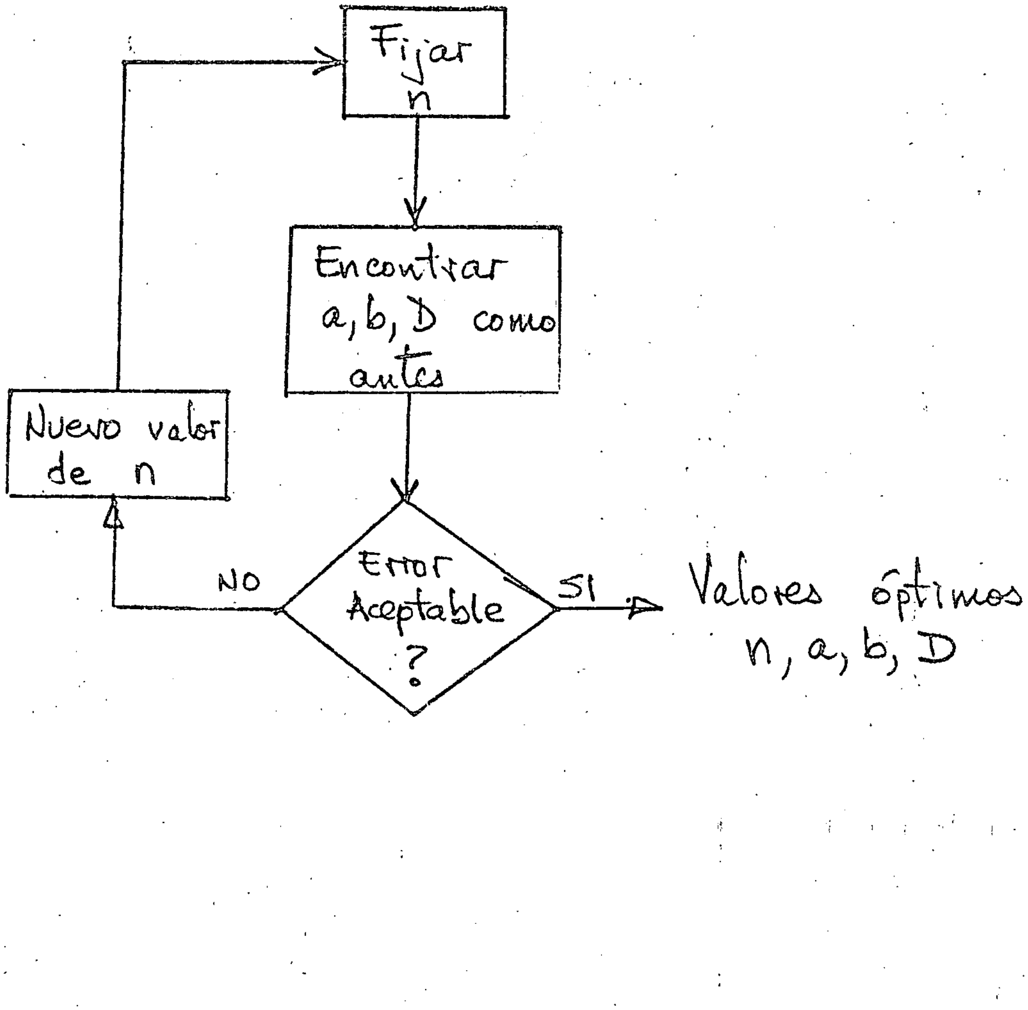
SISTEMAS CON RETRASO $e^{-\theta s}$

$$G(s) = \frac{K e^{-\theta s}}{\tau s + 1}$$

Supongamos $\theta = nT$ (buena aproximación si T es pequeño)

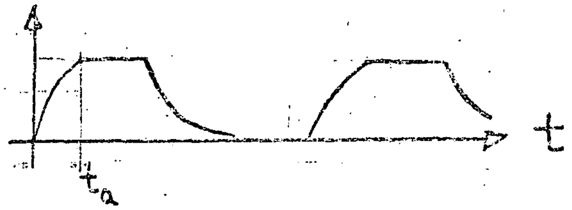
$$x_{i+1} = a x_i + b u_{i-n} + D$$

Ya no es posible usar regresión lineal. Por ello se propone el siguiente método:



Resultados de Dahlin ("On line identification
of process dynamics"
IBM Journal of Research
and Development. Vol 11. No 4
Julio 1967, pp. 406 – 425)

Pulsos de prueba :



Modelo	Errores en n	Efecto de errores en D	Errores en a, b	Divergencia*	Efectos de filtrado previo de datos
$x_{i+1} = a x_i + b u_i + D$	muy pequeños. Especialmente si t_f y τ pequeños	pequeños independ. de la longitud del exper.	muy grandes si hay ruido Indep de u	Posible	Excelente
$\bar{X}_{i+1} = a\bar{X}_i + b\bar{U}_i + D$	muy grandes si hay ruido	aumentan con la longitud del exper.	Pequeños si T pequeño	Posible	Ayuda

(D se estimó manteniendo u constante durante un tiempo largo y midiendo la salida)

* No necesariamente con el mismo conjunto de datos.

REGRESION NO LINEAL DE MINIMOS CUADRADOS:

EJEMPLO

$$G(s) = \frac{Ke^{-\theta s}}{(\tau_1 s+1)(\tau_2 s+1)}$$

$$HG(z) = \frac{z^{-n}(b_1 z^{-1} + b_2 z^{-2})}{1 - a_1 z^{-1} + a_2 z^{-2}}$$

$$\Longrightarrow \quad X_i = a x_{i-1} - a_2 x_{i-2} + b_1 u_{i-(n+1)} + b_2 u_{i-(n+2)}$$

a_1, a_2, b_1 y b_2 son funciones de τ_1, τ_2 y K.

$$J(e) = \frac{1}{N}\sum_{i=1}^{N} (x_i - \hat{x}_i)^2$$

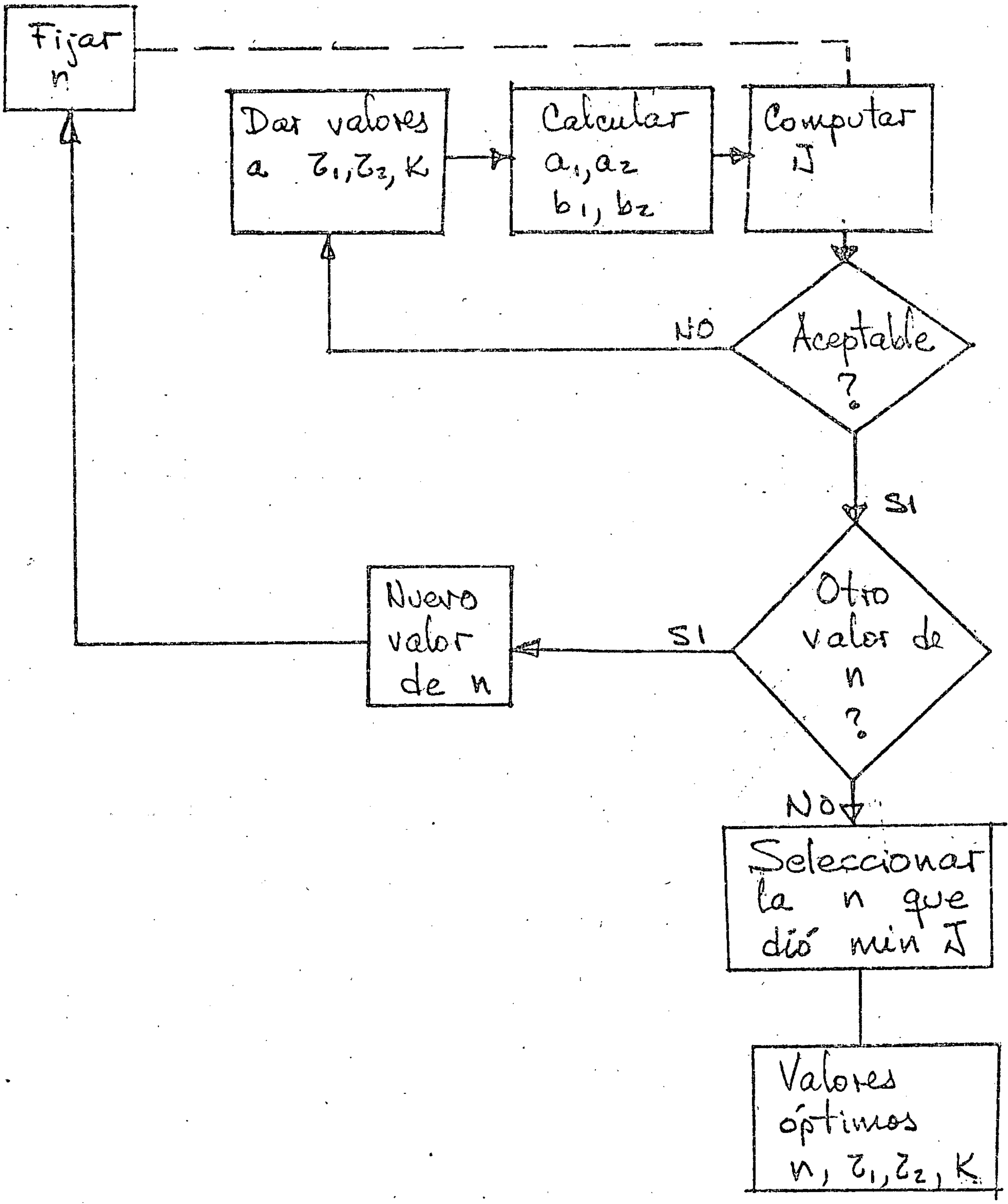
$$= \frac{1}{N}\sum_{i=1}^{N} \left(x_i - a x_{i-1} + a_2 x_{i-2} - b_1 u_{i-(n+1)} + b_2 u_{i-(n+2)}\right)^2$$

Minimizar J con respecto a $\boxed{n; \tau_1, \tau_2, K}$ ← PARAMETROS

La estrategia a seguir para minimizar J sera similar a la expuesta en la pág. 10.

Puesto que generalmente se conoce el posible rango de variación de θ, es factible fijar desde el principio ún conjunto de valores de n sobre los

$n = n_1, n_2, \ldots, n_r$



EVALUACION DE $J(e)$

En algoritmos como el de la página anterior, uno de los problemas principales desde el punto de vista de la operación en línea es el cálculo reiterado del criterio de error $J(e)$

En esta sección se discute la evaluación de J enfocada a cálculos en línea. Se busca, por tanto, obtener un mínimo de operaciones necesarias, y esto se logra identificando factores constantes en J, es decir, que no dependen de los parámetros sobre los que se está optimizando:

En J aparecen tres tipos de factores:

- Términos en X (salida) tales como:

$$\Sigma X_i^2, \quad \Sigma X_i X_{i-1}, \quad \Sigma X_i X_{i-2} \quad \text{etc.}$$

Es claro que éstos no dependen de

los parámetros y por ello se pueden calcular y almacenar independientemente.

- Términos en u (entrada) tales como Σu_{i-n-1}^2, $\Sigma u_{i-n-2} u_{i-n-1}$, etc.
  Aunque estrictamente estos términos dependen de n (que es uno de los parámetros que se está buscando), se pueden también considerar como <u>constantes</u> y evaluar a priori, si se toma en cuenta que, para <u>valores grandes de N</u>:

$$\Sigma u_{i-n-1}^2 \simeq \Sigma u_{i-n-2}^2 \simeq \Sigma u_{i-n-3}^2 \quad \cdots$$

y $\Sigma u_{i-n-1} u_{i-n-2} \simeq \Sigma u_{i-n-2} u_{i-n-3} \quad \cdots$

- Términos en xu tales como

$$\Sigma u_{i-n-1} x_i \; , \; \Sigma u_{i-n-2} x_i \; , \; \text{etc.}$$

Aquí no es posible aproximar los factores como constantes. Sin embargo si recordamos que en general se tiene un conjunto restringido de valores para n, se pueden calcular y almacenar dichos términos, para esos valores de n, en un arreglo matricial (que en general no ocupará mucha memoria). Para este ejemplo, una columna del arreglo (para un valor particular de n) sería:

$$S_{n+1} = \Sigma U_{i-n-1} x_{i-2}$$

$$S_n = \Sigma U_{i-n-1} X_{i-1} = \Sigma U_{i-n-2} X_{i-2} .$$

$$S_{n-1} = \Sigma U_{i-n-1} X_i = \cdots$$

$$S_{n-2} = \Sigma U_{i-n-2} X_i = \cdots$$

$$S_{n-3} = \Sigma U_{i-n-3} X_i$$

y, si por ejemplo hubiese 10 posibles valores para n, sería necesario almacenar en memoria un arreglo de

$$10 \times 5 = 50 \text{ elementos.}$$

VENTANA EXPONENCIAL

Como se ha visto, es necesario calcular varios factores del tipo

$$\sum_{i=1}^{N} X_i^2 \qquad \textcircled{I}$$

que no es más que N veces la media aritmética de X_i^2

Comparemos $\textcircled{I}$ con la recursión

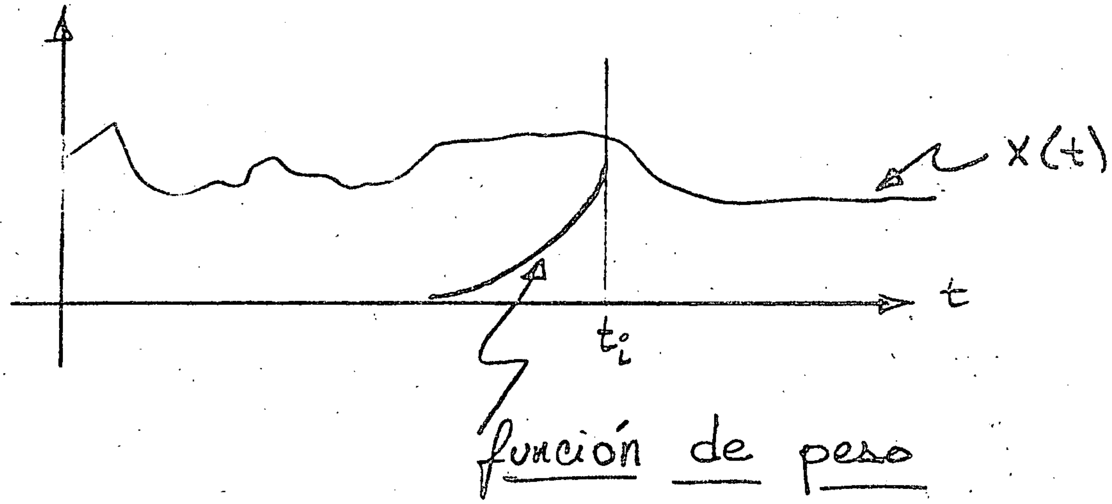
$$\begin{aligned} \overline{X}_i^2 &= \alpha\, x_i^2 + (1-\alpha)\,\overline{x}_i^2 \\ \overline{X}_1^2 &= x_1^2 \end{aligned} \qquad \textcircled{II}$$

$\textcircled{II}$ se conoce como ventana exponencial (o media exponencial) y presenta varias ventajas sobre $\textcircled{I}$:

- No hay problemas de overflow
- Se da mas peso a los datos mas recientes
- Se computa recursivamente y ofrece

continuidad en los datos, permitiendo su actualización cada vez que se obtiene un nuevo dato

Gráficamente, el efecto de II sobre la información es :



II es equivalente a un filtro

$$X(s) \longrightarrow \boxed{\frac{a}{s+a}} \longrightarrow \bar{X}(s)$$

$$\alpha = 1 - e^{-aT}$$

CONCLUSIONES

Se ha visto, mediante el desarrollo exhaustivo de dos ejemplos, una forma general de plantear modelos de regresión lineal para resolver el problema de estimar los parámetros de una estructura matemática propuesta La estimación se basa en la observación, durante un tiempo suficientemente largo, de la entrada y la salida del sistema en estudio. los algoritmos se han obtenido a partir de modelos en tiempo discreto, obtenidos al muestrear un sistema continuo (transformada z)

Cabe señalar que es posible plantear, desde el principio, una estructura en tiempo discreto, cuya forma general sería

$$x_i = \sum_{j=1}^{p} a_j x_{i-j} + \sum_{j=1}^{q} b_j u_{i-j} + D$$

$$HG(z) = \frac{b_1 z^{-1} + \ldots + b_q z^{-q}}{1 + a_1 z^{-1} + \ldots + a_p z^{-p}}$$

Sobre esta estructura se estimarían directamente los parámetros a_j, b_j y D

REFERENCIAS

- Digital computer process control.
Cecil L. Smith
Intext Educational Publishers 1972.

- Digital Signal Processing
Alan Oppenheim, Ronald Schafer
Prentice Hall

- Uncertain Dynamic Systems
Fred Schweppe
Prentice Hall 1973.

- System Identification of Linear Time Invariant Systems.
Fred Schweppe
Reporte del Laboratorio de Ingeniería de Sistemas Eléctricos de Potencia.
Massachusetts Institute of Technology 1976

centro de educación continua
división de estudios superiores
facultad de ingeniería, unam

INGENIERIA DE CONTROL DIGITAL DE PROCESOS Y SUS APLICACIONES

CONTROL OPTIMO

DR. VICTOR GEREZ GRESIER

NOVIEMBRE 1978.

CONTROL SUPERVISORIO

El más usado en la actualidad Controles analógicos realizan las funciones de control al primer nivel. La computadora calcula las ganancias de las acciones controladoras del control analógico -

Prerequisitos: Modelo apropiado de la planta

Optimización de las condiciones de operación.

La disponibilidad del modelo limita su aplicación. Su costo puede ser elevado.

Modelos:

Matemáticos

Físicos:

Ejemplo:

Empíricos

Sist Eléctrico de potencia

¿Dinámicos o de estado estable?

De Procedimiento:

Mas consistencia que el operador.

Permiten devolver a la planta a estado normal si hay una emergencia, la detectan.

Pasar de un estado (calidad de producto) a otro repitiendo procedimientos desarrollados por el operador; el control optimo con frecuencia no puede aplicarse

Económicos.

Función objetivo

Restricciones

Establecer funciones de costo (objetivo) puede ser complicado: Se desconocen los costos exactos.

MODELOS PARA CONTROL SUPERVISORIO:

Los modelos de diseño son en general no adecuados.

Se recomiendan modelos de.. estado estable cuando sea posible, aveces ecuaciones de continuidad de flujos, y balances térmicos y de energía son suficientes.

Modelos matemáticos:

Ventajas: Da mejor idea sobre el proceso y permite extrapolar fuera del rango normal de operación.

Desventajas: Relaciones complicadas que pueden implicar tiempos largos de computo

Modelos empíricos:

La disponibilidad de datos puede dificultar su establecimiento.

Tecnicas de Optimización:

6.2.4 Método de búsqueda

En la sección 6.2.1 se estudió el método de optimización por diferenciación y en la sección 6.2.3 el de los multiplicadores de Lagrange.

*Estos métodos requieren para poder ser usados que la función por optimizar f(x) sea continua y diferenciable. En muchos problemas prácticos es muy difícil determinar si se cumple esta condición.

*La diferenciación directa o los multiplicadores de Lagrange requieren de funciones continuas y diferenciables. Estas condiciones son difíciles de checar.

*Los métodos de busca directa que se exponen en esta sección para funciones de una sola variable independiente y en la sección 6.3.3 para funciones de varias variables no requieren para aplicarse que la función sea diferenciable ni continua. La función tiene que ser solamente ***computable,*** es decir, debe poderse calcular el valor de la variable dependiente, si se conoce el valor de las variables independientes.

*Los métodos de búsqueda directa requieren que la función sea sólo computable.

*Todos los métodos de búsqueda directa que se exponen en esta sección para funciones de una variable independiente y en la sección 6.3.3 para funciones de varias variables son aplicables a problemas sin restricciones.

*Búsqueda directa para problemas sin restricciones.

6.1. INTRODUCCION

6.1.1 Función objetivo y restricciones

El objetivo de este capítulo es describir las técnicas de optimización que se emplean con mayor frecuencia en el análisis de sistemas. Se ha señalado en el capítulo 1 que durante la síntesis de sistemas es necesario maximizar o minimizar una cantidad, que es la medida de efectividad de una determinada operación.

No se pretende cubrir en forma exhaustiva este tópico que es sumamente amplio. Solamente se darán a conocer las técnicas de optimización más importantes. *Se hará hincapié fundamentalmente en los aspectos de aplicación. Al lector interesado en conocer las bases teóricas de estos procedimientos se le refiere a la bibliografía que aparece al final del capítulo.

*Aspectos de aplicación.

*La formulación matemática general de estos problemas es la siguiente:

*Formulación matemática.

Encuéntrese el valor de las variables $(x_1, x_2 \ldots x_n)$ que maximicen (o minimicen) a la función M llamada ***función objetivo.***

*Función objetivo.

$$M = M(x_1, x_2, \ldots, x_n) \qquad (6.1.1)$$

*Sujeta a las siguientes restricciones.

*Restricciones.

$$C_i \ (x_1, x_2, \ldots, x_n) = 0 \text{ para } i = 1, \ldots p$$

$$C_i \ (x_1, x_2, \ldots, x_n) \leq 0 \text{ para } i = p + 1, \ldots r \qquad (6.1.2)$$

$$C_i \ (x_2, x_2, \ldots, x_n) \geq 0 \text{ para } i = r + 1, \ldots m \qquad (6.1.2)$$

*Por razones que se señalan en la sección sobre programación lineal es deseable que todas las restricciones sean igualdades, es decir, del tipo

*Restricciones de igualdad.

$$C_i \ (x_1, x_2, \ldots, x_n) = 0 \text{ para } i = 1,2, \ldots m \qquad (6.1.3)$$

En las siguientes secciones de este capítulo se representan diversos ejemplos que sirven para aclarar al lector la naturaleza de los problemas de optimización.

*Para la solución de este tipo de problemas existen fundamentalmente dos estrategias. En la primera se emplea un cierto procedimiento de gradientes (hillclimbing) similar al que se estudia en la sección 6.4 al tratar el problema del análisis marginal. La segunda estrategia consiste en enumerar en forma explícita diversas combinaciones posibles de variables, y seleccionar entre ellas la mejor. Este camino es el seguido por la programación dinámica, tema de la sección 6.6 de este capítulo. En ambos procedimientos

*Dos estrategias de optimización. por gradiente y por enumeración.

se realiza una búsqueda de acuerdo con determinadas reglas que permiten detectar el valor óptimo, cuando éste se ha encontrado.

*La programación lineal es la más empleada.

*Entre las técnicas de optimización, la programación lineal es la más empleada, ya que al no ser una técnica de enumeración de posibles soluciones y posterior búsqueda entre ellas de la óptima, no requiere de la gran capacidad de memoria que se necesita para los problemas de programación dinámica. Además resulta un método computacionalmente muy eficiente (rápido).

*La función objetivo y las restricciones deben cumplir determinadas condiciones.

*Como se verá en este capítulo al tratar el problema de programación lineal y el de programación dinámica, cada una de las técnicas de optimización impone tanto a la función objetivo como a las restricciones, determinadas condiciones. Entre más estrictas son estas condiciones, tanto más eficiente es la técnica de optimización correspondiente. La programación lineal al imponer condiciones sumamente estrictas, es una de las técnicas más rápidas y poderosas de optimización.

Como se verá en los ejemplos de las siguientes secciones la naturaleza del problema de optimización fija el tipo de técnica que debe emplearse para su solución. Si un problema no cumple con las condiciones que impone alguna de las técnicas de optimización, es posible, frecuentemente, reformularlo para que cumpla con las restricciones de determinada técnica de optimización.

Antes de proceder con el primer método de optimización, el del cálculo diferencial se introducen algunos conceptos preliminares adicionales.

6.1.2 Solución factible

*Espacio de N dimensiones.

*Coordenadas de un punto = variables ($x_1, x_2, \ldots, x_n$)

*Probablemente el lector no esté familiarizado con el concepto de punto en un espacio de N dimensiones, donde N es un número que puede ser mayor de tres. En este capítulo al hablar de las coordenadas de un punto, éstas no necesariamente se restringirán a tres. Es decir, se hará una extensión del concepto geométrico de tres coordenadas de un punto del espacio, a N coordenadas. *Se emplearán en forma indiferente los términos de coordenadas de un punto o variables (x_1, x_2, ... x_n). Se designará con R la región del espacio de N dimensiones, cuyos puntos satisfacen todas las restricciones (6.1.2). Para poder ilustrar este concepto, consideremos las siguientes condiciones:

$$x_1 + x_2 \leq 4 \qquad (6.1.4)$$

$$2x_1 + x_2 \leq 6 \qquad (6.1.5)$$

$$x_1 \geq 0 \qquad (6.1.6)$$

$$x_2 \geq 0 \qquad (6.1.7)$$

El lector no debe tener problema en encontrar que los puntos que satisfacen la restricción 6.1.4 son los situados en el área anchurada de la Fig. 6.1.1, es decir, el área situada a la izquierda de la recta AB.

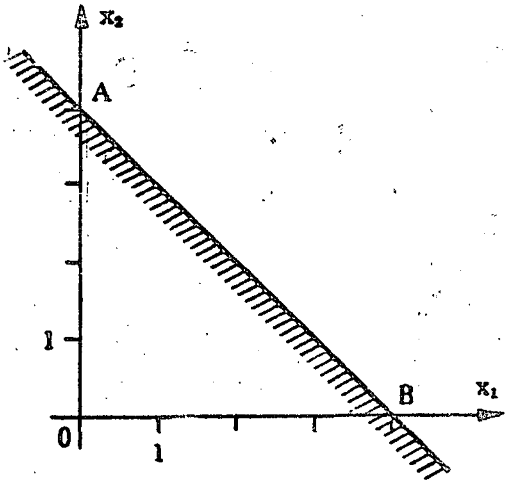


Fig. 6.1.1. Zona donde se cumple la restricción $x_1 + x_2 \leq 4$.

Los puntos que satisfacen la restricción (6.1.5) aparecen en la Fig. 6.1.2, y están situados a la izquierda de la recta CD.

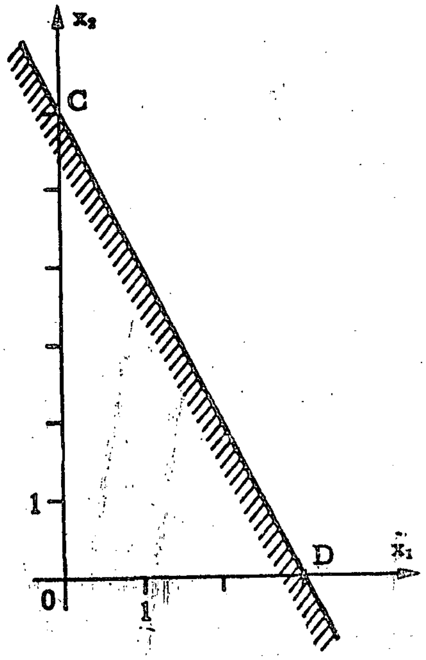


Fig. 6.1.2. Zona donde se cumple la restricción $2x_1 + x_2 \leq 6$.

Finalmente los puntos del plano donde se cumplen las restricciones $x_1 > 0$ y $x_2 > 0$ están situadas arriba del eje de las abscisas y a la derecha del de las ordenadas, tal como muestra la Fig. 6.1.3.

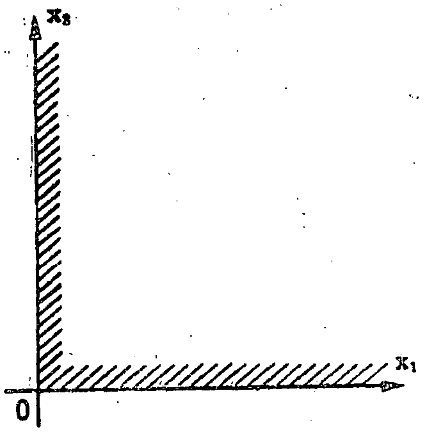


Fig. 6.1.3 Región donde se cumplen las rectricciones $x_1 > 0$ y $x_2 > 0$.

Para determinar la zona donde se cumplen las 4 restricciones (6.1.4) a (6.1.7) es necesario encontrar la región del plano, donde se satisfacen simultáneamente las 4 restricciones. Para visualizar esta zona se sobreponen las zonas mostradas en las Figs. 6.1.1 a 6.1.3 tal como aparece en la Fig. 6.1.4.

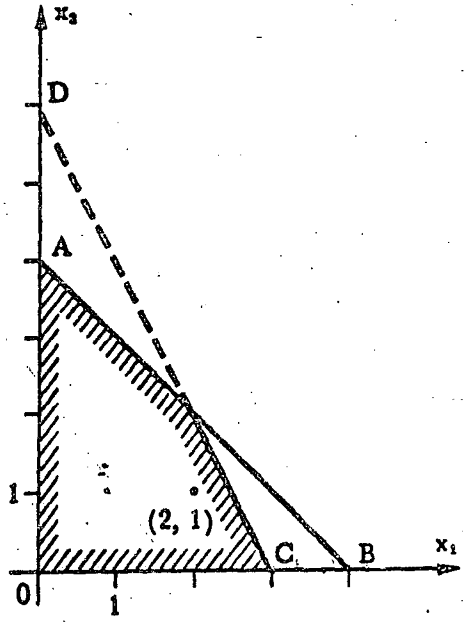


Fig. 6.1.4 Región donde se cumplen las restricciones $x_1 + x_2 \leq 4$, $2x_1 + x_2, \leq 6$.

Para las restricciones (6.1.4) a (6.1.7) la Fig. 6.1.4 muestra la región R. Todo punto de esta región, por ejemplo el (2,1) satisface las condiciones señaladas. En efecto: Sustituyendo $x_1 = 2$ y $x_2 = 1$ en las fórmulas (6.1.4) a (6.1.7) se obtiene:

$$\begin{aligned} 2 + 1 &\leq 4 \\ 2.2 + 1 &\leq 6 \\ 2 &\geq 0 \\ 1 &> 0 \end{aligned}$$

Lo que muestra que el punto (2,1) en efecto pertenece a la región R, cuyos puntos satisfacen todas las restricciones del problema de optimización. *Recibe el nombre de *solución factible* de un problema de optimización, cualquier punto o conjunto de

---

*Una solución factible es aquella que satisface todas las restricciones.

En las secciones 6.5 y 6.6 se exponen diversos métodos de optimización que requieren en general del uso de la computadora digital para su implementación y son aplicables a problemas con restricciones.

Varios de los principales métodos de búsqueda directa aparecen en la tabla 6.2.2.

Tabla 6.2.2 Principales métodos de búsqueda directa.

A. Métodos de búsqueda unidimensionales (una sola variable independiente)

- a). Métodos simultáneos
  1. Búsqueda exhaustiva
  2. Búsqueda aleatoria
- b). Métodos secuenciales
  1. Método de la trisección
  2. Método de Fibonacci

Funciones computables (a, b) — Funciones unimodales (b)

B. Métodos de búsqueda multidimensional (varias variables dependientes)

- a). Métodos simultáneos
  1. Búsqueda exhaustiva
  2. Búsqueda aleatoria
- b). Métodos secuenciales
  1. Búsqueda de rejilla
  2. Búsqueda univariada
  3. Métodos de gradiente
  4. Métodos de Fletcher-Powell
  5. Búsqueda de patrón.

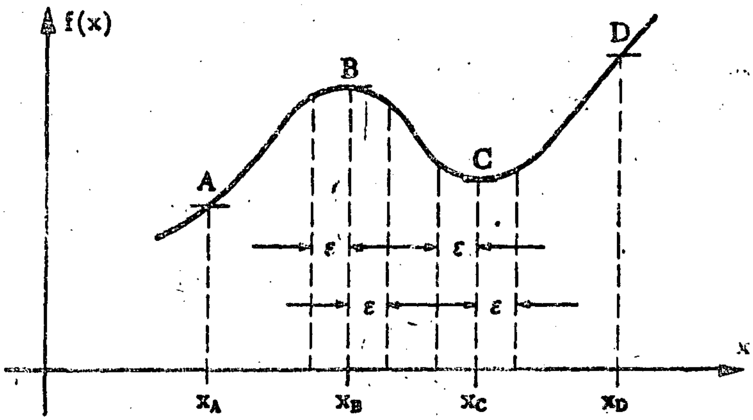
Funciones comparables (a, b) — Funciones unimodales (b)

*Los métodos de búsqueda determinan el máximo o mínimo global de la función en un determinado intervalo, □mientras que los métodos de optimización por diferenciación expuestos en la sección 6.2.1 permiten encontrar máximos o mínimos locales.

*La búsqueda directa encuentra máximos mínimos globales.

□Los métodos de diferenciación encuentran máximos o mínimos locales.

Se dice que la función f(x) tiene un máximo (o mínimo) global en el intervalo $a \leq x \leq b$ en el punto $x = x_o$, $a \leq x_o \leq b$ si f(x) es mayor (o menor) en $x = x_o$ que en cualquier punto del intervalo [a, b].



Punto A: mínimo global $f(x_A) \leq f(x)$
$x_A \leq x \leq x_D$

Punto B: máximo local $f(x_B) \geq f(x_B \pm \varepsilon)$

Punto C: mínimo local $f(x_C) \leq f(x_C \pm \varepsilon)$

Punto D: máximo global $f(x_D) \geq f(x)$
$x_A \leq x \leq x_D$

Fig. 6.2.2 Función con máximos y mínimos locales y globales.

Por otra parte, la función $f(x)$ tiene un máximo (o mínimo) local en $x = x_1$, $a \leq x_1 \leq b$ si solamente se cumple que $f(x)$ es mayor (o menor) en $x = x_1$, que en cualquier otro punto de la vecindad de x_1. Donde esta vecindad puede estar tan próxima del punto x_1 como se quiera. La figura 6.22 ilustra estos conceptos.

*Antes de describir algunos métodos de búsqueda directa es necesario aclarar la diferencia que existe entre métodos de búsqueda simultánea y métodos de búsqueda secuencial.

*Búsqueda simultánea → selección *a priori* de todos los valores de x.

En los primeros, al iniciar la búsqueda se determinan todos los puntos x donde se va a evaluar la función.

*En los métodos secuenciales, los puntos x donde se va a efectuar la determinación de $f(x)$ no pueden determinarse *a priori* y dependen de los valores de $f(x)$ que se hayan observado previamente.

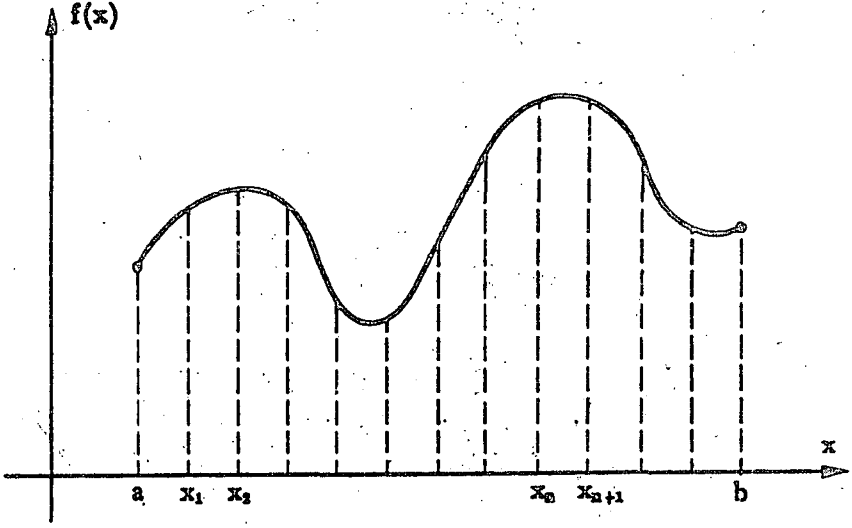
*Búsqueda secuencial → el siguiente valor x depende de valores previos de $f(x)$.

En esta sección se estudian algunos métodos de búsqueda *unidimensional* que se emplean directamente en diferentes problemas de análisis de sistemas y en ciertas etapas en la búsqueda multidimensional.

*En el *método de búsqueda exhaustiva* se subdivide el intervalo [a, b], se evalúa la función $f(x)$ en los puntos centrales de cada intervalo, o en sus extremos, y se busca el máximo o mínimo entre los valores de $f(x)$ encontrados.

*Búsqueda exhaustiva.

Este método requiere de un gran número de evaluaciones, y la precisión del resultado depende del tamaño del intervalo que se haya seleccionado, entre más fino sea éste es mayor la precisión pero también mayor el tiempo de cálculo. La figura 6.2.3 ilustra cómo se procede en este método.



(Se evalúa $f(a)$, $f(x_1)$ $f(x_2)$ ... $f(x_n)$ ... $f(x_b)$ y se selecciona el mayor (o menor).

Fig. 6.2.3 Búsqueda unidimensional y exhaustiva.

En el método de búsqueda aleatoria se genera un número aleatorio** en el intervalo [a, b] y se evalúa la función para ese número aleatorio. El procedimiento se continúa hasta un número predeterminado de veces. En cada etapa de cálculo se retiene el valor más grande que se haya encontrado. La figura 6.2.4 muestra el diagrama de bloque para este método de búsqueda directa y simultánea para un problema de optimización con N evaluaciones de f(x).

** Ver sección 5·2 y programa A8.

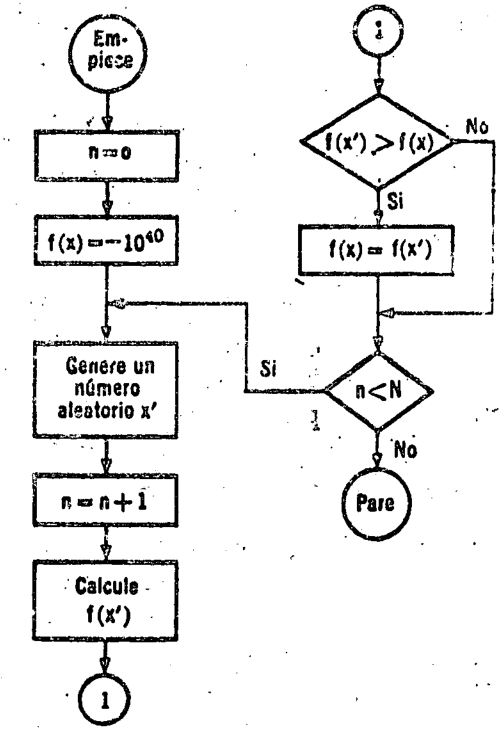


Fig. 6.2.4 Diagrama de bloque para el método de búsqueda aleatoria.

En el programa A.14 del apéndice A se ha incorporado el programa A.8 de generación de números aleatorios para buscar un máximo global de una función por el método de búsqueda aleatoria.

Este programa se ha empleado para encontrar el máximo de la función:

$$y = -0.4\,x^2 + 4x$$

Los resultados del método de búsqueda aleatorio para diferentes valores de N, aparecen en la tabla 6.2.3. *El lector puede encontrar fácilmente por diferenciación directa que el máximo de esta función es:

*Por diferenciación directa:

$$\text{máx: } f(x)\,\Big|_{x\,=\,5} = 10$$

Tabla 6.2.3 Evaluación del máximo global de $y = -0.4 x^2 + .4x$ en el intervalo (0, .10), por el método de búsqueda aleatoria.

Número de números aleatorios generados	x	f(x)
25	4.9051	9.9964
100	4.9051	9.9964
250	4.9091	9.9966
500	5.0088	9.9999

*Los métodos de búsqueda simultánea, a pesar de su ineficiencia encuentran aplicación en aquellas situaciones donde no existe suficiente tiempo para realizar secuencialmente los cálculos. El tiempo disponible reducido tiene que emplearse para efectuar los cálculos en forma simultánea.

*Los métodos de búsqueda simultánea son ineficientes.

A continuación se estudian dos métodos de búsqueda simultánea, el de trisección y el de Fibonacci.

*Todos los métodos de búsqueda secuencial requieren que la función sea *unimodal* dentro del intervalo de búsqueda, es decir, debe tener un solo máximo o mínimo en el intervalo de búsqueda [a, b]. Si se trata de una función unimodal con un máximo en [a, b], el valor de la función debe incrementarse a partir de $x = a$, hasta llegar a un máximo en $x = x_o$ y decrecer después. Desde luego el máximo puede encontrarse tanto en $x = a$, como en $x = b$, es decir, en los extremos del intervalo. La figura 6.2.5 muestra 3 funciones unimodales.

*Los métodos de búsqueda secuenciales requieren que la función sea unimodal.

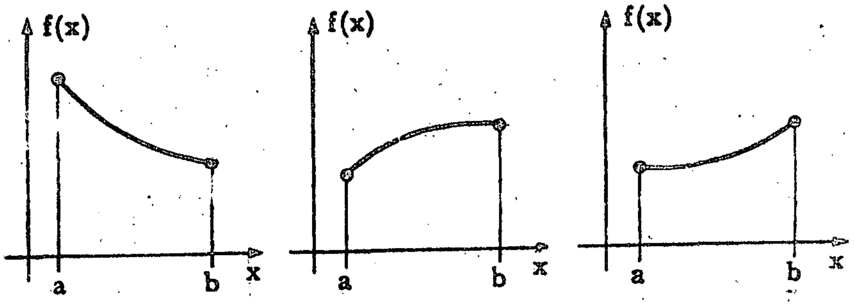
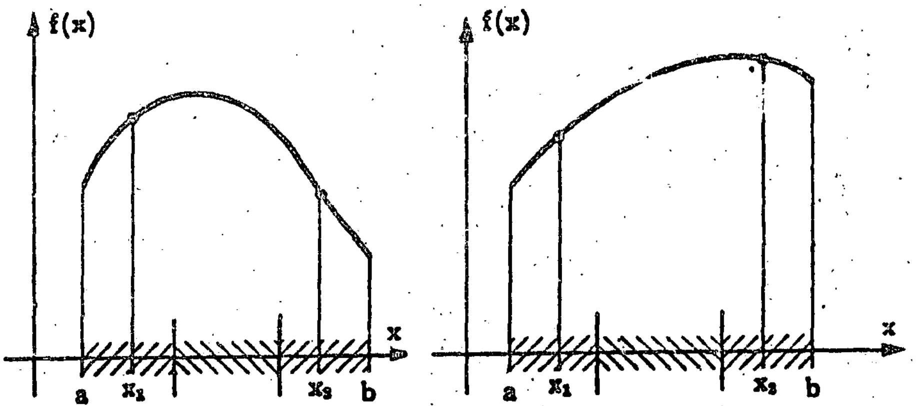


Fig. 6.2.5 Tres funciones unimodales en el intervalo [a, b]

*El primer método de búsqueda secuencial que se estudia en esta sección es el de trisección. En este método se subdivide el intervalo de búsqueda [a, b], en tres subintervalos iguales y se

*En el método de la trisección se subdivide el intervalo en 3 partes iguales.

evalúa la función al centro del 1er. y 3er. intervalos (puntos $_1$ y x_2), tal como muestra la figura 6.2.6. Los valores calculados de la función se comparan.



$f(x_1) > f(x_2)$
máximo entre
(a, x_2)
$b \leftarrow x_2$

$f(x_1) < f(x_2)$
máximo entre
(x_1, b)
$a \leftarrow x_1$

Fig. 6.2.6 **Primer paso en la búsqueda del máximo por el procedimiento de trisección.**

De esta comparación se concluye que el máximo se encuentra o en (a, x_2) o (x_1, b), tal como ilustra la figura 6.2.6. El procedimiento continúa empleando (a, x_2) o (x_1, b) como nuevos intervalos de búsqueda, hasta llegar a un intervalo de longitud suficientemente pequeño para la precisión que se desea, la figura 6.2.7 muestra el diagrama de bloque para este procedimiento de búsqueda. *Nótese que en cada etapa de la búsqueda se reduce la longitud del intervalo donde puede encontrarse el máximo. *Además, es necesario calcular en cada etapa el valor de la función en dos puntos x_1 y x_2.

***En cada etapa se reduce la longitud del intervalo.**

***Se calcula en cada etapa el valor de la función en dos puntos.**

En funciones complicadas estos cálculos toman más tiempo que todas las operaciones restantes del procedimiento de búsqueda. Un procedimiento de búsqueda que necesita una sola evaluación funcional por etapa ahorraría tiempo de computación. *El método de búsqueda por números de Fibonacci tiene esta característica.

***En la búsqueda con números de Fibonacci se hace una evaluación funcional por etapa.**

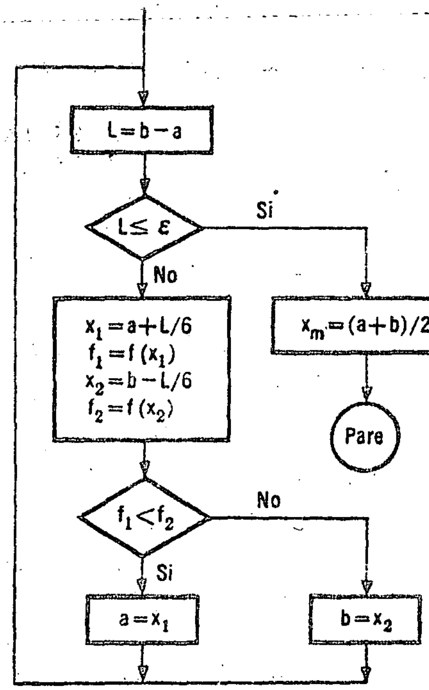


Fig. 6.2.7 Diagrama de flujo para la búsqueda de un máximo por el procedimiento de trisección.

Los números de Fibonacci fueron descubiertos por Leonardo de Pisa (1180-1250), llamado Fibonacci o hijo de "Bonaccio". Fibonacci, el mejor matemático de la época medieval en Europa, popularizó el empleo de los caracteres numéricos arábigos en el mundo occidental y en su obra principal *Liber abaci* plantea el siguiente problema:

> "Cuántas parejas de conejos se producirán en un año, empezando con una sola pareja, si cada mes cada pareja tiene una nueva pareja, que a su vez tiene una pareja a partir del segundo mes".

Este famoso problema da lugar a la secuencia de *números de Fibonacci F_n que aparecen en la tabla 6.2.4.

*Números de Fibonacci F_n.

Tabla 6.2.4 Números de Fibonacci.

n	0	1	2	3	4	5	6	7	8	9	10	11
F_n	1	1	2	3	5	8	13	21	34	55	89	144

Estos números se forman de la siguiente manera:

$$F_0 = 1$$
$$F_1 = 1$$
$$F_n = F_{n-1} + F_{n-2}$$

Es decir, cada número de la serie es igual a la suma de los dos números precedentes.

A continuación se verá cómo se emplean los números de Fibonacci para buscar el máximo o mínimo global en el intervalo [a, b] de una función *unimodal*.

***Empleo de los números de Fibonacci en un proceso de búsqueda secuencial.**

Sea L_1 la longitud del intervalo [a, b]:

$$L_1 = b - a$$

*Al iniciarse el procedimiento de búsqueda se calcula la función unimodal f(x) en los dos puntos siguientes:

° 1er. cálculo funcional

$$x_1 = a + \Delta_2$$

$$x_2 = b - \Delta_2 \tag{6.2.27}$$

donde Δ_2 es igual a:

$$\Delta_2 = L_1 \frac{F_{n-2}}{F_n} \tag{6.2.28}$$

Obsérvese que el cociente de los números de Fibonacci en la relación es:

$$\frac{F_{n-2}}{F_n} < \frac{1}{2} \tag{6.2.29}$$

por lo que el intervalo Δ_2 definido por la relación (6.2.28) cumple con:

$$\Delta_2 \le \frac{b - a}{2} \tag{6.2.30}$$

*Al igual que en el procedimiento de la trisección se empieza comparando los siguientes valores de la función f(x).

°Empiece comparando $f(x_1)$, $f(x_2)$

*y de acuerdo con el resultado de la comparación y por cumplirse $\Delta_2 \le \frac{b-a}{2}$ se descarta cualquiera de los dos intervalos siguientes:

*Se descarta por ser

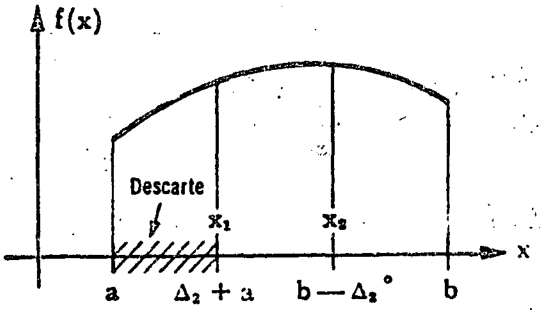
$$\Delta_2 \le \frac{b-a}{2}$$

$(a, a + \Delta_2)$

ó

$(b - \Delta_2, b)$

La figura 6.2.8 aclara este primer paso para un posible caso.



$f(x_1) < f(x_2)$

$a \leftarrow a + \Delta_2$

Fig. 6.2.8 Primer paso en una búsqueda secuencial.

*Observe que el intervalo en el que puede encontrarse el máximo (o mínimo) después de la primer etapa (y dos evaluaciones funcionales) tiene siempre por longitud

°Longitud del intervalo después del 1er. paso:

$$L_2 = b - a - \Delta_2 = L_1 - \Delta_2$$

Tal como lo ilustra la figura 6.2.9.

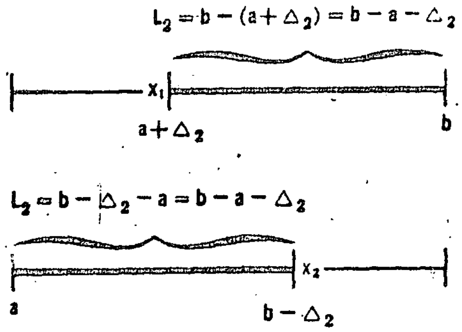


Fig. 6.2.9 Intervalos residuales L_2 después del 1er. paso.

Teniendo presente el valor de Δ_2:

$$\Delta_2 = L_1 \frac{F_{n-2}}{F_n} \tag{6.2.28}$$

y sustituyendo en la nueva longitud L_2 del intervalo

en

$$L_2 = L_1 - \Delta_2$$

se obtiene

$$L_2 = L_1 \frac{F_n - F_{n-2}}{F_n}$$

pero de la regla de generación de los números de Fibonacci

$$F_n = F_{n-1} + F_{n-2}$$

$$F_n - F_{n-2} = F_{n-1}$$

$$L_2 = L_1 \frac{F_{n-1}}{F_n} \tag{6.2.31}$$

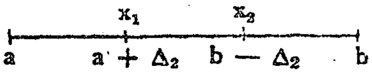
*A continuación se define de manera similar una distancia Δ_3 para dividir el intervalo que ha quedado después del 1er paso

*Para dividir intervalo residual

$$\Delta_3 = L_2 \frac{F_{n-3}}{F_{n-1}} \tag{6.2.32}$$

*Véase ahora qué relación guarda la distancia Δ_3 con la distancia entre los puntos x_1 y x_2.

*Relación entre Δ_3 y x_1 y x_2



La distancia entre estos dos puntos es:

$$\begin{aligned} x_2 - x_1 &= b - \Delta_2 - (a\text{-}\Delta_2) \\ &= b - a - 2\,\Delta_2 \\ &= L_1 - 2\,\Delta_2 \end{aligned}$$

como $\Delta_2 = L_1 \dfrac{F_{n-2}}{F_n}$

$$x_2 - x_1 = L_1 \left(1 - 2\,\frac{F_{n-2}}{F_n} \right)$$

$$= L_1 \frac{F_n - 2F_{n-2}}{F_n}$$

$$= L_1 \frac{F_n - F_{n-2} - F_{n-2}}{F_n}$$

y por la forma de generación de los números de Fibonacci

$$= L_1 \frac{F_{n-1} - F_{n-3}}{F_n}$$

$$x_2 - x_1 = L_1 \frac{F_{n-3}}{F_n} \qquad (6.2.33)$$

pero de la relación (6.2.31)

$$L_2 = L_1 \frac{F_{n-1}}{F_n} \qquad (6.2.31)$$

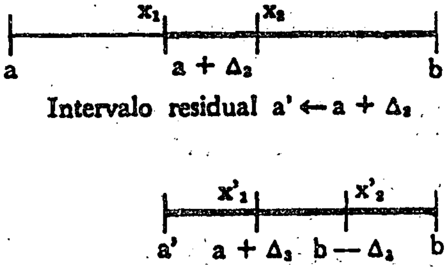
o

$$L_1 = L_2 \frac{F_n}{F_{n-1}}$$

sustituyendo en (6.2.33) se obtiene

$$x_2 - x_1 = L_2 \frac{F_{n-3}}{F_{n-1}} = \Delta_3 \qquad (6.2.34)$$

A continuación se señala la importancia que tiene el resultado anterior en el método de búsqueda secuencial propuesto. Supóngase que en la 1er. etapa se descartó el intervalo $[a, a + \Delta_2)$, tal como ilustra la fig. 6.2.10. En esta etapa además se calculó la función en x_1 y x_2. Al pasar al 2do. paso de cálculo se tiene el intervalo de longitud reducida que aparece en la parte inferior de la figura donde se ha hecho la equivalencia $a' = a + \Delta_2$. En el 2do. paso se deben conocer los valores de la función en los puntos x'_1 y x'_2. Pero por tenerse que $x_2 - x_1 = \Delta_3$, los puntos x_2 y x'_1 coinciden, y para hacer la comparación funcional en la segunda etapa hay que evaluar solamente en este caso $f(x'_2)$.



$\Delta_3 = x_2 - x_1$

$x'_1 = x_2$

Debe conocerse:

$f(x'_1)$ y $f(x'_2)$

pero: $f(x'_1) = f(x_2)$

Fig. 6.2.10 Evaluaciones funcionales en la 2da. etapa.

Si se hubiese descartado en el 1er. paso $(b - \Delta_2, b]$, se hubiese tenido que evaluar en el 2do. paso solamente $f(x'_1)$. *En resumen, a diferencia del método de la trisección donde en cada paso hay que realizar dos evaluaciones funcionales, en este método sólo hay que hacer a partir del segundo paso, una sola evaluación funcional.

*Método de la trisección: 2 evaluaciones funcionales por paso.

Método de Fibonacci: 1 evaluación funcional por paso.

*Para el tercer paso puede demostrarse como se hizo anteriormente, que la longitud del intervalo habrá quedado reducido a:

*Longitud en el 3er. paso:

$$L_3 = L_1 \left(\frac{F_{n-3}}{F_n} \right) \qquad (6.2.35)$$

Continuando con este procedimiento, puede llegarse después de n pasos a la relación

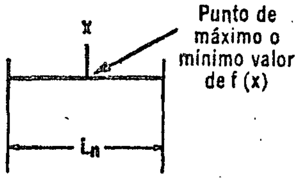
$$L_n = L_1 \left(\frac{F_o}{F_o} \right)$$

$$\frac{L_n}{L_1} = \left(\frac{F_o}{F_n} \right) \qquad (6.2.36)$$

*Esta última relación permite determinar cuántos pasos de evaluación deben de ejecutarse para que la relación entre el último intervalo y el primero, que es una medida de la precisión deseada, tenga un cierto valor. *El cociente $\frac{L_n}{L_1}$ es una medida de la precisión del procedimiento de búsqueda, ya que después de n pasos, el valor del máximo se encuentra en un entorno de longitud L_n alrededor de un punto x_o. Esta relación sirve para calcular el número de etapas que se necesitan, para una determinada precisión. Por ejemplo, si se quiere maximizar la función $y = -0.4\ x_2 + 4x$ en el intervalo [0, 10], la longitud L_1 es de 10 y si se desea que el resultado esté en un entorno de longitud $L_n = 0.1$, debe tenerse:

*$\frac{L_n}{L_1}$ es una medida de precisión del procedimiento de búsqueda.

*$\frac{L}{L_1}$ es una medida de la precisión del procedimiento de búsqueda.



$$\frac{L_n}{L_1} = \frac{0.1}{10} = \frac{F_o}{F_n}$$

$$F_n = 100$$

de la *tabla 6.2.3 se encuentra que $F_n = 100$ corresponde a $10 < n < 11$, es decir, debe tomarse $n = 11$.

*n	9	10	11	12
F_n	55	89	144	233

La fig. 6.2.11 muestra el diagrama de bloque para obtener el máximo de una función con el método de Fibonacci. El programa A.15 del apéndice A permite encontrar el máximo de una función por este procedimiento. Se ha empleado este programa para obtener el máximo de la función:

$$y = -0.4x^2 + 4x$$

en el intervalo [0, 10]

con una precisión de $\frac{.1}{10} = \frac{1}{100}$

es decir, con n 11

Los resultados de que se obtienen aparecen en la tabla 6.2.5.

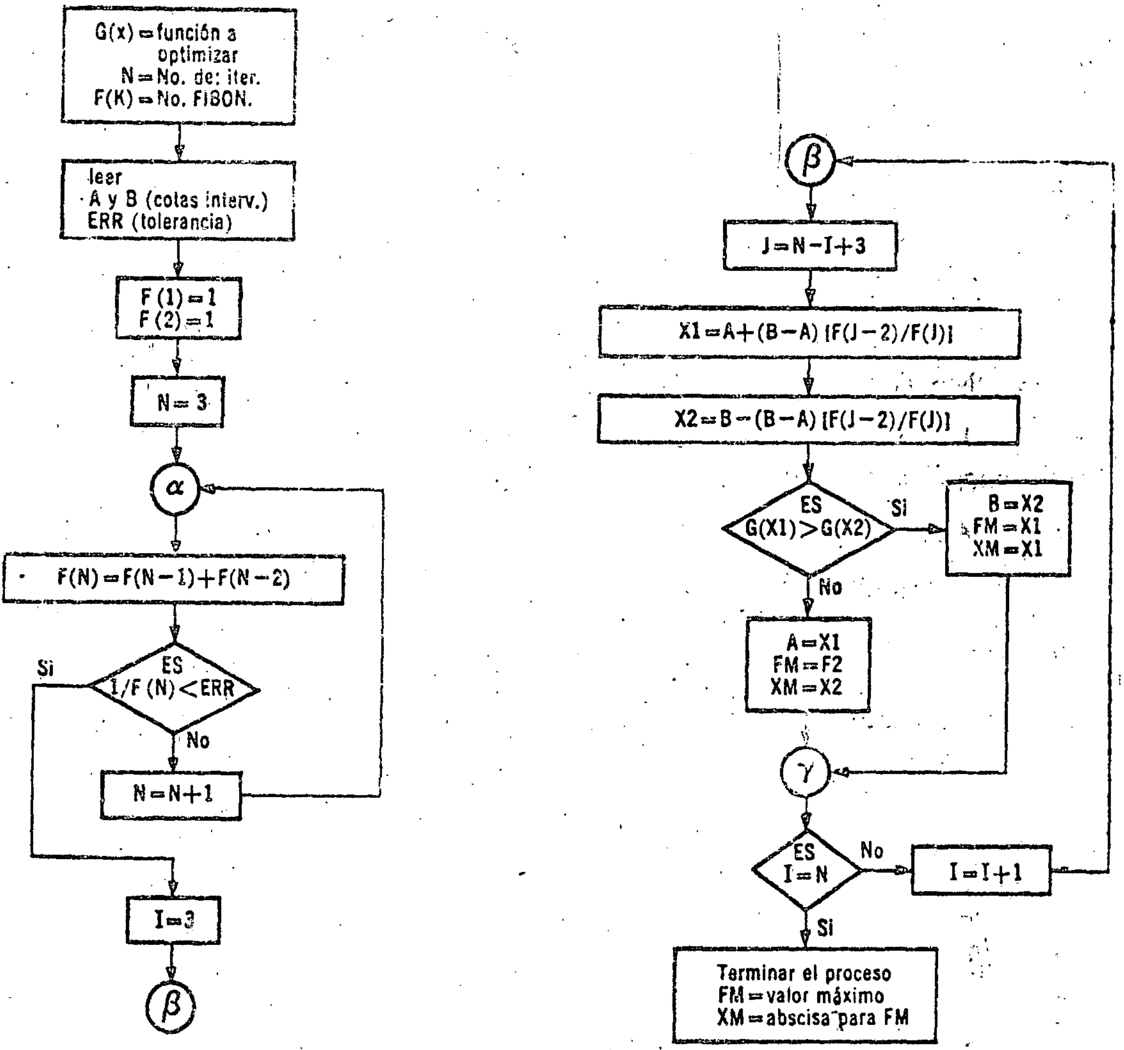


Fig. 6.2.11 Diagrama de flujo para la búsqueda de un máximo global de una función números de Fibonacci.

Tabla 6.2.5 Resultados del programa de maximización A.15 con la función $y = -0.4x^2 + 4x$.

```
              F_N               DELTA              F(x1)              F(x2)
 1      1.44000E+02      3.81944E+00      9.44252E+00      9.44252E+00
 2      8.90000E+01      2.36111E+00      9.44252E+00      7.21451E+00
 3      5.50000E+01      1.45833E+00      9.96914E+00      9.44252E+00
 4      3.40000E+01      9.02778E+01      9.96914E+00      9.96914E+00
 5      2.10000E+01      5.55556E+01      9.96914E+00      9.84375E+00
 6      1.30000E+01      3.47222E—01      9.99807E+00      9.96914E+00
 7      8.00000E+00      2.08333E—01      9.99807E+00      9.99807E+00
 8      5.00000E+00      1.38889E—01      9.99228E+00      9.99807E+00
 9      3.00000E+00      6.94444E—02      9.99807E+00      1.00000E+01
10      2.00000E+00      6.94444E—02      1.00000E+01      1.00000E+01

   ITERACIONES EMPLEADAS = 10

   COTA INFERIOR DEL INTERVALO FINAL = 5.00000000E+00

   COTA SUPERIOR DEL INTERVALO FINAL = 5.06944444E+00

   VALOR MAXIMO ENCONTRADO DE LA FUNCION = 1.00000000E+01

   COTA PARA LA QUE SE OBTUVO EL VALOR MAXIMO = 5.00000000E+00
```

En la sección 6.3 se estudia un método de búsqueda secuencial para funciones de varias variables independientes, que en cada paso hace uso del programa A.15 para maximizar.

6.3. TECNICAS DE GRADIENTE

6.3.1 Inicialización

En la introducción al presente capítulo se señaló que los métodos de optimización pertenecen a dos tipos básicos, los de gradiente y los de enumeración. Los primeros tienen la siguiente característica. Dada una función:

$$M = M(x_1, x_2, \ldots, x_n) \qquad (6.1.1)$$

que hay que maximizar o minimizar, *se empieza encontrando para un punto $x_o = (x_{1o}, x_{2o}, \ldots x_{no})$ el valor de la función y su gradiente en este punto. Este paso se conoce con el nombre de inicialización del problema. *Posteriormente se encuentra la dirección para la cual la función $M(\underline{x})$ tiene el máximo aumento en valor, si el problema es de maximización o la mayor disminución en su valor para problemas de minimización. En un problema de maximización debe tenerse por lo tanto:

*Encuentre primero $M(\underline{x_o})$.

*Encuentre la dirección para la cual la función $M(\underline{x})$ varía más rápidamente de valor.

$$M(x_{1o}, x_{2o}, \ldots, x_{no}) < M(x_{1o} + \Delta x_1, x_{2o} + \Delta x_2, \ldots, x_{no} + \Delta x_n) \qquad (6.3.1)$$

Las técnicas de búsqueda permiten encontrar valores de Δx_1, Δx_2, . . ., Δx_n para los cuales la función $M(x)$ varía más rápidamente. Con el objeto de poder ilustrar gráficamente diversos conceptos que se emplean en esta sección considera que la función $M(\underline{x})$ solamente tiene dos variables independientes x_1 y x_2.

Sea

$$\underline{x}_0 = \begin{bmatrix} x_{10} \\ \cdot \\ x_{20} \end{bmatrix}$$

y

$$M_0 = M\ (x_{10}, x_{20})$$

Las derivadas parciales de la función $M(x_1, x_2)$ pueden estimarse en el punto (x_{10}, x_{20}) de la siguiente manera. Para calcular $\left.\frac{\delta M}{\delta x_1}\right|_{\underline{x}_0}$ se incrementa el valor de x_1 en Δx_1 y se mantiene constante la variable x_2. El valor de la derivada parcial está dada aproximadamente por:

$$\left.\frac{\delta M}{\delta x_1}\right|_{\underline{x}_0} \cong \frac{M(x_{10} + \Delta\ x_1, x_{20}\ ) - M(x_{10}, x_{20})}{\Delta x_1} \cong \frac{\Delta M'}{\Delta x_1} \tag{6.3.2}$$

La figura 6.3.1 ilustra la evaluación de esta derivada. En esta figura

$$\text{tg}\ \alpha = \left.\frac{\delta M}{\delta x_1}\right|_{\underline{x}_0} = \frac{\Delta M'}{\Delta x_1}$$

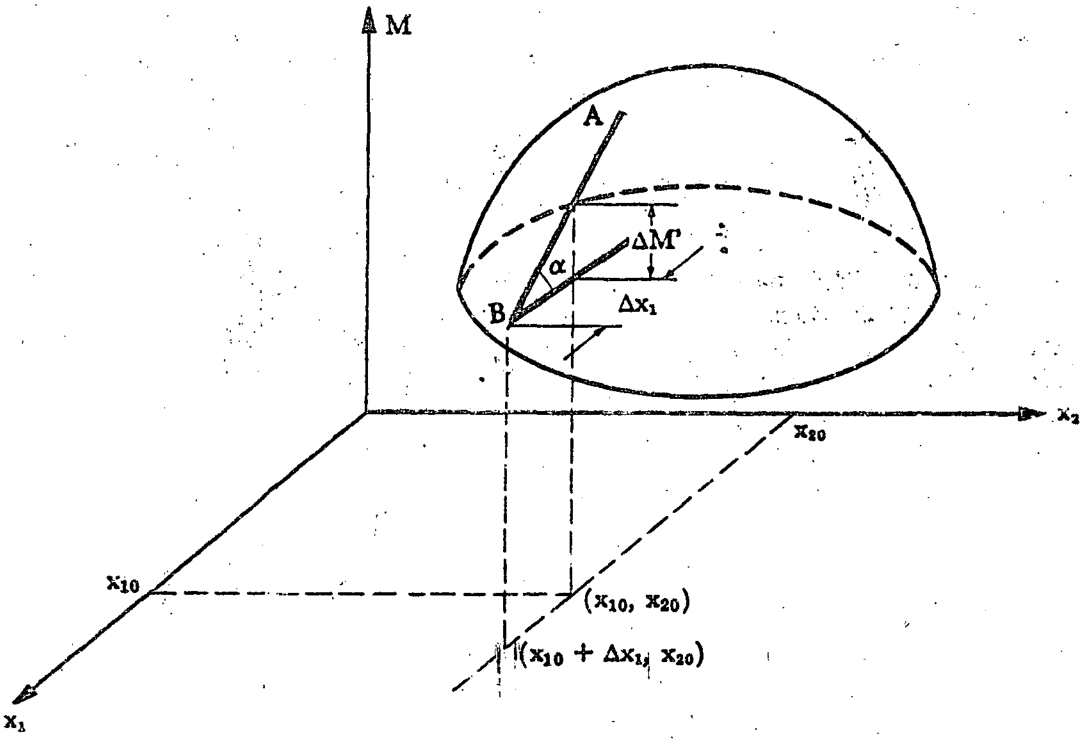


Fig. 6.3.1 Cálculo de la derivada parcial $\frac{\delta M}{\delta x_1}$

Para el cálculo de la derivada parcial $\frac{\delta M}{\delta x_2}$ se emplea la siguiente relación:

$$\left.\frac{\delta M}{\delta x_2}\right|_{x_0} \cong \frac{M(x_{10}, x_{20} + \Delta x_2) - M(x_{10}, x_{20})}{\Delta x_2} \cong \frac{\Delta M''}{\Delta x_2} \qquad (6.3.3)$$

La figura 6.3.2 ilustra el cálculo de esta derivada parcial. En esta última figura

$$\operatorname{tg} \beta = \left.\frac{\delta M}{\delta M}\right|_{x_0} = \frac{\Delta M^{M''}}{\Delta x_2}$$

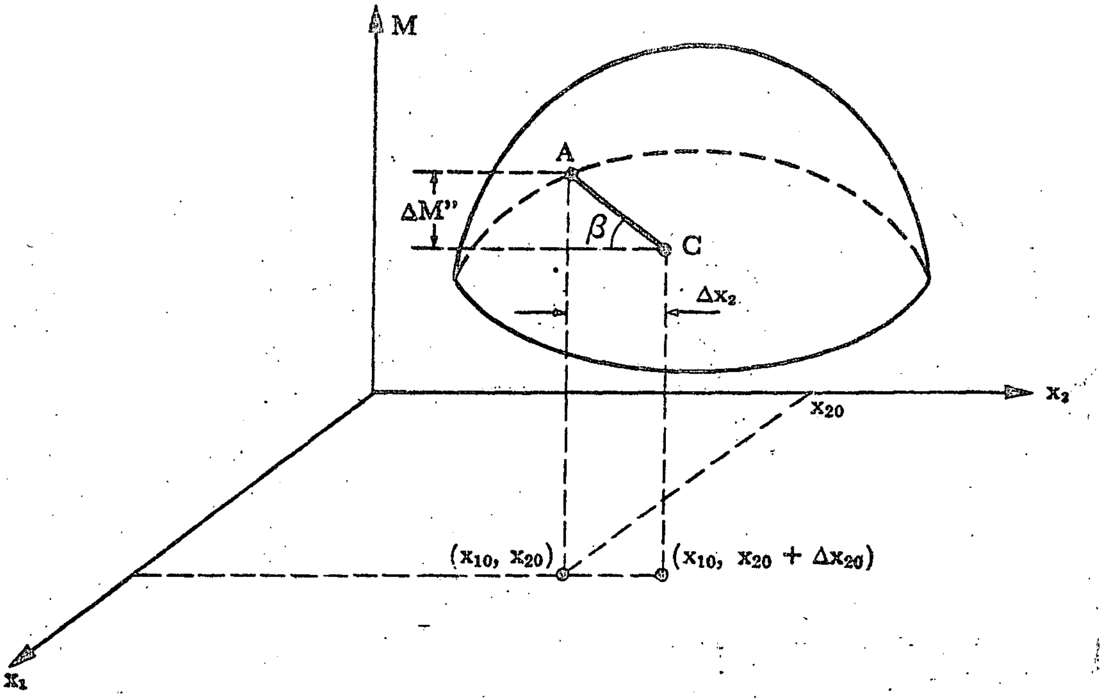


Fig. 6.3.2 Cálculo de la derivada parcial $\frac{\delta M}{\delta x_2}$

En la figura 6.3.3 aparecen los tres puntos A, B, C de las figuras 6.3.1 y 6.3.2. Estos tres puntos definen un plano. Si los incrementos Δx_1 y Δx_2 de las variables x_1 y x_2 disminuyen, el plano ABC tiende a ser tangente a la superficie $M = M(x_1, x_2)$ en el punto (x_{10}, x_{20}).

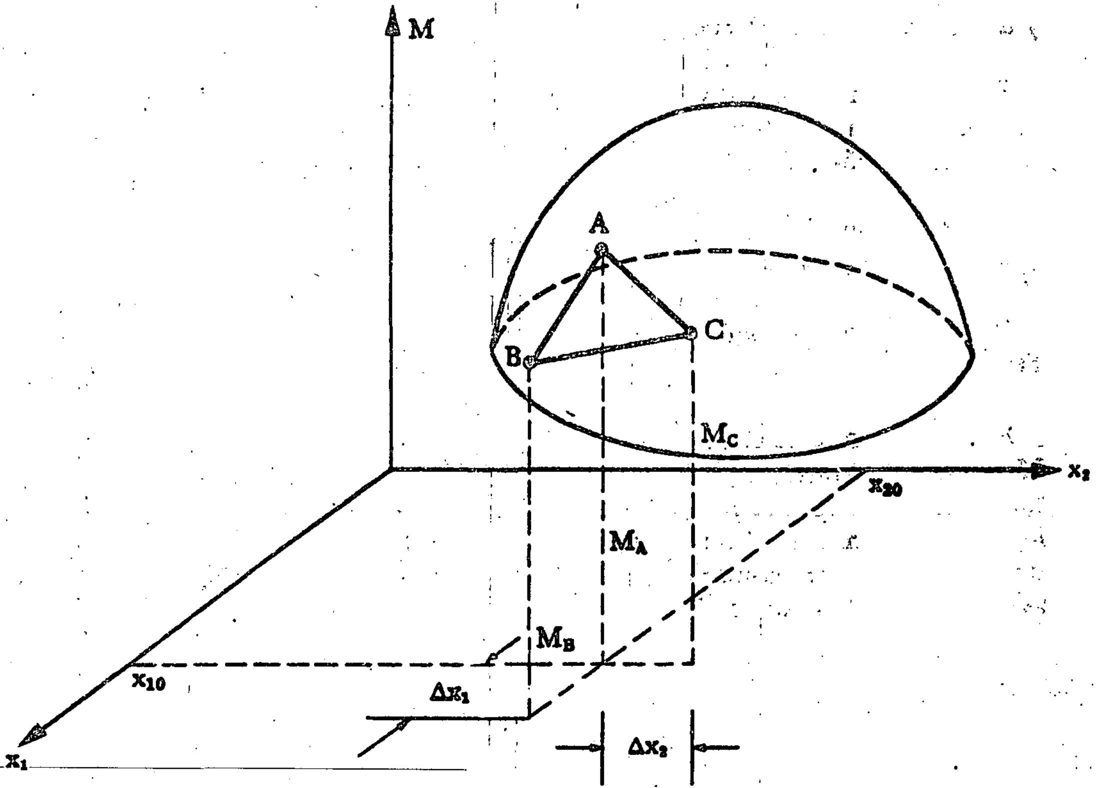


Fig. 6.3.3 Determinación del plano tangente a una superficie.

A continuación se estudia cómo puede determinarse la ecuación de dicho plano.

*La ecuación de un plano en el espacio de tres dimensiones está dada por:

*Ecuación del plano.

$$M(x_1, x_2) = m_0 + m_1 x_1 + m_2 x_2 \qquad (6.3.4)$$

Aplicando esta ecuación a los tres puntos A, B y C de la figura 6.3.3 se tiene:

$$(1)\ M_A = m_0 + m_1 x_{10} + m_2 x_{20}$$

$$(2)\ M_B = m_0 + m_1 (x_{10} + \Delta x_1) + m_2 x_{20} \qquad (6.3.5)$$

$$(3)\ M_C = m_0 + m_1 x_{10} + m_2 (x_{20} + \Delta x_2)$$

Restando la 2da. ecuación de la 1ra. se obtiene:

(1) - (2)

$$M_B - M_A = m_1 \Delta x_1 \qquad (6.3.6)$$

y restando la 3ra. de la 1ra.

(1) - (3)

$$M_C - M_A = m_2 \Delta x_2 \qquad (6.3.7)$$

De la relación (6.3.6) y recordando la figura 6.3.1 se obtiene:

$$m_1 = \frac{M_B - M_A}{\Delta x_1} = \frac{\Delta M'}{\Delta x_1}$$

Por lo tanto de acuerdo con la relación (6.3.2) se tiene:

$$m_1 = \left.\frac{\delta M}{\delta x_1}\right|_{\underline{x}_0} \qquad (6.3.8)$$

De la fórmula (6.3.7) y de la figura 6.3.2 se llega a:

$$m_2 = \frac{M_C - M_A}{\Delta x_2} = \frac{\Delta M''}{\Delta x_2}$$

y de acuerdo con la relación (6.3.3) se tiene:

$$m_2 = \left.\frac{\delta M}{\delta x_2}\right|_{\underline{x}_0} \qquad (6.3.9)$$

*Para no tener que evaluar en la relación (6.3.4) la constante m_o conviene emplear como variable independiente los incrementos de la función $M(x_1, x_2)$.

*Evítese el cálculo de m_o

Para un punto D en la vecindad del punto A, con coordenadas $x_{10} + \Delta x_1$, $x_{20} + \Delta x_2$ se tiene:

$$M_D = m_o + m_1\,(x_{10} + \Delta x_1) + m_2\,(x_{20} + \Delta x_2) \qquad (6.3.10)$$

Restando esta ecuación de la correspondiente al punto A se tiene:

$$\Delta M = M_D - M_A = m_1\,\Delta x_1 + m_2\,\Delta x_2 \qquad (6.3.11)$$

Esta ecuación permite calcular el incremento de la función $M(x_1, x_2)$ que corresponde a incrementos arbitrarios Δx_1, Δx_2 de las variables independientes x_1, x_2. La figura 6.3.4 ilustra esta idea. De las fórmulas (6.3.8) y (6.3.9) se sabe que los coeficientes m_1 y m_2 son precisamente los gradientes de la función $M(x_1, x_2)$. Sustituyendo estos valores en (6.3.11) se tiene:

$$\Delta M = \left.\frac{\delta M}{\delta x_1}\right|_{\underline{x}_0} \Delta x_1 + \left.\frac{\delta M}{\delta x_2}\right|_{\underline{x}_0} \Delta x_2 \qquad (6.3.12)$$

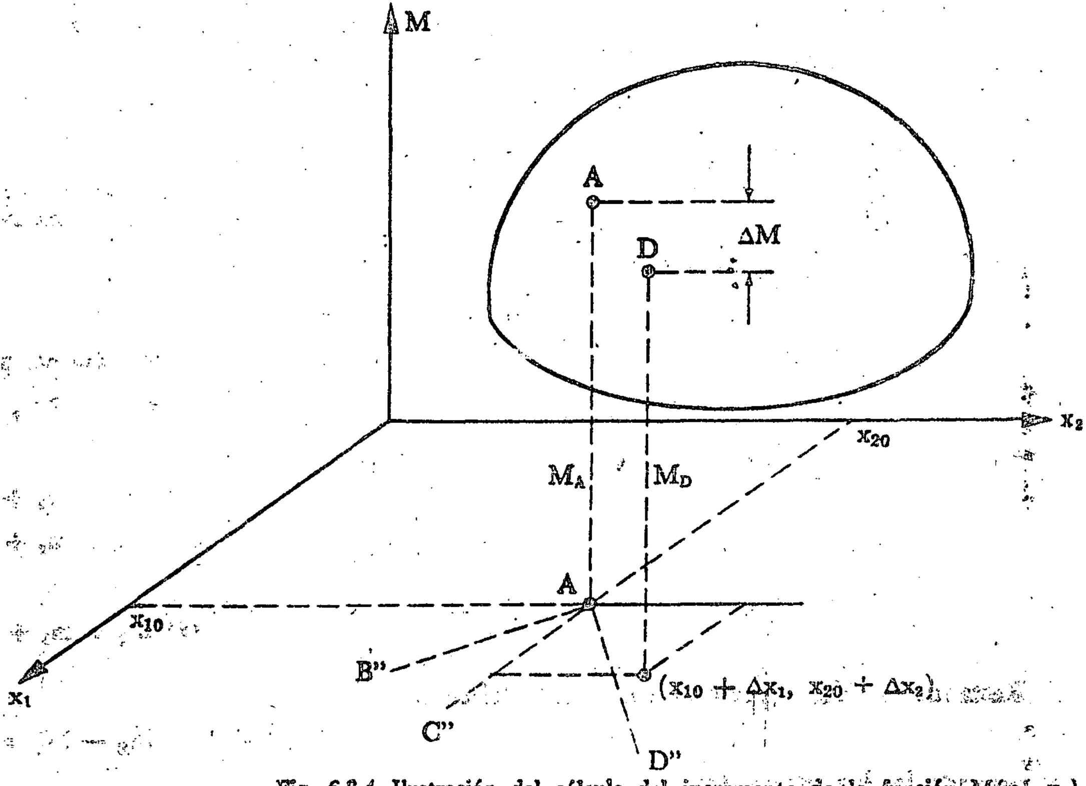


Fig. 6.3.4 Ilustración del cálculo del incremento de la función $M(x_1, x_2)$.

El lector reconocerá de inmediato, en esta relación los primeros dos términos del desarrollo de una serie de Taylor, de acuerdo con la relación (6.2.15).

$$\Delta M\ (\Delta x_1, \Delta x_2) = M\ (x_{10}, + \Delta x_1, x_{20} + \Delta x_2) - M\ (x_{10}, x_{20}) = \frac{\delta M}{\delta x_1}\bigg|_{\underline{x}_0} \Delta x_1 + \frac{\delta M}{\delta x_2}\bigg|_{\underline{x}_0} \Delta x_2 \qquad (6.2.15)$$

*En cálculo se define con el nombre de gradiente, de la función $M(x_1, x_2)$ y se representa con ∇M al siguiente vector de renglón:

*Gradiente de una función.

$$\nabla M = \left(\frac{\delta M}{\delta x_1} \frac{\delta M}{\delta x_2} \right) \qquad (6.3.13)$$

*Si además se define el vector de columna "incremento de las variables independientes" Δx:

*Vector incremento.

$$\Delta \underline{x} = \begin{bmatrix} \Delta x_1 \\ \Delta x_2 \end{bmatrix} \qquad (6.3.14)$$

*La fórmula (6.3.12) para el cálculo del incremento de la función $M(x_1, x_2)$ puede escribirse de la siguiente forma compacta, si se introducen los dos vectores previamente definidos:

*Cálculo del incremento de una función.

$$\Delta M\ (\Delta x_1, \Delta x_2) = \nabla M\bigg|_{\underline{x}_0} \Delta \underline{x} \qquad (6.3.15)$$

Està relación es igualmente válida para el cálculo de incrementos ΔM de funciones de más de dos variables. Si n es el número de variables de la función objetivo (6.1.1).

$$M\ (x_1, x_2, \ldots, x_n) = M\ (x_1, x_2, \ldots, x_n) \qquad (6.1.1.)$$

*el gradiente ∇M de la función se define como:

*Gradiente de la función.

$$\nabla M = \left(\frac{\delta f}{\delta x_1}, \frac{\delta f}{\delta x_2}, \ldots \frac{\delta f}{\delta x_n} \right) \qquad (6.3.13a)$$

*y el incremento de las variables independientes como:

*Vector incremento.

$$\Delta \underline{x} = \begin{bmatrix} \Delta x_1 \\ \Delta x_2 \\ \cdot \\ \cdot \\ \cdot \\ \Delta x_n \end{bmatrix} \qquad (6.3.14a)$$

La fórmula (6.3.15)

$$\Delta M\bigg|_{\underline{x}_0} = \nabla M\bigg|_{\underline{x}_0} \Delta x \qquad (6.3.15)$$

permite calcular el incremento de una función alrededor de un punto $\underline{x}_0$. *Es decir, si se conoce $M(x_{10}, x_{20}, \ldots, x_{n0})$ y si se desea $M(x_{10} + \Delta x_1, \ldots x_{n0} + \Delta x_n)$ este valor puede calcularse de la siguiente manera en forma aproximada:

*Cálculo de $M\ (x_0 + \Delta x)$ a partir de $M\ (x_0)$

$$M\ (x_{10} + \Delta x_1, x_2 + \Delta x_2, \ldots, x_n + \Delta x_n) = M\ (x_{10}, x_{20}, \ldots, x_n) + \Delta M\bigg|_{\underline{x}_0}$$

donde $\Delta M\big|_{\underline{x}_0}$ está dado por la relación (6.3.15).

*Para evaluar el gradiente hay que calcular derivadas parciales.

*Desde luego que es necesario evaluar el gradiente de la función. Este puede evaluarse recordando las relaciones (6.3.2) y (6.3.3). Estas expresiones señalan que la derivada parcial de la función M(x̲)
con respecto a la i' si una variable está dada por:

$$\frac{\delta M}{\delta x_i} = \frac{M(x_{10}, x_{20}, \ldots, x_{i-1,o}, x_{io} + \Delta x_i, x_{j+1,o}, \ldots x_n) - M(x_{10}, \ldots X_{io}, \ldots x_{no})}{\Delta x_i} \qquad (6.3.16)$$

Esta fórmula señala que la i'sima derivada parcial puede obtenerse calculando el incremento de la función M(x̲) si solamente aumenta de valor la i'sima variable y dividiéndolo entre el valor de ese incremento. El siguiente ejemplo sirve para ilustrar el cálculo del incremento de una función empleando el concepto de gradiente.

Ejemplo 6.3.1

$$y = x_1{}^2 x_2{}^3$$

calcule

$$y\Big|_{(1,1)} \quad y \quad y\Big|_{(1.05,1)}$$

*Dada la función $y = x_1^2 x_2^3$ calcule el valor de la función para
y para

$$\underline{x}_o = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \qquad \underline{x} = \begin{bmatrix} 1.05 \\ 1.1 \end{bmatrix}$$

empleando la relación (6.3.15) y directamente por sustitución.

Solución

El valor de la función en el punto $\underline{x}_o = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

es:

$$y\Big|_{(1,1)} = 1^2\ 1^3 = 1 \qquad (6.3.17)$$

El valor de las derivadas parciales de la función puede calcularse empleando la relación (6.3.16). Para la 1er. derivada parcial se tiene:

$$\frac{\delta y}{\delta x_1}\Bigg|_{\underline{x}_o} = \frac{(1.05)^2(1)^3 - (1)^2(1)^3}{0.05} = 2.05 \qquad (6.3.18)$$

y para la 2da. derivada parcial:

$$\frac{\delta y}{\delta x_2}\Bigg|_{\underline{x}_o} = \frac{(1)^2(1.1)^3 - (1)^2(1)^3}{0.1} = 3.31 \qquad (6.3.19)$$

Estos valores son solamente aproximados, el valor exacto de estas derivadas es:

$$\frac{\delta y}{\delta x_1}\Bigg|_{\underline{x}_o} = 2x_1\ x_2{}^3 = 2.1.1\ ^3 = 2$$

$$\frac{\delta y}{\delta x_2}\Bigg|_{\underline{x}_o} = x_1{}^2\ 3x^2 = 1^2.3.1^2 = 3$$

Como el lector puede apreciar la diferencia entre el valor aproximado y el valor real de las derivadas es pequeño y será tanto menor cuanto más se disminuya el valor de los incrementos. (Ver Problema 6).

El valor aproximado del incremento de la función de acuerdo con la fórmula (6.3.15) y empleando como vector de incremento a:

$$\Delta \underline{x} = \begin{bmatrix} 1.05 \\ 1.1 \end{bmatrix} - \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 0.05 \\ 0.1 \end{bmatrix}$$

$$\Delta y \Big|_{\underline{x}_0} = (2.05,\ 3.31) \begin{bmatrix} 0.05 \\ 0.1 \end{bmatrix} = 0.4335$$

y el valor de la función en

$$x = \begin{bmatrix} 1.05 \\ 1.1 \end{bmatrix}$$

será:

$$y \Big|_{\underline{x}} = y \Big|_{\underline{x}_0} + \Delta y \Big|_{\underline{x}_0} = 1 + 0.4335 = 1.4335$$

Este valor es solamente aproximado ya que el valor real es de:

$$y \Big|_{\underline{x}_0} = (1.05)^2\ (1.1)^3 = 1.4674$$

Este ejemplo ilustra el empleo de la relación (6.3.16) para el cálculo de las derivadas parciales, de la fórmula (6.3.13) para la determinación del gradiente de una función y finalmente de la expresión (6.3.15) para la evaluación aproximada del incremento. Como ilustra el problema 6, estas fórmulas son tanto más exactas, cuanto menores son los incrementos.

*Para el problema de optimización que nos interesa en esta sección, la utilidad de la fórmula (6.3.16) para el cálculo de las derivadas parciales estriba en que permite su cálculo sin necesidad de tener que realizar la operación de derivación. Esto constituye una gran ventaja, sobre todo si se emplea la computadora digital para realizar los cálculos, como es lo más probable, o se desconoce la expresión algebráica de la función por optimizar.

***No es necesario encontrar derivados parciales para evaluar el gradiente.**

Con el cálculo de la función M(x) en el punto arbitrario $\underline{x}_0$ y el cálculo del gradiente en este punto termina la fase de inicialización del problema. A continuación se señala cómo debe procederse para encontrar el máximo o el mínimo de la función.

***Con la evaluación de M ($\underline{x}_0$) y $\nabla M \Big|_{\underline{x}_0}$. termina la inicialización.**

6.3.2. Búsqueda

Una vez inicializado el problema, es decir, conocido $M(x_{10}, x_{20}, \ldots, x_{n0})$ y el gradiente $\nabla M \Big|_{\underline{x}_0}$ es necesario *encontrar qué incremento *$\Delta \underline{x}$ debe dársele a las variables independientes, que son las componentes del vector $\underline{x}$ para que la función objetivo "mejore" de valor (aumente en un problema de maximización o disminuya en uno de minimización).

***Encuentre $\Delta \underline{x}$ para que $M(\underline{x})$ "mejore" de valor.**

*En el cálculo se demuestra que la función $M(\underline{x})$ varia más rápidamente si la variable independiente se incrementa en dirección del gradiente. A continuación demostraremos esta aseveración empleando multiplicadores de Lagrange introducidos en la sección 6.2.3.

*Varíe $\underline{x}$ en dirección del gradiente.

Para ilustrar esta demostración se volverá a emplear una función de dos variables. La extensión de la demostración a funciones de n variables es inmediata.

*El problema consiste en determinar en qué dirección deben incrementarse las variables x_1 y x_2 para que la función $M(x_1, x_2)$ tenga su mayor rapidez de variación. Volviendo a hacer referencia a la figura 6.3.4, el problema consiste en determinar si el nuevo valor de $\underline{x}$ debe estar sobre la recta A' B'' C'', A' D'' o cualquier otra que parta del punto A', para que la función $M(x_1, x_2)$ varíe más rápidamente de valor.

*En qué dirección debe variar $\underline{x}$ para que la rapidez de variación de $M(\underline{x})$ sea máxima.

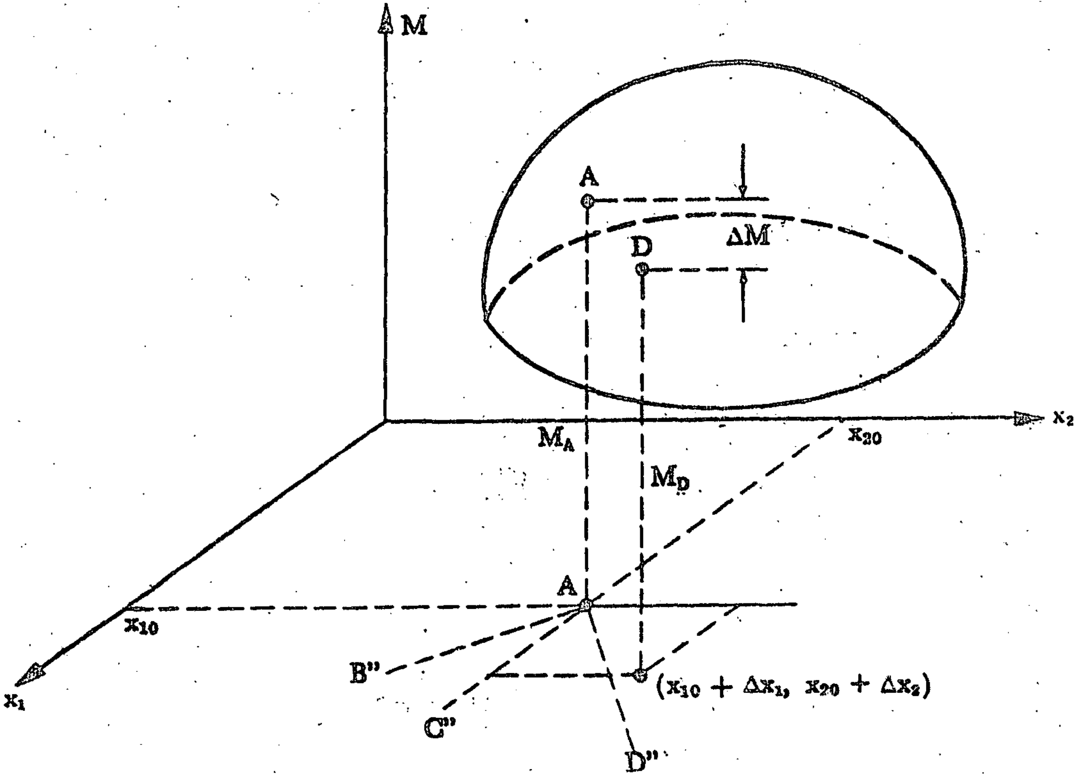


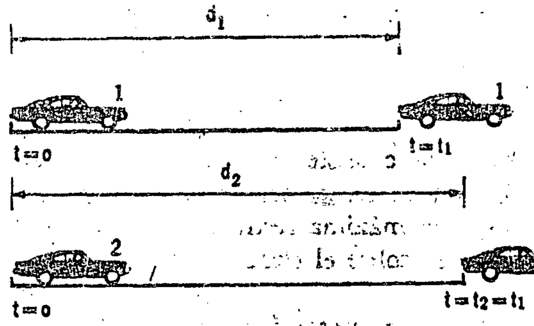
Fig. 6.3.4 Ilustración del cálculo del incremento de la función $M(x_1, x_2)$ (repetición).

*Para medir rapideces de variación, se limita la variación de la variable independiente y se miden los incrementos correspondientes de la función. Para aclarar esta idea, considérese que se

*Para comparar variaciones de $M(\underline{x})$ mantenga constante la variación de $\underline{x}$.

están comparando velocidades, que son la rapidez con que se recorren distancias.

Un vehículo tiene mayor velocidad que otro, si en igual tiempo (variable independiente constante) recorre más distancia. La fig. 6.3.5 ilustra esta idea de comparación de variaciones de una determinada función.



como $t_1=t_2$ y $d_2>d_1$, $V_2>V_1$

Fig. 6.3.5 Comparación de las velocidades de dos vehículos, $v_2 > v_1$.

Después de esta aclaración puede comprenderse fácilmente por qué hay que mantener constante la variación de x si se desean comparar variaciones de una función de x, como en este caso $M(x)$.

Supongamos que esta variación es tal que todo valor de x se encuentra sobre el círculo del plano (x_1, x_2) mostrado en la figura 6.3.6. La magnitud de cualquier $\underline{x}$ sobre este círculo y la

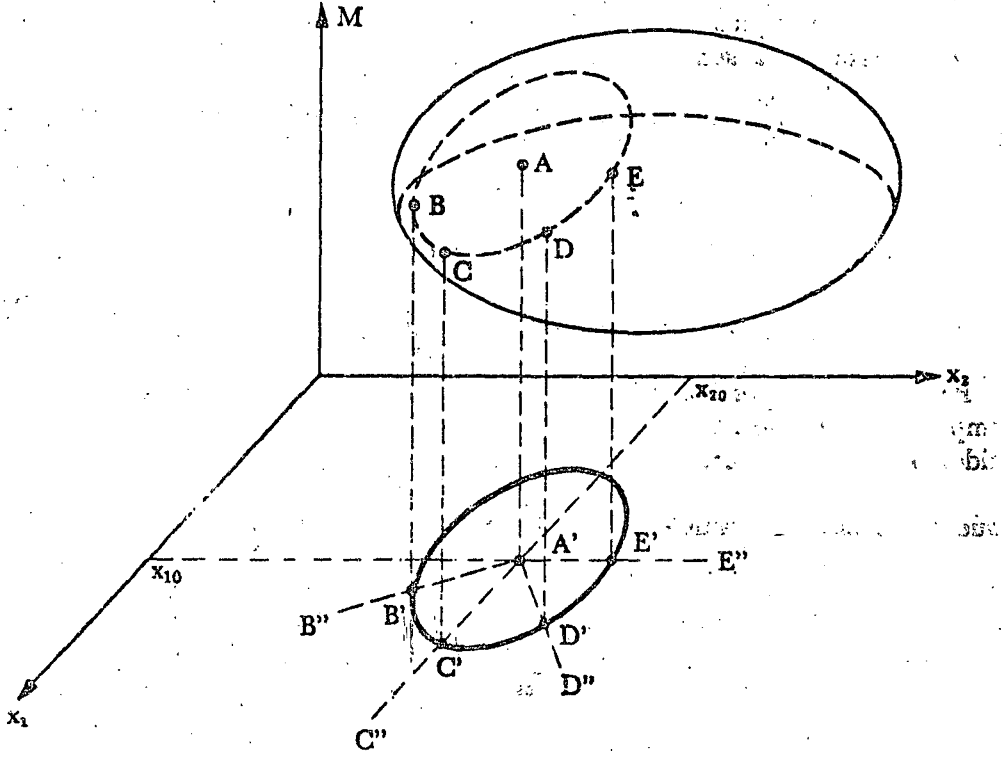


Fig. 6.3.6 Búsqueda por gradiente.

magnitud de $\underline{x}_o$ difieren precisamente en el radio r del círculo, como ilustra la figura 6.3.7.

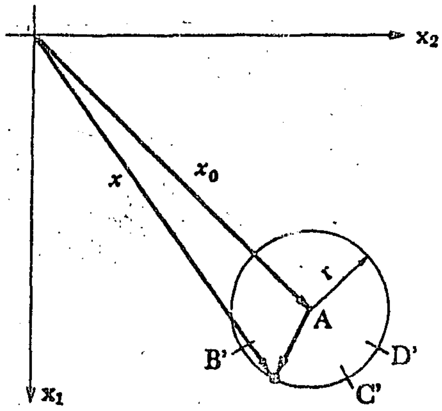


Figura 6.3.7 Variaciones del vector x.

El problema consiste en encontrar la dirección en que debe variar $\underline{x}$ a partir de $\underline{x}_o$ para que la variación ΔM de la función $M(x_1, x_2)$, sea máxima restringiendo a que el extremo del vector x se encuentre sobre el círculo de radio r y centro en A.

La formulación matemática del problema es por lo tanto.

$$\text{máx } \Delta M\Big|_{\underline{x}_o} = \nabla M\Big|_{\underline{x}_o} \Delta\underline{x} \qquad (6.3.15)$$

dado que

$$|\ \underline{x} - \underline{x}_o\ | = r \qquad (6.3.20)$$

donde

$$\underline{x} - \underline{x}_o = \Delta\ \underline{x}$$

*En general el símbolo $|x|$ representa la magnitud del vector $\underline{x}$. Aunque es posible definirla de varias maneras, en esta obra se empleará como expresión para la magnitud de un vector, a la raíz cuadrada de la suma de los cuadrados, es decir:

*$|x|$ es la magnitud del vector $\underline{x}$.

|||

Raíz cuadrada de la suma de los cuadrados de las componentes del vector $\underline{x}$.

Si

$$\underline{x} = \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_n \end{bmatrix}$$

su magnitud es: $$|x| = \sqrt{\sum_{i=1}^{n} x_i^2} \qquad (6.3.21)$$

Para resolver el problema de maximización propuesto, puede emplearse el método de los multiplicadores de *Lagrange introducidos en la sección 6.23. La restricción del problema es:

*Restricción

$$|\Delta\underline{x}| = r$$

que puede escribirse también de la siguiente forma:

$$|\Delta\underline{x}|^2 - r^2 = 0$$

El lagrangiano pone este problema, de acuerdo con la relación (6.2.23), es:

$$L\,(x_1, x_2, \lambda) = \nabla M\,\Delta x - \lambda\,\{\Delta x^2 - r^2\}$$

$$= \left.\frac{\delta M}{\delta x_1}\right|_{\underline{x}_0} \Delta x_1 + \left.\frac{\delta M}{\delta x_2}\right|_{\underline{x}_0} \Delta x_2 - \lambda\,\{\Delta x_1^2 + \Delta x_2^2 - r^2\}$$

Igualando a cero las derivadas parciales se tiene

$$\frac{\delta L}{\delta \Delta x_1} = \left.\frac{\delta M}{\delta x_1}\right|_{\underline{x}_0} - 2\lambda\,\Delta x_1 = 0$$

$$\frac{\delta L}{\delta \Delta x_2} = \left.\frac{\delta M}{\delta x_2}\right|_{\underline{x}_0} - 2\lambda\,\Delta x_2 = 0$$

$$\frac{\delta L}{\delta \lambda} = -\Delta x_1{}^2 - \Delta x_2{}^2 + {}_1^2 = 0$$

De las primeras relaciones se obtiene de inmediato:

$$\Delta x_1 = \frac{1}{2\lambda}\left.\frac{\delta M}{\delta x_1}\right|_{\underline{x}_0}$$

$$\Delta x_2 = \frac{1}{2\lambda}\left.\frac{\delta M}{\delta x_2}\right|_{\underline{x}_0}$$

Recordando la definición de gradiente (6.3.16) se tiene:

$$\Delta \underline{x} = \frac{1}{2\lambda}\,\nabla M\Big|^{T}_{\underline{x}_0} \qquad (6.3.22a)$$

Esta relación indica que los componente del vector Δx deben ser proporcionales, con una constante de $\frac{1}{2\lambda}$ a las componentes del vector gradiente, transpuesto, ∇M^T calculadas en $\underline{x}_0$.

Falta por determinar el valor de λ. A continuación se presenta la teoría previamente estudiada en forma de una serie de pasos, un llamado algoritmo, para su programación digital.

6.3.3 Algoritmo de búsqueda

Los pasos que deben seguirse para buscar el máximo o mínimo de una función ($M\underline{x}$) por el método descrito en las secciones 6.3.1 y 6.3.2 son:

Paso 1:

Seleccione un punto $\underline{x}_0$ para inicializar la búsqueda.

Paso 2:

Evalúe el gradiente en ese punto empleando las relaciones (6.3.13a) y (6.3.16).

$$\nabla M\Big|_{\underline{x}_0} = \left(\frac{\delta M}{\delta x^1}, \frac{\delta M}{\delta x_2}, \ldots, \frac{\delta M}{\delta x_n} \right)_{\underline{x}_Q} \qquad (6.3.13a)$$

$$\frac{\delta M}{\delta x_i}\Big| = \frac{M(x_{10}, \ldots x_{i-1,0}, x_{i0} + \Delta x_i, x_i, x_{i+1,0}, \ldots x_n) - M(x_{10}, \ldots, x_{10}, \ldots x_{no})}{\Delta x_i} \qquad (6.3.16)$$

Paso 3:

Calcule el incremento de la variable independiente de acuerdo con la relación

$$\Delta \underline{x} = \varrho \, \nabla M\Big|_{\underline{x}_0}^{T} \qquad (6.3.22b)$$

donde el factor ϱ_o tiene por valor

$$\varrho_o = \frac{1}{2\lambda} \qquad (6.3.23)$$

Sea ϱ_o *el valor $\frac{1}{2\lambda}$ que maximiza la función

Paso 4:

Encuentre un nuevo punto x_1, para el cual se tiene que

$$M(\underline{x}_0) < M(\underline{x}_1)$$

Para encontrar los valores de ϱ_o que proporcionan el máximo incremento de la función $M(\underline{x})$ en $\underline{x} = \underline{x}_o$, es necesario expresar a la variable dependiente $(M\underline{x})$ como función de ϱ_o. Sustituyendo al incremento $\Delta\underline{x}$ por la relación (6.3.22b) se tiene:

$$\underline{x} = \underline{x}_o + \Delta\underline{x}$$

$$M(\underline{x}) = M(\underline{x}_o) + \varrho_o \, \nabla M\Big|_{\underline{x}_o}^{T}$$

$$M(\underline{x}) = M^1(\varrho_o) \qquad (6.3.24)$$

La igualdad (6.3.24) señala que es posible expresar a la variable dependiente $M(\underline{x})$ como función de la variable escolar ϱ_o. Mediante una búsqueda es posible determinar el valor de ϱ_o que maximice $M'(\varrho_o)$.

Este valor se emplea en la ecuación (6.3.22b) para encontrar el incremento $\Delta\underline{x}$.

en un problema de maximización $\underline{x}_1$ está dado por

$$\underline{x}_1 = \underline{x}_o + \varrho_o \, \nabla M\Big|_{\underline{x}_o}^{T}$$

Paso 5:

Calcule el valor de la función para (x_1). Si en un problema de maximización $M(x_1) > M(x_0)$ continúe al paso 6, si no pare el procedimiento. El punto (x_0) será el mejor que permite calcular este procedimiento.

Paso 6:

Considere al punto x_1 como nuevo punto inicial y vuelva al paso 2.

Este procedimiento puede programarse y en el apéndice A aparece el programa A16 que realiza este tipo de búsqueda.

Para familiarizar al lector con este procedimiento de búsqueda se le aplica en el siguiente ejemplo:

Ejemplo 6.3.2

Obtenga el mínimo de función

$$y = x_1^2 + x_2^2 + x_1 x_2 - x_1 + x_2$$

por diferenciación, y empleando el método de búsqueda de gradiente.

Solución:

De acuerdo con la fórmula (6.2.1) el mínimo (o máximo) de la función debe encontrarse para aquellos puntos (x_1, x_2) para los cuales

$$\frac{\delta y}{\delta x_1} = 0$$

$$\frac{\delta y}{\delta x_2} = 0$$

Efectuando estas operaciones se tiene de inmediato:

$$\frac{\delta y}{\delta x_1} = 2x_1 + x_2 - 1 = 0$$

$$\frac{\delta y}{\delta x_2} = 2x_2 + x_1 + 1 = 0$$

Este sistema de ecuaciones tiene como solución:

$$x = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

Para este punto el valor de la función es:

$$y_{min}^{(1,-1)} = -1$$

Empleando el método iterativo la solución se obtiene de la siguiente forma:

Paso 1

Se inicia el problema con

por ejemplo

$$x_0 = \begin{bmatrix} 0.1 \\ 0.1 \end{bmatrix}$$

Paso 2

Se calcula el gradiente de la función en este punto. En este caso

$$\nabla y \Big|_{\underline{x}_0} = [0.7,\ 1.3]$$

Paso 3

Se calcula el incremento de la variable independiente de acuerdo con la relación

$$\Delta \underline{x} = \varrho_0 \nabla M \Big|_{\underline{x}_0}^{T} \qquad (6.3.22b)$$

donde ϱ_0 se encontró por búsqueda aleatoria, su valor fue

$$\varrho_0 = -\ 0.895843$$

Por lo que en la primera iteración del programa del apéndice A16 se obtuvieron los siguientes incrementos

$$\Delta \underline{x} = \begin{bmatrix} 0.627087 \\ -0.96459 \end{bmatrix}$$

Después del 1er. ciclo de iteración se ha encontrado que en el

punto

$$\underline{x}_1 = \begin{bmatrix} 0.7227087 \\ -1.06459 \end{bmatrix}$$

la función tiene un valor menor que en el punto inicial

$$\underline{x}_0 = \begin{bmatrix} 0.1 \\ 0.1 \end{bmatrix}$$

En efecto

$$y(x_1) = -\ 0.903719$$

mientras que

$$y\ (\underline{x}_0) = 0.03$$

Siguiendo con las iteraciones se llega al punto deseado de

$$(\underline{x}) = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

para el cual la función $y(x_1, x_2)$ vale:

$$y(1, -1) = -\ 1$$

Al llegar a este punto el lector forzosamente tendrá que preguntarse cuáles son las ventajas de este método de optimización por búsqueda con respecto a la aplicación de la relación (6.2.1). La solución de las ecuaciones (6.2.1) en general no es tan fácil como en este problema, donde las ecuaciones (6.2.25) resultaron lineales. *En general el sistema de ecuaciones al que dan lugar las ecuaciones (6.2.1) son no lineales. Como para su solución también se requieren métodos iterativos, similares al aquí presentado, no tiene ningún caso obtener primero las derivadas parciales para después aplicar un procedimiento iterativo.

$$\frac{\delta M}{\delta x_i} = 0, \forall i \qquad (6.2.1)$$

***Las ecuaciones (6.2.1) en general son no lineales.**

*Antes de terminar con este tema es necesario indicar que este procedimiento en uno de los pasos de iteración puede "brincarse" el máximo. Si en un momento determinado $M(\underline{x}_o) = M(\underline{x}_1)$ puede suceder que el máximo (o mínimo) se encuentre entre los dos puntos, o que entre $\underline{x}_o$ y $\underline{x}_o$ la función no cambie para seguir creciendo posteriormente.

***El máximo o mínimo de la función puede encontraste entre el punto $\underline{x}_o$ y el $\underline{x}_1$.**

Para determinar si se ha presentado el 2do. caso, puede hacerse una búsqueda incrementando el valor de ϱ_o en una cantidad menor que el valor dado en el paso 3.

Si la función sigue creciendo (o decreciendo) se emplea este último punto para inicializar una nueva iteración.

En caso de encontrarse el máximo (o mínimo) entre $\underline{x}_o$ y $\underline{x}_1$ puede recurrirse a una interpolación.°

Ejemplo 6.3.3.

El siguiente ejemplo ilustra la aplicación del método descrito a un problema de localización de una planta para minimizar los costos de instalación.

Determine la localización más adecuada de una planta, dentro de la zona mostrada. El terreno es horizontal. Es necesario tender tuberías de agua, gas, drenaje y combustible y una línea eléctrica, desde los puntos que muestra la fig. 6.3.8. Además, es necesario construir un camino de acceso a la fábrica desde la carretera que pasa al frente del predio.

Solución:

La función objetivo por minimizar incluirá solamente los costos que dependen de la localización de la planta. Se tendrá para este problema:

$$y(x_1, x_2) = 50\ |x_2| + 15\ \{x_1^2 + (x_2 - 300)^2\}^{\frac{1}{2}} + 50\ \{x_1^2 + (1300 - x_2)^2\}^{\frac{1}{2}} +$$
$$15\ \{(1300 - x_1)^2 + (900 - x_3)^2\}^{\frac{1}{2}} + 20\ \{(1300 - x_1)^2 + x_2^2\}^{\frac{1}{2}}$$

° Ref. 1, Cap. 10.

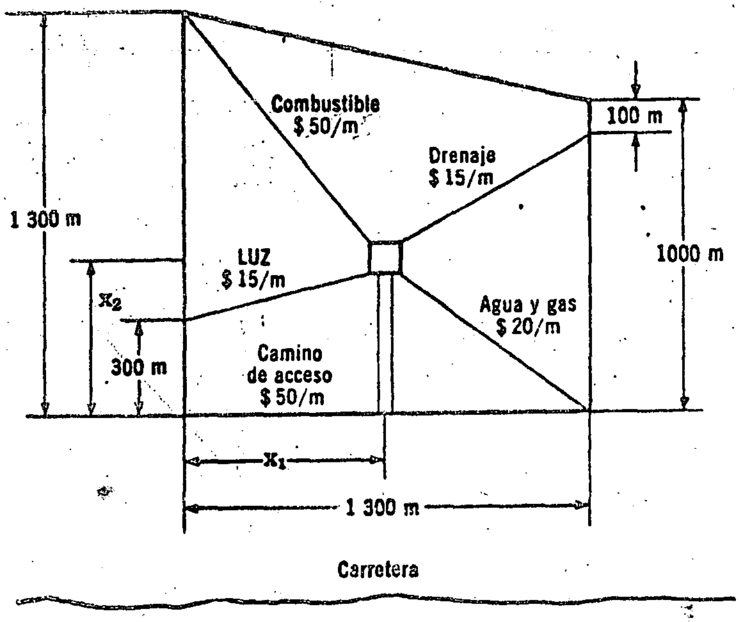


Fig. 6.3.8 Plano de localización de la fábrica del ejemplo 6.3.3.

El lector puede proceder a calcular las derivadas parciales con respecto a x_1, y x_2 e igualarlas a cero. Para la solución del sistema de ecuaciones algebráicas no lineales resultantes, es necesario emplear un método también iterativo.

La tabla 6.3.1 muestra los valores de las coordenadas x_1 y x_2 y de la función costo para diversas iteraciones a partir del punto (500, 500). Para obtener estos resultados se empleó el programa A.16 que ejecuta una búsqueda por gradiente. En cada etapa el valor del parámetro ϱ_0 se encuentra por búsqueda aleatoria. Para esta minimización el programa A.16 emplea como subrutina un programa basado en el A.14.

Tabla 6.3.1 Resultados del programa A.16 para el ejemplo 6.3.3.

```
                    LOS RESULTADOS PARA CADA ITERACION SON
      ϱo                      XI                        X2                        F
                      5.00000000E+02            5.00000000E+02            1.12532023E+05
—2.37119662E+00
                      3.86932534E+02            2.73865068E+02            1.10015658E+05
—4.24713455E—02
                      3.82882151E+02            2.69814685E+02            1.10014699E+05
 3.77092041+—01
                      3.64901001E+02            2.69814685E+02            1.10007395E+05
—8.13736115E—03
—8.13736115E—03
 3.77092041E—01
                      3.64512982E+02            2.87795835E+02            1.09997591E+05
—4.46264156E—02
                      3.60257075E+02            2.87795835E+02            1.09996887E+05
 3.39754126E—02
                      3.58637001E+02            2.87795835E+02            1.09996843E+05
—8.13736115E—03
—8.13736115E—03
—8.13736115E—03

          EL VALOR MINIMO OBTENIDO DE LA FUNCION ES = 1.09996843E+05
                  LOS VALORES QUE OPTIMIZAN LA FUNCION SON
                      VARIABLE            VALOR DE LA VARIABLE
                          1                  3.58637001E+02
                          2                  2.87795835E+02
```

Desde luego que en este ejemplo, como en otros donde la variación posible de $(x_1, x_2, \ldots, x_n)$ está restringida, en cada iteración hay que ver si el punto sigue estando dentro de la zona posible. En el ejemplo 6.3.3, dentro del predio mostrado.

*Este procedimiento de búsqueda podría compararse con la estrategia que puede seguir un alpinista para llegar lo más pronto posible a la cumbre de una montaña. Una posible forma de hacerlo es subir por la recta de mayor pendiente, o empleando el lenguaje del cálculo, siguiendo la dirección que marca el gradiente. Un alpinista que sube una montaña con densa neblina, puede llegar a un punto donde al siguiente paso se baja. De acuerdo con la estrategia que sigue el alpinista, concluye que ha llegado a la cumbre. Puede suceder que en efecto ésta sea la cumbre de la montaña, o solamente un promontorio local. La densa neblina no le permite ver lejos. En el método de búsqueda descrito puede suceder exactamente lo mismo. Como el procedimiento avanza de punto en

***Para subir rápido una montaña siga la ruta de mayor pendiente y siga hasta que el paso siguiente sea de bajada.**

***Puede llegarse a un promontorio local.**

punto puede llegarse a un llamado máximo (o mínimo) local, equivalente a un promontorio local en una montaña, que sin embargo no es el máximo (o mínimo global), o sea el punto donde M(x) es máximo o mínimo en toda la zona posible de variación. *Este procedimiento trabaja sin problema con funciones con un solo máximo o mínimo. Frecuentemente la naturaleza propia del problema permite determinar si se ha encontrado un máximo (o mínimo) global.

***La búsqueda por gradiente trabaja cuando las funciones tienen un solo máximo o mínimo.**

El programa A.16 del apéndice A permite resolver problemas de búsqueda por gradiente. Los problemas 6 a 9 de la sección 6.8 permiten al lector adquirir mayor destreza con este método.

***Programa A.16 de búsqueda por gradiente.**

En la siguiente sección se introduce al lector al análisis marginal, otro método de optimización que tiene importantes aplicaciones en estudios económicos.

***El análisis marginal se emplea en estudios económicos.**

En la siguiente sección se establece además un enlace entre el capítulo 4, en particular entre la sección 4.5 dedicada a funciones de producción y métodos de optimización.

6.5. PROGRAMACION LINEAL

6.5.1 Ejemplos

Existen muchos problemas de optimización cuyo modelo matemático es de tal naturaleza que se pueden resolver con la técnica de optimización conocida con el nombre de programación lineal. Se han desarrollado algoritmos y basados en ellos, programas de computadora digital para la solución de estos problemas.

*La estructura de los problemas que pueden resolverse con esta técnica es siempre la misma, de manera que contando con un buen programa para la solución de éstos, pueden resolverse sin necesidad de tener que escribir programas especiales para la solución de problemas particulares. Los problemas de optimización que se pueden resolver con *la técnica de programación dinámica por otra parte no tiene esta característica y con frecuencia es necesario desarrollar programas particulares para obtener la solución de un problema específico.

***Todos los problemas de programación lineal tienen el mismo modelo matemático.**

***No existen modelos generales para problemas de programación dinámica.**

En esta sección se empezará a ilustrar con ejemplos la formulación de modelos matemáticos que permiten aplicar la programación lineal. A continuación, la ilustración geométrica de la solución del problema de programación lineal, sirve para introducir el método simplex de solución de problemas.

El primer ejemplo ilustra un problema de transporte. Supóngase que una embotelladora tiene dos plantas, una en Tlaxcala y otra en Tehuacán, con capacidad de 7 000 y 13 000 cajas de refrescos al día, además tiene dos centros de consumo que son Puebla y Orizaba, que pueden consumir hasta 12 000 y 8 000 cajas diarias respectivamente. El costo de envío de una caja de refrescos de los diferentes lugares de producción a los diferentes destinos está dado en la tabla 6.5.1.

Ejemplo 6.5.1

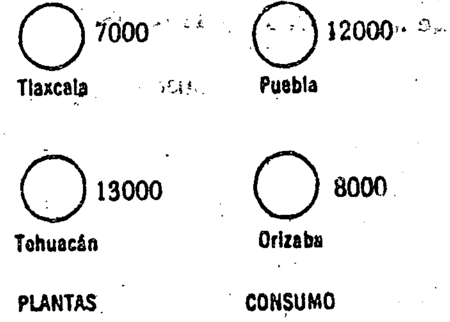


Tabla 6.5.1 Costos de transporte en el ejemplo 6.4.1.

de / a	Tlaxcala 1	Tehuacán 2
Puebla 1	0.8	1.00
Orizaba 2	1.30	0.90

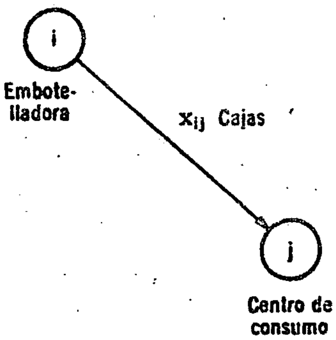
El administrador de la empresa debe determinar cuántas cajas deben enviarse de cada embotelladora a cada centro de consumo, de manera que se satisfagan las siguientes condiciones:

1) Cada embotelladora no puede enviar más cajas que el máximo que puede producir.
2) Cada centro de consumo puede obtener tantas cajas como puede consumir.
3) Deben minimizarse los gastos de transporte.

Para plantear este problema en el marco de las ecuaciones (6.1.1) y (6.1.2).

$$M = M(x_1, x_2, \ldots, x_n) \qquad (6.1.1)$$

$$C_i = C_i(x_1, x_2, \ldots, x_n) \geq O \text{ para } i = 1,2,\ldots\ p$$
$$C_i = C_i(x_1, x_2, \ldots x_n) \leq O \text{ para } i = p+1, \ldots\ r$$
$$C_i = C_i(x_1, x_2 \ldots, x_n) = O \text{ para } i = r+1, \ldots\ n$$
$$(6.1.2)$$



es necesario definir la siguiente variable: x_{ij} es el número de cajas enviadas de la embotelladora situada en la localidad i'sima (i = 1 corresponde a Tlaxcala e i = 2 a Tehuacán) al centro consumidor j'simo (1 es el índice de Puebla y 2 el de Orizaba). Con la introducción de esta variable el problema puede plantearse de la siguiente forma:

Las cajas enviadas de la localidad 1 (Tlaxcala) al centro de consumo 1 (Puebla), que se ha acordado representar con x_{11} más las cajas enviadas de la localidad 1 al centro de consumo 2 (Orizaba), x_{12}, no deben exceder la capacidad de la embotelladora de la localidad 1 que es de 7 000 cajas, es decir,

$$x_{11} + x_{12} \leq 7\,000 \qquad (6.5.1)$$

La figura 6.5.1 ilustra el planteamiento de esta ecuación:

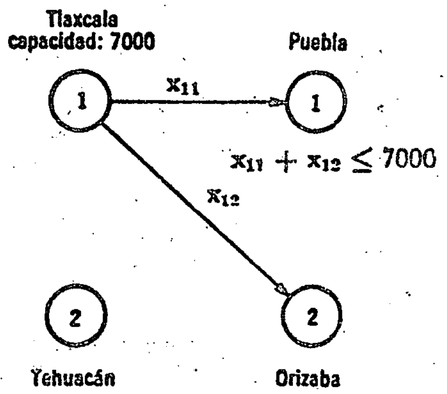


Fig. 6.5.1 Cajas enviadas desde la embotelladora en Tlaxcala.

En forma similar puede establecerse la siguiente ecuación que limite la producción total de la embotelladora de la 2da. localidad a 13 000 cajas, a saber:

$$x_{21} + x_{22} \leq 13\,000 \qquad (6.5.2)$$

La figura 6.5.2 ilustra el planteamiento de otras ecuaciones.

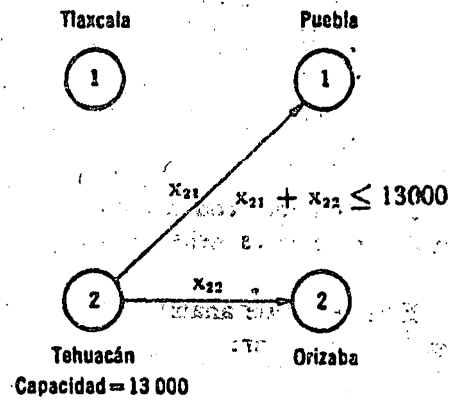


Fig. 6.52 Cajas enviadas desde la embotelladora en Tehuacán.

Por otra parte, se ha señalado que cada centro de consumo puede obtener tantas cajas como desea.

Al centro consumidor 1, Puebla, le llegan x_{11} cajas de Tlaxcala y x_{21} cajas de Tehuacán tal como ilustra la fig. 6.5.3. Por lo tanto, como el consumo de Puebla es de 12 000 cajas:

$$x_{11} + x_{21} \geq 12000 \qquad (6.5.3)$$

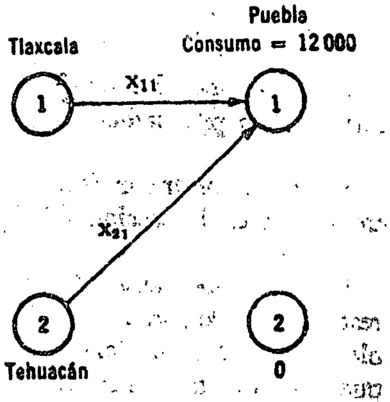


Fig. 6.5.3 Cajas recibidas en Puebla.

Finalmente como última restricción se tiene que las cajas que recibe Orizaba, centro consumidor 2, deben ser iguales o mayor a 8 000 cajas. Se tiene por lo tanto;

$$x_{12} + x_{22} \geq 8\,000 \qquad (6.5.4)$$

La figura 6.5.4 ilustra el significado de esta ecuación.

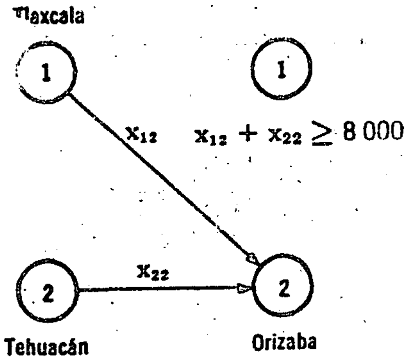


Fig. 6.5.4 Cajas recibidas por Orizaba.

Para terminar con el establecimiento del modelo matemático de este problema es necesario establecer la función objetivo.

El objetivo de análisis es minimizar los costos de transporte que están dados por:

$$M = 0.8\, x_{11} + 1\, x_{21} + 1.3\, x_{21} + 0.9\, x_{22} \qquad (6.5.5)$$

Debe además imponerse la siguiente condición:

$$x_j i \geq 0 \; \forall_i, \forall i \qquad (6.5.6)$$

ya que no tendrán significado valores negativos de envíos de cajas.

En resumen puede decirse que el problema consiste en minimizar la función objetivo.

$$M = 0.8\, x_{11} + 1.\, x_{21} + 1.3\, x_{12} + 0.9\, x_{22} \qquad (6.5.5)$$

Sujeto a las restricciones

$$\begin{array}{lll} x_{11} + x_{12} \leq & 7{,}000 & (6.5.1) \\ x_{21} + x_{22} \leq & 13{,}000 & (6.5.2) \\ x_{11} + x_{21} \geq & 12{,}000 & (6.5.3) \\ x_{12} + x_{22} \geq & 8{,}000 & (6.5.4) \end{array}$$

$$x_{ij} \geq 0, \; \forall_i \text{ y } \forall_j. \qquad (6.5.6)$$

Todos los modelos matemáticos de problemas de programación lineal tienen precisamente esta forma.

Antes de continuar conviene recordar algunas definiciones introducidas en la sección 6.1.2.

*Un conjunto de valores de las variables que satisface todas las restricciones del problema se llama una *solución factible* del problema de programación lineal. Empleando la definición anterior, puede decirse que la solución del problema consiste en encontrar una solución factible que sea óptima. En este caso del problema del transporte una solución factible que minimice la función objetivo (6.5.5).

*La solución factible satisface todas las restricciones.

*Este problema tiene cuatro variables que hay que determinar, x_{11}, x_{12}, x_{21} y x_{22}. Con objeto de visualizar geométricamente la solución de los problemas de programación lineal e introducir otro tipo de problemas de optimización de este tipo, se incluye un segundo ejemplo:

*Supóngase que una compañía de transporte tiene x_1 camionetas de 2 toneladas y x_2 camionetas de 4 toneladas y desea maximizar su capacidad de transporte. La función objetivo es y el problema consiste en maximizar dicha expresión.

*Además la compañía tiene las siguientes restricciones:

*La primera es la siguiente: Las camionetas chicas requieren 1 día de mantenimiento al mes, y las grandes 4 días y la compañía sólo tiene disponibles 24 días de mecánico al mes. Matemáticamente esta restricción se expresa de la siguiente forma:

*La segunda restricción en este problema se refiere a la disponibilidad de andenes de carga. Ambos tipos de vehículo, requieren de igual número de andenes de carga, y que la compañía sólo cuenta con 9 andenes. Empleando las variables x_1 y x_2 esta restricción establece:

*La última restricción se refiere al personal que se requiere para cargarlas. Este personal está restringido a 21 personas. Las camionetas chicas requieren tres personas para cargarlas y las grandes solamente una persona. Se tiene por lo tanto

*Desde luego que las variables x_1 y x_2, número de camionetas de 2 toneladas y de 4 toneladas con que cuenta la compañía respectivamente, no pueden ser negativas, por lo tanto las últimas restricciones en este problema son:

Desde luego existen otros muchos problemas donde puede aplicarse la programación lineal. Entre ellos pueden citarse problemas de mezclado y planeación de la producción como el ejemplo 6.5.4 de la sección 6.5.5.

Después de estos ejemplos se procederá a planear en forma formal el problema de programación lineal y se estudiarán las condiciones que debe satisfacer tanto la función objetivo como las restricciones.

***Variables del problema x_{11}, x_{12}, x_{21} y x_{22}**

Ejemplo 6.5.2

***x_1 camionetas de 2 ton., x_2 camionetas de 4 ton.**

$$m = 2x_1 + 4x_2 \qquad (6.5.7)$$

***Restricciones.**

***Mantenimiento:**
24 días mecánico/mes.

$$x_1 + 4x_2 \leq 24 \qquad (6.5.8)$$

***2da. Andenes de carga:**
9 andenes.

$$x_1 + x_2 \leq 9 \qquad (6.5.9)$$

***3ra. Cargado:**
21 personas.

$$3x_1 + x_2 \leq 21 \qquad (6.5.10)$$

***Ultima:**
no negatividad.

$$x_1 \geq 0;\ x_2 \geq 0 \qquad (6.5.11)$$

6.5.2. Planteamiento formal

*Si se analiza la formulación de los problemas de los dos ejemplos introducidos en la sección anterior, pueden detectarse ciertas variables que se llaman en forma genérica *actividades*.

°Actividades.

*En el ejemplo 6.5.1 las actividades consisten en enviar cajas de refrescos de la embotelladora al centro consumidor y se han representado con los símbolos:

*Envío de cajas de refresco.

x_{ij}, i, j = 1,2

*En el ejemplo 6.5.2 estas actividades consisten en operar camiones de carga y se han empleado los símbolos x_1 y x_2 para representarlas.

*Operación de camiones de carga

x_1, x_2

*Cada actividad queda caracterizada por una variable que se designa como *nivel de actividad*.

*Nivel de actividad.

Además se observa que los problemas de los ejemplos anteriores satisfacen las siguientes condiciones:

1. No negatividad de los niveles, es decir

$x_i \geq 0, \forall_i$

*Tanto las restricciones como la función objetivo son funciones lineales de los niveles de actividad. Al ser lineales estas funciones son *homogéneas* y *aditivas*.

°Funciones objetivo y restricciones son lineales →homogéneas y aditivas.

Una función

$f(x_1, x_2, \ldots, x_n)$

es lineal si dados dos conjuntos:

°Conjuntos de variables

x_i, i = 1, 2,... n y x'_i, i = 1, 2, ...

*y dos constantes cualquiera K y K′ se tiene:

°Constantes K y K'

$$f(Kx_1, + K'x_1', \ldots, Kx_n + K'x'_n) = Kf(x_1, x_2, \ldots, x_n) + K'f(x'_1, x'_2, \ldots, x'_n) \qquad 6.5.12)$$

*La condición de linealidad (6.5.12) es equivalente a dos condiciones. En primer lugar una función lineal tiene un factor constante de escala, es decir.

*Condición de linealidad → factor constante de escala

$$f(Kx_1, Kx_2, \ldots, Kx_n) = Kf(x_1, x_2, \ldots, x_n) \qquad (6.5.13)$$

*y en segundo lugar es aditiva:

°Condición de linealidad → aditividad.

$$f(x_1 + x'_1, x_2 + x'_2, \ldots, x_n + x'_n = f(x_1, x_2, \ldots, x_n) + f(x'_1, x'_2, \ldots, x'_n) \qquad 6.5.14)$$

*Un ejemplo servirá para ilustrar este importante concepto y señalar que funciones del tipo

$$f(x) = a + bx \qquad (6.5.15)$$

*no son lineales. Es decir, si en las funciones hay cargos fijos (el término a) no es posible aplicar directamente el concepto de programación lineal.

°Función no lineal.

Ejemplo 6.5.3.

Determine si las siguientes funciones son lineales y justifique la respuesta.

a) $y = 3x_1 + 2x_2$

b) $y = 3x + 5$

Solución:

a) Como $a3x_1 + b3x'_1 + a2x_2 + bx'_2$
$= a(3x_1 + 2x_2) + b(3x'_1 + 2x'_2)$

se cumple la condición (6.5.12) y la función es lineal.

b) Como $a3x + 5 + b3x' + 5 \neq a(3x + 5)$
$+ b(3x' + 5)$

la función no es lineal.

El °problema de programación lineal por lo tanto puede plantearse de la siguiente forma.

*Hay que determinar el valor de los niveles de actividad $x_1, x_2, \ldots x_n$, que maximicen a la función objetivo:

°Encontrar $\underline{x}$ que maximice:

$$m = c_1 x_1 + c_2 x_2 + \ldots + c_n x_n \quad (6.5.6a)$$

y satisfaga:

sujeto a las siguientes restricciones:

$$ai_1 x_1 + a_{i2} x_2 + \ldots + a_{in} = b_i \quad i = 1, 2, \ldots p$$

$$a_{i1} x_1 + a_{i2} x_2 + \ldots + a_{in} x_n \leq b_i \quad i = p + 1, \ldots r$$

$$a_{i1} x_1 + a_{i2} x_2 + \ldots + a_{in} x_n \geq b_i \quad i = r + 1, \ldots m$$

$$x_j \geq 0 \quad j = 1,2 \ldots n$$

(6.5.16b)

*Los coeficientes C_i de la función objetivo se conocen con el nombre de *coeficientes de costo,* y los coeficientes a_{ij} de las ecuaciones de restricción se llaman *coeficientes estructurales.*

°c_i = coeficientes de costo

a_{ij} = coeficientes estructurales.

Como se ilustra en el ejemplo 6.5.3 un problema de maximización puede siempre convertirse en uno de minimización. Como muestra el sistema de ecuaciones (6.5.16) las restricciones pueden ser del tipo de desigualdad o igualdad. *Para la solución del problema de programación lineal conviene convertir todas las desigualdades en igualdades introduciendo *variables de holgura,* que de preferencia deben de ser positivas. La siguiente desigualdad:

°Variables de holgura > 0 para convertir desigualdades en igualdades.

Desigualdad

$$a_{q1} x_1 + a_{q2} x_2 + \ldots + a_{qn} x_n \leq b_q$$

$+$

Variable de holgura

$x_{n+q} > 0$

↓

Igualdad

puede convertirse en una igualdad introduciendo una variable no negativa ~~positiva~~ x_{n+q} llamada de holgura. En efecto:

$$a_{q1} x_2 + a_{q2} x_2 + \ldots + a_{qn} x_n + x_{n+q} = b_q$$

*Si por otra parte se tiene en la ecuación de restricción la desigualdad en sentido contrario.

°**Desigualdad**

$$a_{q1} x_1 + a_{q2} x_2 + \ldots + a_{qn} x_n \geq b_q$$

$+$

Variable de holgura

$x_{n+q} > 0$

↓

Igualdad

nonegativa

la introducción de la variable de holgura positiva x_{n+q}, convierte la desigualdad en una igualdad, ya que:

$$a_{q1} x_1 + a_{q2} x_2 + \dots + a_{qn} x_n - x_{n+q} = b_1$$

Además, los métodos de solución del problema de programación lineal exigen que los niveles de actividad sean positivos, es decir, $x_i \geq 0, \forall_i$. *Si un nivel de actividad no está sujeto a esta restricción se le puede sustituir por la diferencia de dos niveles de actividad positivos. Supongamos que el nivel x_i no está restringido. Si se introducen las variables

*Si nivel de actividad

$$x_i \leq 6 \geq 0$$

x_i^+ y x_i^- relacionadas con la variable x_i mediante la siguiente diferencia.

$$x_i = x_i^+ - x_i^- \qquad (6.5.17)$$

$$x_i^+ \geq 0, \; x_i^- \geq 0$$

la variable o nivel de actividad original puede ser mayor, igual o menor que cero, sin que las variables x_i^+ y x_i^- tomen valores negativos. El siguiente ejemplo ilustra tanto la introducción de variable de holgura como el empleo de la relación (6.5.17) y la transformación de un problema de minimización en uno de maximización.

Ejemplo 6.5.3

Convierta el siguiente problema de minimización en un problema de maximización, transforme todas las ecuaciones de restricción en igualdades mediante la introducción de variables de holgura y transforme todas las variables en no negativas:

$$\min : m = 3x_1 + 5x_2$$

$$3x_1 + 2x_2 \geq 6$$

$$x_1 - 6x_2 \leq 4$$

$$x_1 \geq 0; \; x_2 \text{ sin restricción}$$

Solución:

se sabe que:

Min. $m = 3x_1 + 5x_2$ es equivalente a:
Max. $-m = -3x_1 - 5x_2$.

Definiendo una nueva función objetivo.

° Nueva función objetivo n:

$$n \equiv -m$$

la función objetivo se convierte en:

$$\downarrow$$

$$\max: n = -3x_1 - 5x_2$$

Para convertir las dos desigualdades de restricción en igualdad es necesario introducir dos nuevas variables x_3 y x_4 para realizar los siguientes cambios en las restricciones.

$$x_1 - 6x_2 \leq 4 \rightarrow x_1 - 6x_2 + x_4 = 4$$

$$3x_1 + 2x_2 \geq 6 \rightarrow 3x_1 + 2x_2 - x_3 = 6$$

*Finalmente la variable x_2, no restringida debe sustituirse por la diferencia de dos variables no negativas

°x_2 variable sin restricción

$$x_2 = x_2^+ - x_2^-$$

Realizando esta sustitución, las ecuaciones o condiciones de restricción tienen la siguiente forma:

$$3x_1 + 2x_2^+ - 2x_2^- - x_3 = 6$$

$$x_1 - 6x_2^+ + 6x_2^- + x_4 = 4$$

$$x_1, x_2^+, x_2^-, x_3, x_4 \geq 0$$

y la función objetivo es:

$$\max : n = -3x_1 - 5x_2$$

También es posible resolver un problema de minimización recurriendo a su formulación dual que se estudia en la sección 6.5.5.

***Formulación matricial**

***Coeficientes estructurales**

* La estructura del problema de programación lineal se presta para el empleo de la notación matricial. Si se definen *la matriz de coeficientes estructurales

$$\underline{A} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & \cdots & & \\ \cdot & & & \\ \cdot & & & \\ \cdot & & & \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \quad (6.5.17)$$

***Actividades**

* los vectores de actividades:

$$\underline{x} = \begin{bmatrix} x_1 \\ x_n \\ \cdot \\ \cdot \\ \cdot \\ x_n \end{bmatrix} \quad (6.5.18)$$

***Costos**

de *costos

$$\underline{c} = \begin{bmatrix} c_1 \\ c_2 \\ \cdot \\ \cdot \\ \cdot \\ c_n \end{bmatrix} \quad (6.5.19)$$

***Restricciones**

y *de restricciones

$$\underline{b} = \begin{bmatrix} b_1 \\ b_2 \\ \cdot \\ \cdot \\ \cdot \\ b_n \end{bmatrix} \quad (6.5.20)$$

El problema de programación lineal queda planteado de la siguiente forma:

$$\text{máx: } m = \underline{c}^T \underline{x} \quad (6.5.21)$$

Sujeto a las restricciones

$$\underline{A}\,\underline{x} \leq b \quad (6.5.22)$$

$$\underline{x} \geq 0 \quad (6.5.23)$$

En la siguiente sección se ilustra gráficamente la forma de obtener la solución del problema de programación lineal.

6.5.3 Solución gráfica

En esta sección ilustraremos gráficamente la solución del problema de programación lineal. Como es difícil representar gráficamente funciones de más de dos variables, se empleará el ejemplo 6.5.2 para realizar esta representación.

El modelo matemático de este problema es el siguiente:

$$\max : m = 2x_1 + 4x_2 \quad (6.5.7)$$

Sujeto a las restricciones

$$x_1 + 4x_2 \leq 24 \qquad (6.5.8)$$
$$x_1 + x_2 \leq 9 \qquad (6.5.9)$$
$$3x_1 + x_2 \leq 21 \qquad (6.5\text{-}10)$$
$$x_1, x_2 \geq 0$$

Las restricciones de este problema establecen una zona del plano (x_1, x_2) donde deben encontrarse las soluciones factibles, tal como se señaló en la sección 6.1.2. Observe que la ecuación $x_1 + 4x_2 = 24$, corresponde a una recta, que divide al plano en dos regiones. En la inferior se cumple $x_1 + 4x_2 \leq 24$, por lo tanto, la solución factible debe estar "abajo" de dicha recta. La figura 6.5.5 ilustra la zona definida por esta restricción.

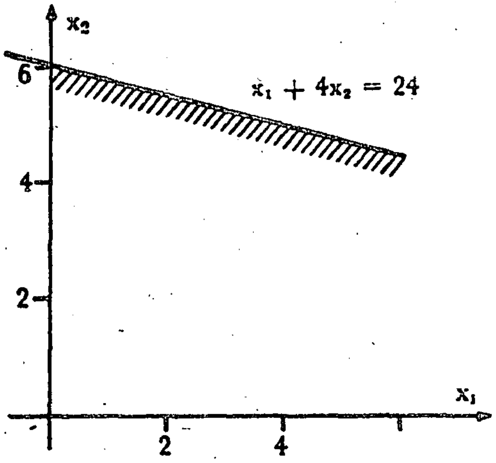


Fig. 6.5.5 Zona con restricción $x_1 + 4x_2 \leq 24$.

Un razonamiento similar lleva a concluir que la solución factible también debe estar a la "izquierda" de las rectas $x_1 + x_2 = 9$ y $3x_1 + x_2 = 21$ (fig. 6.5.6).

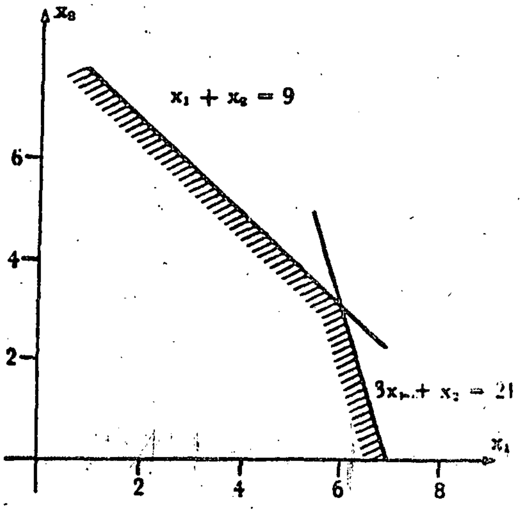


Fig. 6.5.6 Zona con restricciones $x_1 + x_2 \leq 9$ y $3x_1 + x_2 \leq 21$.

Además, la condición $x_1 \geq 0$ y $x_2 \geq 0$ impone que debe estar en el primer cuadrante. La región del plano donde se cumplen todas las restricciones es por lo tanto polígono convexo OABCDO que aparece en la figura 6.5.7.

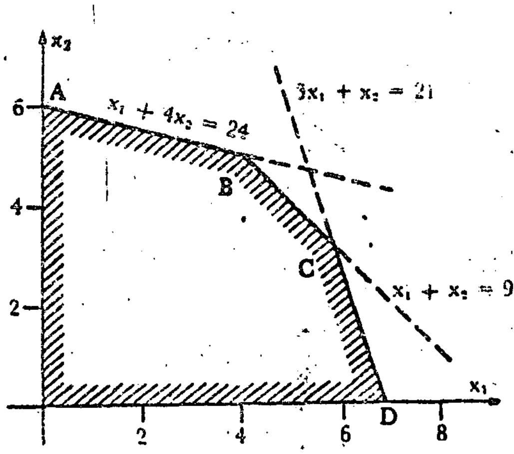


Fig. 6.5.7 Zona de soluciones factibles del ejemplo 6.5.2.

El siguiente paso en la solución consiste en encontrar dentro de los puntos de dicho polígono, que son soluciones factibles todos ellos, aquel punto para el cual la función objetivo 6.5.7 $2x_1 + 4x_2$ es máxima. Nótese primero que cualquier recta dependiente $^{-}2/4$ cumple con la condición $2x_1 + 4x_2$. Además, entre mayor sea la distancia al origen de una recta dependiente $^{-}1/2$, tanto mayor es $2x_1 + 4x_2$ tal como se ilustra en la figura 6.5.8.

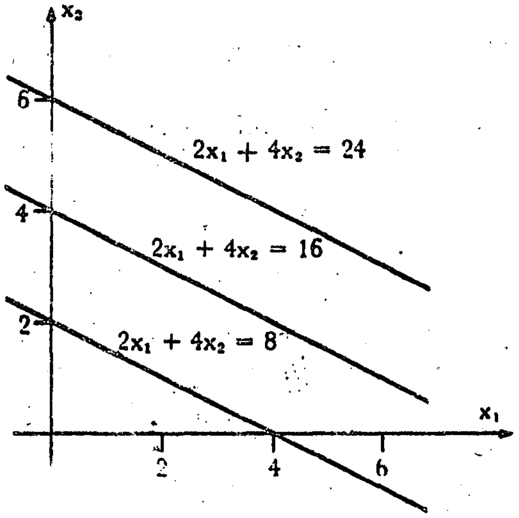


Fig. 6.5.8 Funciones objetivo del ejemplo 6.5.2.

Para obtener el valor máximo de la función objetivo $2x_1 + 4x_2$ es necesario desplazar una recta dependiente $^{-}2/4$ de manera que su distancia al origen sea máxima, pero tenga por lo menos un punto dentro de la región OABCDO. En la figura 6.5.9 se ilustra este procedimiento de búsqueda del máximo. En el punto B de coordenadas (4, 5) el valor de la función objetivo $2x_1 + 4x_2$ es de 28 y se cumplen todas las restricciones. Por lo tanto $x_1 = 4$, $x_2 = 5$ es la solución del problema de programación lineal. Haciendo referencia a la fig. 6.5.9 obsérvese además que para dicho punto, tiene las características resumidas en el cuadro de la tabla 6.5.1.

Problema

Función objetivo.

$$M = 2x_1 + 4x_2 \ (\text{max.})$$

Restricciones.

$$x_1 + 4x_2 \leq 24 \ (a)$$
$$x_1 + x_2 \leq 9 \ (b)$$
$$3x_1 + x_2 \leq 21 \ (c)$$
$$x_1 \geq 0 \ (d)$$
$$x_2 \geq 0 \ (e)$$

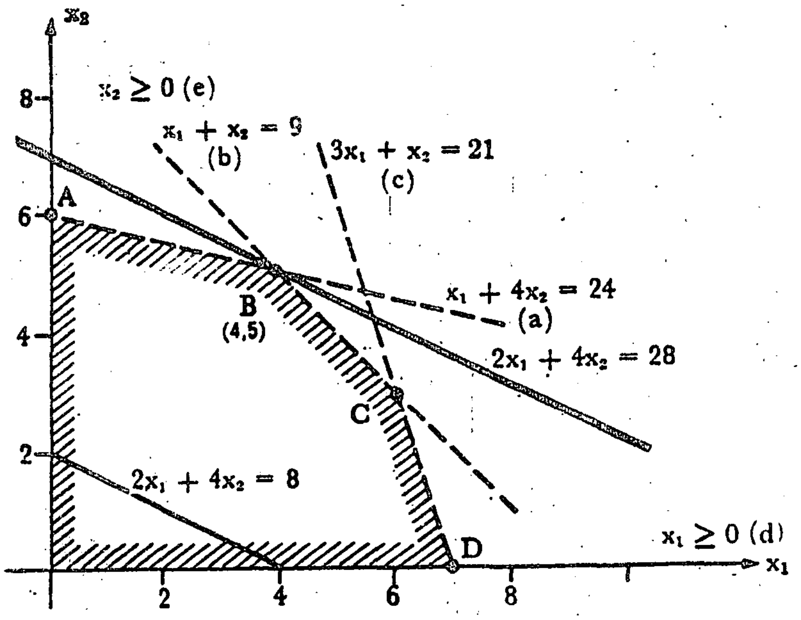


Fig. 6.5.9 Ilustración de la solución gráfica del problema de programación lineal.

Tabla 6.5.1 Propiedades de punto óptimo B del ejemplo 6.5.2.

Restricción	Holgura
$x_1 + 4x_2 = 24$	0
$x_1 + x_2 = 9$	0
$3x_1 + x_2 = 17 \leq 21$	4

Es decir, el recurso mecánico "del que se cuenta con 24 días más, el de "andenes de carga" con el que se cuenta con 9, se emplea plenamente si se usan 4 camionetas de dos toneladas y 5 de 4 toneladas. Mientras que de tercer recurso, del que se cuenta con 21 unidades, sólo se usan 17. Sin embargo, ninguna otra combinación de x_1 y x_2 permite obtener mayor volumen de carga sin violar las restricciones (6.5.8), (6.5.10). Antes de continuar, nótese que la región definida por las restricciones (6.5.8 - 6.5.10) es convexa, como muestra la figura 6.5.10, ya que cualquier recta que une dos puntos cualquiera de la periferia de la zona se encuentra en la frontera o dentro de la región.

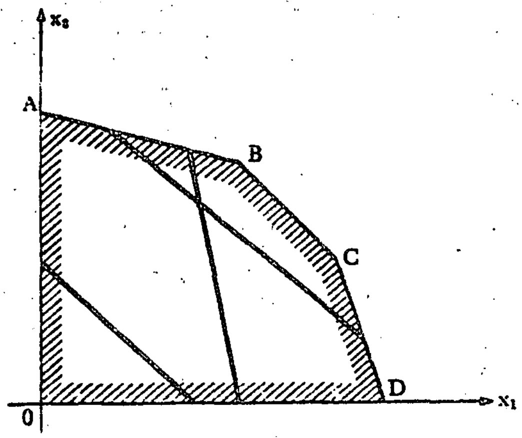


Fig. 6.5.10 Zona convexa de soluciones factibles.

En la sección 6.5.4 se empleará la representación gráfica de la solución de programación lineal para visualizar fácilmente diversos casos especiales de problemas de este tipo.

*Método gráfico para problemas con dos variables.

*El método gráfico de solución del problema de programación lineal está restringido a modelos con dos variables. Prácticamente todos los problemas de interés para el analista tienen más de dos variables, por lo cual el método gráfico no se puede emplear en estos casos. *Es necesario contar con métodos algebráicos que se puedan programar en una computadora digital, con objeto de resolver problemas con un gran número de variables, como son la mayoría de los que se encuentran en la práctica. El método simplex que se introduce en la siguiente sección tiene esta propiedad. Sin embargo, es importante familiarizarse con la solución gráfica estudiada en esta sección, ya que ayuda a entender la naturaleza de la solución del problema.

***Métodos algebraicos para resolver sistemas con muchas variables.**

Al ir desarrollando el método simplex de solución analítica, continuamente se hará referencia a la solución gráfica. Los autores consideran que de esta forma el lector lo comprenderá con mayor facilidad.

6.6 PROGRAMACION DINAMICA

6.6.1 Características

En la sección 6.1.1 se señaló que los métodos de optimización pueden clasificarse en *métodos de gradiente y métodos de búsqueda.* *En la sección 6.5 se estudió el método de programación lineal que constituye un método de gradiente. En la siguiente sección se establecen las bases de la *programación dinámica,* un método de optimización de búsqueda. Este último método, todavía más que el de programación lineal requiere del uso de la computadora digital. *Como se trata de una técnica enumerativa, los tiempos de cómputo para este método son en general grandes, así como los requerimientos de memoria. Debido a ello el empleo de esta técnica es un cuanto limitado, a pesar de su extensivo número de aplicaciones potenciales.

***Métodos de optimización de gradiente y búsqueda.**

*** La programación dinámica es un método de búsqueda.**

***Requiere mucho tiempo de cómputo y memoria.**

*La programación dinámica es una técnica de optimización enumerativa aplicable a problemas con restricciones y funciones objetivo que pueden ser *no lineales* y regiones factibles *no convexas.*

***Puede aplicarse a problemas no lineales y regiones no convexas.**

*Se aplica en forma natural a problemas que pueden descomponerse en etapas a lo largo del tiempo, pero también puede emplearse en problemas *no* secuenciales o con estructura en serie.

***El problema debe poder expresarse en forma secuencial.**

En el análisis de sistemas, la programación dinámica se usa en general en problemas de toma de decisiones, frecuentemente relacionados con la asignación de recursos.

*Para resolver este tipo de problemas, se establece un modelo matemático cuyas principales componentes son:

***Modelo matemático.**

1). Un estado inicial x que da toda la información relevante sobre el sistema antes de la toma de una decisión

Como el problema de *decisiones* se presenta en aquellas situaciones, donde un problema tiene varias soluciones factibles o alternativas, con objeto de poder seleccionar entre éstas, es necesario asociar a todas las posibles soluciones una función de beneficio o ganancia, que mida la utilidad que se asocia a cada una de las posibles soluciones.

Esta función o relación de transformación puede ser una relación matemática o puede estar dada en forma tabular.

Para representar estas componentes del modelo de toma de decisiones resulta útil introducir un diagrama de bloque (figura 6.6.1).

Como la función de transformación *T* es univaluada puede sustituirse (6.6.2) en (6.6.1) para obtener:

*Es decir, la función de beneficio r sólo depende de los estados iniciales y las variables de decisión.

Recordando que la función de transformación es univaluada puede obtenerse la transformación inversa T', a saber

Sustituyendo este valor en (6.6.1) se llega a:

o bien

Un problema de toma de decisiones consiste en maximizar o minimizar la función de beneficio r, si las variables independien-

2). Un estado final, $\underset{\sim}{\tilde{x}}$ que da toda la información relevante sobre el sistema después de haberse tomado la decisión.

3). La variable de decisión $\underline{D} = (d_1, d_2 \ldots, d_n)$ que puede manipularse para obtener determinado cambio del sistema de su estado inicial $\underline{x}$, a su estado final $\underset{\sim}{\tilde{x}}$.

4). El beneficio r que es una función escalar que depende del valor de los estados iniciales, de las decisiones tomadas, y de los estados finales, es decir

$$r = r\ (\underline{x}, \underline{D}, \underset{\sim}{\tilde{x}})$$

5). Una transformación *T*, univaluada que relaciona los estados finales, con los estados iniciales, y las variables de decisión.

$$\underline{x} = T\ (\underline{x}, \underline{D}) \qquad (6.6.2)$$

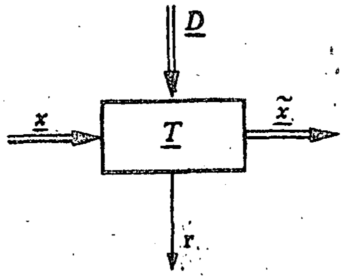


Fig. 6.6.1 Modelo de un problema de toma de decisión.

$$r = r\ (\underline{x}, \underline{D}, \underline{T}\ (\underline{x}, D))$$

***Función de beneficio**

$$r = r'\ (\underline{x}, \underline{D}) \qquad (6.6.3)$$

$$\underline{x} = T'\ (\underline{x}, \underline{D})$$

$$r = r\ (T^{o}\ (\underline{\tilde{x}}, \underline{D}),\ D\ \tilde{x})$$

$$r = r''\ (\tilde{x}, \underline{D}) \qquad (6.6.4)$$

***Maximizar o minimizar el beneficio.**

tes o de decisión toman todos los posibles valores, dentro de las restricciones que fija el problema.

Estos problemas de toma de decisiones son, por lo tanto, problemas de optimización entre los que podemos distinguir dos tipos:

El problema de optimización de estado inicial x consiste en encontrar el máximo (o mínimo) del beneficio como función del *estado inicial,* es decir:

*Optimización de estado inicial $\underline{x}$

$$f(\underline{x}) = \underset{\underline{D}}{\text{máx}}\ r'(\underline{x}, \underline{D}) \qquad (6.6.5)$$

(6.6.5)

En el problema de *estado final x,* debe determinarse el máximo (o mínimo) del beneficio como función del estado final, es decir:

*Optimización de estado final $\underline{\tilde{x}}$

$$f(\underline{\tilde{x}}) = \underset{\underline{D}}{\text{máx}}\ r''(\underline{\tilde{x}}, \underline{D}) \qquad (6.6.6)$$

Con objeto de facilitar la presentación del material subsecuente e ilustrar la naturaleza de estos problemas, conviene introducir algunos símbolos:

*Símbolos empleados en programación dinámica.

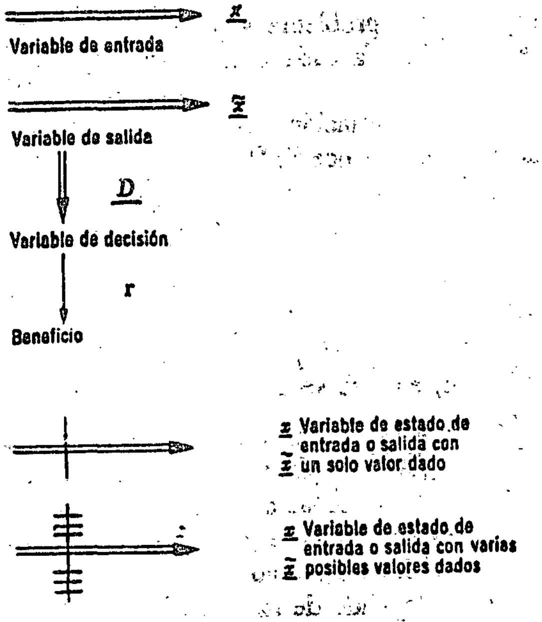
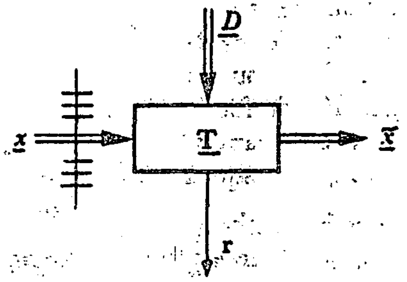


Fig. 6.6.2 Símbolos en problemas de programación dinámica.

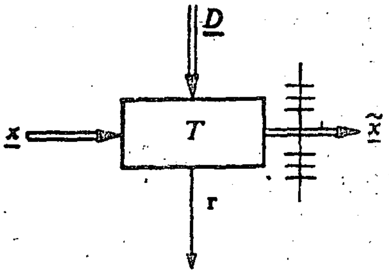
Usando estos símbolos el problema de valor inicial puede simbolizarse:



$$f(\underline{x}) = \underset{\underline{D}}{\text{máx}}\ r\ (\underline{x}, \underline{D})$$

Fig. 6.6.3 Problema de valor inicial.

y el de valor final



$$f(\tilde{x}) = \max_{D} r\,(\tilde{x}, D)$$

Fig. 6.6.4 Problema de valor final.

*Problemas de optimización como los planteados en las figuras (6.6.3) y (6.6.4) contienen muchas variables. La programación dinámica transforma un problema de esta naturaleza en una serie de problemas más sencillos, que contienen pocas variables.

***Programación dinámica:**

Un problema con muchas variables. $\Longrightarrow$

Muchos problemas de pocas variables.

Esta transformación es invariante en el número de soluciones factibles del problema y se conserva el valor de la función beneficio asociada a cada una de las posibles soluciones.

*La programación dinámica se basa en el principio de optimalidad expuesto por R.D. Bellman: (ref. 2).

***Principio de optimalidad de Bellman.**

"Una serie de decisiones óptimas (políticas óptimas) tiene la propiedad, de que cualquiera que sea el estado inicial y la decisión inicial, las decisiones restantes deben ser óptimas con respecto al estado que resulte de la primera decisión".

Un ejemplo adaptado de la ref. 8 servirá para aclarar este concepto, en que se basa la programación dinámica.

*Supóngase que se desea asignar recursos a tres proyectos industriales, A, B y C con el objeto de maximizar las ganancias, *sean R_A, B_B y R_C las cantidades que se asignan a los proyectos A, B y C respectivamente y sean *R_T los recursos totales disponibles que son limitados. Debido a ello, la cantidad que se asigna a cada proyecto, depende de la cantidad asignada a los dos restantes. La asignación a C no debe exceder $R_T - R_A - R_B$ *Sin embargo, cualquiera que haya sido la asignación a los proyectos A y B, la asignación R_C al proyecto C, debe ser óptima con respecto a todas las posibles cantidades residuales que pueden quedar para el proyecto C, después de asignar fondos a los proyectos A y B. *La asignación de fondos a los proyectos B y C debe ser óptima con respecto a la cantidad residual que queda después de asignar recursos a A, cualesquiera que haya sido ésta asignación.

*** Proyectos industriales A, B, C.**

$^*R_{A,\,B,\,C}$ recursos para cada proyecto
R_T recursos totales disponibles.
$^*R_A + R_B + R_C \leq R_T$

***La asignación a C debe ser óptima con respecto a $R_T - R_A - R_B$.**

***La asignación a B y C debe ser óptima con respecto a $R_T - R_A$.**

La asignación óptima al proyecto B, se encuentra maximizando el beneficio, que ocurre de la asignación al proyecto B, junto con el beneficio óptimo del proyecto C, como función de los fondos que quedan de asignar recursos a B y A. La asignación óptima a A finalmente se encuentra para maximizar el beneficio de A más el beneficio óptimo de B y C, como función de los fondos que quedan después de asignar recursos a A.

Obsérvese que se ha descompuesto el problema, en una secuencia de toma de decisiones, asignando recursos a un solo proyecto a la vez.

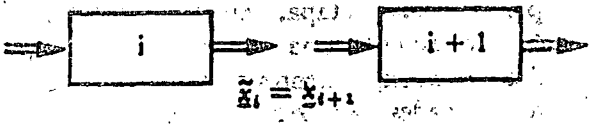
*Se asignan recursos a un proyecto a la vez.

En realidad la asignación de recursos es simultánea, pero la descomposición del problema, en una asignación secuencial o en serie de los recursos, permite tomar decisiones una a la vez.

El concepto de sistema secuencial o en serie es muy importante en este tipo de problemas y se discute con mayor detalle en la siguiente sección.

6.6.2 Estructuras serie

*En una estructura en serie, como se señaló en la sección 1.3.4, la salida de un elemento está conectada a la entrada del siguiente, sin haber realimentación, ésta, como se indicó en la sección 1.3.5, implica que la salida de un sistema influye sobre su entrada. La presencia de realimentación en un problema de programación dinámica puede resolverse sustituyendo la porción del sistema con realimentación por un subsistema equivalente no realimentado. Los ingenieros llaman a esta operación: sustituir el sistema realimentado por su función de transferencia.**



Conexión serie

*En un problema con estructura serie en el tiempo, que son los más frecuentes en el análisis de sistemas, las decisiones que se toman en un determinado instante de tiempo, no alteran los eventos anteriores, sólo tienen influencia sobre los eventos posteriores.

*En una estructura serie las decisiones no afectan eventos anteriores.

En la construcción de una casa, el levantamiento de muros, es posterior a la construcción de los cimientos pero anterior a la colocación de ventanas y puertas. Si durante la construcción de los muros, se cambia la posición y tamaño de los huecos para las puertas y las ventanas, este cambio, resultado de una decisión, no afecta a la etapa anterior, o sea la construcción de los cimientos, pero sí influye sobre la etapa posterior, la de colocación de puertas y ventanas.

**Gerez Greiser V. y Murray-Lasso, M. A. Teoría de Sistemas y Circuitos I, Cap. 8.
Servicios y Representaciones de Ingeniería, S. A.
México, D. F. 1972.

Esquemáticamente un problema con estructura en serie, puede representarse usando los diagramas de bloque de la sección 1.3.4, de la forma mostrada en la figura 6.6.5.

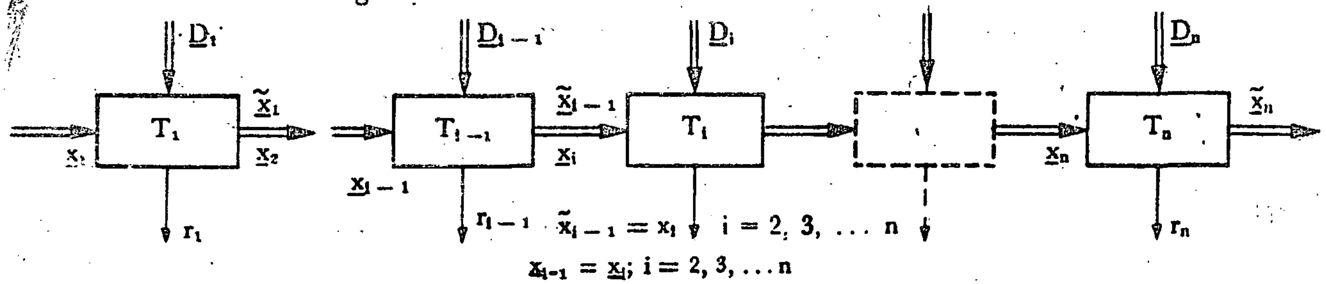


Fig. 6.6.5 Estructura en serie.

A continuación se hace una presentación formal del principio de optimalidad y se deduce la fórmula recursiva para resolver este tipo de problemas.

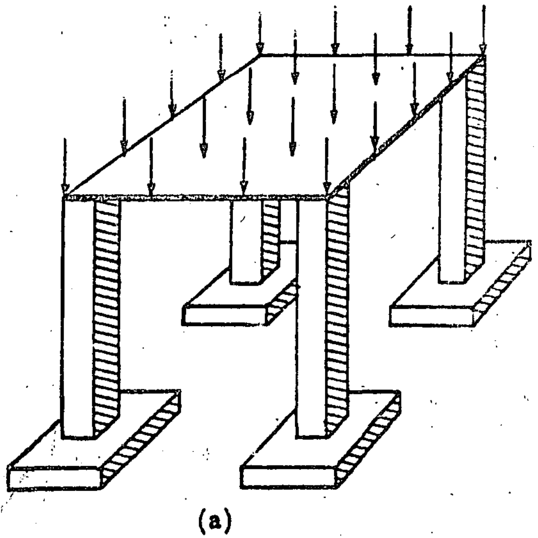
6.6.3 Principio de optimalidad

*Se señaló en la sección anterior que el objetivo de la descomposición del problema de optimización en una serie de problemas secuenciales, es reducir el número de variables que se manipulan en cada etapa, trabajando, de preferencia, con una variable de estado y una variable de decisión. Por esta razón en los desarrollos subsecuentes se emplean los símbolos que corresponden a cantidades escalares, como por ejemplo x, y no los correspondientes a vectores como x, tampoco se seguirá empleando el trazo doble para representar las variables en los diagramas de bloque.

*Trabajar de preferencia con una variable de estado y una de decisión.

A continuación aplicaremos el principio de optimalidad a un problema de valor inicial adaptado de la ref. (1).

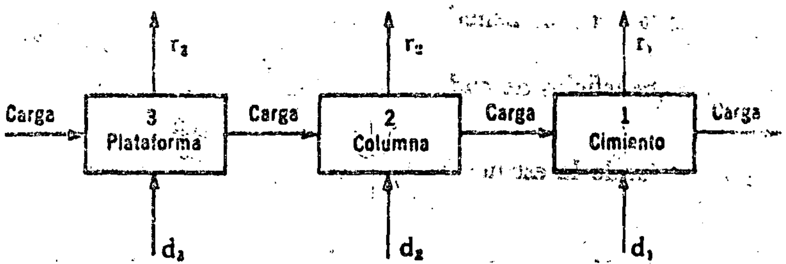
La figura 6.6.6a muestra una plataforma que debe soportar una carga dada de ωkg/m². El objetivo del problema es diseñar una plataforma, las columnas de soporte y los cimientos necesarios para soportar el peso minimizando el costo de la obra. Para aplicar la técnica de la programación dinámica a este problema, conviene descomponerlo en una serie de problemas más fáciles de optimizar.



(a)

Plataforma para soportar ω Kg/m²

La solución de este problema puede esquematizarse como muestra la fig. 6.6.6 b



(b)

Estructura secuencial para la solución del problema de diseño de una plataforma de carga

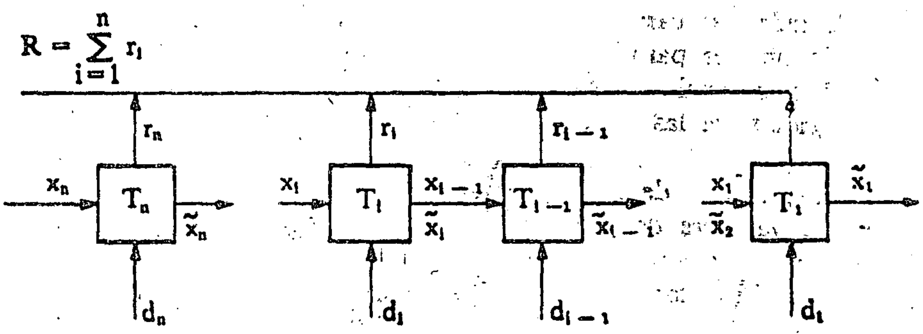
Fig. 6.6.6 Ejemplo de aplicación del método de programación dinámica.

Supóngase que se empieza analizando las columnas; si se encuentra que la solución más económica son las columnas de concreto, esta solución implica mayor peso sobre los cimientos que el producido por las columnas de hierro. Esta solución afecta el beneficio (costo) de todas las etapas subsecuentes (En este caso los cimientos). Por lo tanto no puede empezarse analizando las columnas.

*Resulta evidente que la estrategia adecuada de solución consiste en empezar analizando aquella parte del proyecto, que no influye sobre los restantes, en este caso los cimientos. Al igual que en la asignación de recursos a tres proyectos industriales en la sección 6.6.1, posteriormente pueden agruparse las dos últimas etapas, columnas y cimientos, para suboptimizarse posteriormente, sin afectar a ninguna otra etapa.

***Empiece por aquellas partes que no afectan otras etapas.**

Como se ve, el proceso de optimización se realiza en orden inverso, primero se estudian los cimientos, después los cimientos en combinación con las columnas y finalmente todo el proyecto. Conviene por lo tanto numerar los pasos de solución en este orden, tal como aparece en la fig. 6.6.6 o en general como se muestra en la figura 6.6.7.



$$x_i = x_{i-1} \quad i = 2, 3, \ldots n$$

Fig. 6.6.7 Estructura secuencial de n pasos.

*Recuérdese que el beneficio en un problema de valor inicial puede expresarse como función del estado inicial x_i y de la variable de decisión d_i (ecs. 6.6.4)

***Beneficio de la etapa i'sima:**

$$r_i = r_i\ (x_i, d_i) \qquad (6.6.7)$$

*Si la función beneficio R para todo el problema, es la suma de los beneficios de cada una de las etapas, se tiene:

***Para beneficios aditivos:**

$$R = \sum_{i=1}^{n} r_i\ (\underline{x}_i, d_i) \qquad (6.6.8)$$

*recordando la estructura serie del problema que implica

***Como la estructura es serie:**

$$x_i = x_{i-1} \quad i = 2,3,\ldots,n \qquad (6.6.9)$$

*y la relación entre la variable de entrada x_i, la de salida x_i y la de decisión d_i

***relación entrada — salida**

$$x_i = T_i\ (x_i, d_i) \qquad (6.6.2)$$

*se obtiene para la primera etapa de la serie

***1ra. etapa**

$$r_1 = r_1\ (x_1, d_1)$$

Por ser la entrada al primero x_1, igual a la salida del segundo $\bar{x}_2$, se tiene:

$$x_1 = \bar{x}_2$$

$$r_1 = r_1\ (\bar{x}_2, d_1)$$

pero

$$\bar{x}_2 = T_2\ (x_2, d_2)$$

sustituyendo esta relación en la anterior

$$r_1 = r_1\ (T_2\ (x_2, d_2, d_1) \qquad (6.6.10)$$

y como

$$x_3 = \bar{x}_3$$

y

$$\bar{x}_3 = T_3\ (x_3, d_3)$$

se tiene al sustituir en (6.6.10)

$$r_1 = r_1\ (T_2\ (T_3\ (x_3, d_3), d_2), d_1)$$

Siguiendo con esta sustitución se obtiene:

$$r_1 = r_1\ [T_2\ (T_3\ [T_4 \ldots \{T_n\ (x_n, d_n), d_{n-1}\}, \ldots]\ d_2)\ d_1]\ 6.6.11)$$

*Obsérvese que esta relación indica que el beneficio r_1 asociado a la etapa 1 es función solamente de la variable de estado inicial y de todas las variables de decisión. Una conclusión idéntica se puede obtener para todas las etapas subsecuentes, por lo tanto el beneficio total del proyecto es función exclusiva del estado inicial y de todas las variables de decisión, es decir,

***El beneficio total depende del estado inicial y de las variables de decisión.**

$$R = R(x_n, d_1, d_2, \ldots, d_n) \qquad (6.6.12)$$

*El problema de optimización consiste en encontrar los valores de las variables de decisión d_1, d_2, ... d_n que para un valor dado x_n del estado inicial maximicen o minimicen la función de beneficio R de todo el proyecto.

***Encuentre $d_1, d_2, \ldots, d_n$ que optimice el beneficio total R, dado el estado inicial x_n.**

Analícese ahora el problema empezando con la 1ra. etapa.

Para ésta etapa, sea f_1 el máximo (o mínimo) de la función beneficio.

*Para cada valor posible de x_1, la función beneficio tiene un valor óptimo, que se encuentra optimizando esta función con relación a la variable de decisión d_1, es decir

***Beneficio óptimo $f_1(x_1)$ para cada valor de x_1**

$$f_1(x_1) = \max_{d_1} r_1(x_1, d_1) \qquad (6.6.13)$$

*Si se considera a continuación la segunda etapa su beneficio será:

***Para la 2da. etapa.**

$$r_1 (x_1, d_1) + r_2 (x_2, d_2)$$

*y el óptimo será:

***Valor óptimo**

$$\max_{d_1, d_2} \{(r_1, d_1) + r_2 (\tilde{x}_2, d_2)\}$$

*El beneficio óptimo de la primera etapa ya ha sido calculado en (6.6.13) y por lo tanto se tiene como beneficio óptimo de la primera y segunda etapas combinadas, por el principio de optimalidad.

***Beneficio para la 1ra. y 2da. etapas.**

$$\max_{d_2} \{r_2 (d_2, x_2) + f_1(x_1)\} \qquad (6.6.14)$$

*Nótese que en esta segunda etapa ya solamente es necesario buscar el óptimo con respecto a d_2.

***Sólo se busca el óptimo respecto a d_2.**

*Por la conexión serie entre etapas se tiene

***Conexión serie**

$$x_1 = \tilde{x}_2$$

y por la transformación que ejerce la segunda etapa

$$\tilde{x}_2 = T_2(x_2, d_2)$$

Sustituyendo en (6.6.14)

$$\max_{d_2} \{r_2 (d_2, x_2) + f_1(T_2(x_2, d_2)\}$$

*El beneficio óptimo de la primera y segunda etapas combinadas es por tanto:

***Beneficio óptimo de la 1ra. y 2da. etapas.**

$$f_2(x_2) = \max_{d_2} \{r_2(x_2, d_2) + f_1 (T_2(x_2, d_2))\}$$

*Procediendo con este razonamiento se llega a la n'sima y última etapa y se obtiene una relación similar para el beneficio óptimo.

***Para la última etapa.**

$$f_n(x_n) = \max_{d_n} \{r_n(x_n, d_n) + f_{n-1} (T_n(x_n) d_n))\} \qquad (6.6.15)$$

***Fórmula de recursión.**

*Toda esta deducción puede por lo tanto resumirse en las siguientes ecuaciones de recursión para el problema de programación dinámica:

$$f_i(x_i) = \underset{d_i}{\text{máx}}\ Q_i(x_i, d_i) \qquad i = 1, 2, \ldots n$$

$$Q_i(x_i, d_i) = r_i(x_i, d_i) \qquad i = 1 \qquad (6.6.16)$$

$$Q_i(x_i, d_i) = r_i\ (x_i, d_i) + f_{i-1}\ (T_i(x_i, d_i)$$

$$i = 2, 3, \ldots n$$

El problema siguiente ilustra el empleo de la programación dinámica.

Ejemplo 6.6.1.

*Supóngase que se desea maximizar el beneficio que se obtiene de un programa de desarrollo industrial.

*El proyecto prevé la instalación de un máximo de tres industrias diferentes. El beneficio que se obtiene de cada industria depende del nivel de inversión en las mismas. *Sea x_i el nivel de inversión en la i'sima industria, y $g_i(x_i)$ el beneficio que se obtiene de la misma, si el nivel de inversión en ella es de x_i. Además se cuenta con un capital máximo de 3 billones de pesos para el Programa. Debido a la naturaleza de cada proyecto de inversión, los niveles de inversión sólo pueden ser múltiplos enteros de 1 billón de pesos. La figura 6.6.8 y la tabla 6.6.1 muestran el beneficio que se obtiene de cada proyecto de acuerdo con el nivel de inversión.

*Maximización del beneficio.

*Tres unidades industriales.

*x_i nivel de inversión en industria i'sima y $g_i(x_i)$ su beneficio.

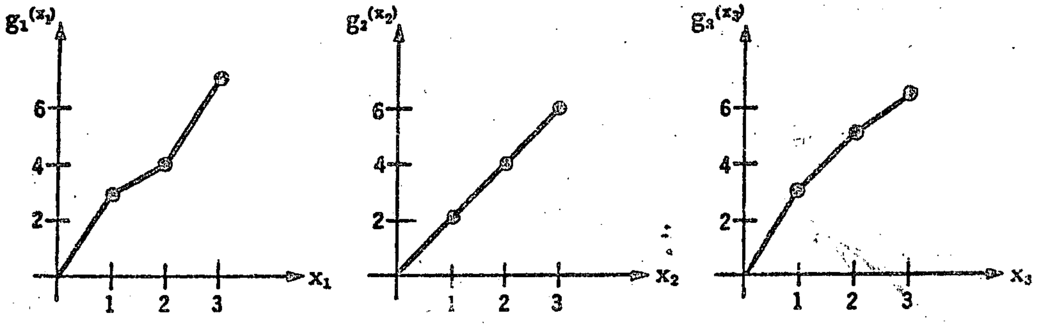


Fig. 6.6.8 Funciones de beneficio del ejemplo 6.6.1.

Tabla 6.6.1 Beneficio de los proyectos del ejemplo 6.6.1.

Función de beneficio	Industria i		
	1	2	3
g_i (0)	0	0	0
g_i (1)	3	2	3
g_i (2)	4	4	5
g_i (3)	7	6	6

Solución.

*Debido a la naturaleza del proyecto, la función objetivo o beneficio total que se obtiene de este proyecto es de carácter aditivo, es decir:

°Función de beneficio total aditiva.

$$R = \sum_{i=1}^{3} g_i(di)$$

*Además, se tiene la restricción en los fondos de:

°Restricción de fondos.

$$3 \geq x_1 + x_2 + x_3$$

*Como el orden de asignación de recursos en este caso es irrelevante puede establecerse cualquier secuencia en la serie. Si empleamos la del enunciado se tiene el diagrama de bloque de la figura 6.6.9

°La secuencia de asignación es irrelevante.

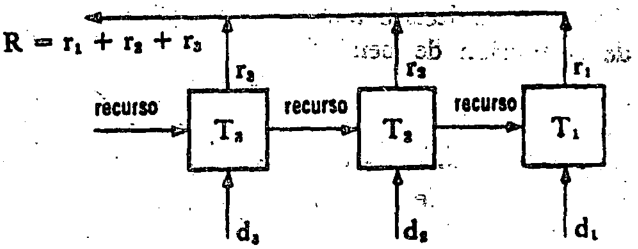


Fig. 6.6.9 Diagrama de bloque del ejemplo 6.6.1.

Como variable de entrada a cada proyecto puede considerarse el recurso que queda por asignársele, después de asignados recursos a los anteriores, y como salida lo que queda por asignar, una vez asignados fondos al mismo. La entrada, al tercero es fijo e igual a 3. Si se toma la decisión de asignar dos billones de pesos a este proyecto, es decir, $d_3 = 2$, la salida del tercer bloque x_3 será 1, y el beneficio r_3 de acuerdo con la tabla 6.6.1 serie de 4 tal como lo ilustra la figura 6.6.10

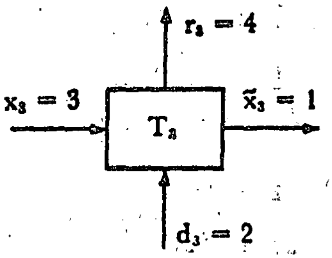
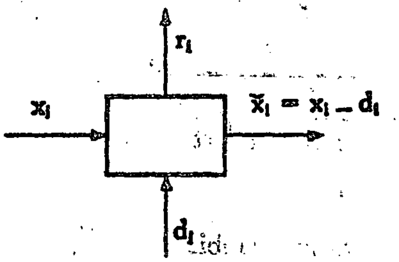


Fig. 6.6.10 Ejemplo de asignación de recursos al proyecto 3.



En este ejemplo, la transformación tiene esta forma simple $\tilde{x}_i = x_i - d_i$ y las asignaciones de recursos están sometidas a la limitación

$$+ d_1 + d_2 + d_3 \leq 3$$

Como la variable que entra a cada bloque es el recurso disponible, se debe tener además que

$$x_i - d_i \geq 0 \text{ ó } x_i \geq d_i$$

es decir, no se puede gastar en un proyecto más de los recursos disponibles.

*La función de beneficio $r_i(x_i, d_i)$ en este caso solamente depende de la decisión que se tome, es decir:

*Función de beneficio

$$r_i(x_i, d_i) = g_i(d_i)$$

*La fórmula de recursión para la solución del problema es

*Fórmula de recursión

$$Q_i(x_i, d_i) = r_i(x_i, d_i) + f_{i-1}\,(T_i(x_i, d_i)) \qquad (6.6.17)$$

En este caso la transformación es:

$$T_i(x_i, d_i) = x_i - d_i$$

Sustituyendo en la relación (6.6.17) se obtiene

$$Q_i(x_i, d_i) = g_i\,(d_i) + f_{i-1}\,(x_i - d_i) \qquad (6.6.18)$$

*Recordando que para i = 1 la función óptima de beneficio es:

*Para el 1er. proyecto.

$$f_1(x_1) = \max_{d_1} g_1(d_1)$$

Con la importante restricción señalada de que

$$x_1 \geq d_1$$

Puede establecerse por lo tanto la tabla 6.6.2 para el cálculo de la función de beneficio óptima del 1er. proyecto.

Tabla 6.6.2 Asignación de recursos a la etapa 1.

Valor de x_1	Posibles valores de d_1 $d_1 \leq x_1$	Beneficio $g_1(d_1)$	Beneficio óptimo $f_1(x_1)$	Valor de d_1* que produce el óp.
0	0	0	0	0
1	0 1	0 3	3	1
2	0 1 2	0 3 4	4	2
3	0 1 2 3	0 3 4 7	7	3

*Para la segunda etapa la fórmula de recursión establece:

*para la 2da. etapa

$$f_2(x_2) = \max_{d_2} \{g_2(d_2) + f_1(x_2 - d_2)\}$$

Este máximo también tiene que encontrarse para todos los valores posibles de x_2. La tabla 6.6.3 ilustra cómo se obtiene esta serie de máximos para los diversos valores de x_2. *Nótese además que tanto en la tabla anterior como en ésta, se anotan los valores de las variables de decisión que llevan al beneficio óptimo.

*Anote el valor de las variables de decisión "óptimas".

Finalmente para la etapa 3 se tiene $$f_3(x_3) = \underset{d_3}{\text{máx}} \{g_3(d_3) + f_2(x_3) - d_3\}$$

En la tabla 6.6.4 se resumen los valores de esta etapa.

Tabla 6 6.3 Asignación de recursos a la etapa 2.

Valor de x_2	Posibles Vals. de d_2 $d_2 \leq x_2$	Beneficio de la etapa $g_2(d_2)$	Diferencia $x_2 - d_2$	Beneficio ópt. de la etps. ants. $f_1(x_2 - d_2)$ (Tabla 6.6 2)	Valor d_1^* que prod. $f_1(x_2-d_2)$	Beneficio acumulado $Q_2(x_2, d_2)$	Beneficio óptimo $f_2(x_2)$	Val. de var. de decs. que prod. el ópt.	
								d_1^*	d_2^*
0	0	0	0	0	0	0	0	0	0
1	0	0	1	3	1	3	3	1	0
	1	2	0	0	0	2			
2	0	0	2	4	2	4	5	1	1
	1	2	1	3	1	5			
	2	4	0	0	0	4			
3	0	0	3	7	3	7	7		
	1	2	2	4	2	6		3	0
	2	4	1	3	1	7		1	2
	3	6	0	0	0	6			

Tabla 6.6.4 Asignación de recursos a la etapa 3.

Valor de x_3	Posibles: valores de d_3 $d_3 \leq x_3$	Beneficio de la etapa $g_3(d_3)$	Diferencia $x_3 - d_3$	Beneficio opt. de las ets. ants. $f_2(x_3-d_3)$ (Tabla 6.6.3)	Valores d_1^* y d_2^* que prod. $f_2(x_3-d_3)$		Beneficio acumulado $Q_3(x_3, d_3)$	Beneficio óptimo $f_3(x_3)$	Valores variables d_1^*, d_2^* y d_3^* que prod. el beneficio óptimo		
					d_1^*	d_2^*			d_1^*	d_2^*	d_3^*
0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	1	3	1	0	3	3	1	0	0
	1	3	0	0	0	0	3		0	0	1
2	0	0	2	5	1	1	5	6	1	0	1
	1	3	1	3	1	0	6				
	2	5	0	0	0	0	5				
3	0	0	3	7	1 3	0 2	7	8			
	1	3	2	5	1	1	8		1	1	1
	2	5	1	3	1	0	8		1	0	2
	3	6	0	0	0	0	6				

*Esta última tabla 6.6.4 permite concluir que el beneficio óptimo que se obtiene dentro de los límites de los recursos disponibles $x_3 \leq 3$ es de 8. El beneficio de 8 se obtiene asignando recursos de las dos maneras que muestra la figura 6.6.11.

*Beneficio óptimo.

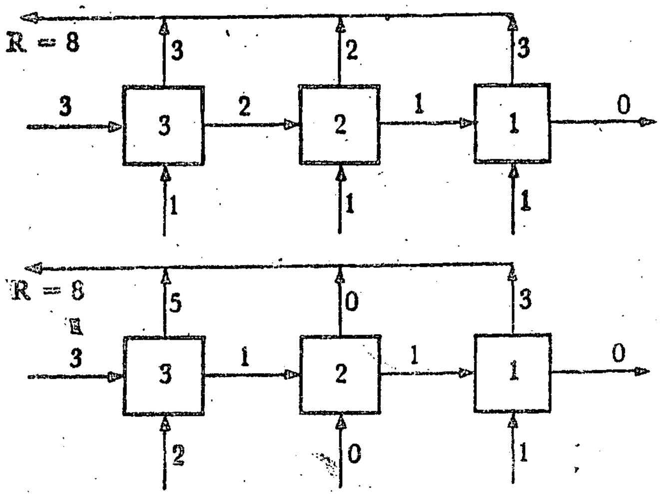


Fig. 6.6.11 Asignación óptima de recursos al proyecto del ejemplo 6.6.1

Obsérvese que en este caso existen dos estrategias de asignación de recursos que llevan al mismo beneficio de 8, dentro de la limitación $x_3 \leq 3$ ó $d_1 + d_2 + d_3 \leq 3$. La tabla 6.6.5 resume los resultados de este problema.

Tabla 6.6.5 Estrategias óptimas de inversión en el proyecto del ejemplo 6:6.1.

Proyecto	Asignación de recursos	Beneficio	
1	1	3	
	1		3
2	1	2	
	0		0
3	1	3	
	2		5
	Beneficio total		

*Para aclarar la razón por la cual la programación dinámica es una técnica enumerativa y por la cual el principio de optimalidad reduce el número de alternativas entre las que hay que buscar el máximo, se procede a continuación a ilustrar la solución de este problema empleando árboles de decisiones, como los empleados en la sección 1.3.9.

*El principio de optimalidad reduce el número de alternativas a explorar.

Empezando asignando recursos al proyecto 1, se tienen las alternativas mostradas en la figura 6.6.12. La cantidad dentro de los nodos indica el beneficio que se ha obtenido siguiendo las asignaciones de recursos asociadas a los segmentos de recta del nodo en cuestión hasta el origen del diagrama. El símbolo $g_i(d_i)$ representa el beneficio que se obtiene al asignar d_i recursos al proyecto i

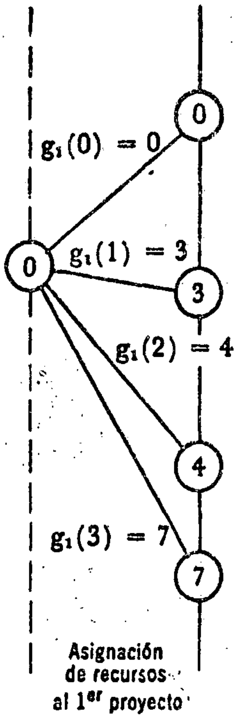


Fig. 6.6.12 Arbol de combinaciones para la asignación de 3 unidades al 1er. proyecto del ejemplo 6.6.1.

La asignación de recursos al segundo proyecto, depende de la que ya se asignó al primero. Si por ejemplo al 1er. proyecto se le asigna 1 unidad y se obtiene un beneficio de 3, al segundo proyecto solamente pueden asignársele 0, 1 ó 2 unidades sin excederse de los recursos totales de 3. Los beneficios totales que se obtienen después de estas posibles asignaciones al segundo proyecto aparecen en la figura 6.6.13

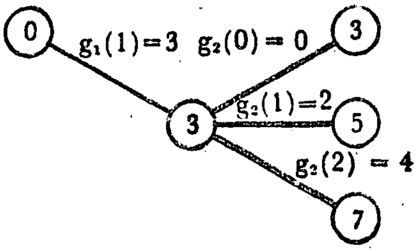


Fig. 6.6.13 Arbol con algunas posibles asignaciones de recursos al 2do. proyecto.

Siguiendo con el método expuesto, se puede construir el árbol de asignación de recursos para todo el proyecto. Este árbol se muestra en la figura 6.6.14.

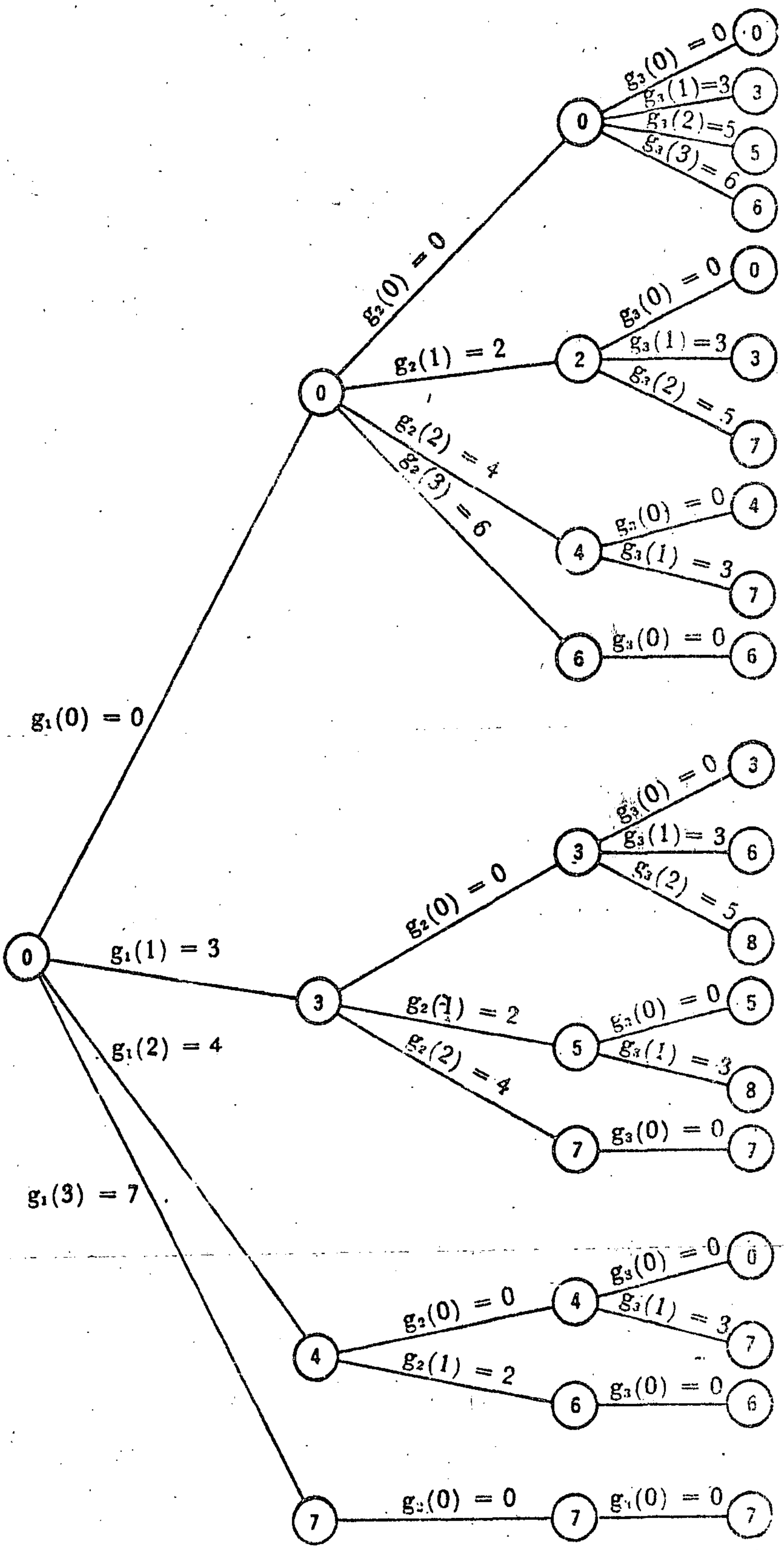


Fig. 6.6.14 Arbol de todas las posibles combinaciones de 3 unidades de recursos a 3 proyectos.

Este árbol muestra de inmediato las dos estrategias óptimas que aparecen en la figura 6.6.15

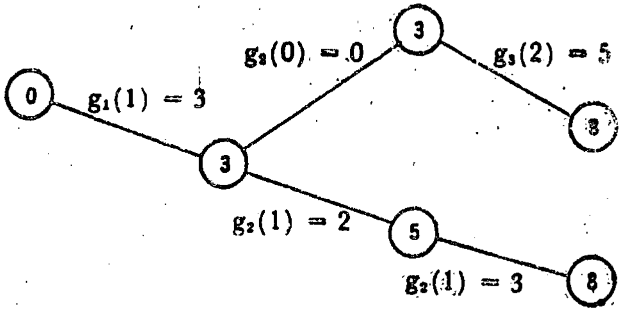


Fig. 6.6.15 Asignación óptima de recursos al proyecto del ejemplo 6.6.1.

El árbol de decisiones de la figura 6.6.14 enumera todas las posibles alternativas del proyecto, y constituye un método de fuerza bruta. *A continuación se señala cómo la programación dinámica refina este método reduciendo el número de alternativas entre las que se tiene que buscar el máximo.

***La programación dinámica reduce las alternativas entre las que se busca el óptimo.**

*Recuérdese que el prrceso empieza en la primera etapa señalando que la función de beneficio es:

***Función de beneficio para la 1ra. etapa:**

$$f_1(x_1) = \max_{d_1} g_1(d_1)$$

°y. para la segunda etapa se tiene:

°Para la 2da. etapa

$$f_2(x_2) = \max_{d_2} \{(g_2\ (d_2) + f_1(x_2 - d_2)\}$$

Esta fórmula indica que no es necesario buscar el óptimo beneficio que se obtiene al asignar recursos a los proyectos 1 y 2 buscando entre todos los posibles valores de los beneficios de las etapas 1 y 2, sino solamente entre las posibles combinaciones de beneficios de dos con beneficios óptimos de la primera etapa.

Finalmente para la última etapa se tiene:

$$f_3(x_3) = \max_{d_3} \{(g_3\ (d_3) + f_2(x_3 - d_3)\}$$

°Igualmente el beneficio óptimo no se busca entre las posibles combinaciones de beneficios de la primera, segunda y tercera etapas, sino simplemente entre las combinaciones de beneficios de la última etapa y del óptimo de las dos anteriores. Esta estrategia de búsqueda, resultado del principio de optimalidad, reduce el número de alternativas entre las que hay que buscar el óptimo. Las figuras 6.6.16 *a, b, c,* ilustran cómo se eliminan alternativas de acuerdo con la descripción anterior.

°Se busca entre los beneficios de una etapa y el óptimo de la combinación de las anteriores.

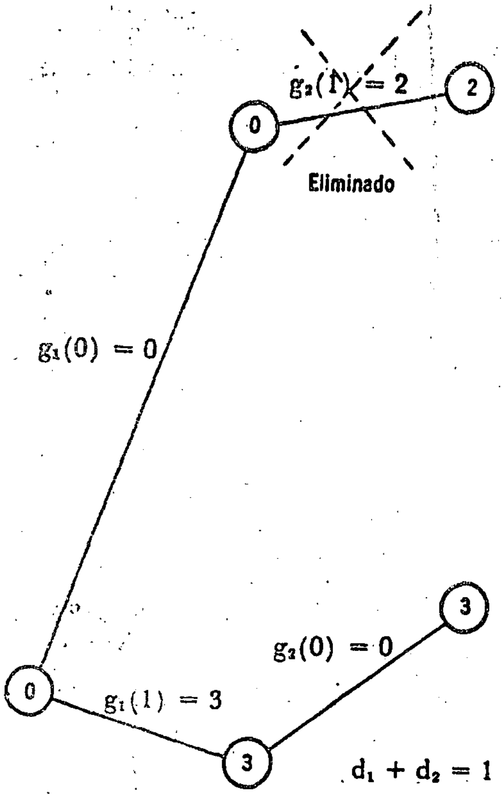


Fig. 6.6.16 *a* Asignación de una unidad de recurso en 2 etapas.

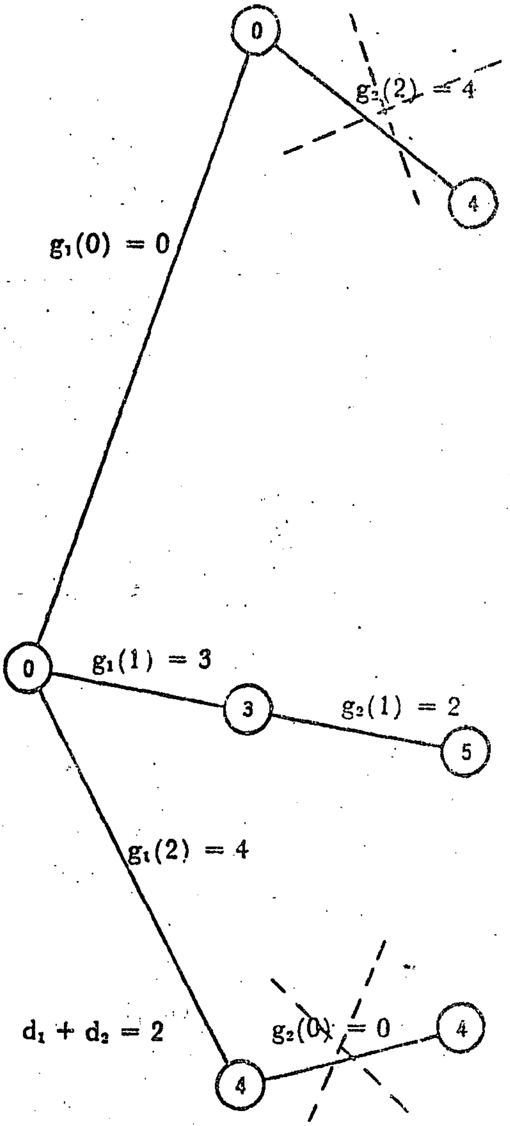


Fig. 6.6.16 *b* Asignación de 2 unidades de recurso en dos etapas.

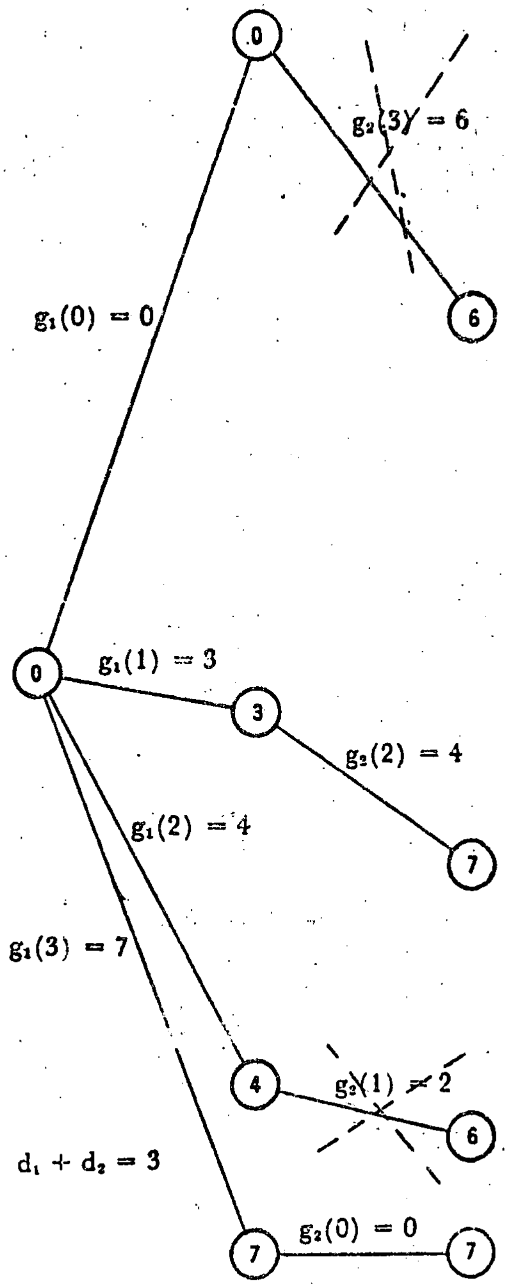


Fig. 6.6.16 *c* Asignación de 3 unidades de recurso en 3 etapas.

La eliminación de estas alternativas reduce la búsqueda a los casos que muestra el árbol de la figura 6.6.17 con trazo grueso.

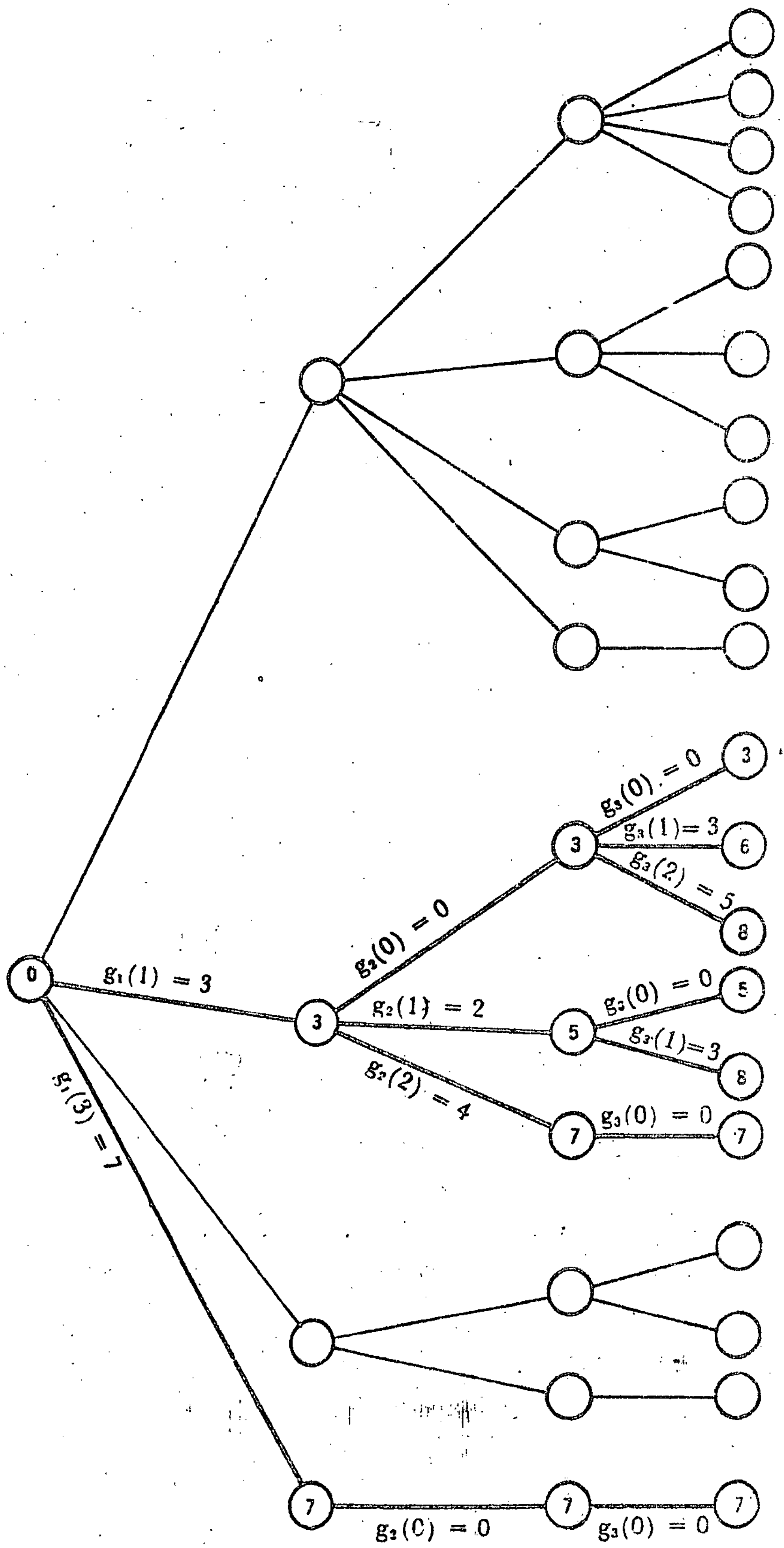


Fig. 6.6.17 Reducción de alternativas a explorar.

*La figura 6.6.14 muestra que este problema tiene 20 posibles alternativas. Si se emplea una búsqueda directa es necesario buscar entre estas posibles alternativas, para las cuales debe conocerse la combinación de decisiones que llevan a cada una de ellas, como ilustra la figura 6.6.18 para una de ellas.

***Búsqueda directa:**

20 alternativas

Programación dinámica:
8 alternativas

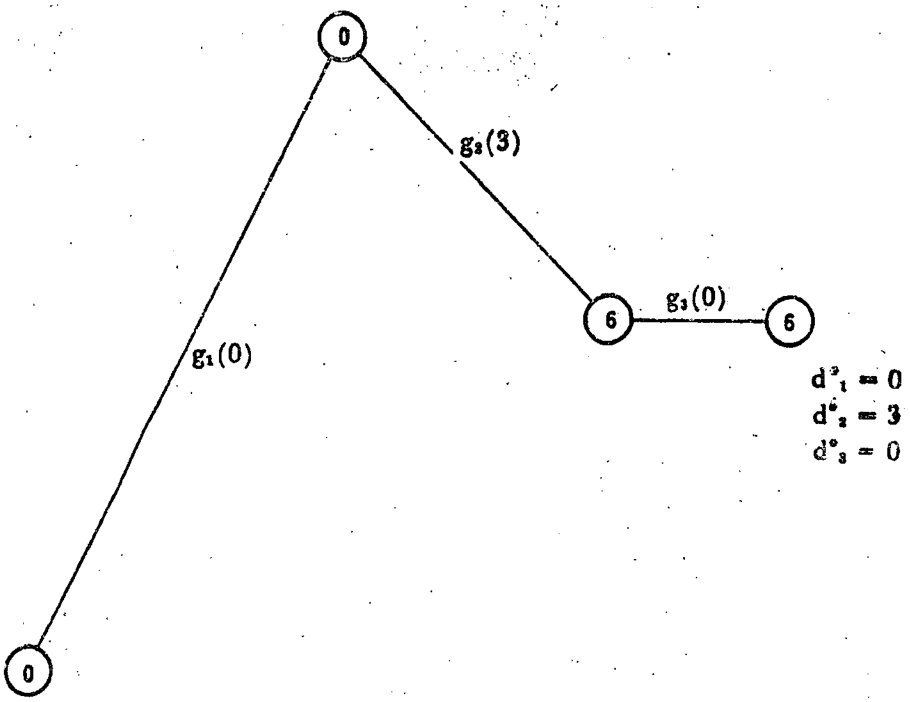


Fig. 6.6.18 Secuencia de decisiones que llevan a un beneficio determinado.

Como estos problemas tienen en general muchas más alternativas que las que se presentan en este ejemplo y más etapas de decisión, el método enumerativo directo requería de una gran cantidad de operaciones y de conservar en la memoria una gran cantidad de información: todas las posibles secuencias de la variable de decisión entre otros datos. La programación dinámica, al reducir el número de alternativas entre las que hay que buscar el óptimo, disminuye los tiempos de computación y los requerimientos de memoria. A pesar de ello, uno de los factores que ha limitado la aplicación de este método es precisamente el requerimiento de memoria que se necesita. En el capítulo 11 de la ref. 1 el lector puede encontrar una presentación formal sobre el problema de reducción del esfuerzo computacional entre la búsqueda directa y la programación dinámica.

La solución de un problema de asignación de recursos con un número mayor de etapas que el del ejemplo 6.6.1 puede encontrarse empleando el programa A18 del apéndice A. Este programa requiere de los siguientes datos:

a) Número de industrias
b) Monto de la inversión
c) Funciones de beneficio de cada industria.

El resultado de este programa aparece en la tabla 6.6.6.

Tabla 6.6.6 Resultados del programa A18 para el ejemplo 6.6.1.

LOS RESULTADOS OBTENIDOS SON (LOS VALORES DE LA MATRIZ CORRESPONDEN A LAS INVERSIONES NECESARIAS A EFECTUAR EN CADA INDUSTRIA)

BENEFICIO	INDUSTRIA		
	1	2	3
0	0	0	0
3	1	0	0
3	0	0	1
6	1	0	1
8	1	1	1
8	1	0	2

Antes de continuar debe hacerse notar que en cada etapa de la solución es necesario encontrar un máximo (o mínimo). Para encontrarlo, de acuerdo con el tipo de problema se aplica alguna de las técnicas expuestas en las secciones anteriores de este capítulo, o bien una búsqueda del tipo introducido en las secciones 3.5.2 ó 3.5.3.

6.6.4 Redes de transporte

Una aplicación importante de la programación dinámica es la determinación de rutas más largas o más cortas en redes de transporte entre dos localidades. En esta sección se ilustra este problema.

Ejemplo 6.6.2

***Posibles rutas del litoral al interior.**

*La figura 6.6.19 ilustra las posibles rutas entre una localidad V y dos puertos de un litoral. Supóngase que las poblaciones intermedias son de tres tipos, cercanas a la localidad, cercanas al litoral e intermedias, agrupadas como muestra la figura 6.6.19.

Los números asociados a las carreteras indican su longitud. Se trata de obtener la ruta más corta entre la población V y el litoral.

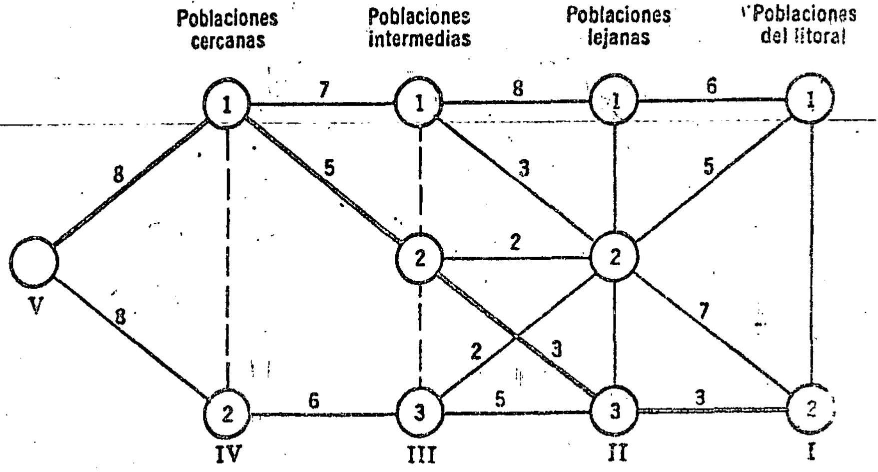
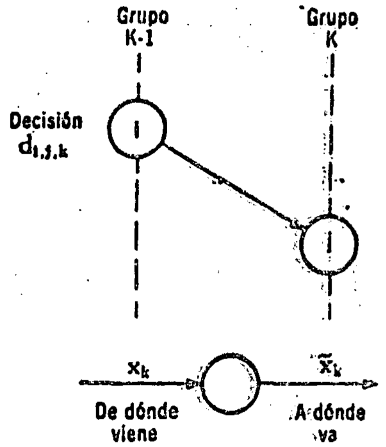


Fig. 6.6.19 Red de caminos entre la localidad V y puertos de un litoral.

*Para resolver todo problema conviene introducir una notación adecuada. Designemos con $d_{i,j,k}$ con la decisión de ir de la población i del grupo k-1, a la población j del grupo k. *Cada variable de estado de entrada x_k indica de qué población de la zona anterior viene la carretera, y la variable de salida $\tilde{x}_k$ indica hacia qué población de la zona siguiente va la carretera.



Con esta nomenclatura se puede empezar a resolver el problema.

*Para la 1ra. etapa, o sea la ruta entre el litoral y las poblaciones lejanas se tiene como óptimo de la función objetivo:

*Del litoral a las poblaciones lejanas

$$f_1(x_1) = \min_{d_1} \{ r_1(x_1, d_1) \}$$

La tabla 6.6.7 resume los resultados para encontrar el óptimo.

Tabla 6.6.7 Obtención del beneficio óptimo en la 1ra. etapa.

Población anterior x_1	Indices de la 1ra. decisión	Longitud del camino r_1	Población siguiente $\tilde{x}_1$	Optimo $f_1(x_1)$	Decisión óptima d_1
II₁	1 1 1	6	1	6	1 1 1
II₂	2 1 1 2 2 1	5 7	1 2	5	2 1 1
II₃	3 2 1	3	2	3	3 2 1

Para la comunicación entre las poblaciones lejanas y las intermedias, etapa 2, se tiene:

*Entre poblaciones lejanas e intermedias.

$$f_2(x_2) = \min_{d_2} \{ r_2(x_2, d_2) + f_1(x_2, d_2) \}$$

Estos valores se resumen en la tabla 6.6.8

Tabla 6.6.8 Obtención del beneficio óptimo en 2 etapas.

Población anterior x_2	Indices de la 2da. decisión	Longitud r_2	$x_1 = \tilde{x}_2$	$f_1(x_1)$	$r_2 + f_1$	Optimo $f_2(x_2)$	Decisiones óptimas d_I°	d_{II}°
III₁	1 1 2	8	1	1	9	8	2 1 1	1 2 2
	1 2 2	3	2	5	8			
III₂	2 2 2	2	2	5	7	6	3 2 1	2 3 2
	2 3 2	3	3	3	6			
III₃	3 2 2	2	2	5	7	7	2 1 1	3 2 2
	3 3 2	5	3	3	8			

*Para la etapa 3 la fórmula para determinar el beneficio es:

°Entre poblaciones intermedias y cercanas

$$f_3(x_3) = \min_{d_3} \{r_3(x_3, d_3) + f_2(x_3, d_3)\}$$

La búsqueda en este óptimo se resume en la tabla 6.6.9

Tabla 6.6.9 Obtención del beneficio óptimo en 3 etapas.

Población anterior x_4	Indices de la 3ra. decisión	Longitud r_3	$x_2 = \tilde{x}_3$	$f_2(x_2)$	r_3+f_2	Optimo valor $f_3(x_3)$	Decisiones óptimas		
							d_I^o	d_{II}^o	d_{III}^o
IV₁	1 1 3	7	1	8	15	11	321	232	123
	1 2 3	5	2	6	11				
IV₂	2 3 3	6	3	7	13	13	211	322	233

*Finalmente para elegir las rutas entre la 1ra. localidad y las poblaciones cercanas se tiene:

° Tramo final

$$f_4(x_4) = \min_{d_4} \{r_4(x_4, d_4) + f_3(x_3, d_3)\}$$

Para encontrar este mínimo se realizan los cálculos que aparecen en la tabla 6.6.10

Tabla 6.6.10 Obtención del beneficio óptimo en 4 etapas.

Población anterior x_4	Indices de la 4a. decisión	Longitud r_4	$x_3=\tilde{x}_4$	$f_3(x_3)$	$r_3 + f_3$	Valor óptimo $f_4(x_4)$	Decisiones óptimas			
							d_I^o	d_{II}^o	d_{III}^o	d_{IV}^o
x_5	1 1 3	8	1	11	19	17	321	232	123	114
	1 2 3	8	2	13	21	21	211	322	233	123

De esta última tabla se concluye que el camino de mínima longitud entre los puestos del litoral y la población V tiene una longitud de 17 a lo largo de la ruta 114, 123, 232 y 321, marcada con trazo grueso en la figura 6.6.18.

El lector interesado en profundizar sobre este tema puede consultar las refs. 1, 2, 5, 8 y 9. Los problemas 16 a 19 de la sección 6.8 ilustran diferentes aplicaciones de este método.

Ejemplo:

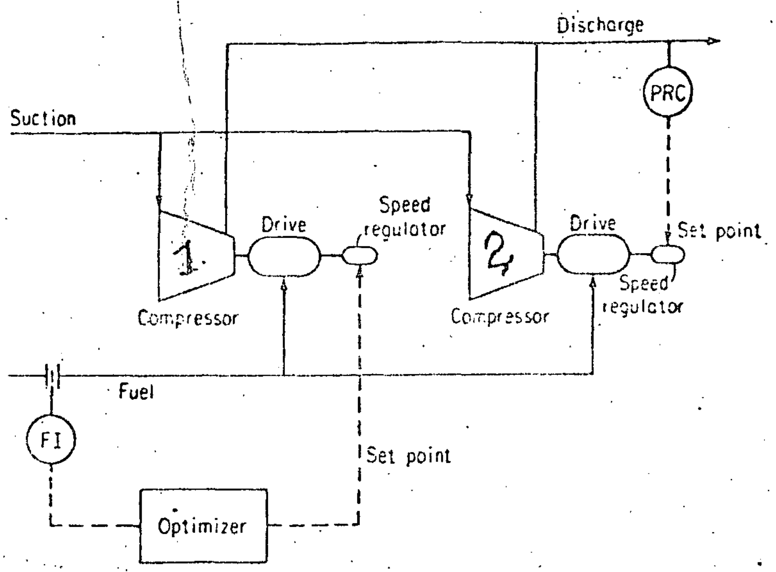
Objetivo. Regular la velocidad de los accionamientos para mantener la presión de descarga, gastando mínimo combustible.

Restricción: Mantener presión

Objetivo: Minimizar el consumo de combustible.

Es difícil emplear control óptimo porque cambian las características con el tiempo

Emplee el proceso para evaluar la bondad de la técnica de control



Estategia:

1.- Estimar la velocidad correcta y medir el consumo

2.- Cambie la velocidad de la compresora 1 en $\Delta \omega_1$

3 - Después de ajustarse automáticamente la velocidad ω_2 para mantener la presión, se mide Q

4 - Si Q ha disminuido se repite el procedimiento en igual

dirección.

Es un problema de busqueda unidireccional.

La tecnica descrita es lenta.

Anexo 3

CONTROL AUTOMATICO DEL SISTEMA ELECTRICO NACIONAL DE SERVICIO PUBLICO.

1. - INTRODUCCION.

El siguiente escrito consta de seis secciones. Después de esta introducción se revisan diferentes conceptos de Ingeniería de Sistemas, señalando la jerarquización que existe en los controles de un sistema a diferentes niveles, con objeto de establecer una secuencia lógica de automatización.

En la siguiente sección se señalan diversas razones por las que debe automatizarse un sistema eléctrico de servicio público.

Posteriormente se señalan los objetivos de un sistema de control de producción en la industria eléctrica.

A continuación se propone una estructura para un sistema de control del sistema eléctrico nacional.

Finaliza este artículo con un párrafo de conclusiones.

2. - INGENIERIA DE SISTEMAS.

Actualmente las empresas eléctricas de servicio público son complejos sistemas. Para obtener una adecuada solución a los problemas que se presentan en su operación, es preciso recurrir a la metodología más avanzada de la Ingeniería de Sistemas.

La metodología de la Ingeniería de Sistemas se basa en el reconocimiento formal de la importancia que tiene la interacción entre las partes de un sistema con su funcionamiento.

Diseñar un sistema consiste en traducir una serie de objetivos ó funciones del mismo a especificaciones del sistema por construir.

El análisis ó síntesis de sistemas se inicia substituyendo el problema real por un modelo, éste a su vez se caracteriza por una serie de relaciones matemáticas que representan el sistema con sus objetivos y restricciones. La simulación que se realiza con este modelo desempeña un papel de gran importancia en la búsqueda de una solución al problema. Permite ensayar varias soluciones alternativas, evaluarlas y solamente después de este paso se procede a construir el sistema.

La industria eléctrica, como toda industria, tiene una estructura piramidal que consiste en un proceso físico y su controlador. (Fig. 1).

El controlador manipula el proceso con el fin de alcanzar los objetivos de la industria, que en este caso son satisfacer la demanda de energía eléctrica con la máxima confiabilidad y los mínimos gastos.

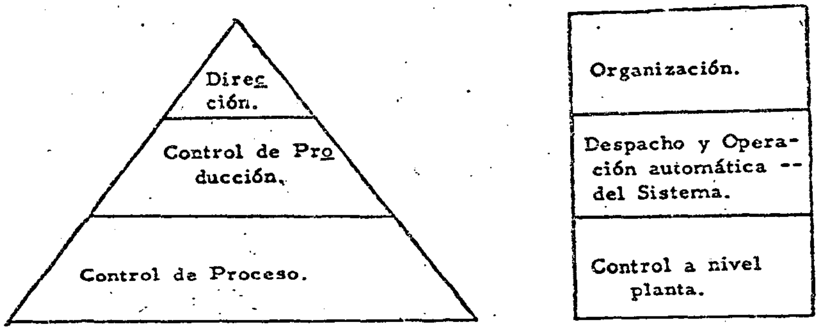


Fig. 1. - Estructura jerárquica del control.

Pueden distinguirse en general tres funciones de control a diferentes niveles. En el primer nivel se tienen aquellas funciones asociadas con el control de las unidades de manufactura, que en el caso de la industria eléctrica corresponden a las plantas generadores. En el segundo nivel las funciones de control guían las actividades de producción mediante despacho de carga, operaciones de conexión, etc.

En el último nivel las funciones de control corresponden a la dirección empresarial e incluyen el establecimiento de objetivos para ser alcanzados dentro de las restricciones del sistema.

En paralelo con las jerarquías señaladas en el nivel de control, al movernos hacia la cumbre de la pirámide, podemos identificar una jerarquía de funciones de control, regulación, optimización, adaptación y organización automática. Puede observarse que a medida que se avanza hacia la cúspide, el énfasis en las variables físicas disminuye y aumenta la importancia de las variables económicas en el proceso de toma de decisiones o funciones de control.

Otra característica del control de sistemas es la decreciente frecuencia de las acciones controladoras y la creciente complejidad del proceso de toma de decisiones al ascender a través de la jerarquía de control.

Debe notarse también que dentro del primer nivel los problemas de control son determinísticos mientras que se vuelven crecientemente probabilísticos al ascender a través de la jerarquía del sistema de control.

Todos estos controles, ya sean máquinas o seres humanos, son procesadores de información. Reciben información sobre el estado del sistema y en función de ésta y del conocimiento de los objetivos del sistema y sus restricciones, ejecutan acciones controladoras.

Durante varias décadas no fue posible implantar la automatización de los sistemas más allá del primer nivel, o sea el nivel planta, por limitaciones que imponía la tecnología existente.

3.- NECESIDADES DE AUTOMATISMO.

Para satisfacer la creciente demanda de energía eléctrica, cada vez se emplean por razones económicas, unidades generadoras de mayor capacidad. Para mantener con estas unidades una adecuada confiabilidad del servicio dentro de límites económicos, es necesario interconectar los sistemas. La interconexión presenta además beneficios adicionales derivados de otros aspectos del aprovechamiento económico del sistema.

Debido al crecimiento de la demanda y a la interconexión de los diferentes subsistemas eléctricos, la complejidad del sistema va en aumento, haciendo cada vez más difícil mantener una adecuada seguridad y calidad en el suministro de energía y minimizar los costos de producción mediante técnicas manuales de operación del sistema.

La ingeniería de sistemas permite conceptualizar sistemas de control automático implementados a diferentes niveles jerarquícos que no tienen las limitaciones de los sistemas de control manuales.

4.- OBJETIVOS.

Un sistema automático de control permite, mediante un mejor conocimiento del estado del sistema y una predicción de los efectos sobre el mismo de diversas acciones de operación, aumentar la seguridad del sistema eléctrico. Un sistema de control de este tipo permite además minimizar los costos de operación y mediante una mejor distribución de los reactivos en

la red hace posible sostener los niveles de voltaje requeridos en el sistema.

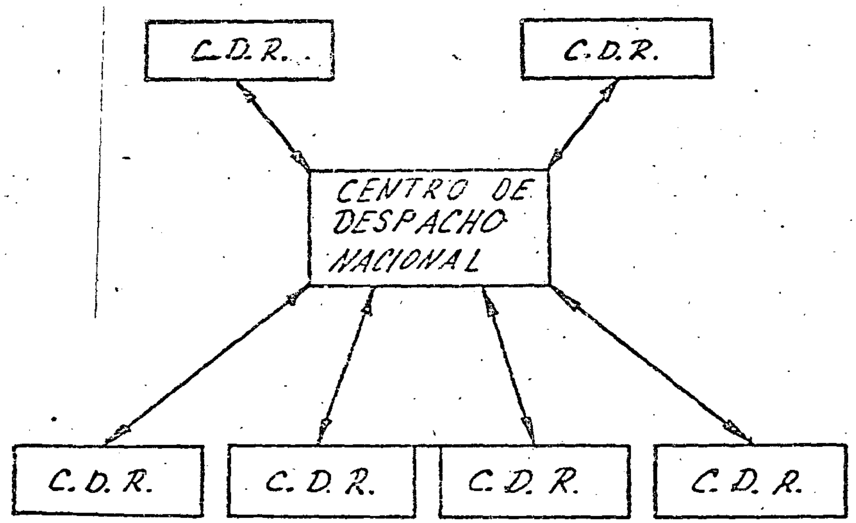
5.- ESTRUCTURA DEL SISTEMA.

El avance tecnológico actual permite alcanzar varios de los objetivos señalados empleando sistemas de control cuyos elementos básicos son computadoras digitales que trabajan en tiempo real. El control digital presenta respecto al analógico varias ventajas. Su mayor flexibilidad permite implementar mejores esquemas de control en tiempo real. Además, puede emplearse la computadora trabajando en tiempo compartido para realizar cálculos de apoyo a la operación del sistema.

Debido a la contínua aparición de mejores técnicas de control, la flexibilidad de un sistema digital permite su implementación con cambios mínimos en el equipo (hardware).

El tamaño de la República y la distribución geográfica no uniforme de los centros de carga y de los recursos de generación han determinado la estructura actual del sistema; una serie de subsistemas hasta hace poco aislados eléctricamente. Por las razones señaladas éstos subsistemas se han ido interconectando. La capacidad de estos enlaces en general no permite el libre flujo de energía en ellos. Por lo tanto, cada subsistema debe absorber sus propias variaciones de carga, manteniéndose en los enlaces flujos de energía programados en base a consideraciones físicas y económicas.

Las razones anteriores apuntan hacia la conveniencia de implementar un control automático de producción a dos niveles por área y central.



C.D.R. Centro de despacho regional

Fig. 2. - Control Nacional y Controles Regionales.

Como muestra la figura 2. Esta estructura de control además presenta otras ventajas:

a). - Las necesidades de canales de telemedición se reducen, consideración muy importante dado el tamaño de la República.

b). - Disminuye el tamaño y la complejidad de los sistemas de control digital.

c). - Permite hacer consideraciones más precisas sobre pérdidas de transmisión.

Las funciones de los centros de control locales son básicamente de supervisión y de reparto económico de la generación asignada al área. El control central recibe información sobre el estado de las diferentes áreas a nivel de transmisión y asigna a cada área su participación en la generación total del sistema en base a consideraciones de seguridad y económicas y controla el flujo en los enlaces.

La Fig. 3 esquematiza un posible funcionamiento del sistema de control jerarquizado.

6. - CONCLUSIONES.

Este escrito ha mostrado la factibilidad y necesidad de implementar un sistema de control de la red eléctrica nacional a diferentes niveles con objeto de garantizar la continuidad de servicio y los costos mínimos de --

generación que el crecimiento económico del país requiere.

Noviembre 9 de 1972.

Dr. Víctor Gerez.

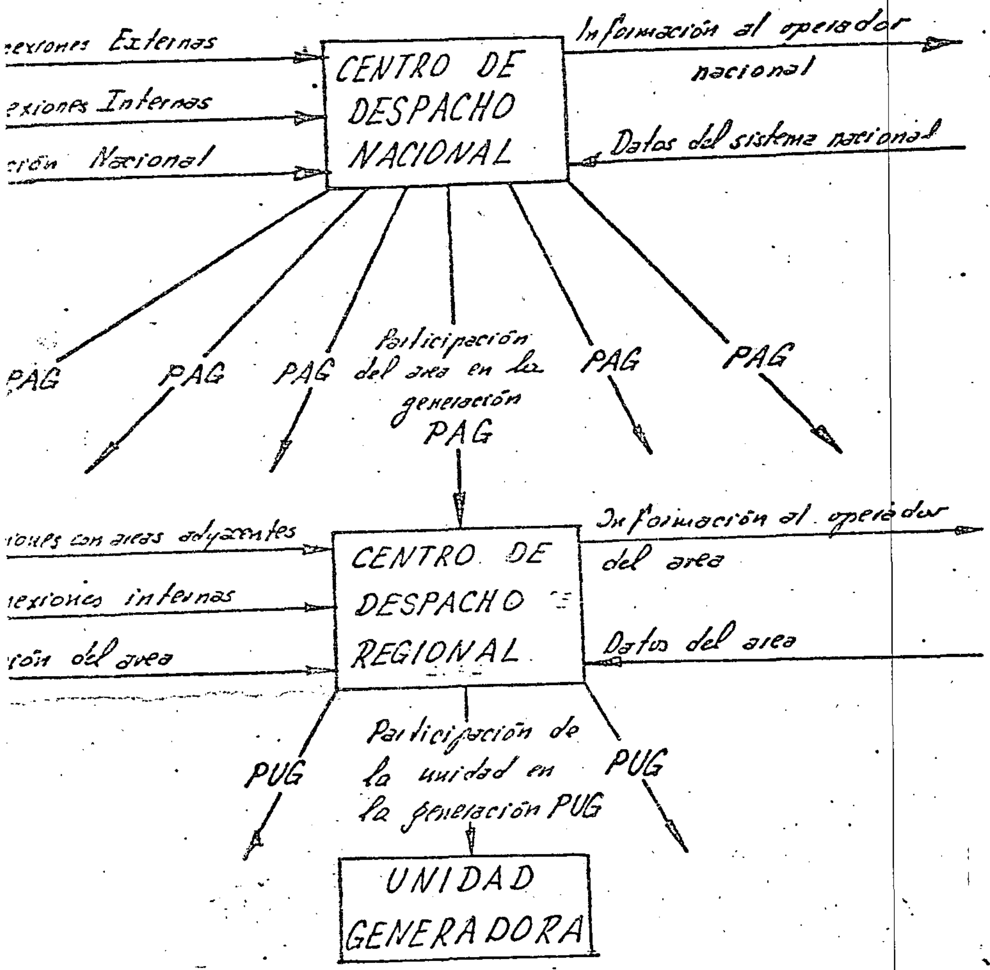


Fig. 3.- Esquema del sistema de control propuesto.

Nvo

Anexo 4

REQUERIMIENTOS FUNCIONALES DE UN SISTEMA DE CONTROL DE AREA.

Se describirán los requerimientos funcionales generales de un sistema de control y adquisición de datos para una área del sistema eléctrico de potencia nacional. Posteriormente se darán los detalles de estos requerimientos.

I.- GENERALIDADES.

El elemento principal de este sistema es una computadora digital de proceso con respaldo digital ó analógico en el centro de control del área. Está comunicada con el sistema de potencia en el área y el centro de despacho nacional mediante una serie de dispositivos digitales de control y adquisición de datos.

Existirán canales de telemetría a las centrales eléctricas y a las subestaciones más importantes del área. Canales de telecontrol a las plantas eléctricas más importantes permitirán el control digital directo de estas unidades generadoras.

Las responsabilidades de los operadores en el centro de control del área son la generación, el intercambio de energía con otras áreas, la seguridad del sistema de potencia del área y la coordinación con otras áreas.

II. - CENTRO DE CONTROL DEL AREA.

Una responsabilidad primordial del centro de control del -- área (CCA) será el control de la carga del área. Para realizar ésta así como otras funciones, contará con una computadora digital, el equipo central de adquisición de datos y de control, un sistema de telerecepción y de trasmisión, consolas de operadores, tableros de instrumentos y un diagrama -- mímico del área a nivel de trasmisión.

El centro de control de área tendrá dos modos de operación, uno primario y otro secundario o de respaldo, en caso de falla del primario. El centro ejecutará los cálculos para el control de la frecuencia-carga cada dos segundos, y los cálculos necesarios para el despacho económico de carga cada 5 minutos (ó cuando resulte necesario debido a cambios en la carga).

El programa de control de frecuencia-carga genera las señales que en forma de pulsos eleva/disminuye se envían por medio de los canales de telecomunicación a las diversas plantas bajo control.

1. - Facilidades de Computación Digital. Consistirán de una computadora digital con memoria adicional de tambor ó disco, una lectora de tarjetas, una perforadora de tarjetas, una impresora de línea y unidades de - cinta, así como el sistema operativo necesario. Incluye también el equipo necesario de interfase para comunicar la computadora con los canales de - teletrasmisión, las consolas de los despachadores, otras computadoras y - terminales remotas con tubos de rayos catódicos.

##

La Tabla No. 1 lista los diferentes programas con que debe contarse, su frecuencia y modo de empleo.

2. - Equipo Central de Adquisición de Datos y Control. Este equipo - suministra a la computadora los datos enviados mediante los canales de telecomunicación desde las terminales remotas en las plantas eléctricas y subestaciones. Esta información contendrá datos del sistema de potencia tales como: niveles de voltaje de línea, flujo de potencia real y reactiva, estado de las unidades, sus límites eléctricos y la generación real y reactiva. El equipo trasmitirá también las acciones de control determinadas por la computadora a las plantas eléctricas. Así mismo proveerán la comunicación -- con el centro de control nacional.

3. - Sistemas de Telemetría. Este sistema recogerá la información sobre el estado del área. Esta información servirá para accionar diversos instrumentos registradores, suministrará las variables de entrada para la computadora principal y su respaldo. Se sugiere el empleo de sistemas de telemetría digitales para las razones que se indican a continuación:

Generalmente, las señales recibidas de los sensores y transductores que miden los parámetros físicos de interés en el sistema son de - forma analoga (contínua). De igual forma las señales que operan los aparatos electromecánicos que se emplean en el control del sistema son también contínuas. Es razonable entonces que en muchos casos, el proceso de control ó computación pueda ser llevado a cabo en forma contínua directamente sin necesidad de técnicas digitales y la conversión necesaria A/D - D/A. --

##

	Continuo	2seg	10 min.	1hora	24hrs.	Cuando se requiere
1 Control de Frecuencia-Carga		E-L				
2 Despacho Económico			E-L			
3 Programación de la Generación Hidroeléctrica					F-L	F-L
4 Predicción de Carga					F-L	F-L
5 Cálculo de Intercambios				E-L		
6 Monitoreo de datos teletransmitidos		E-L				
7 Reserva Rodante del Sistema					F-L	
8 Estimación de Estado			E-L			
9 Identificación del Sistema			E-L			
10 Verificación de Capacidad					F-L	
11 Verificación de Calibracion de teletransmisión				E-L		
12 Elaboración del Relatorio				F-L		
13 Flujo lineal de carga D.C.			E-L			
14 Análisis de imprevistos.			E-L			
15 Flujo de carga C.A.				F-L		
16 Análisis posteriores a disturbios						F-L
17 Preprogramación de la generación					F-L	
18 Costo de producción					F-L	
19 Determinación de las constantes B					F-L	
20 Valores proyectados de almacenamiento					F-L	
21 Programas varios de investigación y desarrollo						F-L
22 Procesamiento de E/S del sistema de potencia			E-L			
23 Procesamiento de interfase hombre/máquina	E-L	-(				
24 Comunicación con otras computadoras y TRC	E-L					
25 Administración de datos					F-L	
26 Servicio de rnóstico				F-L		F-L

Clave: E L Cálculo en línea
F L Cálculo fuera de línea

Existen sin embargo diversas razones que justifican el paso adicional de digitalización. Pueden citarse las siguientes razones:

En general, las técnicas digitales ofrecen la ventaja de una mayor exactitud, la posibilidad de minimizar el ruido en la medición y un mejor procesamiento, trasmisión y almacenamiento de la información. Además la resolución puede ser incrementada aumentando el número de bits utilizados en el código.

Por otra parte, las limitaciones de formato en las mediciones analógicas dá como resultado una resolución pobre (aunque tales señales teóricamente poseen resolución infinita), además la exactitud de la medición se degrada después de cada operación. Tal degradación en la exactitud ocurre con operación digital.

Una buena razón de la creciente popularidad de los aparatos digitales puede ser que el costo de fabricación de estos es cada día menor.

4. - Consolas de Operadores y Unidades Visuales de Despliegue. Con objeto de seleccionar las unidades visuales de despliegue se hace una comparación entre los diagramas de pared y las unidades de despliegue visual controladas por computadora.

En la operación manual del sistema el operador cuenta con información contenida en diversos instrumentos registradores y en el diagrama de pared.

##

No obstante que el diagrama de pared se considera una herramienta adecuada para el control de la red, se observa que tiene dos limitaciones importantes:

a). - El diagrama de pared es rígido por naturaleza, ya que siempre presenta toda la información al mismo tiempo y además los cambios físicos que ocurren en la red, deben ser efectuados también en el diagrama.

b). - Para sistemas de más de 1,500 megawatts, las dimensiones del diagrama de pared se vuelven prohibitivas (40 m. X 5 m.).

Otro aspecto importante se deriva de que para lograr un control efectivo es esencial que se presente el nivel de información adecuado al operador en el momento preciso, esto es de especial importancia cuando existen condiciones anormales, es decir en aquellos casos en que la restauración de la operación correcta depende del hábil manejo del operador que a su vez depende de la información que recibe.

La solución a las limitaciones del diagrama de pared y la presentación efectiva de datos se ha obtenido con unidades de despliegue visual, estos equipos son manejados por un computador y permiten llevar la supervisión y el control de la red. Existe una gran variedad de equipos de despliegue visual, los cuales varían en complejidad desde mecanismos muy simples que presentan únicamente textos alfa-numéricos hasta equipos que permiten una representación completa del sistema.

##

La unidad de despliegue visual más empleada es un tubo de rayos catódicos que está conectado al sistema de computadora a través de una unidad de control que regenera la imagen representada.

Como estas unidades son completamente programables y las imágenes que se van a desplegar están almacenadas en la memoria del computador digital, ellas proporcionan una herramienta muy flexible para el uso del operador. El despachador de la red puede de esta manera tener en cualquier momento y de cualquier sitio de la red, la información deseada. Es decir que se tiene una visión telescópica dentro de la red de potencia empezando con un diagrama unifilar que muestra una visión general de la red, después se puede hacer un despliegue que muestre alguna parte de la red en detalle. A continuación se puede presentar el diagrama de alguna subestación específica y obtener información de algún aparato particular como generadores, transformadores o interruptores.

Las mismas unidades pueden presentar información como son las curvas de carga, datos de aparatos y otras características del sistema, así como una gráfica de la variación del voltaje durante los últimos 30 minutos en las líneas principales.

Podemos concluir que con las unidades de despliegue visual es posible una presentación seleccionada de gran cantidad de información sobre el sistema de potencia, lo cual es impráctico, o bien imposible, de representar en un diagrama de pared o en instrumentos registradores. En el control de redes de potencia se emplean tubos de rayos catódicos TRC en

##

blanco y negro o en color.

El uso de TRC a color ha ido en aumento, apesar de su mayor costo debido a:

a) Los TRC a color permiten desplegar una mayor densidad de información que sus contraportes en blanco y negro.

b) Se logra una mayor claridad en la información desplegada.

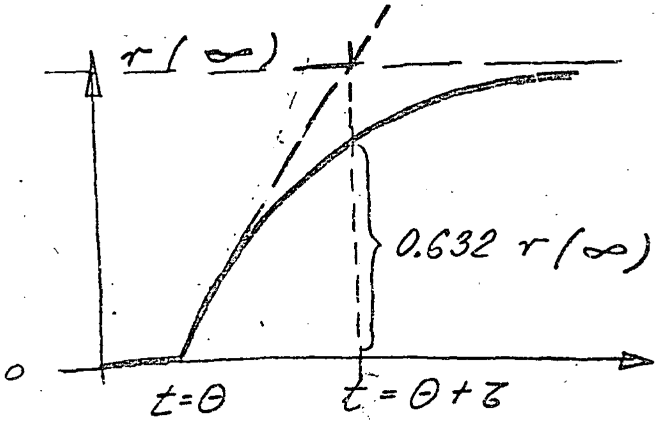
c) La tecnología actual a reducido el precio de los TRC a color de manera que es posible adquirirlos por sumas comparables a las necesarias para adquirir unidades semejantes en blanco y negro.

Se recomienda el empleo de dos consolas. Estas serán funcionalmente idénticas y cada una auxiliará a un operador. Cada consola tendrá dos tubos de rayos catódicos a color. Además contarán cada uno con una impresora silenciosa y un conjunto de teléfonos para comunicación con los CSAD, las subestaciones y generadores bajo control, los demás centros de control de área y el nacional. Los tubos de rayos catódicos serán utilizados para introducir datos, desplegar alarmas, tablas de datos, resultados, etc. Además pueden mostrar diagramas unifilares de las subestaciones. Un teclado se utilizará para introducir datos, pedir despliegues en el tubo de rayos catódicos, pedir impresiones, etc. Las impresoras servirán para registrar alarmas, entradas manuales a la computadora e imprimir en general a solicitud tablas de datos contenidos en la memoria de la computadora.

5.- <u>Instrumentos Registradores y Diagrama Mímico.</u> Se ha señalado en el inciso anterior la ventaja de los despliegues visuales controlados por computadora respecto a los despliegues de datos convencionales. A pesar de estas ventajas varios centros de despacho modernos cuentan con los despliegues en TRC y con los métodos clásicos de despliegue, los instrumentos registradores y los diagramas mímicos, ya que los operadores así lo han solicitado. El personal técnico encargado de la planeación de estos centros sin embargo parece considerarlos superfluos. Para el centro de control de área piloto probablemente sea conveniente contar con estos sistemas clásicos, para facilitar la transición entre la operación manual y automática del sistema.

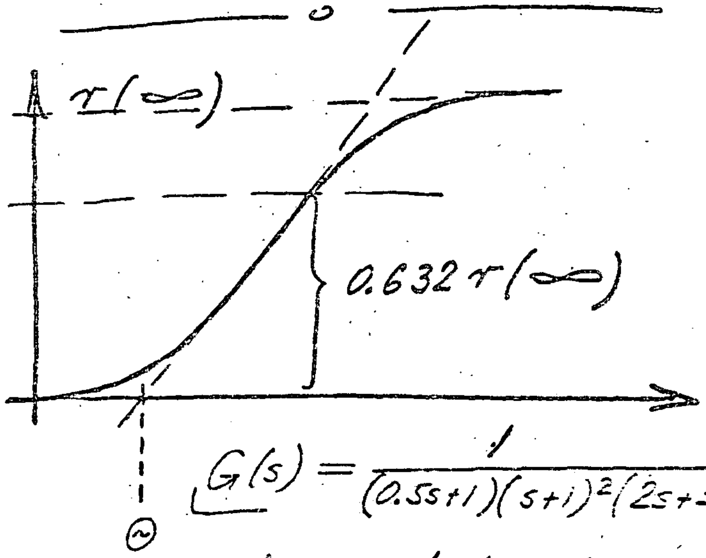
6.- <u>Terminales Remotas.</u> Cada una de las terminales remotas de adquisición de datos y control localizada en las plantas generadoras tiene dos propósitos. Primero, localizar y recoger todos los datos de generación y trasmitirlos al centro de control de área cada vez que reciba una señal de explorar, segundo trasmitir directamente ordenes de eleva/disminuye generación a las unidades bajo control.

②



Sistema autoregulado (Primer orden con atraso + tiempo muerto)

$$G_m(s) = \frac{Ke^{-\theta s}}{\tau s + 1}$$



$$\left. G(s) = \frac{1}{(0.5s+1)(s+1)^2(2s+1)} \right\} \xrightarrow[?]{\text{Exacta}}$$

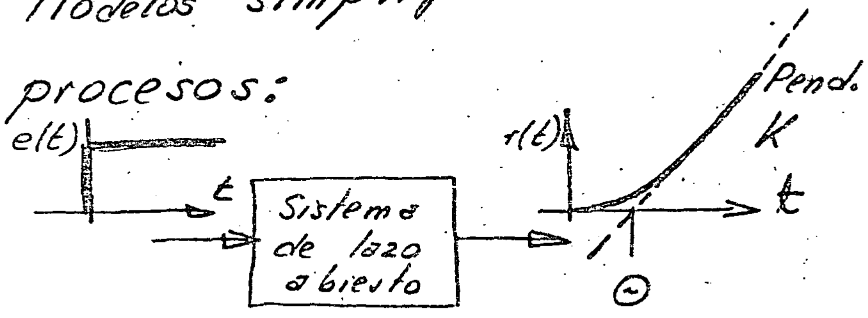
Sistema autoregulado de orden superior

①

ALGORITMOS DE CONTROL

$$\text{Diseño} \begin{cases} \text{Transformada Z} \\ \\ \text{Tecnicas Convencionales} \end{cases}$$

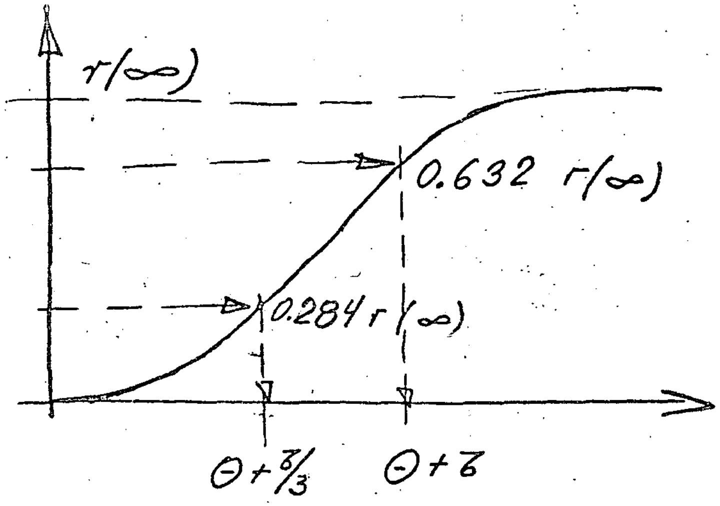
Modelos simplificados de procesos:



$$\underbrace{\text{Función de transferencia ?}}$$

Sistema no autoregulado $r(\infty) \rightarrow \infty$

$$G(s) = \frac{Ke^{-\theta s}}{s}$$



Ejemplo $\left.\begin{aligned} 2.88 &= \theta + \tau/3 \\ 4.80 &= \theta + \tau \end{aligned}\right\}$ 2 ecs. 2 incog.

Comparación de los métodos:

	T. muerto θ	Const. tiempo τ
Milne	1.46	3.34
Analítico	1.92	2.88

Es raro requerir modelos superiores al primero:

Ver: Smith, C.L. Digital Computer Process Control. pp - 141-145

Aproximación: $G(s) = \dfrac{Ke^{-\theta s}}{\tau s + 1}$ (

Determinación de θ y τ (tiempo muerto + const de tiempo)

a) Metodo de Miller (Ver Figura)

b) Método analítico:

$$G(s) = \frac{Ke^{-\theta s}}{\tau s + 1} \xrightarrow{\mathcal{L}^{-1}}$$

$$r(t) = E_m\left(1 - e^{-(t-\theta)/\tau}\right),\ t >$$

$t = \theta + \tau/3 \qquad r = 0.284\ r(\infty)$

$t = \theta + \tau \qquad r = 0.632\ r(\infty)$

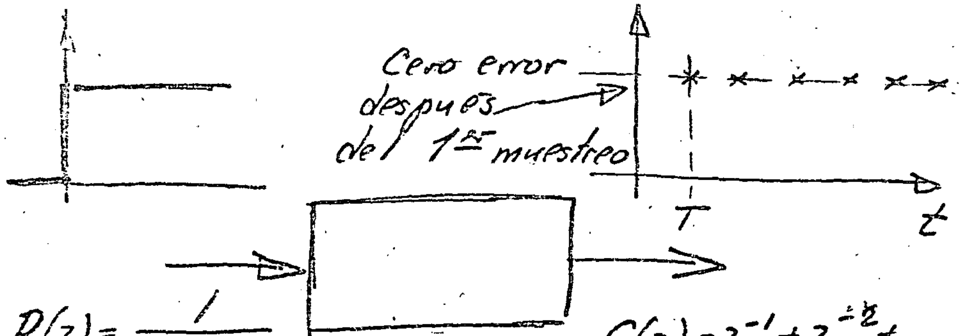
Algoritmos deadbeat.

Especificaciones:

Tiempo de asentamiento finito.

Tiempo de respuesta minimo

$Error(\infty) = 0$



$R(z) = \frac{1}{1-z^{-1}}$

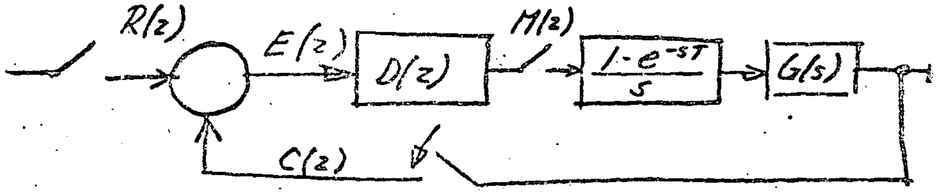
$C(z) = z^{-1}+z^{-2}+$

$= \frac{z^{-1}}{1-z^{-1}}$

Sea $HG(z) = \frac{z^{-1}(0.65+0.12z^{-1})}{1-0.22z^{-1}}$

Nota: Corresponde al sistema especificado en pg 4 T=5

Diseño empleando la transformada z

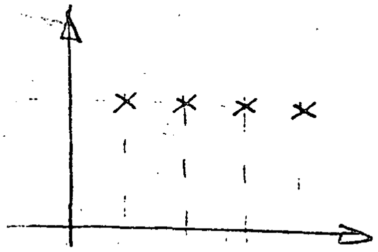


$$C(z) = HG(z)D(z)[R(z)-C(z)]$$

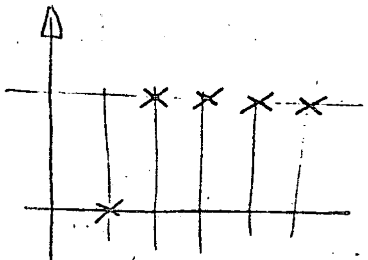
Datos: $\frac{C(z)}{R(z)}$; $HG(z) \longrightarrow$

$$D(z) = \frac{1}{HG(z)} \frac{\frac{C(z)}{R(z)}}{1-\frac{C(z)}{R(z)}} \quad *$$

Observación: Si el proceso contiene tiempos muertos, ninguna $D(z)$ puede eliminarlos, en la especificación de $\frac{C(z)}{R(z)}$ debe haber z^{-N}



$$C(z) = \frac{z^{-1}}{1-z^{-1}}$$



$$C(z) = \underset{\uparrow \text{atraso}}{z^{-1}} \frac{z^{-1}}{1-z^{-1}}$$

Método de Dahlin.

$$\underbrace{c(s) = \frac{e^{-\theta s}}{(\lambda s+1)} \quad \frac{1}{s}}_{\text{respuesta buscada}}$$

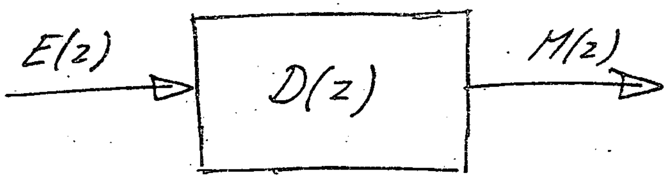
$C(T) = c(\infty)$ es muy estricto:

$$C(z) = \frac{(1-e^{\frac{T}{\lambda}})z^{-N-1}}{(1-z^{-1})(1-e^{\frac{T}{\lambda}}z^{-1})}$$

$*$ $\theta = NT + \theta'$ $\quad \theta' < 1$

Sustituyendo en pg 5 $*$

$$D(z) = \frac{1-0.22z^{-1}}{z^{-1}(0.65+0.12z^{-1})} \quad \frac{z^{-1}}{1-z^{-1}}$$



$$D(z) = \frac{M(z)}{E(z)} =$$

Solución:

$$m_n = \frac{e_n - 0.22e_{n-1} + 0.53m_{n-1} + 0.12m_{n-2}}{0.65}$$

Observación: Si $T = 1$ y como $\theta =$

$C(T) \neq C(\infty)$

Especifique: $C(0) = C(T) = 1$
$C(nT) = c(\infty)$:

$$R(z) = \frac{1}{1-z^{-1}} \quad \text{(escalón)}$$

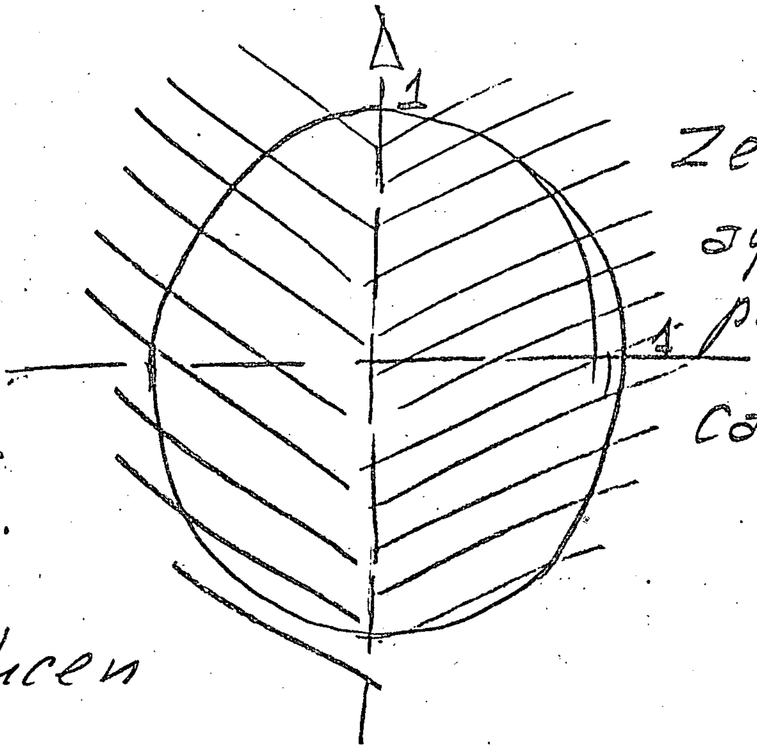
$$\underbrace{\frac{C(z)}{R(s)} = \frac{1-e^{\frac{T}{\lambda}}Z^{-N-1}}{1-e^{\frac{T}{\lambda}}Z^{-1}}}_{\downarrow}$$

igual que en el caso anterior.

Trace la respuesta y observe que λ es "mejor"

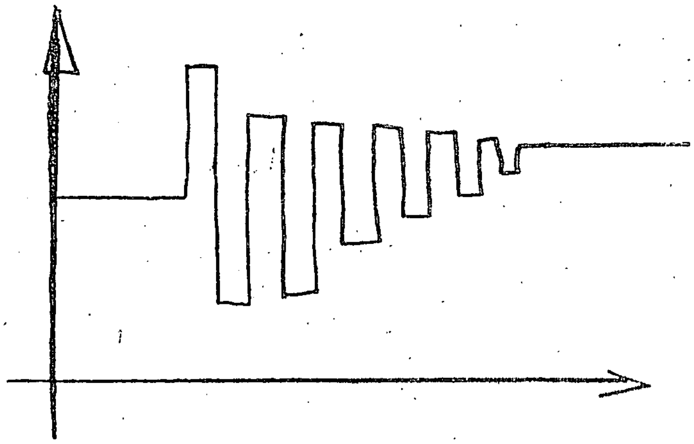
CAMPANEO

La localización de polos y ceros de una función determina a este fenómeno.-



Zeros aqui producen camp.

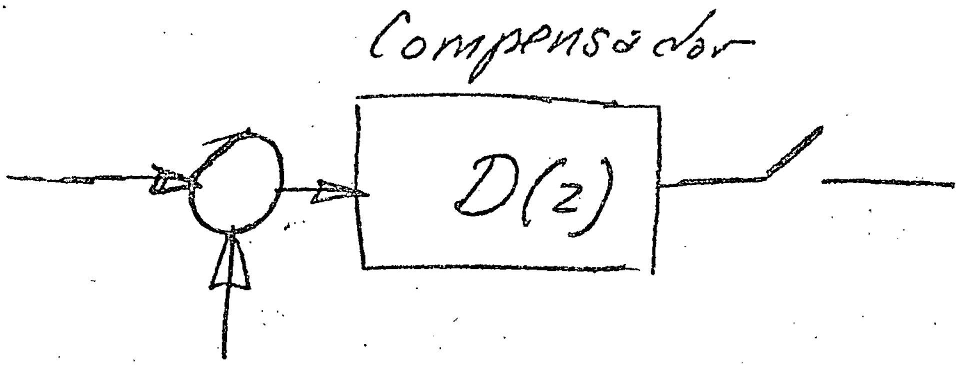
Polos aqui producen campaneo



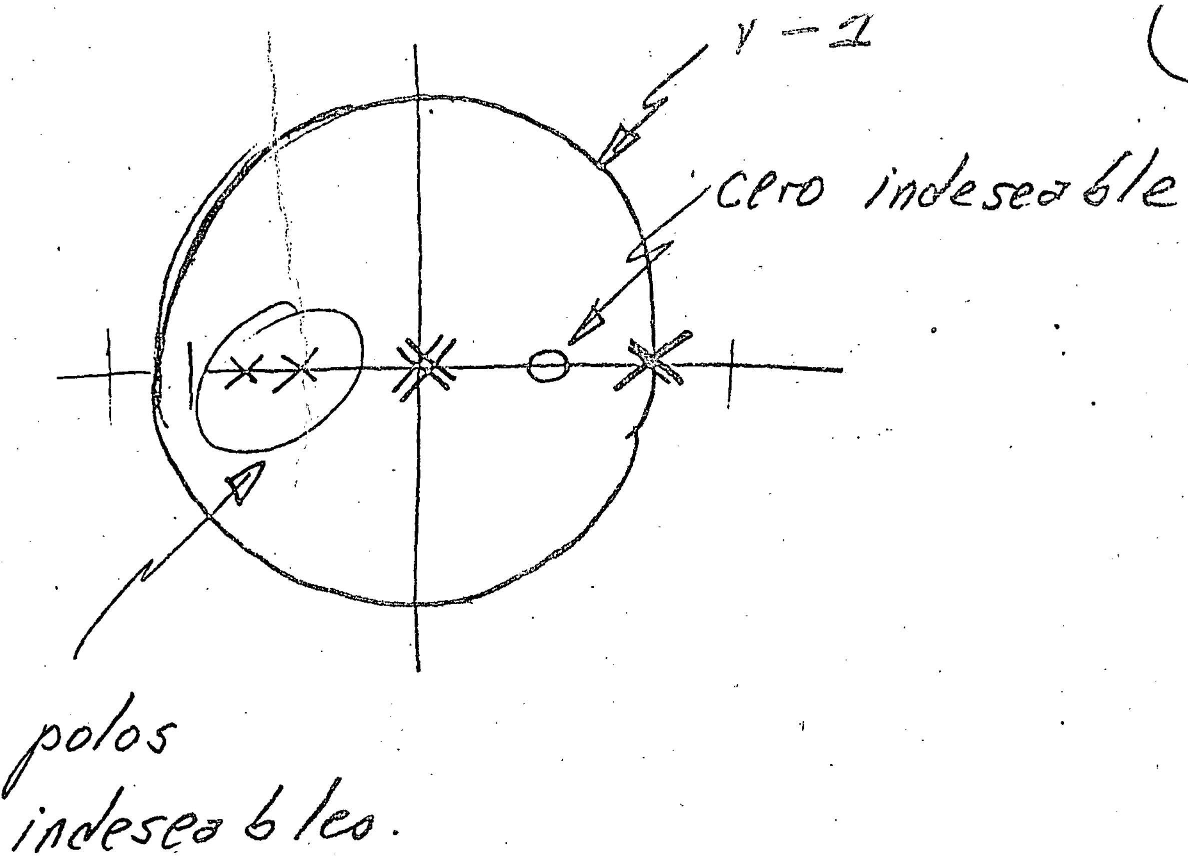
Ejemplo de campaneo.

<u>Observación</u>: Puede <u>no</u> existir en la señal de salida, pero <u>si</u> en alguna intermedia.

Ejemplo:



$$\underbrace{D(z)=\frac{1.96z^2(z-0.741)}{(z-1)(z+0.392)(z+0.738)}}_{\text{Diseñado por el método de Dahlin (Sección anterior)}}$$



Remedio:

Pruebe eliminando el más negativo ajustando la ganancia

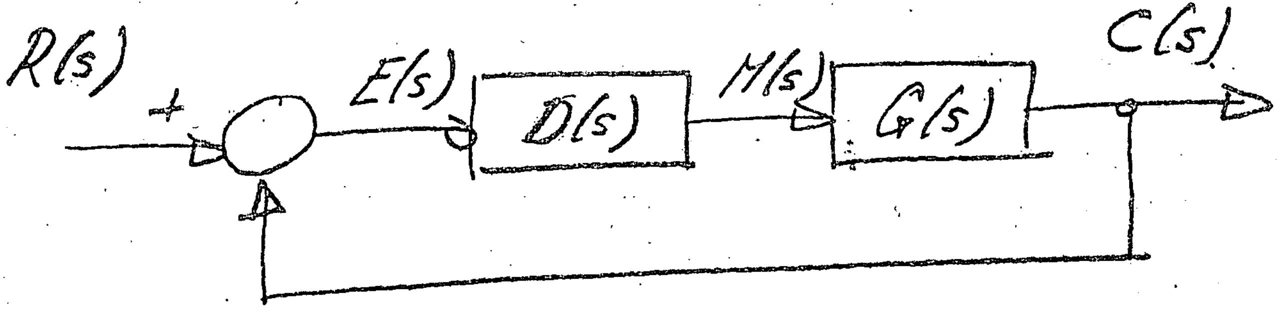
Ganancia ($t \to \infty$) correspondiente a cada termino se obtiene sustituyendo $z = 1$

$$D(z) = \frac{1.96\, z^2 (1 - 0.741 z^{-1})}{(1 - z^{-1})(1 + 0.392 z^{-1})(1 + 0.738)}$$

EQUIVALENTE DISCRETO DE UN CONTROL ANALOGICO

$$\text{Sea} \quad G(s) = \frac{K e^{-\theta s}}{\tau s + 1} \longrightarrow (1)$$

$$\underset{\text{controlador}}{D(s)} = \frac{M(s)}{E(s)} \cdot \frac{1}{G(s)} \; \frac{C(s)/R(s)}{1 - C(s)/R(s)} \qquad (2)$$



Especificación:

$$\frac{C(s)}{R(s)} = \frac{e^{-\theta s}}{\lambda s + 1} \qquad (3)$$

(3) y (1) en (2) ⟶

$$D(s) = \frac{M(s)}{E(s)} = \frac{(\tau s + 1)}{K(\lambda s + 1 - e^{-\theta s})}$$

Equivalente discreto:

$$\lambda \frac{dm}{dt} + m - m(t-\theta) = \left(\tau \frac{de}{dt} + e\right) / K$$

T tiempo de muestreo

$\theta = NT$

$$\lambda \frac{m_n - m_{n-1}}{T} + m_{n-1} - m_{n-N-1} = \quad *$$

$$\left[\tau \frac{e_n - e_{n-1}}{T} + e_{n-1}\right] / K$$

$$\downarrow$$

despejar m_n

Nota: Solo emplear primeras diferencias para aproximar $\frac{d}{dt}$

si $T \leq 0.2\,\tau \quad \lambda \gg 2.5T$

CONTROL DIGITAL DE PROCESO Y APLICACIONES

CONTROL ADOTIVO Y CONTROL OPTIMO

M.enC. RAFAEL LOPEZ.

Noviembre,1978.

CONTROL ADAPTIVO Y CONTROL OPTIMO

INTRODUCCION

El objetivo de esta sesión será presentar las ideas principales de dos campos de aplicación del control moderno

Control adaptivo, que consiste en la implantación de un sistema de control capaz de ajustarse a variaciones en la dinámica del sistema, y Control óptimo, técnica que permite definir el control que deberá aplicarse a un sistema en base a algún criterio de optimización, como puede ser mínimo costo, mínimo tiempo, máxima ganancia, etc.

Se dará énfasis a la aplicación de estos conceptos a sistémas lineales.

CONTROL ADAPTIVO

Existen dos características que degradan la calidad de los algoritmos lineales de control utilizados comunmente en un proceso

1- La planta no es lineal

2- La planta no es estacionaria (es decir, sus características varían con el tiempo)

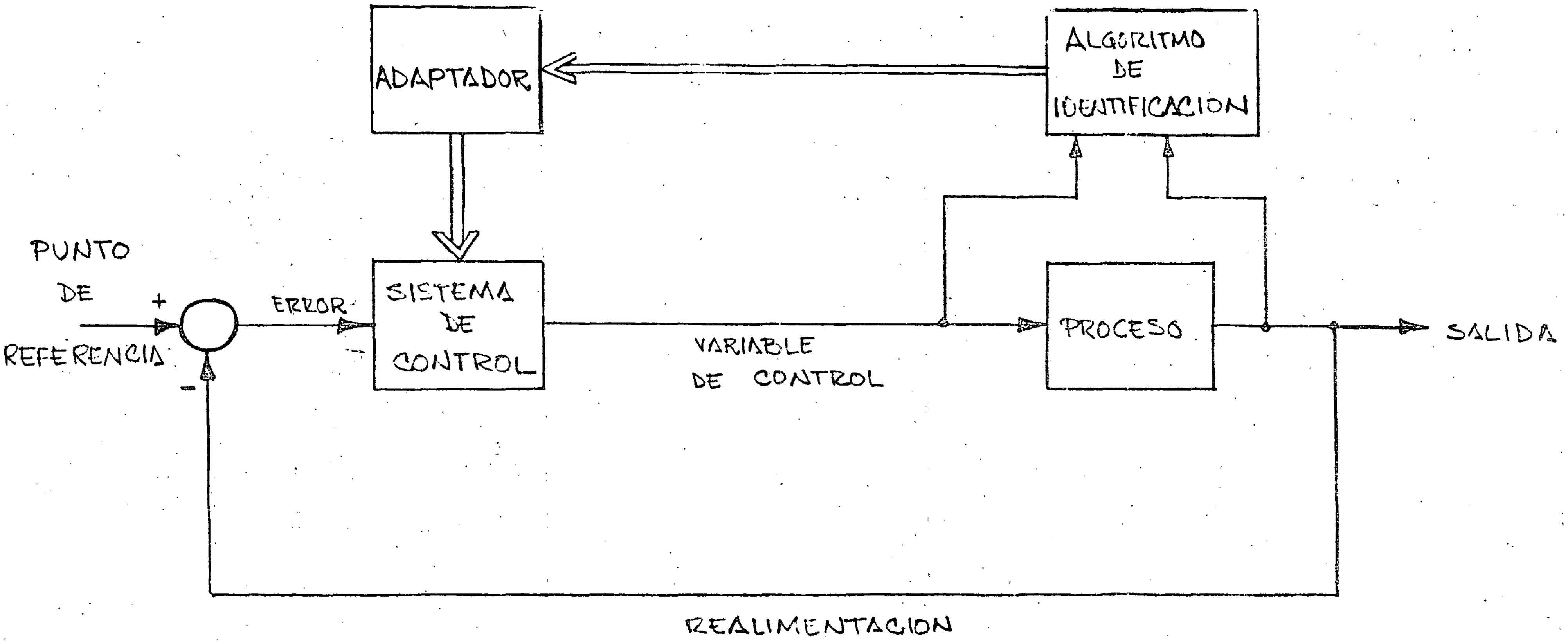
En ambos casos, el resultado neto es que los valores de los parámetros del modelo que se está utilizando para describir la planta cambian continuamente de valor. Es por ello deseable implantar algoritmos de control que tomen en cuenta este hecho y produzcan un comportamiento satisfactorio del sistema global a pesar de los cambios mencionados.

Con esta idea podemos definir un sistema adaptivo en la siguiente forma:

Un sistema adaptivo es aquel que compensa automáticamente las variaciones en la dinámica del sistema, ajustando las características del controlador en forma tal que el comportamiento del sistema global sea satisfactorio.

Un sistema tal deberá incluir elementos para medir o estimar la dinámica del proceso y, en base a esto, cambiar las características del controlador. Estas ideas se presentan esquemáticamente en la figura de la página siguiente.

El bloque descrito en la figura como "Algoritmo de identificación" incluiría alguno de los métodos para la identificación de procesos en-línea, descritos en la sesión del pasado 15 de octubre.



SISTEMA DE CONTROL ADAPTIVO

A fin de simplificar la parte de estimación, o identificación de parámetros, comunmente se emplea un modelo lineal simple para el proceso. Para facilitar esta identificación, se ha sugerido excitar al proceso periódicamente con un pulso (agregado a la entrada "normal" del sistema). Esto presenta claramente la desventaja de que el pulso de prueba debe aplicarse al proceso en línea, por lo que el sistema se desvía de su punto deseado de operación durante algún tiempo. Sin embargo, si este tiempo es corto y además se logra una mejoría apreciable en el funcionamiento del sistema de control, tal prueba estará justificada.

AJUSTE ADAPTIVO DE LA GANANCIA.

Existen casos donde no es posible utilizar señales de prueba, y por ello no se pueden inferir los parámetros del modelo directamente a partir de mediciones en el proceso.

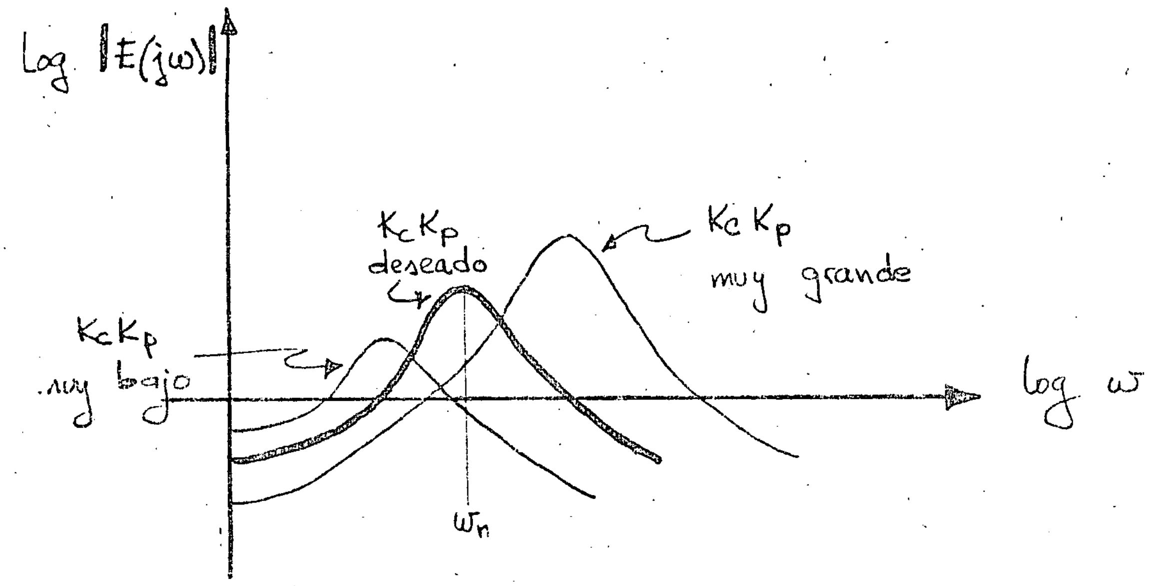
Una alternativa para implantar un control adaptivo se muestra a continuación (ver página 8).

El objetivo es mantener el producto $K_c K_p$ en algún valor deseado, siendo K_c la ganancia del controlador y K_p la del proceso.

El método se basa en las características de frecuencia de la señal de error E, donde, de la figura,

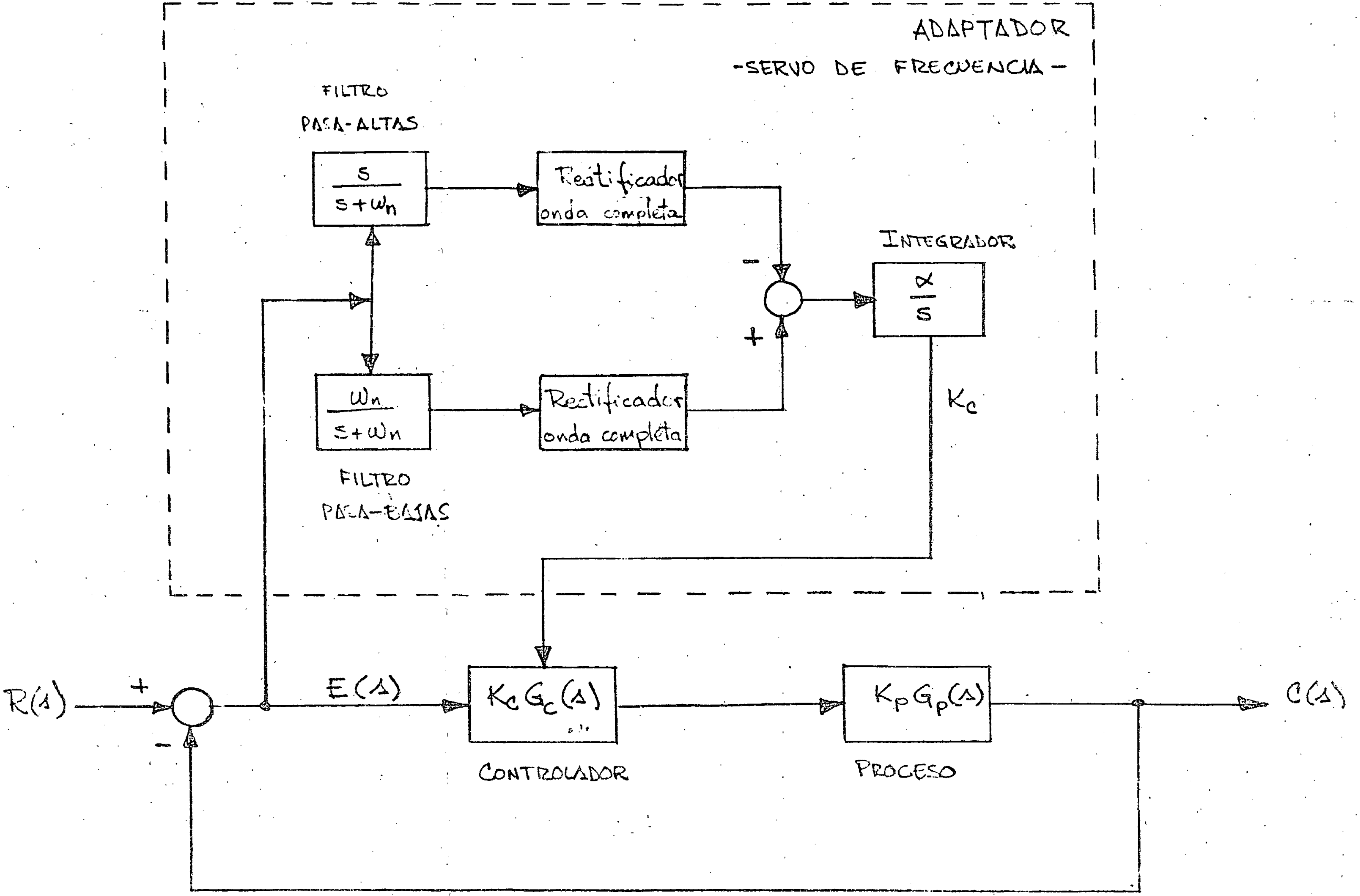
$$E(s) = \frac{1}{1 + K_c K_g \, G_c(s) G_p(s)} R(s)$$

Haciendo $R(s) = \frac{1}{s}$ (entrada escalón) es posible obtener la traza de Bode de $E(j\omega)$ para distintos valores del producto $K_c K_g$, como se muestra en la siguiente figura



Como se ve, la magnitud del error es muy selectiva a la ganancia. De ahí que se emplee un servomecanismo de frecuencia como elemento adaptador, en el diagrama de la página siguiente.

Este servo de frecuencia utiliza un filtro pasa-altas seguido de un rectificador de onda completa (valor absoluto) para detectar cambios en la región de alta frecuencia. En forma similar se procesa la región de baja frecuencia. Finalmente, dado que las amplitudes en altas frecuencias son iguales a las de



AJUSTE ADAPTIVO DE GANANCIA

(α)

bajas frecuencias para el valor deseado de $K_c K_p$, las salidas de los dos rectificadores se comparan, integrando la diferencia para cambiar la ganancia del controlador. Por ejemplo, cuando la ganancia es muy baja, la salida rectificada del filtro pasa-bajas es mayor que la del pasa-altas, produciendo un error positivo que incrementará la ganancia del controlador.

El algoritmo descrito puede ser mejorado en varias formas. Por ejemplo:

1- Reemplazar los filtros por filtros pasa-banda, para mayor selectividad.

2- Utilizar funciones de peso en el comparador, para dar mas importancia a una de las dos regiones (alta o baja frecuencia).

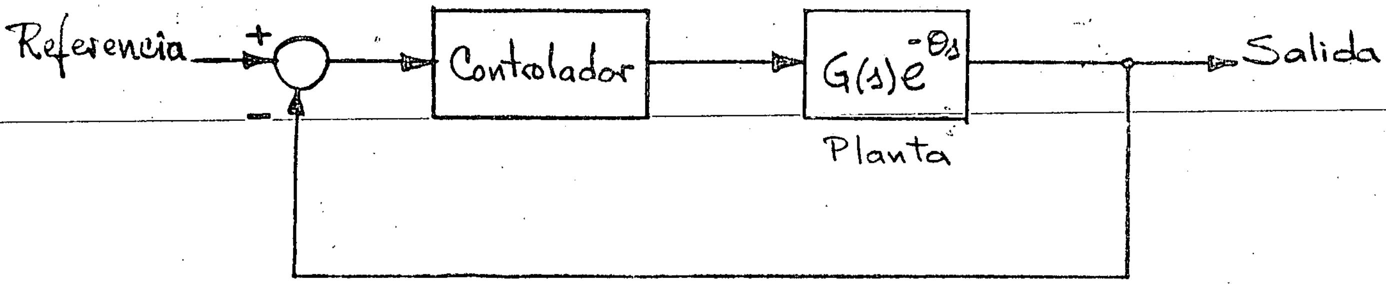
3- Para obtener la ganancia del controlador utilizar, en vez del integrador, una forma más general (PD, PI, PDI)

COMPENSADOR DE RETRASO (TIEMPO MUERTO)

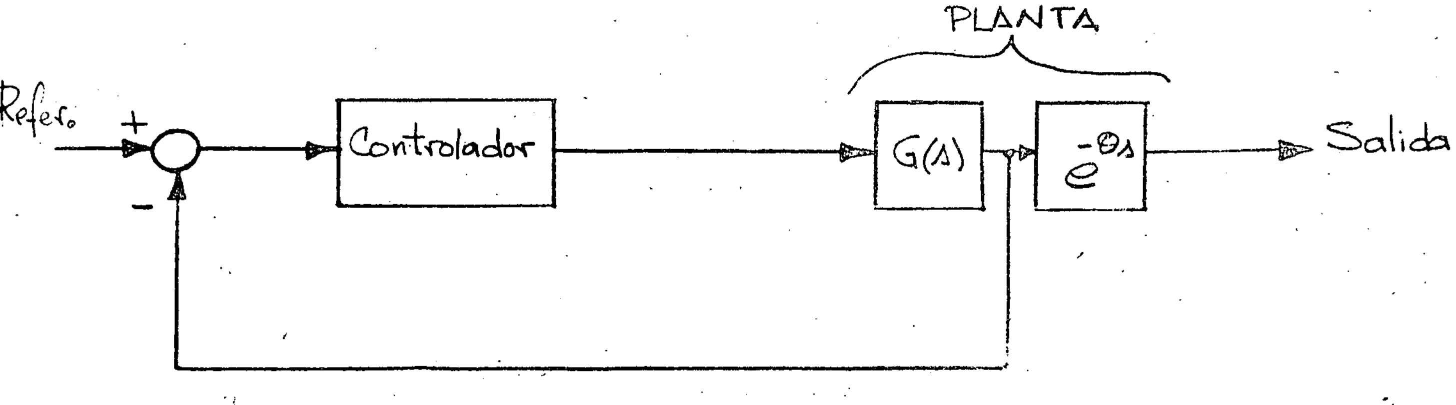
Cuando el proceso que se desea controlar tiene un retraso apreciable

$$G_p(s) = G(s)\, e^{-\theta s}$$

aparece un problema adicional: la señal de realimentación, que debía afectar al sistema en un tiempo t_1, no lo afecta sino θ seg. después, o sea en $t_1 + \theta$.



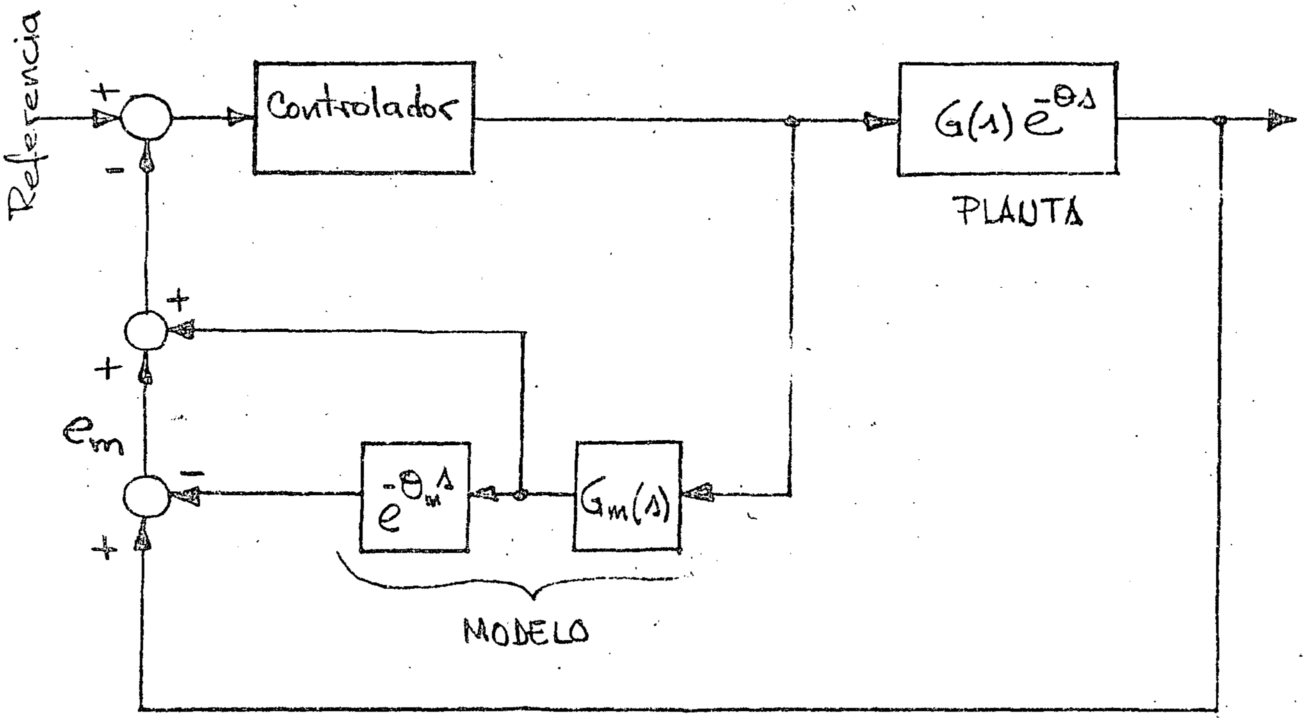
Sería desable tomar la señal "antes" que se retrasara y realimentarla



Esto, desde luego, no es factible en la mayoría de los casos, por lo que se propone la siguiente solución:

Obtener un modelo de $G(s)$, $G_m(s)$ y otro del retraso, $e^{-\theta_m s}$.

El arreglo siguiente proporcionaría el control deseado.



Note que si el modelo fuese exacto, el error de modelado sería $e_m = 0$, obteniéndose la realimentación deseada. Más aún, e_m

puede utilizarse para ajustar adaptivamente el retraso θ_m

El sistema mostrado en la página anterior se conoce como compensador de retraso

Los retrasos, si bien difíciles de implantar en sistemas analógicos de control, son de fácil implantación en controles digitales por computadora, debido a lo cual las ideas anteriores han ganado aceptación últimamente.

CONTROL OPTIMO

PROBLEMA : Definir un algoritmo de control para un proceso dado, buscando la minimización (o maximización) de alguna función objetivo. Esta función puede representar el costo, el tiempo, las ganancias, las desviaciones respecto a una variable de referencia, etc.

Es importante que se obtenga un sistema de control realimentado, es decir que las decisiones de control que se tomen en un tiempo dado estén basadas en el estado del sistema en dicho tiempo. Esto es particularmente importante cuando existe ruido en el sistema

Por ruido entendemos variaciones aleatorias desconocidas en el sistema. Estas pueden deberse a múltiples factores como son imperfecciones en el modelo, errores en el sistema de comunicación, diferencias entre la señal que queremos alimentar al sistema, y la que realmente entra debido a imperfecciones en el equipo, etc. Como se ve, cualquier sistema físico estará afectado por ruido, y este ruido nos impedirá conocer de antemano el estado del sistema en un tiempo dado. Por ello no es conveniente utilizar un esquema de control de malla abierta (donde la entrada al sistema se fija desde el principio y es independiente de la evolución del estado del mismo).

Hay que definir, en cambio, un control realimentado que determine el valor de la entrada en base al estado actual del sistema.

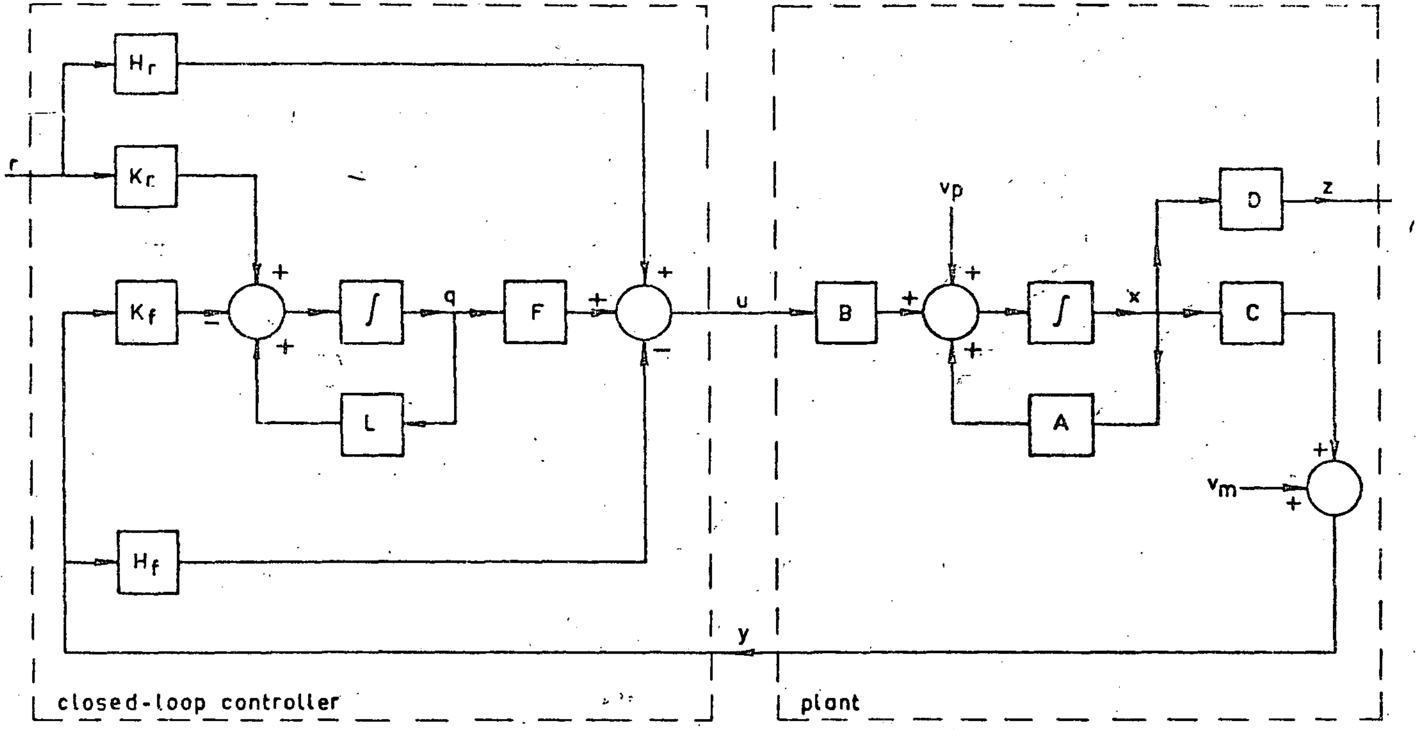


Fig. 2.7. A closed-loop control system.

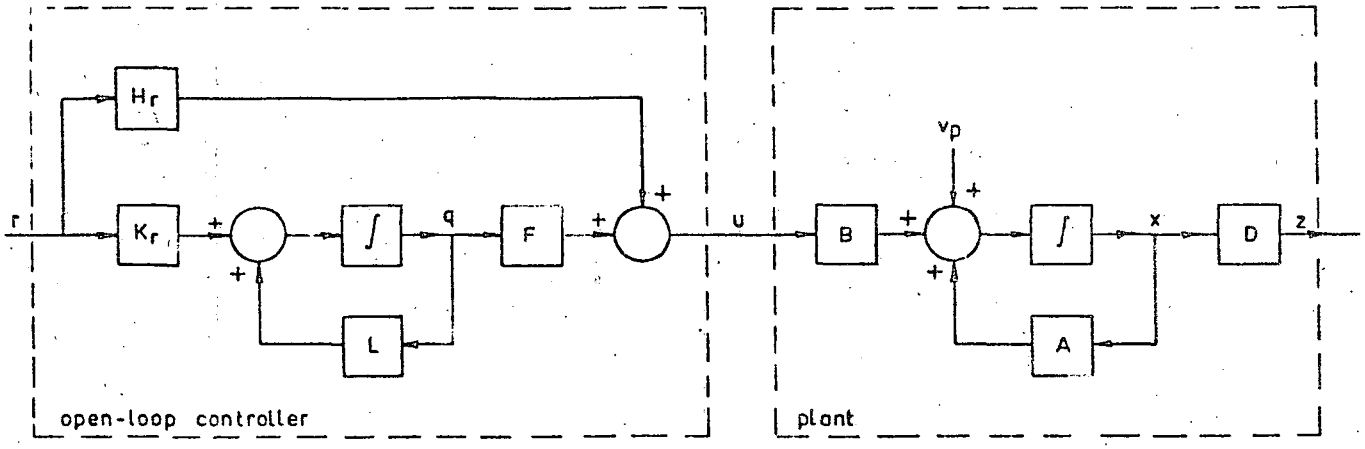


Fig. 2.8. An open-loop control system.

MODELO DE UN SISTEMA: ECUACION DE ESTADO.

Antes de enunciar formalmente el problema de control óptimo, estudiaremos una forma general en la cual puede expresarse el modelo matemático de cualquier sistema. Esta forma es un modelo en variables de estado:

$$\underline{\dot{x}}(t) = \underline{f}(\underline{x}(t), \underline{u}(t), t)$$

$\underline{x}(t)$ es un vector de n dimensiones (llamado vector de estado)

$\underline{u}(t)$ es un vector de m dimensiones, que incluye todas las entradas del sistema

$\underline{f}$ es una función vectorial.

El estado $\underline{x}(t)$ es tal que, conociendo el estado inicial $\underline{x}(t_0)$ y las entradas para tiempos posteriores, $\underline{u}(t)$, $t \geq t_0$, es posible determinar la salida del sistema para todo tiempo posterior a t_0

Si el modelo del sistema es lineal y además invariable con el tiempo, la ecuación de estado toma la forma

$$\dot{\underline{x}}(t) = A\,\underline{x}(t) + \underline{B}\,u(t)$$

siendo A y B matrices constantes, de orden nxn y nxm, respectivamente.

La salida del sistema, $\underline{y}(t)$, estará dada, en forma general, por

$$\underline{y}(t) = \underline{g}(x(t), u(t), t)$$

y, si el modelo es lineal y no varía con el tiempo,

$$\underline{y}(t) = C\,\underline{x}(t) + D\,\underline{u}(t)$$

Note que $\underset{\sim}{y}(t)$ es también un vector (en general de r dimensiones), lo cual implica que se está considerando la posibilidad de que existan mas de una variables consideradas como salidas del sistema. Esto se conoce como un <u>sistema multivariable</u>.

La dimensión de las matrices C y D será entonces $r \times n$ y $r \times m$, respectivamente.

Para la construcción de estos modelos en variables de estado, véanse los ejemplos adjuntos

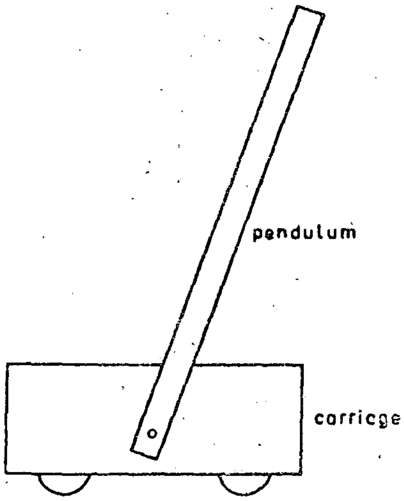


Fig. 1.1. An inverted pendulum positioning system.

Example 1.1. *Inverted pendulum positioning system.*

Consider the inverted pendulum of Figure 1.1 (see also, for this example, Cannon, 1967; Elgerd, 1967). The pivot of the pendulum is mounted on a carriage which can move in a horizontal direction. The carriage is driven by a small motor that at time t exerts a force $\mu(t)$ on the carriage. This force is the input variable to the system.

Figure 1.2 indicates the forces and the displacements. The displacement of the pivot at time t is $s(t)$, while the angular rotation at time t of the pendulum is $\phi(t)$. The mass of the pendulum is m, the distance from the pivot to the center of gravity L, and the moment of inertia with respect to the center of gravity J. The carriage has mass M. The forces exerted on the pendulum are

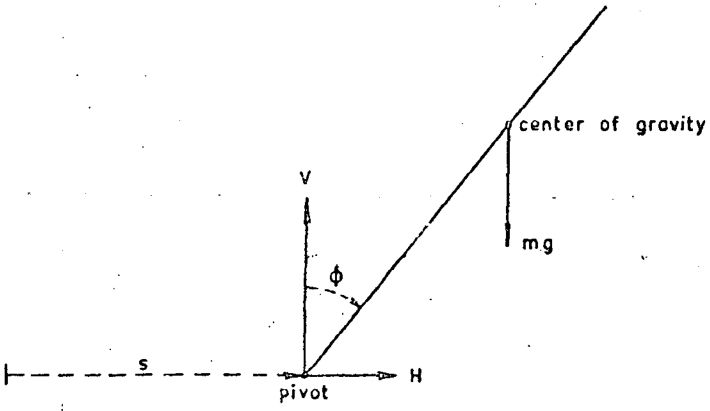


Fig. 1.2. Inverted pendulum: forces and displacements.

the force mg in the center of gravity, a horizontal reaction force $H(t)$, and a vertical reaction force $V(t)$ in the pivot. Here g is the gravitational acceleration. The following equations hold for the system:

$$m\frac{d^2}{dt^2}[s(t) + L\sin\phi(t)] = H(t), \tag{1-11}$$

$$m\frac{d^2}{dt^2}[L\cos\phi(t)] = V(t) - mg, \tag{1-12}$$

$$J\frac{d^2\phi(t)}{dt^2} = LV(t)\sin\phi(t) - LH(t)\cos\phi(t), \tag{1-13}$$

$$M\frac{d^2s(t)}{dt^2} = \mu(t) - H(t) - F\frac{ds(t)}{dt}. \tag{1-14}$$

Friction is accounted for only in the motion of the carriage and not at the pivot; in **1-14**, F represents the friction coefficient. Performing the differentiations indicated in **1-11** and **1-12**, we obtain

$$m\ddot{s}(t) + mL\ddot{\phi}(t)\cos\phi(t) - mL\dot{\phi}^2(t)\sin\phi(t) = H(t), \tag{1-15}$$

$$-mL\ddot{\phi}(t)\sin\phi(t) - mL\dot{\phi}^2(t)\cos\phi(t) = V(t) - mg, \tag{1-16}$$

$$J\ddot{\phi}(t) = LV(t)\sin\phi(t) - LH(t)\cos\phi(t), \tag{1-17}$$

$$M\ddot{s}(t) = \mu(t) - H(t) - F\dot{s}(t). \tag{1-18}$$

To simplify the equations we assume that m is small with respect to M and therefore neglect the horizontal reaction force $H(t)$ on the motion of the carriage. This allows us to replace **1-18** with

$$M\ddot{s}(t) = \mu(t) - F\dot{s}(t). \tag{1-19}$$

Elimination of $H(t)$ and $V(t)$ from **1-15**, **1-16**, and **1-17** yields

$$(J + mL^2)\ddot{\phi}(t) - mgL\sin\phi(t) + mL\ddot{s}(t)\cos\phi(t) = 0. \tag{1-20}$$

Division of this equation by $J + mL^2$ yields

$$\ddot{\phi}(t) - \frac{g}{L'}\sin\phi(t) + \frac{1}{L'}\ddot{s}(t)\cos\phi(t) = 0, \tag{1-21}$$

where

$$L' = \frac{J + mL^2}{mL} \tag{1-22}$$

This quantity has the significance of "effective pendulum length" since a mathematical pendulum of length L' would also yield **1-21**.

Let us choose as the nominal solution $\mu(t) \equiv 0$, $s(t) \equiv 0$, $\phi(t) \equiv 0$. Linearization can easily be performed by using Taylor series expansions for $\sin \phi(t)$ and $\cos \phi(t)$ in **1-21** and retaining only the first term of the series. This yields the linearized version of **1-21**:

$$\ddot{\phi}(t) - \frac{g}{L'}\phi(t) + \frac{1}{L'}\ddot{s}(t) = 0. \qquad \textbf{1-23}$$

We choose the components of the state $x(t)$ as

$$\begin{aligned} \xi_1(t) &= s(t), \\ \xi_2(t) &= \dot{s}(t), \\ \xi_3(t) &= s(t) + L'\phi(t), \\ \xi_4(t) &= \dot{s}(t) + L'\dot{\phi}(t). \end{aligned} \qquad \textbf{1-24}$$

The third component of the state represents a linearized approximation to the displacement of a point of the pendulum at a distance L' from the pivot. We refer to $\xi_3(t)$ as the displacement of the pendulum. With these definitions we find from **1-19** and **1-23** the linearized state differential equation

$$\begin{aligned} \dot{\xi}_1(t) &= \xi_2(t), \\ \dot{\xi}_2(t) &= \frac{1}{M}\mu(t) - \frac{F}{M}\xi_2(t), \\ \dot{\xi}_3(t) &= \xi_4(t), \\ \dot{\xi}_4(t) &= g\phi(t) = \frac{g}{L'}[\xi_3(t) - \xi_1(t)]. \end{aligned} \qquad \textbf{1-25}$$

In vector notation we write

$$\dot{x}(t) = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{F}{M} & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -\frac{g}{L'} & 0 & \frac{g}{L'} & 0 \end{pmatrix} x(t) + \begin{pmatrix} 0 \\ \frac{1}{M} \\ 0 \\ 0 \end{pmatrix} \mu(t), \qquad \textbf{1-26}$$

where $x(t) = \text{col}\,[\xi_1(t), \xi_2(t), \xi_3(t), \xi_4(t)]$.

Later the following numerical values are used:

$$\frac{F}{M} = 1\ \mathrm{s}^{-1},$$
$$\frac{1}{M} = 1\ \mathrm{kg}^{-1},$$
$$\frac{g}{L'} = 11.65\ \mathrm{s}^{-2}, \qquad \text{1-27}$$
$$L' = 0.842\ \mathrm{m}.$$

Example 1.2. *A stirred tank.*

As a further example we treat a system that is to some extent typical of process control systems. Consider the stirred tank of Fig. 1.3. The tank is fed

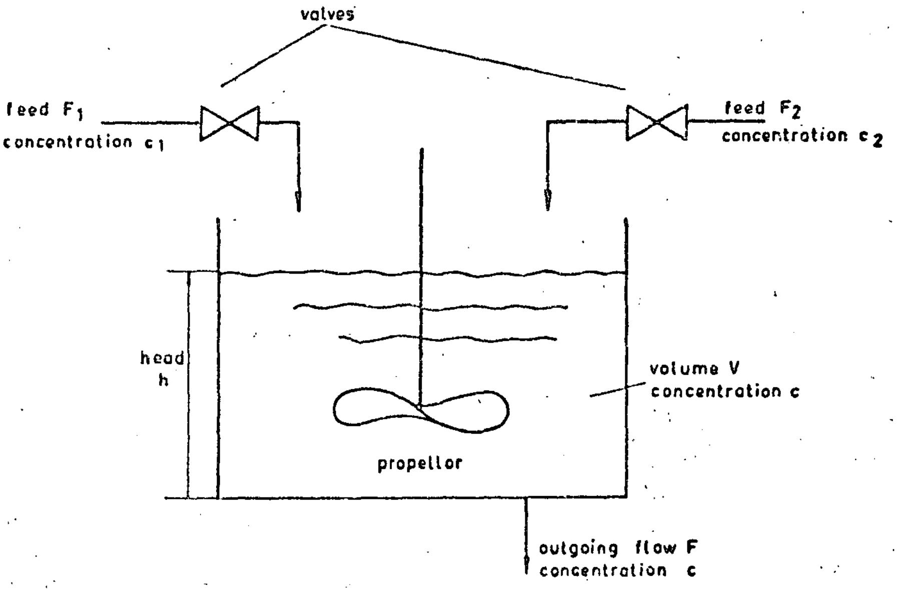


Fig. 1.3. A stirred tank.

with two incoming flows with time-varying flow rates $F_1(t)$ and $F_2(t)$. Both feeds contain dissolved material with constant concentrations c_1 and c_2, respectively. The outgoing flow has a flow rate $F(t)$. It is assumed that the tank is stirred well so that the concentration of the outgoing flow equals the concentration $c(t)$ in the tank.

The mass balance equations are

$$\frac{dV(t)}{dt} = F_1(t) + F_2(t) - F(t), \tag{1-28}$$

$$\frac{d}{dt}[c(t)V(t)] = c_1F_1(t) + c_2F_2(t) - c(t)F(t), \tag{1-29}$$

where $V(t)$ is the volume of the fluid in the tank. The outgoing flow rate $F(t)$ depends upon the head $h(t)$ as follows

$$F(t) = k\sqrt{h(t)}, \tag{1-30}$$

where k is an experimental constant. If the tank has constant cross-sectional area S, we can write

$$F(t) = k\sqrt{\frac{V(t)}{S}}, \tag{1-31}$$

so that the mass balance equations are

$$\frac{dV(t)}{dt} = F_1(t) + F_2(t) - k\sqrt{\frac{V(t)}{S}}, \tag{1-32}$$

$$\frac{d}{dt}[c(t)V(t)] = c_1F_1(t) + c_2F_2(t) - c(t)k\sqrt{\frac{V(t)}{S}}. \tag{1-33}$$

Let us first consider a steady-state situation where all quantities are constant, say F_{10}, F_{20}, and F_0 for the flow rates, V_0 for the volume, and c_0 for the concentration in the tank. Then the following relations hold:

$$0 = F_{10} + F_{20} - F_0, \tag{1-34}$$

$$0 = c_1F_{10} + c_2F_{20} - c_0F_0, \tag{1-35}$$

$$F_0 = k\sqrt{\frac{V_0}{S}}. \tag{1-36}$$

For given F_{10} and F_{20}, these equations can be solved for F_0, V_0, and c_0. Let us now assume that only small deviations from steady-state conditions occur. We write

$$\begin{aligned} F_1(t) &= F_{10} + \mu_1(t), \\ F_2(t) &= F_{20} + \mu_2(t), \\ V(t) &= V_0 + \xi_1(t), \\ c(t) &= c_0 + \xi_2(t), \end{aligned} \tag{1-37}$$

where we consider μ_1 and μ_2 input variables and ξ_1 and ξ_2 state variables. By assuming that these four quantities are small, linearization of **1-32** and **1-33** gives

$$\dot{\xi}_1(t) = \mu_1(t) + \mu_2(t) - \frac{k}{2V_0}\sqrt{\frac{V_0}{S}}\,\xi_1(t), \tag{1-38}$$

$$\dot{\xi}_2(t)V_0 + c_0\dot{\xi}_1(t) = c_1\mu_1(t) + c_2\mu_2(t) - c_0\frac{k}{2V_0}\sqrt{\frac{V_0}{S}}\,\xi_1(t) - k\sqrt{\frac{V_0}{S}}\,\xi_2(t). \tag{1-39}$$

Substitution of **1-36** into these equations yields

$$\dot{\xi}_1(t) = \mu_1(t) + \mu_2(t) - \frac{1}{2}\frac{F_0}{V_0}\,\xi_1(t), \tag{1-40}$$

$$\dot{\xi}_2(t)V_0 + c_0\dot{\xi}_1(t) = c_1\mu_1(t) + c_2\mu_2(t) - \frac{1}{2}c_0\frac{F_0}{V_0}\,\xi_1(t) - F_0\xi_2(t). \tag{1-41}$$

We define

$$\frac{V_0}{F_0} = \theta, \tag{1-42}$$

and refer to θ as the *holdup time* of the tank. Elimination of $\dot{\xi}_1$ from **1-41** results in the linearized state differential equation

$$\dot{x}(t) = \begin{pmatrix} -\dfrac{1}{2\theta} & 0 \\ 0 & -\dfrac{1}{\theta} \end{pmatrix} x(t) + \begin{pmatrix} 1 & 1 \\ \dfrac{c_1 - c_0}{V_0} & \dfrac{c_2 - c_0}{V_0} \end{pmatrix} u(t), \tag{1-43}$$

where $x(t) = \text{col}\,[\xi_1(t), \xi_2(t)]$ and $u(t) = \text{col}\,[\mu_1(t), \mu_2(t)]$. If we moreover define the output variables

$$\begin{aligned} \eta_1(t) &= F(t) - F_0 \simeq \frac{1}{2}\frac{F_0}{V_0}\,\xi_1(t) = \frac{1}{2\theta}\,\xi_1(t), \\ \eta_2(t) &= c(t) - c_0 = \xi_2(t), \end{aligned} \tag{1-44}$$

we can complement **1-43** with the linearized output equation

$$y(t) = \begin{pmatrix} \dfrac{1}{2\theta} & 0 \\ 0 & 1 \end{pmatrix} x(t), \tag{1-45}$$

where $y(t) = \text{col}\,[\eta_1(t), \eta_2(t)]$. We use the following numerical values:

$$\begin{aligned} F_{10} &= 0.015 \text{ m}^3/\text{s}, \\ F_{20} &= 0.005 \text{ m}^3/\text{s}, \\ F_0 &= 0.02 \text{ m}^3/\text{s}, \\ c_1 &= 1 \text{ kmol/m}^3, \\ c_2 &= 2 \text{ kmol/m}^3, \\ c_0 &= 1.25 \text{ kmol/m}^3, \\ V_0 &= 1 \text{ m}^3, \\ \theta &= 50 \text{ s}. \end{aligned} \qquad \text{1-46}$$

This results in the linearized system equations

$$\begin{aligned} \dot{x}(t) &= \begin{pmatrix} -0.01 & 0 \\ 0 & -0.02 \end{pmatrix} x(t) + \begin{pmatrix} 1 & 1 \\ -0.25 & 0.75 \end{pmatrix} u(t), \\ y(t) &= \begin{pmatrix} 0.01 & 0 \\ 0 & 1 \end{pmatrix} x(t). \end{aligned} \qquad \text{1-47}$$

1.2.4 State Transformations

As we shall see, it is sometimes useful to employ a transformed representation of the state. In this section we briefly review linear state transformations for time-invariant linear differential systems. Consider the linear time-invariant system

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t). \end{aligned} \qquad \text{1-48}$$

Let us define a transformed state variable

$$x'(t) = Tx(t), \qquad \text{1-49}$$

where T is a constant, nonsingular transformation matrix. Substitution of $x(t) = T^{-1}x'(t)$ into 1-48 yields

$$\begin{aligned} T^{-1}\dot{x}'(t) &= AT^{-1}x'(t) + Bu(t), \\ y(t) &= CT^{-1}x'(t), \end{aligned} \qquad \text{1-50}$$

or

$$\begin{aligned} \dot{x}'(t) &= TAT^{-1}x'(t) + TBu(t), \\ y(t) &= CT^{-1}x'(t). \end{aligned} \qquad \text{1-51}$$

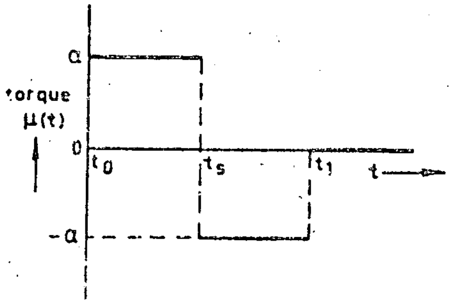


Fig. 1.12. Input torque for satellite repositioning.

(e) Consider the problem of rotating the satellite from one position in which it is at rest to another position, where it is at rest. In terms of the state, this means that the system must be transferred from the state $x(t_0) =$ col $(\phi_0, 0)$ to the state $x(t_1) =$ col $(\phi_1, 0)$, where ϕ_0 and ϕ_1 are given angles. Suppose that two gas jets are available; they produce torques in opposite directions such that the input variable assumes only the values $-\alpha$, 0, and $+\alpha$, where α is a fixed, given number. Show that the satellite can be rotated with an input of the form as sketched in Fig. 1.12. Calculate the switching time t_s and the terminal time t_1. Sketch the trajectory of the state in the state plane.

1.2. *Amplidyne*

An amplidyne is an electric machine used to control a large dc power through a small dc voltage. Figure 1.13 gives a simplified representation (D'Azzo and Houpis, 1966). The two armatures are rotated at a constant speed (in fact they are combined on a single shaft). The output voltage of each armature is proportional to the corresponding field current. Let L_1 and R_1 denote the inductance and resistance of the first field windings and L_2 and R_2 those of the first armature windings together with the second field windings.

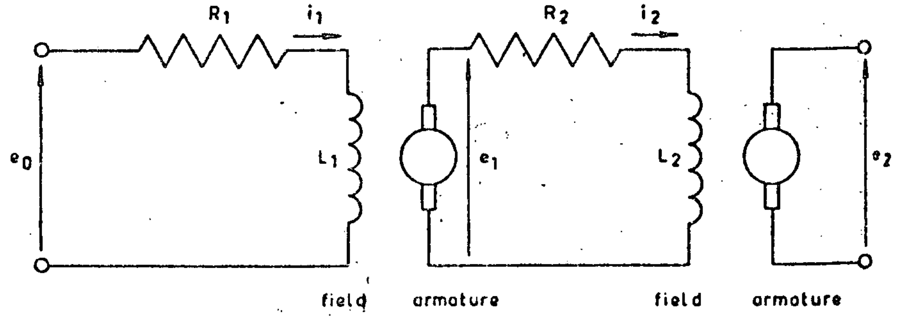


Fig. 1.13. Schematic representation of an amplidyne.

The induced voltages are given by

$$\begin{aligned} e_1 &= k_1 i_1, \\ e_2 &= k_2 i_2. \end{aligned} \qquad \text{1-576}$$

The following numerical values are used:

$$\begin{aligned} R_1/L_1 &= 10\ \mathrm{s}^{-1}, & R_2/L_2 &= 1\ \mathrm{s}^{-1}, \\ R_1 &= 5\ \Omega, & R_2 &= 10\ \Omega, \\ k_1 &= 20\ \mathrm{V/A}, & k_2 &= 50\ \mathrm{V/A}. \end{aligned} \qquad \text{1-577}$$

(a) Take as the components of the state $\xi_1(t) = i_1(t)$ and $\xi_2(t) = i_2(t)$ and show that the system equations are

$$\begin{aligned} \dot{x}(t) &= \begin{pmatrix} -\dfrac{R_1}{L_1} & 0 \\ \dfrac{k_1}{L_2} & -\dfrac{R_2}{L_2} \end{pmatrix} x(t) + \begin{pmatrix} \dfrac{1}{L_1} \\ 0 \end{pmatrix} \mu(t), \\ \eta(t) &= (0, \quad k_2) x(t), \end{aligned} \qquad \text{1-578}$$

where $\mu(t) = e_0(t)$ and $\eta(t) = e_2(t)$.

(b) Compute the transition matrix, the impulse response function, and the step response function of the system. Sketch for the numerical values given the impulse and step response functions.

(c) Is the system stable in the sense of Lyapunov? Is it asymptotically stable?

(d) Determine the transfer function of the system. For the numerical values given, sketch a Bode plot of the frequency response function of the system.

(e) Compute the modes of the system.

1.3. *Properties of time-invariant systems under state transformations*

Consider the linear time-invariant system

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t). \end{aligned} \qquad \text{1-579}$$

We consider the effects of the state transformation $x' = Tx$.

(a) Show that the transition matrix $\Phi(t, t_0)$ of the system 1-579 and the transition matrix $\Phi'(t_1, t_0)$ of the transformed system are related by

$$\Phi'(t, t_0) = T\Phi(t, t_0)T^{-1}. \qquad \text{1-580}$$

(b) Show that the impulse response matrix and the step response matrix of the system do not change under a state transformation.

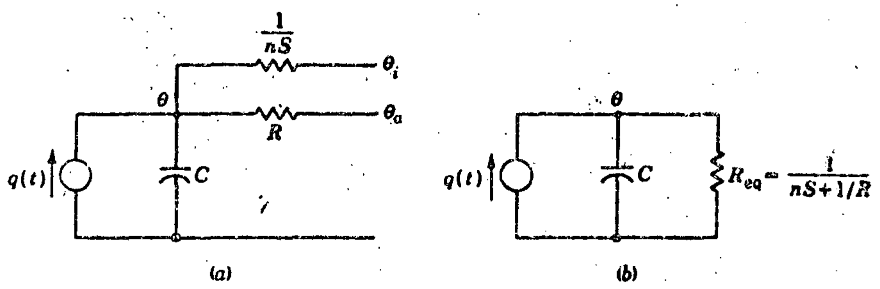


FIGURE 2-27
(*a*) Thermal network of an electric water heater; (*b*) simplified network.

2-10 HYDRAULIC LINEAR ACTUATOR

The valve-controlled hydraulic actuator is used in many applications as a power amplifier. Very little power is required to position the valve, but a large power output is controlled. The hydraulic unit is relatively small, which makes its use very attractive. Figure 2-28 shows a simple hydraulic actuator in which motion of the valve regulates the flow of oil to either side of the main cylinder. An input motion x of a few thousandths of an inch results in a large change of oil flow. The resulting difference in pressure on the main piston causes motion of the output shaft. The oil flowing in is supplied by a source which maintains a constant high pressure P_h, and the oil on the opposite side of the piston flows into the drain at low pressure P_s. The load-induced pressure P_L is the difference between the pressures on each side of the main piston:

$$P_L = P_1 - P_2 \tag{2-96}$$

The flow of fluid through an inlet orifice is given by[10]

$$q = ca\sqrt{2g\frac{\Delta p}{\omega}} \tag{2-97}$$

where c = orifice coefficient
a = orifice area
w = specific weight of fluid
Δp = pressure drop across orifice
g = gravitational acceleration constant
q = rate of flow of fluid

Simplified Analysis

As a first-order approximation, it can be assumed that the orifice coefficient and the pressure drop across the orifice are constant and independent of valve position. Also, the orifice area can be expressed in terms of the valve displacement x. Equation (2-97), which gives the rate of flow of hydraulic fluid through the valve, can be rewritten as

$$q = C_x x \tag{2-98}$$

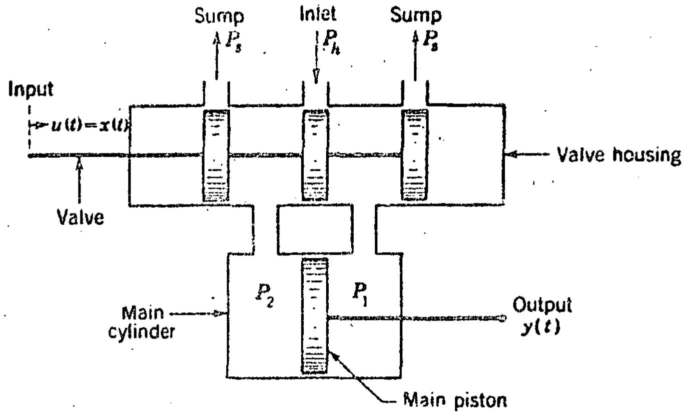


FIGURE 2-28
Hydraulic actuator.

where x is the displacement of the valve. The displacement of the main piston is directly proportional to the flow of fluid into the main cylinder. By neglecting the compressibility of the fluid and the leakage around the valve and main piston, the equation of motion of the main piston is

$$q = C_b\,Dy \tag{2-99}$$

Combining the two equations gives

$$Dy = \frac{C_x}{C_b}x = C_1x \tag{2-100}$$

This analysis is essentially correct when the load reaction is small.

More Complete Analysis

When the load reaction is not negligible, a more complete analysis should take into account the pressure drop across the orifice, the leakage of oil around the piston, and the compressibility of the oil.

The pressure drop Δp across the orifice is a function of the source pressure P_h and the load pressure P_L. Since P_h is assumed constant, the flow equation is a function of valve displacement x and load pressure P_L:

$$q = f(x,P_L) \tag{2-101}$$

The differential dq, expressed in terms of partial derivatives, is

$$dq = \frac{\partial q}{\partial x}\,dx + \frac{\partial q}{\partial P_L}\,dP_L \tag{2-102}$$

If q, x, and P_L are measured from zero values as reference points, and if the partial derivatives are constant at the values they have at zero, the integration of Eq. (2-102) gives

$$q = \left(\frac{\partial q}{\partial x}\right)_0 x + \left(\frac{\partial q}{\partial P_L}\right)_0 P_L \tag{2-103}$$

By defining

$$C_x \equiv \left(\frac{\partial q}{\partial x}\right)_0 \quad \text{and} \quad C_p \equiv \left(\frac{-\partial q}{\partial P_L}\right)_0$$

the flow equation for fluid entering the main cylinder can be written as

$$q = C_x x - C_p P_L \quad (2\text{-}104)$$

Both C_x and C_p have positive values. A comparison with Eq. (2-98) shows that the load pressure reduces the flow into the main cylinder. The flow of fluid into the cylinder must satisfy the continuity conditions of equilibrium. This flow is equal to the sum of the components:

$$q = q_o + q_l + q_c \quad (2\text{-}105)$$

where q_o = incompressible component (causes motion of piston)
q_l = leakage component
q_c = compressible component

The component q_o, which produces a motion y of the main piston, is

$$q_o = C_b\, Dy \quad (2\text{-}106)$$

The compressible component is derived in terms of the bulk modulus of elasticity, which is defined as the ratio of incremental stress to incremental strain. Thus

$$K_B = \frac{\Delta P_L}{\Delta V/V}$$

Solving for ΔV and dividing both sides of the equation by Δt gives

$$\frac{\Delta V}{\Delta t} = \frac{V}{K_B}\frac{\Delta P_L}{\Delta t}$$

Taking the limit as Δ approaches zero and letting $q_c = dV/dt$ gives

$$q_c = \frac{V}{K_B} DP_L \quad (2\text{-}107)$$

where V is the effective volume of fluid under compression and K_B is the bulk modulus of the hydraulic oil. The volume V at the middle position of the piston stroke is often used in order to linearize the differential equation.

The leakage component is

$$q_l = LP_L \quad (2\text{-}108)$$

where L is the leakage coefficient of the whole system.

Combining these equations gives

$$q = C_x x - C_p P_L = C_b\, Dy + \frac{V}{K_B} DP_L + LP_L \quad (2\text{-}109)$$

and rearranging terms gives

$$C_b\, Dy + \frac{V}{K_B} DP_L + (L + C_p)P_L = C_x x \quad (2\text{-}110)$$

32

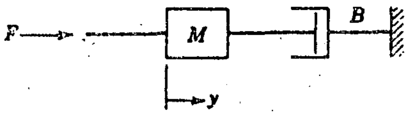


FIGURE 2-29
Load on a hydraulic piston.

The force developed by the main piston is

$$F = n_F A P_L = C P_L \qquad (2\text{-}111)$$

where η_F is the force conversion efficiency of the unit and A is the area of the main actuator piston.

An example of a specific type of load consisting of a mass and a dashpot is shown in Fig. 2-29. The equation for this system is obtained by equating the force produced by the piston, which is given by Eq. (2-111) to the reactive load forces:

$$F = M\,D^2y + B\,Dy = C P_L \qquad (2\text{-}112)$$

Substituting the value of P_L from Eq. (2-112) into Eq. (2-110) gives the equation relating the input motion x to the response y:

$$\frac{MV}{CK_B}D^3y + \left[\frac{BV}{CK_B} + \frac{M}{C}(L + C_p)\right]D^2y + \left[C_b + \frac{B}{C}(L + C_p)\right]Dy = C_x x \qquad (2\text{-}113)$$

The analysis above is based on perturbations about the reference set of values $x = 0$, $q = 0$, $P_L = 0$. For the entire range of motion x of the valve, the quantities $\partial q/\partial x$ and $-\partial q/\partial P_L$ can be determined experimentally. Although they are not constant at values equal to the values C_x and C_p at the zero reference point, average values can be assumed in order to simulate the system by linear equations. For conservative design the volume V is determined for the main piston at the midpoint.

To write the state equation for the hydraulic actuator and load of Figs. 2-28 and 2-29 the energy-related variables must be determined. The mass M yields one energy-storage variable, the output velocity Dy. The compressible component q_c represents an energy-storage element in a hydraulic system. The compression of a fluid produces stored energy, just as in the compression of a spring. The equation for hydraulic energy is

$$E(t) = \int_0^t P(\tau)q(\tau)\,d\tau \qquad (2\text{-}114)$$

where $P(\tau)$ is the pressure and $q(\tau)$ is the rate of flow of fluid. The energy storage in a compressed fluid is obtained in terms of the bulk modulus of elasticity K_B. Combining Eq. (2-107) with Eq. (2-114) for a constant volume yields

$$E_c(P_L) = \int_0^{P_L} \frac{V}{K_B} P_L\,dP_L = \frac{V}{2K_B} P_L^2 \qquad (2\text{-}115)$$

The stored energy in a compressed fluid is proportional to the pressure P_L squared; thus P_L may be used as a physical state variable.

Since the output quantity in this system is the position y, it is necessary to increase the state variables to three. Further evidence of the need for three state variables is

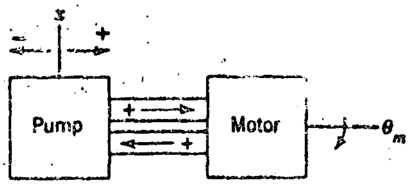


FIGURE 2-30
Hydraulic transmission.

the fact that Eq. (2-113) is a third-order equation. Therefore, in this example, let $x_1 = y, x_2 = Dy = \dot{x}_1, x_3 = P_L$, and $u = x$. Then, from Eqs. (2-110) and (2-112), the state and output equations are

$$\dot{\mathbf{x}} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -\frac{B}{M} & \frac{C}{M} \\ 0 & -\frac{C_b K_B}{V} & -\frac{K_B(L + C_p)}{V} \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0 \\ 0 \\ \frac{K_B C_x}{V} \end{bmatrix} u \qquad (2\text{-}116)$$

$$y = [1 \quad 0 \quad 0]\mathbf{x} = x_1 \qquad (2\text{-}117)$$

The effect of augmenting the state variables by adding the piston displacement $x_1 = y$ is to produce a singular system; that is, $|\mathbf{A}| = 0$. This property does not appear if a spring is added to the load, as shown in Prob. 2-10. In that case $x_1 = y$ is an independent state variable.

2-11 POSITIVE-DISPLACEMENT ROTATIONAL HYDRAULIC TRANSMISSION[11]

When a large torque is required in a control device, it is possible to use a hydraulic transmission. The transmission contains a variable-displacement pump driven at constant speed. It pumps a quantity of oil that is proportional to a control stroke and independent of back pressure. The direction of fluid flow is determined by the direction of displacement of the control stroke. The hydraulic motor has an angular velocity proportional to the volumetric flow rate and in the direction of the oil flow from the pump.

The assumption is made that over a limited range of operation the hydraulic transmission is linear. A schematic picture of the system is shown in Fig. 2-30.

The following symbols are used:

q_p = total volumetric flow rate from pump
q_m = volumetric flow rate through motor
q_l = volumetric leakage flow rate of both pump and motor
q_c = compressibility flow rate
x = control stroke (x varies from 0 to ±1)
ω_p = angular velocity of pump shaft (constant)
ω_m = angular velocity of motor shaft (variable)

To illustrate these concepts, suppose we consider the description of the reactor in Fig. 9-2. The reactor is a well-mixed, continuous flow unit in which the second-order reaction

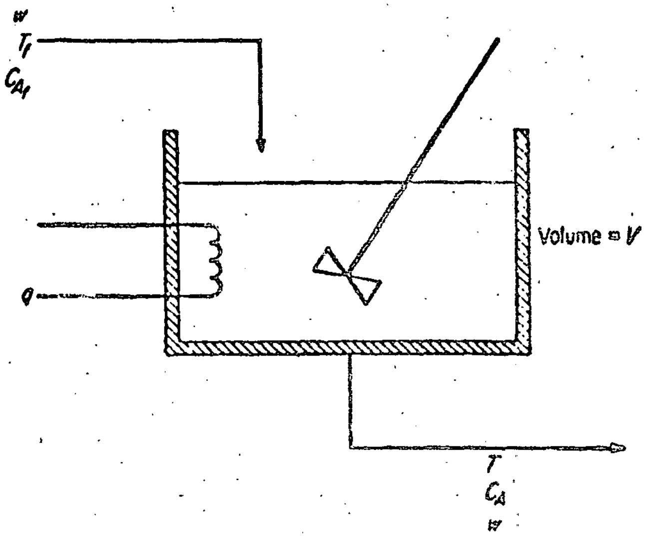


FIG. 9-2. Stirred chemical reactor.

$$2A \longrightarrow B$$

occurs. For simplicity, the rate constant k is assumed to be independent of temperature. The heat of reaction ΔH is based on one mole of A consumed. Making a heat and material balance over the reactor gives the following equations:

$$\frac{dT}{dt} = \frac{w}{V\rho}(T_f - T) + \frac{q}{V\rho c_p} - \frac{(\Delta H)kC_A{}^2}{\rho c_p} \tag{9-5}$$

$$\frac{dC_A}{dt} = \frac{w}{V\rho}(C_{A_f} - C_A) - kC_A{}^2 \tag{9-6}$$

The state variables for this system would be the reactor temperature T and concentration C_A, i.e.,

$$\mathbf{x}(t) = \begin{bmatrix} T \\ C_A \end{bmatrix} \tag{9-7}$$

As manipulated inputs, the reactor feed rate w and rate of heat input q are logical selections. Thus the vector $\mathbf{u}(t)$ is

$$\mathbf{u}(t) = \begin{bmatrix} q \\ w \end{bmatrix} \tag{9-8}$$

The functions f_1 and f_2 in the state Eq. 9-2 become the right-hand sides of Eqs. 9-5 and 9-6.

These equations are of course nonlinear. As usual, a linear set would be much more convenient. Such equations can be obtained by linearizing Eqs. 9-5 and 9-6 about an equilibrium point $\overline{T}$, $\overline{C}_A$, $\overline{w}$, and $\overline{q}$:

$$\frac{d\hat{T}}{dt} = -\frac{\overline{w}}{V\rho}\hat{T} + \frac{T_f - \overline{T}}{V\rho}\hat{w} + \frac{\hat{q}}{V\rho c_p} - \frac{2(\Delta H)k\overline{C}_A}{\rho c_p}\hat{C}_A \tag{9-9a}$$

$$\frac{d\hat{C}_A}{dt} = -\frac{\overline{w}}{V\rho}\hat{C}_A + \frac{C_{A_f} - \overline{C}_A}{V\rho}\hat{w} - 2k\overline{C}_A\hat{C}_A \tag{9-9b}$$

where $\hat{C}_A = C_A - \overline{C}_A$
$\hat{T} = T - \overline{T}$
$\hat{w} = w - \overline{w}$
$\hat{q} = q - \overline{q}$

Equation 9-9 may be conveniently represented by the following matrix differential equation:

$$\mathbf{x}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \tag{9-10}$$

where

$$\mathbf{x}(t) = \begin{bmatrix} \hat{T} \\ \hat{C}_A \end{bmatrix} \quad \text{state vector}$$

$$\mathbf{u}(t) = \begin{bmatrix} \hat{q} \\ \hat{w} \end{bmatrix} = \text{manipulated inputs}$$

$$\mathbf{A} = \begin{bmatrix} -\dfrac{\overline{w}}{V\rho} & -\dfrac{2(\Delta H)k\overline{C}_A}{\rho c_p} \\ 0 & -\dfrac{\overline{w}}{V\rho} + 2k\overline{C}_A \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} \dfrac{1}{V\rho c_p} & \dfrac{T_f - \overline{T}}{V\rho} \\ 0 & \dfrac{C_{A_f} - \overline{C}_A}{V\rho} \end{bmatrix}$$

Equation 9-10 is simply a linear version of the state Eq. 9-2.

EL PROBLEMA DE CONTROL OPTIMO.

Determinar el control $\underline{u}(t)$ en el intervalo $t_0 \leq t \leq t_1$ que transfiera al sistema de un estado inicial $\underline{x}_0$ a algún estado final $x(t_1)$, en forma tal que se minimice una funcional especificada J.

La forma general de J será

$$J = \int_{t_0}^{t_1} \mathcal{L}\left[\underline{x}(t), \underline{u}(t)\right] dt$$

siendo $\mathcal{L}$ una función escalar del estado y la entrada

El problema de optimización puede atacarse de tres maneras.

1- Cálculo de variaciones

2- Programación dinámica.

3- Principio mínimo de Pontryagin

La primera forma no ofrece flexibilidad en el tratamiento de restricciones en las variables de estado, por lo que no es de utilidad práctica.

Veremos a continuación la tercera alternativa, recordando que programación dinámica ya se trató en una sesión anterior.
Antes de esto, es necesario definir algunos términos.

Dado $\dot{x}(t) = f(x(t), u(t), t)$

y la funcional

$$J = \int_{t_0}^{t_1} \mathcal{L}[x(t), u(t)]\, dt$$

se define el <u>Hamiltoniano</u> H como

$$H[x(t), p(t), u(t)] = \mathcal{L}[x(t), u(t)] + p^T(t)\, f[x(t), u(t)]$$

(H, $\mathcal{L}$ y f continuas y diferenciables)

donde $p(t)$ se llama <u>vector</u> <u>adjunto</u> o co-estado y satisface

$$\dot{p}(t) = -\frac{\partial H}{\partial x(t)}$$

El estado satisface, a su vez

$$\dot{x}(t) = \frac{\partial H}{\partial p(t)}$$

Trataremos únicamente dos casos de control óptimo:

CASO 1: PUNTO FINAL FIJO

Se trata aquí de especificar el estado final $x(t_1)$, pero dejando a t_1 libre. Un caso especial es cuando se desea llegar a un estado $x(t_1)$ en el mínimo tiempo posible; en cuyo caso la funcional J es simplemente

$$J = \int_{t_0}^{t_1} dt = t_1 - t_0$$

El principio mínimo de Pontryagin establece, para este caso las siguientes condiciones necesarias para minimizar J :

1- $$\dot{p}(t) = -\frac{\partial H}{\partial x(t)}$$

$$\dot{x}(t) = \frac{\partial H}{\partial p(t)}$$

$x(t_0) = x_0$ $x(t_1) = x_1$ (dado)

$p(t_1) = 0$ (t_1 no especificado)

2- $$\frac{\partial H}{\partial u(t)} = 0$$ (tiene un mínimo absoluto)

3- $H \equiv 0$ para $t_0 \leq t \leq t_1$

CASO 2 PUNTO FINAL LIBRE

En este caso no se especifica el estado final $x(t_1)$, sino solamente el tiempo final t_1 (que inclusive puede ser infinito).

J toma generalmente la forma

$$J = \int_{t_0}^{t_1} [x^T Q x + u^T R u] \, dt$$

$Q > 0$ (positiva definida)

$R \geq 0$ (positiva semidefinida)

Las condiciones necesarias definidas por el principio mínimo son las mismas excepto por las condiciones de frontera, que ahora son

$$x(t_0) = X_0$$

$$p(t_1) = 0 \qquad t_1 \text{ especificado}$$

9-6 APPLICATION OF THE MINIMUM PRINCIPLE

The minimum principle is a very powerful and useful tool for determining the optimal control for problems falling into either of the above categories. Its application to process problems is beset by several difficulties. One of these lies with the cost functional. The natural cost functional to propose for process operation is to maximize the return or minimize the loss. However, the mathematical formulation of such a cost functional is not practical under most situations. The alternative generally selected is to substitute a cost functional that should give approximately the same results as one based on economics. The one frequently selected is minimum time.

To justify the reasoning behind this, suppose it is found that the process is currently operating at state $\mathbf{x}_0$. However, for current conditions, the optimal return would be for operation at state $\mathbf{x}_1$. Thus it seems reasonable to propose that the optimal control should transfer the process from $\mathbf{x}_0$ to $\mathbf{x}_1$ as soon as possible; i.e., in minimum time.

A second problem occurs in determining the optimal control from the minimum principle. While the minimum principle applies to nonlinear systems, to constraints on the manipulated variable, and other common complications encountered in process systems, constraints on the state variables, e.g., pressure or temperature limitations, cannot be readily incorporated. Even when these are absent, the computational requirements, especially for nonlinear systems, are considerable, basically due to the split boundary conditions on the canonical equation encountered in both cases considered in the last section.

A third difficulty arises from the fact that the minimum principle as formulated applies to what process engineers typically refer to as the open-loop control problem. That is, the minimum principle gives the control $\mathbf{u}$ as a function of time. For process systems, feedback control, i.e., control in which $\mathbf{u}$ is given as a function of the state $\mathbf{x}$, is almost mandatory due to modeling errors, unknown disturbances, etc. Only for linear cases can a feedback control law be derived from the minimum principle with certainty.

Although these considerations reduce the utility of the minimum principle for process applications, it still offers a definite potential. Most of the above complications can be avoided if the minimum principle is applied to a simple, linear process model. Latour et al. (11) suggest that a model of wide utility is the following:

$$\frac{C(s)}{M(s)} = \frac{K_p e^{-\theta s}(\alpha s + 1)}{(\tau_1 s + 1)(\tau_2 s + 1)} \tag{9-21}$$

where τ_1, τ_2 = time constants
K = process gain
α = reciprocal of process zero
θ = dead time

Processes that can often be adequately represented by this model include extractors, mixing in agitated vessels, heat exchangers, distillation columns, and chemical reactors.

It thus seems reasonable to propose that an optimal control strategy for these units be based upon this model. The control problem is to drive the system from some known initial state $c(0)$, $\dot{c}(0)$ to some known final state $c(T)$, $\dot{c}(T)$ using a control subject to the constraint

$$U_{\min} \leqslant u \leqslant U_{\max}$$

The final time T is to be minimized. This is obviously identical to the "fixed-end-point problem" discussed above.

For the case in which both α and θ in Eq. 9-21 are zero, the control will always be at one of the extremes. For a second-order system, there will be two switches. If we let τ_1 be the larger of the time constants, the optimum switching times for the system are given by the equations in Table 9-1. Note that the equation for t_1 (the time at the first switch) requires an implicit solution. Figure 9-3 shows a typical input and a typical response. Note that t_1 and t_2 are the switching times; r_0 and r are the old and new set points, respectively; and K and k are the upper and lower constraints on the manipulated variable respectively. The response rises quickly but does not overshoot, which is typical of minimum time responses.

The application for which this procedure was proposed is for use in supervisory control. For example, suppose the computer calculates that for optimum operation the set point should be changed from r_0 to r. This transition should be made as follows:

TABLE 9-1

Switching Times for a Second-Order System

(For $r_0 < r$, interchange K and k in the equations below. These equations are specifically for the initial state $c(0) = r_0; \dot{c}(0) = 0$).

$r < r_0$

$0 < \tau_2/\tau_1 \leqslant 0.9$

$$\left[\frac{(K-k)-(r_0/K_p-k)\exp(-t_1/\tau_2)}{(K-r/K_p)}\right]^{\frac{\tau_2}{\tau_1}} = \frac{(K-k)-(r_0/K_p-k)\exp(-t_1/\tau_1)}{(K-r/K_p)}$$

$0.9 < \tau_2/\tau_1 \leqslant 1$

$$\left[\frac{r_0/K_p-k}{K-k}-\exp\left(\frac{t_1}{\tau_2}\right)\right] \ln\left[\frac{(r_0/K_p-k)-(K-k)\exp(t_1/\tau_2)}{r_0/K_p-K}\right] + \frac{t_1}{\tau_2}\exp\left(\frac{t_1}{\tau_2}\right) = 0$$

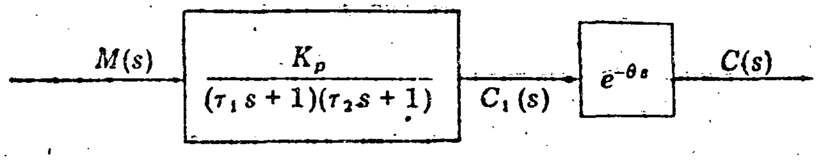
$0 < \tau_2/\tau_1 \leqslant 1$

$$t_2/\tau_2 = \ln\left[\frac{(r_0/K_p-k)-(K-k)\exp(t_1/\tau_2)}{r/K_p-K}\right]$$

1. At time zero, the feedback controller should be placed on manual.
2. The manipulated variable should be switched from maximum to minimum or vice versa as discussed above.
3. At time t_2, the feedback controller should be returned to automatic.

Thus the feedback controller is present to "trim out" any modeling errors, load disturbances, and the like which may cause the optimal control to fall short of its stated objectives.

As for the case in which the process dead time is nonzero, consider the following representation of the process model:



Using the concepts presented above, the control $M(s)$ can be determined to give the optimum response $C_1(s)$ prior to the dead time. However, the dead time simply delays this response by time θ, which is completely independent of $M(s)$. Thus, the response $C(s)$ is optimized when $C_1(s)$ is optimized. In other words, the optimal control for the system with dead time is identical for the same system without the dead time. Note carefully that the above system is open loop, i.e., no feedback. The only modification to the control strategy in this paragraph is that the feedback controller should not be returned to automatic until time t_2 plus θ.

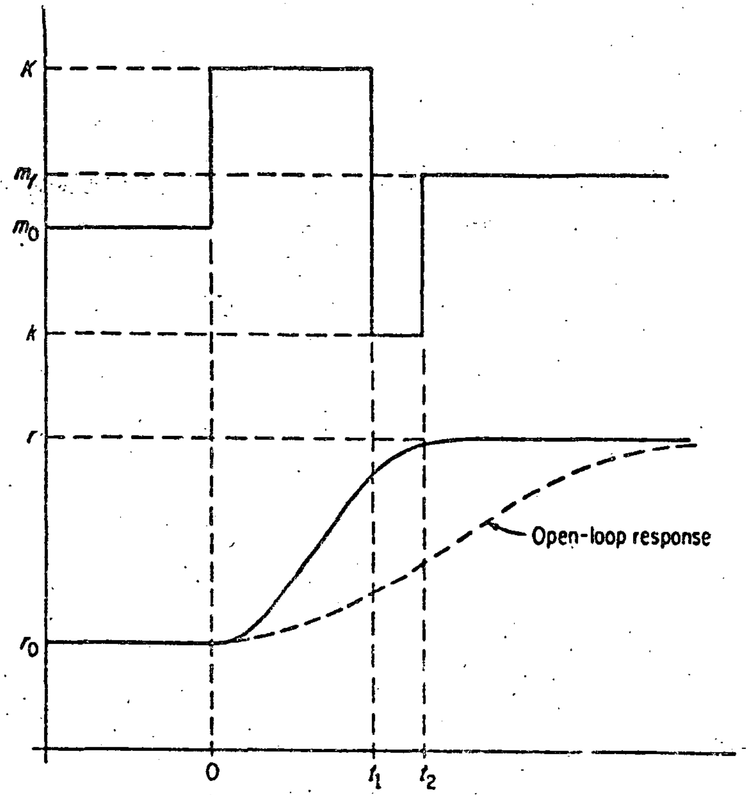


FIG. 9-3. Optimal response to a change in set point.

For cases in which α is not equal to zero, the optimum response is that the manipulated variable should follow a prescribed transient after the initial bang-bang action. As control of this type is difficult to achieve, Latour et al. (11) suggest the use of the same switching times presented above.

As pointed out previously, because of unmeasured load changes or other random disturbances, it is desirable to formulate the control strategy so that it can be implemented in a feedback manner. That is, we determine from the states $c(t)$ and $\dot{c}(t)$ if a switch should be made. This is readily implemented using a switching curve in the c-$\dot{c}$ plane as illustrated (for $\alpha = \theta = 0$) in Fig. 9-4. Note that the state

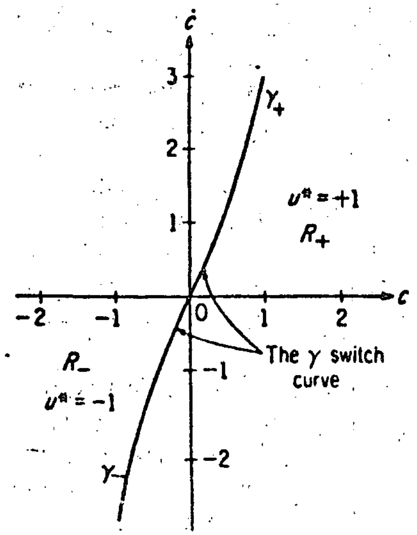


FIG. 9-4. Switching curve for a two-time-constant plant. (Reproduced by permission from M. Athans and P. L. Falb, *Optimal Control*, McGraw-Hill Book Company, New York, 1966.)

$c(t)$, $\dot{c}(t)$ specifies a location in the c-$\dot{c}$ plane. Depending upon the location of this point relative to the switching curve, the control u will be at one of its extremes. The procedure for developing the switching curve for the exact system considered is presented on pages 526–536 in Athan and Falb's book on optimal control (1).

Although a dead time θ in the process has no effect on the switching times, it will change the switching curve. As illustrated in Fig. 9-5, this is because the feedback is not the state vector $\mathbf{x}(t)$, but instead the delayed value $\mathbf{x}(t - \theta)$. Unfortunately, the method customarily used to determine the switching curve does not readily treat dead times in a direct fashion. Moore et al. (12) suggest incorporating the Smith predictor or dead time compensator (discussed in Sec. 8-8) as illustrated in Fig. 9-6a. Since this effectively moves the dead time outside the loop (Fig. 9-6b), the switching curve can be determined directly from the gains and time constants, ignoring the dead time. Note that the model required for the compensator is the same model used to determine the switching curve.

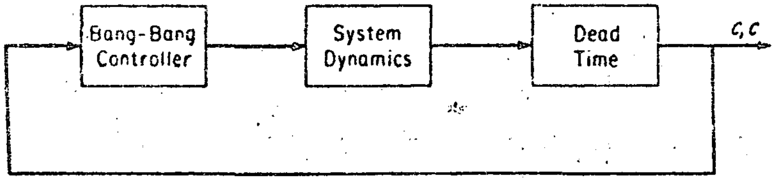
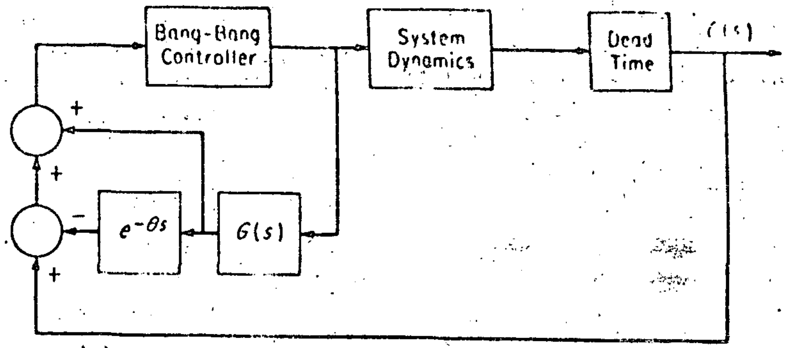
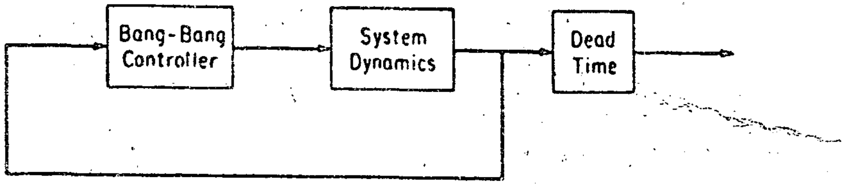


FIG. 9-5. Bang-bang control loop for systems with dead time.



(a) Control loop



(b) Effective control loop for perfect modeling

FIG. 9-6. Bang-bang controller coupled with the dead-time compensator.

9-7 OPTIMAL CONTROL OF LINEAR SYSTEMS USING A QUADRATIC PERFORMANCE CRITERION (1)

This section will consider the optimal control of a linear, time invariant system given by the state equation

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \qquad (9\text{-}22)$$

$$\mathbf{x}(0) = \mathbf{x}_0 \qquad (9\text{-}23)$$

It is desired to control the system in such a manner as to minimize the cost functional

$$J = \frac{1}{2}\int_0^T [\mathbf{x}^T(t)\mathbf{Q}\mathbf{x}(t) + \mathbf{u}^T(t)\mathbf{R}\mathbf{u}(t)]\,dt \qquad (9\text{-}24)$$

This formulation is that of the state regulator problem, since in order to minimize the above cost functional the control will tend to drive

$x(t)$ toward 0. The state equation further indicates that the equilibrium state corresponding to $x(t)$ equal 0 is $u(t)$ equal zero also.

To formulate the optimal control law, we begin by defining the Hamiltonian for this problem.

$$H = \frac{1}{2}x^T(t)Qx(t) + \frac{1}{2}u^T(t)Ru(t) + p^T(t)Ax(t) + p^T(t)Bu(t) \tag{9-25}$$

The equation for the costate is

$$\dot{p}(t) = \frac{\partial H}{\partial x(t)} = -Qx(t) - A^T p(t) \tag{9-26}$$

From the minimum principle, the boundary condition should be

$$p(t) = 0 \tag{9-27}$$

This equation and the state equation 9-22 form the canonical set of equations for this problem.

As presented in detail by Athans and Falb (1), the linearity of the canonical equations can be used to prove that the costate vector $p(t)$ is a linear combination of the state vector $x(t)$, or mathematically,

$$p(t) = K(t)x(t) \tag{9-28}$$

This fact permits a reasonably simple solution to this optimal control problem.

In Sec. 9-3 it was noted that one of the requirements for $u(t)$ to be optimal is that the Hamiltonian be minimized. Taking the partial of Eq. 9-25 and setting to zero gives

$$\frac{\partial H}{\partial u(t)} = Ru(t) + B^T p(t) = 0$$

or

$$u(t) = -R^{-1}B^T p(t) = -R^{-1}B^T K(t)x(t) \tag{9-29}$$

Thus we see that the control is also a linear function of the state, as illustrated by the feedback arrangement in Fig. 9-7.

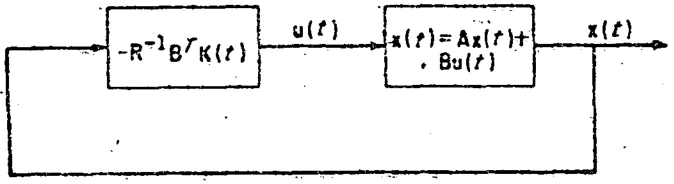


FIG. 9-7. Optimal controller.

Equation 9-29 is also quite suitable for a control law provided $K(t)$ can be evaluated. To develop such a procedure, we begin by taking the derivative of Eq. 9-28:

$$\dot{p}(t) = K(t)\dot{x}(t) + \dot{K}(t)x(t)$$

Substituting Eq. 9-26 for $\dot{p}(t)$ and Eq. 9-22 for $\dot{x}(t)$ followed by Eq. 9-28 for $p(t)$ and Eq. 9-29 for $u(t)$ gives ($K(t)$ is also symmetric):

$$\dot{K}(t) = -K(t)A - A^T K(t) + K(t)BR^{-1}B^T K(t) - Q \tag{9-30}$$

This equation is known as the matrix Riccati equation, and can be solved for $K(t)$ provided a boundary condition is available. From Eqs. 9-27 and 9-28 it is seen that

$$K(T)x(T) = 0$$

Since the final state $x(T)$ is free (can assume any value), it follows that

$$K(T) = 0 \tag{9-31}$$

As this boundary condition is at the final time, Eq. 9-30 must be solved in reverse time to give $K(t)$ over the interval $0 \leqslant t \leqslant T$.

A special case of interest is when $T \longrightarrow \infty$, or the control is over the infinite interval. For this case it can be shown that $K(t)$ is a constant. Consequently, its derivative is zero, reducing Eq. 9-30 to

$$-KA - A^T K + KBR^{-1}B^T K - Q = 0 \tag{9-32}$$

The only difficulty is in solving for K. It turns out that a practical approach is to continue to use the differential equation 9-30 with the boundary condition of 9-31 and solve in reverse time until a "steady state" is reached, at which the value of K will be the solution to Eq. 9-32. This is illustrated in Fig. 9-8.

9-8 OPTIMAL CONTROL FOR SET-POINT CHANGES

The conventional control loop typically considered is illustrated in Fig. 9-9. The normal procedure is to design the controller either to a prescribed change in set point or to a prescribed change in disturbance (load). Unfortunately, the optimal controller as formulated in the previous section does not quite match either of these. Instead, it is designed to take the system from some initial state x_0 to the state 0 in an optimal fashion. In the remainder of this section and the next, we shall discuss the transformation of the conventional control problem into a form to which optimal control theory can be readily applied.

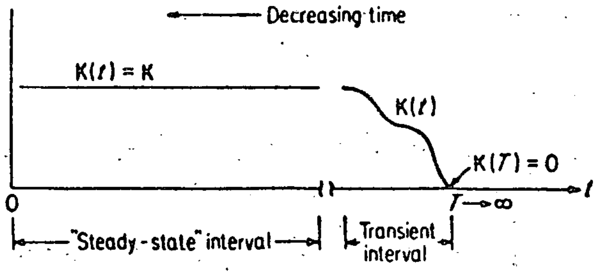


FIG. 9-8. A loose interpretation of the constant matrix K. As $T \longrightarrow \infty$, the "transient interval" tends to infinity and the "steady-state interval" occupies all finite times. (Reprinted by permission from M. Athans and P. L. Falb, *Optimal Control*, McGraw-Hill Book Company, New York, 1966.)

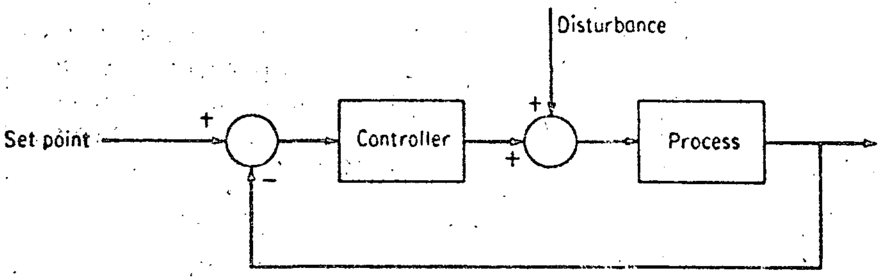


FIG. 9-9. Conventional representation of control loop.

We shall first consider the set point case. Specifically, suppose the first-order system

$$\frac{dx(t)}{dt} + x(t) = u(t) \tag{9-33}$$

is initially at state x_0. Suppose that at time zero the set point is changed to x_f. The typical response in this case is as shown in the top two graphs in Fig. 9-10.

To cast this problem into the optimal control formulation, it is necessary that the final value of the state variable be zero and the final value of the control be zero also. Thus we define two new variables as

$$x_1(t) = x(t) - x_f \tag{9-34}$$

$$u_1(t) = u(t) - u_f \tag{9-35}$$

Substituting into Eq. 9-33 gives

$$\frac{dx_1(t)}{dt} + x_1(t) + x_f = u_1(t) + u_f$$

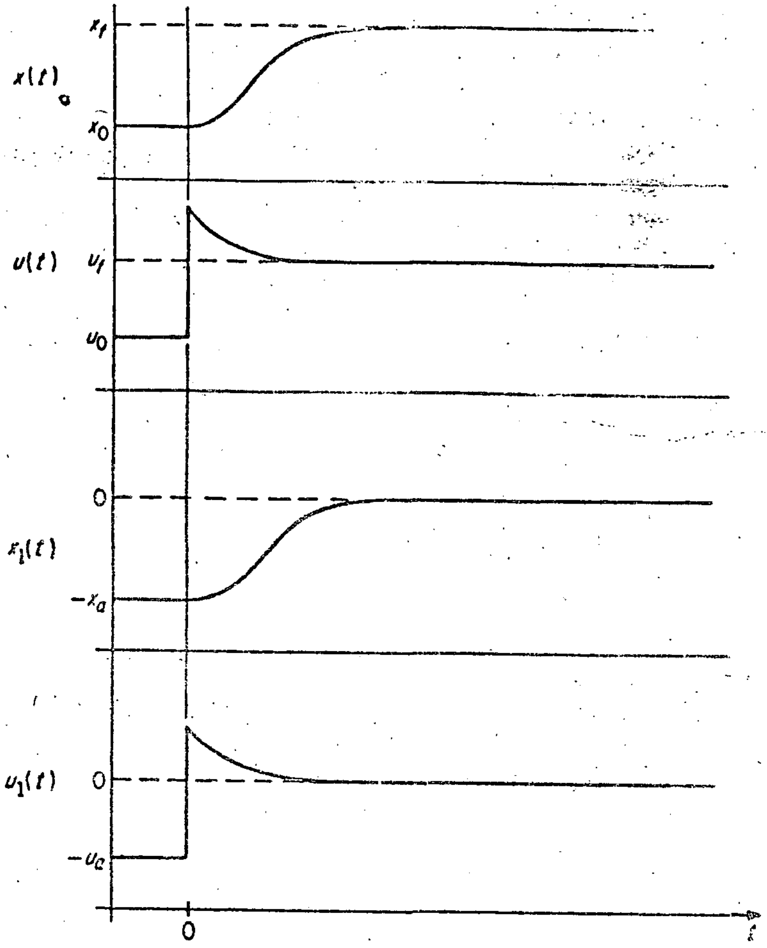


FIG. 9-10. Control and response for first-order system.

As Eq. 9-33 indicates that x_f equals u_f at steady state, this equation reduces to

$$\frac{dx_1(t)}{dt} + x_1(t) = u_1(t) \tag{9-36}$$

The boundary condition is

$$x_1(0) = x_0 - x_f = -x_a \tag{9-37}$$

As the control is now such that the state $x_1(t)$ is to be transferred from $-x_a$ (the initial condition) to zero (the origin), optimal control theory can be applied. Let the cost functional be defined as follows:

$$J = \frac{1}{2}\int_0^\infty [x_1(t)^2 + u_1(t)^2]\,dt \tag{9-38}$$

Substituting for the corresponding quantities in Eq. 9-32 gives

$$+ 2k + k^2 - 1 = 0$$

The solution is

$$k = 0.416$$

Thus the controller is a pure proportional controller with a gain of 0.416.

Here we begin to have some difficulties. Using a pure proportional controller, we are proposing to make a set point change and not have any offset (i.e., error) at the new operating point. The only case in which the proportional control will not exhibit such offset is at its equilibrium point. By making the above change of variable, we effectively defined this equilibrium point to be at the new set point.

It is also interesting to note that the controller does not exhibit the integral mode. As the control is simply a linear combination of the states of the system (see Eq. 9-29), we will have an integral mode only if we define a state corresponding to the integral of the state variable. For the first-order system considered above, this could potentially be accomplished by the approach in Fig. 9-11. The performance functional must be of the form

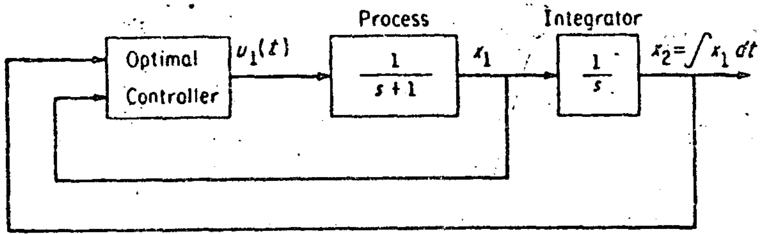


FIG. 9-11. A possible means for introducing an integral term into the optimal control law.

$$J = \int_0^\infty [q_1 x_1^2(t) + q_2 x_2^2(t) + u^2(t)]\,dt \tag{9-39}$$

The difficulty arises in assigning a "cost" to the state $x_2(t)$ which corresponds to the integral mode, i.e., select a value for q_2. Since this mode was added with the supposition that it could be used in the control law to achieve better control and is not part of the original process, it seems reasonable to set q_2 equal to zero. However, this leads to a zero value of the gain corresponding to the integral state variable, thus defeating the purpose for which the integral state was originally proposed.

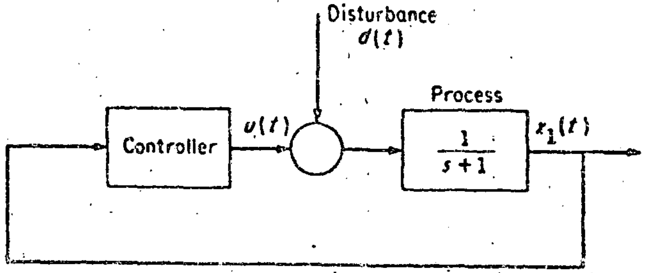
9-9 OPTIMAL CONTROL TO DISTURBANCE CHANGES

As the example to illustrate how an optimal controller may be designed for disturbance changes (13), consider the system in Fig. 9-12a. The state equation describing the process for this case is

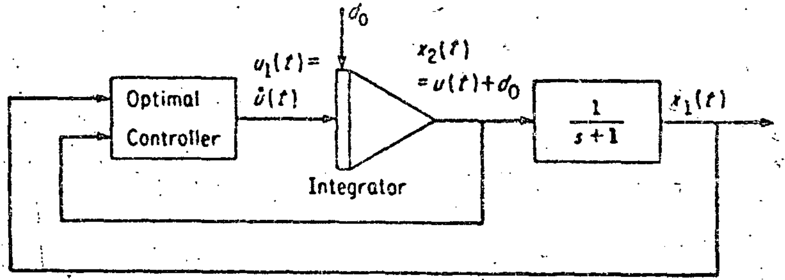
$$\dot{x}_1(t) = -x_1(t) + d(t) + u(t) \tag{9-40}$$

$$x_1(0) = 0 \tag{9-41}$$

Note that the disturbance appears as an input along with the control $u(t)$. To be cast into the optimal control formulation, we must transform the problem in such a manner that the disturbance appears a an initial condition.



(a) Disturbance regulator



(b) Equivalent state regulator formulation

FIG. 9-12. Transformation of the conventional disturbance regulator control problem into the optimal state regulator problem.

For the specific case in which the disturbance is a step chang this may be accomplished by the formulation in Fig. 9-12b. If th disturbance is a step change from 0 to d_0 at time zero, this may e fectively appear as an initial condition on an integrator. If the co tinuous input to the integrator is $\dot{u}(t) = u_1(t)$, the output is the sui of $d(t)$ and $u(t)$, as illustrated in Fig. 9-12b.

From this point, we proceed as usual. First, note that the state equations are (in terms of the new variables).

$$\dot{x}_1(t) = -x_1(t) + x_2(t) \tag{9-42}$$

$$\dot{x}_2(t) = u_1(t) \tag{9-43}$$

$$x_1(0) = 0$$

$$x_2(0) = d_0$$

In matrix form, this becomes

$$\frac{d}{dt}\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} = \begin{bmatrix} -1 & 1 \\ 0 & 0 \end{bmatrix}\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u_1(t) \tag{9-44}$$

The cost functional could be

$$J = \int_0^\infty [x_1^2(t) + u_1^2(t)]\, dt \tag{9-45}$$

Although the proper matrices could be substituted into Eq. 9-32, the resulting equations cannot be analytically solved for the coefficients of matrix K. Instead, the solution of Eq. 9-30 in backward time until steady-state is reached will yield the solution

$$\mathrm{K} = \begin{bmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{bmatrix}$$

Substituting into Eq. 9-29 gives

$$u_1(t) = -k_{21}x_1(t) - k_{22}x_2(t) \tag{9-46}$$

Again the control is a linear function of the states.

However, in this case the integrator is not really part of the process, but a part of the controller instead. Therefore we may eliminate $x_2(t)$ by substituting Eq. 9-42 into Eq. 9-46. Also noting that $u_1(t)$ is really $\dot{u}(t)$ gives

$$\dot{u}(t) = -k_{21}x_1(t) - k_{22}[x_1(t) + \dot{x}_1(t)]$$

Integrating gives

$$u(t) = -k_{22}x_1(t) - (k_{21} - k_{22})\int_0^t x_1(\tau)\, d\tau + U_0 \tag{9-47}$$

where U_0 = constant of integration. Thus we have proportional-plus-integral control.

It should also be noted that the cost functional in Eq. 9-45 is actually

$$J = \int_0^\infty [x_1^2(t) + \dot{u}^2(t)]\, dt$$

That is, the cost functional penalizes changes in control rather than for actual magnitude.

The application of the approach to control one of the unit operations has been reported by Miller (14). The system was a simulated distillation column, subjected to feed disturbances. The boilup rate was ratioed to the feed rate, and a feedback controller regulated the distillate rate to control the overheads composition (15). The scheme is illustrated in Fig. 9-13.

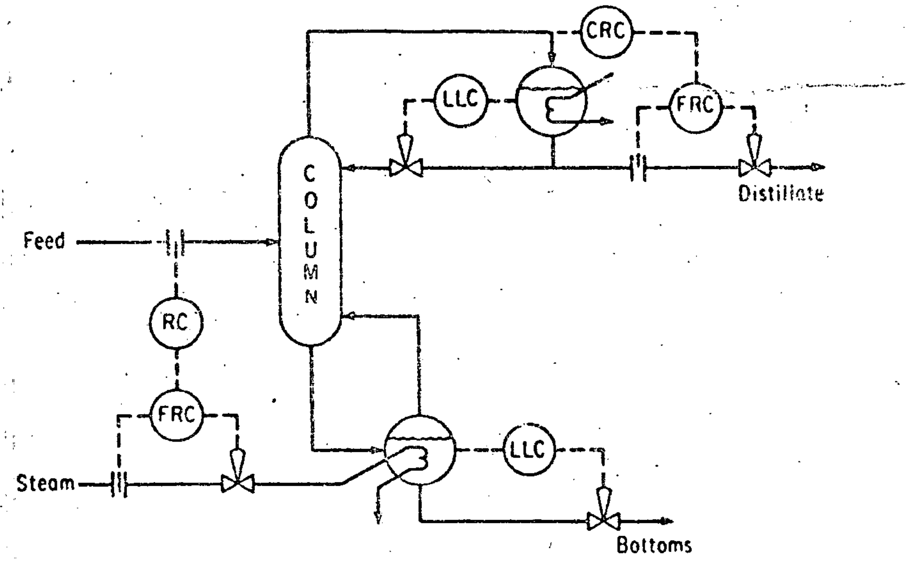


FIG. 9-13. Control scheme for distillation column.

As the manipulated variable is the distillate rate, a transfer function is needed to relate changes in overhead composition to changes in distillate rate. In Laplace transform notation, this model is

$$Y_D(s) = G_1(s)D(s) + F_e(s)$$

where $F_e(s)$ is the effect of a given change in feed rate on the overhead composition. From step responses such as those given in Fig. 9-14, it is apparent that $G_1(s)$ is a first-order lag for all practical purposes. Basing the time constant on the 63.2 percent point and the gain on the final steady-state values, averaging the values for the four responses in Fig. 9-14 gives the following model:

$$Y_D(s) = \frac{-0.0140}{0.55s + 1} D(s) + F_e(s)$$

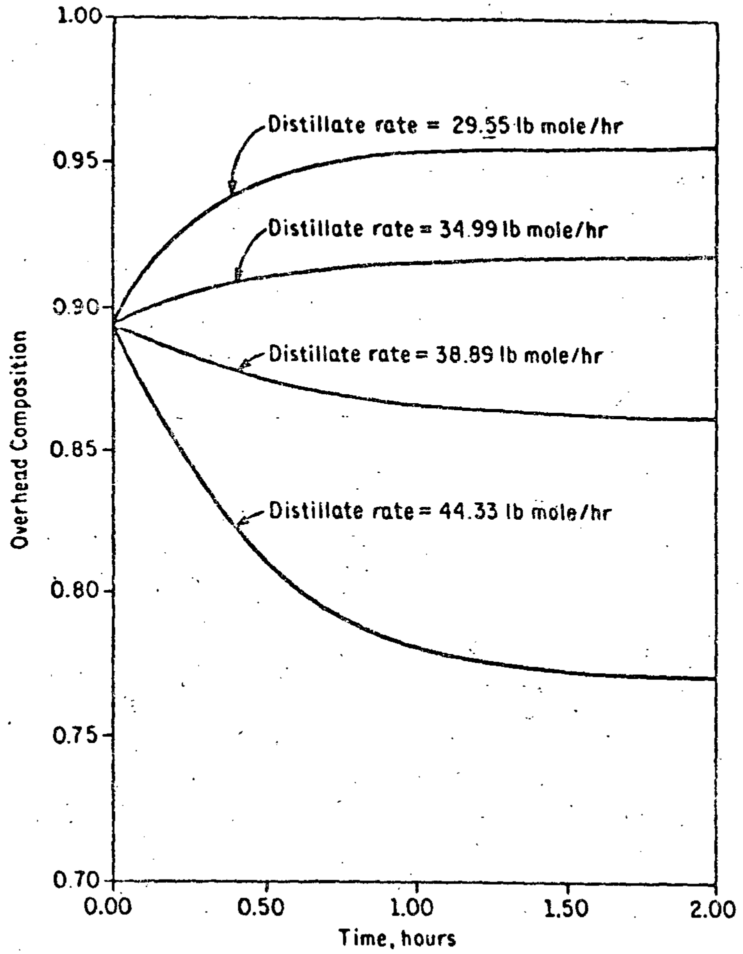


FIG. 9-14. Open-loop responses to several step changes in distillate rate.

Expressing in state-variable form gives the equation

$$\dot{y}_D(t) = \frac{dy_D(t)}{dt} = \frac{-1}{0.55}\, y_D(t) + \frac{-0.0140}{0.55}\, D(t) + \frac{1}{0.55}\, F_e(t)$$

Using the approach outlined previously in this section, the criterion function should be

$$J(D) = \int_0^\infty \{[y_{D_{\text{set}}} - y_D(t)]^2 + r\dot{D}^2\}\, dt$$

The cost $J(D)$ consists of two parts. The first part $[y_{D_{\text{set}}} - y_D(t)]^2$ penalizes for deviations of the controlled variable y_D from its desired value $y_{D_{\text{set}}}$. The second part D^2 penalizes for changes in the manipulated variable D.

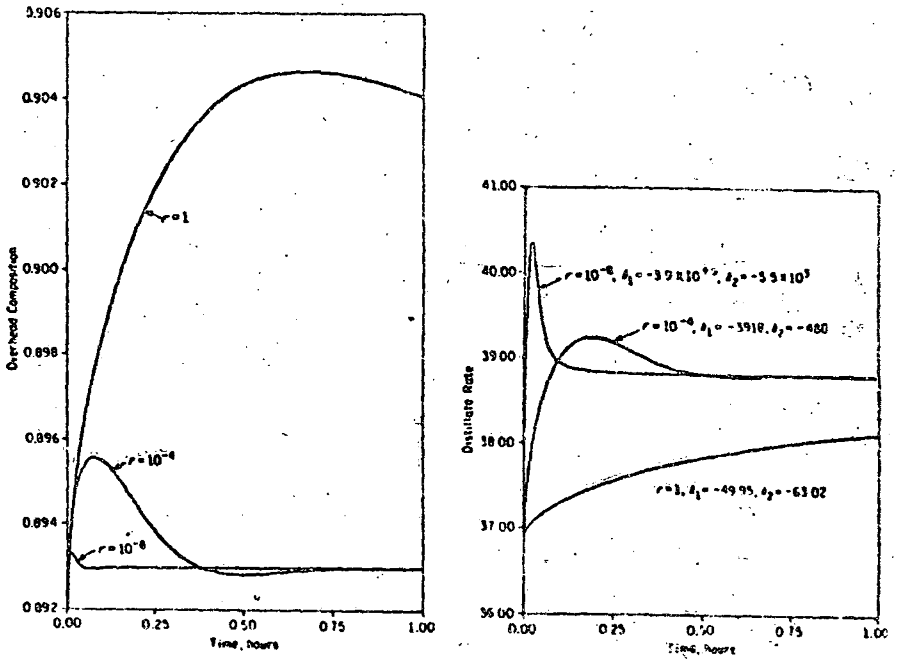
Applying the method presented previously gives the following control law.

$$D(t) = K_2 \int_0^t [y_{D_{\text{set}}} - y_D(\tau)]\, d\tau + K_1[y_{D_{\text{set}}} - y_D(t)] + D_0$$

where K_1, K_2 = control parameters
D_0 = constant of integration

The optimal controller is the familiar proportional plus integral feedback controller.

Figure 9-15 shows the effect of r on the resulting responses of y_D, the controlled variable, and D, the manipulated variable, to a step change in the feed rate. As would be suspected, small values of r lead to tight control and large changes in D. Large values of r produce the opposite results. Thus the parameter r is essentially a tuning parameter whose value must be determined by experimenting with the process in much the same manner as current controllers are tuned.



(a) Closed-loop feedback response

(b) Distillate rate response

FIG. 9-15. Effect of r on the performance of the control system.

REFERENCIAS

- MATHEMATICS OF SAMPLED DATA SYSTEMS
  C. SMITH INTEXT 1972

- OPTIMAL CONTROL, ATHANS Y FALB
  Mc. GRAW HILL, 1966

- LINEAR OPTIMAL CONTROL SYSTEMS
  KWAKERNAAK Y SIVAN
  WILEY, 1972

- LINEAR CONTROL SYSTEMS, ANALYSIS AND DESIGN
  D'AZZO AND HOUPIS
  Mc. GRAW HILL, 1975

centro de educación continua
división de estudios superiores
facultad de ingeniería, unam

INGENIERIA DE CONTROL DIGITAL DE PROCESOS

ALGORITMOS DE CONTROL

DR. VICTOR GEREZ GREISER

NOVIEMBRE 1978.

Algoritmos de Control

Dr. Gerez Nov. 78

Todavia se emplean mucho los algoritmos PID

Aumenta el empleo de los algoritmos "avanzados" diseñados empleando la transf. Z.

Modelos simples de procesos:

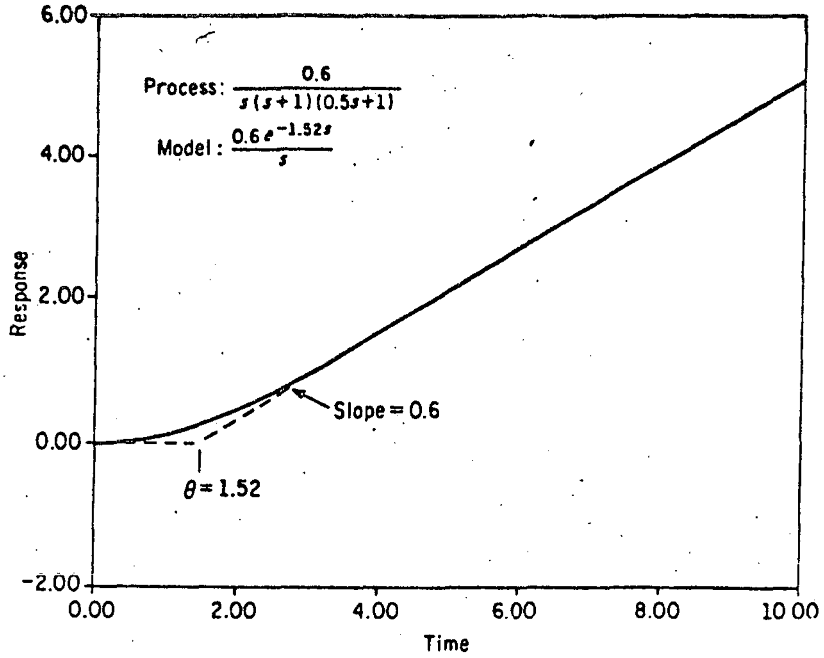
Características: Se obtienen de la respuesta a un escalón de lazo abierto.

Se emplean procedimientos sencillos basados en gráficos.

Procesos: No autoregulados $\int \rightarrow \frac{1}{s}$

Autoregulados

AC2



Respuesta, función de transferencia y modelo de un sistema no autoregulado

---

Modelo para procesos autoregulados:

$$G_M(s) = \frac{Ke^{-\theta s}}{\tau s+1}$$

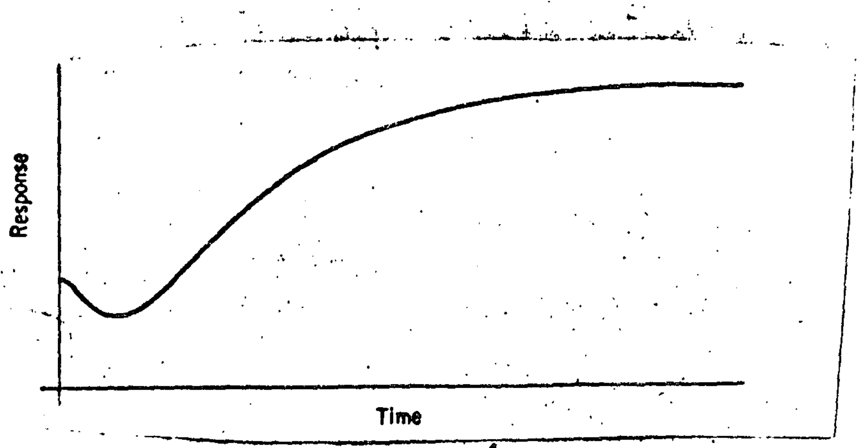
$$= \frac{Ke^{-\theta s}}{(\tau_1 s+1)(\tau_2 s+1)}$$

$$= \frac{Ke^{-\theta s}}{\frac{s^2}{\omega_n^2}+2\zeta\frac{s}{\omega_n}+1}$$

Mayor precisión

Mayor dificultad

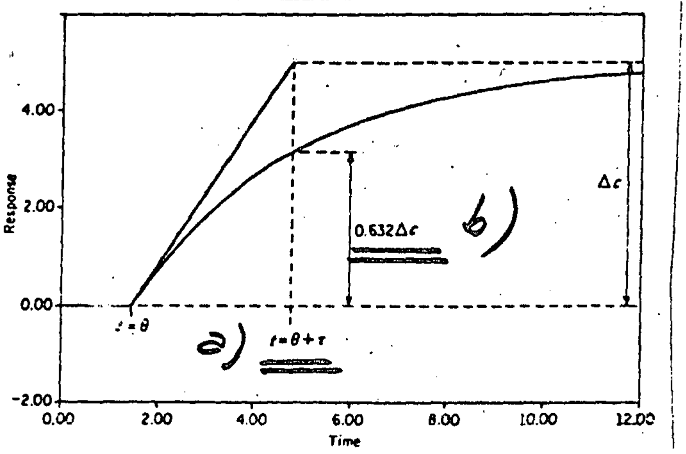
AC3



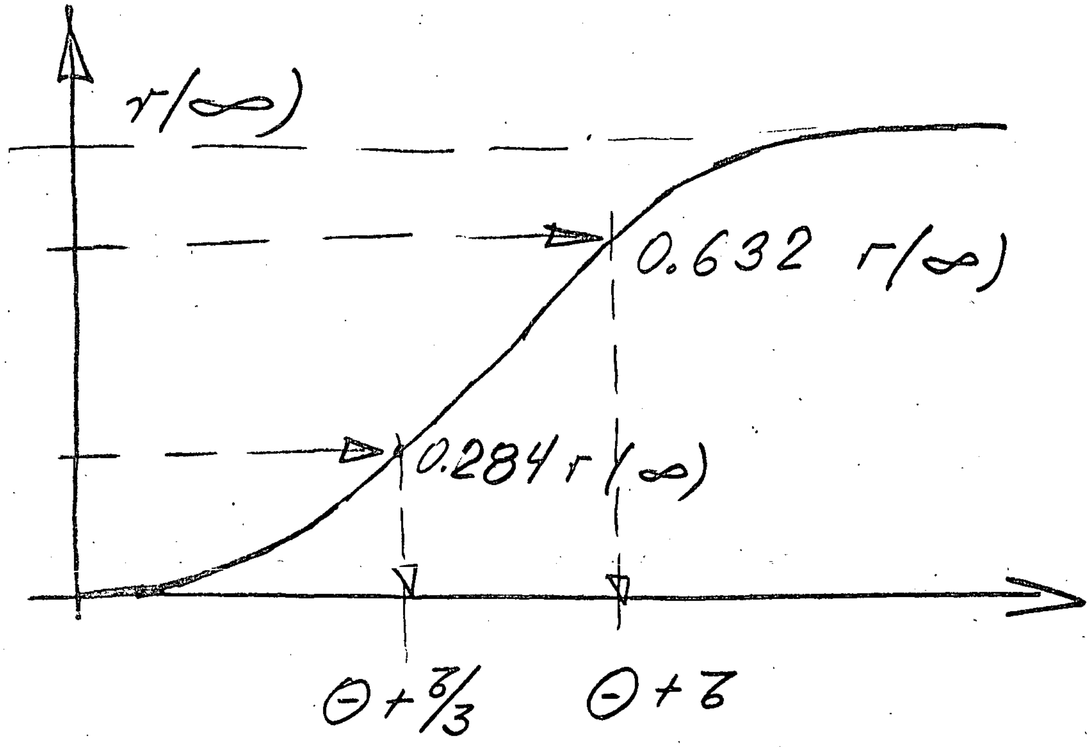
Respuesta al escalon de un sistema cuyo modelo es:

$$G_m(s) = \frac{Ke^{-\theta s}\left(\frac{s}{4}+1\right)}{(\tau_1 s+1)(\tau_2 s+1)}$$

[adelanto - atraso]



Caracteristicas de la respuesta de un sistema con atraso (un polo) y tiempo muerto ($e^{-\theta s}$) (a y b)



Ejemplo:

$$\left.\begin{aligned} 2.88 &= \Theta + \tau/3 \\ 4.80 &= \Theta + \tau \end{aligned}\right\} \begin{array}{l} 2 \text{ ecs.} \\ 2 \text{ incog.} \end{array}$$

Comparación de los métodos:

	T. muerto Θ	Const. tiempo τ
Milne	1.46	3.34
Analítico	1.92	2.88

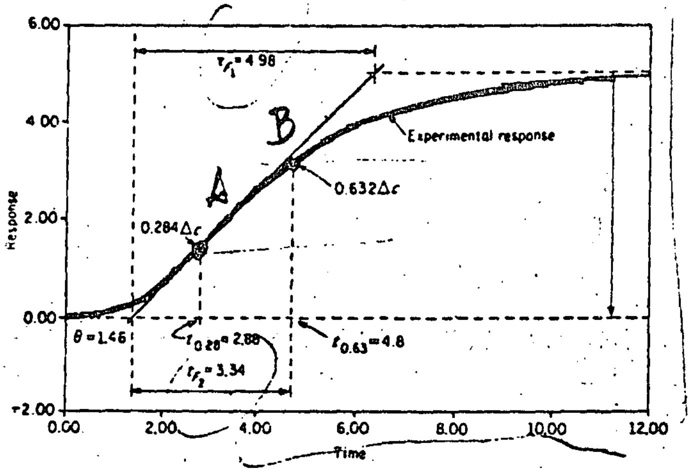
Es raro requerir modelos superiores al primero.

Ver: Smith, C.L. Digital Computer Process Control Pp - 141-145

Respuesta de un sistema de orden superior AC 4
(4to)

F.T. $$G(s)_{(real)} = \frac{1}{(0.5s+1)(s+1)^2(2s+1)}$$

$$G(s)_{(modelo)} = \frac{K e^{-\theta s}}{(\tau s+1)}$$



Metodo 1.- Medir θ y τ_{F_1}
" 2.- Medir θ y τ_{F_2} (punto $0.632\Delta c$)
3.- Analítico.-

$$c(t) = \Delta m\left(1 - e^{-(t-\theta)/\tau}\right) \quad t > 0$$

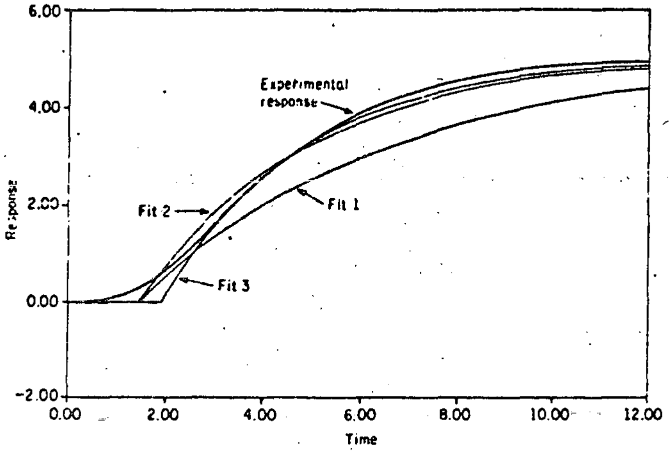
$$c\left(\theta + \frac{\tau}{3}\right) = \underline{0.284\,\Delta c} \rightarrow t = 2.88 = \theta + \tau/3$$

$$c(\theta + \tau) = \underline{0.632\,\Delta c} \rightarrow t = 4.80 = \theta + \tau$$

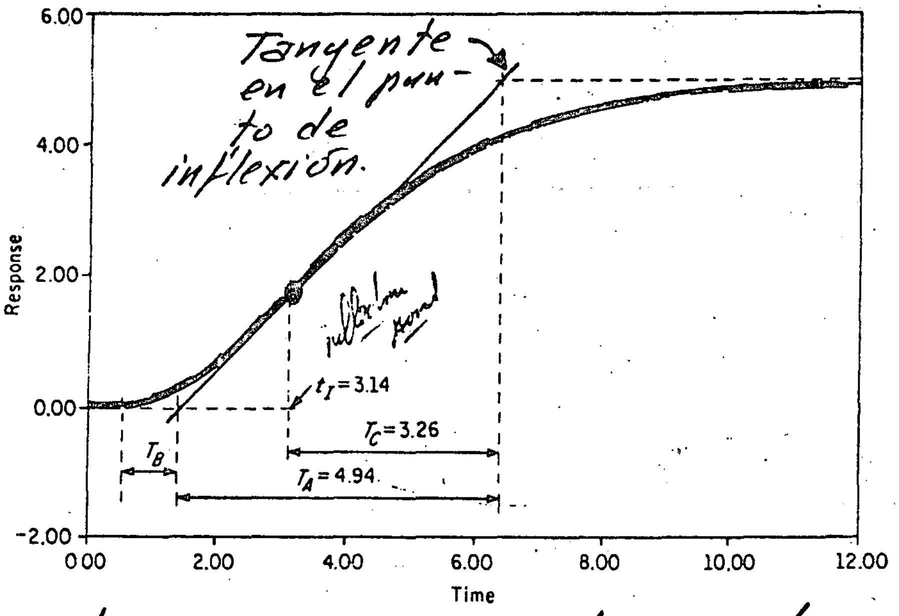
2 Ecs y 2 Incog.

Comparación de resultados

	Dead Time	Time Constant	Gain
Fit 1	1.46	4.98	1.0
Fit 2	1.46	3.34	1.0
Fit 3	1.92	2.88	1.0



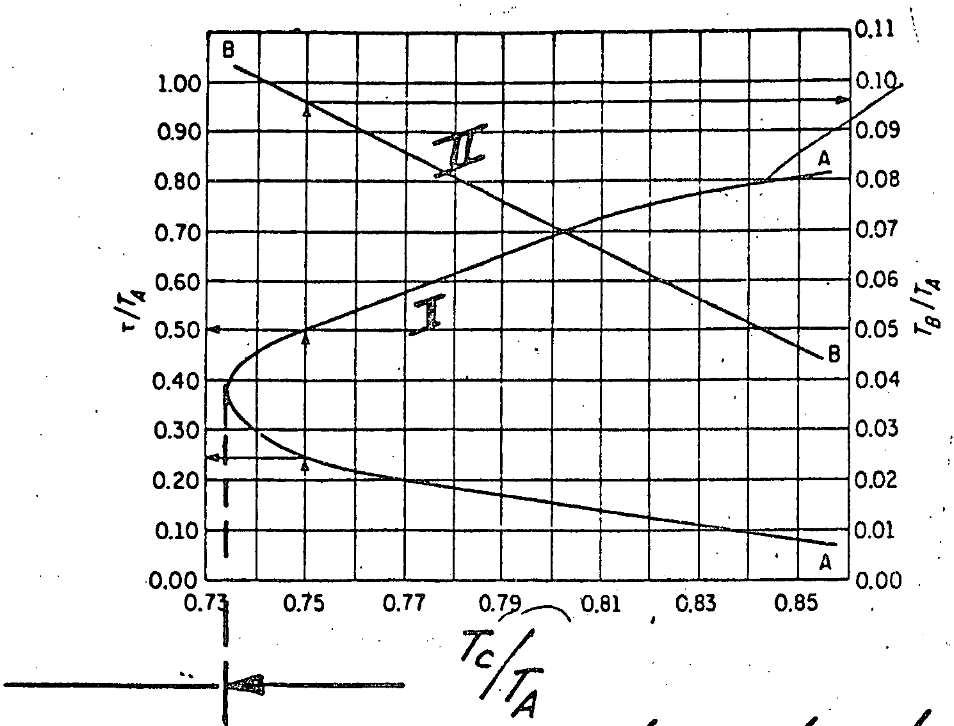
Método Gráfico para obtener modelos con dos polos.



a) Trace tangente en el punto de inflexión

b) Determine T_A y T_C

AC6



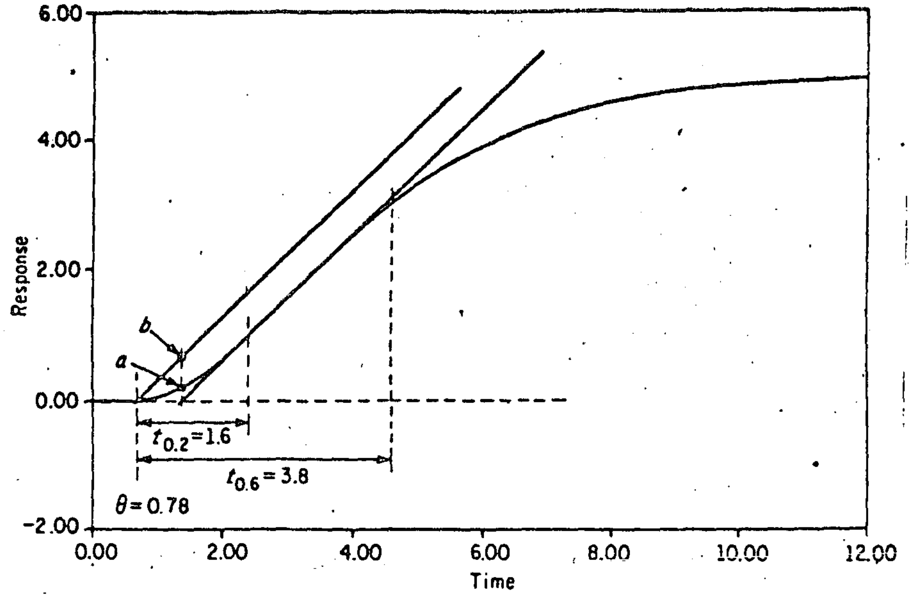
c) Con T_c/T_A y empleando la curva I [illegible] lea dos valores de τ/T_A y de ahí τ_1 y τ_2

d) Con T_c/T_A y empleando la curva II calcule T_B/T_A y despeje T_B

$$\Theta = t_I + T_C - T_A - T_B$$

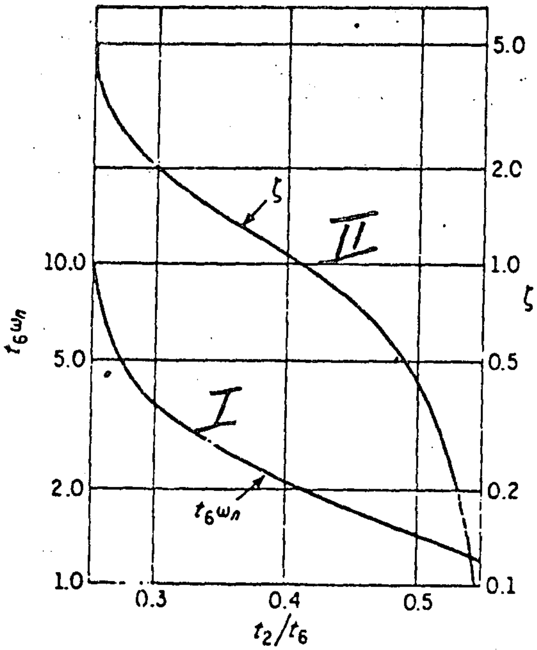
Observ: Para $T_c/T_A < 0.735$ el proceso es subamortiguado

AC 7



a) Encuentre $t_{0.2}$ donde se ha alcan. 20% del valor final

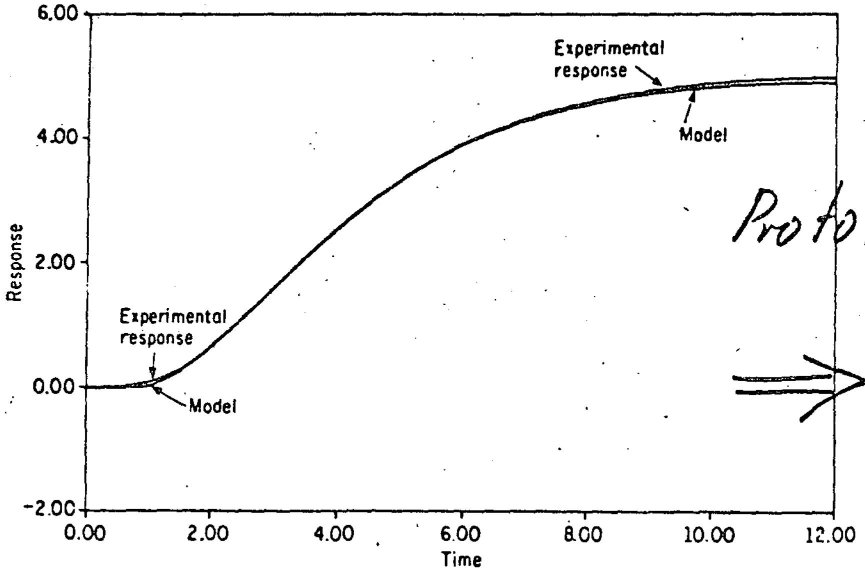
b) $t_{0.6}$ el 60%



c) Curva I para calcular ω_n

d) Curva II para calcular ζ

$$\text{Modelo} = \frac{Ke^{-\theta s}}{\frac{s^2}{\omega_n^2}+2\zeta\frac{s}{\omega_n}+1}$$

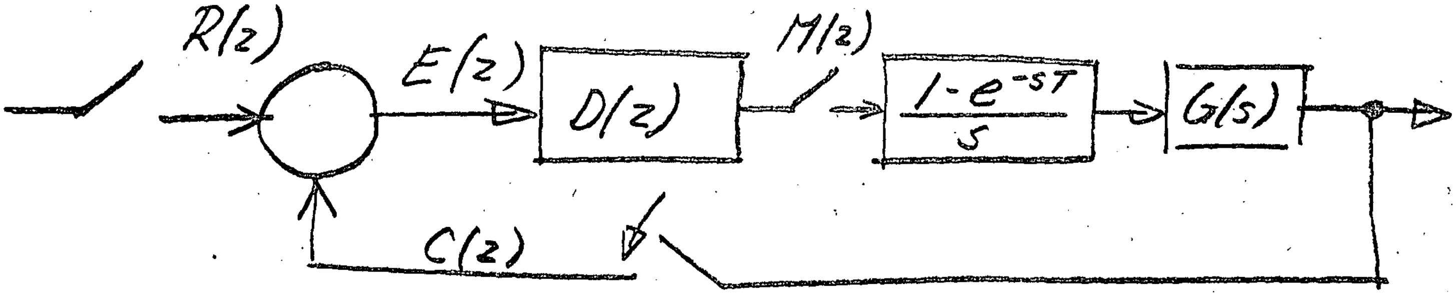


Comparación de resultados

$$\text{Prototipo} = \frac{1}{(0.5s+1)(s+1)^2(2s+1)}$$

$\Longrightarrow$ Mejor que los modelos de 1^er orden

Diseño empleando la transformada z



$$C(z) = HG(z)\,D(z)\left[R(z) - C(z)\right]$$

Datos: $\frac{C(z)}{R(z)}$; $HG(z) \longrightarrow$

$$D(z) = \frac{1}{HG(z)} \; \frac{\frac{C(z)}{R(z)}}{1 - \frac{C(z)}{R(z)}} \quad *$$

Observación: Si el proceso contiene tiempos muestos, ninguna D(z) puede eliminarlos; en la especificación de $\frac{C(z)}{R(z)}$ debe haber z^{-N}

AC-9

Model Pulse Transfer Functions

AC 12 →

	First-Order Model	Second-Order Model
Continuous transfer function	$\frac{1.0e^{-1.46s}}{3.34s+1}$	$\frac{1.0e^{-0.78s}}{4s^2+3.6s+1}$
Pulse transfer function, $T = 5$	$\frac{z^{-1}(0.6535+0.1227z^{-1})}{1-0.2238z^{-1}}$	$\frac{z^{-1}(0.6634+0.2491z^{-1}+0.0011z^{-2})}{1-0.09754z^{-1}+0.01111z^{-2}}$
Pulse transfer function, $T = 1$	$\frac{z^{-2}(0.1493+0.1095z^{-1})}{1-0.7413z^{-1}}$	$\frac{z^{-1}(0.005664+0.1167z^{-1}+0.3910z^{-2})}{1-1.2451z^{-1}+0.4066z^{-2}}$
Pulse transfer function, $T = 0.2$	$\frac{z^{-8}(0.0410+0.0171z^{-1})}{1-0.9419z^{-1}}$	$\frac{z^{-5}(0.004709+0.004434z^{-1})}{1-1.8261z^{-1}+0.8353z^{-2}}$

Prototipo : $\frac{1}{(0.5s+1)(s+1)^2(2s+1)}$

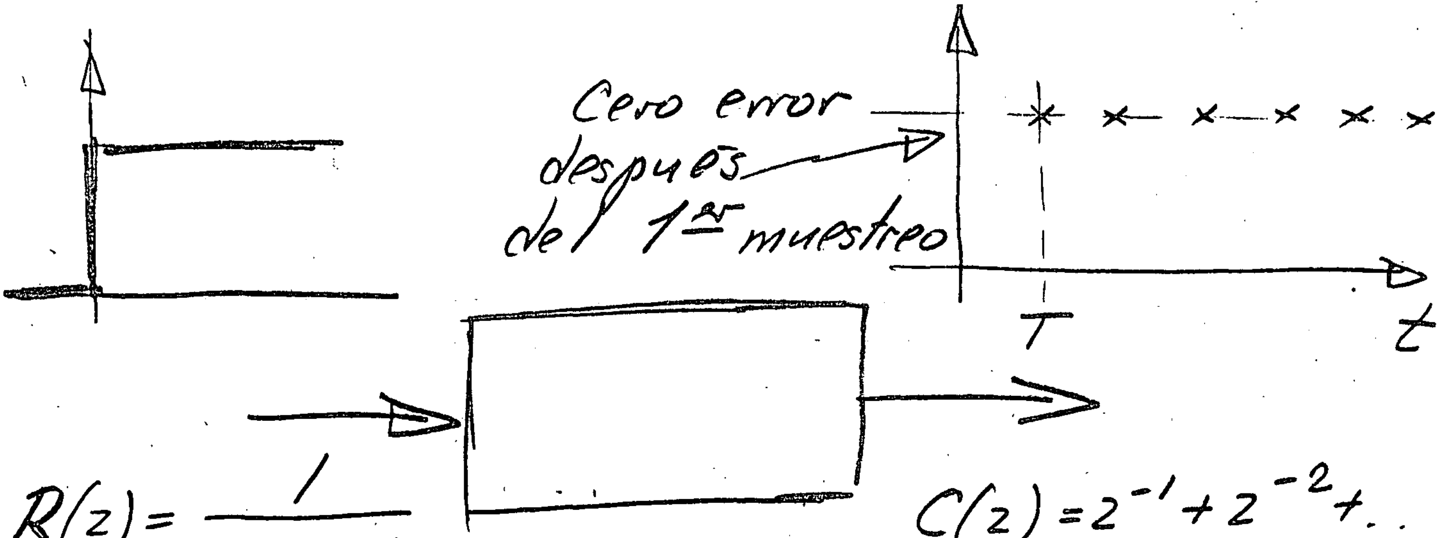
Algoritmos deadbeat

Especificaciones:

Tiempo de asentamiento finito

Tiempo de respuesta mínimo

Error $(\infty) = 0$



$R(z) = \dfrac{1}{1 - z^{-1}}$

$C(z) = z^{-1} + z^{-2} + \dots$

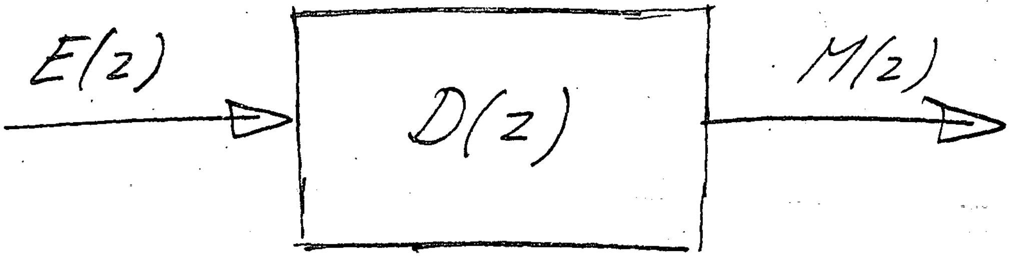
$= \dfrac{z^{-1}}{1 - z^{-1}}$

Sea $HG(z) = \dfrac{z^{-1}(0.65 + 0.12z^{-1})}{1 - 0.22z^{-1}}$

Nota: Corresponde al sistema especificado en pg. 4 $T=5$

Sustituyendo en pg 5 *

$$D(z) = \frac{1-0.22z^{-1}}{z^{-1}(0.65+0.12z^{-1})} \cdot \frac{z^{-1}}{1-z^{-1}}$$



$$D(z) = \frac{M(z)}{E(z)} =$$

Solución:

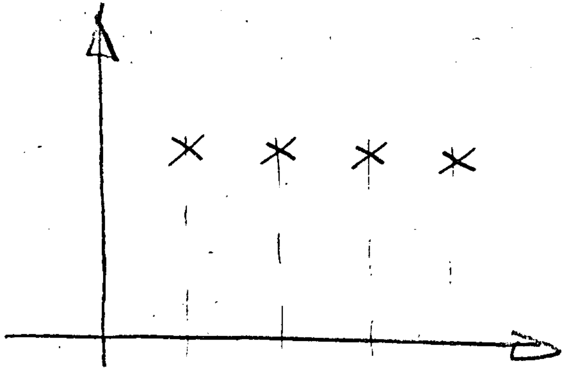
$$m_n = \frac{e_n - 0.22e_{n-1} + 0.53m_{n-1} + 0.12m_{n-2}}{0.65}$$

Observación: Si $T=1$ y como $\theta = 1.46$

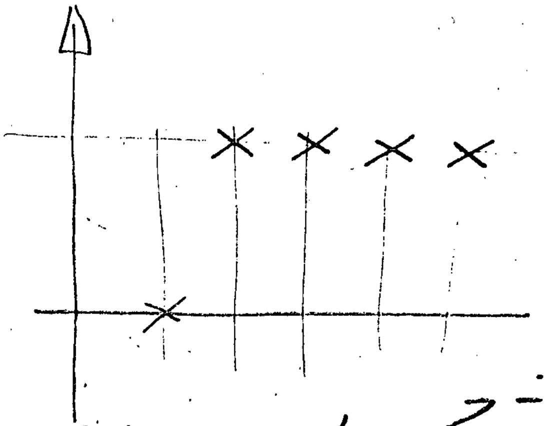
$$c(T) \neq c(\infty)$$

Especifique: $c(0) = c(T) = 0$

$$c(nT) = c(\infty)$$



$$C(z) = \frac{z^{-1}}{1-z^{-1}}$$



$$C(z) = \underset{\substack{\uparrow \\ \text{atraso}}}{z^{-1}} \frac{z^{-1}}{1-z^{-1}}$$

Método de Dahlin.

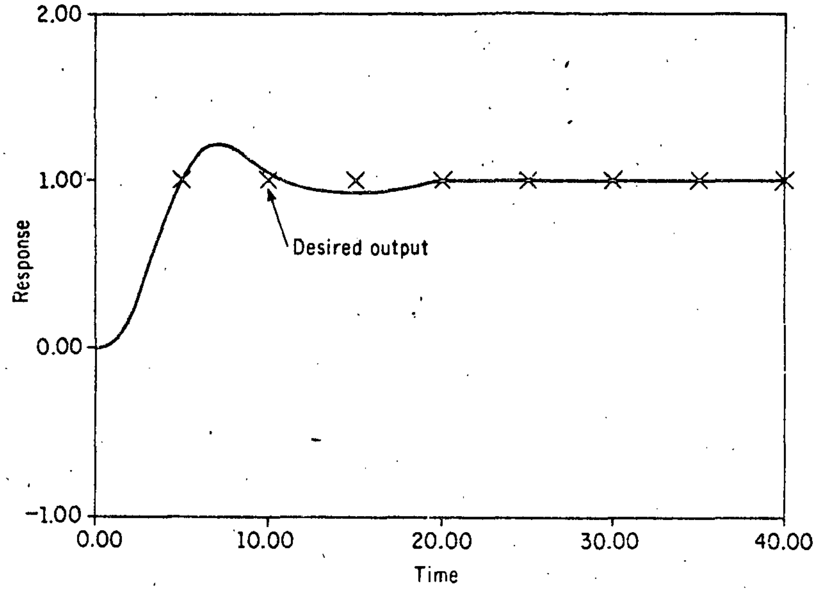
$$\underbrace{c(s) = \frac{e^{-\theta s}}{(\lambda s+1)} \cdot \frac{1}{s}}_{\text{respuesta buscado}}$$

$C(T) = c(\infty)$ es muy estricto.

$$C(z) = \frac{\left(1-e^{-\frac{T}{\lambda}}\right) z^{-N-1}}{(1-z^{-1})\left(1-e^{-\frac{T}{\lambda}} z^{-1}\right)}$$

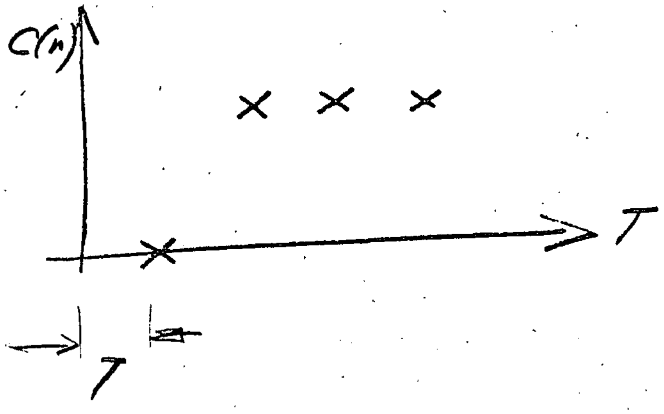
* $\theta = NT + \theta' \qquad \theta' < 1$

AC-13



Respuesta real del sistema de 4to orden con el controlador *AC-11

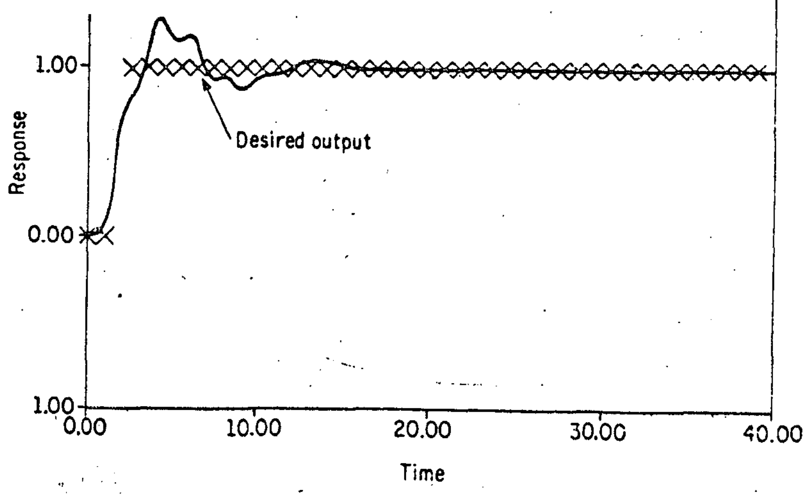
$$\text{Para } C(z) = \frac{z^{-2}}{1-z^{-1}}$$



$$\frac{C(z)}{R(z)} = z^{-2}$$

De AC-9

$$D(z) = \frac{1}{HG(z)} \cdot \frac{z^{-2}}{1 - z^{-2}}$$
$$= \frac{1 - 0.7413z^{-1}}{(0.1493 + 0.1095z^{-1})(1 - z^{-2})}$$



$$R(z) = \frac{1}{1-z^{-1}} \quad (\text{escalón})$$

$$\underbrace{\frac{C(z)}{R(s)} = \frac{1-e^{\frac{T}{\lambda}}z^{-N-1}}{1-e^{\frac{T}{\lambda}}z^{-1}}}_{\downarrow}$$

igual que en el caso anterior.

Trace la respuesta y observe que λ es "mejor"

$$D(z) = \frac{C(z)/R(z)}{1 - C(z)/R(z)} \cdot \frac{1}{HG(z)}$$

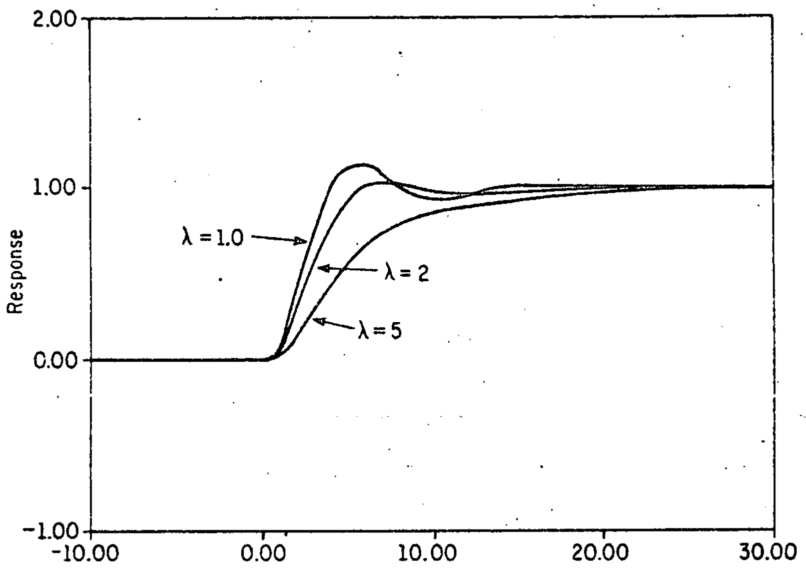
$$= \frac{(1 - e^{-T/\lambda})z^{-N-1}}{1 - e^{-T/\lambda}z^{-1} - (1 - e^{-T/\lambda})z^{-N-1}} \cdot \frac{1}{HG(z)}$$

Para el sist. de 4to orden AC-4 con $T = 1$ y una $\lambda = 2$ se tiene.

$$D(z) = \frac{0.292(1 - 0.741z^{-1})}{(1 - 0.608z^{-1} - 0.392z^{-2})(0.149 + 0.110z^{-1})}$$

$$= \frac{1.96(1 - 0.741z^{-1})}{(1 - z^{-1})(1 + 0.392z^{-1})(1 + 0.738z^{-1})}$$

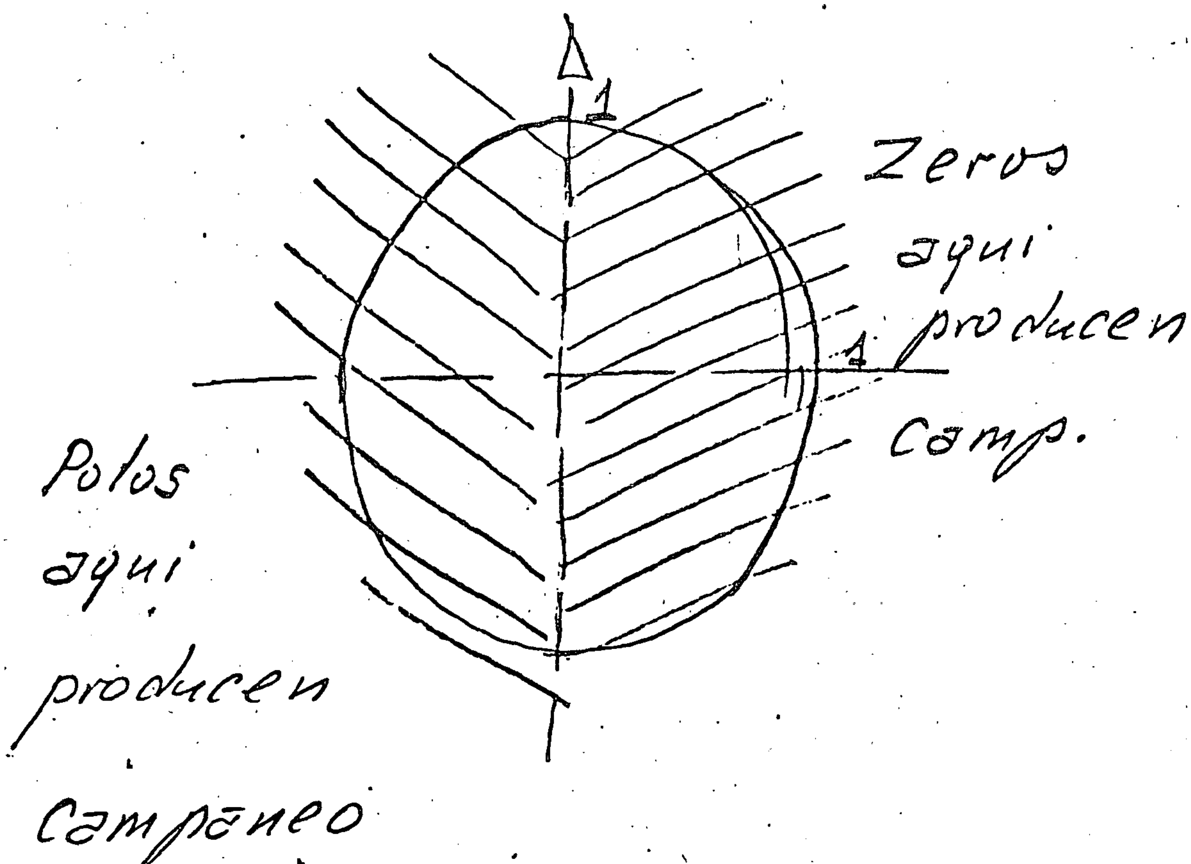
$$= \frac{1.96z^2(z - 0.741)}{(z - 1)(z + 0.392)(z + 0.738)}$$

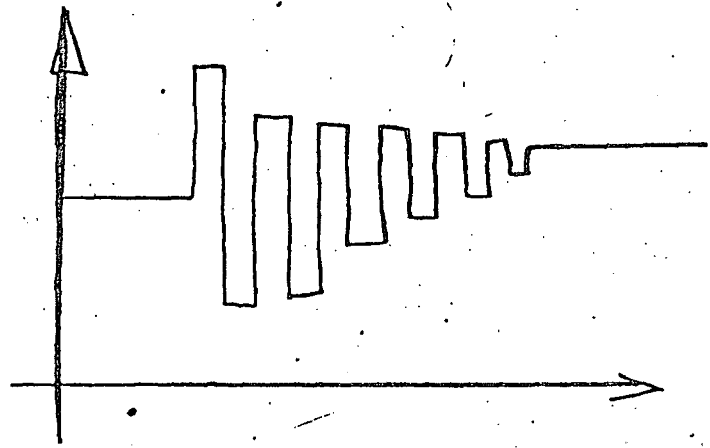


Comportamiento del algoritmo de Dahlin para diversos valores de λ.

CAMPANEO

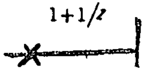
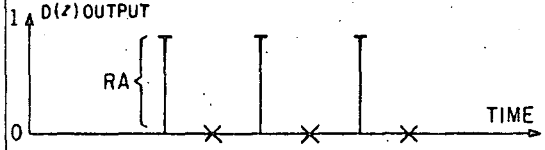
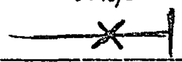
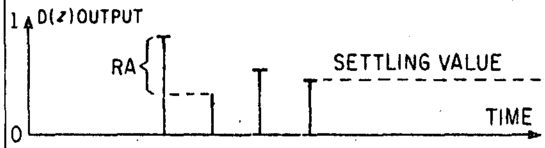
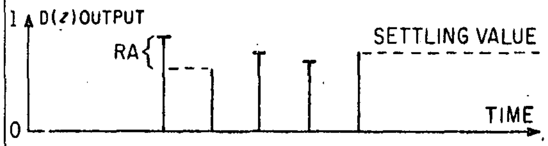
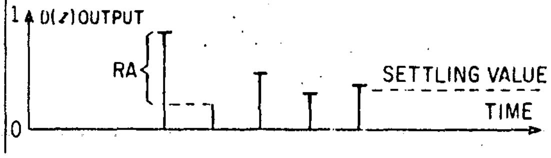
La localización de polos y ceros de una función determina a este fenómeno.-





Ejemplo de campaneo.

AC-17

	D(Z)	Impulse Response	Step Response	RA	Time Plots
1	$\frac{1}{1+1/z}$ 	1 -1 1 -1 1	1 0 1 0 1	1	
2	$\frac{1}{1+.5/z}$ 	1 -.5 .25 -.125	1 .5 .75 .625	.5	
3	$\frac{1}{(1+.5/z)(1-.2/z)}$ 	1 -.3 .19 -.087 .045	1 .7 .89 .803 .848	.3	
4	$\frac{1-.5/z}{(1+.5/z)(1-.2/z)}$	1 -.8 .36 -.188 .0924	1 .2 .50 .37 .46	.8	

1.- Polo en -1 Dif de resp. 1 a 0 = 1

2.- " en -0.5 Dif de resp 1 a 0.5 = .5

3- " en +0.2 (Semiplano derecho)

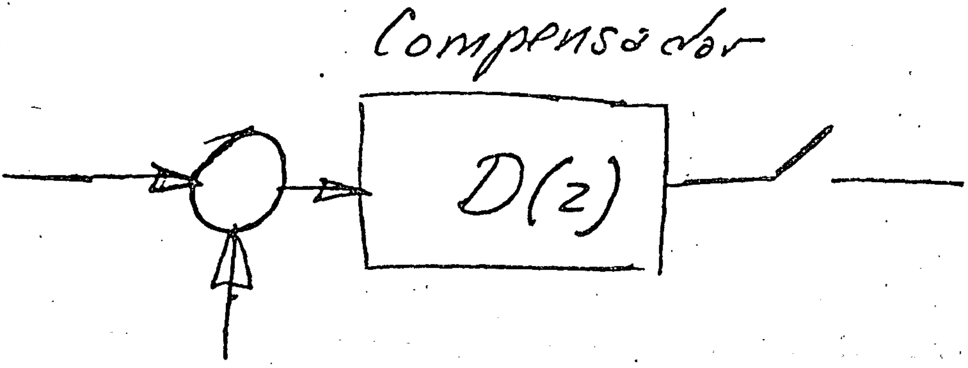
Resp: 1 a 0.7 = 0.3

4 Zero en 0.5

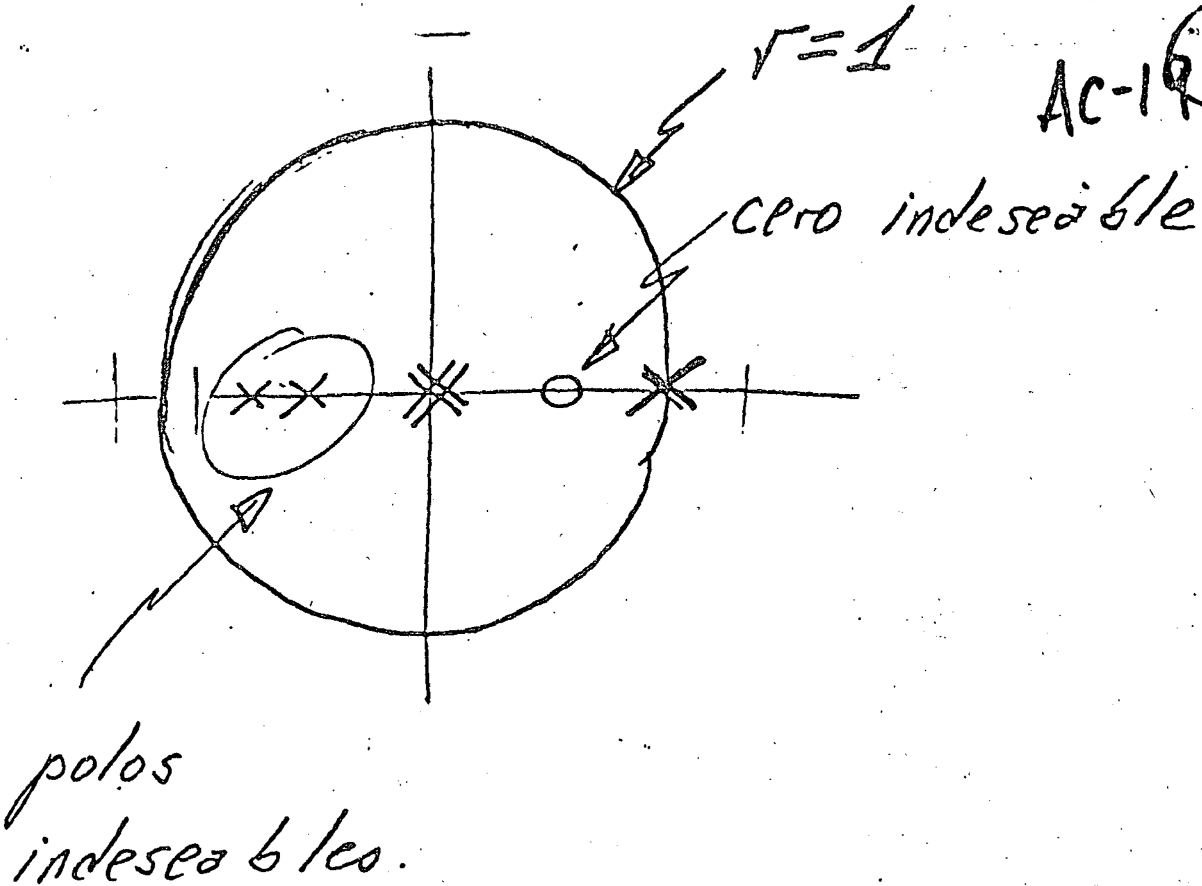
Resp 1 a 0.2 = 0.8 se agrava

Observación: Puede **no** existir en la señal de salida, pero **si** en alguna intermedia.

Ejemplo:



$$\underbrace{D(z)=\frac{1.96z^2(z-0.741)}{(z-1)(z+0.392)(z+0.738)}}_{\text{Diseñado por el método de Dahlin (Sección anterior)}}$$

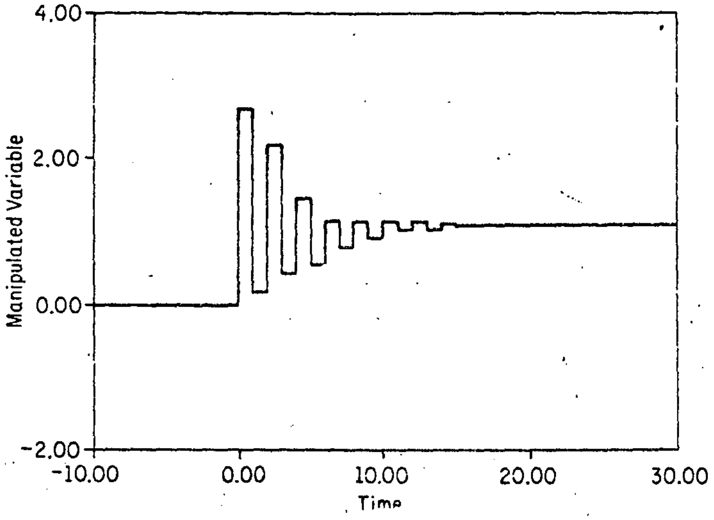


Remedio:

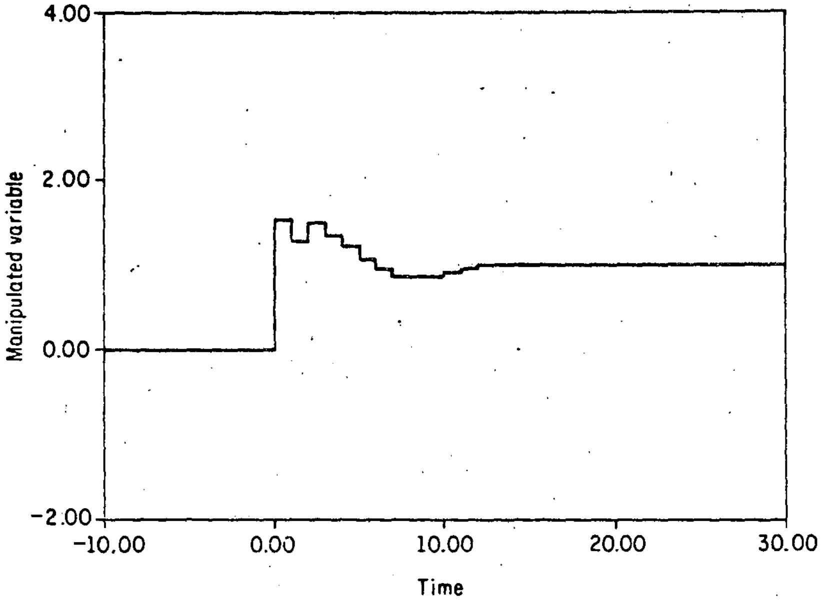
- Pruebe eliminando el más negativo ajustando la ganancia
- Ganancia ($t\to\infty$) correspondiente a cada termino se obtiene sustituyendo $z=1$

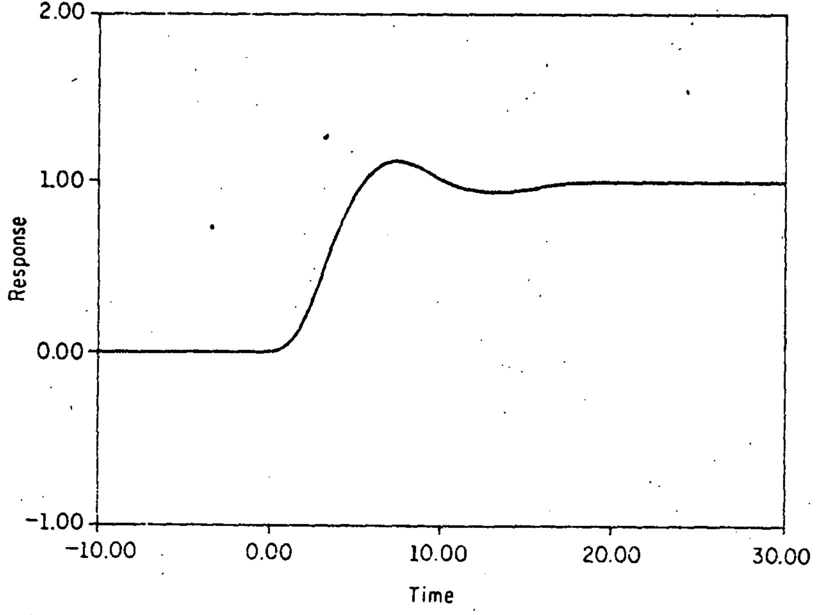
$$D(z)=\frac{1.96\, z^2(1-0.741z^{-1})}{(1-z^{-1})(1+0.392z^{-1})(1+0.738)}$$

AC-20



Con campaneo



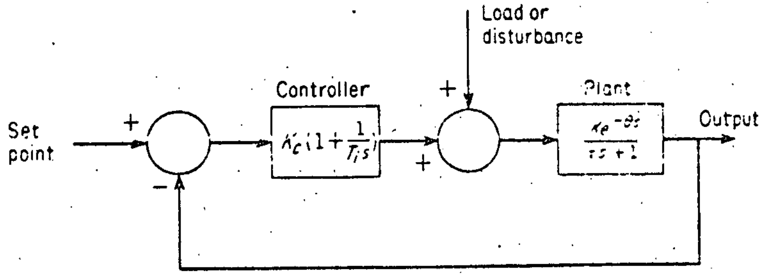


Campaneo removido

Algoritmo de Dahlin para λ=2 T=1

AJUSTE DEL ALGORITMO.-

1- Aproxime el próceso con un modelo
2.- Simule el proceso con el modelo y ajuste las ganancias
3- Aplique los ajustes anteriores al proceso
4.- Reajuste las ganancias en el proceso



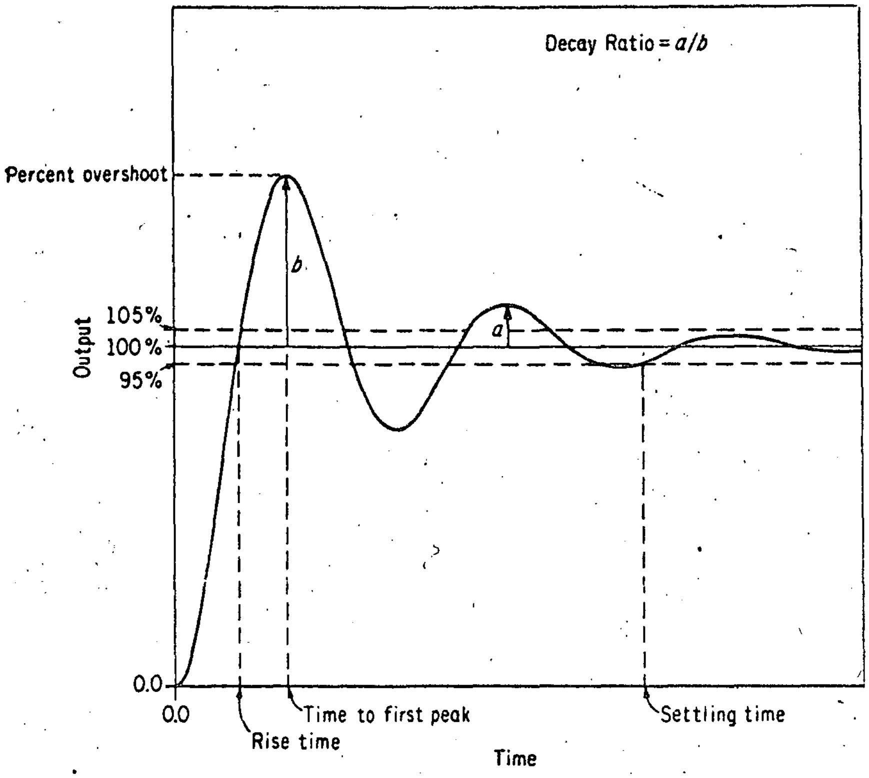
$$\underset{K_c,\,T_i}{\text{opt}}\{c(t)\}$$

¿Con que criterio se optimiza?

a) $ISE = \int_0^\infty e(t)^2\,dt$

b) $IAE = \int_0^\infty |e(t)|\,dt$

c) $ITAE = \int_0^\infty t|e(t)|\,dt$



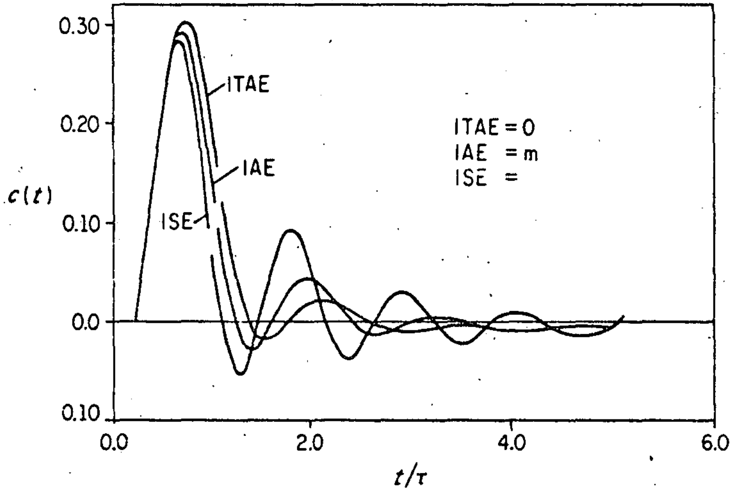


FIG. 6-27. Comparison of responses to a unit step change in load for different performance criteria; PI algorithm, $G(s) = (e^{-0.2\tau s})/(\tau s + 1)$; $T/\tau = 0.10$. (Reprinted by permission from Ref. 16.)

Nota: Todavia se hacen ajustes con frecuencia simplemente reduciendo el sobretiro y/o tiempos de asentamiento

TECNICAS DE IDENTIFICACION EN LINEA

Tecn. exper. de modelado $\left\{ \begin{array}{l} \text{dominio del } t \\ \\ \text{dominio } f \end{array} \right.$

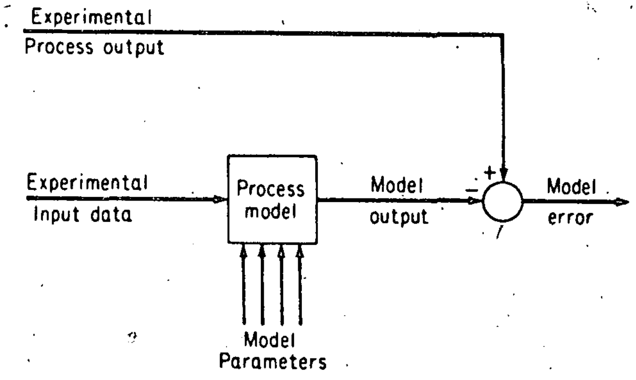
	Ventajas	Desventajas
Tecnicas del dominio t	Parametros faciles de determinar	Debe postularse el modelo. No indica como mejorarlo
Dominio de la frecuencia	No debe postularse el mod. Si indica hacia donde debe mejorarse el modelo	Dificilmente aplicables en linea

Procedimiento:

a) Haga prueba Si es necesario con mayores amplit. que las normales

b) Postule modelo y asuma valores

c) Calcule respuesta del modelo y compare con el prototipo.



Use modelos discretos:

Ventajas: a) Se requieren menos cálculos

b) Aveces puede emplearse regresión lineal

Criterios: $\int [e(t)]^2 dt$ $\quad \Sigma e_i^2$

$\int |e(t)| dt$ $\quad \Sigma |e_i|$

min desv. max $\quad \min\{\max(e(t))\}$

$\min\{\max(e_i)\}$

Los parámetros dependerán precisamente del criterio empleado

ESTIM. DE PARAMETROS CON R.L.

Dado:

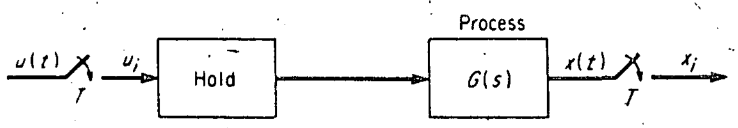
$$G(s) = \frac{q_m S^m + \ldots \qquad + q_1 S + q_0}{a_n S^n + \qquad + a_1 S + 1}$$

o la versión en el dominio t

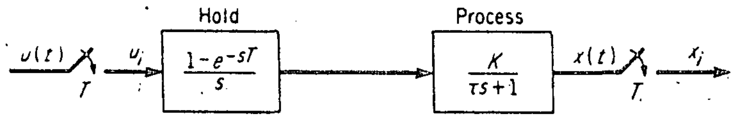
=

Encuentre $q_m, \ldots q_1, q_0, a_n, \ldots a_1,$

Transf. a modelo discreto, pueden usarse técnicas de regresión lineales.



(a) General system



(b) First-order system with zero-order hold

FIG. 7-2. Block diagram for deriving pulse transfer function.

Considere que el sist. es de 1er orden y el reten es de orden zero:

$$HG(z) = \frac{k(1-e^{-T/\tau})z^{-1}}{1-z^{-1}e^{-T/\tau}} = \frac{X(z)}{U(z)}$$

$$x_{i+1} = e^{-T/\tau}x_i + K(1-e^{-T/\tau})u_i + (D)$$

En estado permanente

$$x_{i+1} = x_i = x_s \quad u_i = u_s$$

$$x_s = K u_s + D'$$

$$D' = \frac{D}{1-e^{-T/\tau}}$$

D' es x_s para $u = 0$

Definiendo: $a = e^{-T/\tau}$

$$b = K(1 - e^{-T/\tau})$$

$$D = D'(1 - e^{-T/\tau})$$

$\hat{}$ Valores predecidos (respuestas del modelo)

Dos alternativas:

$$\hat{x}_{i+1} = a x_i + b u_i + D \quad \text{Reg. lineal}$$

$$\hat{x}_{i+1} = a \hat{x}_i + b \hat{u}_i + D \quad \text{Reg. no lineal}$$

$$\sum e_i^2 = \sum (x_{i+1} - \hat{x}_{i+1})^2$$

$$= \sum (x_{i+1} - a x_i - b u_i - D)^2$$

$$\frac{\partial}{\partial a}\left[\sum e_i^2\right] =$$

$$\frac{\partial}{\partial b}\left[\sum e_i^2\right] =$$

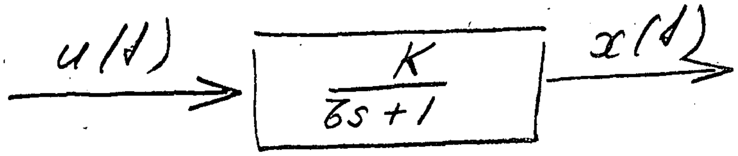
$$\frac{\partial}{\partial D}\left[\sum e_i^2\right] =$$

TI5

3 ecs en 3 incognitas.

<u>Camino alternativo:</u>

$$\tau \frac{dx}{dt} + x = Ku + D'$$



Empleando $1^{\underline{eras}}$ diferencias

Despejando x_{i+1}

TI 6

Igual ecs. que en el caso anterior.

El método de los dif. finitos es solo aplicable cuando T sea pequeño, pero siempre produce una regresión lineal

El método de la Z solo prod. un mod. lineal si # P = # Z+1 (Exactamente)

También existe la versión "integral"

$$\sum_{j=1}^{i+1} x_j = a \sum_{j=0}^{i} x_j + b \sum_{j=0}^{i} u_j + iD$$

$$\div\, i$$

$$\overline{X}_{i+1} = a \overline{X}_i + b \overline{U}_i + D$$

Si hay tiempo muerto:

$$G(s) = \frac{Ke^{-\theta s}}{\tau s + 1}$$

$$x_{i+1} = a x_i + b u_{i-n} + D$$

$$\underline{n\tau = \theta}$$

TI-7

Para continuar con una regresión lineal:
Suponga un valor de Θ ó n.
Itere el proceso sobre n hasta
minimizar el error.

METODOS DE DISEÑO DISCRETOS.

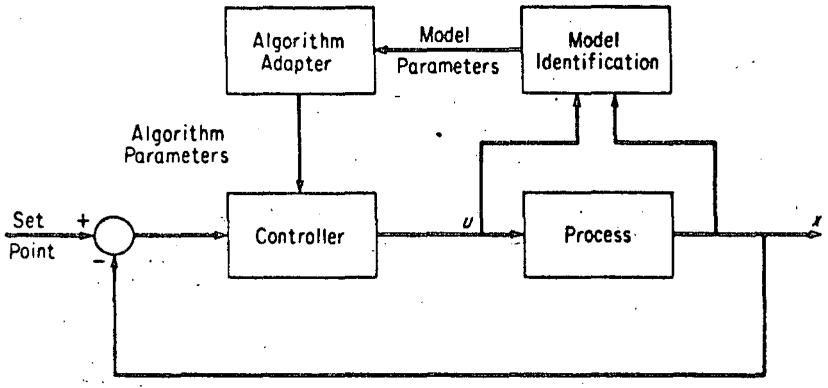
TI-8

Empiece con un modelo discreto:

$$x_i = -\sum_{j=1}^{p} a_j x_{i-j} + \sum_{j=1}^{q} b_j u_{i-j} + D$$

Siempre puede usarse regresión lineal y una técnica de diseño del alg. de control

$$HG(z) = \frac{b_1 z^{-1} + \ \dots \ + b_q z^{-q}}{1 + a_1 z^{-1} + \dots + a_p z^{-p}}$$



Esquema de Control Autoadaptivo de Kahlmann.

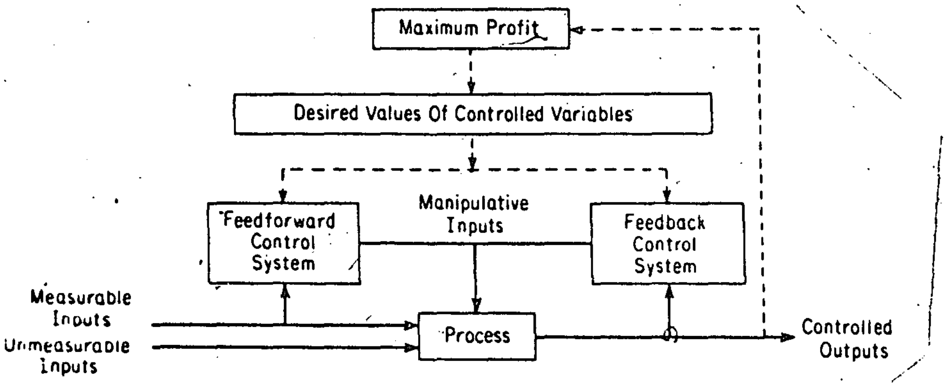
Algunos problemas:

Si hay poca desviación en la señal de control, las ecs. de la regresión lineal pueden no dar solución, pero en este caso no hay mucho interes en actualizar los parámetro.

TAC 1

TECNICAS AVANZADAS DE CONTROL

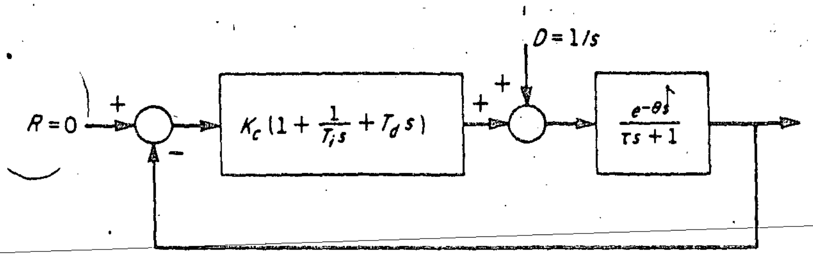
Control hacia adelante o predictivo:



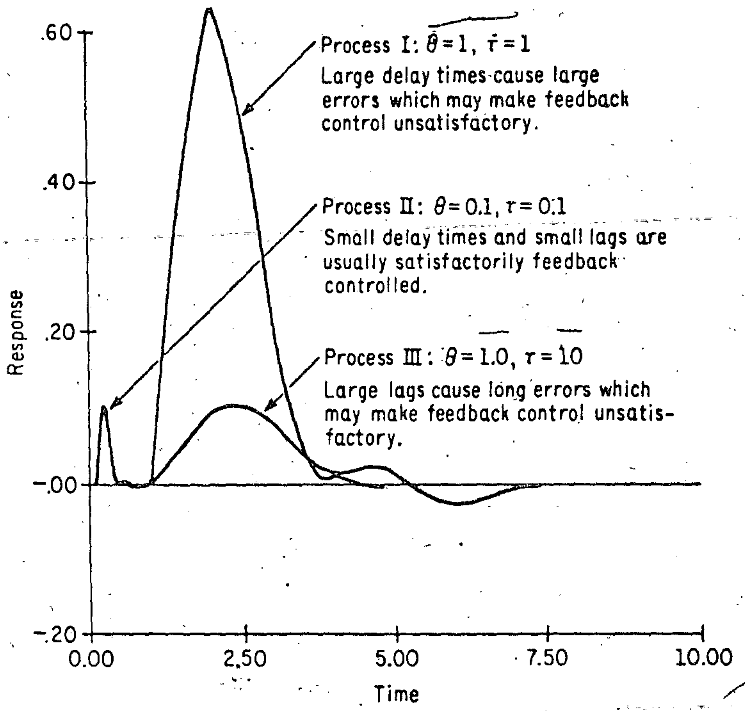
Desventajas del control realimentado:

TAC 2

El control R.A. no es satisfactorio cuando hay:
Disturbios frecuentes y/o grandes tiempos muertos.



(a) Block diagram of system



(b) Response plots for minimum ITAE

Se requiere sin embargo combinar control R.A. con control predictivo. Este último tiene que compensar: disturbios no medidos

errores en la salida debidos a errores en el modelo.

Ejemplo de Estado Permanente.

TAC 3

$$W C_p (T_0 - T_i) = F \Delta H_F$$

Variables: W = flujo de liq. Kg/hr

C_p = calor esp. del liq. Kcal/Kg °C

T_i

T_0

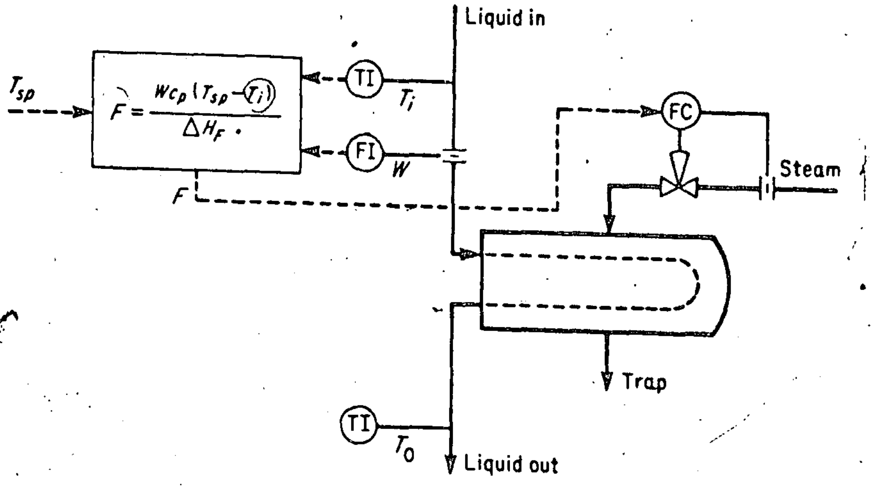
F = flujo de vapor Kg/hr

ΔH_F = calor cedido por el vapor Kcal/Kg

$$F = \frac{W C_p (T_0 - T_i)}{\Delta H_F} \qquad T_0 = T_{set}$$

$$F = \frac{W C_p (T_{set} - T_i)}{\Delta H_F}$$

El control predictivo no emplea T_0



TAC 4

Posibles discrepancias entre T_0 y T_{set} pueden compensarse con control hacia adelante con tres posibles esquemas

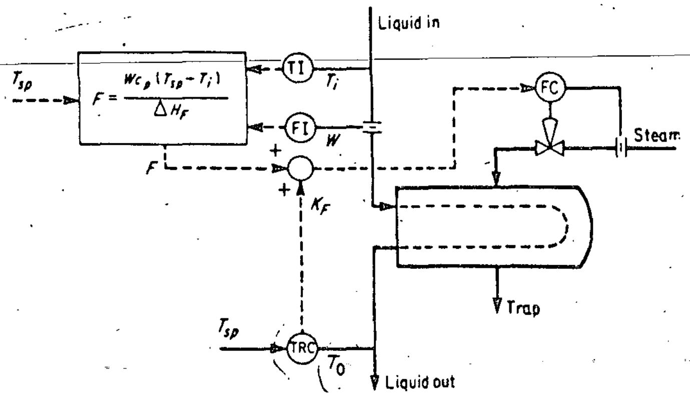


FIG. 8-4a. Feedback controller used to bias output of feedforward controller.

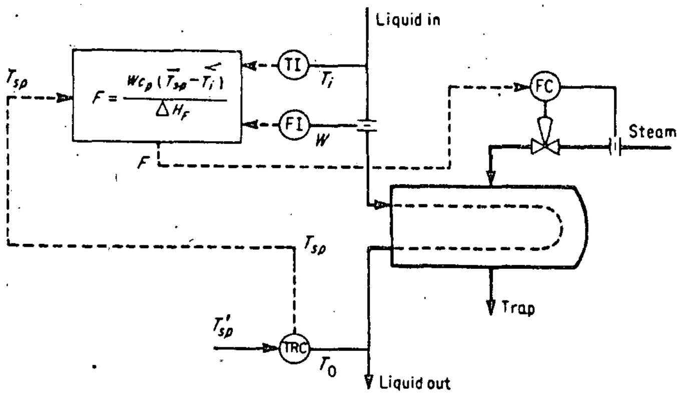
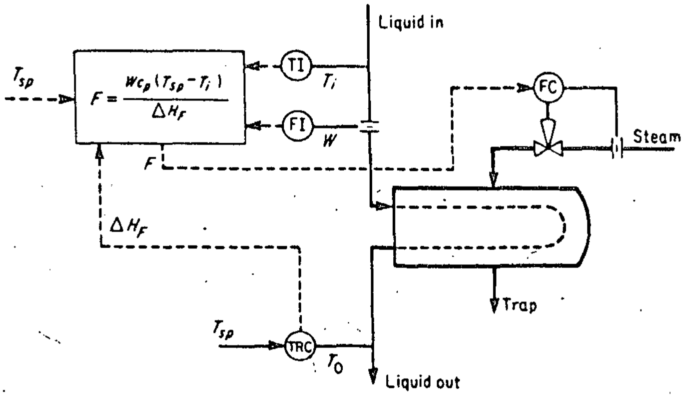


FIG. 8-4b. Output of feedback controller used to adjust the set point of the feedforward controller.



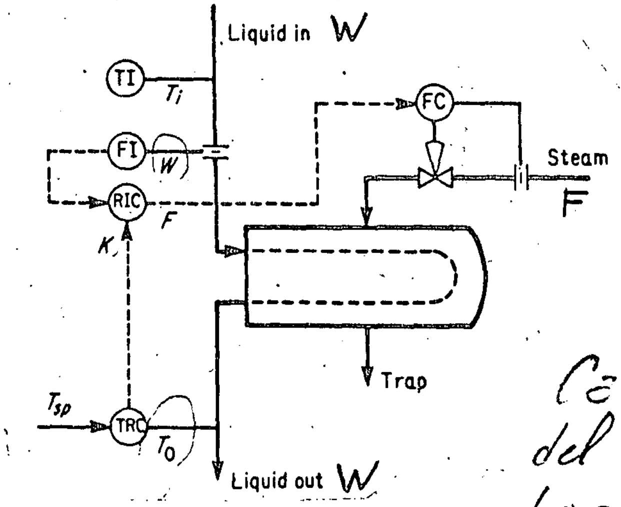
Use of feedback controller to adjust a parameter in the feedforward controller.

TAC 5

Si en: $F = \frac{W c_p (T_{sp} - T_i)}{\Delta H_F}$

T_{sp} y T_i cambian poco

$F \sim KW$ control proporcional



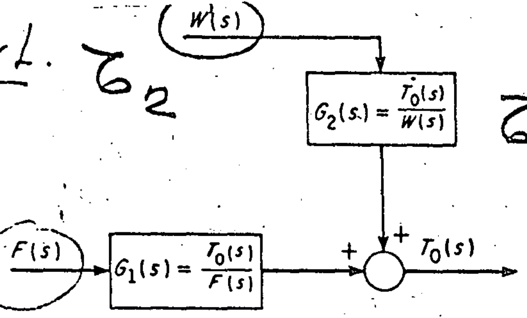
Cambios en el flujo del líquido son disturbios W

V. manipulada F

$\tau_1 < \tau_2$

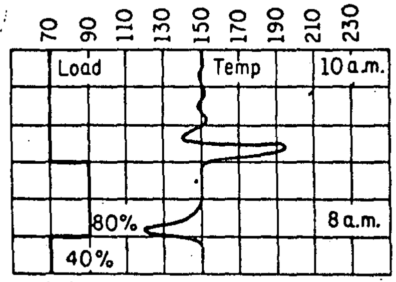
Emplee $G(s) = \frac{\tau_1 s + 1}{\tau_2 s + 1}$ en la tray. de predicción

Dist. τ_2

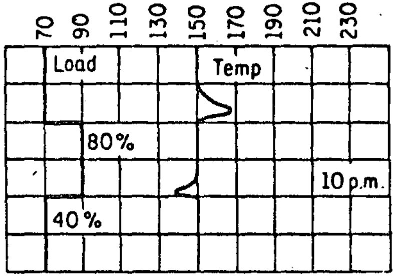


manip τ_1

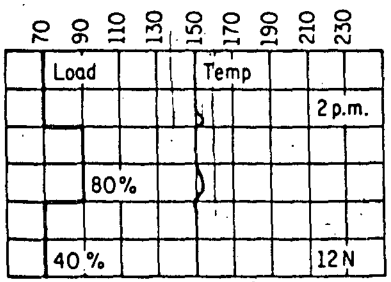
FIG. 8-7. Linearized block diagram representation of the heat exchanger.



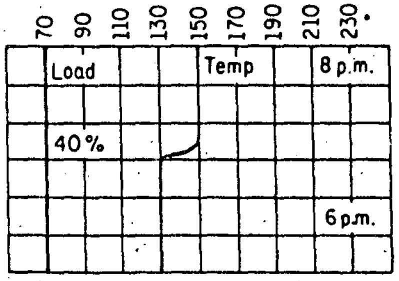
(a) Three-mode feedback control



(b) Steady-state feedforward control



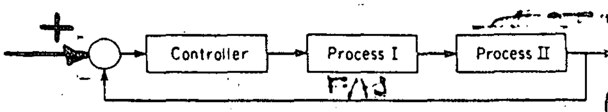
(c) Feedforward with dynamic lag-lead compensation



(d) Set-point effect on lag-lead compensation

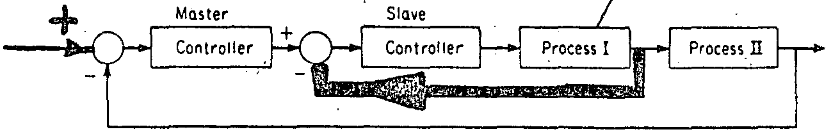
CONTROL EN CASCADA

TAC 6

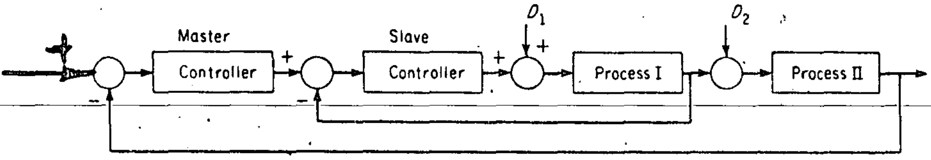


(a) Classical feedback

Faster



(b) Cascade control



(c) Loop showing disturbances

Proceso I más rápido que II

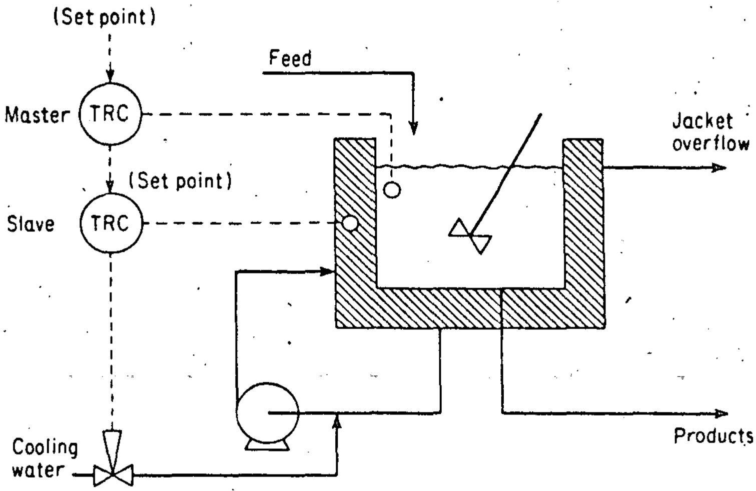
Lazo interior hace a I más rápido

Salida del maestro ajuste del controlador esclavo.

Principal ventaja: Elimina compensa sobre todo disturbios que entran al lazo interior

D_1

Ejemplo de aplicación:



El lazo interior debe ser más rápido que el exterior

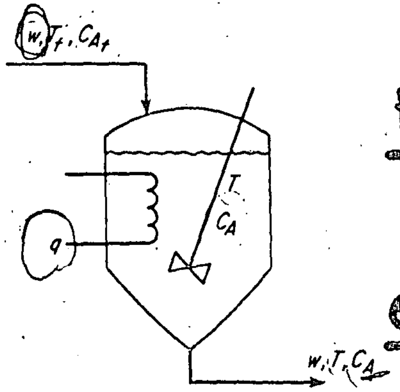
Ajuste:

1.- Maestro en manual y ajuste el interior.

2.- Con el interior en automático ajuste el exterior.

Con un sist. analógico, se requieren dos controladores, con el DIGITAL SOLO UN TRANSDUCTOR adicional. A veces el maestro es solamente el digital

SISTEMAS DE CONTROL MULTIVARIABLE. | TAC 7



$$\underline{r} = \left| \begin{matrix} T \\ C_A \end{matrix} \right|$$

V. por Control.

temp del reactor °F

conc. en el reactor lbmol/ft³

V. DE CONTROL

Calor suminist. BTU/hr

alim al reactor lb/hr

$$\underline{e} = \left. \begin{matrix} q \\ \omega \end{matrix} \right]$$

w = reactor feed rate, lb/hr ✓

T_f = feed temperature, °F

C_{A_f} = feed concentration, lb-mole/ft³

T = reactor temperature, °F

C_A = reactor concentration, lb-mole/ft³

V = reactor volume, ft³

q = rate of heat addition, Btu/hr

ΔH = heat of reaction, Btu/mole A consumed

c_p = heat capacity of reacting mass, Btu/lb °F

ρ = density of reacting mass, lb/ft³

Reaction: $2A \longrightarrow B$

Rate of disappearance of A (lb-mole/hr): $k_0 e^{-a/T} C_A^2$

FIG. 8-10. Chemical reactor.

$$T = f_1(q, \omega)$$

$$C_A = f_2(q, \omega)$$

$$\Delta T = \left.\frac{\partial T}{\partial q}\right|_w \Delta q + \left.\frac{\partial T}{\partial w}\right|_q \Delta w$$

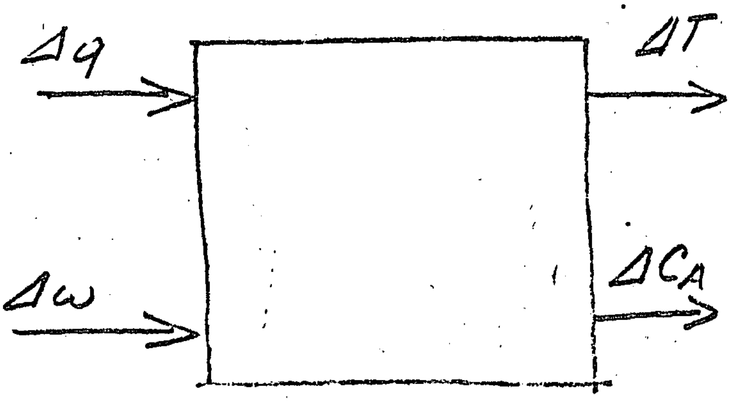
$$\Delta C_A = \left.\frac{\partial C_A}{\partial q}\right|_w \Delta q + \left.\frac{\partial C_A}{\partial w}\right|_q \Delta w$$

The matrix representation of these equations is

$$\begin{bmatrix} \Delta T \\ \Delta C_A \end{bmatrix} = \begin{bmatrix} \left.\frac{\partial T}{\partial q}\right|_w & \left.\frac{\partial T}{\partial w}\right|_q \\ \left.\frac{\partial C_A}{\partial q}\right|_w & \left.\frac{\partial C_A}{\partial w}\right|_q \end{bmatrix} \cdot \begin{bmatrix} \Delta q \\ \Delta w \end{bmatrix}$$

or in matrix form,

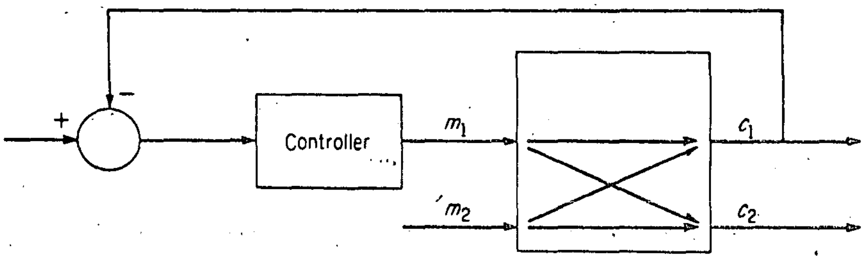
$$\mathbf{c} = \mathbf{M}\mathbf{m}$$



hay acoplamiento.

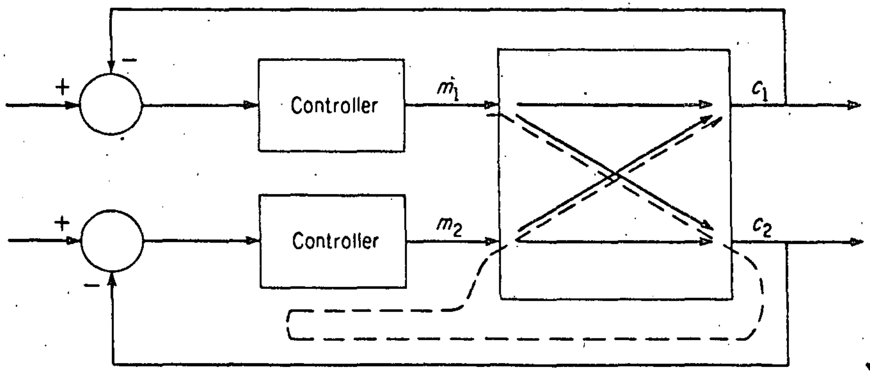
TAC 9

Asociar aquellas salidas con " entradas donde acoplamiento es máximo y emplear lazos de R.A.



(a) Single control loop

1er Ajuste
NO HAY
PROBLEMA

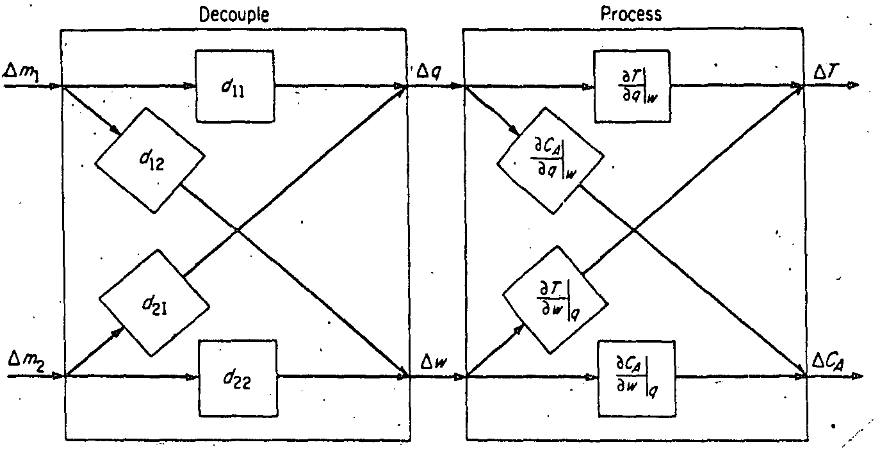


(b) Control loop pair

2do Ajuste.
CAMBIA LA
GANANCIA DEL
1ER LAZO DEBido
al de abajo
sobretodo si
hay mucha interacción.

TAC 10

Solución: Desacople:



Cambio en Δm_1 → solo cambio en ΔT
" " Δm_2 → solo cambio en ΔCA

$$\begin{bmatrix}\Delta T\\ \Delta C_A\end{bmatrix}=\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}\Delta m_1\\ \Delta m_2\end{bmatrix}$$

$$\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}=\begin{bmatrix}\left.\frac{\partial T}{\partial q}\right|_w & \left.\frac{\partial T}{\partial w}\right|_q\\ \left.\frac{\partial C_A}{\partial q}\right|_w & \left.\frac{\partial C_A}{\partial w}\right|_q\end{bmatrix}\begin{bmatrix}d_{11} & d_{12}\\ d_{21} & d_{22}\end{bmatrix} \Longrightarrow \begin{bmatrix}d_{11} & d_{12}\\ d_{21} & d_{22}\end{bmatrix}=\begin{bmatrix}\left.\frac{\partial T}{\partial q}\right|_w & \left.\frac{\partial T}{\partial w}\right|_q\\ \left.\frac{\partial C_A}{\partial q}\right|_w & \left.\frac{\partial C_A}{\partial w}\right|_q\end{bmatrix}^{-1}$$

matriz de desacoplamiento.
(puede no existir el inverso)

En el caso dinámico:

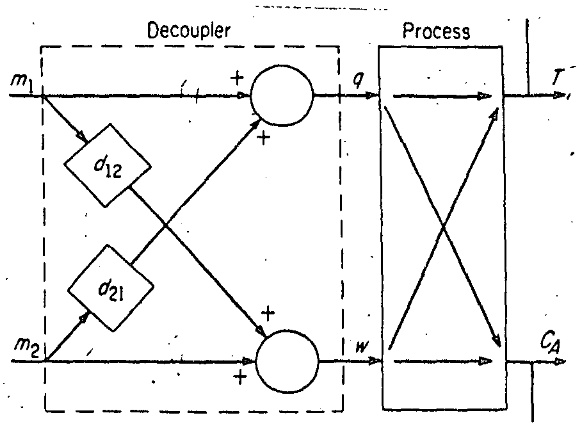
Sea $G_{11}(s)=\frac{1}{\tau_{11}s+1}$ $\quad G_{12}(s)=$

$G_{21}(s)=$ $\quad G_{22}(s)=$

y se desea que el desacoplador sea

$$\begin{bmatrix}1 & D_{12}(s)\\ D_{21}(s) & 1\end{bmatrix}$$

TAC II



$$\begin{bmatrix} K_1(s) & 0 \\ 0 & K_2(s) \end{bmatrix} = \begin{bmatrix} G_{11}(s) & G_{12}(s) \\ G_{21}(s) & G_{22}(s) \end{bmatrix} \begin{bmatrix} 1 & D_{12}(s) \\ D_{21}(s) & 1 \end{bmatrix}$$

$$\Longrightarrow \quad \begin{aligned} G_{11}(s) + G_{12}(s)D_{21}(s) &= K_1(s) \\ G_{21}(s) + G_{22}(s)D_{21}(s) &= 0 \\ G_{11}(s)D_{12}(s) + G_{12}(s) &= 0 \\ G_{21}(s)D_{12}(s) + G_{22}(s) &= K_2(s) \end{aligned} \quad \longrightarrow \quad \begin{aligned} D_{21}(s) &= -\frac{G_{21}(s)}{G_{22}(s)} = -\frac{\tau_{22}s + 1}{\tau_{21}s + 1} \\ D_{12}(s) &= -\frac{G_{12}(s)}{G_{11}(s)} = -\frac{\tau_{11}s + 1}{\tau_{12}s + 1} \end{aligned}$$

Una vez desacoplado puede realimentarse

SYSTEM CONTROL CENTER DESIGN

by

T. E. Dy Liacco

A report prepared for the United States
Energy Research and Development Administration

July, 1975

JOHN M. THORSON, JR.

SYSTEM CONTROL CENTER DESIGN

Tomas E. Dy Liacco

The Cleveland Electric Illuminating Company
Cleveland, Ohio

ABSTRACT

This report reviews the state-of-the-art of the design of system control centers. It summarizes the structures of control centers and the operating philosophies which affect design. It examines the extent to which advanced ideas relating to security and control have in fact been implemented. Important design factors, features, and problem areas which should be considered in the implementation of control center projects are discussed. Trends and needs for further development are presented.

Included in this report is an up-to-date table providing basic information about modern control centers which are presently in operation and which are in the process of development throughout the world.

INTRODUCTION

This report is about System Control Centers--the state-of-the-art, design, implementation, and operating experience.

A system control center is designed and built for system operation. A control center can be built for other purposes as well but our concern is with system operation.

For all the multi-processing, redundant, computer equipment, for all the thousands of pieces of data being monitored and processed, and for all the color CRT's and the dynamic wall display in the control room--what functions are actually being done to help system operation? And, more specifically, the system operator?

With all the advanced, state-of-the-art hardware, how advanced is the application software? To what extent and how effectively have security-oriented functions been integrated into the system control center? Is there too much equipment for so little done? Or, not enough equipment for what is attempted to be done?

These are questions of design and application. The purpose of this report is to review the state-of-the-art of system control centers in the light of these questions plus a few more.

What have we learned so far about how to design such control centers? What are the problems of implementation and maintenance and how may they be avoided? What, in general, has been the experienced performance of system control centers? And finally, what development activities in control center design and application look promising?

Nature and Scope of Report

For this report I consider system control centers dedicated to the operation of at least, the generation and transmission systems. Centers solely for distribution systems are not included in my investigation. Also not included are strictly supervisory control (so-called SCADA) centers and strictly generation dispatch centers.

In Table I, Appendix A, is a list of system control centers in operation or being built. These centers represent the state-of-the-art. To obtain as much first-hand information as possible for this report, I visited twenty control centers which are in operation, twelve of them in the United States, four in Japan, and four in Europe. Information on control centers which I could not visit and those which are still not in service was obtained by communicating directly with the utility engineers having direct responsibilities for the respective projects. In a few cases of recently authorized projects I have relied on information furnished by the vendors.

Actually this fact-finding and information exchange with utilities as well as with hardware manufacturers, systems vendors, consultants, research organizations, and universities, in the US and abroad, have been continuing activities of mine for the last five years. These activities are directly related to my responsibilities at CEI and to my participation in IEEE, EEI, EPRI, and ERDA in the areas of system planning, system security, and control. The comments and opinions that I make in this report are based on this accumulated background information.

The most recently published summary of the state-of-the-art of system control centers was the paper I wrote for the July, 1974 issue of the Proceedings of the IEEE.[54] This present report is, in a sense, an outgrowth and expansion of that previous survey.

Discussion of control functions in this report is restricted only to those carried out at system control centers. Local controls at plants or substations are not included although it should be realized that they are part of the overall control hierarchy for operation.[4,38,62]

There are also many functions being done in system control centers for operation planning purposes. These are run in an off-line or batch processing mode or via a remote terminal linked to a large computer center. These functions were not investigated for this report. This is not to say that these planning functions are not important but, rather, that their discussion properly belongs to some other report. My purpose is to review functions which are integrated into the real-time environment of the computer system in system control centers. I will however touch upon the general requirements of doing batch processing in the real-time computer system.

It is not within the scope of this report to discuss the theory and the algorithms of various security control functions. While algorithms may be fairly well understood and tested on paper, their execution in real-time is a different matter altogether. This report addresses itself to the problems of real-time implementation.

This report consists of discussions on the following topics:

- Overview of the State-of-the-Art
- Functions Implemented at System Control Centers
- Design Structures and Criteria
- Data-Acquisition and Control Subsystem
- Computer Subsystem
- Man-Machine Interface Subsystem
- Software Subsystem
- Steps in the Implementation of a System Control Center
- System Maintenance and Enhancement
- Trends in Control System Design
- Problem Areas and Research Needs
- Bibliography

- Addendum - Needs in Related Areas

- Appendix A
  Table I - System Control Centers for Generation-Transmission Systems
- Appendix B
  Summary of Guidelines and Pitfalls to Avoid
- Appendix C
  Acknowledgments

OVERVIEW OF THE STATE-OF-THE-ART

In simple terms, the goal of system control center design is the implementation of security control.

Security control requires the proper integration of both automatic and manual control functions, i.e., a total systems approach with the human operator being an integral part of the control system design. Security control requires that all conditions of operation be recognized and that control decisions by the man-computer system must be made not only when the power system is operating normally, but also when it is operating under abnormal conditions.

Basic Security Control Concepts

The nature of security control in terms of a design organization or structure was presented at the 1966 IEEE Summer Power Meeting in the paper, "The Adaptive Reliability Control System."[4] The concepts advanced in this initial paper were further developed in a suceeding work, "Control of Power Systems via the Multi-Level Concept" published in 1968.[6] I will briefly discuss the basic ideas originally presented in these two references.

The power system may be assumed as being operated under two sets of constraints: load constraints and operating constraints.

The load constraints impose the requirement that the load demands must be met by the system. The operating constraints impose maximum or minimum operating limits on system variables and are associated with both steady-state and stability limitations. Mathematically, the load constraints can be expressed in the form of the familiar load flow equations. The operating constraints can be expressed in the form of inequalities, such as on equipment loadings, bus voltage, phase angle differences, generator real and reactive powers, etc.

The conditions of operation can then be categorized into three operating states -- normal (or preventive), emergency, and restorative.

A system is in the normal state when the load and operating constraints are satisfied. It is reasonable to assume that in the normal state the power system is in a quasi-steady-state condition. For any given time, the intersection of the load constraints and the operating constraints defines the space of all feasible normal operating states. The power system may be operated anywhere in this space.

A system is in the emergency state when the operating constraints are not completely satisfied. Two types of emergency may be noted. One is when only steady-state operating constraints are being violated, e.g., an equipment loading limit is exceeded or the voltage at a bus is below a given level. The other is when a stability operating constraint is violated and as a result of which the system cannot maintain stability. The first type of emergency may be called "steady-state emergency" and the second type, "dynamic emergency." For the moment, however, we shall not distinguish between the two types of emergency.

A system is in the restorative state when the load constraints are not completely satisfied. This means a condition of either a partial or a total system shutdown. In case of a partial shutdown the reduced system may be in an emergency state. This is the start of a cascading situation and, if uncorrected, would lead to a further deterioration of the system.

A normal operating point can be classified as being either secure or insecure with reference to an arbitrary set of disturbances or next-contingencies. A normal system is said to be secure, i.e., at a secure operating point, if it can undergo any contingency in the next-contingency set without getting into an emergency condition. On the other hand if there is at least one contingency in the next-contingency set which would bring about an emergency the normal system would be called insecure.

My choice of terms has been arbitrary. Other ways of describing the same operating problem are just as valid. For instance, what I have characterized as an insecure normal state is defined in Reference 19 as a fourth operating state called "alert."

The concept of three operating states breaks up the complex operating problem into three operating subproblems with different control objectives. Of primary interest and of major impact on the design of system control centers is the control done in the normal state. It is basically the development and implementation of functions in this area that represent the state-of-the-art in system control centers. Emergency and restorative controls are needed for a complete security control system, but so far their implementation at control centers has been very limited in scope and in ingenuity.

General Characteristics of System Control Centers

It should not be inferred from the foregoing discussion that the state-of-the-art in system control centers has been due solely to considerations of system security. We should recall the previously stated objective of system control center design as the implementation of security control in the broad sense of integrating all required automatic and manual functions for all conditions of operation. From this perspective we can see the general patterns in which system control centers have been developing in the last five years.

The necessity for integration has brought together the previously separately implemented functions of generation control and transmission control, into one system. For geographically small power systems the integration is carried out in the system control center. For large systems or systems with existing regional or area control centers this integration is accomplished by linking the centers at various levels into a computer hierarchy.

In addition, the integration of automatic and manual functions is being manifested in the form of advanced display devices and techniques. The CRT with limited graphics has become the universal man-machine interface for system control centers.

Operating decisions by the human operator are being supported by the presentation of a more complete and coherent information about the power system than was ever done before. Advanced programs making use of real-time data are being developed and applied to aid the operating decision process.

Digital computers which allow the easy integration of many functions also make possible the implementation of more sophisticated logic and processes for automatic generation control and economic dispatch. Thus we see analog dispatch controllers slowly disappearing from the scene and optimum power flows possibly replacing transmission loss B-coefficients for economic dispatch.

It is of interest to refer to a previous state-of-the-art review of automation of power systems made in 1965 by N. Cohn.[1] In that review it was pointed out that for system operation there were two functions to be implemented:

1. area regulation (which we now call automatic generation control)
2. economic dispatch.

Cohn reported that "some two dozen or so large centralized analog computer controls" to carry out these two functions were "currently in operation, or in the process of installation." He further reported that several digitally-directed analog systems and some direct digital controls were in operation or being installed. Also mentioned as a noteworthy feature was the solid-state analog console replacing earlier types using electromechanical elements. That was the state-of-the-art in 1965.

Let us now take a closer view of system control centers which have been placed in service over roughly the last five years and also those in the process of implementation. And let us restrict ourselves only to real-time features and functions.

As of this writing there are approximately 40 system control centers in operation or under construction throughout the world which fall in the category of the new generation of modern control systems. These centers are listed in Table I, Appendix A.

The totality of real-time features and functions at these centers include the following:

1. Hierarchical structure consisting of several levels of computer systems.
2. Dual real-time processors or multiprocessors plus redundant peripherals.
3. High-speed digital telemetry and data-acquisition equipment.
4. System-wide instrumentation of electrical quantities and device status.
5. Color CRT's with graphics for interactive display.
6. Dynamic wallboard group display.
7. Automatic generation control. ✓
8. Economic dispatch calculation. ✓
9. Automatic voltage (var) control.
10. Supervisory control (breakers, capacitors, transformer taps, generating unit startup and shutdown).
11. Security monitoring. ✓
12. State estimation. ✓
13. On-line load flow. ✓
14. Steady-state security analysis. ✓
15. Optimum power flow. ✓
16. Automatic system trouble analysis.
17. On-line short-circuit calculation.
18. Emergency control -- automatic load shedding, generator shedding, line tripping.
19. Automatic circuit restoration.

There is no control system that has all of the functions just enumerated. This is to be expected. Operating problems differ due to different networks and generation resources. Operating philosophy and the structure of operating responsibilities are not the same for all companies. A few centers have adopted an evolutionary approach, adding something new to existing control equipment and telemetry. Finally there is the significant time gap between the testing of a new idea on paper and its implementation in a real-time control system.

FUNCTIONS IMPLEMENTED AT SYSTEM CONTROL CENTERS

In this section we shall summarize the functions that are presently being carried out by the digital computers at the control centers in Table I, Appendix A.

Automatic Generation Control (AGC)

The automatic generation control (AGC) function is, with very few exceptions the only closed-loop control being implemented at system control centers. Some companies in Table I still are using analog systems for the AGC, either digitally-directed for base point settings or completely independent of the digital computer. In Japan, only Hokuriku Electric Power has implemented AGC digitally. The EDF National Control Center in France, the RWE Center in West Germany, and the Laufenburg Center in Switzerland, are all using analog AGC's.

In England, at the CEGB, there is no requirement for automatic generation control. In the US all of the system control centers, except for one, have or are planning to have digital AGC. Of those control centers with digital AGC some have completely disabled their old analog control systems while others have retained some kind of analog control for backup. Still others have analog hardware which merely determines the area requirement. In both types of analog backup, an independent analog telemetry is used for inputs.

The sampling time for digital AGC varies from 1 second to 4 seconds. Most control centers send raise and lower signals or MW deviations to the generating units. A few send the desired MW outputs to the units.

The use of plant computers communicating with the system control center offers flexibility for carrying out the AGC function. An example of this application is at the Cleveland Electric Illuminating Company (CEI). The AGC software at CEI's system control center sends desired MW signals for each regulating unit to the plant computers. The plant computers act as local closed-loop controllers for each unit.[33] The control algorithms at the plant computers recognize the individual rate of response of each unit. Over the same data links the plant computers report to the system control center every second the control status of each unit and its short-term raise and lower capability. This information is used by the AGC algorithm such that the desired MW requested is within the dynamic capability of the unit. The computer-to-computer link also handles special requests by a unit operator to place a unit off or on regulation or to change a unit's operating limits.

The basic AGC algorithms, i.e., the calculation of area control error and the assignment of regulation to each unit recognizing the desired base points, are well-known. To apply these algorithms in a system control center requires the addition of modules which in effect interface with the real-time environment. These modules should take care of initializing the AGC function, coordinate all information from other programs which affect AGC, prepare and hand off to the data-acquisition subsystem the signals to be sent to the plants, and communicate with the display subsystem.

The use of digital AGC has resulted in benefits due to less frequent pulsing of units as contrasted to continuous analog control. This has kept many more units on regulation and has virtually eliminated situations of plant operators taking units off regulation because they were being moved around too much. Faster individual unit response has also been realized in going from analog to digital. However it is not entirely clear what the overall regulating performance of digital AGC has been. System operators, in general, are satisfied. But we need a more objective evaluation. An investigation into the performance of existing AGC algorithms to identify any basic problems and to seek further improvements utilizing the flexibility of the computer is well-justified.

Economic Dispatch Calculation (EDC)

Economic dispatch calculation is performed every few minutes using the set of coordination equations which requires that the incremental cost of delivered power from each generating unit to an arbitrary reference point be the same for each unit. The incremental cost of delivered power to a given point from a generating unit is equal to the incremental cost of generated power multiplied by a penalty factor. Traditionally the penalty factors are calculated using transmission loss B-constants.

Except for CEI, all of the control centers with EDC use B-constants. These are calculated off-line and usually only one set of constants is stored in the control center. There is no on-line update of B-constants. Updates are made, off-line, on long range network changes. Some companies store two or more sets of either pre-calculated B-constants or pre-calculated penalty factors.

At CEI the calculation of penalty factors is done on-line using a real-time optimum power flow.[33] Everytime there is a network change or when the system load has changed significantly in magnitude or in relative distribution between areas, the optimum power flow runs automatically and a new set of penalty factors is passed on to the EDC. The penalty factor calculation takes less than 40 seconds on the Sigma 5 computer. This is the total response time and includes network configuration update, 3 to 4 fast decoupled load flows,[50] Jacobian calculation at the optimum solution point, calculation and transfer of new penalty factors to the database.

Some centers are planning to use the optimum power flow for the same purpose.

Although EDC should be made only for those units which are regulating, it is desirable to make another calculation including all the other units on local control. This second-pass EDC is made everytime the regular EDC is run. The results of the second-pass EDC are displayed to the operator so that he may manually direct the units on local control to be moved closer to their optimum generating points. Considerable additional economy may be realized this way.

One company is planning to replace the EDC coordination algorithm and the use of variable participation factors with what is deemed to be a simpler scheme. Basically the units will be ordered by increasing incremental costs. Raise-generation requirements will be assigned to the units at the low-end of the ordered sequence and lower-generation requirements to the high end of the sequence.

Some operators find EDC and AGC to be at odds with each other at high loading periods. That is, the efficient, fast-responding units are out of regulating capacity in the upward direction because of EDC, and regulation is being done by the older, slower-responding units. It would appear that to obtain better regulation under these circumstances one would have to sacrifice economics or else sacrifice regulation for the sake of economy. This looks like another trade-off problem. EDC and AGC should not be a hopeless contradiction.

Automatic Voltage/Var Control (AVC)

The automatic control of system voltage and of var allocation has been in service in Japan for several years now. There is no such control as yet in the US or in Europe. The Potomac Electric Power Company (PEPCO) is planning an automatic voltage control[51] for their system control center which is now under development.

The AVC in Japan regulates the voltage profile and also minimizes losses due to reactive power flow.[12] The control variables are generator reactive powers, transformer taps, shunt capacitors, and shunt reactors. The control is a two-step operation. Voltages and var flows are checked periodically and when there is any deviation beyond certain tolerances the voltage profile control calculation is initiated. At less frequent intervals the minimum loss calculation and control is executed.

Security Monitoring (SM)

Security monitoring (SM) is the on-line identification and the display of the actual operating conditions of the power system. This one function has made the difference between the traditional dispatch center and the modern system control center. SM requires a systemwide instrumentation on a greater scale and variety than that required by a center without SM. The types of measurements include: MW and MVAR flows, branch currents, bus voltages, bus MW and MVAR injections, frequencies, energy readings, circuit breaker status or operation counts, manual switch positions, protective relaying operations, transformer tap positions, and miscellaneous substation status and alarms.

The SM function, in general, checks the analog values against limits basically to determine whether the system is close to, or at, the emergency state. The limit-checking also allows some kind of data validation and the rejection of incongruous data. Limit-checking is done as often as the data is brought in which is usually in the order of every one to a few seconds. In Tokke (Norway) power flows are monitored every 80 milliseconds in order to detect oscillations.

The display required for SM entails the use of CRT's and a large number of display formats. The dynamic wall display is also used for SM. Part of the SM function is the on-line determination of the network topology.[39,42] In most cases it is sufficient to determine the network configuration. In centers where there is a direct responsibility for transmission switching and safety is a paramount factor, the SM function should include an identification of the electrical status (energized or de-energized) of every physically isolatable segment. At CEI this is accomplished by a network status analysis program.[42] While in other centers it is necessary only to display the fact that one or more ends of a circuit is open and that the circuit cannot conduct a power flow, at CEI the electrical status of the circuit is determined and displayed. The network status analysis program is a straightforward tracing routine.

Static State Estimation (SE)

State estimation (SE) may be defined as a mathematical procedure for calculating, from a set of system measurements, a "best" estimate of the vector of bus voltage magnitudes and phase angles of the network.

The measurement set is understood to contain an adequate degree and spread of redundancy to allow the statistical correlation and correction of the measurements, detect and preferably identify bad data, and yield calculated values for non-telemetered quantities.

As stated previously, a discussion of mathematical procedures is not within the scope of this report. An excellent summary of SE and its methods is given in the 1974 Proceedings of the IEEE paper by Schweppe and Handschin.[53] In referring to methods in use at system control centers I will adopt the terminology of this reference.

In the world today, to the best of my knowledge, there are five system control centers with SE in operational use. I will briefly describe each SE application in the order of approximate time of implementation.

1. Tokke (Norway) - SE has been in operation in this control center since March, 1972.[37] Originally the weighted least squares (WLS) method was used but this was later replaced by the sequential or simplified Kalman filter approach . Although Tokke prefers this second method for their small system they have not obtained equally good results in simulations of larger systems.

The network model consists of 10 busses which is part of the Norwegian bulk power system. The measurement set consists of active and reactive line flows, bus injections, and voltage measurements. The SE runs every minute using the last set of measurements which are scanned every second. In addition, SE is started whenever there is a network change. The purpose of the SE is to: obtain the vector of bus injections; check for abnormal metering errors. The vector of bus injections is then used by a Newton-Raphson on-line load flow for security analysis and for determining closed-loop corrective control for certain line-outages. Accuracy improvement is not considered to be a major justification for this SE.

Bad data detection is based on the value of the sum of the squared residuals, i.e., the performance index, J. After two consecutive failures of the J-test procedures are initiated for bad data identification based on individual residuals. This involves repetition of the estimation cycle. If this still fails after a few attempts, a logic procedure is initiated for determining network model errors.

2. Laufenburg (Switzerland) - At the Laufenburg control center , SE has been in operation since November, 1973.[56] The AEP or "lines-only" method is used with some slight modifications . SE runs every 15 minutes or on request for major system changes. The size of the model is 46 busses and the number of measurements is 122. The results of the SE is used for SM and the operator is informed of overloads or other critical conditions. The SE results are stored in a historical file for a 7-day period. This data is available to companies whose lines are represented in the SE model. The bad data detection is based on the performance index, J. At present there is no explicit bad data identification routine. On detection of bad data the operator is informed via a suitable message.

The SE routine takes about 1-1.5 minutes on the IBM 1800.

3. Hokuriku (Japan) - At Hokuriku Electric Power Corporation the state variables for SE are the line flows. The measurement set consists of real and reactive power flows, real and reactive bus injections, and bus voltages. The purpose of the SE is for security monitoring and for security analysis of the 145 kV and 275 kV network. The SE consists of two estimators: a DC-type estimator for estimating the real-power flows and an AC-type estimator for estimating both real and reactive power flows. The DC estimator, which has been in service since November, 1973 uses a weighted least squares approach and is run every 3 minutes or immediately on a network change. The number of state variables is 71 and the estimation process takes 0.3 seconds on the TOSBAC 7000. In May, 1974 the AC estimator was added to supplement the DC estimator and is run every 15 minutes. The AC estimator uses a sequential calculation approach and takes 10 seconds to run.

The inputs to the state estimators are 10-second averages of the measurements which are obtained from the system every 2-seconds. The averaging routine takes care of the problem of non-simultaneous measurements. In this connection, the values displayed on the CRT's are the 10-second averages instead of the raw data. SE results are substituted for bad data.

Hokuriku's reasons for a two-step SE are the following:

- The set of locations of real power measurements is not the same as that for reactive. That is, on some lines only real power is measured.

- The reliability required for reactive power data is not as stringent as for real power data.

- The DC-type SE runs so fast that it serves as a quick check for bad data detection and identification purposes.

The experience with the DC estimator has been very good for bad data detection. The bad data detection of the AC estimator is not as good.

4. Interbrabant (Belgium) - At Interbrabant, SE has been in service at the control center since November, 1974, after about a year of experimentation. The SE uses the WLS method and a model of 120 busses representing the 380 kV, 150 kV, and 36 kV system. This is the largest-dimensioned SE in operation. The primary purpose of SE is to obtain the vector of bus injections. The number of measurements is over 350 and is a mixture of approximately 25% real and reactive power flows, 25% real and reactive bus injections, and nearly 50% line currents. There are also 17 voltage measurements. The measurements are scanned every 3 seconds and an averaging routine calculates 15-minute averages of all readings.

SE runs every 15 minutes, or whenever there is a network change, using the 15-minute average of the measurement set as inputs. However the absolute deviations of the inputs are first checked against the last estimate. If the maximum deviation is less than a prescribed value SE will not be run. The SE runs for about 3.5 minutes on the Westinghouse 2500. There is no bad data identification routine as yet.

The bus injections calculated by the SE are used, together with previous SE results, to forecast bus injections for the next 15-30 minutes. Using this predicted bus loads, an on-line load flow is run to yield the base case for security analysis.

5. American Electric Power (US) - AEP developed the "lines-only" approach in 1970.[23] This is considered to be the fastest SE method available. Experimental results were obtained successfully for a portion of the AEP system in September, 1972,[48] but it was not until January, 1975 that SE was placed in service at the control center. This is illustrative of the time gap I mentioned earlier, between the development of an algorithm and its actual implementation in real-time. The purpose of SE is for the security monitoring of the EHV network (345 kV, 500 kV, and 765 kV).

The SE model consists of 63 busses and uses a measurement set of 123 pairs of line flows and one voltage. Measurements are periodically gathered every 5 minutes. In addition measurements are obtained whenever there is a circuit breaker status change. After the measurements are received, a chi-square test is made using the new measurements and the results of the previous SE to determine if a new SE is needed. AEP has found so far that on the average the chi-square test is passed 50% of the time. The total running time of SE is approximately 1 minute on the IBM 1800. AEP believes this could be improved with a faster-access disk.

The results of SE are displayed on a 4-CRT composite picture of the AEP network, showing the real and reactive power flows and the bus voltage magnitudes. On operator request the voltage phase angles or the line amperes can be shown on the CRT diagram. The SE results are also used to update the dynamic wall display. Breaker status and directions of power flows are indicated. Part of the SM function is to check the flows against security limits. When a limit is exceeded the line is shown flashing on the wall display.

The SE uses the chi-square test for bad data detection. If this test fails, indicating bad data, the identification routine is run. The bad data identification has been found to be very good. It does not work for a radial line in the sense that with only two sets of measurements it is not possible to identify which end is in error. In such a case, both measurements are flagged as bad.[66]

AEP points out the value of SE not only for obtaining missing non-telemetered data and for bad data identification but also for providing accurate voltage magnitudes. Knowledge of the actual voltage levels at which the EHV network is operating is of great importance to the AEP operator. This is a significant point about SE application since voltage measurements are generally not of high accuracy.

Relaxing our definition of state estimation, we find two examples of what may be called partial state estimation.

At EDF, a DC-type state estimator, i.e., for estimating phase angles only, has been in use since 1973. This estimator runs every minute or on a network change, and is used for security monitoring of real power flows and injections.

At CEGB, since 1972 there has been in service a systematic logic routine for cross-checking status data with analog measurements in order to validate data and to establish the network configuration. A DC load flow is then run and the results are compared with the measurements. Discrepancies beyond a certain value are noted for the operator to look into.[36]

The above examples of actual working state estimation applications should be convincing evidence that the concepts that Schweppe and others have brought in from estimation theory can indeed be applied to power systems. What is more important is that there are real advantages to be gained. Specifically, these benefits are:

- bad data identification

- calculation of non-telemetered or missing data

- establishment of base case for security analysis

- better quality voltage "readings"

I have identified four, real, practical benefits of state estimation. These are more than enough reasons for recommending the consideration of state estimation as a necessary part of system control centers.

Note that the only application program that uses the results of state estimation is security analysis. Note also that except for the voltage magnitudes improved accuracy is not claimed by the present practitioners to be one of the main benefits of state estimation. Probably the most important aspect of state estimation is bad data identification. This alone could be a worthwhile justification for including state estimation in a system control center.

Referring to Table I, Appendix A we find 11 system control centers planning to have the state estimation function. Of these, the center that is probably closest to implementing SE is Bonneville Power Administration.[47,74]

On-Line Load Flow (OLF)

By "on-line load flow" I do not mean a load flow that is made available to the operator for planning or study purposes. However such a load flow is run, either by conventional batch processing or interactively, it is still an off-line load flow. An on-line load flow (OLF) is one which is used for real-time functions such as security monitoring, security analysis, and penalty factor calculation, and can also be used for study purposes. OLF makes use of real-time data.

In the previous sections we have mentioned cases of OLF applications. Security analysis, which will be discussed later in the next section, consists of contingency evaluation and corrective action strategy. OLF may be used for one or both of these functions.

The OLF requires a vector of bus injections. In the general case, the bus injections are calculated from statistical data obtained on-line and some off-line historical information. In the preceding section we discussed how at three control centers the bus load injections are obtained from the state estimation results. These injections are used as they are or are normalized to produce a set of load distribution factors. These distribution factors may be projected to a future time for predictive purposes.

At present the control centers using OLF are the following:

1. CEGB - The OLF at CEGB is a DC load flow with about 250 nodes. This has been in use since 1970 and, as an operating tool, even before that time. The OLF is run every 20 minutes to establish the base case for security analysis. The OLF is also used for contingency evaluation. Before this function is started, however, the base case OLF results are first compared with the telemetered data and any discrepancies are brought to the attention of the operator.

2. Houston Lighting & Power - The OLF at this center is a Gauss-Seidel load flow for 344 busses and has been in use since 1972. The OLF is used for contingency evaluation and is run only upon operator's request.

3. Tokke - The OLF at this center uses the Newton-Raphson method, the injections being obtained from the SE. The OLF combined with some logic for corrective action strategy in security analysis. This OLF has been in service since 1972.

4. Commonwealth Edison - At this center a Newton-Raphson load flow is used for OLF and has been in service since 1973. The system size is 500 busses. Automatically, every 10 minutes after the hour, OLF runs to generate the security analysis base case and also to perform the contingency evaluation. The base case run takes about 2 minutes on the Sigma 5.

5. CEI - The OLF at this center has been in service since 1974 and is the first real-time application of Stott's Fast Decoupled Load Flow.[50] Originally, a Newton-Raphson load flow was almost ready for OLF but this plan was discontinued when Stott's load flow was verified to be more efficient in storage and speed. The OLF is for a 230 bus system and the base case takes 5-6 seconds on the Sigma 5. Actually, the OLF is used as a subroutine of an optimum power flow (OPF). That is, the load flow solution is always an optimum solution. Hence the true solution time is more like 40 seconds. In a study mode it should also be possible to run the OLF as a straightforward load flow but this has not been implemented yet.

   The purpose of the OLF, or more correctly, OPF, is to establish the base case for security analysis and for penalty factor calculation. The OLF routine will also be used for contingency evaluation in security analysis.

   The bus injections to the OLF are obtained from distribution factors derived from 5-minute averages of real and reactive injections. Non-telemetered injections are calculated using distribution factors obtained off-line.

6. Interbrabant - The OLF at this center which has been in service since 1974 uses the Z-Bus method with triangular factors rather than an explicit Z-Bus calculation. The OLF is run automatically every 15 minutes following SE and the bus load forecasting routine, as described in the preceding section. The approximate running time is 6 seconds on the Westinghouse 2500. After the base case run the OLF is used for contingency evaluation.

7. EDF - Presently a DC load flow is used as the OLF. This is used for contingency evaluation in security analysis.

The on-line load flow is a necessary function for system control centers. It should not be interpreted, however, as supplanting state estimation. As we have seen, these two functions serve different needs. Since the on-line load flow uses bus injections which are statistical in origin, the ultimate OLF should give results with some kind of statistical interpretation, i.e., an stochastic load flow. We are not yet there with the present state-of-the art. However, the basic formulation of the OLF for penalty factor calculation, for establishing the base case of security analysis, and as an alternative method for performing contingency evaluation is of value now at system control centers. The stochastic load flow would be valuable for predictive-type analyses and also for security monitoring in the absence of state estimation.

Due to be in service in the near future is the OLF for Middle South Services. This will be a Stott's decoupled load flow of about 900 busses. The program is sized for 1200 busses. The OLF will also be used for contingency evaluation.

Lastly, a word about the industry's only experience with a hybrid load flow analyzer. In 1970 the New England Power Exchange installed a hybrid system to do the on-line load flow analysis. The purpose of this pioneering installation was to gain experience for possible future uses of the hybrid concept. For the last year or so, NEPEX has not been using the hybrid due to a severe component maintenance problem and the difficulty of making updates due to hardware inflexibility.

Steady-State Security Analysis (SA)

The first function of security analysis (SA) is to determine whether the normal system is secure or insecure. The second function is to determine what corrective action strategy should be taken when the system is insecure.

The first function is commonly known as contingency evaluation since, by definition, the security of a system is determined with reference to a set of next-contingencies. In present state-of-the-art, only steady-state contingency evaluation is done at system control centers. That is, the emergency condition that is to be avoided is overloading of equipment or poor bus voltages. There is still nothing in the way of dynamic security analysis. A recent paper[50] reports on the results of a study by Tokyo Electric Power Company (TEPCO) on fast, dynamic security analysis. The results look promising, in fact, exciting, but TEPCO has no plans as yet for implementing this function on their control center, not until after further studies are made.

The earliest method used for contingency evaluation is the distribution factor method derived from elements of the X-bus matrix.[16,18] This method is used at: Michigan Electric Power, PJM, Tokke, New York Power Pool, and Philadelphia Electric.

The DC load flow is used for contingency evaluation at the CEGB and EDF control centers.

At Houston Lighting & Power, the Gauss-Seidel OLF is run on demand to evaluate a single contingency. Each contingency requires a separate request.

At Commonwealth Edison, the Newton-Raphson OLF makes a contingency analysis every hour. Twenty contingencies are examined and there is a built-in flexibility for manually changing some contingencies in the set. The whole process of contingency evaluation takes about 20-40 minutes. A unique feature at this center is the partial on-line update of the interconnection equivalent. A Ward equivalent is developed off-line, as a planning responsibility, for a normal external configuration with critical lines retained. Every time the contingency evaluation is started the OLF in the process of establishing the base case adjusts the equivalent injections that the tie-line flows in the model match the measured tie-line flows. External network changes are handled manually.

At Interbrabant the OLF is run every half hour for contingency evaluation with the aid of the superposition principle. A standard set of about 40 contingencies is used. The contingency evaluation can also be run as often as the SE is run, which is every 15 minutes. At present, two sets of interconnection equivalents, developed off-line, are stored in the computer. These represent two different operating conditions. It is planned to store one or two more equivalents.

Contingency evaluation by an OLF should be improved with the use of Stott's Fast Decoupled Load Flow. This application is nearing completion at CEI.

There is so far, in system control centers in operation or under development, no approach to the steady-state equivalent other than the traditional Ward equivalent. There is also no method yet developed for obtaining the interconnection equivalent in real-time. The closest to an on-line equivalent is that of Commonwealth Edison.

As discussed in the Introduction, the space of feasible normal states may be partitioned into secure and insecure regions. This, of course, is a dynamic situation. As the system generation, load, and topology change so does the space of normal states and so does the boundary between secure and insecure regions. In fact, either region could be a null subspace. Clearly, as system conditions change the contingencies in the next-contingency set which yield insecure operating points also change. If at times the system is very strong that no contingency in the next-contingency set can cause an emergency, the insecure region is null and contingency evaluation is not required. At other times only certain contingencies need be evaluated. This leads us to the idea that we should have a more scientific or systematic way of determining, every time we do contingency evaluation, which contingencies we should be looking at. We might call this desired function, "adaptive contingency evaluation."

It is quite common for power system networks to have multi-terminal lines, such as lines with a tap for a transformer connection. For a 3-terminal line, a line outage would mean an outage of three load flow branches and the isolation of one node. This fact is often lost sight of by a software designer with little power system background. The contingency evaluation program gets erroneously developed on the basis of a line outage being a branch outage in the load flow sense.

The second part of security analysis is "corrective action strategy." If the system is insecure, can it be made secure? If so, how and at what cost? If we don't take corrective action now, when will conditions improve? Suppose the system is now insecure and there is no way of making it secure, how much load would be shed or have to be shed in case the causative contingency does occur?

These questions are actually related to operating policy. In some organizations such as EDF and CEGB it is a matter of policy to keep the system secure regardless of the extra cost of doing so. The rule is that the system must always be operated on an "n-1" basis. The preventive action is determined by running security constrainted optimization programs in an off-line computer, several hours to a day in advance.

Optimum Power Flow (OPF)

An optimum power flow (OPF) is a steady state solution to an optimization problem where the load flow equations and limits on system variables and on functions of these variables constitute the set of constraints.

At present the only system control center with an OPF in service is CEI. The OPF is used, with the OLF as a subroutine, to produce a real-time base case for penalty factor calculation and eventually for security analysis. The OPF at CEI is a simplified formulation in that inequality constraints related to system security cannot be handled.

What is needed at a system control center is an OPF which can be used for determining corrective action strategies. In spite of the large amount of work already done in optimization techniques for power systems,[55] there is no application as yet for security-constrained optimization.

Automatic System Trouble Analysis (ASTA)

The Automatic System Trouble Analysis (ASTA) function is unique with CEI. ASTA is a logical procedure for analyzing circuit breaker trippings and reclosings and protective relaying operations[14]. ASTA identifies:

Faulted circuits, permanent and temporary
Primary and backup relay operations
Breaker failures
Breaker misoperations
Primary relay failures
Primary relay misoperations

Besides CEI, the only other control centers which monitor protective relaying operations are in Japan. At Kansai, transmission line relays, primary and backup, are monitored. At Hokuriku, transmission line, transformer, bus, and generator relays, primary and backup are monitored. In both centers, relay operations are displayed on a CRT list or on the wall display. No other processing is done.

On-Line Short Circuit Calculation (OSC)

The function of an on-line short circuit calculation (OSC) is presently unique with CEGB. OSC is run automatically every 30 minutes or on demand to determine the maximum short circuit duty at each bus and compares the number with the switchgear rating. An alarm is initiated when excessive short circuit duties are found and a suitable display is presented to the operator. CEGB has a considerable amount of configuration flexibility plus a wide variation of generation patterns.

A similar application is being planned by RWE.

Emergency Control (EC)

There are a few scattered examples of emergency control (EC) in service at system control centers. These are closed-loop controls which automatically initiates load shedding, generation shedding, system splitting or line tripping in order to relieve overloads, to restore generation-load balance, or to prevent cascading situations.

Load shedding by computer is quite common in Japan. This is in addition to local underfrequency relays. The purpose of this emergency control is to relieve overloads on transmission lines and transformers and also to maintain generation-load balance in subareas. Overload relay operations indicating one or more loading levels are monitored by a control center and the corrective action strategy is determined. If load shedding is required the signals are sent to the appropriate stations. Presently, in Japan, this function is implemented on a local control center or district office basis, except at Hokuriku. The EC at Hokuriku has been in service since 1973 and includes both load shedding and generator shedding. In order to avoid misoperations, the load shedding signals are supervised locally at the receiving station by underfrequency relays or by overload relays depending upon whether the need to shed load is to maintain generation-load balance or to relieve overloads.

At Tokke there are two EC functions. One is automatic generator shedding and/or line tripping on the occurrence of certain line outages in order to avoid overloads. The other is the automatic reduction of generation if power oscillations are detected on the transmission lines. For this second function, power flow measurements are taken every 80 milliseconds.

At CEI a limited EC is in service. Under certain outage conditions it is possible for the loadings of a specific set of lines to exceed their emergency ratings. Should this happen the EC will act like a backup overcurrent relay with a pickup setting and a definite time delay of 50 seconds. After the time delay the EC will send a signal to trip the appropriate breaker which would relieve the overload.

Automatic Circuit Restoration (ACR)

Automatic circuit restoration (ACR) is not in service yet but it is a function that is worth mentioning to illustrate the potential of a system control center for some type of restorative action. Strictly speaking the restoration of a circuit is not the same as restoring load but, clearly, circuit restoration should benefit system security. At CEI the function is planned to supplement local automatic reclosing functions. If local reclosing is unsuccessful the ACR will attempt to restore a circuit after making all sorts of checks to ensure that it would be safe to reclose. This function is superior to a second-shot automatic reclosing since it is based on a more comprehensive picture of the status of the system.

Supervisory Control (SBC, SVC)

Supervisory control is not a new operating function. Its integration into a system control center is new. Since supervisory control is a manual function it is exercised via the man-machine interface or the display subsystem. Seven of the control centers in service have supervisory control of circuit breakers (SBC). Some of these also have supervisory control of voltage regulating devices (SVC). There are some other uses of supervisory control such as for starting and stopping of units.

With supervisory breaker control the operator at a system control center has the ability to open or close breakers for load shedding and load restoration as a manual type of emergency control. SBC may also be used for system splitting.

DESIGN STRUCTURES AND CRITERIA

Computer Hierarchies

One distinguishing feature of modern system control centers is the implementation of computer hierarchies. There are several 2-level hierarchies consisting of a system center at the top level and division (or member company) centers at the lower level. There are 3-level hierarchies such as that of Middle South Services or of General Public Utilities. The latter is in turn a part of a 4-level hierarchy with the PJM center at the top level.

Progress in data communication technology has fostered the growth of computer-to-computer information exchange. System control centers are just beginning to tap the potential of these data links for security monitoring purposes.

Elements of a System Control Center

A system control center consists of the following elements or subsystems: data-acquisition and control; communications; computers; display; software; uninterruptible power supply; the building; and people. The communication channels, the power supply, and the building facilities design are all important to the proper functioning of a control center but I will not discuss them in this report. Their design requirements are not as intimately woven in with the control design problem as those of the data-acquisition, the computer, the display, and the software subsystems. The successful implementation of functions such as those described in the preceding section, in terms of the control center hardware and software elements, is a difficult design problem.

Design and Performance Criteria

Before we look at the design considerations for each of the component subsystems we should first examine some overall criteria.

The three most important design and performance criteria for a system control center are: system response, system availability, and system maintainability.

System response is measured in terms of response time. Immediate response to an operating problem should be a basic capability of a system control center. Fast response is what makes a real-time system "real-time."

Response or response time is the length of time it takes from the instant a function is requested until the instant the outputs from that function are available. The response time requirement depends upon the nature of the function in question. The actual response time obtained depends upon the speed of the hardware facilities, the speed of program execution, and the length of time taken up by waiting in queues and by interruptions by higher-priority functions.

In a system control center the response time requirements may be divided into two broad categories corresponding to critical and non-critical functions. Critical functions require fast response times in the order of milliseconds to a few seconds. Non-critical functions can have slower response times, in the order of several seconds to a few minutes.

System availability is measured in terms of the availability of operating functions. System availability depends upon, but is not the same as, either hardware reliability or software reliability, or both combined. The measure of system availability also depends upon system response. For any given function, the availability, A, is given by:

$$A = \frac{\text{Available Time}}{\text{Period of Interest}}$$

$$= 1 - \frac{\text{Unavailable Time}}{\text{Period of Interest}}$$

Unavailable time is the total time during the period of interest when the function is not available. Any time beyond the maximum prescribed response time of a function is also considered as unavailable.

It is an extreme design requirement to specify the same availability for all functions, critical and non-critical. An overall system availability is difficult to define let alone measure. Besides it does not necessarily ensure a good response and availability of a critical function. Availability of a single critical function or of a number of critical functions is a more tractable concept. A critical function which should have a high availability is the man-machine interface. The availability of this function could be used as a simple, readily measurable criterion for system design and performance. The thinking behind this is that as long as the man-machine interface is available the operator is not completely helpless. Even if other operating functions are not working, the operator could do something manually if the interface is there to provide some information and to permit manual corrections.

It would also be reasonable to specify different availabilities, such as one for critical functions and another for non-critical functions.

Achievement of good response and high availability should be pursued from the very start of system design, through implementation, and during the life of the system. Very much a factor in this achievement is system maintainability. The levels of response and availability are obviously affected by hardware and software maintenance. The repair times following hardware or failures depend upon the maintenance capability, diagnostic aids, and equipment that are available to maintenance personnel. Preventive maintenance, system debugging, corrections, updates, tests, and enhancements are on-going activities which have to be performed using the computer system facilities. The system design from the very beginning should provide for this type of work to be done at any time with little or no impact on the performance of the real-time system.

Software maintenance could be a serious problem in a system control center. Even with an adequate staff of trained people, it is highly advisable to have enough computerized maintenance and testing aids in order to reduce significantly the time required to do maintenance. Just like system response and availability, system maintenance must be designed into the system at the start and not as an after-thought.

THE DATA-ACQUISITION AND CONTROL SUBSYSTEM

The data-acquisition and control subsystem consists of: remote terminal equipment for interfacing with power system instrumentation and control devices; interfaces with communication channels; and master station equipment for interfacing with the system control center. In some centers a dedicated channel is assigned to each remote station. In others there are less channels than remote stations requiring more than one remote to share a channel. Analog data is scanned periodically in the order generally of 1 second to a few seconds. Each scan is triggered by the system control center at the prescribed interval by issuing a request to all remote stations to send in data. Data is received at the master equipment in a random order. The hardware equipment which converts the bit-serial data into a bit-parallel word does error-checking and raises an interrupt to the computer for each word received. There are two approaches to this: one is to have a single interrupt for all channels; the other is to have one hardware interrupt for each channel. The single-interrupt method requires polling by a software routine to find out where the data word came from. The multiple interrupt approach results in a much better response time due to the very fast interrupt processing.

Status data is also processed in the same way as analog data except that there are two ways of reporting status changes. The first way is to send in all status information from all remotes at the required intervals regardless of whether or not there has been a change. This approach requires a software routine at the system control center to check each new status with the old status to determine any changes. Considering the very large number of status points that is monitored in a power system this approach represents a sizable burden on the central processor at the control center. The second way is to send status data from the remote only when there has been an actual change of status. Since normally the system is quiescent and since, if there are any status changes, only a certain number of stations are involved, the second method results in a better overall system response for the same amount of computer resources. There are, however, many systems in service which use the continuous status scan approach and which apparently are not bothered by this processing overhead. At least, not yet. Having a not-so-frequent scan helps. Assigning data-acquisition to a front-end computer also helps.

The use of front-end computers for the data-acquisition function is a desirable option as it off-loads the main computers which would be doing the rest of the real-time functions. In some applications the front-end computer serves only as a message switcher. This does not help system response.

The data link procedures and word structures are different with each data-acquisition equipment manufacturer and sometimes with different models from the same manufacturers. While this situation creates a constraint on the expansion of an existing system it should not lock in a utility company to an obsolescent model when, due to the fast-changing technology, better and more cost-effective equipment may be available. Microprocessors would resolve this industry problem by making it easier and less expensive to convert from one data format to another The day is not far off when some kind of standard data transmission will be in use by the industry for data-acquisition. There is already standardization in computer-to-computer communication in the general data-processing field and in power system pool centers. There is now a trend to so-called programmable remotes which would inevitably become microcomputers or minicomputers. Eventually the data-acquisition subsystem would be a computer network using a standard data link format and control. As with present-day computer-to-computer communications, the standard data transmission would be or should be synchronous, binary, and character-oriented. There is an industry interest in pushing this standardization forward. Within the EEI Engineering Technical Systems and Computers Committee this is a topic that is actively being worked on.

The data-acquisition software besides managing the collection of data and placing them in computer memory, also performs: error-checking; conversion to engineering units; limit-checking; and interfacing with application programs. For fast response, the data-acquisition software must be: resident in main memory; of the highest hardware priority of all application software; as independent of the operating system as possible, making use of hardware interrupts for scheduling its routines, and doing its own interrupt controls and its own I/O's. The real-time database must also be resident in main memory.

THE COMPUTER SUBSYSTEM

Real-Time Computer Characteristics

A system control center is a real-time system and the computers selected for this application must be designed for real-time. Essentially this means that the computer must have outstanding real-time hardware feature and must have a proven and efficient real-time operating system. Some of the hardware features that have been found to be important at control centers are the following:

-- memory cycle times of 1 microsecond or lower

-- multiple external interrupt structure with a fair number of interrupts

-- fast access disk in the order of less than 20 milliseconds access time

-- multiport memory banks with provision for interleaving

-- memory expandability to, at least, 64k 32 bit words or equivalent

-- direct memory access (DMA) with multiplexor for several peripherals sharing the DMA channel

-- floating point hardware

-- internal interrupts for various trap conditions

-- internal real-time clocks

-- watchdog timer

The majority of the computers listed in Table I has these hardware features.

The actual performance of a computer system for the same hardware depends upon the configuration, the operating system, and the software design.

Computer Configuration

The design criteria of response, availability, and maintainability dictate the use of more than one processor. Placing all functions--real-time monitoring and control, background processing, software maintenance and testing--in a single processor makes it extremely difficult or impractical to obtain a high level of response and availability. One would have to limit the scope of functions assigned to the digital computer (such as use analog for AGC) and also accept a degraded security monitoring function. But this is not representative of the new breed of system control centers that we are investigating, where functions such as AGC, EDC, SM, SA, SE, OLF, SBC are being integrated into one system and CRT response times of 1 second or better are common. There are four control centers in Table I with single processors. One is a very small system without CRT displays, the other three have taken an evolutionary approach to their control center development and are planning future modernizations. For the present all three have analog AGC controllers either working continuously or in backup.

In the majority of the control centers in Table I, the computer configuration used is the "dual" computer system. This is shown in greatly simplified form in Figure 1a. A and B are basically identical computer systems, each consisting of a centeral processor, main memory, and auxiliary memory.

There are several ways in which functions may be assigned to A and B. This depends primarily on the availability requirements. One way would be to say that all functions, critical and non-critical, as well as some types of background processing, must be fully supportable on one computer. This takes us back to the response problem of a single processor. However, in this case the availability would be much better since there is a second computer in stand-by. The more common practice is to share the functions between the two computers. Critical real-time functions would be assigned, say to A, which would be called "primary", and non-critical real-time functions, off-line functions, and background processing would be assigned to B, designated "secondary".

In case of failure of the primary computer, the secondary can assume the critical real-time functions by manual or automatic failover of the real-time interfaces, such as the data-acquisition equipment and

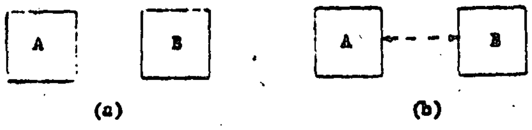


Fig. 1 - Dual Computer Configuration

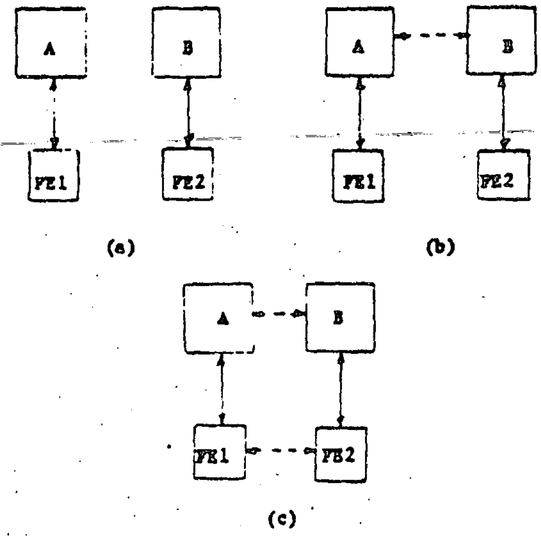


Fig. 2 - Dual Computer Configuration with Front-End Computers

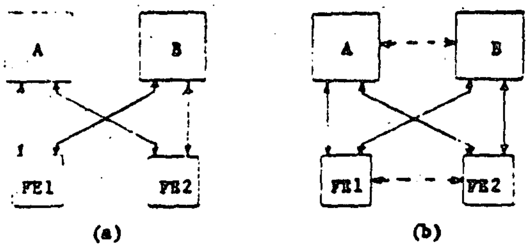


Fig. 3 - Dual Computer Configuration with Task-Sharing Front-End Computers

operators' consoles, and initializing itself to the real-time environment. Figure 1b with the dashed line between computers, represents the failover arrangement.

In actual operating experience, a computer system will recover from most failures by simply starting it over again. Doing this automatically is a desirable design feature. Actually, automatic restart is more beneficial to system availability than is automatic failover. The latter function entails some hardware and software complication and works only if the other computer system is available. Failover also takes longer. A few control centers have both automatic restart and automatic failover. Most have automatic failover only. Definitely worth considering in design is automatic restart only plus manual failover.

The automatic restart routine is designed for a certain number of re-tries. Prior to each trial pre-assigned areas of main memory would be dumped on disk or on magnetic tape for later diagnosis. A minimum up-time is also observed such that, after a restart, if the computer does not stay up for so many minutes, re-trials would not be attempted. Obviously, if the failure is in the auxiliary memory automatic restart would not be initiated. Failover must be resorted to.

After a failover from A to B, there are some design options as to what functions should be assumed by B. B could take on only the real-time critical functions that A had been doing and abandon all of the other functions. Or B could have all functions that were originally assigned to it plus all of A's, as long as we accept a reduced response and availability. Now this second option is not so bad if you have enough main memory in each computer. This ensures good response and availability most of the time but gives you a slightly degraded performance during the times that the system is down to one computer.

The system design should make it possible to operate one of the computers in a stand-alone mode for maintenance, testing, or large program development. It should also be possible to operate a computer in a pseudo-real-time mode with one console and a data-acquisition channel or two attached to it so that a program change or a new program may be tested in a real-time environment.

Each half of the dual configuration need not consist only of one computer. As discussed previously, front-end computers could be used for data-acquisition thus enhancing the response times of the main computer. Figure 2a shows a dual configuration with front-end computers FE1 and FE2. There are two possible schemes for failover. The simpler one, shown in Fig. 2b is to consider FE1 as an extension or slave of A and FE2 as a slave of B. The other failover scheme, shown in Fig. 2c allows the front-end computers to be switched to either one of the main computers.

Typically, the front-end computers are 16-bit minicomputers of a size large enough so that either one can handle all of the data-acquisition channels. The other would be a purely redundant backup. If the number of channels (or remotes) increase beyond the expandability of the minicomputer it would be time for a re-design with larger front-end capability. One could at the outset divide the channels between FE1 and FE2 so that both are sharing the work of data-acquisition. Such an arrangement is shown in Figure 3a where both FE's now have links with A and B. The failover scheme is shown in Figure 3b. Since one minicomputer is not quite big enough to handle the entire data-acquisition load then on a failure of FE1 or FE2, the remaining mini would have to do everything in a degraded mode.

There are a few exceptions in Table I to the dual configuration. These are multi-processor installations and each one is different. The desired length of this report unfortunately does not permit getting into a discussion of each one of them.

<u>The Real-Time Operating System</u>

Superior hardware features in a computer system do not necessarily guarantee good performance unless used effectively and to the overall system advantage by the real-time operating system. One of the more difficult problems of control center design and operation is obtaining a computer with a well-designed, field-proven, real-time operating system.

Historically, computer software has lagged behind hardware development by at least a year. This is still true with some of the new computers that are coming out on the market today. Also some computers, not necessarily new, are not intended by the manufacturers for real-time application and therefore are not supported by any real-time operating systems.

The industry picture so far shows: special operating systems developed by vendors for an initial project and offered on subsequent projects; standard real-time operating systems with slight modifications and with some enhancements to provide additional real-time features; real-time operating systems operating as a job under a general-purpose operating system; and time-sharing operating systems modified to handle real-time processing. The situation has improved recently. There are now some standard real-time operating systems with practically all of the features desired for a system control center.

For a system control center, what is a good real-time operating system? I would say one that carries out its functions without unduly impairing system response. The less overhead the operating system takes, the better for the response.

Overhead is incurred and hence response is affected wherever there is a shared resource. Main memory, the I/O channel, subroutines, and files are shared resources. Operating systems differ in the way they manage these resources. These are key features of an operating system which should be investigated and understood and should also be considered in software design. I will just report on two of these features, namely memory management and the I/O channel.

Main memory is a critical resource in a real-time computer system which must be shared by the programs. Typically, main memory is divided into several areas or partitions. Certain areas are occupied by the resident portion of the real-time operating system, system tables, interrupt processors, and buffers; other areas are dedicated to the database, to the data-acquisition software and to other resident foreground programs. Another area may be allocated for background processing. Whatever area is left is called the non-resident foreground area and is used for running foreground programs which reside in bulk memory and must be loaded into main memory for execution. The aggregate size of all the non-resident programs are several times the size of the foreground area in memory. Since not all programs have to run at the same time, the concept of memory management is the allocation of the foreground area among the several programs that have to run. Task scheduling is part of the memory management function.

There are presently two basic approaches which are in use at most system control centers. One involves foreground check-pointing or swapping. This means that a program in memory must be rolled out to disk if a higher priority program overlaps the memory locations of the first program. After the high priority program has completed execution the first program will be rolled back in provided no other higher priority program is contending for some of the same memory space. The response time of a control system with such a memory management scheme is inherently at a disadvantage. Initially this may not be apparent but as more critical functions are added overhead would start piling up in the swapping process. There are two variations to the swapping scheme that are being used. One is to divide the foreground area into several fixed-length segments of equal length. If swapping is being done within one or more contiguous segments the CPU can be executing a program in some other segment or segments until the swapping is completed. The other variation is the fixed-length segment idea plus time-slicing where each program running in memory is given an equal time-slice on a round-robin basis.

The other approach to memory management is to divide the non-resident foreground area into fixed partitions, not necessarily of equal lengths. A hardware interrupt priority level is assigned to each partition and each partition serves as an overlay area for a group of programs. As many programs may be in memory as there are partitions. A program in a partition runs to completion without swapping out for a higher priority program.

There are some systems where the memory management scheme is the so-called "dynamic memory allocation." This method is best used with computers having certain advanced hardware features. Depending upon these hardware features the dynamic memory allocation differs from one computer to the next. The basic concept, however, is that the operating system searches for an area in memory large enough to run a program. This area is a multiple of a hardware-specified length and, in some computers, need not be contiguous. A program does not always run in the same area of memory whenever it runs. Dynamic memory allocation also does swapping on the basis of program priority.

Dynamic memory allocation requires a complex operating system. Several of the new control centers coming into service in a year or two will have this type of operating system.

It is evident that the needs of memory management of whatever type plus the I/O requests that programs make heavily burden another critical shared resource--the I/O channel. This has to be watched in system design and during operation. First of all the operating system should not do more I/O's for memory management than is necessary. Secondly, the software design and priority structure should be thought out carefully with the objective of minimizing accesses to bulk memory.

THE MAN-MACHINE INTERFACE SUBSYSTEM

In system control centers the man-machine interface or the display subsystem consists of CRT's, dynamic wall displays, trend recorders, loggers, and alarm devices.

The CRT Display

CRT's are in universal use for the system operators, maintenance personnel, and programmers. Discussions of CRT display design considerations may be found in Reference 64. CRT applications at control centers are described in the various papers (see Bibliography) written about specific control centers.

The earliest reported uses of the CRT for power system operation are in References 15 and 17. Every control center in the US since the Houston Lighting & Power project have color CRT's with limited graphics in operator's consoles. There are full graphic stroke-type CRT's in black-and-white at CEGB and RWE. These have remarkable clarity but have relatively shorter tube life and a problem of "burning in" of pictures left on the screen for long periods. Although distribution control is outside the scope of this report I should mention the particularly effective and sharp one-line diagrams that can be obtained with a full-graphic, color CRT such as those in service at EDF's distribution control center for the city of Paris. I know of one system control center under development that is planning to use similar full graphic color CRT's. The effect of full graphics is also obtainable with lower cost raster-type, dot-addressable color CRT's. I anticipate that in the near future the design of new system control centers will take into consideration the option of full-graphic color CTT's.

There is no clear indication as to which cursor control device is preferred at control centers. Apparently the light pen, joy stick, track ball, and keyboard cursor controls can be used with almost equal facility by the human operator. The light pen is sometimes ruled out by the console design, e.g., the CRT's are mounted too far from the operator or at an awkward angle. In one control center the light pen is the only means of interaction--not even a standard keyboard is provided. In others, in addition to the light pen, there is a track ball or a joy stick. In some centers special function keys are used, plus the standard keyboard. In one center, special function keys are used to profusion. They cover a large part of the desk top area.

Of special interest are two unique CRT applications at CEGB. One is the rolling map technique where, by using the track ball, the operator can view any part of the system diagram and from there move the picture in any direction to view some other part of the system. Another feature is the retrospective display which allows the operator to go back in time to any specific minute within the last 30 minutes and view all the system data and network conditions pertaining to that minute.

Display Response Times

In the system control centers that I visited the CRT response times were very good. Normally, with a quiescent system, response times of 1 second or less are easily obtainable. This is the time from the instant the light pen is activated to request a one-line diagram until the time that the diagram is fully on view with all of the real-time information that goes with it.

An average response time of 2 seconds is a reasonable specification, with a maximum of 5 seconds during worst case conditions. If the average response time is longer than 2 seconds or the response frequently exceeds 5 seconds something is wrong and steps should be taken to determine and correct the cause of the slow response.

The Dynamic Wall Display

In Europe and in Japan, analog systems had been used not only for AGC but also, via special hardware, for driving dynamic wall displays. The coming of the digital computer has not displaced these analog systems. The analog AGC is kept. The hardwired dynamic wall display has been and continues to be considered of major importance, and is kept. This philosophy of having an independent logic and, in some centers, an independent telemetry, for the dynamic wall display will probably prevail for at least the next 5 to 10 years. New European systems under development and some in the offing still plan on implementing this concept.

In the US there are differences of opinion on the need for dynamic wall display at a control center.

As can be seen from Table I, many US centers have dynamic wall displays while quite a few just have a static system representation on the wall or none at all. Philadelphia Electric has something unique in the way of a matrix of CRT's showing the complete system diagram.

The dynamic wall display is intended to give an overview of the power system. The overview concept is best accomplished by a simplified representation preserving as much as possible the geographical orientation of the system. Details spoil the overview perspective and are best left to the CRT's. Simplification is specially important in representing very large systems. The larger the power system, the less resolution should the wall display have.

THE SOFTWARE SUBSYSTEM

The software in a control system may be divided into three categories:

1. System software consists of: the real-time operating system; processors for assembly, compiling, loading and overlay structures, file management, system generation, utility routines for debugging and testing. The system software is usually supplied by the computer manufacturer.

2. Application software includes all the programs which perform tasks for the operation of the power system, such as the data-acquisition software, display software, software to implement various control functions, and operation planning programs.

3. Support software are programs used by support personnel for computer system monitoring, real-time diagnostics and debugging, maintenance, and testing.

Application Software Development

The development of application software is perhaps the greatest source of technical problems, people problems, delays, and physical hardships in the implementation of a control center. For a system control center there can be as many as 100 to 200 application programs required, all to be integrated into one real-time system.

The software development cycle consists of three parts: analysis and design; coding and debugging; checkout and test.

Analysis and design is the most important phase of software development and should be carried out as thoroughly as possible. Problems in coding, debugging, checkout and testing, and in maintenance, are largely attributable to a poor analysis and design. This initial phase of software development specifies the individual program modules, the inputs, outputs, tables used, tables updated, and the algorithms. Design decisions are made on the database structure and access method, table and file structures, data update requirements, and backup requirements. All hardware-to-software and software-to-software interfaces are spelled out. Initialization procedures, CRT and logger messages, maintenance requirements, test procedures, and acceptance criteria are all specified. All this must be done, reviewed, and agreed upon before one line of code is put on paper. All too often the natural tendency to get going and start coding gets the better of the software designers and the analysis and design effort gets hurried through. This is an invitation to disaster.

The coding and debugging phase typically should take less than 25 percent of the entire software development work. Most of the application software in system control centers is written in assembly language as this gives programmers direct control over the program features that affect real-time linkages, response, and reliability.

The real-time linkages of a program consist of: accesses to the database; I/O requests to use and/or update files; I/O requests for CRT display and logger messages; program execution requests. Good response means an efficient code, a minimum of I/O, effective use of subroutines, and proper use of interrupt control.

Program reliability means use of fail-safe logic, proper initialization on system startup, avoidance of timing problems, invulnerability to bad parameters, control of possible arithmetic overflow, and proper handling of error returns on I/O and program execution requests.

The checkout and test phase takes up the rest of the software development work. The individual program is tested in the foreground in as complete a real-time environment as possible. The real-time environment is built up gradually as each program is integrated into the system. When the entire system is put together, hardware and software, the individual program tests are repeated as part of the system checkout. The acceptance tests complete the test cycle. The test drivers written for the individual program tests are kept for later use in system maintenance.

Support Software

There are two groups of support software. The first group consists of foreground diagnostic programs which are run on-line to monitor and control the performance of the control system. These programs provide the following functions:

1. Summary and control of remote station status: - On request, the status of the data-acquisition subsystem is displayed on the CRT. The display, which is dynamically updated, shows the scan conditions of each remote. Using this display it should be possible to place any remote off or back on the scan.

2. Display of all data received from a remote station: - On the CRT, all the data being received from a selected station may be viewed. This is a dynamic display and the data is seen as it changes from one scan to the next. Color is used to indicate when a piece of data is not updated or when it is out-of-limits. This function is very useful for checking data and for trouble-shooting programs which use the database.

3. Summary of data link errors: - A summary of all types of data link errors is kept on file. The summary shows the remote station, the type of error, the number of times the error has occurred, and the times of the first occurrence and of the last occurrence. This summary is viewable on the CRT. It is periodically printed out on the logger for review by maintenance personnel.

4. Dynamic display of activity in computer overlay area: - This is a CRT display which shows dynamically what programs are in memory executing or waiting for I/O, what programs called them, and what programs are waiting in the queue.

5. Display of main or auxiliary memory and the ability to patch any memory locations: - With this function an area of main or auxiliary memory starting with a specified address may be displayed on the CRT. In this manner, tables or segments of program code may be examined. The patch capability is useful for on-line debugging, program testing, and an immediate correction of an erroneous condition. The patch on the on-line system is intended to be an interim measure until a permanent system revision can be made.

6. On-line measurement of computer system performance: - This on-line function, sometimes known as "software accounting", gathers statistics at specified time intervals about CPU utilization, CPU idle time, I/O waits, number of I/O transfers, what programs have run and how often, etc. This function may run automatically or at operator's request. The statistics are useful for evaluating system loading and performance. The impact of adding new programs may also be measured by taking before-and-after statistics.

7. Dump of real-time data on tape: - Via the CRT the operator can initiate the dumping of selected real-time data at specified time intervals on magnetic tape. The operator specifies on the CRT the data to be dumped and the time interval. The dump continues until the operator stops it by a CRT entry. The magnetic tape is later printed out on the line printer by an editing program on the secondary computer.

Several control centers have some or all of the support functions mentioned above. Actually these programs are of tremendous value during the implementation of the control system and not just after the control center is placed in service. The system design should therefore include such on-line support programs. These should be completed by the system supplier early in the implementation period so that the development work could be speeded up.

The second group of support software consists of programs for system maintenance. These are off-line programs which are used for updating files to match changes or additions in the power system. A necessary file maintenance subsystem is one that updates all files related to the man-machine interface. This includes a "picture compiler" which would allow a maintenance programmer to compose a CRT picture at a console and then run the compiler to generate the CRT code of the picture and store it in the correct file location. In an interactive mode the maintenance program will step through whatever operator inputs are required to update all other tables related to the new picture. A good maintenance program design should minimize the amount of operator inputs. Other file maintenance subsystems may be designed for other families of programs, such as the SE, SA, and OLF group or the AGC, EDC group. Although it is not necessary to have one overall file maintenance system to update all affected files in one operation, some control centers have such a support program.

An interesting program is in service at General Public Utilities which allows updates of files on-line in the primary computer system.[70] This interactive program also updates application programs which are affected by the power system change or addition.

STEPS IN THE IMPLEMENTATION OF A CONTROL CENTER

The implementation of a system control center from concepts to initial operation is a process that takes about 4 to 5 years. In what follows I shall briefly enumerate the sequential steps involved in the process. The starting point of the sequence assumes that some preliminary conceptual studies have been made by the utility company. If the proposed center will have advanced security control functions this preliminary study of methods and algorithms may take 1 to 2 years making the total time to implementation about 5 to 7 years.

Phase I - System Analysis

Step 1 - Operating Requirements: - This is a document that defines the objectives and scope of the control center. Constraints are identified, such as communication channels to be used, integration of existing control equipment, or preference for certain types of equipment. The desired operating functions and concepts are described in general terms. This document may be used in obtaining the services of a consultant, if one is required, for the performance of Steps 2, 3, and 4.

Step 2 - System Requirements: - The purpose of this step is to determine the specific monitoring and control functions which should be done by the control center. During this step the operating personnel should be interviewed so that operating needs and problems may be adequately analyzed. Proposed computer functions should be studied for practicality of implementation and benefits to be obtained. Requirements for functions such as SM, SE, SA, OLF, or OPF should be analyzed as to their impact on the system requirements. The present and future data requirements including measurement redundancies are identified.

The end-result of this step is a System Requirements document. The document constitutes an overall statement of requirements for all the system control center subsystems--data-acquisition and control, computer, display, application software, building, and power supply. Usually the communication subsystem is specified as a constraint in Step 1.

Phase II - System Design, Pre-Contract

Step 3 - Functional Specifications: - In this step the system requirements of Step 2 are translated into a set of functional specifications. In order to allow room for creative design which takes advantage of new techniques and technology, the specifications should not be too rigid or specific about how the functions should be implemented. A model computer configuration may be used as a frame of reference for specifying functions and performance but the actual configuration should be left to the prospective designer.

The functional specifications should include performance requirements such as response times and system availability.

At this stage, decisions should be made, if not already made, about the following:

- -- univendor versus multivendor type of procurement
- -- which application programs will be written in-house
- -- responsibility and organization for project management and for system integration
- -- responsibility for system maintenance

The work in Step 3 also includes budgetary cost estimates for project authorization purposes.

By the end of this step or soon after project authorization is obtained, an in-house project team must be organized. This is a full-time staff made up predominantly of software-type specialists. The in-house team should have a combined background in power system engineering, digital hardware, computer application, and system operation. The team would be charged with the mission of becoming thoroughly familiar with the control system and of making sure that it carries out the functions according to specifications. The team would have responsibilities for reviewing and approving all design and test documentation, training itself, working with vendors and other company specialists, witnessing and approving checkouts and acceptance tests, and for training others in the company. Part of the team may later assume system maintenance and enhancement until a permanent group will have been established. Deciding to have an outside group do maintenance does not obviate the necessity of an in-house project team.

Phase III - Procurement

Step 4 - Procurement Specifications: - The procurement specifications will consist of those parts of the functional specifications which will not be done in-house plus instructions on the proposal organization, bidding format and other terms and conditions. The procurement specifications should not place unnecessary burden on the bidders by asking for too much data which have little relevance to bid evaluation. Relevant data should include exhibits of programming standards, documentation standards, and project management tools.

Step 5 - Bid Evaluation: - We know what is to be done at this step, there are no pat rules on how to do it. Expert assistance should be sought, if not available in-house, in evaluating the technical aspects of the bids. Especially important is the assessment of hardware and software subtleties which affect system response and availability.

Step 6 - Statement of Work: - This document is prepared by the successful bidder and consists basically of the original proposal with agreed-upon changes and additions; definition of implementation responsibilities; procedures for project interfaces, control and coordination; enumeration of deliverable documents and description and description of documentation requirements; project schedule.

Phase IV - System Design, Post-Contract

Step 7 - System Interface Specifications: - This document defines in precise detail all hardware-to-hardware, hardware-to-software, and software-to-software interfaces. The software interface specifications includes: all real-time linkages; the database structure and access method; table and file structures to the bit level; data update requirements. storage requirements; and initialization requirements. Even if the end-user, i.e., the utility company, will not supply any of the application software it is necessary that the interface document be produced formally. Several iterations with the initial efforts in Step 8 are to be expected. The interface specifications is an important design document since many individual hardware and software designers are involved in the project. Some are with the prime contractor, some are with a subcontractor for advanced application programs, and others are with the utility company. Serious problems will tend to come about if the interfaces are not fully documented at the very beginning. The system interface specifications must be reviewed and have the approval of the in-house project team.

Step 8 - Detailed Design Specifications: - This is the main design effort. Drafts of the hardware and software design specifications must be thoroughly reviewed by the in-house project team. Sufficient time must be allowed for this review as draft revisions are likely to be made. In addition to checking the correctness of the design with regard to functions required, it should also be checked for built-in adaptability to future needs. The testing procedures and maintenance features of the design must also be carefully checked.

Phase V - System Build

Step 9 - Hardware Fabrication and Software Coding and Debugging: - This stage of the system implementation requires collaboration between the contractor and the in-house project team, especially in the software area. Close interaction between the vendor and team must be encouraged at the programming leval and not just at the supervisory level. The idea is to create a spirit of joint venture with both groups working for the same end rather than being in opposing camps. This is best achieved if the in-house project team is also doing its share of programming. That is why I recommend that the in-house team reserve for itself some if not a large portion of the application programs to do. This not only gives valuable training in real-time systems but gives the team members fluency in discussing software problems and ideas with other programmers. Each team member must, if possible, write at least one real-time program in assembly language and must also be assigned to monitor and help in work being done by the vendor in one specific area. Only by knowing through direct experience how a real-time program is designed and coded can a team member really begin to understand and be of assistance in the area that he is assigned to monitor. Discussions of programming problems at regular meetings is of mutual benefit to all participants.

Some companies have worked out arrangements with vendors to have resident teams at the vendors' premises. This works best if the resident team has authority to make routine design decisions without the nuisance of being overruled by someone in the home office.

It is of great value to project implementation for a "design freeze" to be instituted at some time during system build. This milestone must be explicitly noted in the project schedule. After the design freeze no more design changes must be requested by the utility company. The contract, however, must allow for table or CRT format changes to be made as often as desired if the in-house project team will make these changes. This is highly desirable since all the power system data will come from the utility anyway. It is also good training for system maintenance. To do this requires that the support software as well as basic CRT interactive programs must be completed early in the project. I recommend that such an early completion of all support software, on-line and off-line, as described in the Software Subsystem section be stipulated in the State of Work (Step 6). The early availability of the support software will benefit all programmers in their debugging, testing, original construction and subsequent corrections of files. Since support software requires the use of the CRT, such a working facility must be available early in this step.

Any design changes thought of after the design freeze must be noted by the head of the project team for later implementation after system acceptance. Such an implementation of a design change by the project team would be a true test of its ability to take care of the system without outside assistance. The design freeze does not preclude the necessity for making corrections of design fallacies uncovered later. Some of these design errors could be major in nature.

Step 10 - Acceptance Test Procedures: - Towards the end of Step 9 the vendor, with the assistance of the in-house project team, should prepare the acceptance test procedures. This document, just like all other documentation, must be reviewed and approved by the project team. A significant portion of the test procedures must be contributed by the project team as it is to the utility company's interest to make the test as realistic and as meaningful as possible. Tests should be included which would test the fail-safe capability of the system design. Failure and error conditions must be simulated to really test the system. "Try to make the system fail" must be one of the test objectives.

Phase VI - System Checkout and Integration

Step 11 - System Checkout: - Each program after debugging and testing must be checked out within the framework of the completely integrated system. Since this cannot be done all at once, a systematic checkout procedure must be set up. First it must be assumed that the System Software is available and in the system. The resident data base and the data-acquisition software must be checked out first. A number of remote terminals with simulated inputs must be part of the test set-up. When an application program is to be checked out all the real-time linkages must be operative. This means actual access to the database and actual execution requests for other programs. If these programs are not yet, dummy programs must be put in. I/O's must be made to actual files. If these are not in because the updating program is not yet in the system, test data must be loaded in. The program checkout must be initiated by a test driver. The CRT is the most convenient and eventually the logical place to trigger the test driver. This process is repeated with each program until the entire system is fully integrated.

Step 12 - System Integration: - After the system has been put together in Step 11 the system integration is the period when all errors found are corrected. The individual program checkouts done in Step 11 must now be repeated on the completely integrated system. This is a process of correcting errors, re-running tests, finding more errors, making corrections; until no more errors are found. The final step of system ingeration is to make a dry run of the acceptance tests. After a successful dry run the system is ready for acceptance testing. The decision of readiness for acceptance testing must have the approval of the in-house project team.

Phase VII - Acceptance Tests

Step 13 - Ready for Shipment Tests: - In this step the acceptance test procedures, except for AGC dynamic tests, are followed for the purpose of determining whether the system is ready for shipment from the vendor's premises to the system control center building. Approval of these tests by the project team does not constitute system acceptance. The head of the project team may approve shipment even if the tests are not 100% successful provided that in his judgment the errors are minor and can be corrected at the control center site. Major problems are best corrected before shipment and the ready-for-shipment tests should be repeated in their entirety after the corrections are made.

It has not been shown as steps in the implementation sequence but it should be understood that during all this time the system control center was being built, communication channels installed, and the remote terminals required for initial service installed at generating plants and transmission substations. For shipment of the control system, the building must be ready with air conditioning and uninterruptible power supply installed and working.

Step 14 - On-Site Checkout and Timing: - The system is installed at the control center under the vendor's supervision. The vendor makes all the necessary checkouts, corrects all pending errors, and makes tuning adjustments of control parameters of the AGC. It is of course assumed that arrangements had been worked out before hand for the ability to transfer control temporarily to the new center for tuning and transfer it back without intentional delay to the existing AGC. Tuning procedures must be witnessed and assisted by the in-house project team. AGC tuning would be required for several generating and load conditions. After the AGC and all other functions have been checked out and enough remote terminals hooked up with the communication channels, the system would be ready for final acceptance tests.

<u>Step 15 - Final Acceptance Tests</u>: - The final acceptance tests constitute the formal demonstration that the system is indeed operational. The test environment must consist of actual system conditions plus simulations of abnormal conditions to be done by test personnel at remote stations. Actual interconnection schedules should be negotiated as part of the AGC testing. The test records must include accurate descriptions of all problem encountered and of the solutions made. These records must be carefully checked and approved by the in-house project team as all corrections must appear in the updated documentation to be submitted later by the vendor.

Phase VIII - Operator Training

<u>Step 16 - System Indoctrination</u>: - This is out of sequence, but prior to the shipment of the system, some training should have been started of power system operators. This initial training consists of an overall view of the control system followed by a description of each subsystem. Special emphasis must be placed on the use of the man-machine interface using proper visual aids. This indoctrination should not be given until after the design freeze. Otherwise design changes made before the freeze and after the operators' training sessions would merely create confusion.

<u>Step 17 - Hands-on Training</u>: - At sometime during Step 14 the system operators should be scheduled to come to the new control center and use one of the consoles as long as this does not interfere with the checkout work. This is possible since with a dual computer configuration the design of the system should allow a console to be used with the secondary computer. The schedule for training must be arranged even if only for a couple of hours a day. A member of the in-house team must sit with the operator initially to help in the training process.

After the completion of the acceptance tests (Step 15), more training can be scheduled as deemed necessary, prior to placing the system in service. At this stage the system is available for as much training as desired. In fact, during this training the new system could be unofficially controlling the power system for certain periods of time.

Phase IX - System Operation

<u>Step 13 - Initial In-Service</u>: - At this step the system shall be officially in service. This shall also be the start of the availability demonstration period if one had been specified. If there is an availability demonstration, no design changes must be made. The system should be the same as it was at the completion of the final acceptance tests. If minor errors are found, these should be corrected by the in-house team with the agreement of the vendor. If errors require significant software or hardware modifications to correct them the vendor should be called in to make the corrections. If a maintenance contract had been made with either the vendor or a third party for hardware maintenance, correction of hardware errors would be made by this contract maintenance personnel.

<u>Step 19 - Delivery of Documents</u>: - The vendor delivers a complete set of updated "as-built" documents which include design specifications, reference manuals, and maintenance manuals. This usually completes the vendor's obligations except for the availability guarantee if any and the standard one or two-year service warranty.

SYSTEM MAINTENANCE AND ENHANCEMENT

During the life of a system control center there will be continuing work in two areas: system maintenance and system enhancement.

System maintenance for the hardware subsystems in the control center is well understood. The scope of maintenance responsibility can be defined as to which electronic and electrical equipment are to be included. The tasks to be performed are preventive and corrective maintenance plus daily routine checks. The hardware maintenance group must keep a documentation of all equipment failures and problems, the repairs made, plus other data for failure analysis.

Software maintenance has some overlap with system enhancement and needs some arbitrary definition. Software maintenance consists of the following activities:

-- System debugging
-- File maintenance
-- Testing of new data-acquisition remote stations and of new data points
-- Software system rebuilds
-- Changes in existing programs
-- Installation of new programs
-- Resource (main and auxiliary memory) usage monitoring and management

System enhancement consists of:

-- Planning of future software development
-- Design and development of new programs
-- Planning and initiation of computer configuration changes and expansion

The reason for this division into two areas is that the business of software maintenance requires a dedicated group which will not have the time for planning and developing system improvements. Another group must have the system enhancement responsibility.

Whether both groups should report to the same supervisor or to separate supervisors is not too important. There are instances of both types of organization which are working out very well. One should be careful about too much structuring of the control center personnel organization. The control center is not a data processing center nor is it a batch software factory. For example, creating a separate group with its own head and serving as a software house for system maintenance and system enhancement may just generate more overhead and implementation problems. Just as in the original system implementation process, teamwork is the touchstone of getting things done right in a real-time control system. A looser type of organization is more conducive to getting people to work as a team.

TRENDS IN CONTROL SYSTEM DESIGN

Examining Table I for functions being planned at existing and forthcoming control centers, the trends appear to be the wider implementation of the SA, SE, and OLF functions. There are two control centers planning on installing the OPF function. It is not certain when this will be in service at these centers but probably we will have to wait at least a year. At least two of the SE's being planned will have some algorithm innovations different from the methods in operation today. We will see more of the fast decoupled load flow method being used for contingency evaluation. At least three centers are planning on using this approach. There are no trends in computer configuration which depart from the basic dual computer approach. It is too early to tell whether there will be a trend toward the distributed computer concept which one hears so much about in other computer applications.

There is a trend exhibited in the use of new and more powerful minicomputers and also midicomputers. Manufacturers of CRT display generators and device controllers are beginning to use microprocessors. New data-acquisition equipment are also being manufactured or designed with programmable capabilities.

The overall picture of system control center implementation is one of slow movement. One major factor has been the economic slowdown. In 1974 several utilities deferred their plans for system control centers. In the US only one major contract was awarded. In 1975 there are indications of resumed activity and several new projects are in the bid evaluation or specification development work. Several utilities have started the system requirements phase of the implementation cycle. System control center projects are spreading throughout the world. In late 1975 and by 1976 we should see several new projects commissioned.

We should see increased activity in the direction of distribution control centers interfacing with a system control center. There will also be more hierarchical systems as pool centers get implemented. This trend would open up the possibility of more information exchange to help each individual member company of a pool do a better job of its own security control.

PROBLEM AREAS AND RESEARCH NEEDS

In this section I will mention some problem areas in system control center design and not in the functions intended for control centers. There are of course many problems which are known, some of which I have discussed previously, and for which solutions already exist. I will attempt in this section to identify only those which are in need of improved methods of attack.

In my opinion, some of the problems and needs in control center design are the following:

1. A set of standard benchmark programs representative of power system control operation for evaluating real-time operating systems and computer hardware capabilities.
2. A study of the most suitable types of data structures which would facilitate processing and maintenance but not impose excessive overhead.
3. Realistic and reasonable definition of system availability and its measurement.
4. Improved techniques of translating power system physical descriptions into software tables.
5. Hardware rolling map capability for color CRT's.
6. Large dimension, yet compact, dynamic color displays.
7. Methods and criteria for on-line tuning and performance measurement of the automatic generation control function.
8. Methods and criteria for on-line tuning and performance measurement of the computer system.
9. Investigation of which routines may be implemented in microcode to benefit overall system performance.
10. Investigation of standard format for data transmission in data-acquisition and control systems.

At this point I have no industry consensus as to what problem areas should be pursued as a research activity. It would be best to wait for more suggestions of needs to add to the list I have above.

Almost every utility company starting on a new system control center project has had little or no previous experience in real-time computer control. What then can one learn from the experiences of others? I am providing answers to this question in the way of "Summary of Guidelines and Pitfalls to Avoid" in Appendix B.

BIBLIOGRAPHY

This bibliography consists of published papers on control center concepts, design, and implementation. Papers describing algorithms and methods are included if they relate directly to functions which have been implemented at control centers. Papers referenced in the text are also included in this bibliography.

1965

(1) N. Cohn, "The Automatic Control of Electric Power in the United States", IEEE Spectrum, November 1965

1966

(2) C. W. Ross, "Error Adaptive Control Computer for Interconnected Power Systems", IEEE Transactions, Vol. PAS-85, July 1966

(3) G. Quazza, "Noninteracting Controls of Interconnected Power Systems," IEEE Transactions, Vol. PAS-85, July 1966

1967

(4) T. E. Dy Liacco, "The Adaptive Reliability Control System", IEEE Transactions, PAS-86, May 1967

(5) N. Cohn, "Considerations in the Regulation of Interconnected Areas", IEEE Transactions, /Vol. 86, December 1967

1968

(6) T. E. Dy Liacco, "Control of Power Systems via the Multi-Level Concept", Systems Research Center, Case Western Reserve University, SRC-68-19, Cleveland, 1968

(7) G. S. Hope, B. J. Cory, "Development of Digital Computer Programs for the Automatic Switching of Power Networks", IEEE Transactions, Vol. PAS-87, July 1968

(8) I. Hano, A. Chiba, T. Umezu, K. Ode, "Optimal Control of Voltage and Reactive Power in Power Systems", CIGRE, June 10-20, 1968

(9) J. Peschon, D. S. Piercy, W. F. Tinney, O. J. Tueit, "Sensitivity in Power Systems", Vol. PAS-87, August 1968

(10) W. K. Pence, "New Michigan Pool Center Stresses System Security", Electrical World, October 21, 1968

(11) IBM Research Division, "Power System Computer Feasibility Study", Final Report for the Department of Public Service of the State of New York December, 1968

(12) K. Fumai, K. Ode, "Power System Voltage Control by Using a Process Control Computer", IEEE Transactions, Vol. PAS-87, December, 1968

(13) M. Couvreur, "Logic Adaptive Process for Control and Security in Interconnected Power Systems", IEEE Transactions, Vol. PAS-87, December 1968

1969

(14) T. E. Dy Liacco, T. J. Kraynak, "The Processing by Logic Programming of Circuit Breaker and Protective Relaying Information", IEEE Transactions, Vol. PAS-88/No. 2, February 1969

(15) F. A. Kramer, "Load Dispatcher's Computer System at Public Service Electric and Gas Company", Proceedings, Power Industry Computer Applications Conference, Denver, May 18-21, 1969

(16) W. S. Ku, P. Van Olinda, "Security and Voltage Applications of the Public Service Dispatche Center", Power Industry Computer Applications Conference, Denver, May 18-21, 1969

(17) J. P. Elliott, J. W. Allen, "An Application of Interactive Graphics to On-Line Debugging for a Digital Dispatch Control System", Power Industry Computer Applications Conference, Denver, May 18-21, 1969

(18) H. D. Limmer, "Techniques and Applications of Security Calculations Applied to Dispatching Computers", Proceedings, Power Systems Computation Conference, Rome, June 23-27, 1969

(19) T. C. Cihlar, J. H. Wear, D. N. Ewart, L. K. Kirchmayer, "Electric Utility System Security", Proceedings, American Power Conference, 1969

(20) I. Hano, Y. Tamura, S. Narita, K. Matsumoto, "Real Time Control of System Voltage and Reactive Power", IEEE Transactions, Vol. PAS-88, October 1969

1970

(21) Y. Komiya, K. Ode, "Determination of Reference Operating Voltages in Power Systems", IEEE Transactions, Vol. PAS-89, January 1970

(22) (a) F. C. Schweppe, J. Wildes, "Power System Static-State Estimation, Part I: Exact Model" IEEE Transactions, Vol. PAS-89, January 1970

(b) F. C. Schweppe, D. B. Rom, "Power System Static-State Estimation, Part II: Approximate Model", ibid

(c) F. C. Schweppe, "Power System Static-State Estimation, Part III: Implementation", ibid

(23) G. W. Stagg, J. F. Dopazo, O. A. Klitin, L. S. Van Slyck, "Techniques for the Real-Time Monitoring of Power System Operations", IEEE Transactions, Vol. PAS-89, April 1970

(24) T. E. Dy Liacco, "The Emerging Concept of Security Control", Proceedings 1970 Symposium on Power Systems, Purdue University, Lafayette, May 1970

(25) M. Cuenod, U. G. Knight, H. Persoz, G. Quazza, "Present Trends in Automatic Power System Security Assessment and Control", CIGRE, Aug. 24-Sept. 2, 1970

(26) H. Ungrad, H. Glavitsch, "Centrally Coordinated Back-up Protection and System Security Monitoring as Constituents of an Integrated System for the Automation of Power Transmission", CIGRE, Aug. 24-Sept. 2, 1970

(27) IBM Research Division, "Bulk Power Security Assessment", Edison Electric Institute, Final Report RP90-3, November 1970

(28) G. Quazza, "Automatic Control in Electric Power Systems", Automatica, Vol. 6, New York: Pergamon, 1970

(29) H. Gordon Stewart, "PJM Center Analyzes, Controls System", Electrical World, Nov. 1, 1970

1971

(30) F. C. Hart, R. S. Johnson, D. W. Magee, E. K. Nielson, H. M. Ray, "Inter-Computer Data Exchange for New England Area Dispatching", Power Industry Computer Applications Conference, Boston, May 23-26, 1971

(31) M. R. Gent, J. W. Lamont, "Minimum-Emission Dispatch", IEEE Transactions, Vol. PAS-90, November/December 1971

(32) J. F. Aldrich, H. H. Happ, J. F. Lener, "Multi-Area Dispatch", IEEE Transactions, Vol. PAS-90, November/December 1971

1972

(33) T. E. Dy Liacco, B. F. Wirtz, D. A. Wheeler, "Automation of the CEI System for Security", IEEE Transactions, Vol. PAS-91, May/June 1972

(34) W. S. Kleinbach, "The Use of a Digital Computer for Monitoring, Controlling, and Scheduling for the Pennsylvania-New Jersey-Maryland Interconnection," CIGRE, Aug. 28-Sept. 6, 1972

(35) T. E. Dy Liacco, "Design Considerations for an Advanced System Operation Center", 4th Power Systems Computation Conference, Grenoble, September 1972

(36) A. St. Johnson, G. Dromey, U. G. Knight, "Data Processing and Computation for Real-Time Security Assessment", 4th Power Systems Computation Conference, Grenoble, September 1972

(37) J. Svoen, S. A. Fismen, H. H. Faanes, A. Johannesen, "The Computer in the Power System--The On-Line, Closed-Loop Approach for Control of Generation and Overall Protection at Tokke Power Plants", CIGRE 32-06, Paris, France, September 1972

(38) G. Quazza, E. Ferrari, "Role of Power Station Control in Overall System Operation", pp. 215-251, "Real-Time Control of Electric Power Systems" (book), Elsevier, 1972

1973

(39) A. M. Sasson, S. T. Ehrmann, P. Lynch, L. S. Van Slyck, "Automatic Power System Network Topology Determination", IEEE Transactions, Vol. PAS-92, March/April 1973

(40) J. C. Russell, W. Rades, "The Wisconsin Electric Power Company Energy Control System," Power Industry Computer Applications Conference, Minneapolis, June 4-6, 1973

(41) R. G. Rice, G. W. Stagg, "Application Program Design Criteria for the Southern Company Control System," Power Industry Computer Applications Conference, Minneapolis, June 4-6, 1973

(42) T. E. Dy Liacco, K. A. Ramarao, A. W. Weiner, "Network Status Analysis for Real-Time Systems", IEEE TP XII-A, Power Industry Computer Applications Conference, Minneapolis, June 4-6, 1973

(43) C. F. Ham, K. B. Rennie, "Control Center Ups System Performance", Electrical World, August 15, 1973

(44) D. W. E. Crook, J. H. Harris, "The Use of On-Line Computing Facilities at the National Control Centre of the CEGB", Proceedings of Symposium on Implementation of Real-Time Power System Control by Digital Computer, Imperial College of Science and Technology, London, September 1973

(45) T. E. Dy Liacco, "Experiences in the Design, Implementation and Operation of the Cleveland System Operation Center", Proceedings of Symposium on Implementation of Real-Time Power System Control by Digital Computer, Imperial College of Science and Technology, London, September 1973

(46) K. Kano, M. Nanba, T. Nishimori, "Automatic Load Dispatching System in Kansai Electric Power Company", Proceedings of Symposium on Implementation of Real-Time Power System Control by Digital Computer, Imperial College of Science and Technology, London, September 1973

(47) D. E. Johannson, "Operation Experience with the Dittmer On-Line Control Center", Proceedings of Symposium on Implementation of Real-Time Power System Control by Digital Computer, Imperial College of Science and Technology, London, September 1973

(48) J. F. Dopazo, S. T. Ehrmann, O. A. Klitin, A. M. Sasson, "Justification of the AEP Real Time Load Flow Project", IEEE Transactions, Vol. PAS-92, September/October 1973

1974

(49) T. C. Cilhar, J. H. Wear, W. L. Carroll, "Dispatch Center Fulfills Expectations", Electrical World, March 15, 1974

(50) B. Stott, O. Alsac, "Fast Decoupled Load Flow", IEEE Trans. Vol. PAS-92, May/June 1974

(51) W. A. Johnson, S. H. Bouchey, D. R. Penders, "The Potomac Electric Power Company's Consolidated Control Center", C74 366-1, IEEE Summer Power Meeting, Anaheim, California, July, 1974

(52) W. B. Hennington, T. W. Landers, S. H. Bouchey, "An Interactive Energy Control System Data Base Compiler", C74 365-3, IEEE Summer Power Meeting, Anaheim, California, July, 1974

(53) F. C. Schweppe, E. J. Handschin, "Static State Estimation in Electric Power Systems", Proceedings of the IEEE, July 1974

(54) T. E. Dy Liacco, "Real-Time Computer Control of Power Systems", Proceedings of the IEEE, July 1974

(55) A. M. Sasson, H. M. Merrill, "Some Applications of Optimization Techniques to Power System Problems", Proceedings of the IEEE, July 1974

(56) H. Asal, J. Caneel, K. Reichert, "State Estimation for Power System Monitoring in a Section of the Swiss EHV System: System Configration and Operating Experience", CIGRE, August 21-29, 1974

(57) K. Reichert, E. Handschin, J. Caneel, H. Asal, "On-Line Lastflussberechung (State Estimation) in elektrischen Energieversorgungssystemen, Grundlagen", Elektrotechnische Zeitschrift-A Vol. 95 (1974)

(58) J. A. Bubenko, L. O. petterson, J. H. Svoen, A. Syvertsen, "On-Line State Estimation", C74 321-6, IEEE Summer Power Meeting, Anaheim, July 14-19, 1974

(59) F. Ariatti, C. Brasioli, A. DiPerna, F. Galli, G. Quazza, "Orientamenti Per un Nuovo Sistema di Controllo della Produzione e Trasmissione dell'ENEL", 75a Riunione AEI, Rome, 1974

1975

(60) O. Saito, K. Koizumi, M. Udo, M. Sato, H. Mukae, "Security Monitoring Systems Including Fast Transient Stability Studies", T75 085-6, IEEE Winter Power Meeting, New York, January 26-31, 1975

(61) S. A. Arafeh, R. E. Kilmer, J. H. Rumbaugh, "Closed-Loop Computer Control of a System of Radial Load Busses, Using Transformers, Capacitors and Reactors", C75 032-8, IEEE Winter Power Meeting, New York, January 27-31, 1975

(62) R. K. Green, "Centralized System Dispatching Integrated with Localized Power Plant Control", American Power Conference, April 23-25, 1975

(63) N. M. Peterson, J. P. Britton, T. J. Varney, "Advanced Applications System for the Wisconsin Electric Power Company Control Center", Power Industry Computer Applications, New Orleans, June 2-4, 1975

(64) T. E. Dy Liacco, D. L. Rosa, "Designing an Effective Man-Machine Interface for Power System Control", IEEE TP X-B, Power Industry Computer Applications Conference, New Orleans, June 2-4, 1975

(65) E. Hnyilicza, F. C. Schweppe, S. T. Y. Lee, "Steady-State Security Regions: Set-Theoretic Approach", IEEE TP XIII-A, Power Industry Computer Applications, New Orleans, June 2-4, 1975

(66) J. F. Dopazo, S. T. Ehrmann, A. M. Sasson, L. S. Van Slyck, "The AEP State Estimation Monitoring and Security System", presented at Computer Operation of Power Systems (COPOS) Symposium, Sao Carlos, Brazil, Augubt 18-20, 1975

(67) Y. Tamura, T. Yokoyama, A. Chiba, T. Homma, K. Kano, K. Mitani, K. Mitzushima, "Energy Control Centers in Japan", IFAC World Congress, Boston, August 24-30, 1975

(68) J. W. Dillow, U. G. Knight, D. G. Taylor, "System Control in the Central Electricity Generating Board", IFAC World Congress, Boston, August 24-30, 1975

(69) T. C. Cihlar, "New Control System with an Advanced Man/Machine Interface for Commonwealth Edison Company's System Security", IFAC World Congress, Boston, August 24-30, 1975

(70) C. F. Doll, C. B. Woodward, M. H. Ewer, L. A. Schmidbaner, R. P. Whitesell, J. R. Evans, W. A. Johnson, "Multi-Company Electric Power System Control on the Pennsylvania-New Jersey-Maryland Interconnection", IFAC World Congress, Boston, August 24-30, 1975

(71) A. J. Harris, J. W. Shelly, "Ontario Hydro's Data-Acquisition and Control System", IFAC World Congress, Boston, August 24-30, 1975

(72) V. Saminaden, "Present and Future Computer Techniques for Operation and Control of the French Electric System", IFAC World Congress, Boston, August 24-30, 1975.

(73) L. Gustafsson, J. Lindqvist, "The New Information System for the National Dispatching Centre at the Swedish State Power Board", IFAC World Congress, Boston, August 24-30, 1975

(74) A. S. Debs, W. H. Litzenberger, "Implementation of the BPA On-Line State Estimator", to be presented at the Power Systems Computation Conference, Cambridge, September 1-5, 1975

(75) L. Pettersson, "State Estimation in the Swedish TIDAS-Evaluation of Methods", to be presented at the Power Systems Computation Conference, Cambridge, September 1-5, 1975

ADDENDUM

NEEDS IN RELATED AREAS

This is a supplement to the foregoing report on System Control Center Design. The intent is to indicate certain needs in other power system engineering areas as seen from the system control viewpoint. I am offering these brief observations in an addendum since the in-depth discussions of these problems properly belong to other state-of-the-art reports.

The following are some currently needed developments in functions destined for system control centers:

1. On-line steady-state network equivalent for representing systems external to the jurisdictional bounds of a system control center.

2. Efficient general-purpose optimum power flow for on-line application in security-constrained optimization. This could be used in economic dispatch, corrective action strategy, emergency control, and for operating studies.

3. Stochastic load flow for security monitoring, security analysis and for operating studies.

4. Dynamic security analysis methods which do not impose a heavy computer burden and do not require input data not readily obtainable on-line.

5. An adaptive contingency evaluation procedure, i.e., one that can select relevant contingencies and learns from previous experiences.

Although many technical papers and research studies in the past have addressed themselves to some of the problem areas listed above there is still a need for practical algorithms or procedures suitable for real-time implementation. Analytical methods which are satisfactory for off-line work are not necessarily suitable for on-line use.

APPENDIX A

TABLE I

SYSTEM CONTROL CENTERS FOR GENERATION-TRANSMISSION SYSTEMS

July, 1975

This table is a fairly complete listing of state-of-the-art system control centers, in service or under development, throughout the world as of July, 1975. Strictly SCADA-type centers are not included except if they belong to a lower level of a system control hierarchy.

Although I have tried my best to gather as much information as possible about what is going on in the industry, there must be a few systems which I am not aware about. I apologize for these omissions. Any errors or apparent misrepresentation of facts that may appear in this table are entirely unintentional. I would greatly appreciate being informed of any such omissions and errors.

T. E. Dy Liacco
Cleveland Electric Illuminating Co.
P. O. Box 5000
Cleveland, Ohio 44101

(216) 623-1350 ext. 3349

TABLE 1
SYSTEM CONTROL CENTERS FOR GENERATION-TRANSMISSION SYSTEMS

	In-Service Date	Name of Company	Number of Remotes	Computer System	Core Size in Words	Bulk Storage in Megabytes	Number of Color CRT's	Dynamic Wall Display	On-Line Functions[a]	
									In-Service	Planned
1	June 1969	Michigan Electric Power Ann Arbor, Michigan	36	1-GEPAC 4020 + 1-GEPAC 4060 + 2-GEPAC 4010 + Data Links to Detroit Edison & Consumers Power	32k 32k 32k ea.	534k wds 1262k wds 2000k wds ea.	5,2 (bw)	None	AGC, EDC, SM, SA	
2	July 1970	Penn.-Jersey-Maryland (PJM) Interconnection Norristown, Pennsylvania	11	Dual IBM 370/158 + 2 IBM System 7 + Data Links to 12 Pool Member Locations	1000k Bytes ea. 20k ea.	400	8	Gilbert Backlighted	AGC, EDC, SM, SA	
3	June 1970	New England Power Exch. West Springfield, Mass.		1-SIGMA 2 + 1-SIGMA 2 + Data Links to 4 satellite computers	32k 32k	1.5 1.5	3 (bw)	Peklite Backlighted	AGC, EDC SM	
		Member Company								
		Public Service of New Hampshire	23	Dual SIGMA 2	32k ea.	375k wds 1500k wds	2 (bw)		AGC, EDC, SBC, SVC	
4	Dec. 1970	Central Electricity Generating Board London, England		Dual ARGUS 500 + Data Links to 7 Regional Centers	32k ea.	256k wds ea. + 640k wds	17 (bw)[c]	Mosaic[b]	SM, SA, OLF, OSC	
5	Oct. 1971	Kyushu Electric Power Fukuoka, Japan		1-TOSBAC 7000/20 + 1-TOSBAC 3000	32k 8k	4000k wds	2 (bw)	Mosaic[b]	AGC[a], EDC, AVC, SM	SA, SBC, SE, OLF
6	Nov. 1971	Houston Lighting & Power Houston, Texas	160	Dual SIGMA 5	24k ea. + 16k shared	13.5	8	Electro-luminescent	AGC, EDC, SBC, SVC, SM, SA, OLF	
7	March 1972	Norwegian Water Resources & Electricity Board Tokke, Norway		1-NORD 1	24k	128k wds			AGC, EDC, SE, SA, EC, OLF	
8	June 1972	New York Power Pool Albany, New York		Dual IBM 370/155 + Dual Data Links to 8 Member Companies	1000k Bytes ea.	800	10	Datapic Backlighted	SM, SA	AGC, EDC

	In-Service Date	Name of Company	Number of Remotes	Computer System	Core Size in Words	Bulk Storage in Megabytes	Number of Color CRT's	Dynamic Wall Display	On-Line Functions*	
									In-Service	Planned
		Member Companies								
		Central Hudson Gas & Electric	34	Dual CDC SC-1700	32k ea.	3000k wds ea.	3 (bw)		AGC, EDC, SBC, SVC	
		Rochester Gas & Electric	35	1-CDC System 17 +	32k ea.	1000k wds	6		AGC, EDC SBC, SVC	
				2-CDC System 17	32k ea.	1000k wds ea.				
9	Oct. 1972	Tohoku Electric Power Sendai, Japan	37	1-HITAC 7250 +	32k	1000k wds		Mosaic[b]	AGC[a], EDC, SM	AVC, SA, OLF
				1-HIDIC 100 +	8 k					
				1-HIDIC 100 +	16k		2			
				Data Link to Regional Office With						
				1-HIDIC 500 +	24k	256k wds	1			
				1-HIDIC 100	16k	128k wds	1			
10	Oct. 1972	Electric Power Utility Laufenburg Laufenburg, Switzerland		1-IBM 1800 +	64k	17.6	4 (bw)	Mosaic[b]	AGC[a], SM, SE	
				1-IBM S/7	12k					
11	Dec. 1972	Cleveland Electric Illuminating Cleveland, Ohio	32	Dual SIGMA 5 + Data Links to	48k ea.	3 ea.	6	Ferranti-Packard Electro-magnetic Disk	AGC, EDC, SBC, SM OLF, OPF, ASTA, EC	SA, SE SVC, ACR
				4-P2000 Plant Computers	20k ea.	--	--			
12	Feb. 1973	Kansai Electric Power Osaka, Japan		1-HITAC 8300 +	16k	65k wds		Mosaic[b]	AGC[a], EDC, SM	
				1-HIDIC 500 +	16k					
				1-HIDIC 100 + Data Link to 2-IBM 370/158	8 k		3			
13	March 1973	Commonwealth Edison Chicago, Illinois	75	Dual SIGMA 5	48 k ea.	6 ea.	9	Siemens Mosaic	AGC, EDC, SM, SA, OLF	
14	March 1973	Tokyo Electric Power Tokyo, Japan	100	Dual TOSBAC 7000/20 +	32k ea.	3000k wds ea.	1 2 (bw)	Mosaic	AGC[a], EDC, AVC, SM	SA
				Dual TOSBAC 40C	3k ea.	1512 ea.				
15	May 1973	General Public Utilities Reading, Pennsylvania	15	Dual SIGMA 5 + Data Links to PJM and to 3 Member Companies	56k ea.	6 ea.	6	Siemens Mosaic	AGC, EDC, SM	
		Member Companies								
		Metropolitan Edison	50	Dual SIGMA 5 + Data Links to 3 Division Control Centers Each With	40k ea.	6 ea.	3 (bw)		SBC, SVC	
				Dual DS-8000	16k ea.	.256 ea.	6 (bw)		SBC, SVC	

	In-Service Date	Name of Company	Number of Remotes	Computer System	Core Size in Words	Bulk Storage in Megabytes	Number of Color CRT's	Dynamic Wall Display	On-Line Functions[a]	
									In-Service	Planned
		Pennsylvania Electric	52	Dual SIGMA 5 + Data Links to Existing Supervisory Control	40k ea.	6 ea.	3 (bw)		SBC, SVC	
		Jersey Central	109	Dual SIGMA 5 + Data Links to 3 Division Control Centers Each With	40k ea.	6 ea.	3 (bw)		SBC, SVC	
				Dual DS-8000	16k ea.	.256 ea.	6 (bw)		SBC, SVC	
16	July 1973	Interbrabant Schaerbeek, Belgium	20	1-Westinghouse P2000 +	24k	-	1 (bw)	Vynckier[b] Mosaic Quartile	SBC, SM, SE, SA, OLF	
				1-Westinghouse P2500 +	24k	.512				
				1-Westinghouse P2500	28k	2				
17	July 1973	Électricité de France (EDF) National Control Center Paris, France		1-CII 9080 + 1-CII 9040 + Data Links to 5 Regional Control Centers	32k 24k	1000k wds 1000k wds	4	Quartile	AGC[a], SM, SA, SE, OLF	
		Regional Control Centers								
		Paris		1-CII 9040 + 1-CII 9040	32k 32k		4 (bw)	Quartile	SM	OLF
		Lille		1-CII 9040 + 1-CII 9010	16k 32k	3	6 (bw)	Quartile	SM, SA, OLF	
		Nancy		1-CII 9040 + 1-CII 9040	20k 20k	3 3	7	Quartile	SM, SA, OLF	
		Lyon		Dual CII 9040	20k ea.	3 ea.	7	Quartile	SM, OLF	SA
		Marseille		1-CII 9040 + 1-CII 9040	20k 16k	3 3	4	Quartile	SM	SA, OLF
		Toulouse		1-CII 9040 + 1-CII 9040	20k 20k	3 3	7 (bw)	Quartile	SBC, SM	SA, OLF
		Nantes		1-CII 9040 1-CII 9010	16k 32k	3	6	Quartile	SM, SA, OLF	

	In-Service Date	Name of Company	Number of Remotes	Computer System	Core Size in Words	Bulk Storage in Megabytes	Number of Color CRT's	Dynamic Wall Display	On-Line Functions[a] In-Service	On-Line Functions[a] Planned
18	Sept. 1973	Southern Services Birmingham, Alabama	200	Dual IBM 370/158 +	2000k Bytes ea.	500 ea.	12	Planned	AGC, EDC, SM	SA, SE, OLF
				4-ADS 900 + 1 (Spare) +	64k Bytes ea.	--				
				2-IBM S/7 + Video Data Links to Company Dispatch Centers at Alabama Power, Georgia Power, Gulf Power, Mississippi Power			16		SBC, SVC	
				and to 13 Division Control Centers			35		SBC, SVC	
19	Oct. 1973	American Electric Power Canton, Ohio	38	1-IBM 1800 +	64k	3	9 (bw)	Mauell Mosaic	AGC[a], EDC, SE	
				3-HP2116B + 1 (Spare) +	16k ea.					
				Data Link to 1-IBM 370/165						SA
20	Oct. 1973	Philadelphia Electric Philadelphia, Pennsylvania	41	Triple-BURROUGHS 6700 + Data Links to 2 Plant Computers and to PJM (See Item 2)	288k Shared	160	40	None	AGC, SM, SA	EDC
21	Nov. 1973	Hokuriku Electric Power Toyama, Japan	25	1-TOSBAC 7000/20 +	32k	3000k wds	2	Mosaic[b]	AGC, EDC, AVC, SM, SE, EC	
				1-TOSBAC 3000 +	8k	32k wds				
				1-TOSBAC 40	16k					
22	May 1974	Pennsylvania Power & Light Allentown, Pennsylvania	8	Dual SIGMA 5 +	92k ea.	9 ea.	10	Datapic Backlighted	AGC, EDC, SBC	SVC, SM, SA
			300	Data Links to PJM (see Item 2) and to 5 Division Offices Each With						
				1-SIGMA 3 +	32k	.750	10		SBC, SVC	
				1-SIGMA 3	16k	--				
23	Sept. 1974	Carolina Power & Light Raleigh, North Carolina	32	Dual SIGMA 5 +	48k ea.	4.5	6	(Static)	AGC, EDC, SBC, SM	
				2-GEPAC 3010	20k ea.	--				

	In-Service Date	Name of Company	Number of Remotes	Computer System	Core Size in Words	Bulk Storage in Megabytes	Number of Color CRT's	Dynamic Wall Display	On-Line Functions^{a} In-Service	On-Line Functions^{a} Planned
24	Dec. 1974	Bonneville Power Administration Portland, Oregon		Dual PDP-10 + 2-PDP-11 + 1-PDP-11 +	256k ea. 28k ea. 12k ea.	31,744k wds	29		AGC, SM	AVC, SA, BE, OLF
				Dual GEPAC 4010 + Dual GEPAC 3010 + 1-GEPAC 30CS + Future	32k ea. 24k ea. 8k	1792k wds	7	Siemens Mosaic	SBC, SVC	
				Dual SEL85	48k ea.	1500k wds ea.			SBC, SVC	
25	Dec. 1974	Iowa-Illinois Gas & Electric Davenport, Iowa	26	Dual SIGMA 5	32k ea. + 32k shared	9 ea. --	8 (bw)	(Static)	AGC, EDC, SBC, SVC, SM	SA, OLF
26	Mid-1975	Wisconsin Electric Power Milwaukee, Wisconsin	100	Quad CDC SC-1700 +	32k ea.	2-4400k wds	20	(Static)		AGC, EDC, SBC, SM, SA, SE, OLF, OPF
				Dual CDC CYBER 72-13 + Data	49k ea.	12000k wds				
			50	Link to Wisconsin-Michigan Power which has Dual CDC SC-1700 + Video Data Link to 9 Offices	32k ea.	2200k wds ea.	4		SBC	
27	Mid-1975	Rheinisch-Westfälisches Elektrizitätswerk (RWE) Brauweiler, West Germany	80	Dual SIEMENS 306	64k ea.	2304k wds. ea.	5 (bw)^{c} 2 6 (bw)	Mauell^{b} Mosaic		AGC^{a}, SM, SE, SA, OLF, OSC
28	Late 1975	Middle South Services Pine Bluff, Arkansas	31	Dual SIGMA 5 + Data Links to 3 Member Companies	82k ea.	12 ea.	3 9 (bw)	Siemens Mosaic		AGC, ELC, SM, OLF, OPF
		Member Companies								
		Arkansas Power & Light	260	Dual SIGMA 3 + Data Links to 5	40k ea.	3 ea.	4		SBC, SVC	
				Division Control Centers Each With Dual SIGMA 3	2-24k ea. 3-32k ea.	1.5 ea.	16		SBC, SVC	
29	Late 1975	Detroit Edison Detroit, Michigan	150	Dual SIGMA 5 + Data Link to Michigan Electric Power (see Item 1)	64k ea.	27 ea.	13	(Static)		SBM, SVC, SM, SA, SE, OLF

	In-Service Date	Name of Company	Number of Remotes	Computer System	Core Size in Words	Bulk Storage in Megabytes	Number of Color CRT's	Dynamic Wall Display	On-Line Functions[a] In-Service	Planned
30	Late 1975	Tennessee Valley Authority Chattanooga, Tennessee	13	1-SIGMA 5 + 3-GEPAC 3010 + Data Links to 5 Area Dispatch Centers, One of Which has Dual GRI-99 + Video Data Links to Headquarters Office	50k [illegible]0k bytes ea. 32k ea.	12 -- 128k wds ea.	[illegible] 4	(Static)		AGC, EDC, SM, SA, SE, OLF SBC
31	Late 1975	Ontario Hydro Toronto, Canada	88	Univac MP1106-2 + 3-NOVA 1200	262k 24k ea.	80k wds	26	Mauell Mosaic		AGC, EDC, SM, SA, SE, OLF
32	Late 1975	Swedish State Power Board Stockholm, Sweden	150	Dual SIGMA 9 + 2-CDC System 17	192k ea. 48k ea.	54 ea. --	8	Siemens[b] Mosaic		AGC, SM, SA, SE, OLF
33	Early 1976	Portland General Electric Portland, Oregon	50	Dual MODCOMP IV + Video Data Links to 6 Regional Offices	160k ea.	25 ea.	10 6	Mauell Mosaic		AGC, SBC, SM
34	Early 1976	Southern California Edison Los Angeles, California	65	Quad CDC SC-1700 + Dual CYBER 73-14 + Data Links to 8 Switching Centers Each With 1-CDC SC-1700 + Video Data Link to Headquarters Office	28k ea. 65k ea. 7-20k 1-32k	35 356 8.8 ea.	31 8	(Static)		AGC NOX, SM, SA, SE, OLF SBC (at 1 Center)
35	Early 1976	Potomac Electric Power Washington, D.C.	260	Dual SIGMA 9 + 4-SPC 16/65 + Data Link to PJM (See Item 2) + Data Link to IBM 360/65 + Video Data Link to Executive Office	112k ea. 16k ea.	61.8 ea. --	20 5 (bw)	None		SBC, SVC, AVC, SM, SA, OLF, DTA
36	Early 1976	Fuerza Electrica de Cataluna (FECSA) Barcelona, Spain	54	Dual 4010 + Dual 3010	32k ea. 32k ea.	7.5 ea. .262 ea.	10	Siemens Mosaic		AGC, EDC, SBC, SVC, SM, OLF
37	Late 1976	Utah Power & Light Salt Lake City, Utah	85	Dual SIGMA 5	64k ea.	6 ea.	6			AGC, EDC, SBC, SVC, SM, SA, SE, OLF

	In-Service Date	Name of Company	Number of Remotes	Computer System	Core Size in Words	Bulk Storage in Megabytes	Number of Color CRT's	Dynamic Wall Display	On-Line Functions[a] In-Service	On-Line Functions[a] Planned
38	Mid-197[illegible]	Public Service of Oklahoma Tulsa, Oklahoma	56	Dual MODCOMP IV + Data Links to 2 Regional Offices Each With	[illegible] ea.	[illegible] ea.	1[illegible]	Hathaway Mosaic		AGC, EDC, SBC
				Dual MODCOMP II	1-64k ea. 1-56k ea.	5.[illegible] ea.	8			SBC
39	Early 1977	Servicios Electricos del Gran Buenos Aires (SEGBA) Buenos Aires, Argentina	22	Dual MODCOMP IV	128k ea.	26.2 ea.	9	Mauell Mosaic		SM, SA, SE, OLF, OSC
40	Late 1977	Minnesota Power & Light Duluth, Minnesota	50	Dual Xerox 550	96k ea.	20 ea.	13	Siemens Mosaic		AGC, EDC, SBC, SVC, SM, SA, OLF

[a] Implemented by analog controller.

[b] Driven by hard wired logic independent of computer.

[c] Stroke full graphic.

[a] Legend for on-line functions:

ACR	Automatic Circuit Restoration	OLF	On-Line Load Flow
AGC	Automatic Generation Control	OPF	Optimum Power Flow
ASTA	Automatic System Trouble Analysis	OSC	On-Line Short Circuit
AVC	Automatic Voltage/Var Control	SA	Steady-State Security Analysis
DTA	Distribution Trouble Analysis	SBC	Supervisory Breaker Control
EC	Emergency Control	SE	State Estimation
EDC	Economic Dispatch Control	SM	Security Monitoring
NOX	Minimum NO$_x$ Emission Dispatch	SVC	Supervisory Voltage Control

APPENDIX B

SUMMARY OF GUIDELINES AND PITFALLS TO AVOID

This summary was originally prepared in 1973 and has since received considerable industry exposure in the US and abroad. It has been rewarding to find a virtual unanimity of agreement on the points listed in the summary from people who have gone through similar experiences in their own control center projects. Some improvements on the original version have been supplied by some of my colleagues in EEI.

This summary is therefore appended to this report as a synthesis of industry experiences in the design, implementation, and maintenance of system control centers.

I. Analysis and Design

1. Identify and analyze all the needs of system operation.

2. Talk with the system operators--include supervisors in discussions separately from operators. Find out not just what they want but why. What would they do with it after they get it?

3. Examine critically every item requested to be measured, displayed, alarmed, or logged.

4. Do not consider a single processor. You will need another one for testing, maintenance, workload distribution, and backup. An old analog control system is not an adequate backup. No matter how thoroughly software changes are checked out in the background they must be tested in the foreground before being used in the on-line system.

5. A real-time computer with real-time hardware is definitely preferable to a general-purpose business or scientific type computer.

6. A real-time operating system is definitely preferable to a general-purpose operating system with real-time enhancements. Do not try to use a time-sharing operating system for real-time application.

7. An existing, field-proven, real-time operating system is definitely preferable to a non-existent, untried operating system.

8. A standard, fully supported operating system with a few modifications and augmentations to adapt to the application is preferable to a tailor-made, one-of-a-kind operating system.

9. Consider carefully the failover requirements together with the allocation of functions to two or more processors.

10. Avoid complex automatic failover schemes. But you must have automatic re-start.

11. Review carefully the error return philosophy for I/O's (mass storage and CRT) and program execution requests for every program. The philosophy will vary with the function.

12. Check all initialization procedures for programs and tables. There may be several levels of initialization to be considered. Each would have requirements of how, when, and by whom initiated. Each would have requirements as to what is initialized to what.

13. Minimize I/O's and data transfers.

14. As a general rule, provide for manual inputs for every data type and manual overrides for every program (including failover)--via the operator's console.

15. Provide for manual restart of any periodic program.

16. Develop alarming and other program responses to abnormal conditions on the basis that failures or deviations from normal can occur intermittently.

17. Check carefully the electrical source of input data to insure that the data required by a program is indeed monitored.

18. Consider dedicated loggers for specific functions.

19. Review carefully all CRT format designs. Make sure the end-user reviews and approves all diagrams. Question with him the contents of each format. After one-lines are approved, have them field-checked.

20. Make sure the system does not crash on a queue full condition. Examine recovery procedures.

21. Provide a test driver facility on the CRT.

22. Avoid inductive logic. Protect your program from parameter and data errors.

23. Do not fall readily into the decision of coding primary real-time programs in Fortran without a careful investigation and clear understanding of the actual Fortran that you are getting. What are the implications of its use with regard to overhead, shared resources or subroutines, system response, foreground testing?

24. Resist the desire to start program coding right away. Do not take short-cuts in program and file design, test plans, and maintenance plans.

25. At some point after start of system build freeze the design scope; accumulate all subsequent scope changes for implementation after system acceptance.

26. Define your criteria for system functional availability. Shoot for a minimum functional availability of 99.5-99.8 for the man-machine interface.

27. Avoid updates of the same file by two or more programs. If unavoidable, consider making the table core-resident, or scheduling the programs involved so that one runs to completion before the other is started, or using semaphores which are supported by the operating system.

28. Recognize any physical constraints placed on system expansion by file/program designs.

29. Set up simple system documentation procedures which can be followed with a minimum of policing. The procedure should contain provisions

for permanent history, retention of old versions, implementation of temporary patches, addition of new versions of old programs, addition of new programs, file changes, disposition of source material, procedures for new listings, disposition of old listings, maintenance and safe keeping of backup tapes and documentation.

30. Power supply considerations should assure that one power supply failure does not take the system down.

31. Make sure the hardware and software design does not prohibit the future addition of consoles during the life of the system.

32. Console design should be a joint effort (operators, engineers, vendors). Programmers should be able to test the system using one of the operator's consoles or a dedicated programmer's console. A programmer's console should be identical to the operator's console so that actual display interactions can be simulated and tested.

33. Provide a simple means of measuring or monitoring system performance.

II. Procurement and Project Management

1. In the procurement specifications, refrain from requests that cause needless effort on the part of the vendor but add nothing to the bid evaluation.

2. Establish an in-house software team, as early as possible, preferably on or before the time of bid evaluation. The team should be charged with the mission of becoming thoroughly familiar with all details of the system.

3. The team must interact with the vendors' software people. Your specifications should request or provide for a suitable mechanism so that cooperative working relationships may be achieved.

4. Work closely with the vendor. You have the same goals as he has with one difference--you have to live with the system.

5. Among the personnel assigned to work with the vendor at the vendor's site must be one who can make design decisions promptly.

6. Learn assembly language thoroughly. Reserve some real-time programs for your team to write so as to gain insight into the nature of real-time systems.

7. Require your software team to have enough understanding of the hardware with which the computer interfaces, especially the data-acquisition system. Have some team members participate in hardware training courses.

8. Do not give innovative software development to vendors without detailed design specs.

9. The name or experience of a vendor as a corporation will not have too much bearing on getting your system done right. The entire implementation will depend on the individuals assigned to the job, both the vendors' and yours.

10. Establish teamwork. Curb tendency of some programmers to work in isolation.

11. Challenge every program. Try to make it fail. Try to make the system fail. Should the statement be made "No one will ever do that"! --that is exactly what they will do. Try it!

12. Provide facilities for driving analog strip charts for tuning generation control.

13. Provide for statistical data collection on tape for off-line analysis of system characteristics.

14. Insist on diagnostic and test software to be completed prior to system checkout and integration.

15. Insist on file editors and maintenance programs to be completed prior to program testing.

16. Have a primitive CRT software operational prior to program testing.

17. In software management, pay more attention to months and less to microseconds.

18. Make all programmers understand that preliminary resource allocations are not binding and no program should be coded to completion which would slow down system response.

19. Order spare parts and test equipment early, these are as important as the equipment itself.

III. Testing

1. Think out failure possibilities. Try to make the system fail. Verify fail-safe and fail-soft procedures. Test recovery and re-start procedures.

2. Test system response to analog data values of zero and fullscale in addition to normal operating values.

3. Test program response to improper parameter inputs.

4. Test I/O procedures for errors in I/O request as well as for forced failures such as turning device off.

5. Make up standard test procedures or check list for verifying new system builds.

6. Verify man-machine response times. The response time measured from the instant a request is activated to the time that a CRT display is completely on the screen should average at about 1-2 seconds and in the worst case should not exceed 5 seconds. Considerably longer response times are depressing to the human operator.

7. Test the maintenance software thoroughly since the major work effort after shipment will be creating current files.

8. Provide adequate diagnostic error messages, not in code, but in simple English.

9. The acceptance test should include a simulated emergency situation (simultaneous tripout of an entire division) to determine if the system

survives when you need it most.

10. If your system interfaces to other computer systems - lease a phone line from the factory floor to that second system. Test the interface when the "experts" are there for prompt hardware or software repairs.

APPENDIX C

ACKNOWLEDGMENTS

I am specially indebted to the following industry colleagues who took time out of their busy schedules for personal discussions with me on system control center design and operating experiences during my visits to various centers:

Mr. A. H. Willennar	American Electric Power
Mr. J. F. Dopazo	
Dr. L. S. Van Slyck	
Mr. J. Miller	
Mr. D. E. Johannson	Bonneville Power Administration
Mr. I. P. Thurein	
Mr. W. H. Litzenberger	
Mr. D. M. Raikoglo	
Mr. F. Orem	
Mr. T. C. Cihlar	Commonwealth Edison
Mr. E. E. Ciesielski	
Mr. D. H. Hansen	
Mr. R. R. Wozniak	
Mr. T. P. Papson	General Public Utilities
Mr. R. P. Whitsell	
Mr. C. F. Ham	Houston Lighting & Power
Mr. F. Trefny	
Mr. K. Rennie	
Mr. H. R. Koen, Jr.	Middle South Services
Mr. W. R. Barcelo	
Mr. J. E. Brabston, Jr.	
Mr. T. Bradshaw	
Mr. T. R. Woodward	Pennsylvania-Jersey-Maryland Interconnection
Mr. C. F. Doll	
Mr. G. H. Haak	
Mr. J. Evans	Pennsylvania Power and Light
Mr. B. Garner	
Mr. G. LeFever	
Mr. P. M. Davidson	Philadelphia Electric
Mr. J. Furst	
Mr. M. Ewert	
Mr. J. States	
Mr. E. Dudley	
Mr. S. Matraszek	
Mr. R. Dyer	Portland General Electric
Mr. D. Roberts	
Mr. R. O. Usry	Southern Services
Mr. F. I. Denny	
Mr. F. C. Moultrie	
Mr. C. A. Harris	
Mr. J. S. Eiserloh	
Mr. J. G. Tulloss	
Mr. J. R. Huff	Tennessee Valley Authority
Mr. F. Duckett	
Mr. W. C. Stadlin	Leeds and Northrup
Dr. H. A Daniels	TRW Controls
Mr. I. Stander	
Dr. Y. Hirose	Central Research Institute of Electric Power Industry
Dr. K. Ode	
D. Y. Komiya	
Dr. S. Tsuzuki	
Dr. K. Takahashi	
Mr. A. Fukutome	
Mr. H. Suzuki	
Mr. T. Ogawa	
Mr. Y. Oba	
Mr. Y. Ikamoto	Hokuriku Electric Power
Mr. S. Kataoka	
Mr. Y. Fujita	Kansai Electric Power
Mr. H. Kishio	
Mr. M. Aoki	
Mr. T. Minakawa	Tohuku Electric Power
Mr. M. Kouno	
Mr. N. Kobayashi	
Mr. Yokoyama	Tokyo Electric Power
Mr. H. Asakawa	
Mr. Akimoto	
Mr. Koizumi	
Mr. Sinohara	
Prof. Y. Tamura	Waseda University
Prof. S. Narita	
Mr. T. Tsuji	Mitsubishi Electric
Dr. K. Uemura	
Mr. M. Udo	Tokyo Shibaura Electric
Mr. F. Ledger	Central Electricity Generating Board
Dr. U. G. Knight	
Mr. D. March	
Mr. V. Saminaden	Électricité de France
Mr. J. Kowal	
Mr. M. A. LeRoy	
Mr. J. Caneel (BBC)	Electric Utility Leufenburg
Mr. G. Henrion	Interbrabant
Mr. P. Bardiaux	
Mr. F. Markadieu	
Mr. J. P. Evrard	
Mr. J. P. Waha	Societe Intercommunale Belge de Gaz et d'Electricite (Intercom)
Dr. D. Denzel	Rheinisch-Westfälische Elektrizitätswerk
Mr. Hessong	
Mr. P. Goeminne	Laborelec
Mr. R. Wery	
Prof. R. Poncelet	University of Brussels
Dr. K. Reichert	Brown, Boveri & Co.
Dr. G. Schaffer	
Dr. N. Ari	
Dr. A. Sullivan	
Mr. J. Caneel	
Mr. H. Hager	
Ms. S. Salyova	
Dr. D. Rumpel	Siemens AG

I also wish to express my sincere thanks and apologies to all those whom I have met in my visits but whose names, unfortunately, I had failed to keep records of.

I am grateful to those with whom I have communicated to obtain data concerning their control center projects. Without their cooperation this report would be incomplete.

Mr. E. B. Roberts	Carolina Power & Light
Mr. W. P. Challis	Detroit Edison
Mr. K. R. Anderson	Iowa-Illinois Gas & Electric
Mr. F. Petersen	Michigan Electric Power
Mr. R. L. Stromberg	Minnesota Power & Light
Mr. D. E. Stuart	New England Power Exchange
Mr. J. Koehler	New York Power Pool
Mr. J. H. Svoen	Norwegian Water Resources and Electricity Board
Mr. D. Robinson	Ontario Hydro
Mr. S. H. Bouchey	Potomac Electric Power
Mr. F. Ewing	Public Service Company of Oklahoma
Mr. J. Larco	Servicios Electricos del Gran Buenos Aires
Mr. W. M. Davis	Southern California Edison
Mr. J. Lindqvist Mr. L. Pettersson	Swedish State Power Board
Mr. P. Dremann	Utah Power & Light
Mr. W. Rades	Wisconsin Electric Power
Mr. J. C. Russell	Control Data Corporation
Mr. R. M. Parker	Harris Corporation
Mr. E. A. Herrera	Honeywell
Mr. S. E. Grooms	Rockwell International

I acknowledge my gratitude to all my colleagues in the power system engineering field who, over the last few years, have exchanged ideas and information with me on the concepts and implementation of security control.

This report would not have been possible without the work experience and background that I have acquired in the performance of my duties at The Cleveland Electric Illuminating Company. In particular, I have profited immensely from my interaction with the members of the project team who have worked with me in the design, implementation, and maintenace of the CEI System Operation Center.

centro de educación continua
división de estudios superiores
facultad de ingeniería, unam

INGENIERIA DE CONTROL DIGITAL DE PROCESOS Y SUS APLICACIONES

INTRODUCCION (II)

DR. VICTOR GEREZ GREISER

NOVIEMBRE, 1978.

Introducción

COMPUTER CONTROL OF VACUUM DEPOSITION PROCESSES

R. M. Centner and R. A. Wilson
Bendix Research Laboratories
Southfield, Michigan

ABSTRACT

With the advent of the low cost minicomputer, full automatic control of vacuum deposition processes appears both technically feasible and economically attractive. To date, vacuum deposition processes have been largely controlled manually, although simple controllers have been available for controlling portions of the process such as the vacuum pumpdown and the deposition rate during evaporation. Automatic control promises to improve process efficiency and performance, and to improve the uniformity of the resultant products, while freeing personnel from routine operating tasks. The approach to computer control of vacuum deposition processes (evaporation and sputtering) is discussed, and the conceptual design of an automatic process controller based on a minicomputer is presented. The advantages of automating these processes are reviewed.

INTRODUCTION

This paper discusses the application of a small digital computer, or minicomputer, to automatic control of vacuum deposition processes. Included are the establishment and control of the vacuum environment, control of the evaporation process, control of the sputtering process, and control of a number of lesser functions related to these processes. Emphasis is placed on demonstrating the feasibility of applying a dedicated computer to the control of a single vacuum deposition system, although of course other computer/deposition-system relationships may be preferable under certain circumstances.

In the following sections the control requirements for the vacuum deposition processes are reviewed, together with the present methods of control and some of their disadvantages. The approach to computer automation of these processes is then described and the conceptual design of an automatic controller is presented. Finally, it is shown that computer automation leads to improved system efficiency and performance, improved product uniformity, and the freeing of personnel from routine operating tasks. All of these are ultimately reflected as economic advantages.

The following discussion of control requirements and controller design concepts is specifically oriented toward the batch-type vacuum deposition system. Obviously the same general approach can be also applied to the automation of an in-line type system, although the specific control functions will differ somewhat.

VACUUM DEPOSITION PROCESS CONTROL

Control Requirements

The basic vacuum deposition processes covered in this paper are thermal evaporation and sputtering. These two basic processes encompass quite a number of different operations, including:

(1) Vacuum cycle control
(2) Pressure control
(3) Substrate conditioning
(4) Evaporation source control
(5) Sputtering control
(6) Glow discharge cleaning
(7) Substrate rotation
(8) Bell jar and base plate cooling

Each of these functions is a somewhat independent operation, although they must be appropriately grouped and coordinated to yield the desired process sequence. Each of these operations requires control functions. In some, the control is based on the behavior of a sensed parameter relative to a desired or setpoint value. Pressure control and base plate and bell jar cooling are examples of this type of control. In other cases, control is based on a timed sequence, as is generally the case for sputtering and glow discharge cleaning. Evaporation source control is an example of an operation where both bases of control are used: the soak power level is normally maintained for a timed period, whereas during actual desposition source power is usually controlled to yield a specific deposition rate until a specified film thickness is achieved. During the process control sequence, most of the items listed require only simple on-off type control of solenoid valves, power supplies in which the voltage or current levels have been pre-set, and motors. "Pressure control" involves adjustment of a variable valve, while substrate conditioning and evaporation source control may involve the control of variable power supplies. Thus, a vacuum deposition process may include a number of steps or operations, but each operation by itself constitutes a relatively simple control requirement which can readily be automated.

Present Control Methods

To date, vacuum deposition processes have been largely controlled manually, although simple controllers are presently available for controlling portions of the process. The latter are

Reprinted with permission from *1970 WESCON Technical Papers*, vol. 14, paper 4/4.

hardwired, modular devices or units, each controlling a single operation, and are generally limited to two areas: vacuum cycle control and evaporation source control. Evaporation source control is generally accomplished through the combined efforts of two modules or units. One is a monitor unit which determines film thickness and deposition rate, and provides a signal or contact closure when thickness reaches the set point value. The second unit provides a signal for controlling source power during the soak and deposition portion of the cycle, using signals from the monitor unit as the basis for control during the deposition portion.

These methods of control have a number of distinct disadvantages as follows: Frequent attention by an operator is required during the course of the process cycle or run. Even when the previously cited control modules are used, their operation is normally uncoordinated. When the vacuum cycle controller has established the proper environment, the operator is required to initiate the source control cycle or the actual "process". When the latter is completed, the operator must again manually initiate the return of the chamber to atmospheric conditions. Other auxiliary operations, such as glow discharge cleaning, must also be manually introduced in the cycle as required. Thus personnel who might be performing other tasks are tied up in routine equipment operation.

The high degree of operator involvement can also influence the process in at least two other ways. First, since the steps of the process must each be initiated by the operator, unnecessary delays may be incurred between the completion of one operation and the start of the next, thereby reducing the efficiency of the process and increasing the overall run time. Second, since manual control of the process involves a certain degree of operator judgment in some of the steps, the possibility exists for variations in product quality or uniformity from batch-to-batch. All of these disadvantages are ultimately reflected in cost factors which would be improved by automatic control of the process.

Approach to Computer Automation

Automatic computer control of vacuum deposition processes has been technically feasible for some time. The size and cost of the computers which have been available, however, have generally made such automation impractical and economically unsound. Exceptions to this are cases where the computer can be used to control a number of vacuum deposition systems, or where the computer can be used to control a deposition process in addition to performing other duties. The recent advent of small, low-cost minicomputers has changed the picture dramatically. Now an automatic vacuum deposition process controller based upon the use of a small dedicated computer and designed to serve a single system appears to be both technically feasible and economically attractive. It is to this approach that we now direct further attention. The next section describes an automatic controller based on this approach. The advantages of such a controller are outlined in a subsequent section.

An Automatic Vacuum Deposition Process Controller

It was noted earlier in this paper that a vacuum deposition process is made up of a number of different operations, each of which constitutes a straightforward control problem. An automatic controller based on the use of a digital computer can serve to organize, coördinate, and control the execution of these operations.

Controller Functions: The automatic vacuum deposition process controller could be capable of performing the following functions:

(1) Automatic vacuum cycle: control of the bell jar; the vent, roughing, foreline, and hi-vac valves; and the ion tube filament. Protection of the diffusion pump from overheating and/or excessive fore pressure.

(2) Automatic pressure control: control damper valve to keep chamber pressure constant at a preset value for part or all of the operating cycle.

(3) Substrate conditioning control: control heating (to bake or conditioning temperature), annealing, and cooling of the substrate.

(4) Thickness-rate functions: using thickness input signal, calculate deposition rate and determine when thickness reaches set point values. These data would be used by the evaporation source control function.

(5) Evaporation source control: control the cycle of one or two sources (power rise, soak, deposition rate, and shutter).

(6) Sputtering control: turn on preset filament, anode, and target power supplies at programmed point in cycle and maintain for timed sequence. Monitor target current while sputtering. Interrupt timed sequence and sound alarm if current drops below a preset value.

(7) Glow discharge cleaning: turn a fixed power supply on for a preset time interval at any of several pre-programmed points in the cycle.

(8) Substrate rotation: turn substrate rotation motor on and off at predetermined points in the cycle. Fixed speed (manually variable via control not provided).

(9) Bell jar and base plate cooling: turn coolant system on and off. Turn on whenever sensed temperature exceeds a set point value.

Controller Description: The automatic vacuum deposition controller would be based on a small digital minicomputer with a read-only memory. Figure 1 is a block diagram showing the relationship of the controller to the vacuum deposition system, while Figure 2 is a simplified block diagram of the automatic controller itself.

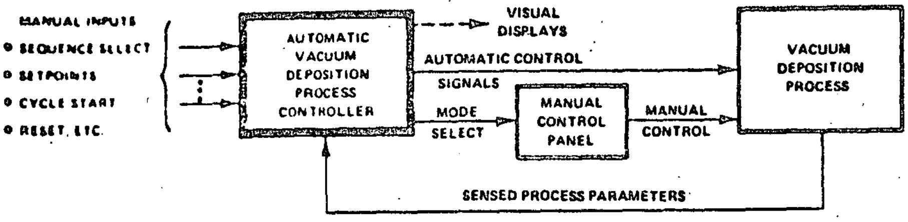


Figure 1 - System Block Diagram

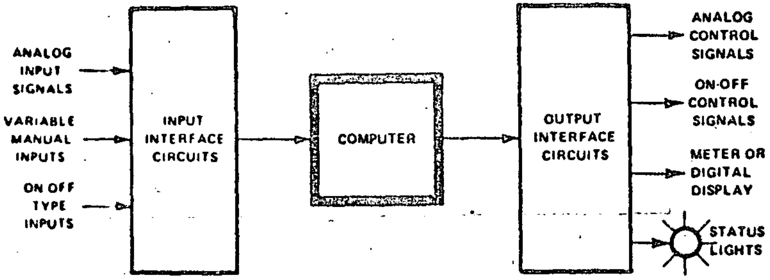


Figure 2 - Simplified Automatic Controller Block Diagram

The automatic controller would be fitted with a front panel typically containing the following: An analog meter and a digital (Nixie tube) readout, each with a function selector switch; a four-digit thumbwheel switch with function selector switch; several toggle and push-button switches; and a number of status or indicator lights.

The unit would have connections at the rear for all input signals and for analog and on-off type output (system control) signals. All variable input signals from external sources are assumed to be analog dc voltages. These would include pressure, temperature,and thickness signals. Normalizing amplifiers would be provided to adjust the relative voltage levels of those signals. The normalized signals are fed to the computer by means of a multiplexer and an analog-to-digital converter.

Variable parameters to be displayed could be read out either on the meter or on the digital display. Variable inputs which are introduced manually (set points, soak power, rise times, etc.) could be introduced via the thumbwheel switch. (Alternative methods of introducing these inputs might include: (1) potentiometers, whose output signals would be sent to the computer via the multiplexer and analog-to-digital converter and (2) a punched card and card-reader arrangement.)

On-off type manual inputs, such as "cycle start", "automatic recycle", and "reset" would be introduced to the computer by means of a status register. On-off signals from the process, such as from bell-jar hoist limit switches will be handled in the same way. Two types of control outputs are provided: Digital-to-analog convertors provide analog voltages for functions where variable control signals are required. On-off type control signals or contact closures are provided for the operation of solenoid valves and solenoid-operated shutters, turning preset power supplies on and off, operating bell jar hoist and substrate rotation motors, and in fact most of the system control functions.

The input and output interface circuits would be mounted on plug-in cards and housed in unused space in the computer cabinet. The entire automatic controller could be packaged in a small bench-top cabinet, or as a small rack-mounted unit, occupying less than 24 inches of panel height.

Once the various manual inputs are set, normal operation of the system consists simply of pressing the "cycle start" button. No further attention is required until the automatic cycle has been completed and the bell jar has been raised. Provisions for reset and other controls would be provided, however, for use when manual intervention is felt necessary.

Advantages of Computer Automation

Computer automation of the vacuum deposition processes has significant advantages with respect to either manual control or the use of separate modular units to automate the control of individual operations.

Figure 3 illustrates the cost advantage of computer automation of the vacuum deposition process, as compared with the use of a number of individual hardwired control modules to accomplish the same objective. The diagram shows relative controller cost versus the relative degree of automation. The cost versus features automated for the modular approach will rise at a fairly uniform rate. The cost of computer automation of only a single operation would be rather high, since it would include the cost of the computer itself. Automation of additional features costs relatively little, however, since this mainly involves a revision to the computer program and the addition of appropriate interface circuits. The crossover point at which the cost of computer automation drops below that of the modular controller approach occurs when only a relatively few operations are to be automated. Modular controllers are not known to be available at present for some of the features included within the scope of the automatic vacuum deposition process controller described herein.

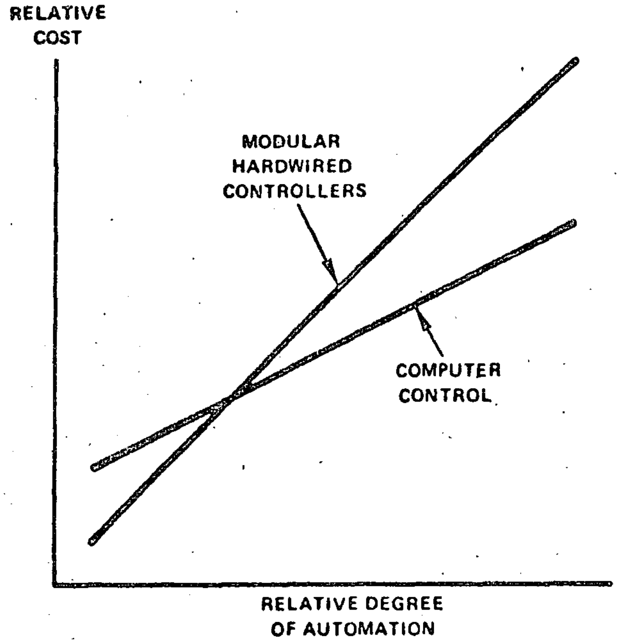


Figure 3 - Cost Versus Degree of Automation

The computer automated controller also results in a number of performance or operational advantages:

(1) Flexibility: For the user having varying process requirements it offers flexibility. The operations of the sequence can be quickly added, deleted, and otherwise altered, and set-points can be established by means of switches and other controls on the panel of the automatic controller.

(2) Process Repeatability: For the user making the same product repetitively, it offers a high degree of process repeatability, once a given sequence has been established and set-point values have been set, resulting in uniformity of the resultant product.

(3) Process Efficiency: The automatic controller will provide smooth and rapid transition from one operation or step of the process to the next, completing the cycle or run in a minimum of time and thus enhancing the efficiency of the process.

(4) Personnel Advantages: Once the sequence and set-point values have been established, the operator is only required to press the "start" button and the complete cycle will be executed unattended. Thus personnel are freed from routine operating tasks.

All of these are ultimately reflected in economic advantages of automatic computer control of vacuum deposition process.

CONCLUSIONS

Although vacuum deposition processes require a relatively large number of control functions, each function is reasonably simple and lends itself quite readily to automatic control techniques. The advent of the low-cost minicomputer now appears to make computer automation of vacuum deposition processes both technically feasible and economically attractive. Automated control offers a number of operational advantages over presently used semiautomatic control methods, many of which are ultimately reflected as additional economic advantages. Hence it may be expected that computer automation of the control of vacuum deposition processes will achieve growing importance in the near future.

Identificación

INDIRECT MEASUREMENT OF PROCESS VARIABLES BY MINICOMPUTER

A. Ben Clymer, P.E.
Consulting Analytical Engineer
Columbus, Ohio 43221

Summary

Mathematical methods and computer techniques for indirect measurement of process variables and parameters are described.

Digital minicomputers are nicely applicable to most tasks of indirect measurement which are too complicated for solution with instrumentation alone. Indirect measurement by computer is desirable and usually feasible when sensors for direct measurement are not available.

The opportunities for such indirect measurements in the service of process control, production accounting, and R&D, are not widely enough appreciated.

The relative advantages of minicomputers (compared with analog or hybrid computers) for indirect measurement are discussed briefly.

The general problem of "indirect identification" of parameters in differential equations is posed, and an approach to the linear case is presented.

Introduction

The purpose of this paper is the stimulation of wider application of minicomputers[1] in process industries for indirect measurement of process variables and parameters by any of a variety of methods.

The paper is aimed at the thousands of engineers in process industries and process control firms whose work requires numerical determination of terms and coefficients in the equations which describe the processes and the equipment in which they take place. The paper is primarily tutorial, pointing out basic mathematical approaches and some of their typical applications in industry. However, the body of the paper does not go into the mathematics at all; it merely refers the reader to helpful treatments in the literature.

The key idea in this paper is that it is often desirable to determine numerical values for dependent variables, forcing functions, or parameters of processes indirectly by use of a computer operating upon other measurements. This possibility is most favorable when the usual practice of direct measurement is impractical. For example, in many situations the required sensor has yet to be developed, and the development effort would be too costly.

Indirect measurement by computer requires the knowledge of some mathematical relationship which holds among the measured quantities and the unknown quantity. This relationship might be a function, an algebraic equation, or a differential equation. The methods differ in detail with the type of relationship, but the indirect measurement principle remains the same.

Indirect measurement is a very common experimental strategy. It is most often implemented with sensors and transducers, which convert measured quantities into the desired quantities. In fact, much of the art of measurement is indirect measurement. Few instruments measure directly the desired quantity. In none of these applications of indirect measurement is a computer necessary, because of the simplicity of the mathematical relationship involved (usually merely a direct proportionality).

The type of indirect measurement which is of concern in this paper is that in which the relationship is sufficiently complicated to render infeasible any hardware short of a small computer. There have been already many isolated applications of this idea, each a clever idea independently conceived. However, there is no standard doctrine or art which an engineer can follow and exploit. The spread of the idea has been haphazard. Thus, although indirect measurement by computer is by no means novel, it deserves wider appreciation as an available approach, a greater degree of formalization of methods, and intensified universal application.

The chief industrial areas of application of the approach are process control, production accounting, and R&D. Indirect measurement by computer is a necessary preliminary to automation of the control of most processes. Many engineers in the management of process industries have made remarks such as the following: "Process control automation is well and good in some industries, but in our unique processes we don't know how to measure some of the chief variables, so why should we think seriously about computer control of our processes?". They do not realize that indirect measurement by computer is a technique which is probably applicable to their problems. Similarly, accountants faced with determination of process inventories and throughputs are not yet accustomed to thinking in terms of indirect measurement by computer as a possible solution to their problems. Likewise, engineers and scientists in an R&D laboratory are more likely to think of Rube Goldberg devices than of computers for measurement, unless their laboratory is well stocked with small computers for just such purposes. Thus, most potential applications lie unimagined.

The organization of the subject which seems most appropriate to this paper is a treatment by

Reprinted from *IEEE Trans. Ind. Electron. Contr. Instrum.*, vol. IECI-17, pp 358-362, June 1970.

considering one type of mathematical relationship at a time. Each type of relationship gives rise to its own set of mathematical methods of calculation of the desired quantities. Moreover, each set of methods is best suited to particular types of computer hardware (analog, digital, or hybrid).

Function Evaluation

The simplest type of mathematical relationship which is relevant to indirect measurement by computer is an algebraic function. Given some measurements of a variable at certain points and times, one is required to determine the variable at other points or times. The general class of methods which is applicable to this problem is function determination and evaluation. Usually the most suitable function to use is a series, such as a power series (Taylor series) or Fourier series.

One class of methods consists of two steps: (1) determination of the coefficients of the series, (2) evaluation of the function at a particular point or time. The first step is accomplished by application of numerical methods of curve fitting, such as "least squares" (minimization of the sum of the squares of the errors), or Fourier series determination. The second step is accomplished by substitution of coordinates and times into the series formula in order to calculate a specific value of the function.

Another class of methods proceeds in one step. In these methods there is derived a direct formula for interpolation or extrapolation of the given measurements. The type of function used is implicit in the formula, having been assumed in the derivation.

There are many situations in process industries in which function evaluation lends itself to indirect measurement by computer. One category of common situations is a one-dimensional grid of sensors continually giving spatial samples of a variable, such as temperature, from which one is to measure indirectly the temperature at some point which might or might not be a grid point where there is a sensor. For example, one could save the cost of additional sensors by interpolating among readings from a smaller number of sensors than would otherwise be necessary. Also, one could monitor the integrity of all sensor readings by comparing each with the local function value obtained from all of the other sensors, so that malfunctions of sensors or signal lines could be detected promptly and so that synthesized signals could be used in the time during which repairs are being made. In fact, if individual sensors have a significant uncertainty in their readings, none of the readings need to be used directly; instead one can work with only synthesized signals in which individual sensor errors have been washed out to some extent by least squares, thereby increasing the precision of measurement. Another purpose might be to get synthesized measurements at points which are inaccessible.

Similarly, one can interpolate or extrapolate along the time dimension, such as for prediction of a future value of a variable. Thus one can monitor trends, such as slow drifts, in order to take corrective action well in advance of critical need.

More generally, one can determine indirectly a variable in two or three dimensions and time from a grid of sensor readings.

All applications of function evaluation can be implemented with any kind of computer hardware. However, hybrid hardware would never be necessary, so the choice lies between analog and digital. In many cases a digital minicomputer would be most satisfactory and economical, but it would not be prudent to make categorical assertions of universal superiority, disregarding particular conditions surrounding a specific application.

Algebraic Equation Solving

A more complicated type of mathematical relationship which can underlie indirect measurement by computer is one or more algebraic equations. In general, one is dealing with measurements of several different variables, rather than with measurements of the same variable at different points or times.

One classification of methods is in terms of explicit and implicit methods. Explicit methods enable direct calculation; implicit methods are indirect or iterative. Usually direct methods are preferred when the equations are linear, whereas implicit methods might be necessary for nonlinear equations. The most common explicit method for a system of linear equations is matrix inversion, for which many algorithms exist. Even nonlinear systems are often treated in a two-stage process ("quasilinearization"), in which an iterative loop contains a local linearization in the forward path.

A simple example of algebraic equation solving would be the use of the gas law for calculation of the density of a gas at known temperature and pressure in a vessel of known volume. Another example would be the calculation of the weight of a gob of glass, knowing its density, given a few measurements of dimensions in two orthogonal profiles, and making a few assumptions about the shape in three dimensions. Another example would be calculation of the steady-state temperature of an object at an interior point, given sufficient boundary measurements and a formula for the temperature profile. Similarly, one can use algebraic equation solving to find a material property parameter, such as thermal conductivity, given an appropriate formula and given sufficient temperatures, or such as viscosity from velocity or flow measurements. A frequently occurring example of matrix inversion is determining chemical composition of a mixture from a set of readings of spectral peak responses from some instrument. Similarly, one can identify something in a field of view by processing with matrix inversion the outputs from parallel measurements through different filters in a "remote sensing" system.

The hardware appropriate to algebraic equation solving applications could be either digital or analog; hybrid hardware is not necessary. Digital computers are better suited than analog for solution of large sets of linear algebraic equations or for sets of equations in which the matrices are ill-conditioned. Digital computers can be used satisfactorily also for most sets of nonlinear algebraic equations. Analog computers are especially powerful and convenient in cases in which nonlinear equations are to be solved by implicit methods, since control of the gains of the forcing loops is best designed by thinking like that of a control engineer seated at the console of a general-purpose analog computer. In contrast, reliance upon standard algorithms in a digital computer is likely to be risky for solution of nonlinear equations, since one can encounter phenomena such as failure to converge or jumping to a wrong root. On the other hand, the limited dynamic range and precision of analog computers can be a severe handicap in dealing with an ill-conditioned system or a system in which the variables range over several decades of magnitude[2].

Differential Equation Solving

Another general class of methods of indirect measurement by computer is differential equation solving. A differential equation is the appropriate relationship to use when a system's behavior involves the rate of change of the dependent variable. Examples are chemical kinetics and transient temperature distribution. A differential equation arises also if a nonlinear algebraic equation is deliberately converted to an initial value problem for solution, as in the case of the method of steepest descent[3]. If there is only one independent variable (usually time), one has an ordinary differential equation, but if there are two or more independent variables (usually time and one or more space variables) then one has a partial differential equation. In this paper only ordinary differential equations will be considered, since a partial differential equation can readily be converted to a set of ordinary differential equations (such as by property lumping, finite differencing, or by use of normal or assumed mode amplitudes as generalized coordinates).

There are two general problems associated with differential equations: (1) the parameters and forcing function are known, but the dependent variable is to be determined at some particular value of the independent variable, and (2) the dependent variable and forcing function are known as continuous functions of the independent variable, but one or more of the parameter values are unknown. The second problem could be construed to contain also the problem of determining a forcing function, knowing the parameters and the dependent variable. The first problem is merely conventional "solution" of the differential equation; the second problem is the "identification" or "parameter determination" problem. A differential equation can be solved by integration and algebraic operations. Identification, however, is a more complicated procedure, for which many methods are known.

If there is only one unknown parameter (or sometimes even if there are two), the method of "implicit synthesis"[4] can be used. A popular method for any number of unknown parameters is the method of steepest descent[3]. There are also other methods which make use of the local gradient of the error in parameter space. Also there are methods of systematic and/or random search[5].

A more difficult task is identification of a function appearing as a coefficient in a differential equation. Methods for dealing with this problem have been developed[4,5,6,7]. One approach is to represent the function as a series expansion in which the constant coefficients can be found by parameter determination methods. Another approach is to treat the function as if it were a constant, then plot its value continuously against the variable upon which it is presumed to depend, thus revealing the function as a curve[4].

A still more complicated problem is parameter determination in a situation in which the dependent variables cannot feasibly be measured. Since there seems to be no prior art for this problem, a preliminary treatment of it is appended to this paper.

A common class of applications of the identification of forcing functions arises in the case of ambient disturbances of a process. For example, a thermal control system might need to determine the net effect of ambient fluctuations of temperature or convection velocity in order to correct for them.

It is more frequently necessary to determine parameter values. The classic problem of this type is determination of constant coefficients in a linear differential equation. Simple examples are afforded by the equations of motion of an airframe[7,8], whose aerodynamic coefficients might be unknown; the transfer function of a human operator[7,9]; the force coefficients of a tire[7]; the dynamic properties of an artery[10]; and the parameters of an ecosystem[11] or physiological system[12].

All kinds of computer hardware have been applied to problems of identification and solution of differential equations. Hybrid computers are either necessary or economically desirable for very large problems of identification, such as determination of the coefficients (rate constants) in the differential equations for chemical kinetics. This type of problem is so widespread in the process industry that special hybrid and digital software is available (e.g., Electronic Associates Inc.'s OPTRAN[13]). Analog computers are especially valuable as tools during development of new applications of identification, but they can be used also for on-line identification in most cases. Digital computers are less commonly used, but they too are widely applicable. If the set of differential equations is not too large, a minicomputer can do the job nicely.

Conclusions

In summary, there are many points to be made in favor of the use of minicomputers in the process industries for indirect measurement:

1. A minicomputer is nearly always applicable.
2. A minicomputer requires less investment than the smallest available general-purpose hybrid computer, so it is preferable when applicable.
3. Any digital computer has more precision than an analog computer, a fact which is sometimes important.
4. A digital computer has a much greater dynamic range than an analog computer, enabling it to handle easily process problems involving a wide range of values of the variables (such as in start-up of a nuclear reactor).
5. A minicomputer is preferable to a larger digital computer for many purposes and in several respects, such as feasibility of full-time commitment, quick computer reassignment, programming flexibility, and avoidance of conflict among users.

Why minis over maxi's

6. If a selection of minicomputers is available, one can be chosen which accommodates the job most nicely and economically.
7. Often the same minicomputer which is used in process control development for indirect measurements can be used also later for automatic control.

Multiple use.

8. If a multiplicity of minicomputers are used in the bottom echelon of an automatic control system, it is fairly simple later to put them all under the supervision of a single higher-echelon digital computer; this step would be more difficult if the lower-echelon control and indirect measurement computers were analog.

¡¡ Que viejo lo cuando se integran !!

Appendix

One occasionally encounters a problem of parameter identification in which it is not practical to measure some or all of the dependent variables, although it is possible to measure their sum or weighted sum or some other function of them. This might be called a problem of "indirect identification", in which name the word "indirect" refers to the fact that the differential equations for the dependent variables cannot be used directly in the identification algorithm.

One example of such a problem would be in the chemical decomposition or radioactive decay of particles which cannot readily be distinguished from one another, whereas one is able to count the total number easily. The task is to determine the values of the decay rate constants or half lives. A closely related problem is determination of the coefficients in differential equations for competition among species, when it is not desirable or feasible to make separate population counts for each species.

One approach to such a problem, which is possible when the differential equations are linear, is to use Laplace transforms[14] to derive a differential equation for the sum of the variables, the coefficients of which, when identified, determine the desired parameters. (The same procedure is applicable to the basic and crucial problem of deriving a differential equation for an aggregated dependent variable for use in an upper-echelon model in a hierarchy of models[15,16].)

Consider, for example, a system represented by two uncoupled differential equations of first order:

$$\begin{cases} \dfrac{dP_1}{dt} = \lambda_1 P_1 + \mu_1 \\ \dfrac{dP_2}{dt} = \lambda_2 P_2 + \mu_2 \end{cases}$$

Here the dependent variables P_1 and P_2 cannot conveniently be measured individually, it will be assumed, whereas it is feasible to measure their sum, say P. The four parameters λ_i and μ_i are to be determined, knowing the time histories of P and its derivatives. The Laplace transform procedure gives:

$$\mathcal{L}(P) = \mathcal{L}(P_1) + \mathcal{L}(P_2)$$

$$= \frac{\mathcal{L}(\mu_1)}{s-\lambda_1} + \frac{\mathcal{L}(\mu_2)}{s-\lambda_2}$$

$$= \frac{(s-\lambda_2)\,\mathcal{L}(\mu_1) + (s-\lambda_1)\,\mathcal{L}(\mu_2)}{(s-\lambda_1)\,(s-\lambda_2)}$$

Then the desired differential equation is:

$$\frac{d^2P}{dt^2} - (\lambda_1 + \lambda_2)\frac{dP}{dt} + \lambda_1\,\lambda_2\,P$$

$$= \frac{d\mu_1}{dt} - \lambda_2\mu_1 + \frac{d\mu_2}{dt} - \lambda_1\mu_2$$

This differential equation can then be used as the basis for identification of the sum and product of λ_1 and λ_2, hence the values of λ_1 and λ_2 individually, and hence the sum of μ_1 and μ_2. In this case it does not seem possible to separate μ_1 and μ_2, however, unless either P_1 or P_2 can be measured.

A similar procedure applies to a case in which any number of differential equations are coupled or in which a weighted sum of the dependent variables can be measured. However, the Laplace transform procedure cannot be applied to nonlinear differential equations. New methods are needed for such problems, which could arise in chemical kinetics, ecology[17], physiology, etc. The writer does not know of any work done on these problems.

References

1. Bartik, J. J., "Minicomputers Turn Classic", Data Processing, Jan. 1970, pp. 42-50.

2. Davidson, R. S., and Clymer, A. B., "The Desirability and Applicability of Simulating Ecosystems", Annals of the New York Academy of Sciences, vol. 128, art. 3, Jan. 31, 1966, pp. 790-794.

3. Rogers and Connolly, "Analog Computation in Engineering Design", McGraw-Hill, New York, 1960, Chap. 6.

4. Clymer, A. B., "Direct System Synthesis by Means of Computers", Communication and Electronics (Trans. AIEE), Jan. 1959, pp. 798-806.

5. IEEE Transactions on Automatic Control have contained the greatest concentration of new methods of identification.

6. Kohr, R. H., "On the Identification of Linear and Nonlinear Systems", Simulation, vol. 8, no. 3, March 1967, pp. 165-174.

7. Potts, T. F., Ornstein, G. N., and Clymer, A. B., "The Automatic Determination of Human and Other System Parameters", Proc. of the Western Joint Computer Conference, May 1961, pp. 645-660.

8. Rubin, A. I., "Continuous Regression Techniques Using Analog Computers", IRE Trans. on Electronic Computers, Oct. 1962, pp. 691-699, vol. EC-11, no. 5.

9. Ornstein, G. N., "Applications of a Technique for the Automatic Analog Determination of Human Response Equation Parameters", Ph.D. dissertation, Dept. of Psychology, The Ohio State Univ., 1960.

10. Stacy, R. W., "Diagnosis of Arterial Disease with Analog Computers", IRE Trans. on Medical Electronics, vol. ME-7, no. 4, Oct. 1960, pp. 269-273.

11. Bledsoe, L. J., and Van Dyne, G. M., "Evaluation of a Digital Computer Method for Analysis of Compartmental Models of Ecological Systems", Oak Ridge National Laboratory, ORNL-TM-2414, Feb. 1969.

12. Clymer, A. B., and Graber, G. F., "Trends in the Development and Applications of Analog Simulations in Biomedical Systems", Simulation, April 1964, pp. 41-60.

13. Carlson, A. M., "A Hybrid/Digital Software Package for the Solution of Chemical Kinetic Parameter Identification Problems", Proc. of the Fall Joint Computer Conference, 1969, pp. 733-750.

14. Gardner and Barnes, "Transients in Linear Systems", John Wiley & Sons, New York, 1942.

15. Clymer, A. B., "The Modeling and Simulation of Big Systems", Proc. of the Simulation and Modeling Conference, Pittsburgh, Pa., April 21-22, 1969, pp. 107-118.

16. Clymer, A. B., "The Modeling of Hierarchical Systems", keynote address for the Conference on Applications of Continuous System Simulation Languages, San Francisco, Cal., June 30, 1969, Proceedings pp. 1-16.

17. Clymer, A. B., and Bledsoe, L. J., "A Guide to the Mathematical Modeling of an Ecosystem", Ford Foundation Workshop on Ecological Effects of Weather Modification, Albuquerque, New Mexico, June 1969 (available from authors).

The Mini Computer as a Control Element

Dudley B. Hartung
Management Methods, Inc., Waltham, Massachusetts

Introducción.

Mini computers have been used to control a wide variety of processes and functions including machine tools, chemical processes, steel mills, and warehousing systems. Articles in technical journals and talks at seminars have described in some detail many of these individual applications. But what is a good application -- when do you use a hard wired control system and when should you consider a mini computer?

The decision must be based on costs -- dollars per function -- and reliability and maintainability. In general, reliability can be disposed of as being indirectly related to dollars. The simplest or mass-produced system is normally the cheapest system and also the most reliable system. Maintainability can be given a cost value. The decision, therefore, can be directly related to costs. Costs naturally refer to the initial capital cost of the equipment and also to recurring cost of operation, including the aforementioned maintainability cost, operator cost, quality of product value, etc. Some of these costs can only be roughly approximated and may be intuitive guesses. The cost of the equipment, however, should be fairly easy to derive by knowledgeable people during initial planning stages.

A control system -- any control system -- consists of inputs, outputs, and decision makers. In comparing hard wired systems with computer systems, the input and output devices probably stay comparable in cost. Input and output devices consist of operator switches, sensors, solenoids, servo or discrete (on-off) motor controls, etc. The decision maker is the logical system which determines the effect of input changes upon output actions. With a hard wired system, each subdecision or each function of the control system has its own logic. A mini computer time shares its logic to accomplish many functions with a relatively small logical device. The mini computer, therefore, becomes essentially the complete decision maker, even when there are hundreds of inputs and outputs with varying degrees of interrelationship. This is where the cost savings of a computer system come. Many more decisions can be made per dollar with a computer compared to hard-wired logic.

There are some inherently costly aspects of a computer system. Holding functions must be stored externally. All inputs and outputs of data are in high speed serial words which means that switch inputs, for instance, must be held on until polled by the computer. The holding device might be the operator's finger. The outputs must have holding relays or their solid state equivalent. Inputs and outputs to the computer are always the same binary words at low levels, requiring filtering and level conversion at the interface. No power is available for force-type functions, requiring amplifiers and power relays. Of course, some of these restrictions apply to many hard-wired controls. But if the decisions are very simple, the input and output buffering, filtering, and holding may be more expensive than the complete hard-wired control.

The computer itself is limited by speed and by the size of internal memory utilized for storing data and computer program. It may be cheaper even in a computer controlled system to do some complex but frequently used and repetitive functions externally. For example, a servo loop could be performed in a computer but in most cases is done externally. Interpolation for a machine tool, which is the precise control of velocities in two or more axis to draw a straight cut or a circular cut is expensive in computer time in that it takes a large portion of a computer. A single computer can do all interpolation and control one or two high speed, high accuracy machine tools. If the interpolation is done externally, 5-20 machine tools can be similarly controlled.

So how is a decision made to go hard-wired or mini computer? The system costs must be estimated in both ways. This requires some understanding of the end of process and the requirements of both a hard-wired system and the capabilities of computers. In many cases, the computer can supply additional functions at very low cost which have to have some value placed on them to honestly compare systems. In other cases, the function to be performed is so complex that it is immediately obvious the computer is the solution. Labor costs of both of the design and building of a system and the operation must be considered.

Reprinted with permission from *1970 Joint Automat. Contr. Conf. Amer. Automat. Contr. Council,* paper no. 8-B, pp. 191–192.

Costs also include effects of lead time variation, set-up speed, and rejects. All of these costs vary and relate to a particular application.

Rules of thumb are dangerous and can be misleading, but there are some systems where computer control should be looked at very carefully. If many simple decisions -- or many monitoring points, such as those on a transfer line, are required or if very complex relay trees or logical decisions must be made, a computer should be considered. Complicated decisions requiring mathematical functions, particularly if changing either between runs or over a period of time or the requirement for a great deal of stored data for look-up tables or individual parts programs suggest computer control. Finally very specialized problems or machines or processes with only one system or a few systems being built, particularly where modifications between initial concept and final operating equipment are foreseen due to technical unknowns, are particularly good applications for computer control. This i true not only because of the possible savi in hardware costs, but more importantly, because of the normally much lower design cost.

The mini computer can be a panacea for many ills, and should be looked at by the builders and users of any controlled system. It will be found that not all systems justify on an economic basis the utilization of computers, but conversely, it will be found that what seems like an expensive and sophisticated control system can often be easily justified purely on an economic basis.

The place of digital backup in the direct-digital control system

—*by* J. M. LOMBARDO
The Foxboro Company
Foxboro, Massachusetts

flexib.
adv. control
Less cost
Introduc.

INTRODUCTION

The key to the success of direct digital control on large industrial processes lies in its flexibility in implementing everyday process control problems as well as advanced control at lower overall system cost. Control concepts for continuous processes use the computing, monitoring, information storage and analytical ability of the direct digital control computer. In the batch or discontinuous process the computer's logic capability is emphasized. To perform batching operations, a comprehensive logic system is necessary. Implementation of such a system using digital techniques provides many advantages over implementation using analog equipment with auxiliary digital logic circuits.

To fully appreciate these advantages, the reader must have a basic understanding of continuous control systems as well as the batch type systems. The following will describe single loop control, several advanced control concepts and control of semicontinuous processes, as an introduction to digital computer application and backup.

Single loop control

Simple single loop feedback control is the most common control found in the process industries. It is used for controlling flow, level, temperature, pressure and many other variables. Both pneumatic and electronic devices are available which provide this type of control.

Sim
Loop
Most
Flo
wh
is

Basically, these controllers compare the measurement of a variable with its desired value or set point. If the two values are not equal, the controller adjusts a control value to minimize the difference (Figure 1).

In action, the controller is an analog computer which calculates a one, two or three term expression,

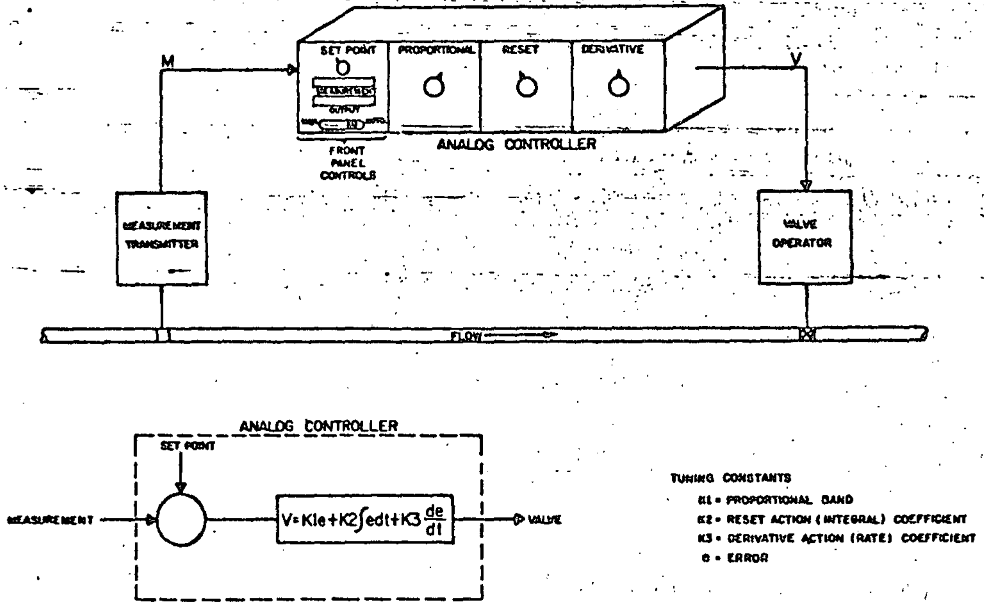
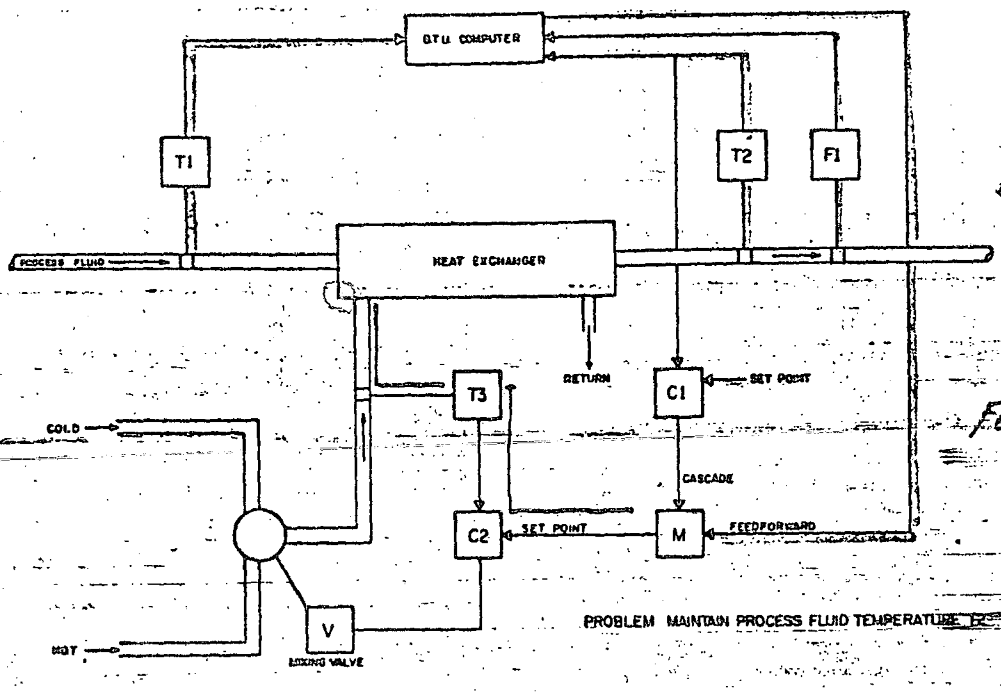


Figure 1—Typical single variable feedback control loop

Reprinted with permission from *AFIPS Conf. Proc., Vol. 30, 1967 Spring Joint Comput. Conf.*, pp. 771–778.



INFE-
RENTIAL

Feedforward.

Figure 2—Advance control techniques applied to a heat exchanger

depending on the type of control action required by the process. The three terms define proportional, reset and derivative control action. During process start-up, coefficients of the three terms are manually set on the controller to provide the best response under normal operating conditions. If operating conditions change, or the process operator changes the set point radically, the coefficients are no longer at optimum values.

Advanced control concepts

As the control problem becomes more complicated, single loop feedback control is no longer sufficient. Figure 2 illustrates three types of advanced control: inferential, feedforward and cascade.

In the inferential control, a relationship is calculated between two or more measurements which is used to control the desired but unmeasurable variable. In Figure 2, the Btu computer performs a calculation based on the difference between the outlet and inlet temperatures to the heat exchanger (T2—T1) and the flow F1 of process fluid through the heat exchanger. This calculation—a measure of the heat transferred to the process fluid—determines the demand of hot or cold fluid needed to maintain process fluid output temperature T2.

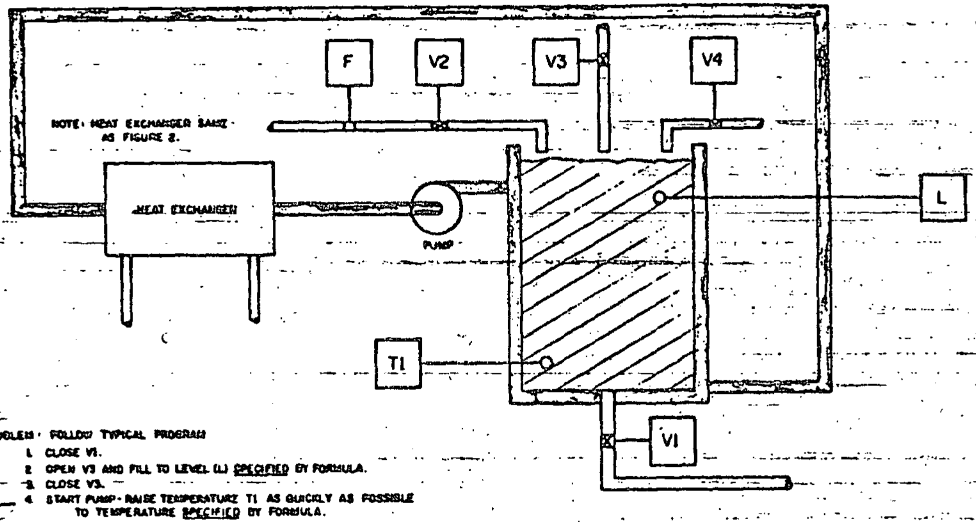
Analog computing devices perform the necessary calculations and control can be executed with conventional analog control devices. Additional calculations may be necessary before some variables are obtained. For example, the differential pressure signal provided by the commonly used orifice plate is proportional to the square of the flow. A computing element is therefore necessary to extract the square root of the differential pressure signal.

Figure 2 also illustrates feedforward control. The calculation of heat transfer (Btu) rate is "fed forward" to adjust the flow of heating or cooling fluid and change temperature T3. This feedforward calculation anticipates disturbances in both inlet temperature T1 and process flow F1. To provide more stable control of T2, the feedforward signal anticipates the change in heat input required. The magnitude of the feedforward action is usually determined by experimentation and may have to be adjusted periodically, since the heat transfer characteristics of the heat exchanger change with age.

Val
Tibo
Schmidt

A third control technique illustrated by Figure 2 is cascade control—a technique where one controller adjusts the set point of another controller. The output of temperature controller C1 is fed (cascaded) to the set point of temperature controller C2 through a multiplying device M. Hence changes in process fluid output temperature T2 affect the set point of controller C2 to ultimately maintain output temperature.

The control loops discussed have been applied to continuous processes which operate at near steady conditions with only nominal process or set point disturbances. Therefore, adjustment of the proportional, reset and derivative coefficients is rarely necessary and set point changes are nominal. In a steady, con-



PROBLEM: FOLLOW TYPICAL PROGRAM

1. CLOSE V1.
2. OPEN V3 AND FILL TO LEVEL (L) SPECIFIED BY FORMULA.
3. CLOSE V3.
4. START PUMP - RAISE TEMPERATURE T1 AS QUICKLY AS POSSIBLE TO TEMPERATURE SPECIFIED BY FORMULA.
5. WHEN TEMPERATURE IS STABLE, OPEN V4 AND RAISE LEVEL (L) TO SPECIFIED HEIGHT.
6. CLOSE V4 AND OPEN V2. BY FLOW MEASUREMENT (F), METER AMOUNT SPECIFIED.
7. CLOSE V2 AND RAISE TEMPERATURE T1 AT SPECIFIED RATE OF RISE UNTIL SPECIFIED TEMPERATURE IS REACHED.
8. HOLD TEMPERATURE (T1) FOR SPECIFIED TIME
9. REDUCE TEMPERATURE (T1) TO SPECIFIED VALVE AND HOLD FOR SPECIFIED TIME.
10. STOP PUMP AND OPEN DRAIN VALVE (V1).
11. START CYCLE OVER WITH NEW SPECIFIED PARAMETERS IF NEED BE.

Figure 3—Simple batch control sequence

tinuous well-behaved process, use of these adjustments would be very limited. Many high production petrochemical processes are in the continuous process category.

Control of semicontinuous processes

Figure 3 presents a process control problem where steady operating conditions are not maintained. This type of process requires a control system which changes operating conditions according to a preplanned event/time schedule. Batch or semicontinuous processes require controlled sequencing because various equipment must be started and stopped frequently, product requirements change frequently and operating parameters change. It should be noted that most batch or continuous processes still use feedback control, but with programmed changes of control set point.

Figure 3 illustrates a simple chemical reactor. Ingredients are added sequentially and temperature is maintained according to various preset programs to provide the chemical reactions necessary for various products. The reaction within the vessel can vary from endothermic to exothermic during the production cycle. Hence in order to hold a set temperature, the control system may be required to switch from heating the reactor to cooling it when the reaction starts to generate its own heat.

In the typical chemical reactor or mixing vessel, different control sequences may be necessary for each new product. For instance, there may be changes in specified ingredient mix and heating and cooling temperatures and temperature rates of change. Process control problems of this nature require more complex control than the feedback, feedforward and multivariable controls previously described. This control requires programmed sequencing of events, including equipment starting and stopping.

In Figure 3, the control of reaction temperature T1 is basically a feedback control problem. However, the problem is complicated, since T1 must change at the proper times, sometimes in step-wise fashion and other times at a controlled rate. Also, the sequence of events must be readily changed, depending on the intended product.

Combinations of special purpose digital and analog control equipment have been built which satisfy the demands of the discontinuous process. However, the programming of this equipment is relatively inflexible and the control cannot be well-tuned because of the cyclic nature of batch processes. Many of these systems are not used at full operating speed, since the control constants are a compromise.

Lees

Applying the digital computer

Digital computers are of significant interest to the industrial process control field due to their ability to store programs, calculate simple and complex control relationships, compute variables which are not directly measurable, monitor the process and take action according to a preplanned schedule. The digital computer easily performs tasks that the analog system finds difficult; it can be easily programmed to adapt the overall control system to changes in process

En
la
de
ref
lo

dynamics, materials, equipment and production demands. Because of this versatility, digital computers are being designed and installed in continuous process plants as well as in batch process plants. Many of the installations use direct digital control techniques on all or some of the control problems.

Table I compares two systems, each using direct digital control exclusively. As shown, a continuous process application in an oil refinery has 530 analog measurements of which 275 are associated with control calculations, the other inputs are for performance monitoring and system operation analysis. Of the 275 control inputs, 180 are used for direct control of simple loops; the remaining 95 are used in advanced control. Therefore, approximately one-third of the 275 inputs associated with control are used to implement multivariable and advanced control techniques.

Table I – Comparison of computer system input/output between continuous and batch process control

	CONTINUOUS PROCESS	BATCH PROCESS
TOTAL ANALOG INPUTS	530	620
ANALOG INPUTS IN CONTROL LOOPS	275	240
CONTROL LOOPS		
SINGLE	180	225
CASCADE	30	70
FEED FORWARD	15	0
DIGITAL INPUT (CONTACTS)	210	1225 *
DIGITAL OUTPUTS (ON-OFF)	355	1300

Table I also shows the input/output distribution for a large batch control installation currently being implemented by a digital computer system. A comparison of the batch with the continuous process reveals a significant increase in contact sensing elements and on-off control outputs. In order to sequence events, the batch system must sense the status of process equipment and conditions. Also, more devices must be turned on and off. With the batch system, man-machine communication needs also increase. Increased number of push buttons, signal lights and the increased size of digital displays require more digital inputs and outputs.

It is also significant that the number of control outputs (295) can exceed the number of analog inputs (240) in the batch system. This situation occurs in batch processes because the same measurement can be used in control of different control elements and with different control algorithms, depending on the sequence of events and the starting and stopping equipment.

The philosophy of DDC

With the introduction of the digital computer to the process control field, it became evident that relatively little was known about most processes. Most processes could not be adequately represented by mathematical models which would permit improved process control.

Kv
model
model

Early attempts at applying the digital computer emphasized supervisory control in which the computer adjusted the set point of an analog controller. In these systems, the analog controller retained the last computer control setting if the computer failed. On continuous processes, this control was quite satisfactory; in fact, once the system was operating satisfactorily, it made little difference whether the computer was there or not. The operator could still adjust control actions, as he did before the installation of the supervisory computer. This made the process operators happy, but in many instances the process engineers and plant supervisors were not. There was no guarantee that the operators would achieve the optimum control settings for the plant.

What additional advantages did the computer provide? If so desired, the computer could make feedforward, cascade and inferential calculations which would optimize control set points for economic or production considerations. Economic constraints relating to material balance, throughput, inventory, etc., could be developed. In a sense, an economic mathematical model was possible, whereas a process model was still difficult to achieve, due to lack of process knowledge. In addition, the on-line process computer performed other useful work to aid operators, plant supervisors and process engineers: see Table II.

Table II – Some non-critical functions of an on-line process computer

- LOG OPERATING DATA IN ENGINEERING UNITS
- CALCULATE AND DISPLAY OPERATOR GUIDES
- INTEGRATION OF MATERIAL FLOW
- REPORT ON PROCESS STATISTICS - MATERIAL USED FUEL USAGE, THROUGHPUT, ETC.
- CALCULATE AND DISPLAY OR RECORD UNMEASUREABLE VARIABLES SUCH AS BTU RATE, MASS FLOW
- MONITOR AND ALARM PROCESS LIMITS
- RECORD PROCESS EVENTS DURING UNUSUAL DISTURBANCES
- MONITOR AND RECORD CHANGES IN SET POINTS, ALARM LIMITS, ETC. MADE BY THE OPERATOR
- PROVIDE ON DEMAND OPERATOR INFORMATION SUCH AS TREND RECORDING, ALARM STATUS REPORT, LOOP SET POINT AND PARAMETER DATA

Direct digital control was under consideration at the same time that the general purpose digital computer was performing process analysis, monitoring and some set point control.[1] It was reasoned that DDC would reduce the cost of a process control computer by eliminating the cost of the individual feedback controllers. Since the controller merely performs a calculation, why couldn't the computer perform the calculation? Several experimental ventures showed that the DDC concept was physically possible.[2,3] The feedback control law was calculated within a general purpose computer and the resulting signal outputted directly to the control valve.

At first, it appeared that the trade-off between individual loop controllers and a direct digital control (DDC) computer was in the area of 200 loops. There was a hooker, however.[3] This trade-off did not include any provisions in case the computer system failed. For most installations this meant using analog controllers to back up the DDC computer on each loop considered critical.

The DDC equipment was designed so that, if the computer failed, each valve would remain in its last directed position unless backed up by analog control. Critical loops were backed by an analog controller which would maintain loop control on computer failure. Control valves of the other DDC loops were "locked in" at their last output, but the operator could manually position each valve from a console on which he could read valve position and process measurement.

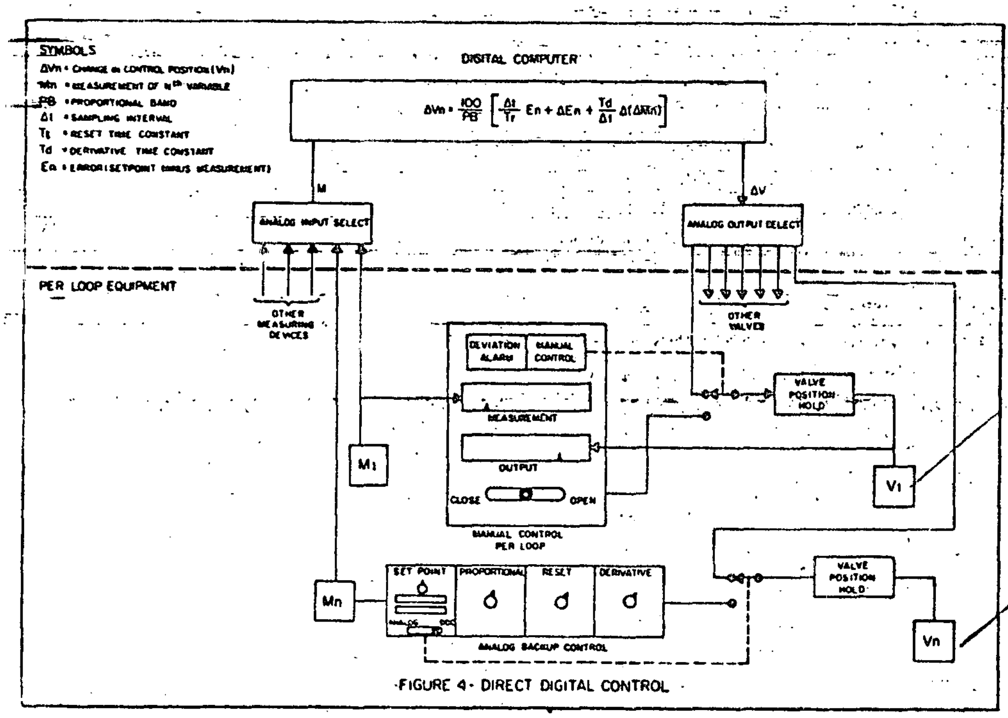
1 loop.

Figure 4 shows two loops from a large system. The measurement M_1, fed to the manual control panel, enables the operator to manually operate valve in case of computer failure. The loop containing measurement M_n and valve V_n has an analog controller for backup since measurement M_n is fast acting and cannot be controlled manually by the operator.

With the evolutionary history of digital process computer equipment, it is impossible to more than estimate mean time between failures (MTBF). For the smaller digital computers, including input/output equipment, that have been applied to the process control problems, calculated MTBF has ranged from 1000 to 2000 hours. Advances in circuit design indicate that reliability will increase, but reliability statistics on integrated circuits are not yet available. However, regardless of the projections and the calculated claims, the time-shared single computer system will never be perfect and will sometimes fail. Therefore, control security must always be considered on any process installation contemplating a digital computer.

Nu
fu
bal

For continuous processes, involving less than 150 loops, it appears that the single computer with set point analog control or DDC with analog backup and some pure DDC on noncritical loops makes the most sense. However, the user must be fully aware that he will give up economic and process control optimization, as well as the functions listed in Table II, if the computer fails. Perhaps most important,



Manual backup

Automatic backup

Figure 4—Direct digital control

On line digital backup

any advanced control that was dependent upon the computer, such as feedforward, cascade and multivariable, will be lost during computer shutdown.

For the installation where control is not continuous, but where control sequencing is imperative, the use of computer-set analog controllers is not sufficient. The computer provides sequencing and logic analysis which must have backup, if process operation is to be assured. The process control problem is not solved by keeping all control settings stationary upon computer system failure. In a chemical reactor, for instance, the contents can solidify or the reaction can "run away," if the process set point is not changed at the proper time.

A parallel DDC computer system

Figure 5 illustrates a parallel DDC computer system which not only provides computer backup but "backs up" the time-shared analog and digital input/output equipment which connects the computer to the various measurement and control elements. It also backs up all interloop controls, as well as all sequence control action.

In addition, this system can continue to perform the noncontrol functions such as those listed in Table II. It therefore permits control to continue even if one computer and/or its time-shared I/O equipment should fail. Note that if any of the time-shared equipment fails, process control is transferred to the backup subsystem.

Table III shows some interesting statistical data[4] which compare the availability of a single computer system with a parallel computer system. The table assumes that the MTBF of a single computer system is the same for each computer subsystem of the parallel computer system. Experience has shown that repair time for various failures, with on-site maintenance personnel, averages between 5 and 8 hours, depending upon the skill of the maintenance personnel, the availability of spare equipment, etc. With the parallel system, it appears that the average repair time can be maintained under 5 hours, since the system incorporates elaborate programs for self-diagnosis to ensure proper transfer to the backup

Table III – Availability – single computer vs. dual computer system

	AVERAGE REPAIR TIME	MEAN TIME BETWEEN FAILURES					
		1000		2000		3000	
		AVAIL(%)	OFF[1]	AVAIL(%)	OFF[1]	AVAIL(%)	OFF[1]
SINGLE COMPUTER SYSTEM	2 HOURS	99.8	17.49 HRS	99.9	8.76 HRS	99.93	5.78 HRS
	5 HOURS	99.5	35.75 HRS	99.75	21.01 HRS	99.83	14.06 HRS
	8 HOURS	99.2	69.55 HRS	99.6	34.95 HRS	99.74	23.04 HRS
DUAL COMPUTER SYSTEM[2]	2 HOURS	99.9998	63 SEC	99.9999	31.5 SEC	99.9999	31.5 SEC
	5 HOURS	99.9987	6.05 MIN	99.99968	1.60 MIN	99.9998	1.05 MIN
	8 HOURS	99.996	14.2 MIN	99.9992	4.2 MIN	99.9996	2.1 MIN

SINGLE COMPUTER FORMULA

AVAIL. = $\frac{\mu}{\mu+\lambda}$

μ = REPAIR RATE (REPAIRS/HR.)

λ = FAILURE RATE (FAILURES/1000 HRS.)

DUAL COMPUTER FORMULA

AVAIL. = $\frac{2\mu^2+\mu\lambda}{2\mu^2+2\mu\lambda+\lambda^2}$

[1] TIME IN A ONE YEAR PERIOD THAT THE SYSTEM DOES NOT PROVIDE COMPUTER FUNCTIONS.

[2] ASSUMING REPAIR HAS BEEN STARTED ON FAULTY COMPUTER OF DUAL SYSTEM BEFORE COMPLETE SYSTEM FAILURE

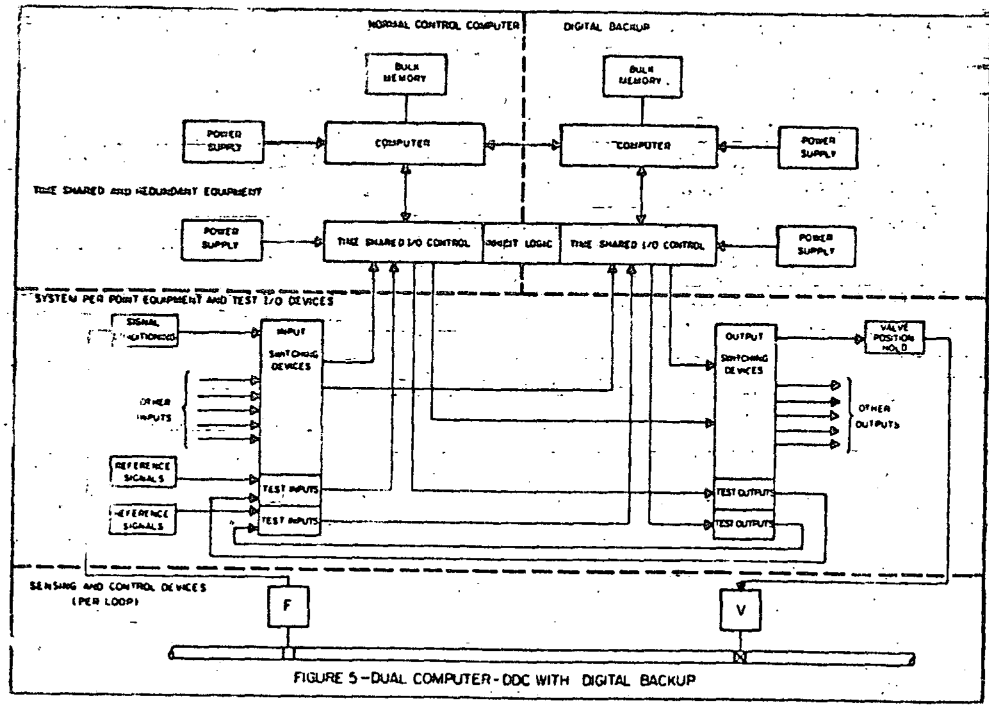


Figure 5 – Dual computer – DDC with digital backup

system. The failed computer subsystem is available for self-checking while the backup subsystem maintains process control.

Systems of this type can be economically attractive since they provide not only the essential control, but the system security essential to batch or start-stop operations. A parallel control processor using direct digital control techniques takes full advantage of the digital computer's process control capability without reservation and compromise. It can include advanced control techniques, such as self-tuning or adaptive control which cannot be obtained with set point control. The parallel computer processing system may provide these features and, in addition, may offer cost advantages over a conventional analog control system for the large continuous process.

For the continuous process in Table I, the computer contains the equivalent of 272 analog controllers. Implementation of a system of this size with DDC and analog backup could exceed the cost of implementation with the parallel or redundant computer scheme.

Input/output equipment

Figures 4 and 5 show that in DDC, as in all control systems, measuring elements and final control devices are still essential. Each measurement is individually conditioned before being fed to the multiplexer of the computer input/output system. Failure of any input or output therefore is similar to failure of a single controller and will not disable other loops. The system should be designed so that failure of any circuit element will not cause the loss of any common power supplies. Also, in case of a power failure, there must be battery backup or a redundant power supply.

Other cautions must be observed in the design of the parallel system interface equipment:

The system must be able to identify and diagnose the fault of any time-shared input/output equipment without disrupting control. The normal control computer and the backup system should both contain several inputs and outputs which can be used for automatic on-line testing of I/O operation, regardless of which subsystem is controlling the process. Some of these test inputs are connected to reference signals, others are connected to output test signals, closing the test loops through each subsystem.

All failed devices must be easily removed for replacement. Any disruption of normal functions during repair should be limited to the few inputs or outputs which share the same printed circuit as the failed element.

There also should be a diagnostic program which verifies correct operation, after the failed component has been replaced.

Output devices, for valve positioning or on-off control, which require power to maintain their status and/or output signal, should have at least a [illegible] battery backup system, in case of system AC loss.

The system must detect the failure of any shared element in the input/output system [illegible] logic and automatically switch to digital backup. While operation is in the backup mode, the failed control logic must be electrically isolated and inhibited from operating input and output control devices. Repair can then proceed with no fear of accidental interference with process control.

In normal operation, with the control computer in command, the backup system must continually check its input/output operations to ensure that backup is available.

The inhibit logic must be fail-safe so that its failure will not disturb the system in control. It must be tested automatically to ensure that transfer to backup can take place if a transfer is commanded by a failure detection. If inhibit logic will not transfer the other computer automatically, the system should annunciate that fact and provide an independent manual override which forces transfer of the control of the input/output equipment to the other computer.

Other system design requirements

The system must have a computer-to-computer communication link which continually updates the backup program data and status on a periodic fixed time basis. The backup computer thus receives dynamic operating conditions within a short time period (in the order of seconds for a batch process).

Any program changes made on-line while the control computer is operating the process must be transferred to the backup control computer, at the same time. This updating must include operator changes to control settings as well as any on-line program changes.

A bulk memory must be used on both computer systems to retain the many formulas and programs that may be required. Bulk memory can also contain interpretive programs to simplify construction of a batch program, diagnostic programs for fault detection and programs to aid maintenance. Sophisticated man-machine communication programs, which involve lengthy message storage, can also be included.

Diagnostic programs for the computer-to-computer communications link should test for link failure, annunciate the failure and command the changeover to the backup system. A program system permits updating and on-line diagnostics while time-sharing the real-time programs in bulk memory.

There should be a system procedure and a system diagnostic program to assist in rapid repair of a failed subsystem. Another procedure and program is required to transfer all operating programs from the backup subsystem back to the repaired computer, without interfering with process control.

When the backup system is not on control, it is available for program compiling, debugging and problem simulation using the test inputs and outputs. It must also perform diagnostics to ensure operation is correct for takeover if necessary. When backup computer takes over process control, these programs are discontinued.

CONCLUSION

By using DDC with complete input/output control and computer backup, the parallel computer processing system permits unrestricted application of computer control techniques. It takes full advantage of the logic and computational ability of the digital computer, whereas a computer system which depends on analog set point control or analog backup cannot.

The parallel control computer system program storage ability, together with backup of logic control, program sequence and formulation, makes it ideally suited for complex batch or start-up and shutdown applications.

Complex continuous control systems would also benefit with this control system. Built with state-of-the-art electronics, the system should challenge the economics of computer set point control and single computer direct digital control with analog backup.

REFERENCES

1 J W BERNARD J F CASHEN
Direct digital control
Instruments and Control Systems Sept 1965

2 E VANDER SHRAFF W I STRAUSS
Direct digital control - an emerging technology
Oil and Gas Journal November 16 1964

3 J W BERNARD J S WUJKOWSKI
DDC experience in a chemical process
ISA Journal December 1965

4 R H MYERS K L WONG H M GORDY
Reliability engineering for electronic systems
John Wiley and Sons New York 1964

Systems Engineering in the Glass Industry

RAYMOND J. MOULY, SENIOR MEMBER, IEEE

Abstract—A survey of current trends of systems engineering in the glass industry is presented. The central theme is that systems engineering is the technique through which the process of our time—the information revolution exemplified by the digital computer—is exerting its impact on the industry.

Systems engineering is examined and basic concepts reviewed, and the production system is defined as a pyramidal, hierarchical structure. Process models which have been developed primarily for control purposes are reviewed; examples of theoretically or experimentally developed models are given. In computer control applications, a major trend is seen toward extensive integrated real-time information-processing systems consisting of several computers connected through a communication network. The development of the human components in the production system, particularly management structure, is considered as an essential aspect of the overall system development.

I. Introduction

About 200 years ago, the invention of the steam engine marked the beginning of the first industrial revolution. The mechanical age had begun, characterized by, in the words of McLuhan [1], "the technique of fragmentation that is the essence of machine technology," with its emphasis on the individual control of the fragmented parts without marked concern for their interaction and the behavior of the process as a whole.

The mechanical age is now receding. We are living in the "electric age." The information revolution—the process of our time—is taking place, forcing us to reshape and restructure our processes and to move inexorably from fragmented, slow, and informal control practices to a philosophy of global, instantaneous, and systematic control.

These statements provide the background for the survey that follows. It consists of three major parts. First, in Section II, some fundamental systems engineering concepts will be reviewed. Then, in Section III examples of the application of these concepts in the glass industry will be presented. Finally, in Section IV the role of human factors in systems engineering will be discussed in a general way.

II. General Systems Engineering Concepts

A. *Definitions*

What do the terms "systems and systems engineering" mean? There are almost as many definitions as there are writers on the subject. The concept of systems is an ancient one. An early reference can be found in this quotation from St. Paul: "There are many members, yet but one body." A modern definition [2] reads as follows: "A system is any collection of interacting elements that operate to achieve a common goal." Systems engineering is the art or the technique of building systems. This, in itself, would not be a new activity were it not for two factors which characterize systems engineering and set it apart from conventional engineering. The first factor is the formal awareness of the importance of interaction between the parts of a system. The second factor is that systems engineering implies integration. It says that the whole is more than the sum of the parts.

Syst.
Eng.
Interfac
Integra

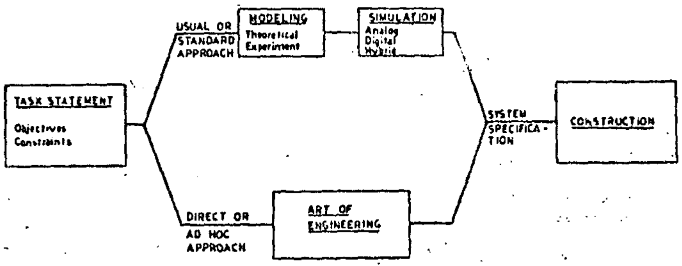


Fig. 1. Physical system design approaches.

Designing a system consists of translating a task statement into a specification of the system to be built. There are two fundamentally different approaches to the system design problem. They are, as defined by Athans [3], the direct or *ad hoc* approach and the usual or standard approach (Fig. 1).

The direct approach is often referred to as the art of engineering. It consists simply of building a system which does the job. The direct approach is acceptable for small systems, but as systems become increasingly complicated and extensive, it is frequently inadequate if optimum design is to be achieved. In addition, the risk and costs involved in extensive experimentation might be prohibitive.

The usual or standard approach is the technical or scientific approach; it begins with the replacement of the real world problem by a problem involving mathematical relationships. In other words, the first step consists of formulating a suitable model of the physical process, the system objectives, and the imposed constraints. Simulations of mathematical relationships on a computer often play a vital role in the search for a solution. Various alternative designs can be compared and evaluated. Then, and then only, a system is built.

Practically the design of a large and complex system is often achieved through the combined use of the direct and the standard approaches. The direct approach is likely to be used in the structuring of the whole system, whereas the standard approach will be taken for the design of

---

Manuscript received December 21, 1968. This paper was presented at the 8th International Congress on Glass, London, England, July 5, 1968.

The author is with the Technical Staffs Division, Corning Glass Works, Corning, N.Y.

Reprinted from *IEEE Trans. Syst. Sci. Cybern.*, vol. SSC-5, pp. 300-312, Oct. 1969.

various components. The standard approach has been extensively used by engineers for the design of control systems.

The manufacturing process is the system we are interested in. I shall discuss its nature from a systems engineering viewpoint and particularly examine the role of the information network and show how it relates to the economics of process control.

B. Hierarchical Process Control [4], [5]

2 elemnts: physic sys for control

The manufacturing system, whether it be a major process, a plant, a multiplant operation, a company, or even a whole industry, can be looked at as the pyramidal structure shown in Fig. 2, consisting of two distinct elements: the physical process and the controller. The controller's function is to manipulate the plant in order to optimize the process with respect to the manufacturing system objectives.

Somewhat arbitrarily, a hierarchy of three interacting control functions can be identified. At the first level, we find the process control functions which include the single- and multiple-variable control activities usually associated with the control of process units. Production control, at the second level, is the guidance for the utilization of production facilities; it covers such activities as scheduling, inventory control, cost control, and invoicing. The management control functions at the third level include the setting of objectives to be achieved by the system within the constraints of policy.

Funct reg optim adap self org

Paralleling the hierarchy of control levels, we can identify a hierarchy of control functions—regulation, optimization, adaptation, and self-organization—as we move toward the top of the pyramid. It can also be observed that, as we advance toward the higher levels of control, the emphasis on the physical variables decreases as the economic variables play an increasingly important role in the decision-making or control functions.

Other important characteristics of the control system are the decreasing frequency of the controller action and the increasing complexity of the decision-making process as one rises through the hierarchy of control levels. It should also be pointed out that control problems at the lowest level are essentially those of a deterministic system, whereas as one rises through the hierarchy, the nature of the problems becomes increasingly probabilistic.

This hierarchical control structure can be identified in most industrial processes although not always in a systematic form. We find that machines, such as controllers, sequential control systems, etc., are carrying out automatically some of the control functions at the lowest level of control, but that most of the control functions are still exerted directly by human beings (process operators, supervisors, schedulers, and managers). All of these controllers, human beings or machines, have one common characteristic: they are processors of information and are part of the information network of the system.

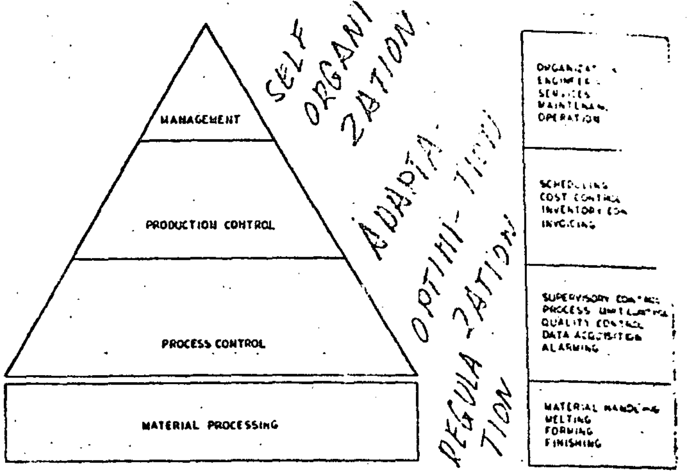


Fig. 2. Plant functions.

The importance of the information network within the manufacturing process cannot be overemphasized. It is the interconnecting tissue which relates the other five process networks: materials, orders, money, personnel, and capital equipment [6].

Efforts to automate process control functions took place initially at the first level of control with the application of process controllers. Little could be done at the higher levels until 20 years ago, when the invention of the digital computer marked the beginning of a new era. This second industrial revolution, the information revolution, which has already deeply affected our concepts of process control, has developed along two somewhat distinct paths. On the one hand, with the availability of data-processing machines, attempts have been made to automate part of the control functions at the third level. On the other hand, during the past 10 years, computers have increasingly penetrated the industrial process production control field at the first and second levels.

Today, the availability of reliable on-line process control computers makes it possible to affect in real time the entire information network of the production process and to implement integrated systems that will perform control functions at all levels of the hierarchy. Such systems are technologically feasible. Why should they be implemented? Technological feasibility is not enough. Powerful economic incentives must exist if the technique is to be applied extensively by competitive industries. In order to answer the question, we should examine the nature of the relationship that exists between the processing of control information and the economics of the process.

C. Process Control and Process Economics

We know, intuitively, that there is a relationship between these two subjects, but it is only recently, however that the quantitative nature of this relationship has been established. Trapeznikov shows in a recent paper [7] that controlling a process consists in ordering information. Any process or system left to itself under natural conditions will tend to become increasingly disorderly; the

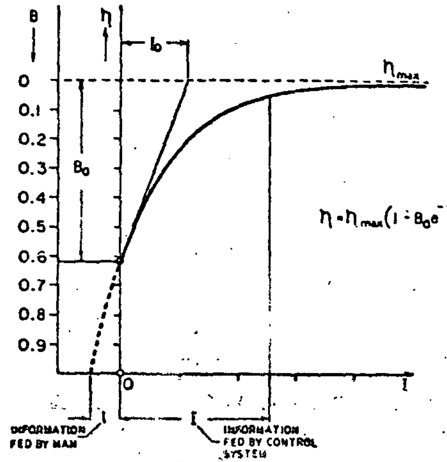


Fig. 3. Process effectiveness—control information curve.

entropy of the system will increase. The purpose of controlling the process is to counteract the growth of disordering. Control is work for ordering.

A fundamental relationship relates the system efficiency to the amount of control information I

$$\eta = \eta_{max} (1 - B_0 e^{-(I/I_0)})$$

B_0 being the measure of the degree of disorder in the system associated with the amount of control information I_0.

Efficiency should be taken here in a very general sense, and in particular, it can be looked at as profit. The relationship, illustrated in Fig. 3, can be looked at as a formal expression of the "law of diminishing returns" or of the "cost–effectiveness" relationship applied to control systems. It is quite similar to the familiar S-shaped relationship between return and effort expressed in monetary units.

Important practical conclusions can be drawn from these considerations:

1) Process effectiveness increases rapidly at first with increasing knowledge, but because of the basic nonlinearity of the relationship, the investment in control should not exceed a certain economically justifiable level.

2) In order to achieve the maximum overall effectiveness, it is necessary to attain the same degree of effectiveness at all levels.

3) So far, the automatic control of information at the higher levels has received little attention as, traditionally, the major function of instrumentation and control engineering has been to increase the ordering of information at the process control level, the first level of the control hierarchy. The automatic coordinated control of major units has not progressed as rapidly, basically because until recently no control tools were available to process reliably control information in real time. It should, consequently, be expected that the economic potential of automatic process control at the higher levels would be high because of the inherent, high information disorder usually found at these levels of control.

III. Systems Engineering in the Glass Industry

I shall now review specific examples of applications of systems engineering concepts in the glass industry. I shall focus on two subjects—process modeling and computer control systems.

A. Process Models and Modeling Techniques

The plant or process is the central and most fundamental issue. In process control, knowledge of process behavior comes first. Models which represent the essential aspects of the process are needed in order to apply the standard approach to systems design.

A model is defined as "a quantitative or qualitative representation of a process or endeavor that shows the effects of those factors which are significant for the purpose being considered" [8]. We shall not consider either physical scale models, such as tank models using viscous solutions [9]–[11], or activity models, such as PERT, but will discuss only models in which mathematics is used to describe the salient features of the process behavior and which are intended primarily for use in the synthesis of control systems. The mathematical relationships of interest are those which relate the process inputs, manipulated variables, and disturbances to the intermediate variables and outputs (Fig. 4). It is essential for process control problem applications that these relationships account for the dynamic behavior of the system.

Def
un mo-
dele

Models can be classified as experimental or theoretical according to the techniques through which they are developed. Experimental modeling [12] requires the observation of the process variables in order that the state of the process may be recorded under a variety of conditions. Intentional perturbation of the process through the manipulated variables and inputs is usually necessary to obtain accurate relationships. The trend is toward the increasing use of automatic data acquisition and processing techniques to determine the quantitative relationships that exist between the process variables.

pert

In theoretical modeling, the mathematical description of the process is built by writing the exact equations which govern the behavior of the process, such as conservation of mass, energy, and momentum, and the fundamental equations of heat transfer and fluid flow.

In any case, the validity and usefulness of the model generally depend heavily upon the ingenuity of the model builder, his clear understanding of the purpose of the model and his prior knowledge of the process.

Several examples of experimental and theoretical models developed for the design of control systems in the glass industry will be reviewed in the following.

Exp

1) *Vello Tubing Process Model* [13]: This first example is one of an experimental model. The problem is to develop an automatic diameter control system for a tube-drawing process used in the manufacture of fluorescent tubing.

The process is shown in Fig. 5. Glass is delivered to the forming process through a refractory ring placed at the

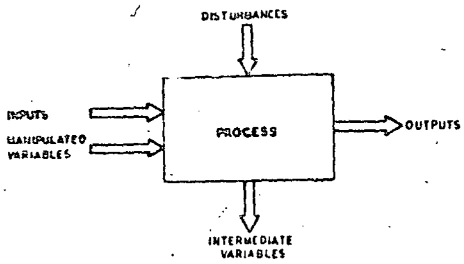


Fig. 4. Basic process.

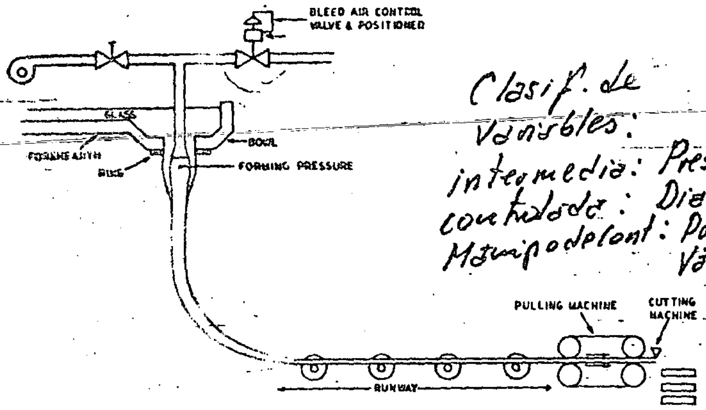


Fig. 5. Vello tubing process.

Clasif. de Variables:
intermedia: Pres.
controlada: Diam
Manipulada: Posición Válvula

bottom of the bowl. Air is blown through a pipe in the center of the ring while the tubing is drawn by a pulling machine. At the end of the runway, a cutting machine cuts the tubing into tubes of proper length.

The experimental mathematical model used to describe this process consists of two parts. The first part is a set of linear, incremental, differential equations expressing the relationships between the manipulated variable, valve position, the intermediate process variable, forming pressure, and the controlled variable diameter. The equations given below were obtained by experimental step-response techniques.

STEP RESP

interm.

$$\frac{\Delta \text{ forming pressure}}{\Delta \text{ valve position}} = \frac{T_1s + 1}{\alpha T_1s + 1} \frac{K_1}{T_2{}^2s^2 + 2\zeta T_2s + 1}$$

manip

contr.

$$\frac{\Delta \text{ diameter}}{\Delta \text{ forming pressure}} = K_2e^{-Ls}$$

Del. de la fec. del dist.

The second part of the model is the statistical description of the controlled variable. This description is in the form of power spectra and histograms. The power spectra, Fig. 6, characterize the way the diameter variations occur. Significant diameter variations still take place at the process natural frequency, 1.6 cycles/min. Consequently, an effective automatic control system must control diameter variations occurring up to this frequency. This information on the statistical behavior of the process provides a basis for the simulation of the process disturbances and a means for estimating the expected improvement in process performance that would result from the implementation of a given control system.

Simul.

Base para la SIM.

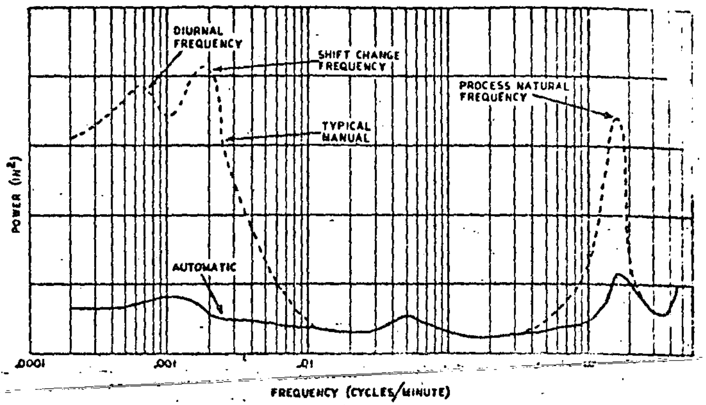


Fig. 6. Power spectra—manual and automatic control of tubing diameter.

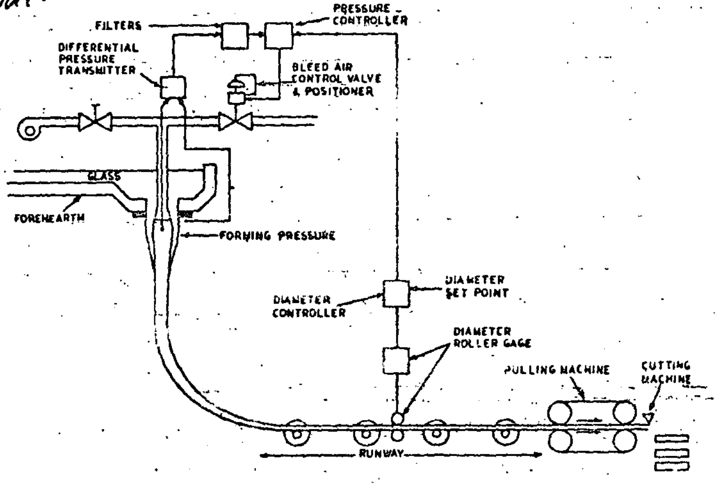
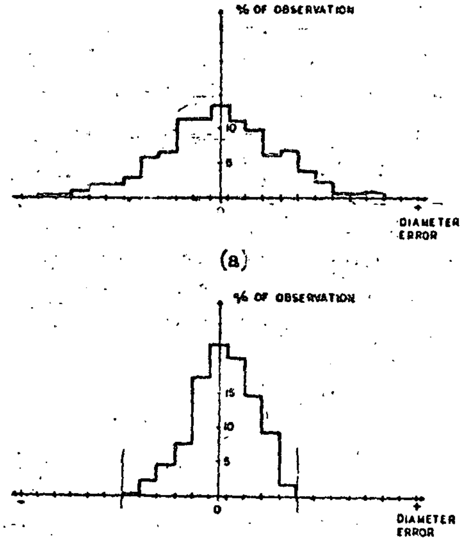


Fig. 7. Vello tubing process with automatic diameter control system.



(a)

(b)

Fig. 8. Histograms of diameter error (a) Manual. (b) Automatic diameter control. (Note: σ automatic = 0.5 σ manual.)

Control de Cascada

These process models, incremental differential equations, and statistical models were used in an analog computer simulation to evaluate a number of possible control system configurations. The control system of Fig. 7 was selected; it is a cascade control system in which the forming pressure is controlled by a high-gain, large-bandwidth loop, and the diameter is controlled by a low-gain, low-bandwidth loop.

The histograms, Fig. 8, characterize the performance of the system under manual and automatic control. It is seen that the automatic control system reduces the diameter variations by 50 percent.

2) *Ribbon Machine Process Model* [14]: Two models were developed in connection with the design of a computer system for the automatic control of the dimensions of bulbs made on a ribbon machine (Fig. 9). These models which account for the process behavior, including the quality control sampling procedures, were used in a digital computer simulation to evaluate alternate control strategies.

The first model is the matrix in Fig. 10. It was determined experimentally and represents the relationships that exist between the most significant process variables.

The second model was developed also for the analysis of the particular control problems resulting from the fact that only small, relatively infrequent samples of the end product quality can be obtained for feedback control. This problem was investigated in a digital computer simulation study of a one-variable control loop with quality data used as the feedback measurement and a long transport delay as the significant process dynamic element. The process disturbances were simulated by the sum of an assignable periodic disturbance and a random disturbance. This study indicated that the sample mean was the best indicator of average process performance and that the stability of the system in response to the assignable disturbance depended only upon the control system design parameters.

The computer process control system schematized in Fig. 11 was developed on the basis of these studies. The automatic control of the low-frequency components of the error signal resulted in a significant reduction of the variability of the product dimensions.

3) *Glass Tank Model.* Another example of experimental modeling is given by Hoetink in his investigation of the dynamics of a glass tank [15]. When the composition of the bath in a continuous furnace is changed abruptly, there will follow a change in the glass composition at the output of the tank. Comparing this change in glass composition to the step change in batch, the transfer function for composition of the melting furnace may be determined; two cases are considered with and without cullet return.

In general, the transfer function may be approximated by a transportation lag T_L and a first-order process with time constant τ. If all of the glass in the furnace were well mixed, the transfer function would have only one time constant T_m, which is also the mean residence time

Modelo: Transp. Lag + First order

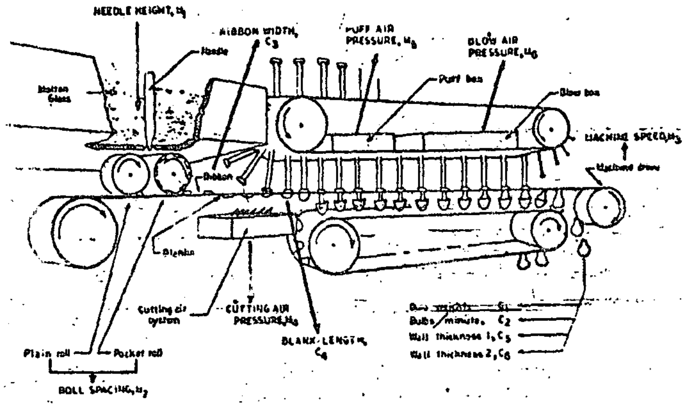


Fig. 9. Ribbon machine process.

$$\begin{array}{cccc}
\text{PROCESS OUTPUTS} & & \text{PROCESS MATRIX} & \text{MANIPULATED VARIABLES} \\
\begin{bmatrix} C_1 \\ C_2 \\ C_3 \\ C_4 \\ C_5 \\ C_6 \end{bmatrix} & = &
\begin{bmatrix}
\frac{a_{11}e^{-L_1 s}}{\tau_{11}s+1} & a_{12}e^{-L_1 s} & \frac{a_{13}e^{-L_1 s}}{\tau_{13}s+1} & 0 & 0 & 0 \\
0 & 0 & a_{23} & 0 & 0 & 0 \\
\frac{a_{31}e^{-L_2 s}}{\tau_{31}s+1} & a_{32} & \frac{a_{33}}{\tau_{33}s+1} & 0 & 0 & 0 \\
\frac{a_{41}e^{-L_2 s}}{\tau_{41}s+1} & a_{42} & a_{43} & a_{44} & a_{45} & 0 \\
0 & a_{52}e^{-L_1 s} & 0 & a_{54}e^{-L_1 s} & a_{55}e^{-L_1 s} & 0 \\
0 & a_{62}e^{-L_1 s} & 0 & a_{64}e^{-L_1 s} & a_{65}e^{-L_1 s} & a_{66}e^{-L_1 s}
\end{bmatrix} &
\begin{bmatrix} M_1 \\ M_2 \\ M_3 \\ M_4 \\ M_5 \\ M_6 \end{bmatrix}
\end{array}$$

Fig. 10. Ribbon machine process model.

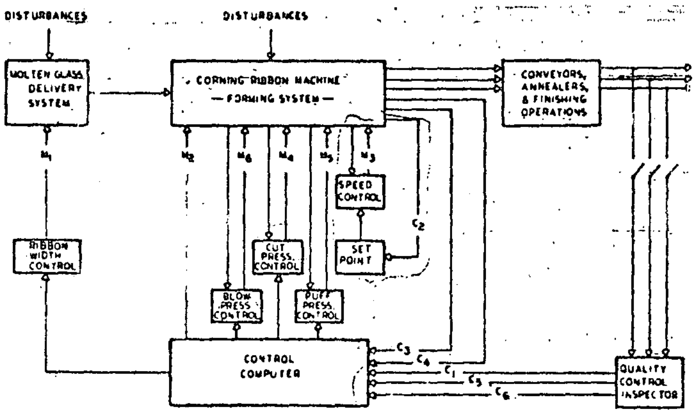


Fig. 11. Ribbon machine process computer control system.

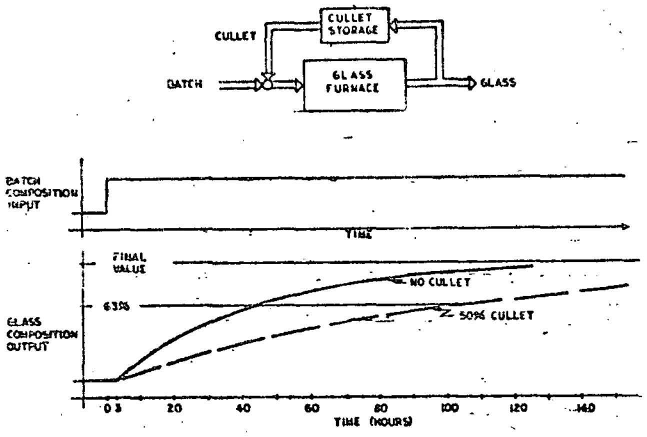


Fig. 12. Glass composition response to a step change in batch composition.

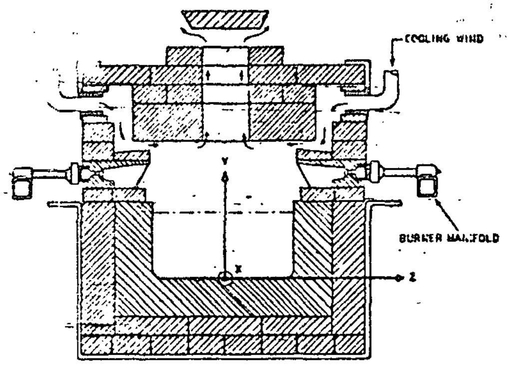


Fig. 13. Forehearth channel—cooling zone cross section.

of the glass in the furnace. The mean residence time can be estimated by dividing the furnace glass capacity M by the average glass output (pull) Q. Comparing T_L, τ, and T_{td} can give some idea as to what extent the glass is ideally mixed. The derivative of the step response gives the residence time distribution of the glass. Fig. 12 illustrates some experimental results.

Larga const. de tiempo

For a furnace with a glass capacity of 200 tons and a pull of 96 tons/day, $T_{td} = 50$ hours; the transfer function without cullet return consisted of a transportation lag $T_L = 3$ hours and a time constant $\tau = 40$ hours. With a cullet return of 50 percent after 20 hours, the transportation lag was 3 hours as before, but the time constant increased to 100 hours.

Theoret

4) *Forehearth Model:* The forehearth model developed by Duffin and Johnson [16] illustrates the methodology used to construct a theoretical model based on physical laws of nature. The development of a theoretical model usually involves the following steps: 1) formulate the system equations based on physical laws, 2) apply appropriate boundary and initial conditions, and 3) solve the equations by analytical or numerical means.

The forehearth delivers the glass in an open channel from the furnace to the forming machine and conditions the glass to a predetermined delivery temperature by means of wind cooling and gas heating as shown in Fig. 13.

a) *Formulation of system equations:* The basic ener equation—The general differential equation f transfer of a flowing stream of molten glass in a rect gular channel is derived based on the principle of conse tion of energy. By taking an energy balance on a diff ntial volume element of dimensions dx, dy, dz, the equation is

$$\underbrace{\frac{\partial}{\partial y}\left(k'\frac{\partial T}{\partial y}\right)+\frac{\partial}{\partial z}\left(k'\frac{\partial T}{\partial z}\right)}_{\text{rate of energy input by conduction and radiation}}-\underbrace{\frac{\partial}{\partial x}\left(\rho C_p V_x T\right)}_{\text{rate of energy input by mass flow}}$$

$$\underbrace{\rho C_p \frac{\partial T}{\partial t}}_{\text{rate of accumulation of energy}}.$$

In deriving (1), the following assumptions are made.

i) Heat flow by radiation can be regarded as being due to a "radiation conductivity" of $8T^3/\alpha$, where T is the absolute temperature and α is the absorption coefficient for the energy of wavelengths corresponding to temperature T. The factor k' in (1) is defined as the true conductivity plus radiation conductivity.

ii) The effective conductivity k', density of gla and the specific heat of glass C_p are not temperature dependent (hence not a function of the space coordinates).

iii) The velocity V_x in the x direction (direction of flow) is not a function of x. Thus (1) reduces to

$$\frac{k'}{\rho C_p}\left[\frac{\partial^2 T}{\partial y^2}+\frac{\partial^2 T}{\partial z^2}\right]-V_x\frac{\partial T}{\partial x}=\frac{\partial T}{\partial t}. \tag{2}$$

Equation (2) is applicable only in the interior of the glass. To completely specify the system, appropriate boundary and initial conditions must be supplied. These are the following.

i) The temperature distribution on the glass-refractory boundaries at the bottom ($y = 0$) and the sides ($z = W$) of the channel are assumed to be time-invariant and linear functions of the space coordinates.

$$\begin{aligned} T(x,0,z) &= \phi_2(x,f) \text{ is specified} \\ T(x,y,w) &= \phi_3(x,y) \text{ is specified.} \end{aligned} \tag{3}$$

ii) At the interface between the glass and the gas ($y = d$), the boundary is a radiating boundary where the glass is exchanging radiant energy with the channel enclosure (refractory crown). Further, the gas in the space between the glass and the crown also exchanges heat with the tem through convection and radiation. The equation r the glass–gas interface is again derived based on energy balance

$$k'\left.\frac{\partial T}{\partial y}\right|_{y=d} = \sigma F\,[T_{\text{crown}}{}^4 - T^4] - h(T - T_{\text{gas}}) \tag{4}$$

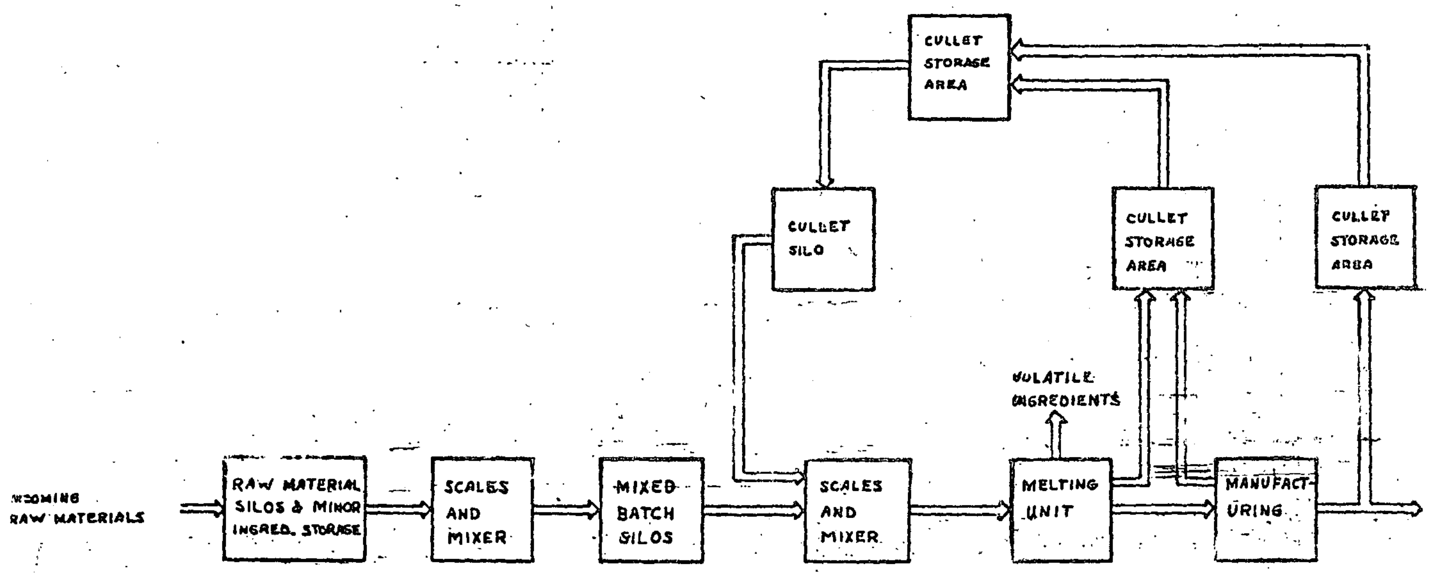


Fig. 14. Melting system schematic.

where

σ	Stefan Boltzman constant
F	view factor derived with the assumption that the glass surface and the crown are two opposite infinite parallel planes
h	gas heat transfer coefficient
T_{crown}, T_{gas}	these temperatures are inputs to the model and must be either assumed or determined by measurement on actual forehearths.

iii) Since glass temperature is symmetric with respect to the center of the channel ($z = 0$),

$$\left.\frac{\partial T}{\partial z}\right|_{z=0} = 0. \qquad (5)$$

iv) At time zero, the temperature distribution at some location X must be specified as an initial condition. Usually the temperature distribution at the inlet to the forehearth is given.

b) *Numerical solution:* Equations (2)–(5) and the appropriate initial conditions completely specify the system. The equations are partial differential equations of the parabolic type with nonlinear boundary conditions. Because of the complexity of the problem, an approximate numerical solution is the best that can be obtained. The equations are written in a finite-difference form and can be solved on a large, digital scientific computer.

c) *Application of the model:* This model is applicable to the systematic design of a temperature control system for an existing forehearth. Studies can be made with the model to evaluate control systems which will deliver glass at constant temperature to the forming machine in the face of disturbances in the inlet glass temperatures, ambient temperatures, and glass flow rate changes.

d) *Melting System Models:* One of the earliest examples of the application of modeling techniques to the analysis of control system problems in the glass industry is given by Oppelt [17]. His paper presents a conceptual elementary multivariate dynamic model of a glass tank and suggests improved control strategies using feedback and feedforward techniques.

Our last example of theoretical modeling concerns the melting system illustrated in Fig. 14, consisting of raw materials input and storage, batch mixing and storage, melting, cullet-recycle, and control systems. The study made by Sting [18] is important in that it develops models for process units, such as storage silos, mixers, etc., and demonstrates the use of these models in the analysis of systems design and operation through simulation.

The first step in approaching the problem is to construct mathematical models for all the process units by taking one of the most important aspects of the entire process into consideration: the physical transformation of granular material.

A general model is developed which, when specialized, can be used to model silos, mixers, and mixing tanks along with other process components. This general model will be described briefly for a silo.

A silo is defined as a temporary storage device whereby granular material is dumped into the top, stored, and at some later time removed from the bottom. The model was developed under the following reasoning.

a) The filled silo is divided into spaces of batch volume size (refer to Fig. 15).

b) Associated with each space is a corresponding batch and its describing constituent vector.

c) When a batch is removed from the bottom, all the batch constituent vectors above it move down one space.

d) When the material is either entered or extracted, it is done discretely in time.

e) Because of the mixing effect between adjacent batches, the output batch is some combination of any input batch.

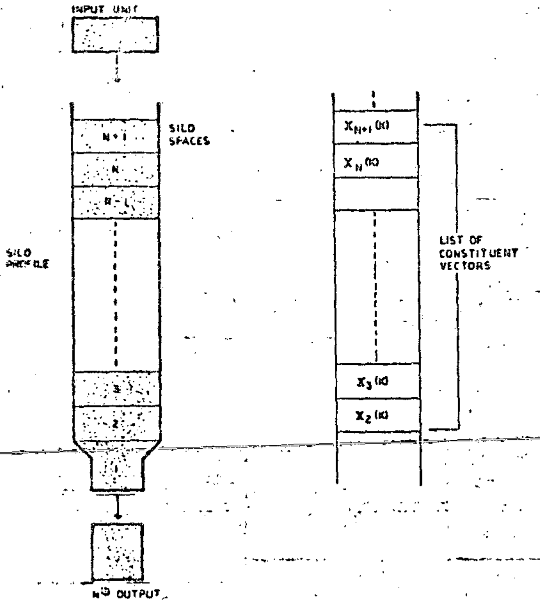


Fig. 15. Schematic silo.

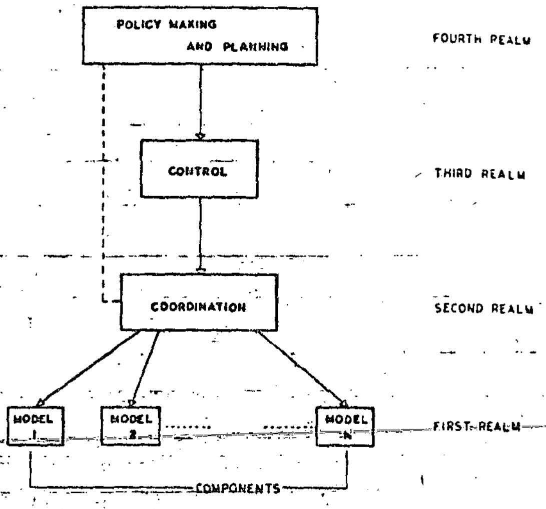


Fig. 16. Activity realms.

f) All materials which are placed in the silo together have equal or nearly equal densities.

g) A batch of materials, or any part thereof, has a maximum and a minimum length of the silo to transverse, and this transversal occurs within some maximum and minimum number of output batches.

These assumptions, together with mass and impulse balance, yield the following set of equations:

$$Y(K) = \sum_{i=1}^{m} W_i(K) - X_i(K) \tag{6}$$

$$\sum_{i=1}^{m} W\ (K) = 1 \tag{7}$$

$$\sum_{i=1}^{m} W_i(K - i + 1) - X_i(K - i + 1) = X_1(K) \tag{8}$$

$$X_1(K) = X_2(K - 1) = X_3(K - 2) = \cdots = X_m(K - m + 1). \tag{9}$$

By substituting (9) into (8), then rearranging it, there results

$$W_1(K) = 1 - \sum_{i=2}^{m} W_i(K - i + 1) \tag{10}$$

where

$X_i(K)$	constituent vector of the material at the *i*th position in the compartmentalized silo, just prior to the *k*th output
$Y(K)$	*K*th output batch constituent vector
m	maximum range over which an input batch will be spread over the output batch
$W(K)$	the weighing value which designates the percentage of inputs that are in the output at time *NT*.

The weighing values are assumed to be of a statistical nature. The particular disturbance associated with the random variables of the model is dependent upon the particular silo to be modeled and the material to be stored. Thus the weighing values not only must satisfy the constraints imposed by (7) and (10), but also must be generated in accordance with the information extracted from the actual data obtained by conducting experiments on particular silo. Once the weighing values are determined, (6) can be used to express the physical transformation taking place between input and output batches within the silo.

The second step is to combine all the component models into a "multiactivity system." Broadly defined, the model is composed of four activity realms (Fig. 16). The first realm defines the functions of the components of the process. The second realm defines the interactions and performs structure coordination. The third realm defines the supervisory functions (control), and the fourth realm defines the policy making and planning functions.

The complete system model for batch systems is amenable to digital computer simulation and has been used to investigate process design and control problems.

6) *Conclusions:* As is the case in other process industries, it appears that the lack of suitable process models still remains the major obstacle to the implementation of advanced control systems in the glass industry. As a rule, relatively unsophisticated control concepts are applied.

Although experimental techniques probably offer the best practical short-term approach to the problem of process modeling, theoretical modeling of process units offers very attractive long-term advantages, especially when the control system modeling can be combined with modeling for unit design. Although the cost of this approach is relatively higher and more time-consuming, the potential gains in the ability to synthesize optimally new process systems are very high.

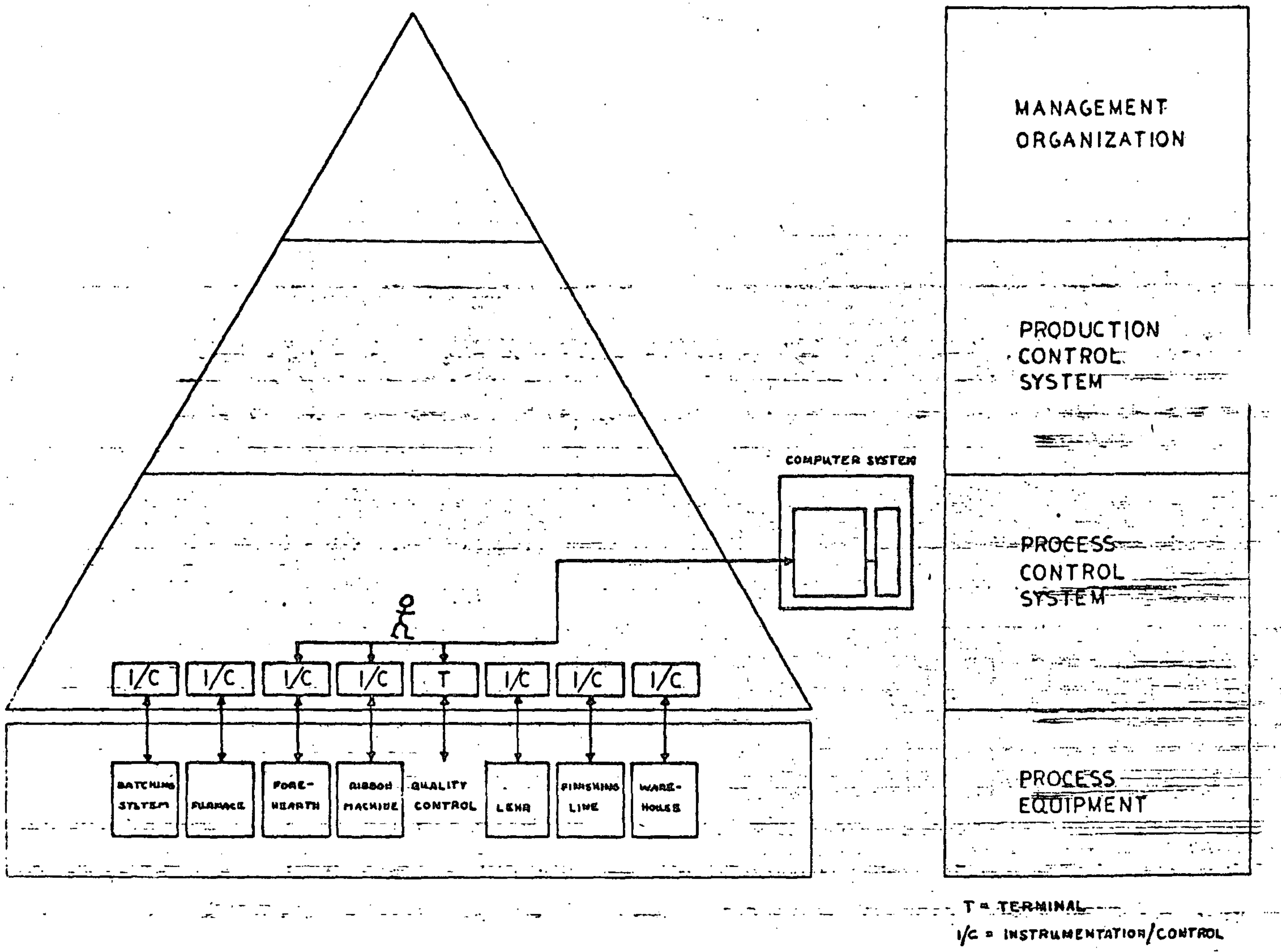


Fig. 17. Plant process control computer system.

Finally, much remains to be done in the area of modeling and control technology for large systems consisting of a number of process units. A particularly important problem is the incorporation in the model of the economic and information aspects of the process.

Computer Control Systems

The essential role played by the controller of the manufacturing process, the information network, was discussed in Section II-B. It was stated that the computer technology makes it now possible to automate control functions at all levels of the hierarchy. It is within this framework that we will now survey, on the basis of scarce published information, the status of the implementation of systems in the glass industry.

One of the first computer control systems implemented in the glass industry was mentioned in the section on modeling (Section III-A). It is the process computer control system developed for the automatic control of the ribbon machine [14]. This system performs control functions only. The structure of the system is depicted in Fig. 17. Quality control information is entered manually and processed by a process control computer which in turn manipulates a number of variables on the forehearth and ribbon machine.

Another example of process computer control application is given by the control system used in the plants of the Owens-Corning Fiberglas Corporation. On the basis of published information, it appears that these systems are essentially process control systems performing first level control functions in the melting and delivery areas of the process, although some production scheduling might be effected in some instances [19]-[21].

Other supervisory control applications have also been announced recently by glass container manufacturers [22], [23]. Computer control systems are being used for the control of batching, melting, and inspecting operations at the Lakeland, Fla., plant of Owens, Illinois. The function of the computer is to supervise and monitor the entire process.

Recent publications indicate significant trends in the process control area. The trend toward central control

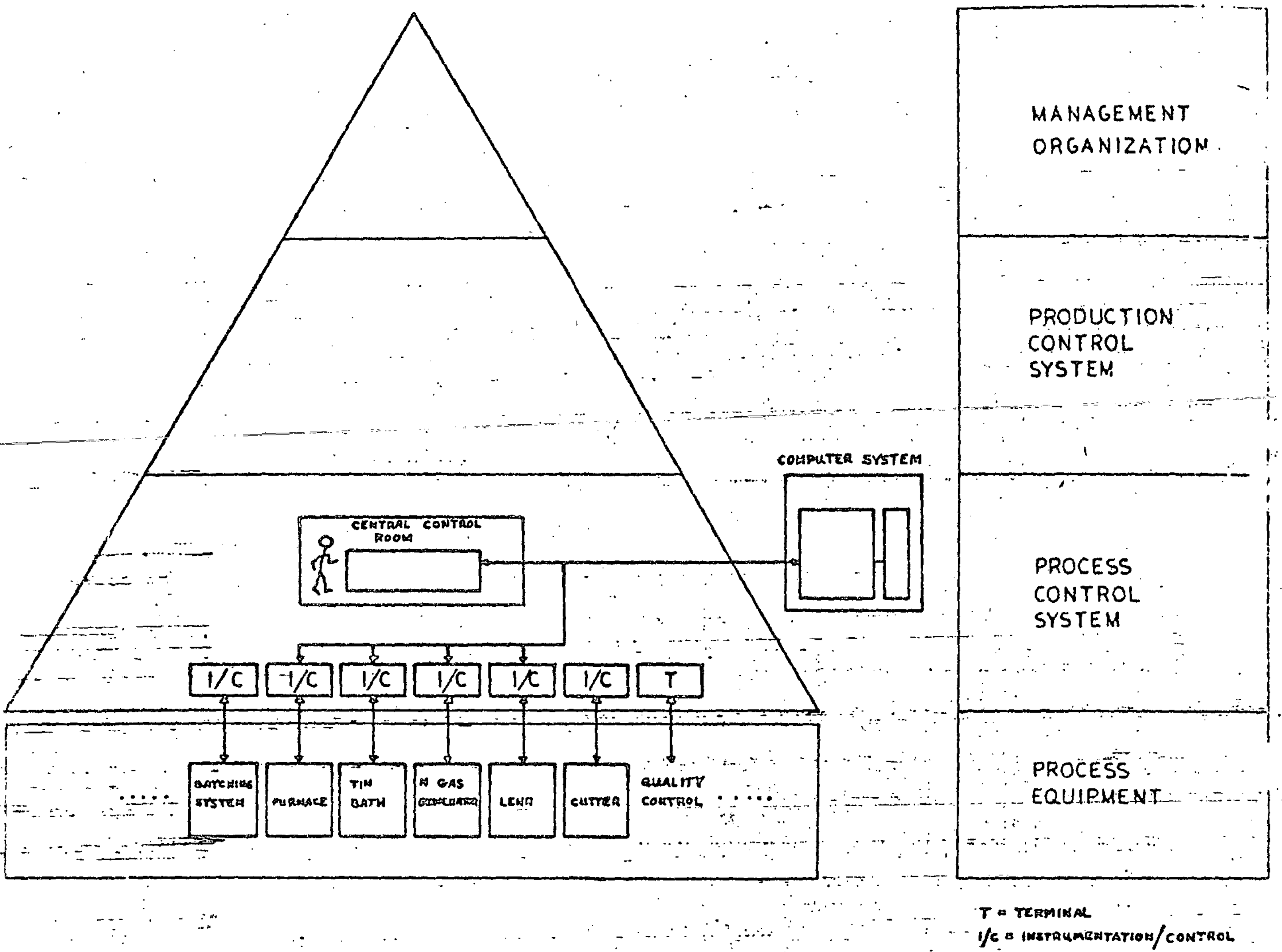


Fig. 18. Plant process control computer system with central control room.

rooms and centralized process control appears in the Ford Motor Company's process control computer system installed in Dearborn, Mich. The process computer control system controls a float glass manufacturing process [24], [25]. The system as illustrated in Fig. 18 handles approximately 80 closed control loops and monitors close to 500 process variables. The real-time, on-line control functions cover the melting furnace, tin bath, annealing lehrs, and gas generators. Monitoring of the batch house and the quality inspection is also effected. The system, which results in reduced manufacturing costs through improved quality and increased productivity, is also capable of handling background work such as generation of new programs, engineering calculations or nonprocess applications at the same time it controls the process. The central control room is represented in Fig. 19. The operator console, on-line printer, alarm typewriter, television display and recording devices, and graphic panels can be identified. The scarcity of recording instruments is apparent.

Leer

On the basis of these examples, it would appear that the glass industry, following the trend pioneered by other

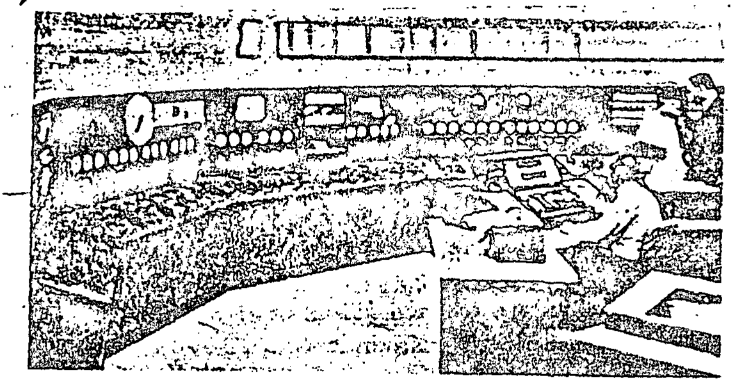


Fig. 19. Central control room—Ford Motor Company (Dearborn, Mich.).

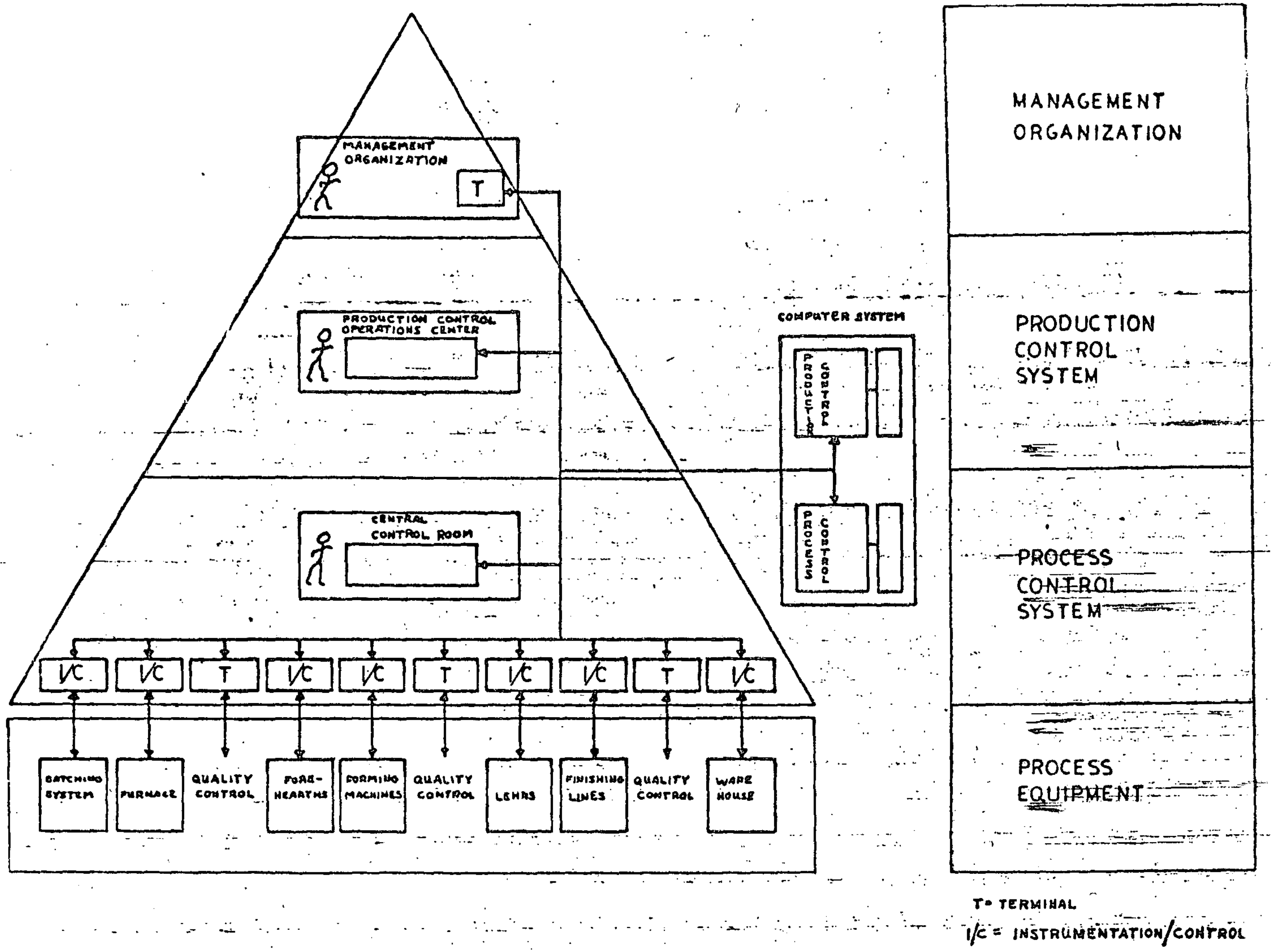


Fig. 20. Integrated plant control system.

process industries, is slowly moving, in an evolutionary fashion, toward computer-directed, central, process control systems.

It is believed that the trend toward integration will not stop at the process control level, but that production control and management control functions will progressively be included into the design of fully integrated on-line, real-time control systems. The diagram in Fig. 20 illustrates the structure of a possible integrated plant control computer system based on functional design. It is an integrated system because it performs both the process and production control functions on line and in real time. The information flow, data collection, and report generation are highly automated. The current status of the entire pla[illegible] available on a minute-to-minute basis. This perm[illegible] effective implementation of advanced managem[illegible]niques with decisions made on the basis of quanti[illegible] information available where and when needed. [illegible] no evidence that such integrated control systems [illegible]eration today although, as we mentioned pre[illegible]me of the existing control systems might al[illegible] developed to include some production control

The series of diagrams, the last one in particular, also suggests a clear trend toward making computing power available as a utility throughout the system in much the same way as electric power is available today.

The integrated control systems approach should naturally be expected to affect our basic concepts of plant design and operation. In particular, it should be expected to have a very significant impact on the management and organizational structure of the plant. This is the subject that will be discussed in the following section.

IV. HUMAN FACTORS [26]–[29]

The emphasis of this survey has been so far on the economic and technological aspects of systems development in the glass industry. We have discussed problems relating to the development of the automatic control loop represented by the diagram in Fig. 21, symbolizing the physical process controlled by an on-line computer. But manufacturing systems are man–machine systems, organizations whose components are men and machines, tied by a communications network, working together to achieve a common goal. Even in highly automatic computer control systems, the place of the human remains vital as Fig. 22

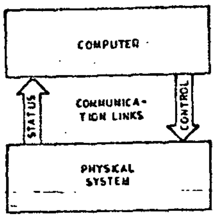


Fig. 21. The automatic control loop.

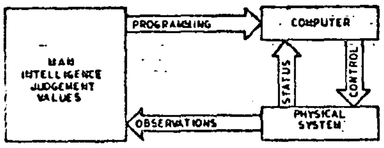


Fig. 22. The man loop.

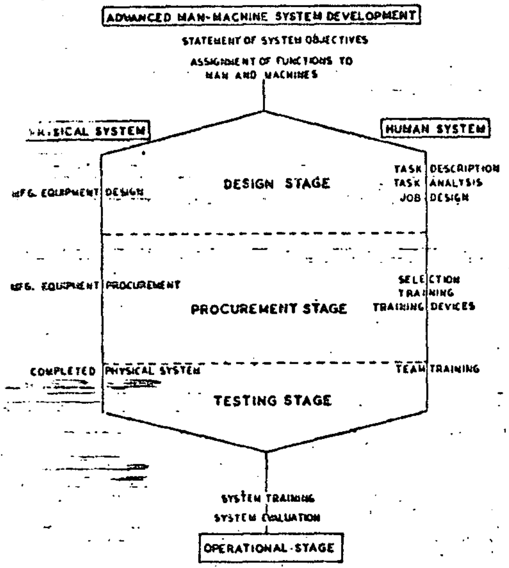


Fig. 23. Man–machine system development.

suggests. Man communicates with the system through programming, manual data entry, and instrumentation. He further observes the process to evaluate, through the use of his intelligence, judgment, and values, the performance of the automatic control loop in relation to his criteria of adequate or optimal system performance.

Of particular concern today to those involved in development of integrated control systems is a need for an awareness of man as a component in man–machine systems whose developmental needs resemble those of the machine or hardware components. Planning for the design and development of human components of systems has not been as systematically pursued in the past as it might have been. Characteristically, systems were designed and developed first, and assumptions were made that the human components required either existed or could be found or could be trained to operate this system. Only in relatively recent years, especially with the advent of extremely complex military and aerospace systems, has an increasing awareness developed of the need for systematic design and development of human system components. In the United States this is probably best reflected in the types of documents required of potential contractors for the development of complex man–machine systems. Qualitative and quantitative personnel requirements information (QQPRI) documents which specify the design and development of the personnel subsystem necessary for implementation, maintenance, and operation of these complex systems are required. This can no longer be an evolutionary development process. It must be planned and designed as the physical subsystem(s) is.

why developing in parallel!

Fig. 23 schematically represents the man–machine system development cycle. Advanced system development involves the initial statement of system objectives and culminates in decisions leading to the assignment of operational functions to men and machines. From this point on, the human and physical systems proceed on parallel courses of development to the point at which the completed components are assembled for testing and [illegible]ing in preparation for operation. Although not specified in Fig. 23, the parallel development of the two major subsystems does not in any way imply independent development. One of the primary values to be gained from such man–machine systems development approach lies in the repeated and ongoing interaction of the developers of the two subsystems at each point in the development cycle. In addition to assuming that all required components are available at a specified end point, the continuing interactions contribute immensely to preventing the need for costly and time-consuming retrofittings of components and major system modifications. To accomplish this, however, implies the development of an ability to communicate effectively and interrelate on the part of representatives of diverse disciplines. Compromises and trade-offs will be required. Ultimate optimization of each subsystem will undoubtedly not be possible, but total system optimization and effectiveness will be more closely approximated.

In light of what has been said about integrated process control possibilities in the future, what are some of the implications for human system components? The implications are numerous. Just a sampling would be the following: 1) Traditional organizational structures may be inappropriate for the management of integrated control complexes, either because they are too cumbersome or because their traditional control concepts are outmoded. 2) Routine, nonmotivating jobs may be eliminated entirely, resulting not only in a smaller, but in a more involved, committed, and motivated work force. 3) General technical and educational backgrounds of higher levels will be required, and programs and methods to prepare individuals for performance of the man functions in the system will have to be developed. 4) The relative status of various jobs, e.g., machine operators and maintenance employees may be modified with attendant needs for modifications of long-standing attitudes and opinions. 5) The traditional protection and security functions of labor organizations may no longer be required, leading to either a change of function or an elimination of the need for such functions entirely.

Whatever the end product of an integrated plant or company control system turns out to be, it is almost certain to require different approaches to the organization, management, development, and maintenance of the human components. What is implied in this paper is that planning for, and awareness of the need for, such an integrated approach to the human component development, along with the physical system development, must begin now if we are to achieve the higher levels of integrated control in the reasonably near future.

V. Conclusions

In this survey we have discussed some of the economic, technological, and human aspects of systems engineering. We see systems engineering as the technique through which the electric technology, exemplified by the digital computer, is being applied to our industry.

Several major trends that characterize the evolution of systems engineering technology in our industry have been identified:

1) There is a marked trend toward the increased integration of process control, production control, and management control functions.

2) Modeling techniques are playing an increasingly important role and should lead to the design of optimum systems through the integration of the design of the process and of its control system.

3) The importance of human factors cannot be overemphasized. Our understanding of these factors is one of the major elements, possibly the most important one, controlling the rate of implementation of modern technology in industry.

As engineers, we find ourselves increasingly moving in a position to influence directly social and human patterns. The nature of our work must change as our essential responsibility becomes one of education of the public in modern technology.

References

[1] M. McLuhan, *Understanding Media: The Extension of Man.* New York: McGraw-Hill, 1964.

[2] S. E. Elmaghraby, *The Design of Production Systems.* New York: Reinhold, 1966.

[3] M. Athans and P. L. Falb, *Optimal Control.* New York: McGraw-Hill, 1966.

[4] K. Chen, "Models for integrated control," presented at the 1965 Systems Engineering Conference, Chicago, Ill.

[5] I. Lefkowitz, "Multilevel approach to the design of complex control systems," presented at the 1965 Systems Engineering Conference, Chicago, Ill.

[6] J. W. Forrester, *Industrial Dynamics.* New York: Wiley, 1961.

[7] V. A. Trapeznikov, "Control, economy, technological progress," presented at the 1966 IFAC Cong., London, England.

[8] H. Chestnut, *Systems Engineering Tools.* New York: Wiley, 1965.

[9] I. M. Sheinkop and L. S. Belousova, "Modeling liquid for investigating glass movement in gas electron furnaces," *Steklo i Keramika,* vol. 23, pp. 23–25, February 1966.

[10] J. C. Hamilton, R. R. Rough, and W. B. Silverman, "Improved techniques for studying the design and operation of glass melting furnaces by means of models," *Advances in Glass Technology, Proc. 1962 Internatl. Cong. on Glass* (Washington, D.C.), pp. 190–195.

[11] J. D. McClelland, "Plastic flow model of hot pressing," *J. Am. Ceramic Soc.,* vol. 44, p. 326, 1961.

[12] P. Eykhoff, P. M. E. M. van der Gruten, H. Kwakernoak, and B. P. Valtman, presented at the 1966 IFAC Cong., London, England.

[13] R. J. Mouly and L. A. Zangari, "The development of an automatic diameter control system for glass drawing processes," *ISA Trans.,* vol. 3, pp. 158–164, April 1964.

[14] A. T. Bublitz, R. J. Mouly, and R. L. Thomas, "Statistical feedback squeezes product variations," *ISA J.,* pp. 55–60, November 1966.

[15] B. J. Hoetink, "Process dynamics of a glass furnace following a step change of one of the batch components," presented at the 1968 Internatl. Cong. on Glass, London, England.

[16] J. Duffin and K. Johnson, "Glass container process: forehearth simulation," IBM Corp., Systems Development Div., San Jose, Calif., Rept. 02-472-1, July 1965.

[17] W. Oppelt, "Regulating processes in furnaces and their representation with the help of block diagrams," *Glastechnische Berichte,* vol. 26, no. 5, pp. 146–150, 1953.

[18] D. Sting, "Granular batch process modeling," Case Institute of Technology, Cleveland, Ohio, Progress Rept. 19-9, October 1965.

[19] P. D. Griem, Jr., "Digital computers for glass process control," presented at the 1967 Am. Ceramic Soc. Meeting, New York.

[20] ——, "Direct digital control of a glass furnace," presented at the 20th Ann. ISA Conf., Los Angeles, Calif., 1965.

[21] R. R. Hudgins, "Fiberglas process control using DDC," presented at the 1967 IFAC/IFIP Internatl. Conf., Menton, France.

[22] "Computer operation in Owens–Illinois glass container plant," *Natl. Glass Budget,* September 25, 1965.

[23] "First computerized glass container plant dedicated at Dayville by Knox," *Natl. Glass Budget,* October 15, 1965.

[24] "Ford's computer controlled plant to operate non-stop for three years," *Natl. Glass Budget,* July 22, 1967.

[25] "Ford's flat glass plant hikes capacity, quality," *Automotive News,* May 1, 1967.

[26] R. M. Gagne, Ed., *Psychological Principles in System Development.* New York: Holt, Rinehart, and Winston, 1962.

[27] J. C. Kennedy, "Psychology and system development," in *Phychological Principles in System Development,* R. M. Gagne, Ed. New York: Holt, Rinehart, and Winston, 1962.

[28] I. L. Auerback, "The information revolution and its impact on automatic control," presented at the 1963 IFAC Cong., Basel, Switzerland.

[29] K. Davis, *Human Relations at Work: The Dynamics of Organizational Behavior,* 3rd ed. New York: McGraw-Hill, 1967.

centro de educación continua
división de estudios superiores
facultad de ingeniería, unam

INGENIERIA DE CONTROL DIGITAL DE PROCESOS Y SUS APLICACIONES

MICAS
(PROGRAMACION LINEAL)

M. EN C. VERONICA CZITROM

NOVIEMBRE, 1978.

PROGRAMACIÓN LINEAL

ARQUITECTURA, INGENIERÍA, CONSTRUCCIÓN, PLANEACIÓN URBANA Y REGIONAL

MINIMIZAR COSTO \ FUNCIONES OBJETIVO
MAXIMIZAR GANANCIA / LINEALES

RESTRICCIONES LINEALES

EJEMPLO

MÁQUINA	# HORAS REQUERIDAS		# TOTAL HORAS DISPONIBLES
	PROD. 1	PROD. 2	
1	2	1	70
2	1	1	40
3	1	3	90
GANANCIA/PIEZA	40	60	

MAX. GANANCIA

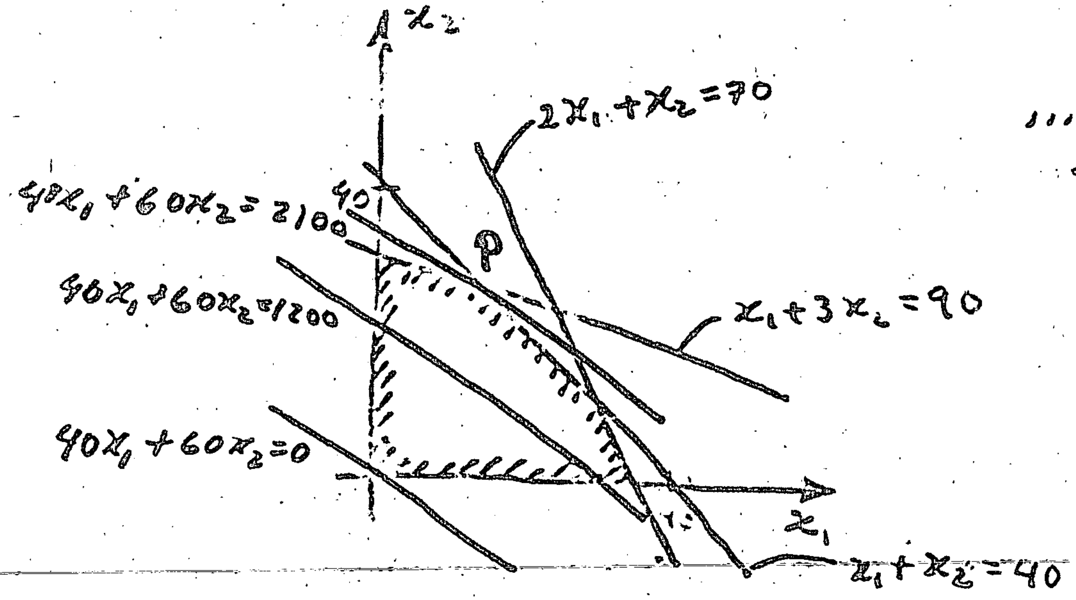
x_1 = # DE PRODUCTOS 1
x_2 = # " " 2

$$\text{RESTRICCIONES} \begin{cases} \text{MAQ. } 1 : & 2x_1 + x_2 \leq 70 \\ \quad '' \quad 2 : & x_1 + x_2 \leq 40 \\ \quad '' \quad 3 : & x_1 + 3x_2 \leq 90 \end{cases}$$

$$\left. \begin{matrix} x_1 \geq 0 \\ x_2 \geq 0 \end{matrix} \right\} \text{ CONDICIONES DE NO NEGATIVIDAD}$$

MAX: GANANCIA $= 40x_1 + 60x_2$

SOLUCIÓN GRÁFICA



//// : REGIÓN DE SOLUCIONES FACTIBLES

SOL. EN FRONTERA PARA REGIÓN CONVEXA

P: $x_1 = 15,\ x_2 = 25$

SOLUCIÓN ANALÍTICA (METODO SIMPLEX)

LADOS POLÍGONO CONVE

$$\begin{cases} 2x_1 + x_2 \le 70 \\ x_1 + x_2 \le 40 \\ x_1 + 3x_2 \le 90 \end{cases} \longrightarrow \begin{aligned} 2x_1 + x_2 + x_3 \quad\quad &= 70 \\ x_1 + x_2 \quad + x_4 \quad &= 40 \\ x_1 + 3x_2 \quad\quad + x_5 &= 90 \end{aligned}$$

$x_i \ge 0 \quad i = 1, 2, 3, 4, 5$

SIST. 3 ECS 5 INCOGNITAS

MAX. $M = 40x_1 + 60x_2$

1ª SOLUCION FACTIBLE: $x_1 = x_2 = 0$

$$x_1 = x_2 = 0 \Rightarrow \left.\begin{aligned} x_3 &= 70 \\ x_4 &= 40 \\ x_5 &= 90 \end{aligned}\right\rangle \text{VARIABLES DE LA BASE}$$

$M = 40 \times 0 + 60 \times 0 = 0$

$$M = 40x_1 + 60x_2 \begin{cases} x_1 : 0 \to 1 \Rightarrow M : 0 \to 40 \\ x_2 : 0 \to 1 \Rightarrow M : 0 \to 60 \Longleftarrow \end{cases}$$

MAXIMO INCREMENTO EN M.

CON $x_1 = 0$

$$\left.\begin{array}{l} 2x_1 + x_2 + x_3 = 70 \\ x_1 + x_2 + x_4 = 40 \\ x_1 + 3x_2 + x_5 = 90 \end{array}\right\} \longrightarrow \begin{array}{l} x_3 = 70 - x_2 \\ x_4 = 40 - x_2 \\ x_5 = 90 - 3x_2 \end{array}$$

$$x_2 \nearrow \Rightarrow \left\{ \begin{array}{l} x_3 \searrow \\ x_4 \searrow \\ x_5 \searrow \end{array} \right.$$

$x_3 = 0 \Rightarrow x_2 = 70$

$x_4 = 0 \Rightarrow x_2 = 40$

$x_5 = 0 \Rightarrow x_2 = \frac{90}{3} = 30 \longleftarrow$ EL MENOR

2ª SOLUCION FACTIBLE

$$\begin{array}{ll} x_1 = 0 & x_2 = 30 \\ x_5 = 0 & x_3 = 40 \\ & x_4 = 10 \end{array} \Bigg\rangle \text{ VARS. BASE}$$

$M = 40x_1 + 60x_2 = 0 + 60 \times 30 = 1800 = M$

$M = 40x_1 + 60x_2 = 40x_1 + 60\left(30 - \frac{x_5}{3} - \frac{x_1}{3}\right)$

$$M = 1800 + 20x_1 - 20x_5 \left\langle \begin{array}{l} x_1 \nearrow 1 \Rightarrow M \nearrow 20 \longleftarrow \\ x_5 \nearrow 1 \Rightarrow M \searrow 20 \end{array} \right.$$

SIST. ECS:

$$\left\{ \begin{array}{l} \frac{5}{3}x_1 + x_3 - \frac{1}{3}x_5 = 40 \\ \frac{2}{3}x_1 + x_4 - \frac{1}{3}x_5 = 10 \\ \underbrace{\frac{1}{3}x_1 + x_2 + \frac{1}{3}x_5}_{\text{VARS. BASE 2ª ITERACIÓN}} = 30 \end{array} \right. \quad \text{USANDO 3ª EC}$$

CON $x_5 = 0$

$$\left\{ \begin{array}{l} \frac{5}{3}x_1 + x_3 - \frac{1}{3}x_5 = 40 \\ \frac{2}{3}x_1 + x_4 - \frac{1}{3}x_5 = 10 \\ \frac{1}{3}x_1 + x_2 + \frac{1}{3}x_5 = 30 \end{array} \right. \quad \left\{ \begin{array}{ll} x_3 = 40 - \frac{5}{3}x_1 = 0 & x_1 = 24 \\ x_4 = 10 - \frac{2}{3}x_1 = 0 & x_1 = 15 \Leftarrow \\ x_2 = 30 - \frac{1}{3}x_1 = 0 & x_1 = 90 \end{array} \right.$$

3ª SOLUCIÓN FACTIBLE

$x_4 = 0 \qquad x_1 = 15$

$x_5 = 0 \qquad x_2 = 25$

$\qquad\qquad\quad x_3 = 15$

$M = 1800 + 20x_1 - 20x_5 = 1800 + 20\left(15 - \frac{3}{2}x_4 + \frac{1}{2}x_5\right) - 2$

$M = 2100 - 30x_4 - 10x_5$

$M = 2100$

$$M = 2100 - 30x_4 - 10x_5 \left\langle \begin{array}{l} x_4 \nearrow 1, \quad M \searrow 30 \\ x_5 \nearrow 1, \quad M \searrow 10 \end{array} \right.$$

$\therefore$ YA SE TIENE SOL. FACTIBLE ÓPTIMA.

INTERPRETACIÓN

$x_1 = 15$: 15 PRODUCTOS # 1

$x_2 = 25$: 25 " " 2

$$\begin{array}{lllll} \text{MAQ. 1:} & 2x_1 + x_2 + x_3 & & & = 70 \\ \text{MAQ. 2:} & x_1 + x_2 & + x_4 & & = 40 \\ \text{MAQ. 3:} & x_1 + 3x_2 & & + x_5 & = 90 \end{array}$$

$x_3 = 15 \Rightarrow$ HOLGURA DE 15 HORAS MAQ. 1

$\left.\begin{array}{l} x_4 = 0 \\ x_5 = 0 \end{array}\right\} \Rightarrow$ NO HAY HOLGURA MAQS. 2 Y 3

MAX. GANANCIA = M = 2100

NOTACIÓN MATRICIAL

$$\begin{cases} 2x_1 + x_2 \leq 70 \\ x_1 + x_2 \leq 40 \\ x_1 + 3x_2 \leq 90 \end{cases}$$

$x_1 \geq 0, \; x_2 \geq 0$

MAX: $M = 40x_1 + 60x_2$

$A\underline{x} \leq \underline{b}$

$\underline{x} \geq \underline{0}$

MAX: $M = \underline{c}^T$

$A = \begin{pmatrix} 2 & 1 \\ 1 & 1 \\ 1 & 3 \end{pmatrix}, \; \underline{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}, \; \underline{b} = \Big(\begin{matrix} 7 \\ 4 \\ 9 \end{matrix}$

$\underline{c}^T = (40 \quad 60)$

METODO SIMPLEX EN TÉRMINOS DE MATRICES

PROGRAMA PARA COMPUTADORA

1ª ITERACIÓN

BASE

TABLEAUX (TABLA)

$$\begin{array}{c|cccccc|l}
 & x_1 & x_2 & x_3 & x_4 & x_5 & b & \\
 & 2 & 1 & 1 & 0 & 0 & 70 & 70 \\
 & 1 & 1 & 0 & 1 & 0 & 40 & 40 \\
 & 1 & \boxed{3} & 0 & 0 & 1 & 90 & 90/3 = 30 \leftarrow \text{MAS CHICO} \\
\hline
M & -40 & -60 & 0 & 0 & 0 & 0 &
\end{array}$$

$\uparrow$ MASNEG

PIVOTE

$M - 40x_1 - 60x_2 = 0$

2ª ITERACIÓN

$$\begin{array}{c|cccccc|l}
 & x_1 & x_2 & x_3 & x_4 & x_5 & b & \\
 & 5/3 & 0 & 1 & 0 & -1/3 & 40 & 40/5/3 = 24 \\
 & \boxed{2/3} & 0 & 0 & 1 & -1/3 & 10 & 10/2/3 = 15 \leftarrow \\
 & 1/3 & 1 & 0 & 0 & 1/3 & 30 & 30/1/3 = 90 \\
\hline
M & -20 & 0 & 0 & 0 & 20 & 1800 &
\end{array}$$

$\uparrow$ MAS NEG.

$M - 20x_1 + 20x_5 = 1800$

3ª ITERACIÓN

$$\begin{array}{cccccc}
x_1 & x_2 & x_3 & x_4 & x_5 & b \\
0 & 0 & 1 & -2.5 & 0.5 & 15 \\
1 & 0 & 0 & 1.5 & -0.5 & 15 \\
0 & 1 & 0 & -0.5 & 0.5 & 25 \\
0 & 0 & 0 & 30 & 10 & 2100
\end{array}$$

$\Rightarrow x_3 = 15$

$x_1 = 15$

$x_2 = 25$

$x_4 = x_5 = 0$

M

$M + 30x_4 + 10x_5 = 2100$

GRAMACION LINEAL

$$\left.\begin{array}{l} a_{11}x_1+a_{12}x_2+\dots+a_{1n}x_n \leq b_1 \\ \vdots \\ a_{j1}x_1+a_{j2}x_2+\dots+a_{jn}x_n \geq b_j \\ \vdots \\ a_{m1}x_1+a_{m2}x_2+\dots+a_{mn}x_n = b_m \end{array}\right\}$$

m ECUACION, LINEALES Ó DESIGUALDADES CON n VARIABLES

COEFICIENTES ESTRUCTURALES

RESTRICCIONES

$x_i \geq 0$ CONDICIONES DE NO-NEGATIVIDAD

$$\text{OPT. } z = \underbrace{c_1x_1+c_2x_2+\dots+c_nx_n}_{\text{COEFICIENTES DE COSTOS}}$$

EJEMPLO

$$\text{MIN: } w = 2x_1+4x_2 \qquad \begin{cases} 3x_1+2x_2 \leq 5 \\ x_1-4x_2 \geq 3 \\ 5x_1+x_2 = 7 \end{cases}$$

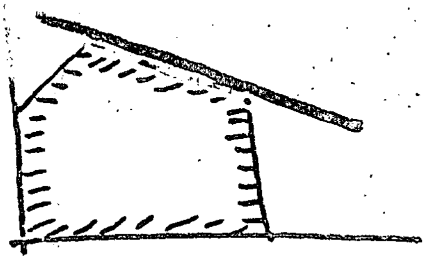
DESIGUALDADES → IGUALDADES

		VARIABLES
$3x_1+2x_2+x_3$	$=5$	+HOLGORA
$x_1-4x_2-x_4+a_1$	$=3$	-HOLGORA+ARTIFICIAL
$5x_1+x_2+a_2$	$=7$	+ARTIFICIAL

$$\text{MIN: } w = 2x_1+4x_2 \underbrace{\overset{-}{+}1000a_1\overset{-}{+}1000a_2}_{\text{PARA MAXIMIZACIÓN}}$$

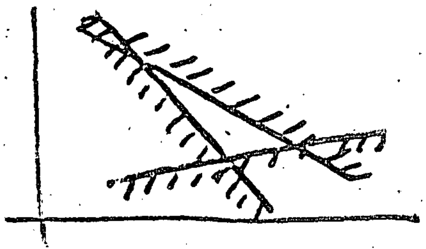
CASOS ESPECIALES

1) SOLUCIONES ÓPTIMAS NO ÚNICAS



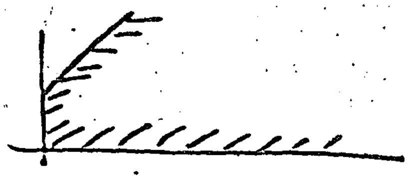
METODO SIMPLEX: $\underbrace{M_i = M_{i+1}}_{2\ \text{ITERACIONES}} = M$ MÁXIMA

2) RESTRICCIONES CONTRADICTORIAS



SIMPLEX: SOL. FACTIBLE CONTIENE VAR. ARTIF.

3) SOLUCIÓN NO ACOTADA



SIMPLEX: M TAN GRANDE COMO SE DESEE

PROBLEMA DUAL

PROBLEMA PRIMO	PROBLEMA DUAL
MAX: $M = \underline{c}^T \underline{x}$	MIN $V = \underline{b}^T \underline{y}$
$A\underline{x} \le \underline{b}$	$A^T \underline{y} \ge \underline{c}$
$\underline{x} \ge \underline{0}$	$\underline{y} \ge \underline{0}$

AL MENOS 21 UNIDADES VITAMINA A
" " 12 " B

ALIMENTO	VITAMINA/UNIDAD ALIMENTO		COSTO/UNIDAD ALIMEN
	A	B	
1 (NARANJA)	1	0	20
2 (MANZANA)	0	1	20
3 (LECHUGA)	1	2	31
4 (CHÍCHARO)	1	1	11
5 (ZANAHORIA)	2	1	12

MIN. COSTO

DIETISTA

x_i = CANTIDAD DE ALIMENTO i

$$\left.\begin{aligned} &\text{VIT. A}: x_1 + x_3 + x_4 + 2x_5 \geq 21 \\ &\text{VIT. B}: x_2 + 2x_3 + x_4 + x_5 \geq 12 \end{aligned}\right\rangle \text{ RESTRICCIONES}$$

$x_i \geq 0$ CONDS. NO NEGATIVIDAD

MIN: COSTO $= 20x_1 + 20x_2 + 31x_3 + 11x_4 + 12x_5$

FUNCIÓN OBJETIVO

DUAL: CIA. FARMACEUTICA "LA CAMPANA"

λ_1 = PRECIO DE CADA PÍLDORA DE VIT. A
λ_2 = " " " " " B

$\lambda_i \geq 0$ CONDS. NO NEGATIVIDAD

MAX. GANANCIA $= 21\lambda_1 + 12\lambda_2$ FUNCIÓN OBJETIVO

PRECIOS COMPETITIVOS:

$$\left.\begin{aligned} 1\lambda_1 &\leq 20 \\ \lambda_2 &\leq 20 \\ \lambda_1 + 2\lambda_2 &\leq 31 \\ \lambda_1 + \lambda_2 &\leq 11 \\ 2\lambda_1 + \lambda_2 &\leq 12 \end{aligned}\right\} \text{ RESTRICCIONES}$$

DIETISTA

$$\begin{pmatrix}1&0&1&1&2\\0&1&2&1&1\end{pmatrix}\begin{pmatrix}x_1\\x_2\\x_3\\x_4\\x_5\end{pmatrix}\geq\begin{pmatrix}21\\12\end{pmatrix}$$

$$\begin{pmatrix}x_1\\x_2\\x_7\\x_4\\x_5\end{pmatrix}\geq\begin{pmatrix}0\\0\\0\\0\\0\end{pmatrix}$$

$$\text{MIN: COSTO} = (20\ \ 20\ \ 31\ \ 11\ \ 12)\begin{pmatrix}x_1\\x_2\\x_3\\x_4\\x_5\end{pmatrix}$$

"LA CAMPANA"

$$\begin{pmatrix}1&0\\0&1\\1&2\\1&1\\2&1\end{pmatrix}\begin{pmatrix}\lambda_1\\\lambda_2\end{pmatrix}\leq\begin{pmatrix}20\\20\\31\\11\\12\end{pmatrix}$$

$$\begin{pmatrix}\lambda_1\\\lambda_2\end{pmatrix}\geq\begin{pmatrix}0\\0\end{pmatrix}$$

$$\text{MAX: GANANCIA} = (21\ \ 12)\begin{pmatrix}\lambda_1\\\lambda_2\end{pmatrix}$$

EJEMPLO

CONSTRUCCIÓN CASAS DE 2, 3 Y 4 RECÁMARAS

MAX. GANANCIA.

RESTRICCIONES:

1.- PROYECTO $\leq$ \$9,000,000

2.- # UNIDADES $\geq$ 350

3.- MAX PORCENTAJE: CASAS 2 REC: 20%

" 3 " : 60%

" 4 " : 40%

4.- COSTOS CONSTRUC. " 2 " : \$20,000

" 3 " : \$25,000

" 4 " : \$30,000

5.- GANANCIA NETA : " 2 " : \$2,000

" 3 " : \$3,000

" 4 " : \$4,000

5.- GANANCIA: $Z = 2x_1 + 3x_2 + 4x_3$ (EN MILES DE \$)

1.- 4.- $20,000x_1 + 25,000x_2 + 30,000x_3 \le 9,000\,000$

$20x_1 + 25x_2 + 30x_3 \le 9,000$

2.- $x_1 + x_2 + x_3 \ge 350$

3.- $x_1 \le 0.2(x_1 + x_2 + x_3)$

$x_2 \le 0.6(x_1 + x_2 + x_3)$

$x_3 \le 0.4(x_1 + x_2 + x_3)$

$20x_1 + 25x_2 + 30x_3 \le 9,000 \rightarrow 20x_1 + 25x_2 + 30x_3 + x_4 = 9,000$

$x_1 + x_2 + x_3 \ge 350 \rightarrow \qquad x_1 + x_2 + x_3 - x_5 = 350$

$x_1 \le .2(x_1 + x_2 + x_3) = .2(350 + x_5) \rightarrow x_1 - .2x_5 + x_6 = 70$

$x_2 \le .6(x_1 + x_2 + x_3) \rightarrow x_2 - .6x_5 + x_7 = 210$

$x_3 \le .4(x_1 + x_2 + x_3) \rightarrow x_3 - .4x_5 + x_8 = 0$

5 ECUACIONES EN 8 INCOGNITAS ⇒ <u>56</u> POSIBLES SOL.

(SIN SIMPLEX: RESOLVER SIST. 5 ECS, 5 INCOGNIT.
56 VECES, PARA DETERMINAR SOL. OPTIMA).

PARA TENER 1a SOLUCION FACTIBLE CON $x_1 = x_2 = x_3 =$

RESTRICCION: $x_1 + x_2 + x_3 - x_5 + \overset{\text{VAR. ARTIFICIAL}}{x_9} = 350$

FUN. OBJETIVO: $Z = 2x_1 + 3x_2 + 4x_3 - \underbrace{1000}_{\text{COEF. COSTO GD.}}x_9$

$$\text{MAX: } Z = 2x_1 + 3x_2 + 4x_3 - 1000x_9$$

EST.

$$\begin{array}{llllllllll}
20x_1 + 25x_2 + 30x_3 + x_4 & & & & & = 9,000 \\
x_1 + x_2 + x_3 & -x_5 & & & + x_9 & = 350 \\
x_1 & -.2x_5 + x_6 & & & & = 70 \\
x_2 & -.6x_5 & + x_7 & & & = 210 \\
x_3 & -.4x_5 & & + x_8 & & = 140
\end{array}$$

MINIMIZACIÓN

MIN → MAX

EJEMPLO: EQUIVALENTE

MIN: $y = 3x_1 + 4x_2 + 7x_3$	MAX: $z = -y = -3x_1 - 4x_2 - 7x_3$
$x_1 - 3x_2 + x_3 \geq 7$	$x_1 - 3x_2 + x_3 \geq 7$
$2x_1 + x_2 - x_3 \geq 9$	$2x_1 + x_2 - x_3 \geq 9$
$x_i \geq 0$	$x_i \geq 0$

APLICACIONES

PROBLEMA DE COMBINACIÓN

COMPONENTES: SE COMBINAN PARA DAR 1 Ó MAS PRODUCTOS

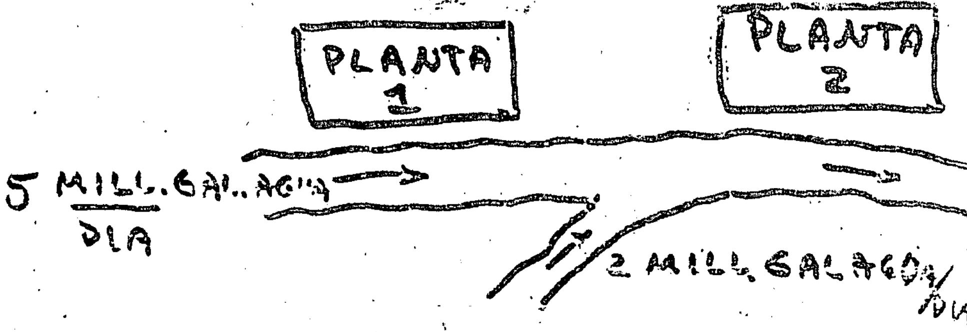
TIENEN CIERTOS COSTOS Y CARACTERISTICA

? = CANTIDAD DE CADA COMPONENTE

SUJETAS A CIERTAS LIMITACIONES

MIN: COSTO TOTAL

EJEMPLO



GENERACIÓN DE UNIDADES DE CONTAMINANTE AL DÍA	20	14
COSTO REMOVER 1 UNIDAD CONTAM.	$1000	$800

20% → CONT. SE ELIMINA

CUANDO MÁS: 2 UNID. CONT/MILLON DE GALONES

? = OPERACIÓN MAS BARATA DE TRATAMIENTO CONTAMIN

$x_{1,2}$ = # DE UNIDADES DE CONTAMINANTE ELIMINADAS POR LAS PLANTAS 1,2

MIN: COSTO = $\$1000 x_1 + \$800 x_2$

PLANTA 1 : $$\underbrace{(\overset{\text{SE GENERAN}}{20} - \overset{\text{SE ELIMINAN}}{x_1})}_{\text{UNIDADES DE CONTAMINANTE}} \le \left(\frac{5 \text{ MILL. GAL. AGUA}}{\text{DIA}}\right)\left(\frac{2 \text{ UNID. CO}}{\text{MILL. GAL}}\right.$$

PLANTA :

$$\underbrace{.8(20-x_1)}_{\text{UNID. CONT. LLEGAN DE PLANTA 1}} + \underbrace{(14-x_2)}_{\text{UN. CONT. SE GENERAN}} \le (5+2)\frac{\text{MILL. G.}}{\text{DIA}}\left(\frac{2 \text{ U.C.}}{\text{M.G.}}\right)$$

$x_1 \le 20$ $\qquad$ $x_i \ge 0$

$x_2 \le 14$

MIN. COSTO = $1000 x_1 + 800 x_2$

RESTRICCIONES: $x_1 \ge 10$

$.8x_1 + x_2 \ge 16$

$x_1 \le 20$

$x_2 \le 14$

CONDS. NO NEGAT. $x_1 \ge 0$ ← SUPERFLUA

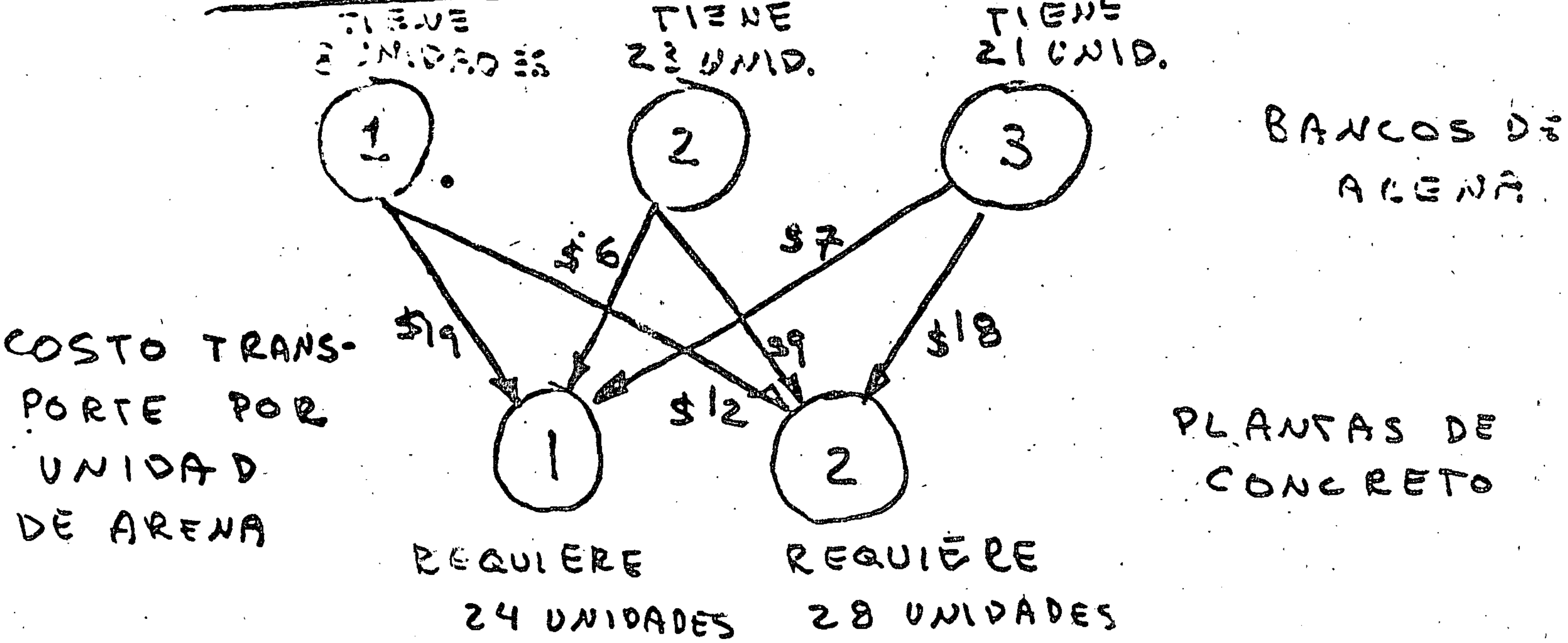
$x_2 \ge 0$

PROBLEMAS DE TRANSPORTE

UN PRODUCTO - SE TRANSPORTA DE m CENTROS DE PRODUCCIÓN EN CANTIDADES $a_1, \dots, a_m$

- SE RECIBE EN n CENTROS EN CANT. $b_1, \dots,$
- SE CONOCEN COSTOS TRANSP. CENTRO i DESTINO j
- ? = CANT. QUE SE TRANSP CENTRO i A DESTINO j
- MIN. COSTO TRANSPORTE

EJEMPLO



?= CUÁNTA ARENA TRANSPORTAR DE CADA BANCO A CADA PLANTA
MIN: COSTOS DE TRANSPORTE.

---

x_{ij} = NUM. UNIDADES DE ARENA DE BANCO i A PLANTA j.

BANCO DE ARENA 1: $x_{11} + x_{12} \leq 8$
" " " 2: $x_{21} + x_{22} \leq 23$
" " " 3 $x_{31} + x_{32} \leq 21$

PLANTA DE CONCRETO 1: $x_{11} + x_{21} + x_{31} = 24$
" " " 2: $x_{12} + x_{22} + x_{32} = 28$

$$x_{ij} \geq 0$$

MIN. COSTOS DE TRANSPORTE =

$$19x_{11} + 12x_{12} + 6x_{21} + 9x_{22} + 7x_{31} + 18x_{32}$$

PROBLEMAS DE TRANSPORTE

TIPO PROG. LINEAL; SE SIMPLIFICAN POR LA ESTRUCTURA PARTICULAR DEL PROBLEMA

m FABRICAS QUE PRODUCEN CANTIDADES $a_1, \ldots, a_m$ DE UN PRODUCTO

n CENTROS DE CONSUMO QUE CONSUMEN $b_1, \ldots, b_n$

c_{ij} : COSTO TRANSPORTAR PRODUCTO DE [illegible] QUE SE FABRICA [illegible] CONSUME

EJEMPLO

		CENTROS CONSUMO 1	2	INVENTARIOS
FABRICAS	1	15	20	15
	2	18	22	25
	3	25	19	20
DEMANDA		35	25	60

COSTOS TRANSPORTE

OFERTA = DEMANDA

MIN COSTO TRANSPORTE
PLAN DE TRANSPORTE = ?

---

x_{ij} = NUM. UNIDADES DE FABRICA i A CENTRO j

$$\text{MIN: } Z = 15x_{11} + 18x_{21} + 25x_{31} + 20x_{12} + 22x_{22} + 19x_{32}$$

$$\left.\begin{aligned} x_{11} + x_{12} &= 15 \\ x_{21} + x_{22} &= 25 \\ x_{31} + x_{32} &= 19 \end{aligned}\right\} \text{PRODUCCIÓN (INVENTARIOS)}$$

$$\left.\begin{aligned} x_{11} + x_{21} + x_{31} &= 35 \\ x_{12} + x_{22} + x_{32} &= 25 \end{aligned}\right\} \text{DEMANDA CENTRO CONSUMO}$$

$$x_{ij} \geq 0 \quad \forall i, \forall j$$

DEMANDA, SOLO 4 DE LAS 6 ECUACIONES SON LINEALMENTE INDEPENDIENTES; SE PUEDE ELIMINAR 1 ECUACIÓN.

POR METODO SIMPLEX, HABRÍA QUE INTRODUCIR 4 VARS. ARTIFICIALES PARA FORMAR UNA SOL. INICIAL FACTIBLE.

SE PUEDE APLICAR UNA SIMPLIFICACIÓN DEL MÉTODO SIMPLEX PORQUE:

1.- LOS COEFICIENTES DE TODAS LAS VARS. $= \underline{1}$

2.- CUALQUIER VARIABLE x_{ij} APARECE UNA SOLA VEZ EN LAS PRIMERAS 3 ECUACIONES
UNA " " " " ÚLTIMAS 2

$$\text{MATRIZ DE COSTOS} = \begin{pmatrix} 15 & 20 \\ 18 & 22 \\ 25 & 19 \end{pmatrix}$$

$$\text{MATRIZ DE DISTRIBUCIÓN} = \begin{pmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \\ x_{31} & x_{32} \end{pmatrix} = ?$$

QUITANDO ULTIMA ECUACIÓN:

$$\left.\begin{array}{r} x_{11} + x_{12} = 15 \\ x_{21} + x_{22} = 25 \\ x_{31} + x_{32} = 20 \\ x_{11} + x_{21} + x_{31} = 35 \end{array}\right\}$$

CONJUNTO DE 4 ECUACIONES LINEALMENTE INDEPENDIENTES CON 6 INCÓGNITAS

UNA SOL. FACTIBLE: DOS VARIABLES IGUALES A CERO, SIN CREAR INCONSISTENCIA

POR EJEMPLO: $x_{12} = 0$, $x_{31} = 0$

$$\left\{\begin{array}{l} x_{11} = 15 \\ x_{21} + x_{22} = 25 \\ x_{32} = 20 \\ x_{11} + x_{21} = 35 \end{array}\right. \qquad \left\{\begin{array}{l} x_{11} = 15 \\ x_{11} + x_{21} = 35 \\ x_{21} + x_{22} = 25 \\ x_{32} = 20 \end{array}\right.$$

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_{11} \\ x_{21} \\ x_{22} \\ x_{32} \end{pmatrix} = \begin{pmatrix} 15 \\ 35 \\ 25 \\ 20 \end{pmatrix}$$

MATRIZ TRIANGULAR => SIST. SE RESUELVE FÁCILMENTE

$x_{11} = 15$ $x_{12} = 0$

$x_{21} = 20$ $x_{22} = 5$

$x_{31} = 0$ $x_{32} = 20$

SOLUCIÓN FACTIBLE, NO NECESARIAMENTE ÓPTIMA.

OPTIMIZACIÓN

FORMULACIÓN MATEMÁTICA GENERAL:

ENCONTRAR EL VALOR DE LAS VARIABLES

$(x_1, x_2, \ldots, x_n)$

QUE MAXIMICEN O MINIMICEN (OPTIMICEN)
A LA FUNCIÓN (OBJETIVO)

$$M = M(x_1, x_2, \ldots, x_n)$$

SUJETA A LAS RESTRICCIONES

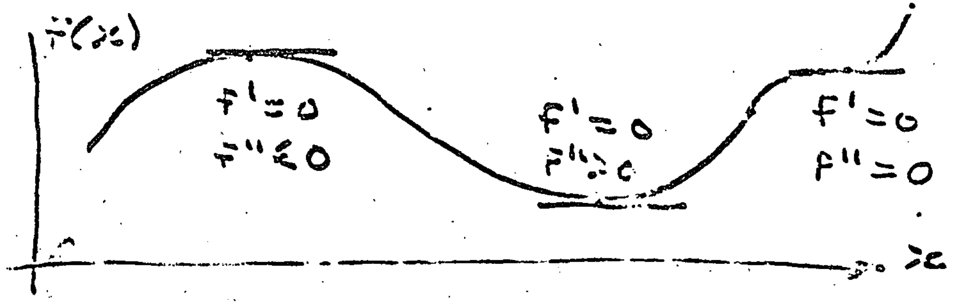
$$C_i(x_1, x_2, \ldots, x_n) = 0 \qquad i = 1, \ldots, p$$
$$C_i(x_1, x_2, \ldots, x_n) \leq 0 \qquad i = p+1, \ldots, r$$
$$C_i(x_1, x_2, \ldots, x_n) \geq 0 \qquad i = r+1, \ldots, m$$

m ECS., n VARIABLES (INCÓGNITAS)

ESTRATEGIAS
DE OPTIMIZACIÓN: $\left\langle \begin{array}{l} \text{GRADIENTE} \\ \text{ENUMERACIÓN} \end{array} \right.$

RESTRICCIONES → REGIÓN DE SOLUCIONES FACTIBLES

OPTIMIZACIÓN POR DIFERENCIACIÓN

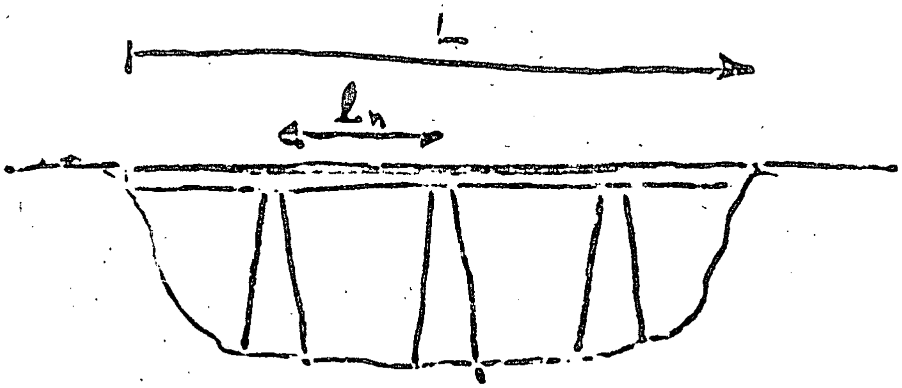


f (x)

↓

UNA SOLA VAR. INDEPENDIENTE

EJEMPLO.



$\ell_n = ?$ PARA MIN. COSTO

COSTO POR METRO PUENTE ~ DIST. ENTRE PILARES

COSTO PILARES $\simeq$ CONST.

$n = \#$ CLAROS $= L/L_n$

A = COSTO (FIJO) PILARES TERMINALES

P = " PILASTRA INTERMEDIA

ω = COSTO POR METRO DE PUENTE

$$\omega = k L_n$$

$$\text{MIN: COSTO} = A + (n-1)P + L \underbrace{k L_n}_{\omega}$$

$$= A + \left(\frac{L}{L_n} - 1\right)P + L k L_n$$

$$\frac{dC}{dL_n} = -\frac{LP}{L_n^2} + kL = 0$$

$$\therefore \boxed{L_n = \sqrt{\frac{P}{k}}}$$

$$\frac{d^2C}{dL_n^2} = \frac{2LP}{L_n^3} > 0 \Rightarrow \text{COSTO } \underline{\text{MÍNIMO}}$$

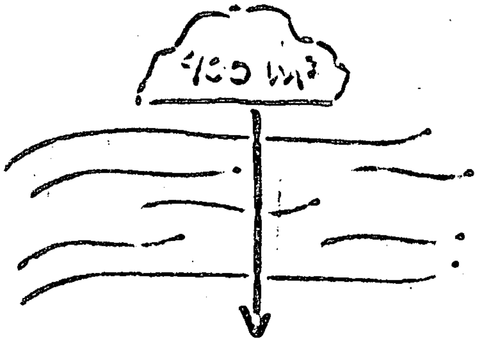
VARIAS VARIABLES INDEPENDIENTES:

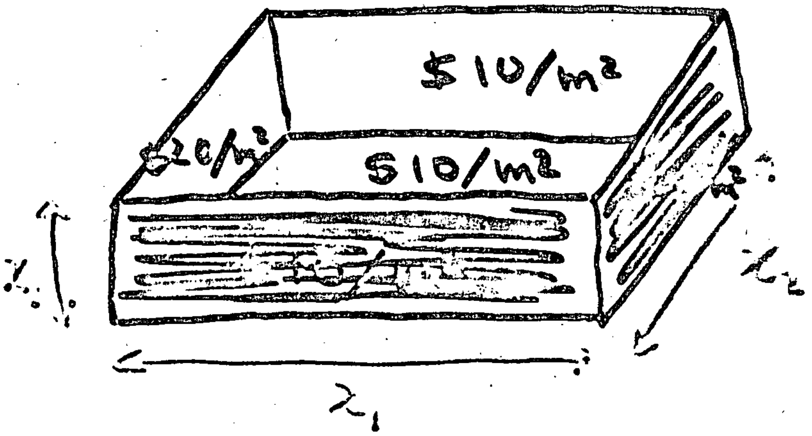
$f(x_1, \dots, x_n)$

$$\text{SI:} \begin{cases} \dfrac{\partial f}{\partial x_1} = 0 \\ \vdots \\ \dfrac{\partial f}{\partial x_n} = 0 \end{cases}$$

ENTONCES LA FUNCIÓN f ES MAX O MIN

EJEMPLO





CADA VIAJE: \$.10

$x_1, x_2, x_3 = ?$, x_4 = NUM VIAJES = ?

MIN: COSTO TRANSPORTE

$$\text{COSTO} = 10(x_1 x_2 + 2x_1 x_3) + 20(2x_2 x_3) + .10x_4$$
$$= 10x_1 x_2 + 20x_1 x_3 + 40x_2 x_3 + .10x_4$$

RESTRICCIÓN: $\underbrace{x_1 x_2 x_3 x_4}_{\#\text{ VIAJES} \times \text{VOLUMEN CAJA}} = \underbrace{400}_{\text{MATERIAL TRANSPORTADO}}$

DESPEJANDO $x_4 = \dfrac{400}{x_1 x_2 x_3}$ Y SUST.

$$\text{MIN: COSTO} = 10x_1 x_2 + 20x_1 x_3 + 40x_2 x_3 + \frac{40}{x_1 x_2 x_3}$$

$$\begin{cases} \dfrac{\partial c}{\partial x_1} = 10x_2 + 20x_3 - \dfrac{40}{x_1^2 x_2 x_3} = 0 \\ \dfrac{\partial c}{\partial x_2} = 10x_1 + 40x_3 - \dfrac{40}{x_1 x_2^2 x_3} = 0 \\ \dfrac{\partial c}{\partial x_3} = 20x_1 + 40x_2 - \dfrac{40}{x_1 x_2 x_3^2} = 0 \end{cases}$$

SOLUCIÓN: $x_1 = 2$

$x_2 = 1$

$x_3 = 1/2$

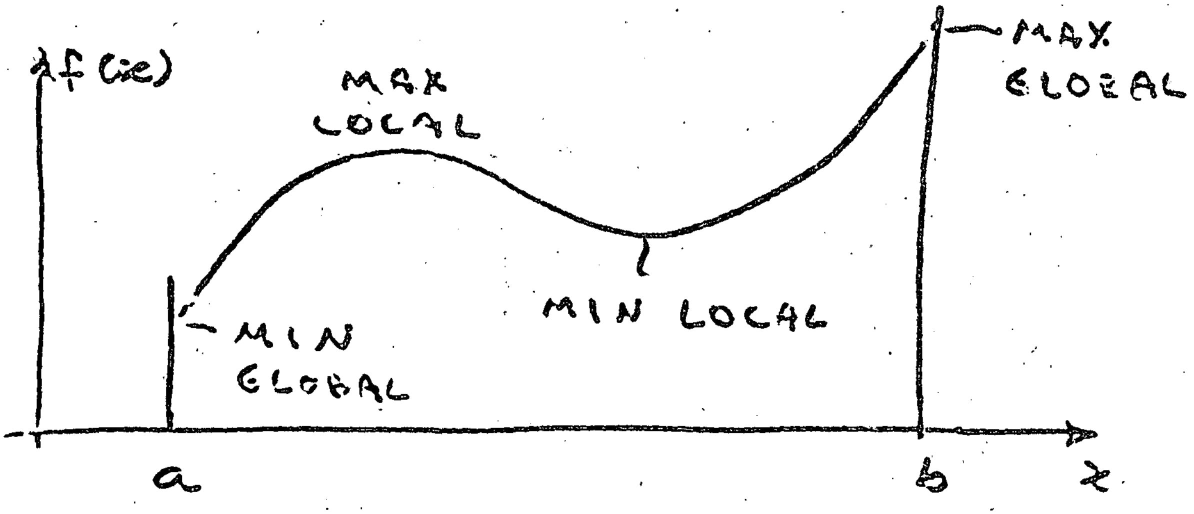
$$x_4 = \frac{400}{x_1 x_2 x_3} = 400$$

$$c = 5100$$

OPTIMIZACIÓN DE FUNCIONES DE UNA SOLA VARIABLE, SIN RESTRICCIONES.

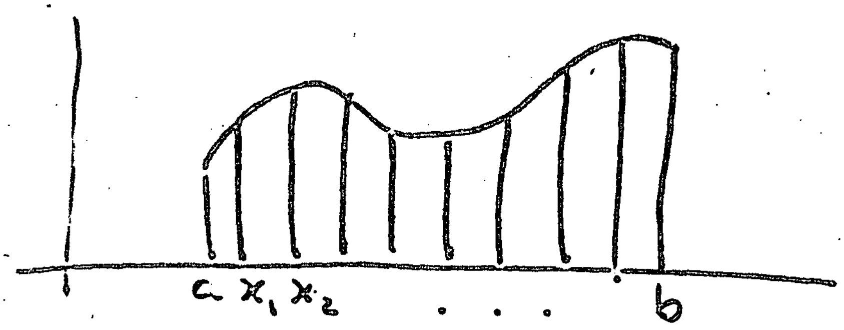
MET. SIMULTAN: BÚSQUEDA EXHAUSTIVA
BÚSQUEDA ALEATORIA

MET. SECUENC: TRISECCIÓN
FIBONACCI } UN SOLO MAX O MIN EN INTERVALO (UNIMOD



DIFERENCIACIÓN → LOCALES

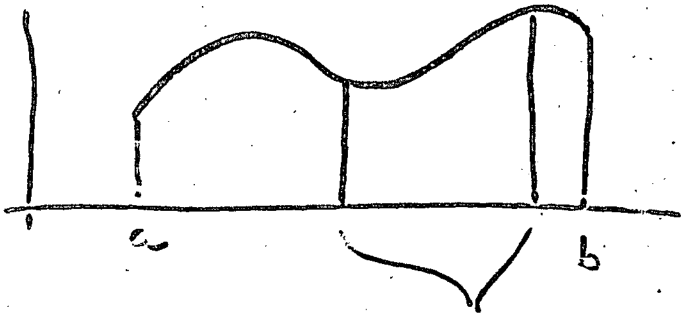
BÚSQUEDA EXHAUSTIVA



MUCHAS EVALUACIONES

AUMENTA NUM. INTERVALOS, AUMENTA PRECISIÓN

BÚSQUEDA ALEATORIA

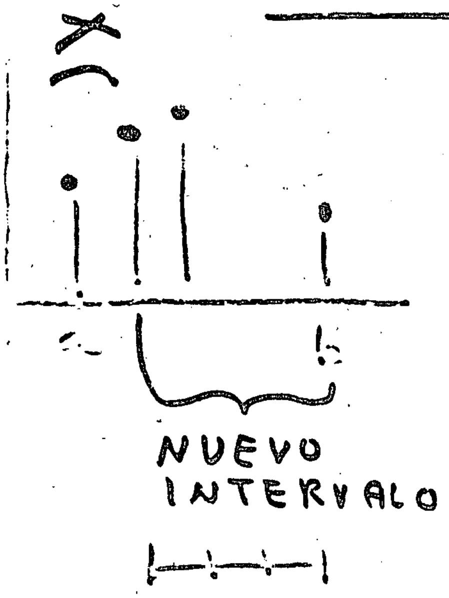


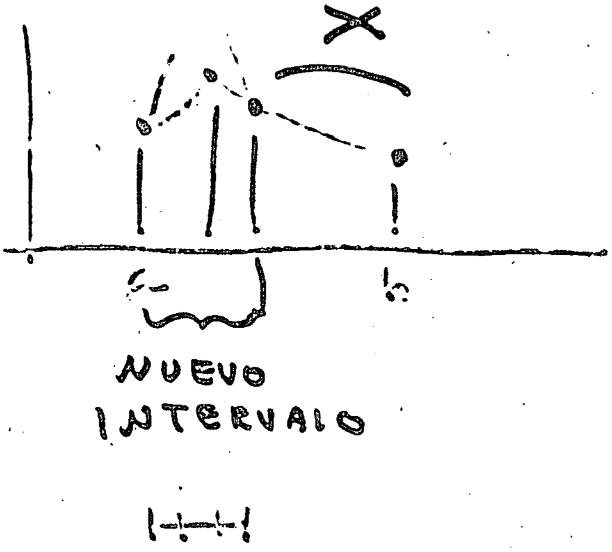
EJEMPLO: $f(x) = -.4x^2 + 4x$ EN $(0,10)$

DIFERENCIANDO: $f(\underset{x}{5}) = 10$ EXACTO

# DE NUMS. ALEATORIOS	x	$f(x)$
25	4.9051	9.9964
100	4.9051	9.9964
250	4.9091	9.9966
500	5.0058	9.9999

TRISECCIÓN

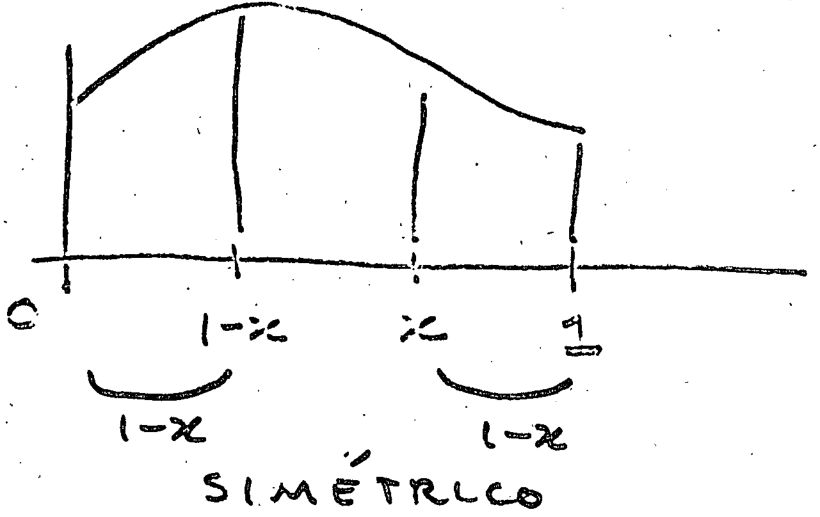


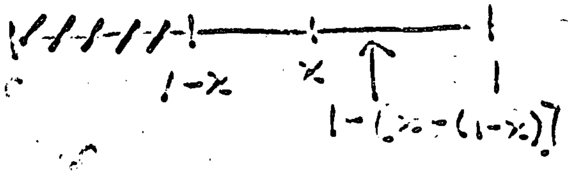


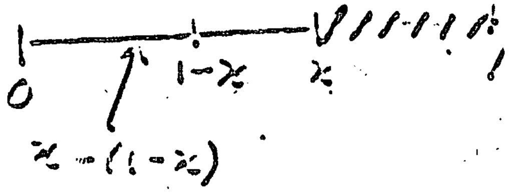
HASTA LLEGAR A INTERVALO SUF. CHICO
EN CADA ETAPA, SE EVALUA F EN 2 PUNTOS

FIBONACCI

1 EVALUACIÓN POR ETAPA





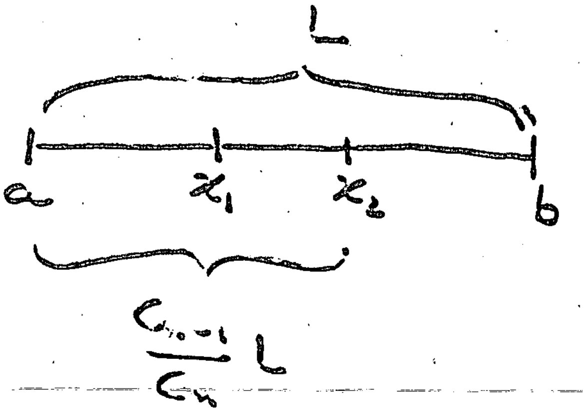


$$r = \frac{\text{INT. DESP. ITER.}}{\text{INT. ANTES ITER.}} = \frac{1}{2}, \frac{2}{3}, \frac{3}{5}, \frac{5}{8}, \frac{8}{13}, \ldots \rightarrow .618$$

(COCIENTE DORADO)

$r = \frac{2}{3}$ TRISECCIÓN (SE ELIMINA 1/3 DEL INTERVALO)

FIBONACCI: 40% EVALUACIONES TRISECCIÓN

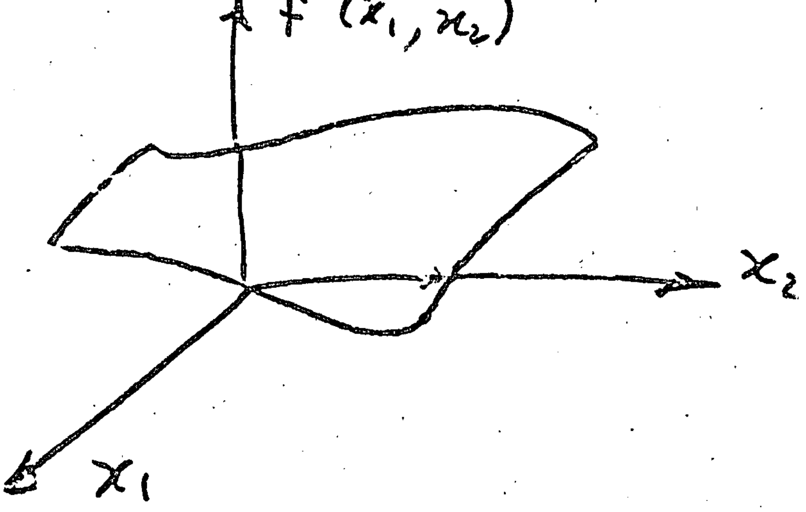


MÉTODOS DE BÚSQUEDA MULTIDIMENSIONAL

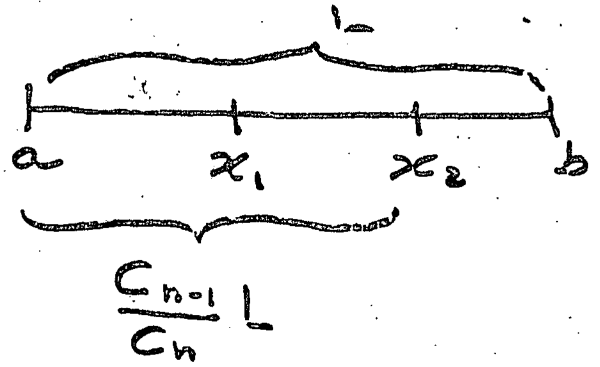
MET. SIMULTÁNEOS: BÚSQUEDA EXHAUSTIVA
: BÚSQUEDA ALEATORIA

MET. SECUENCIALES: BUSQUEDA DE REJILLA
" UNIVARIADA
" DE GRADIENTE
} FUN. UNIMODALES

$F(x_1, x_2, \ldots, x_n)$



FIBONACCI: 40% EVALUACIONES TRISECCIÓN



OPTIMIZACIÓN CON RESTRICCIONES:

MULTIPLICADORES DE LAGRANGE

OPTIMIZAR: FUNCIÓN OBJETIVO $M=M(x_1,x_2,\dots,x_n)$

SUJETA A RESTRICCIONES TIPO IGUALDAD

$$C_i(x_1,x_2,\dots,x_n)=0 \qquad i=1,2,\dots,m$$

EN PUNTO OPTIMO $p^*(\hat{x}_1,\hat{x}_2,\dots,\hat{x}_n)$: $dM=\sum_{i=1}^{n}\frac{\partial M}{\partial x_i}dx_i\Big|_{p^*}=0$ ①

EN CUALQUIER PUNTO: $dC_j=\sum_{i=1}^{n}\frac{\partial C_j}{\partial x_i}dx_i=0$
Y ∴ TAMBIEN EN p^*) $\qquad j=1,2,\dots,m$

MULTIPLICANDO POR λ_j, Y RESTANDO ESAS M ECUACIONES DE ①, SE TIENE:

$$dM-\sum_{j=1}^{m}dC_j=\sum_{i=1}^{n}\left(\frac{\partial M}{\partial x_i}-\sum_{j=1}^{m}\lambda_j\frac{\partial C_j}{\partial x_i}\right)dx_i\Big|_{p^*}=0$$

MO dx_i SON ARBITRARIAS,

$$\frac{\partial M}{\partial x_i}-\sum_{j=1}^{m}\lambda_j\frac{\partial C_j}{\partial x_i}=0 \qquad i=1,2,\dots,n$$

LAS n ECS $\quad \dfrac{\partial M}{\partial x_i} - \sum_{j=1}^{m} \lambda_j \dfrac{\partial C_j}{\partial x_i} = 0 \quad i = 1, \ldots, n$

Y LAS M RESTRICCIONES: $\quad C_j(x_1, \ldots, x_n) = 0 \quad j = 1, \ldots, m$

FORMAN UN SISTEMA DE $n+m$ ECUACIONES EN $n+m$ INCOGNITAS $\hat{x}_1, \ldots, \hat{x}_n$ (COORDENADAS DE P^*) Y $\lambda_1, \ldots, \lambda_m$.

λ_i : MULTIPLICADORES DE LAGRANGE.

SI $\quad L(x_1, x_2, \ldots, x_n, \lambda_1, \ldots, \lambda_m) = M - \sum_{j=1}^{m} \lambda_j C_j$

LAS $m+n$ ECS EN $m+n$ INCOGNITAS SON EQUIVALENTES A:

$$\left.\frac{\partial L}{\partial x_1}\right|_{P^*} = \left.\frac{\partial M}{\partial x_1}\right|_{P^*} - \sum_{j=1}^{m} \lambda_j \left.\frac{\partial C_j}{\partial x_1}\right|_{P^*} = 0$$

$$\vdots$$

$$\left.\frac{\partial L}{\partial x_n}\right|_{P^*} = \left.\frac{\partial M}{\partial x_n}\right|_{P^*} - \sum_{j=1}^{m} \lambda_j \frac{\partial C_j}{\partial x_n} = 0$$

n ECS.

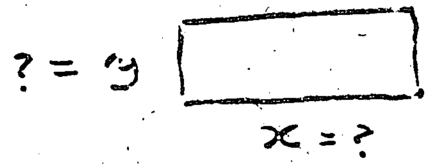
$$\frac{\partial L}{\partial \lambda_1} = -C_1(x_1, \ldots, x_n) = 0$$

$$\vdots$$

$$\frac{\partial L}{\partial \lambda_m} = -C_m(x_1, \ldots, x_n) = 0$$

M RESTRICCIONES

EJEMPLO



AREA MÁXIMA

P = PERIMETRO CONOCIDO

FUNCION OBJETIVO : MAX AREA $= M = xy$

RESTRICCIONES: $P = 2x + 2y$

$$C_1(x, y) = P - 2x - 2y = 0$$

SOLUCIÓN POR DERIVADAS:

$$M = xy = x\left(\frac{P-2x}{2}\right)$$

$$\frac{\partial M}{\partial x} = \frac{P}{2} - 2x = 0, \qquad x = \frac{P}{4} \qquad y = \frac{P}{4}$$

SOLUCION POR MULT. DE LAGRANGE:

$$L = M - \lambda C = xy - \lambda(P - 2x - 2y)$$

$$\begin{cases} \dfrac{\partial L}{\partial x} = y + 2\lambda = 0 \\ \dfrac{\partial L}{\partial y} = x + 2\lambda = 0 \\ \dfrac{\partial L}{\partial \lambda} = -P + 2x + 2y = 0 \end{cases}$$

SOLUCIÓN: $x = \frac{P}{4} \qquad y = \frac{P}{4} \qquad \lambda = -\frac{P}{8}$

METODO CLÁSICO: SE SUSTITUYEN RESTRICCIONES EN FUNCIÓN OBJETIVO PARA DISMINUIR # INCOGNITAS.

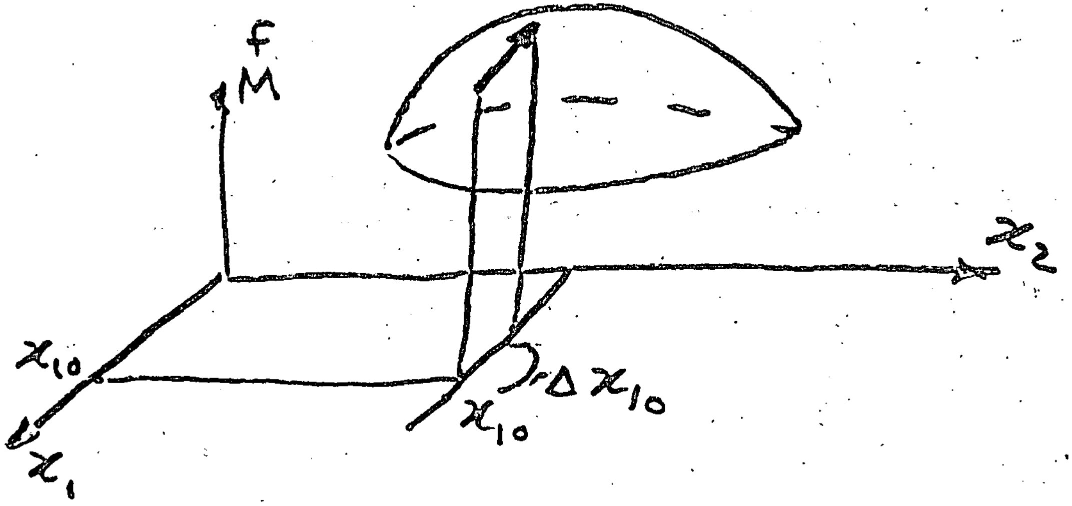
PUEDE SER ALGEBRAICAMENTE COMPLICADO

METODO DE LAGRANGE: NO SE REQUIERE SUSTITUCIÓN.

PUEDE LLEVAR A SIST. DE ECS. COMPLICADO (POR EJ., NO LINEAL)

TÉCNICAS DE GRADIENTE

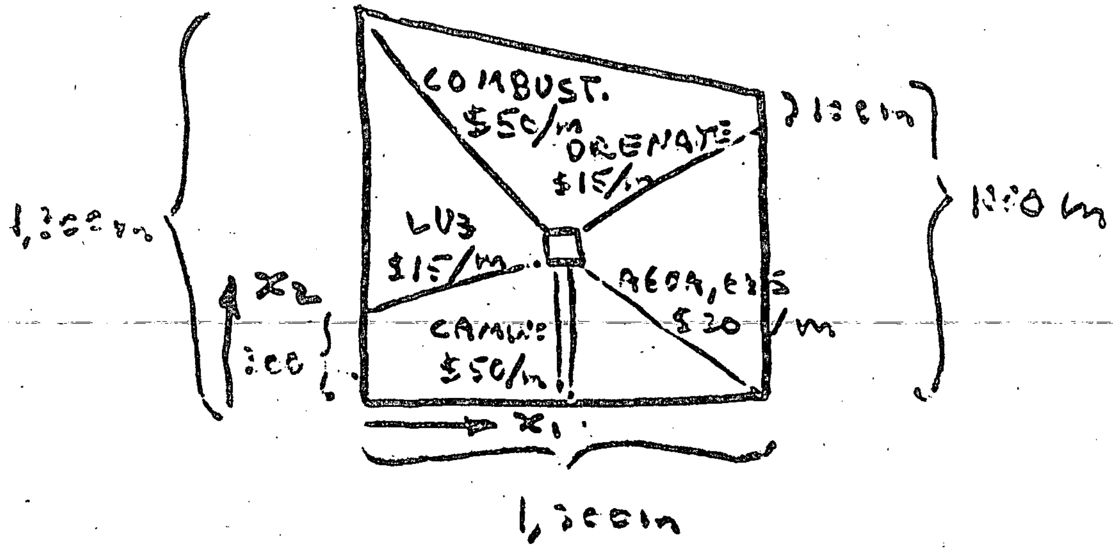
$f(x_1, x_2, \ldots, x_n)$



$$\Delta M = \left.\frac{\partial M}{\partial x_1}\right|_{\underline{x}_0} \Delta x_1 + \left.\frac{\partial M}{\partial x_2}\right|_{\underline{x}_0} \Delta x_2$$

$$\simeq \frac{M(x_{10}+\Delta x_{10}, x_{20}) - M(x_{10}, x_{20})}{\Delta x_1}$$

EJEMPLO: LOCALIZAR ÓPTIMAMENTE LA PLANTA PARA MINIMIZAR COSTOS



$$\text{MIN: COSTO} = 50x_2 ,+ 15\left\{x_1^2 + (x_2-300)^2\right\}^{1/2}$$

$$+ 50\left\{x_1^2 + (1300-x_2)^2\right\}^{1/2} +$$

$$+ 15\left\{(1300-x_1)^2 + (900-x_2)^2\right\}^{1/2}$$

$$+ 20\left\{(1300-x_1)^2 + x_2^2\right\}^{1/2}$$

RESTRICCIONES: NO SALIR DEL PREDIO

INGENIERIA DE CONTROL DIGITAL DE PROCESOS Y SUS APLICACIONES

FOTOCOPIAS DE MICAS

M. EN C. VERONICA CZITROM

NOVIEMBRE, 1978.

CURSO APLICACION DE MINI COMPUTADORAS

PROF. MARCIAL PORTILLA

TEMARIO

1.- INTRODUCCION

2.- ARITMETICA BINARIA

3.- CODIGOS

4.- ALGEBRA BOOLEANA
    4.1 MANIPULACION ALGEBRAICA.
    4.2 MINIMIZACION
    4.3 MAPAS DE KARNAUGH.

5.- COMPUERTAS
    5.1.— DECODIFICADORES
    5.2.- CONTADORES
    5.3.— SUMADORES.

6.- ARQUITECTURA DE UNA MINICOMPUTADORA
    6.1 MAQ. DE VON NEWMAN
    6.2 MEMORIA
    6.3 CPU
    6.4.- LA P.D.P.

QUÉ ES UNA MINICOMPUTADORA?

UNA MINICOMPUTADORA ES UNA "MAXI" DE MENOR COSTO Y MENOR VELOCIDAD, CON LAS SIGUIENTES CARACTERISTICAS:

- CICLO DE MEMORIA (ESCRIBIR, LEER, BORRAR) DE ~ 0.5 μseg → 1μseg
- PALABRA DE 16 BITS (ALGUNAS 8 BITS).
- COSTO MENOR DE $ 50,000
  $ <12,000 CPU y MEMORIA
  $ >$10 → 70,000= EN PERIFERICOS.

EN QUE SE USAN LAS 'MINIS'

VEAMOS ALGUNAS APLICACIONES

TABLE 7-1. *Examples of Minicomputer Applications.*
This list of minicomputer applications – by no means an exhaustive list—was compiled by General Automation, Inc., Anaheim, California.

MANUFACTURING

Production Machines

Controls and monitors automatic and manually operated production machines at a higher sustained efficiency rate. Monitors the actual piece-count production and machine status and signals out-of-limit conditions as they occur. Enables corrective action to be taken immediately.

Package Processing

Controls high-speed packaging equipment and prevents inaccurate operation and breakdowns. Controls the speed of the filling device and the amount of product in the package to be filled, and weighs each package accurately.

Shop-Floor Control Data

Provides an economical data collection system for shop-floor control validation of shipments, monitoring and testing of goods, and the staging of goods for production monitoring of plant facilities; monitoring of the attendance, productivity, and the efficiency of production personnel; and direct dispatching of jobs in a predetermined priority sequence.

Industrial Testing Systems

Monitors and controls complex testing sequences in an industrial testing system. Operations include: product identification, selection of test sequence, calibration check of test equipment, automatic handling of units during testing, source collection and analysis of test data, accept/reject determination of testing units, and printout of test results.

AUTOMOTIVE

Internal Combustion Engines

Acquires data recorded from sensors attached to the internal combustion engine. Measures water temperature, oil temperature, RPM speed, torque, oil pressure, exhaust temperature, manifold pressure, and timing.

Automotive Exhaust Emission

Identifies vehicle for record purposes. Analyzes samples of exhaust gas components during the test cycle. Computes the concentration and/or volume of each monitored gaseous constituent and compiles a test record.

Production Testing

Provides on-line analysis of automotive carburetors in a high-volume production assembly line. Significantly reduces the test time required while providing a greater yield of better quality carburetors.

RUBBER

Rubber Production

Controls in-process inventories and maximizes utilization of machine tools in the production of rubber products. Substantially reduces the time and personnel required to summarize and review the stripcharts of production activity and the manually produced shift-end reports.

ELECTRONICS

Peripheral Test

Computer interfacing and multiunit control of input/output peripherals for large-scale computer installation. I/O units include a line printer, card reader, card punch, and magnetic tapes.

Coil Winding Production

Control of automatic winding machines for the mass production of small coils for magnetic-latching reed switches. Eight winding patterns are stored in the computer's memory. All eight of the patterns can be run on up to 16 machines. Coils are wound on plastic inserts in a steel plate on an 8 in by 8 in matrix. Inserts extend from both plate surfaces and provide 64 winding cores on each side.

PC Board Production

Used for the automatic development, formatting, and conversion of instruction programs for numerically controlled printed circuit board drilling machines. Complete patterns are stored in the computer memory to perform step-and-repeat operations accurately. Frequently repeated patterns for standard devices are entered into memory that will completely define all points in the pattern after only two points have been located by the operator.

Electronic Testing

Tests each of electric/electronic component in a high volume production line. Executes tests at a predetermined maximum rate; variations in manual operator rates are eliminated.

TABLE 7-1. Examples of Minicomputer Applications (*Continued*).

Testing and Analysis of Circuits

Controls circuit testing and analysis systems. Coordinates the testing operations, specifications, signals, and the test sequence at electronic speeds, including: continuity, impedance, test stimuli, and measurement of the circuit output.

AEROSPACE

Aircraft Wing Production

Controls and monitors automatic riveting machines for manufacturing aircraft wings. Riveting patterns are stored in the computer's memory to perform step-and-repeat operations accurately and quickly.

Fatigue Testing

Acquires, processes, and analyzes fatigue stress data for a variety of metals as well as bonded joined materials. Prints out data for corrective action, thereby preventing potential accidents and malfunction due to fatigue stress.

METALS AND WOODWORKING

Steelmaking

Controls and operates steel furnaces, and produces the metal in exact accordance with preset specifications. Calculates oxygen requirements, alloy additions, and power requirements.

Metal Analyses

Monitors and controls optical emission and x-ray spectrometers widely used in the metals industry for high-speed determination of the chemical composition of metal.

Tensile Testing

Provides quality control, production techniques evaluation, product classification, and customer certification. Calculates the product's strength and other characteristics, records and calculates vital material properties, and measures and computes tensile strength.

Transfer Line

Monitors and controls transfer lines producing high production parts, and consisting of many machining stations mechanically connected by work-piece transfer mechanisms and closely interlocked with electrical controls. Receives input from operator or sensors, then concurrently checks the operating condition of line-mounted controls, takes protective action when required, prints out a report of the malfunction, and generates production reports.

MATERIALS HANDLING

Automated Warehousing

Provides optimum space utilization, significant manpower savings, and high turnaround for material requests in an automated "high cube" vertical storage warehouse. Keeps track of numerous units of merchandise and optimizes the movement of stacker cranes. Provides real-time inventory control and warehousing applications to be integrated into a plant-wide information system.

Material-Handling System

Controls complex material-handling system including storage and retrieval equipment. Processes orders, prepares shipping documents and invoices, provides operator guidance, maintains accurate inventories, and provides direct control of transport facilities and stacker units.

PAPER

Paper-Mill Production

Regulates the average basis weight and moisture variables in each paper grade. Manipulates the steam flow valve, adjusts the stock valve to the regulated basis weight, and monitors and/or controls total flow and digital filtering of instruments' signals.

TRANSPORTATION

Railroad

Counts and identifies railroad cars transporting materials and goods. Provides accurate weighing of each railroad car as it passes through the scale weighing system without stopping.

Automobile

Provides centralized computer control of an electronic traffic control system. Records, analyzes, and prints out traffic count and flow at various time periods, accident control and notification, control of traffic lights, etc.

Airline

Monitors and displays airline flight arrivals and departures. Provides accurate up-to-date information on air flights to numerous air terminals and airline offices.

COMMERCIAL

Banking

Reads, analyzes, and tabulates check data and monetary transactions in real-time applications at branch offices Central computer at the main-bank processing center provides fast analyses and totals for management control.

ACCOUNTING

Acquires, processes, and prints out man-hours on-the-job for job/function/time evaluation.

ENVIRONMENTAL CONTROL

Controls and monitors environmental conditions throughout a large office building. Evaluates and monitors temperature, humidity, pollen, airborne dirt and irritants, etc., within the controlled environment.

PRINTING

PRINTING PRESSES

Monitors and controls the operation of large multicolor printing presses. Preset ink fountain and compensator positions are maintained during running, taking into consideration temperature, humidity, ink absorption, etc.

TYPESETTING

Automation of formatting and typesetting in a high-volume newspaper operation.

RETAIL

MERCHANDISE

Automates check-out-stand operations in large retail stores. Computes transactions for accounting and inventory control.

FOOD

Computes transactions quickly and accurately in a fast-food service. Maintains "instant" inventory control throughout the food chain's operation.

DISPLAY SYSTEMS

EXTERNAL

Monitors and controls scoreboard displays located in sports stadiums providing score information, animation displays, and audience messages. Stores repetitive messages and animation programs of all types.

INTERNAL

Provides generation and control of architectural display, light, sound, and temperature effects of remote-site data from microphones and CCTV sets.

COMMUNICATIONS

SYNCHRONOUS DATA EXCHANGE

The synchronous exchange of information on medium-speed to high-speed data links involves multiprocessor or multidevice complexes. Remote computers perform process control or data-gathering distributed tasks, and provide a supervisory computer with information and data for process and/or management decisions and commands.

TELEMETRY DATA ACQUISITION

Monitors remotely the physical status of objects, animals, people, or the environment in space flights. Evaluates incoming data for relative importance and validity. Isolates useful data from "noise" and other spurious signals.

BROADCASTING

Provides automatic timing control of audio/visual processes for radio and television stationbreak advertising. Maintains time-of-day synchronization with the national networks.

EDUCATION

Provides audio/visual control of the teacher's presentation and real-time data acquisition. Processes and tabulates student responses.

TELEVISION

Provides real-time data acquisition and processing of audience-viewer responses in audience participation shows. Tabulates and prints out responses for "instant" results while still on the air.

POWER

SUBSTATION MONITORING AND CONTROL

Monitors and controls high voltage and extra high voltage substations from centralized dispatching offices.

PLANT POWER SYSTEM

Assures proper distribution of available electricity, gas, or steam in utility systems. Monitors powerhouse facilities, schedules distribution of the energy, and produces operating distribution logs.

LABORATORY/MEDICAL

PHYSIOLOGICAL MONITORING

Monitors the patient's disorder, including his blood pressure, respiration, temperature, appearance, urine output, blood and fluid loss, fluid and electrolyte intake, blood chemistry, weight, and electrocardiogram.

SPECTROMETER OPERATION

Provides the computational, functional, and communication capabilities for optimum use of a spectrometer used in industrial applications. This includes checking, sequencing, calculating percentage composition of each element, outputting results, and calculating interelement effect.

GAS CHROMATOGRAPHS

Accurately measures chromatograph signal output and controls instrument functions amenable to external control, such as temperature programming, column switching.

AMINO ACID ANALYSES

Controls instrumentation obtaining amino acid analysis data. Evaluates, acquires, and processes data for meaningful information and displays information for laboratory personnel.

CHEMICAL

AMMONIA AND ETHYLENE PROCESSING

Identifies mechanical problems in large compressor, operation of clearance pockets, compressor speed, power consumption, vibration, discharge temperatures, pressures, suction flow, and gas compositions.

PLASTICS

Acquires, processes, and provides stress analysis for a wide range of plastic materials. Calculates the product's strength, records and calculates material properties, measures and computes elasticity and hardness.

EXPLOSIVES

Provides data acquisition and analysis of explosive shock waves. Measures and calculates explosive force and duration.

DYES

Monitors and controls the processing of dyes used in the textile industry. Provides accurate processing of color blending and matching to predetermined values.

PETROLEUM

GAS TRANSMISSION AND DISTRIBUTION

Data is gathered, measured in the field, and transmitted to the System 18/30.

OIL FIELD

Provides on-site acquisition and processing of data received from drilling rigs on depth, density, etc.

ES LA COMPUTADORA DIGITAL, TAL VEZ LA CARACTERISTICA MAS IMPORTANTE DE LA SEGUNDA MITAD DEL SIGLO XX

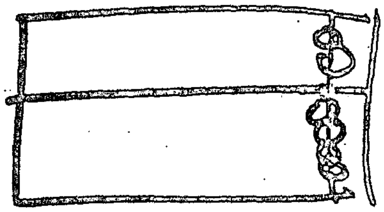
LIMITANTES ?

HISTORIA DE LA COMPUTADORA.

- EL ABACO CHINO (
- CALCULADORA DE BLAS PASCAL.
- LEIBNITZ LA MEJORA
- MAQUINA DE CHARLES BABAGES

SISTEMA DIGITAL .- ES AQUEL EN EL CUAL LA INFORMACION ESTA REPRESENTADA POR CANTIDADES DISCRETAS

= SISTEMA BINARIO
= " OCTAL
= " DECIMAL
= " HEXADECIMAL



APARATOS DIGITALES

= SEÑALES DISCRETAS
- ODOMETRO
- VOLMETRO, AMP.
- CONV. ANALOGICO DIGITAL

SISTEMAS DE NUMERACION.

= EL ROMANO = ¡QUE FEO!

• OTROS SISTEMAS.

REPRESENTACION MATEMATICA DE UN NUMERO

$$\boxed{N = a_0 b^0 + a_1 b^1 + a_2 b^2 + \ldots\ldots a_n b^{\textcircled{n}}}$$

DONDE:

$n = 0, 1, \ldots\ldots \infty$

b = BASE DEL SISTEMA NUMERICO DE QUE SE TRATE

a = NUMERO DE DIGITOS LOS CUALES TOMAN VALORES ENTRE 0 y $b-1$

i.e.

$$8432 = 2 \times 10^0 + 3 \times 10^1 + 4 \times 10^2 + 8 \times 10^3$$

$$= 2 + 30 + 400 + 8000 = 8432.$$

DONDE:

$b = 10$

$a_0 = 2$

$a_1 = 3$

$a_2 = 4$

$a_3 = 8$

LOS NUMEROS (EN GENERAL) SON ESCRITOS CON EL DIGITO MENOS SIGNIFICATIVO A LA DERECHA Y EL MAS SIGNIFICATIVO A LA IZQUIERDA.

VEAMOS A CONTINUACION COMO CAMBIAR DE UNA BASE A OTRA.

$$N = a_0 b^0 + a_1 b^1 + a_2 b^2 + \dots a_n b^n$$

i.e.

$$1001_{(2)} = 1\times 2^3 + 0\times 2^2 + 0\times 2^1 + 1\times 2^0 = 8+1 = 9_{10}$$

$$1001_{(8)} = 1\times 8^3 + 0\times 8^2 + 0\times 8^1 + 1\times 8^0 = 512+1 = 513_{10}$$

$$1001_{10} = 1\times 10^3 + 0\times 10^2 + 0\times 10^1 + 1\times 10^0 = 1000+1 = 1001$$

$$1001_{16} = 1\times 16^3 + 0\times 16^3 + 0\times 16^1 + 1\times 16^0 = 4096+1 =$$

$$4097$$

COMPLICANDO LO ANTERIOR, AL INTRODUCIR NUMEROS FRACCIONARIOS NUESTRA FORMULA QUEDA:

$$N = a_n b^n + a_{n-1} b^{n-1} + \dots a_1 b + a_0 b^0 \boxed{+ a_{-1} b^{-1} + a_{-2} b^{-2} \dots}$$

$N = 1101.011_2$ EN DECIMAL SERA:

$$N = 1\times 2^3 + 1\times 2^2 + 0\times 2^1 + 1\times 2^0 + 0\times 2^{-1} + 1\times 2^{-2} + 1\times 2^{-3}$$

$$N = 8 + 4 + 1 + \frac{1}{4} + \frac{1}{8} = 13\ 3/8 = 13.37$$

$N = 1101.011_{16} = xxx_{10}$?

$N = 1x16^{3} + 1x16^{2} + 0x16^{1} + 1x16^{0} + 0x16^{-1} + 1x16^{-2} + 1x16^{-3}$

$N = 4096 + 256 + 1 + \frac{1}{256} + \frac{1}{4096} = 4353.00415$

HASTA AHORA SABEMOS CONVERTIR UN NUMERO DE CUALQUIER BASE A UN NUMERO DECIMAL. VEAMOS A CONTINUACION EL PROCESO INVERSO

DADO 432_{10} ENCONTRAR SU EQUIVALENTE BINARIO.

	432		
÷ 2	216	0	RESIDUO
	108	0	
	54	0	
	27	0	
	13	1	
	6	1	
	3	0	
	1	1	
	0	1	

↑ SE LEE ASI

EL METODO ES ANALOGO EN CUALQUIER BASE POR EJEMPLO SEA EL NUMERO 432_{10} EL CUAL QUEREMOS CONVERTIR A OCTAL.

$$\div 8 \quad \begin{array}{r|l} 432 & \\ 54 & 0 \\ 6 & 6 \\ 0 & 6 \end{array} \qquad 432_{10} = 660_8$$

$$6 \times 8^2 + 6 \times 8^1 + 0 \times 8^0 = 384 + 48 = 432.$$

QUE PASA SI USAMOS FRACCIONES.

SEA $N = 2.4375_{10}$

$10_2 = 2_{10}$

SE LEE ASI

$$X2 \quad \begin{array}{r|l} .4375 & \\ .8750 & 0 \\ \hline \textcircled{1}.750 & 1 \\ 1.50 & 1 \\ 1.00 & 1 \end{array}$$

$$2.4375_{10} = 10.0111_2$$

EJEMPLO TRATEMOS AHORA DE CONVERTIR

2.10.. = XX vvvv ?

OPERACIONES BINARIAS.

SUMA.

$0+0=0$

$0+1=1$

$1+0=1$

$1+1=0$ y LLEVAMOS UNO

EJEMPLOS

$$\begin{array}{r} 4 \\ +\ 7 \\ \hline 11 \end{array} \qquad \begin{array}{r} 1\,100 \\ 111 \\ \hline 1011 \end{array}$$

$$\begin{array}{r} 20 \\ +\ 31 \\ \hline 51 \end{array} \qquad \begin{array}{r} 1\,10100 \\ 11111 \\ \hline 110011 \end{array}$$

PARA NUMEROS FRACCIONARIOS:

$$\begin{array}{r} 3.375 \\ 5.750 \\ \hline 9.125 \end{array} \qquad = \begin{array}{r} 11.011 \\ +101.110 \\ \hline 1001\ 001 \end{array}$$

RESTA

$0-0=0$

$1-0=1$

$1-1=0$

$0-1=1$ y DEBEMOS 1

$$\begin{array}{r} 9 \\ -5 \\ \hline 4 \end{array} \qquad \begin{array}{r} 1001 \\ -101 \\ \hline 100 \end{array} \qquad \begin{array}{r} 16 \\ -3 \\ \hline 13 \end{array} \qquad \begin{array}{r} 10000 \\ -11 \\ \hline 1101 \end{array}$$

DADO QUE EXISTE PROBLEMA PARA REPRESENTAR NUMEROS NEGATIVOS A CONTINUACION SE MUESTRA A UNA TABLA EN LA CUAL SE TIENEN VARIAS REPRESENTACIONES DE LOS NUMEROS NEGATIVOS

——— O ———

RESTA COMPLEMENTADA.

COMPLEMENTO DE NUEVE
〃 〃 10

COMPLEMENTO DE UNO
〃 〃 DOS.

Integer coded	Sign plus 8421 BCD magnitude									Sign plus ten's complement (8421 BCD)								
	s	d_{80}	d_{40}	d_{20}	d_{10}	d_8	d_4	d_2	d_1	s	d_{80}	d_{40}	d_{20}	d_{10}	d_8	d_4	d_2	d_1
+99	0	1	0	0	1	1	0	0	1	0	1	0	0	1	1	0	0	1
⋮					⋮									⋮				
+11	0	0	0	0	1	0	0	0	1	0	0	0	0	1	0	0	0	1
+10	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0
+9	0	0	0	0	0	1	0	0	1	0	0	0	0	0	1	0	0	1
+8	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0
+7	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	1	1	1
+6	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	1	1	0
+5	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	1	0	1
+4	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0
+3	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	1	1
+2	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0
+1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
−1	1	0	0	0	0	0	0	0	1	1	1	0	0	1	1	0	0	1
−2	1	0	0	0	0	0	0	1	0	1	1	0	0	1	1	0	0	0
−3	1	0	0	0	0	0	0	1	1	1	1	0	0	1	0	1	1	1
−4	1	0	0	0	0	0	1	0	0	1	1	0	0	1	0	1	1	0
−5	1	0	0	0	0	0	1	0	1	1	1	0	0	1	0	1	0	1
−6	1	0	0	0	0	0	1	1	0	1	1	0	0	1	0	1	0	0
−7	1	0	0	0	0	0	1	1	1	1	1	0	0	1	0	0	1	1
−8	1	0	0	0	0	1	0	0	0	1	1	0	0	1	0	0	1	0
−9	1	0	0	0	0	1	0	0	1	1	1	0	0	1	0	0	0	1
−10	1	0	0	0	1	0	0	0	0	1	1	0	0	1	0	0	0	0
−11	1	0	0	0	1	0	0	0	1	1	1	0	0	0	1	0	0	1
⋮					⋮									⋮				
−99	1	1	0	0	1	1	0	0	1	1	0	0	0	0	0	0	0	1

Fig. 2-6 Signed number codes based on 8421 BCD code.

11110110
+1
1110111
−9
11111110
+1
11111111
100 1001
0110

This brief description of a few codes and their properties will provide the background needed for discussing many related matters throughout the book. In addition, since the subject of codes and code properties is a broad one, it will arise again in quite a few sections of the book where the technique being discussed is intimately related to some specific code.

2-5 LOGICAL CONNECTIVES

In order to express Boolean variables as functions of other Boolean variables, we need to define several logical connectives. The role which these serve in Boolean algebra is analogous to the role served by connectives such as addition and multiplication for ordinary algebra.

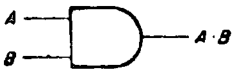
Consider first the AND connective as it relates to two Boolean variables A and B. Figure 2-7*a* illustrates two ways to express the algebraic ANDing of A and B. The *truth table* in Fig. 2-7*b* has nothing to do with truth or falsehood but rather is simply a listing of the four possible combinations A and B and the corresponding values of $A \cdot B$. Note that $A \cdot B = 1$ if and only if $A = 1$ and $B = 1$. The commonly used graphic symbol† is shown in Fig. 2-7*c*. For three (or more) Boolean variables, the function $A \cdot B \cdot C = 1$ if and only if $A = 1$ *and* $B = 1$ *and* $C = 1$.

$A \cdot B$, AB

(*a*)

A	B	$A \cdot B$
0	0	0
0	1	0
1	0	0
1	1	1

(*b*)



(*c*)

Fig. 2-7 The AND connective. (*a*) Algebraic representations; (*b*) truth table; (*c*) graphic symbol.

The OR connective or, more properly the INCLUSIVE-OR connective, is illustrated in Fig. 2-8. Note that $A + B = 1$ if $A = 1$ or if $B = 1$ or both. It is the inclusion of this *or both* condition which leads to the name INCLUSIVE-OR. For three (or more) variables $A + B + C = 1$ if and only if any one or more of the variables A, B, or C are equal to 1.

The EXCLUSIVE-OR connective, illustrated in Fig. 2-9, excludes the *or both* case described in the last paragraph. Consequently, for two variables, the function $A \oplus B = 1$ if and only if either input = 1 while the other input = 0. The extension to three variables is not immediately obvious. However, it can be derived by considering the sequence of operations

$$(A \oplus B) \oplus C$$

The first column of the truth table in Fig. 2-9*d* illustrates the function $A \oplus B$, while the second column is an EXCLUSIVE-ORing between this and the variable

$A + B$

(*a*)

A	B	$A + B$
0	0	0
0	1	1
1	0	1
1	1	1

(*b*)

(*c*)

Fig. 2-8 The OR connective. (*a*) Algebraic representation; (*b*) truth table; (*c*) graphic symbol.

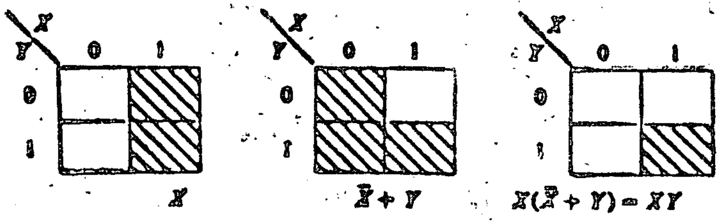
---

†See *Military Standard–Graphic Symbols for Logic Diagrams*, MIL-STD-806B, February 26, 1962. This may be obtained by writing Naval Publications and Forms Center, 5801 Tabor Ave., Philadelphia, Pa. 19120.

/ SUMAR -
/ RESTAR (?)
/ MULTIPLICAR.
— DIVIDIR —— RESTAS y SUMAS

ALGEBRA BOOLEANA.

row related to Y. Because the term X is AND'ed with the function $\bar{X} + Y$, this implies the total area where both X and $\bar{X} + Y$ are true at the same time. This only occurs at the cell represented by the term $X \cdot Y$, as the theorem demands. Further proofs of the theorems will be left to the student.



X $\qquad$ $\bar{X} + Y$ $\qquad$ $X(\bar{X} + Y) = XY$

The following theorems are presented without proof, because the student is assumed to be somewhat familiar with the material or can review some of the references on the subject.

Theorem 10a $\quad \overline{X + Y + Z + \cdots} = \bar{X} \cdot \bar{Y} \cdot \bar{Z} \cdots.$

Theorem 10b $\quad \overline{X \cdot Y \cdot Z} = \bar{X} + \bar{Y} + \bar{Z} + \cdots.$

This is known as De Morgan's theorem, whereby the complement of a function is obtained by changing all "·" to "+" and all "+" to "·", complementing all uncomplemented terms, and uncomplementing all complemented terms.

For example, the following functions are complemented thus:

EXAMPLE 2.15. $\quad \overline{A + B \cdot C} = \bar{A} \cdot \bar{B} + \bar{C}.$

EXAMPLE 2.16. $\quad \overline{\bar{A}(BC + \bar{D})} = A + (\bar{B} + \bar{C})D.$

EXAMPLE 2.17. $\quad \overline{ABC + D(\bar{A} + B)} = (\bar{A} + \bar{B} + \bar{C})(\bar{D} + A\bar{B}).$

Theorem 11a $\quad XY + \bar{X}Y = Y.$

Theorem 11b $\quad (X + Y)(\bar{X} + Y) = Y.$

Summary of Theorems

Theorem 1a $\quad X + 0 = X.$

Theorem 1b $\quad X \cdot 1 = X.$

Theorem 2a $\quad X + 1 = 1.$

Theorem 2b $\quad X \cdot 0 = 0.$

$\bar{X} + X\bar{Y} = \bar{X} + \bar{Y}$

Theorem 3a $\quad X + X = X.$

Theorem 3b $\quad X \cdot X = X.$

Theorem 4a $\quad (\bar{X}) = \bar{X}.$

Theorem 4b $\quad \overline{(\bar{X})} = X.$

Theorem 5a $\quad X + \bar{X} = 1.$

Theorem 5b $\quad X \cdot \bar{X} = 0.$

Theorem 6a $\quad X(X + Y) = X.$

Theorem 6b $\quad X + XY = X.$

Theorem 7a $\quad X(\bar{X} + Y) = XY.$

Theorem 7b $\quad X + \bar{X}Y = X + Y.$ $\sim \bar{X} + XY = \bar{X} + Y$

Theorem 8a $\quad XY + XZ = X(Y + Z).$

Theorem 8b $\quad (X + Y)(X + Z) = X + YZ.$

Theorem 9a $\quad XY + YZ + \bar{X}Z = XY + \bar{X}Z.$

Theorem 9b $\quad (X + Y)(\bar{X} + Z) = XZ + \bar{X}Y.$

Theorem 10a $\quad \overline{X + Y + Z + \cdots} = \bar{X} \cdot \bar{Y} \cdot \bar{Z} \cdots.$

Theorem 10b $\quad \overline{X \cdot Y \cdot Z \cdots} = \bar{X} + \bar{Y} + \bar{Z} + \cdots.$

Theorem 11a $\quad XY + \bar{X}Y = Y.$

Theorem 11b $\quad (X + Y)(\bar{X} + Y) = Y.$

$A \circ B = \overline{\bar{A} + \bar{B}}$ $\sim$ $\overline{\bar{A}\bar{B}} = A + B$

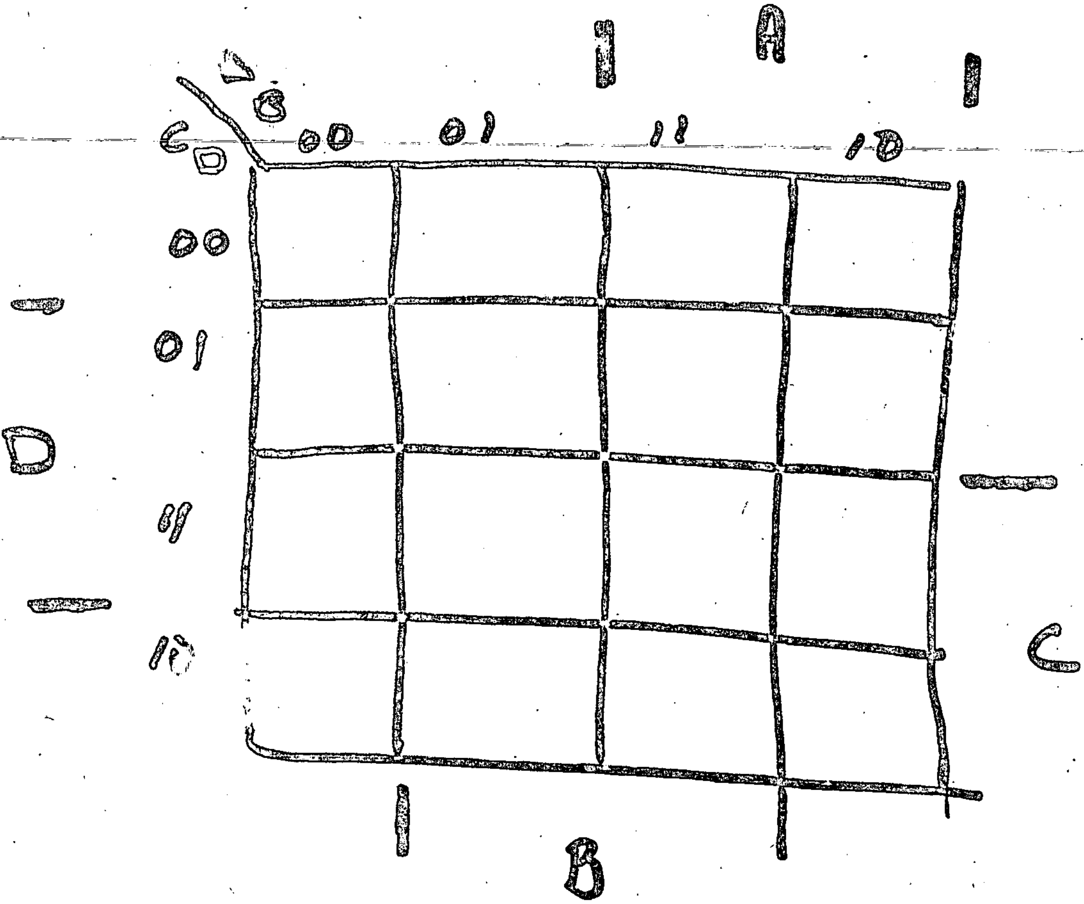
2.4 STANDARD FORMS

There is a lot of reference made in texts to the standard sum form or the standard product form of an expression. These statements refer to the two basic elementary forms in which a Boolean algebraic expression can be written. That is, the expression $AB + BC$ is an OR'ing of the two AND'ed terms. The AND'ed terms, AB and BC, are called product terms. The OR'ing of things is called a sum of things. Thus the above expression is a sum of products. This sum of products is called a standard sum form.

For example, $AB + C(\bar{A} + BD)$ is a hybrid function and only after expanding or multiplying through by C can it become a sum of products, $AB + \bar{A}C + BCD$.

In a like manner the standard product form is a product of sums.

MAPAS DE KARNAUGH



4. Write a minimal expression for the following maps:

A D   A d

C \ AB	00	01	11	10
0	1	0	1	1
1	0	1	1	1

(a)

$= A + \bar{C}\bar{B} + BC$

$= A + \bar{A}\bar{B}\bar{C} + \bar{A}BC$

C \ AB	00	01	11	10
0	1	0	0	0
1	1	1	0	1

(b)

$\bar{A}\bar{B} + AC + \bar{B}C$

C \ AB	00	01	11	10
0	1	1	1	0
1	0	0	1	1

(c)

CD \ AB	00	01	11	10
00	0	0	1	0
01	0	1	1	1
11	0	0	0	1
10	0	0	0	1

(d)

$B\bar{C}D + AB\bar{C} + A\bar{C}D + AC\bar{B}$

CD \ AB	00	01	11	10
00	0	1	1	0
01	0	1	1	0
11	0	1	1	0
10	0	1	1	0

(e)

$= B_{2,3,4}$

CD \ AB	00	01	11	10
00	1	1	0	[illegible]
01	0	0	[illegible]	0
11	0	0	[illegible]	0
10	[illegible]	0	[illegible]	[illegible]

(f)

$\bar{A}\bar{C}\bar{D} + AC\bar{D} + \bar{B}\bar{D}$

CD \ AB	00	01	11	10
00	0	0	1	0
01	1	1	ϕ	ϕ
11	0	ϕ	ϕ	ϕ
10	0	0	1	0

(g)

CD \ AB	00	01	11	10
00	1	0	1	ϕ
01	1	0	1	0
11	ϕ	ϕ	0	0
10	ϕ	0	0	ϕ

(h)

CD \ AB	00	01	11	10
00	0	ϕ	1	0
01	0	0	1	ϕ
11	0	ϕ	1	ϕ
10	ϕ	0	0	0

(i)

CD \ AB	00	01	11	10
00	0	1	1	0
01	0	1	1	0
11	0	0	0	0
10	0	1	1	0

(j)

CD \ AB	00	01	11	10
00	1	0	0	1
01	1	0	0	0
11	1	0	0	0
10	1	0	0	1

(k)

CD \ AB	00	01	11	10
00	0	1	1	0
01	1	0	0	1
11	1	0	0	1
10	0	1	1	0

(l)

CD \ AB	00	01	11	10
00	0	0	0	1
01	0	1	1	1
11	1	1	0	0
10	1	0	0	0

(m)

CD \ AB	00	01	11	10
00	0	0	0	0
01	1	1	1	1
11	0	0	0	0
10	1	1	1	1

(n)

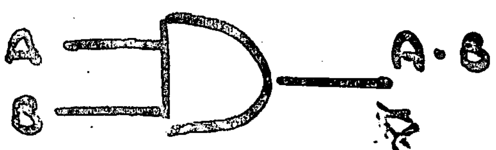
CD \ AB	00	01	11	10
00	0	1	ϕ	0
01	0	1	ϕ	0
11	0	ϕ	1	0
10	0	0	1	ϕ

(o)

5volts

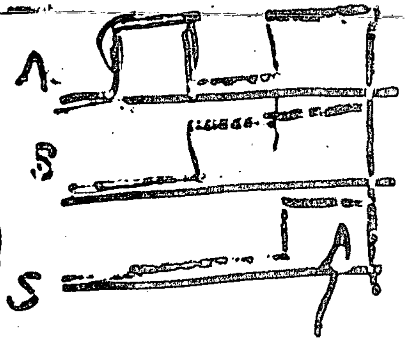
COMPUERTAS

AND



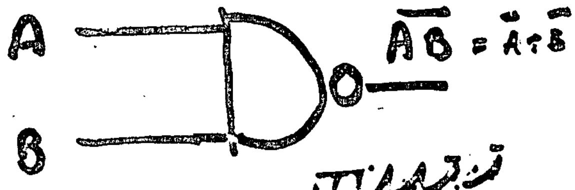
Tabl.

B \ A	0	1
0	0	0
1	0	1



v10

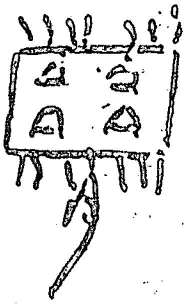
NAND (LA MAS COMUN)



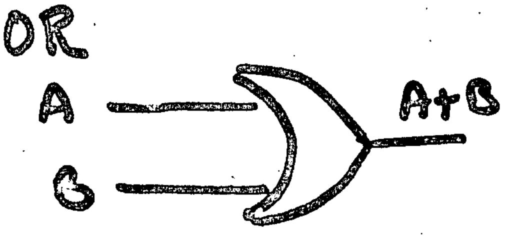
TTL

B \ A	0	1
0	1	1
1	1	0

7400



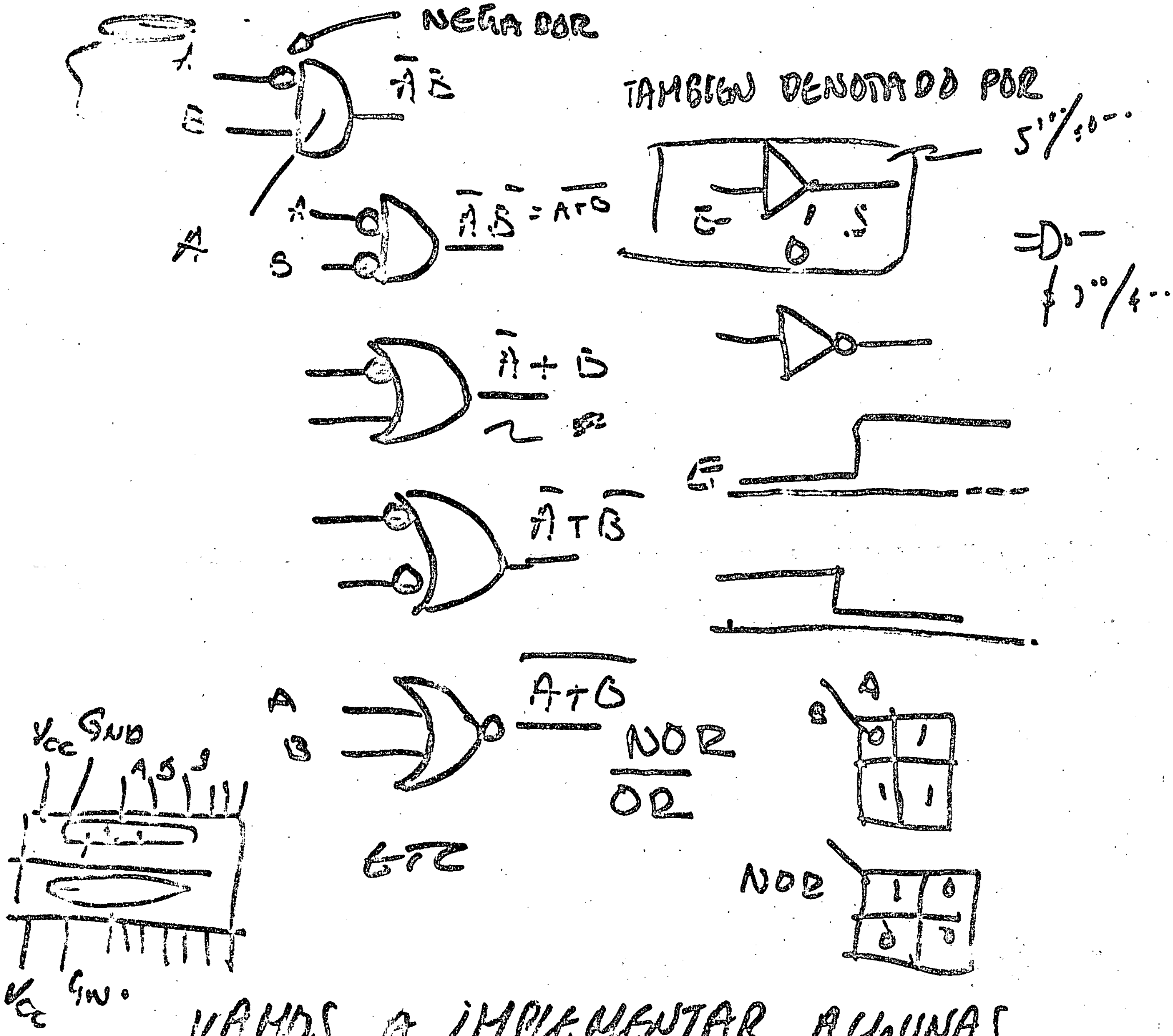
OR



B \ A	0	1
0	0	1
1	1	1

SUMA

SE PUEDE IMPLEMENTAR CUALQUIER TIPO DE COMPUERTA NEGANDO SU ENTRADA (S) O SALIDA.



VAMOS A IMPLEMENTAR ALGUNAS FUNCIONES UTILIZANDO COMPUERTAS.

$$s_i = a_i b_i c_{i-1} + a_i \bar{b}_i c_{i-1} + \bar{a}_i b_i \bar{c}_{i-1} + a_i \bar{b}_i \bar{c}_{i-1}$$

y el CARRY SERA

$$C_i = a_i b_i + a_i C_{i-1} + b_i C_{i-1}$$

EN MAPAS K ESTO LUCE ASI

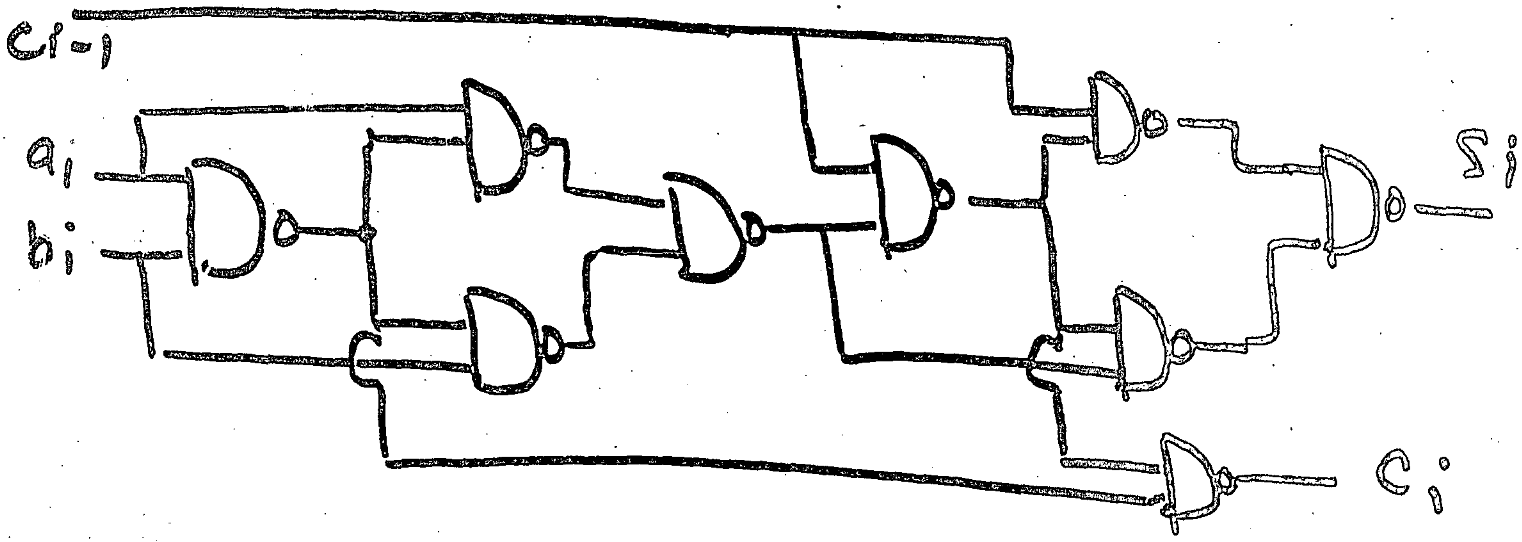
c_{i-1} \ $a_i b_i$				
	0	1	0	1
	1	0	1	0

SUMA.

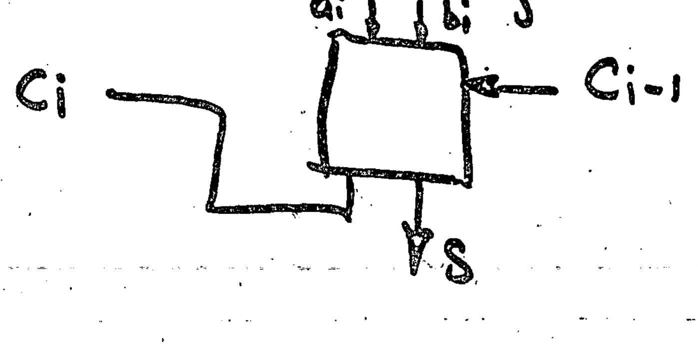
c_i \ $a_i b_i$				
	0	0	1	0
	0	1	1	1

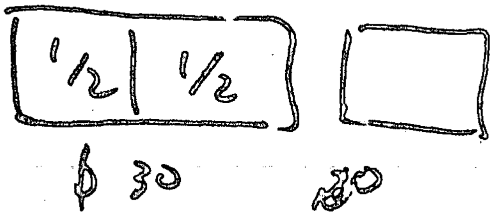
CARRY

ESTE SUMADOR ES IMPLEMENTO usualmente utilizando solamente COMPUERTAS NAND.



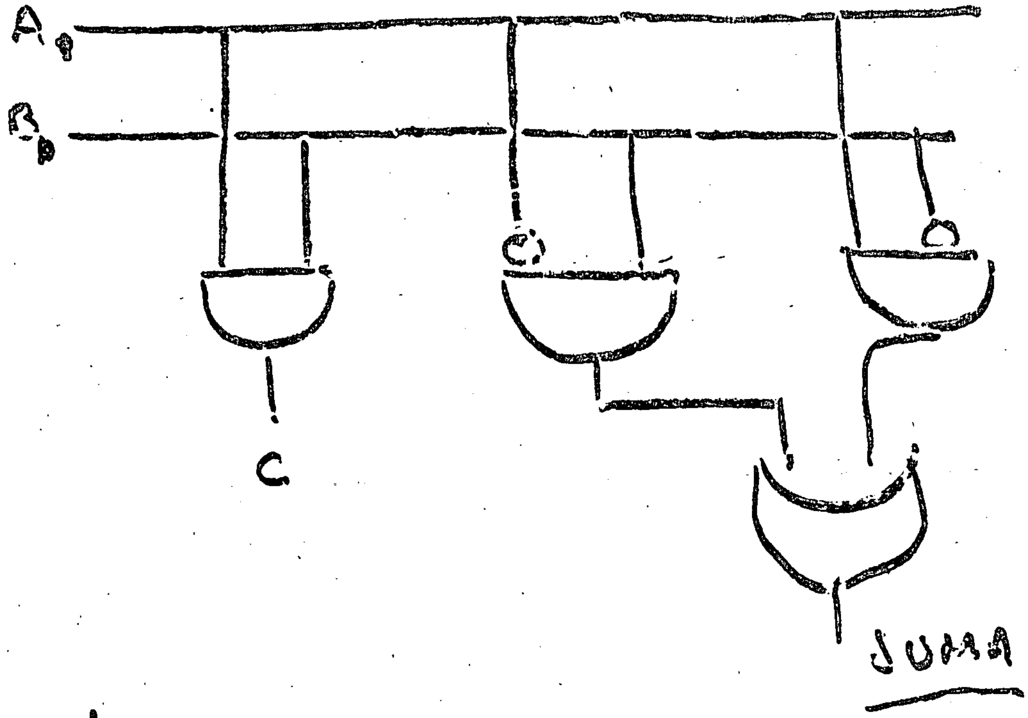
LO CUAL LOS SIMBOLIZAREMOS CON $\frac{1}{2}$ SUMADOR





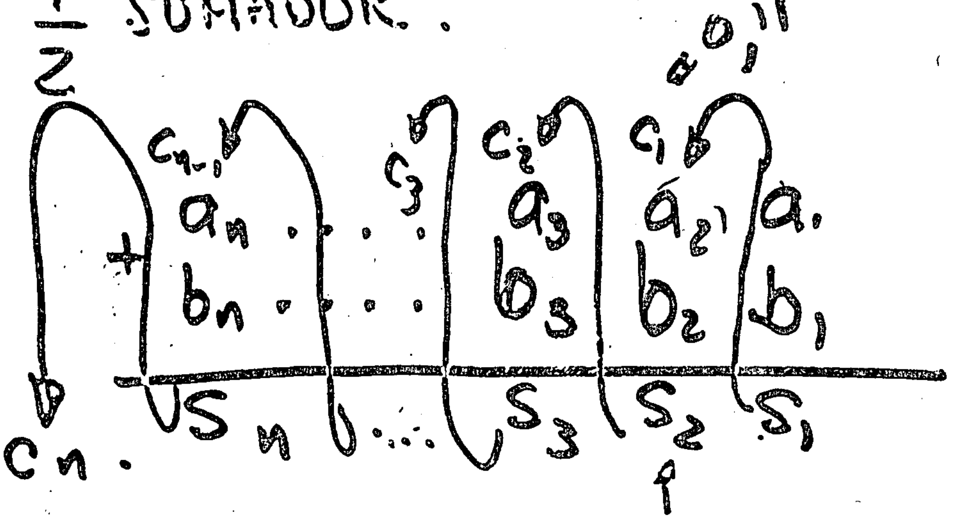
IMPLEMENTACION DE UN SUMADOR.

0+0 = 0
0+1 = 1
1+0 = 1
1+1 = 0 y llevamos uno. (CARRY "C")



$a_i b_i \bar{C}_i$

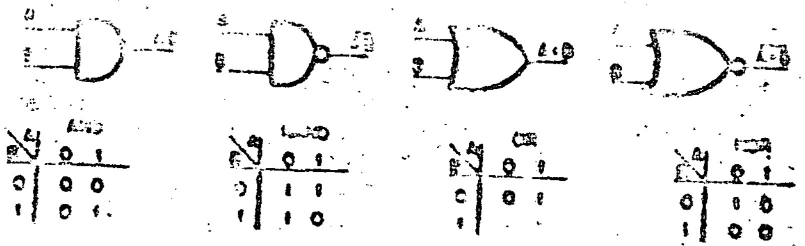
$\frac{1}{2}$ SUMADOR.



111
1011
111
10010

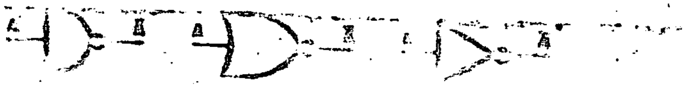
1. Basic Gates and Truth Tables. The basic logic operations [illegible] variables. Each gate function is defined completely by a truth table [illegible] combinations of inputs.

NAND and NOR gates also serve as [illegible] (and vice versa).

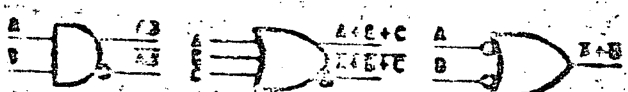


NOTE: In some types of logic, gate outputs can be OR'ed together.

2. Inverters. NAND and NOR gates also serve as logic inverters for complementing a single input. We use the following inverter symbols:



Some gates have two complementary outputs, and most logic modules provide gates with inverting inputs:



3. The Rules of Boolean Algebra. When we proceed to combine simple logic functions into more complicated functions of more variables, we find that the combinations satisfy the following rules of Boolean algebra. These rules are established by a simple examination of the basic truth tables. The rules may be applied to simplify logic circuits (logic optimization).

$$\left.\begin{array}{l} A + B = B + A \\ AB = BA \end{array}\right\}$$ [illegible] LAWS

$$\left.\begin{array}{l} A + (B + C) = (A + B) + C \\ A(BC) = (AB)C \end{array}\right\}$$ (ASSOCIATIVE LAWS)

$$\left.\begin{array}{l} A(B + C) = AB + AC \\ A + BC = (A + B)(A + C) \end{array}\right\}$$ (DISTRIBUTIVE LAWS)

$$A + A = AA = A$$ [illegible]

$$A + B = B \text{ if and only if } AB = A$$ [illegible]

$$A + 0 = A \qquad A1 = A$$

$$A0 = 0 \qquad A + 1 = 1$$

$$A(A + B) = A + AB = A$$ [illegible]

$$\left.\begin{array}{l} \overline{(A + B)} = \bar{A}\bar{B} \\ \overline{(AB)} = \bar{A} + \bar{B} \end{array}\right\}$$ (DUALITY, OR DE MORGAN'S LAWS)

$$\bar{\bar{A}} = A \qquad \bar{1} = 0 \qquad \bar{0} = 1$$

$$A + \bar{A}B = A + B \qquad AB + \bar{A}C + BC = \bar{A}C + BC$$

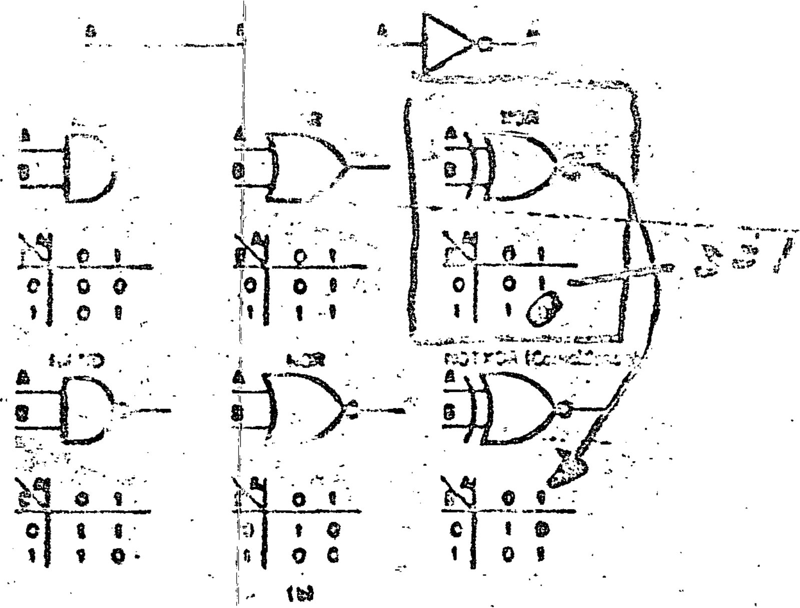


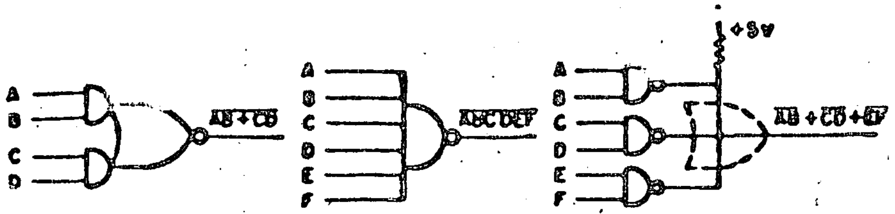
Fig. 1-6. Figure 1-6a shows Boolean functions of a single input. Figure 1-6b defines the most important functions of two inputs by their truth tables and shows commonly used symbols for the corresponding logic gates.

Boolean functions with single gates. In particular, AND gates and inverters alone, NAND gates alone, or NOR gates alone can perform all Boolean operations. This is of great practical importance because some types of solid-state logic make it easier to implement NAND gates, while others lead to a preference for OR and NOR gates. Many commercially available logic systems also offer logic gates with more than two inputs, which are often convenient (Fig. 1-7).

A flip-flop is a 1-*bit memory device* for storing a binary variable; flip-flop registers are ordered sets of flip-flops for storing digital words. Table 1-5b defines each of the most useful flip-flop types by the method of data entry and shows two important applications (see also Secs. 1-7 and 5-3). Figure 1-8 shows how appropriately timed control pulses are used to parallel-transfer the contents of a flip-flop register to other registers.

Digital-computer arithmetic circuits will be designed as logic circuits operating on the bits of binary-number inputs to produce desired binary output numbers, with inputs, outputs, and intermediate results stored in flip-flop registers (Sec. 1-9).

Techniques for simplifying logic circuits (i.e., minimizing the number of gates and flip-flops, gate inputs, interconnections, and/or crossovers) form the subject of logic optimization for digital-system design (Refs. 1 to 5). Optimization of a large digital system, such as a complete computer, is



AND/NOR gate
NOR gate if A = B, C = D
XOR gate if $C = \bar{A}$, $D = \bar{B}$

Multi-input NAND gate

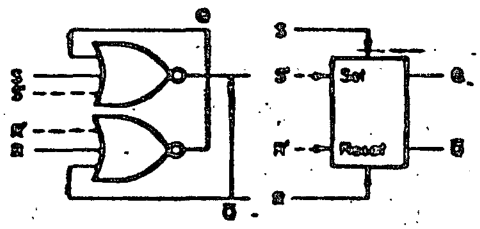
NAND gates with wired OR
(not possible with
all types of logic)

Fig. 1-7. Some multi-input gates available in integrated-circuit form. Many other types exist.

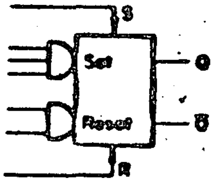
often itself done with the help of a digital computer. On the other hand, a researcher or engineer who merely wants to use a small digital computer, and to interface it to some real-world instruments and controls, will seldom require formal logic optimization. *All we usually require is the material in Table 1-5, some reasonable common sense, and a nice collection of tried logic circuits we can adapt and modify.* Manufacturers' catalogs and application

TABLE 1-5b. Very Little Logic Goes a Long Way: Flip-flop Circuits

1. Flip-flops. A flip-flop, like the familiar toggle switch, will stay in a given output state (0 or 1) even after inputs have been removed. Flip-flops thus implement *memory* for binary variables and permit *data storage* and *automatic sequential operations*. (That is, logic states can determine the sequence of future logic states, as in data transfers, counting, etc.) Although a somewhat bewildering variety of different flip-flops are sold, all are derived from a few simple types. Specifically, the basic *reset/set* (RS) *flip-flop* retains its output state through regenerative feedback until a new reversing input is applied. Other types of flip-flops add different input-gating circuits.



(*a*) Reset/set (RS) flip-flop. $R = 1$ (level or pulse) resets (clears) the flip-flop ($Q = 0$) until $S = 1$ sets the flip-flop ($Q = 1$). $R = S = 0$ leaves output unchanged. $R = S = 1$ is *illegal* (indefinite output), or $R = 1$ may *override* $S = 1$. Multiple set inputs or multiple reset inputs are ORed together.

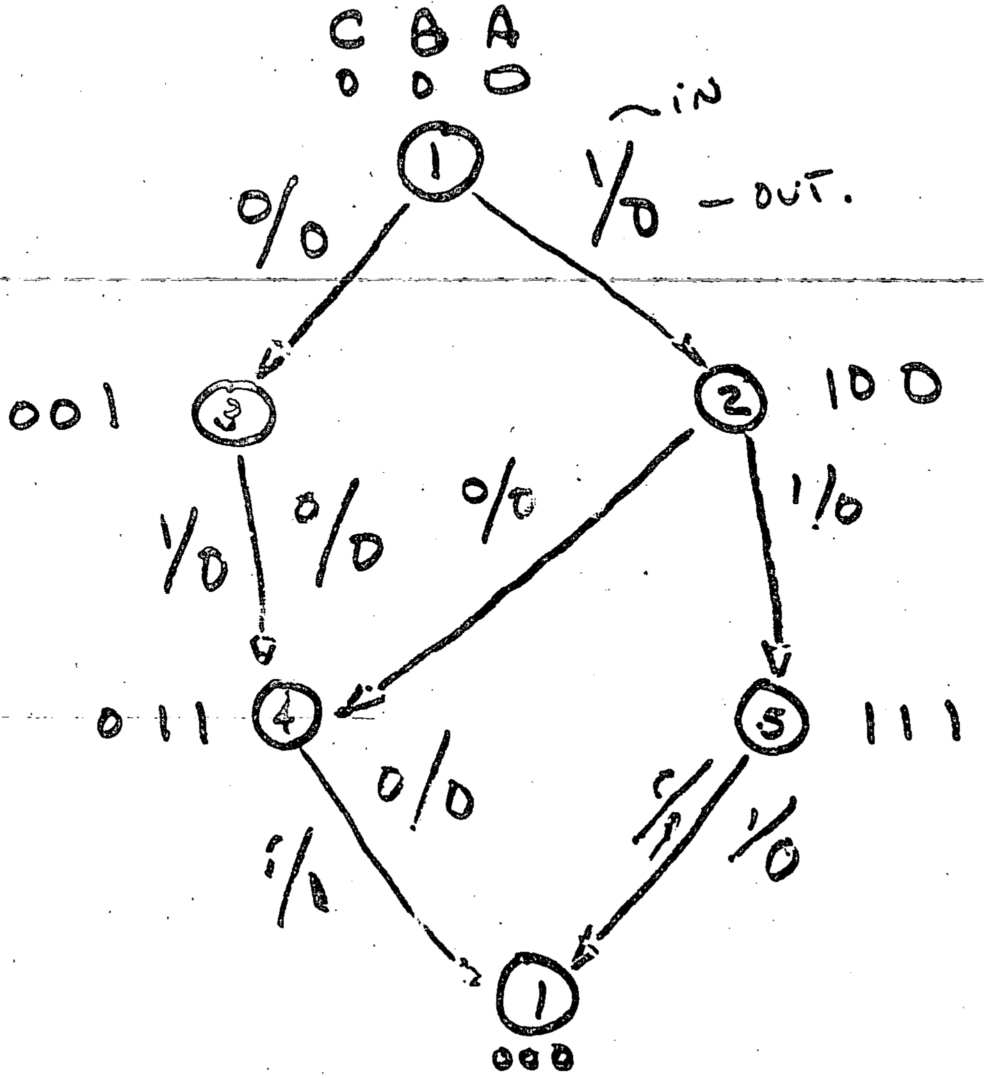


(*b*) General-purpose flip-flop with enabling gates. Many different types exist. Inputs may include inverting set and/or reset inputs, multiple gate inputs, etc. Frequently, the lower reset input is designed to override all other inputs.

In some general-purpose flip-flops (diode/transistor logic, DTL), gates have *ac-coupled inputs*, which set or reset the flip-flop when a voltage step (either up or down, depending on the type) is gated by a logic level.

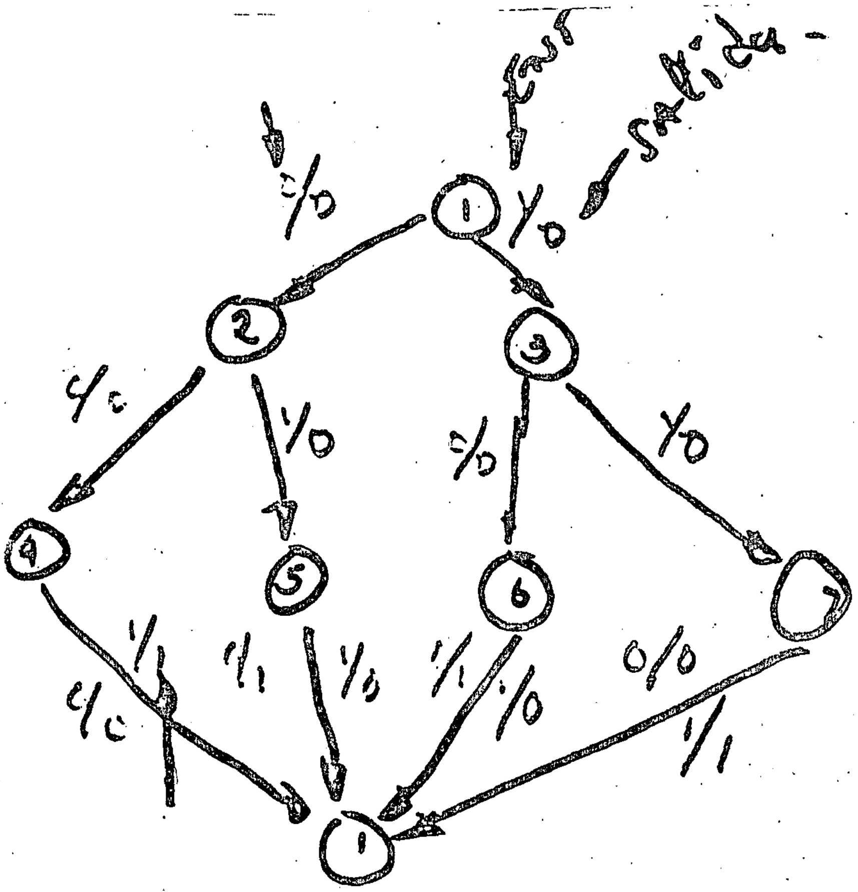
CIRCUITOS SEQUENCIALES.

DIAGRAMAS DE ESTADO.



EJEMPLO.— DISEÑE UN CIRCUITO QUE SU SALIDA SEA UNO SI EL NUMERO DE UNOS CONTENIDO EN UNA SECUENCIA DE 3 BITS ES NON.

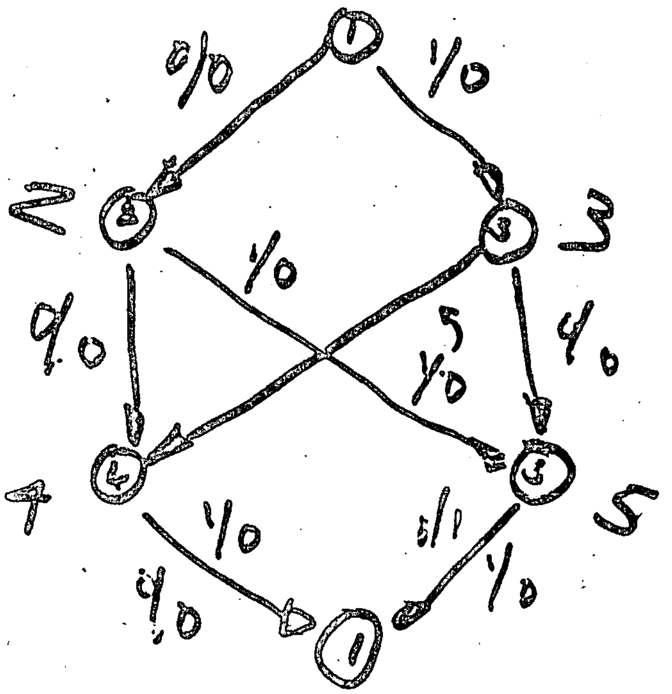
$$\left.\begin{matrix}001\\010\\100\\111\end{matrix}\right\}=1 \qquad \left.\begin{matrix}000\\011\\110\end{matrix}\right\}=0$$



	SIG. EDO		SALIDA.	
	ENTRADA		ENTRADA	
EDO. PRES.	0	1	0	1
1	2	3	0	0
2	4	5	0	0
3	6	7	0	0
4	1	1	0	1
5	1	1	1	0
6	1	1	1	0
7	1	1	0	1

NOTEMOS QUE 4=7
5=6

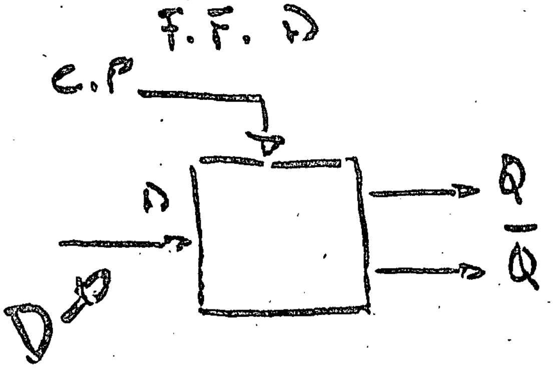
POR LO QUE LA TABLA Y EL DIAGRAMA SE PUEDEN REDUCIR QUEDANDO:



~~5 = 0~~

~~4 =~~

FLIP-FLOPS



D	Q^{n+1}
0	0
1	1

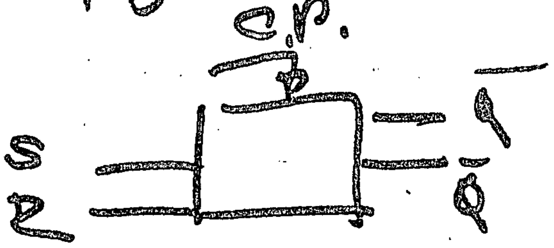
Q^n →	Q^{n+1}	D
0	0	0
0	1	1
1	0	0
1	1	1

(F.F. DELAY).

F.F. T (TOGGLE)

T	Q^{n+1}
0	Q^n
1	$\overline{Q}^n$

Q^n	$\to Q^{n+1}$	T
0	0	0
0	1	1
1	0	1
1	1	0



S.R. (SET. RESET)

S	R	Q^{n+1}
0	0	Q^n
0	1	0
1	0	1
1	1	?

Q^n	$\to Q^{n+1}$	S	R
0	0	0	∅
0	1	1	∅ ← 0
1	0	0	1
1	1	∅	0

F.F. J.K.

J	K	Q^{n+1}
0	0	Q^n
0	1	0
1	0	1
1	1	$\overline{Q}^n$

Q^n	$\to Q^{n+1}$	J	K
0	0	0	∅
0	1	1	∅
1	0	∅	1
1	1	∅	0

EN LA SIGUIENTE FIGURA SE MUESTRA LA ORGANIZACIÓN DE UNA COMPUTADORA DE 'DE UNA DIRECCIÓN.

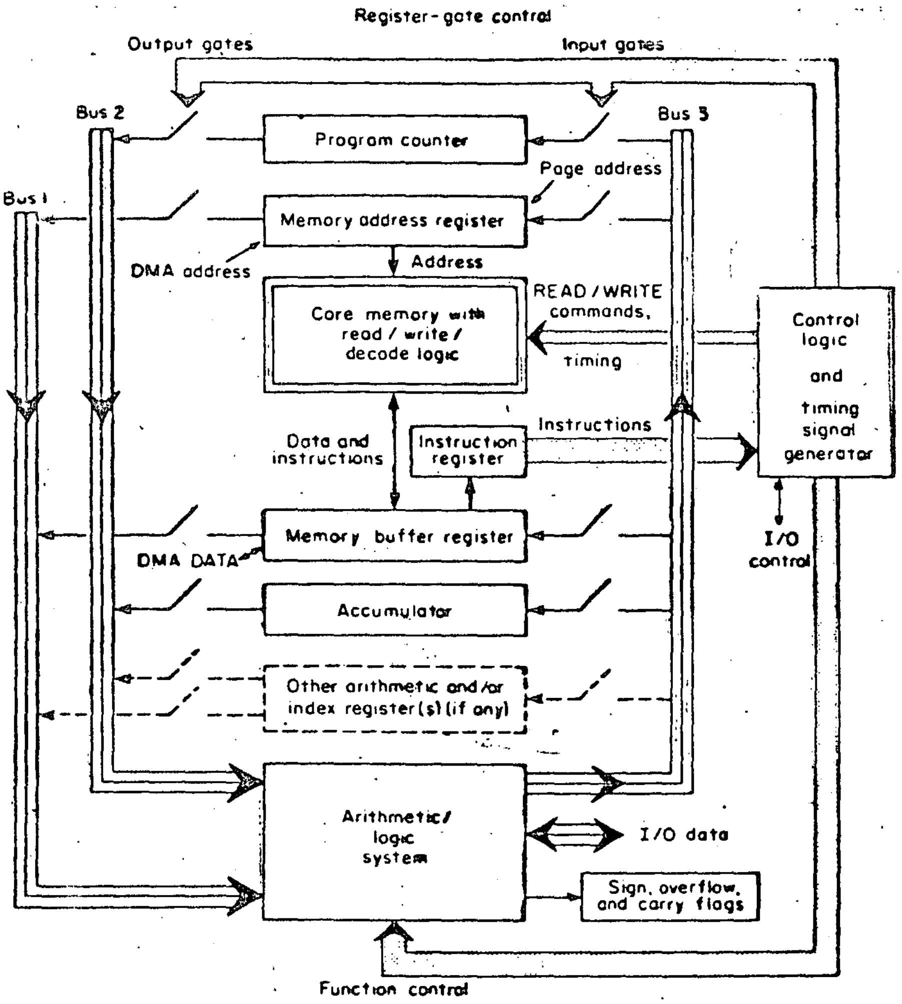


Fig. 2-1. Organization of a "basic" single-address minicomputer.

LA CUAL TIENE LOS SIGUIENTES MODULOS:

MEMORIA.- (CORE O C.I.) GUARDA EL PROGRAMA Ó INSTRUCCIONES Y LOS DATOS.

REGISTRO DE MEMORIA (M.B.R) CONTIENE LA INSTRUCCIÓN O DATO QUE ENTRA Ó SALE DE LA MEMORIA.

REG. DE DIRECCIÓN DE MEMORIA (MAR) - CONTIENE LA DIRECCIÓN EN LA MEMORIA DE LA PALABRA PRESENTE.

CONTADOR DE PROGRAMA (P.C.) CONTIENE LA DIRECCION DE LA INSTRUCCION (SIGUIENTE) A SER EJECUTADA.

REGISTRO DE INSTRUCCION (IR) CONTIENE LA ~~[illegible]~~ INSTRUCCION (PRESENTE).

REGISTRO DE USO GENERAL (ACCUMULADOR, , REGISTRO ARITMETICO REGISTRO DE INDICE ETC).

REGISTRO DE UN BIT — GUARDA EL RESULTADO DE LA OPERACION EFECTUADA. , OVERFLOW, CARRY, SIGNO ETC.

U.A.L (ALU)

LOGICA DE CONTROL .- DECODIFICA LOS CEROS Y UNOS DE LA INSTRUCCION EN EL I.R. PARA GENERAR LOS PULSOS DE TIEMPO Y NIVELES LOGICOS.

BUSES.

CONECCIONES DE E/S.

OPERACION DEL PROCESADOR

INSTRUCCION FETCHING (TOMAR INSTRUCCION)

EL CONTENIDO DEL P.C. PASA AL M.A.R. —→ LA INSTRUCCION DIRECCIONADA ES LEIDA EN EL M.B.R. Y DE AHI AL I.R.

EJECUTA INSTRUCCION. — UNA VEZ QUE LA INSTRUCCION ESTA EN EL I.R. ES DECODIFICADA Y PASA AL CONTROL DE OPERACIONES, SE ~~[illegible]~~ ABREN LAS COMPUERTAS, SE EJECUTA LA INSTRUCCION. — CADA VEZ QUE SE EJECUTA UNA INSTRUCCION, SE ~~[illegible]~~ AUMENTA EN UNO EL CONTADOR DEL PROGRAMA.

MODOS DE DIRECCIONAMIENTO.

COMO SE VIO EN CLASE PASADA CON EL ING. L.P. GRIJALVA UNA PALABRA DE COMPUTADORA LUCE DE LA SIGUIENTE FORMA.

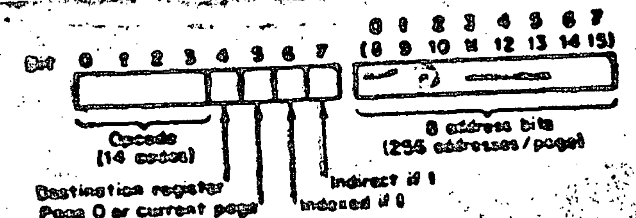


Fig. 2-28. This could be a one-word memory-reference instruction for a 16-bit minicomputer or a two-word memory-reference instruction for an 8-bit machine (see also Fig. 6-8).

EN DONDE USUALMENTE SE UTILIZAN LOS ULTIMOS 8 BITS PARA DIRECCIONAR. LAS COMPUTADORAS (MINIS O MICROS) USUALMEN PERMITEN VARIOS MODOS DE DIRECCION DE LOS QUE A CONTINUACI SE MENCIONAN.

DIRECCION EN PAGINA CERO (SE DIRECCIONAN $2^D - 1$ PALABRAS AL PRINCIPIO DE LA MEMORIA PRINCIPAL. — i.e. SI D=8 SE PUEDEN DIRECCIONAR 255 ($2^8 = 256$) PALABRAS.

PAGINA PRESENTE. SE DIVIDE LA MEMORIA EN 256 PAGINAS (SI D= 8) Y UTILIZA D PARA ESPECIFICAR LA DIRECCION EN LA PAGINA DE LA CUAL SE DIO LA PRESENTE INSTRUCCION.

DIRECCIONAMIENTO RELATIVO .- UTILIZA A D COMO UN NUMERO CON SIGNO Y SE LO SUMA A LA DIRECCION DE LA CUAL SE TOMO LA INSTRUCCION PRESENTE.

DIRECCIONAMIENTO CON REGISTRO BASE .- EL DIRECCIONAMIENTO ES SIMILAR AL ANTERIOR, EXEPTO QUE LA DIRECCION SE GUARDA EN UN REGISTRO BASE (CUYO CONTENIDO SE CONTROLA CON OTRAS INSTRUCCIONES).

DIRECCIONAMIENTO INDEXADO — SUMA EL CONTENIDO DE UN REGISTRO DE INDICE AL CONTENIDO DE LA DIRECCION (D)

DIRECCIONAMIENTO INDIRECTO.— EL DIRECCIONAMIENTO SE LLEVA A CABO EN CONJUNTO CON UNA DE LAS FORMAS ANTERI
EL DIRECCIONAMIENTO INDIRECTO USA EL CONTENIDO ESPECIFICA POR D, PARA DIRECCIONAR OTRA PALABRA, CUYO CONTENIDO SERA UTILIZADO COMO DATO. ~ CON 16 BITS SE PUEDEN DIRECCIONAR 64 K PALABRAS (2^{16}).

* * *

TODAS LAS COMPUTADORAS ACEPTAN Y EMITEN DATOS UTILIZANDO TRANSFERENCIAS PROGRAMADAS DE DATOS EN CONJUNCION CON UN SISTEMA DE INTERRUPCIONES Y PRIORIDADES (PUEDE EXISTIR DMA). LA TRANSFERENCIA DE DATOS EN MINICOMPUTADORAS ES DE 80,000 PALABRAS/SEG (TIPICO

EL SISTEMA DE INTERRUPCIONES Y PRIORIDADES ES VITAL PARA UNA COMPUTADORA, PUES DA SERVICIO A LOS DIFERENTES DISPOSITIVOS CUANDO LO REQUIEREN, O CUANDO EXISTE UNA FALLA, Ó PERDIDA DE LA INFORMACION.

VEAMOS UN EJEMPLO DE INTERRUPCION EN E/S.

1.- GUARDA EL CONTENIDO DE TODOS LOS REGISTROS EN MEMORIA
2.- DETERMINA QUE DISPOSITIVO GENERO LA INTERRUPCION
3.- SE INICIALIZA LA SUBRUTINA DE INTERRUPCION, HASTA HASTA QUE QUEDE SATISFECHO EL DISPOSITIVO QUE LA DISPARA
4.- REGRESA EL CONTENIDO DE LOS REGISTROS QUE FUERON GUARDADOS EN EL PASO 1
5.- CONTINUA EL PROCESO.

ALGUNAS COMPUTADORAS IMPLEMENTAN LO ANTERIOR EN HARDWARE, OTRAS EN SOFTWARE

PARA TENER UNA TRANSMISION DE DATOS EFICIENTE, CASI TODAS LAS MININICOMPUTADORAS UTILIZAN ~~EL~~ ACCESO DIRECTO A MEMORI (DMA). EL DMA PROVEE UN ACOPLAMIENTO 'RAPIDO' ENTRE LA COMPUTADORA Y LA MEMORIA (USUALMENTE DISCO, PARA LLEVAR ACABO TRANSFERENCIAS MASIVAS DE INFORMACION. SI NO SE TIENE D.M.A. HAY QUE INTERRUMPIR Y LLEVAR ACABO EL PROCESO DESCRITO ANTERIORMENTE. — PARA CADA PALABRA (O BLOQUE) TRANSFERIDO EL DMA UTILIZA UN CICLO DE MEMORIA. LA MANERA TIPICA DE UTILIZAR EL CANAL DMA ES LA SIGUIENTE.

COMO FUNCIONA EL DMA.

1.- CUANDO LA COMPUTADORA ESTA LISTA PARA TRANSMITIR DATOS ENTRE LA MEMORIA PRINCIPAL Y EL DISCO, SE TRANSFIERE INFORMACION DE INICIO A LA INTERFASE DEL DMA. ESTA INFO. INCLUYE LA DIRECCION DE LA PRIMERA PALABRA DEL BLOQUE TANTO PARA LA MEM. PRINCIPAL COMO PARA EL DISCO, ASI COMO EL NUM. DE PALABRAS EN EL DISCO FINALMENTE SE INCLUYE LA DIRECCION (SENTIDO) DE LA TRANSFERENCIA, ASI COMO UN COMANOD DE INICIO. LA COMPUTADORA CONTINUA CON SU TRABAJO NORMAL, DEJANDO AL DMA POR SI SOLO

2.- LA INTERFASE DEL DMA SE ESPERA A QUE APAREZCA LA 1ª DIRECCION DEL BLOQUE EN EL DISCO, Y MANDA UNA SEÑAL A LA COMPUTADORA. LA COMPUTADORA TERMINA EL CICLO INICIADO, Y NO HACE NADA DURANTE UN CICLO DE MEMORIA. EL CANAL DEL DMA SE 'ROBA' EL CONTROL DE LA MEMORIA DURANTE ESTE CICLO, TRANSMITIENO UNA PALABRA. UNA VEZ TRANSMITIDA LA PALABRA LA COMPUTADORA CONTINUA CON SU OPERACION SIN HABERSE DADO CUENTA, QUE UNA TRANSMISION A MEMORIA SE LLEVO ACABO.

3.- UNA VEZ QUE TODO EL BLOQUE FUE TRANSFERIDO, LA INTERFASE DMA PARA. MANDA UNA SEÑAL A LA COMPUTADORA QUE TERMINO, HACIENDO USO DEL SISTEMA DE INTERRUPCION. LA COMPUTADORA PUEDE AHORA HACER USO DE LA INFORMACION EN MEMORIA. (O DEL ESPACIO)

DEBIDO AL ROBO DE CICLOS EL DMA TIENE LA PRIORIDAD MAS ALTA DE CUALQUIER OTRO DISPOSITIVO DE E/S.

VEAMOS A CONTINUACION LA ORGANIZACION DE UNA MAQUINA 'POPULAR' LA PDP 11, y PODER UTILIZAR ESTA MAQUINA EN "INTERFAZ". TEMA DE INTERES EN ESTE CURSO.

LA COMPUTADORA PDP 11 ESTA ORGANIZADA ASI:

1.- CUENTA CON UN BUS (CANAL) UNICO AL CUAL ESTAN CONECTADOS EL PROCESADOR, LA MEMORIA, Y TODOS LOS PERIFERICOS. EL BUS TIENE 56 LINEAS, Y ESTAS SON:

16 LINEAS DE DATOS
13 TRANFERENCIA DE PRIORIDADES
18 LINEAS DE DIRECCION
9 CONTROL

TODAS LAS LINEAS SON BIDIRECCIONALES, EXEPTO 5. CADA DISPOSITIVO ESTA DIRECCIONADO COMO UNA LOCALIDAD DE MEMORIA, EXEPTO QUE LA DIRECCION ES MAYOR QUE CUALQUIER LOCALIDAD.

2.- UN SISTEMA DE INTERRUPCIONES Y PRIORIDADES FLEXIBLE, QUE PERMITE A CUALQUIER DISPOSITIVO TOMAR CONTROL DEL BUS, SIEMPRE Y CUANDO EL CPU HUBIESE TERMINADO UNA INSTRUCCION.

UNA TRANSFERENCIA DE DATOS HUBIESE TERMINADO. CUANDO UN DISPOSITIVO TOMA CONTROL DEL BUS, EL CPU DEJA LO QUE ESTABA HACIENDO, E INICIALIZA UNA RUTINA DE SERVICIO, PARA SATISFACER EL DISPOSITIVO. EL DISPOSITIVO EN CONTROL PUEDE TRANSFERIR INFORMACION A CUALQUIER OTRO DISPOSITIVO, ESTO SIGNIFICA QUE CUALQUIER DISPOSITIVO TIENE UN CANAL DMA.

3 LAS INSTRUCCIONES Y LA ORGANIZACION DE LA MAQUINA ESTA HECHA PARA RECONOCER RAPIDAMENTE UNA INTERRUPCION, ASI COMO SU PRIORIDAD.

INTERFAZ EN UNA MINICOMPUTADORA.

LA INTERFAZ ENTRE UNA MINICOMPUTADORA Y CUALQUIER DISPOSITIVO DE E/S, QUE UTILIZA TRANSFERENCIA PROGRAMADA DE DATOS, INVOLUCRA LAS SIGUIENTES DOS FUNCIONES

1.- TRANSFERENCIA DEL DATO (PALABRA) ENTRE LA COMPUTADORA Y EL DISPOSITIVO DE E/S SELECCIONADO.

2.- SINCRONIZAR CUANDO LA COMPUTADORA Y EL DISPOSITIVO SELECCIONADO ESTEN LISTOS PARA TRANSMITIR LOS DATOS, UTILIZANDO EL SISTEMA DE INTERRUPCION Y PRIORIDAD.

LO ANTERIOR SE LLEVA A CABO EN LA COMPUTADOR UTILIZANDO AMBOS 'SOFTWARE' Y 'HARDWARE' Y UN SISTEMA (INTERFAZ) DE CIRCUITERIA EN HARDWARE.

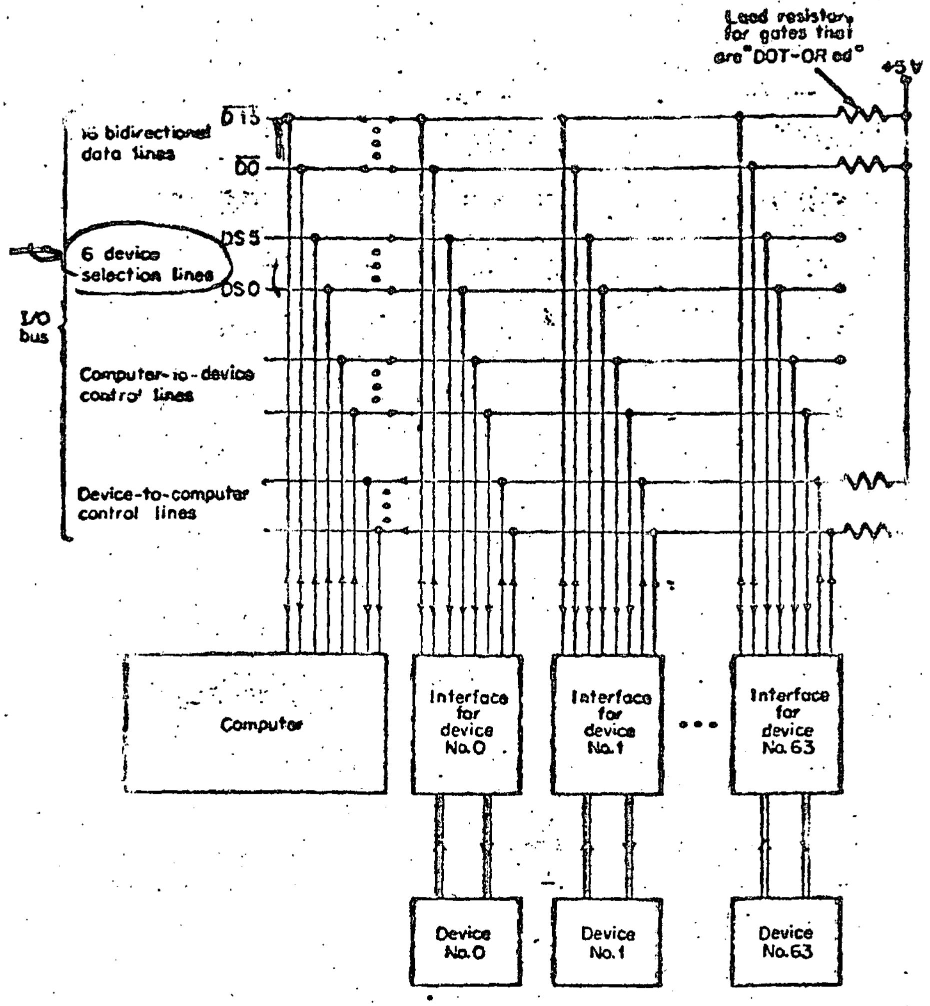


Fig. 6-30 Typical I/O interconnections for devices using programmed data transfers.

SI LA COMPUTADORA TIENE PALABRA DE 16 BITS, SE PODRAN TENER HASTA $2^6 = 64$ DISPOSITIVOS DE E/S Y EL BUS CONSISTIRA DE: (FIGURA ANTERIOR)

- 16 LINEAS DE DATOS
- 6 LINEAS SELECCIONADORAS DE DISPOSITIVOS
- X LINEAS DE CONTROL DE LA COMPUTADORA AL DISPOSITIVO

Y LINEAS DE CONTROL DEL DISPOSITIVO A LA COMPUTA

LAS LINEAS DE CONTROL SINCRONIZAN LA TRANSFERENCIA DE DATOS, EL INICIO Y FINAL DE LAS OPERACIONES. ASI COMO EL CONTROL DEL SISTEMA DE INTERRUPCION Y PRIORIDADES.

DURANTE UNA OPERACION DE SALIDA LOS DATOS SALEN DEL ACUMULADOR (EN LA COMPUTADORA) A UN REGISTRO EN LA INTERFAZ. ESTE REGISTRO SE UTILIZA PARA TENER LOS DATOS LISTOS CUANDO SE NECESITEN

POR EJEMPLO SI SE QUIEREN TRANSMITIR DATOS DE UN ACUMULADOR (A15 ~ A0) A UN REGISTRO (X15 ... X0), (LA INTERFAZ SE MUESTRA EN LA SIGUIENTE FIGURA). LA COMPUTADORA SELECCIONA UN DISPOSITIVO ESPECIFICO CARGANDO 'SU NUMERO', EN UN REGISTRO SELECCIONADOR DE DISPOSITIVO. ~ POR EJEMPLO PARA EL DISPOSITIVO 13 DS0 — DS5 = 001101 = 13_{10}

EL CIRCUITO DECODIFICA LA SEÑAL Y GENERA UNA SEÑAL DE 'DISPOSITIVO SELECCIONADO'

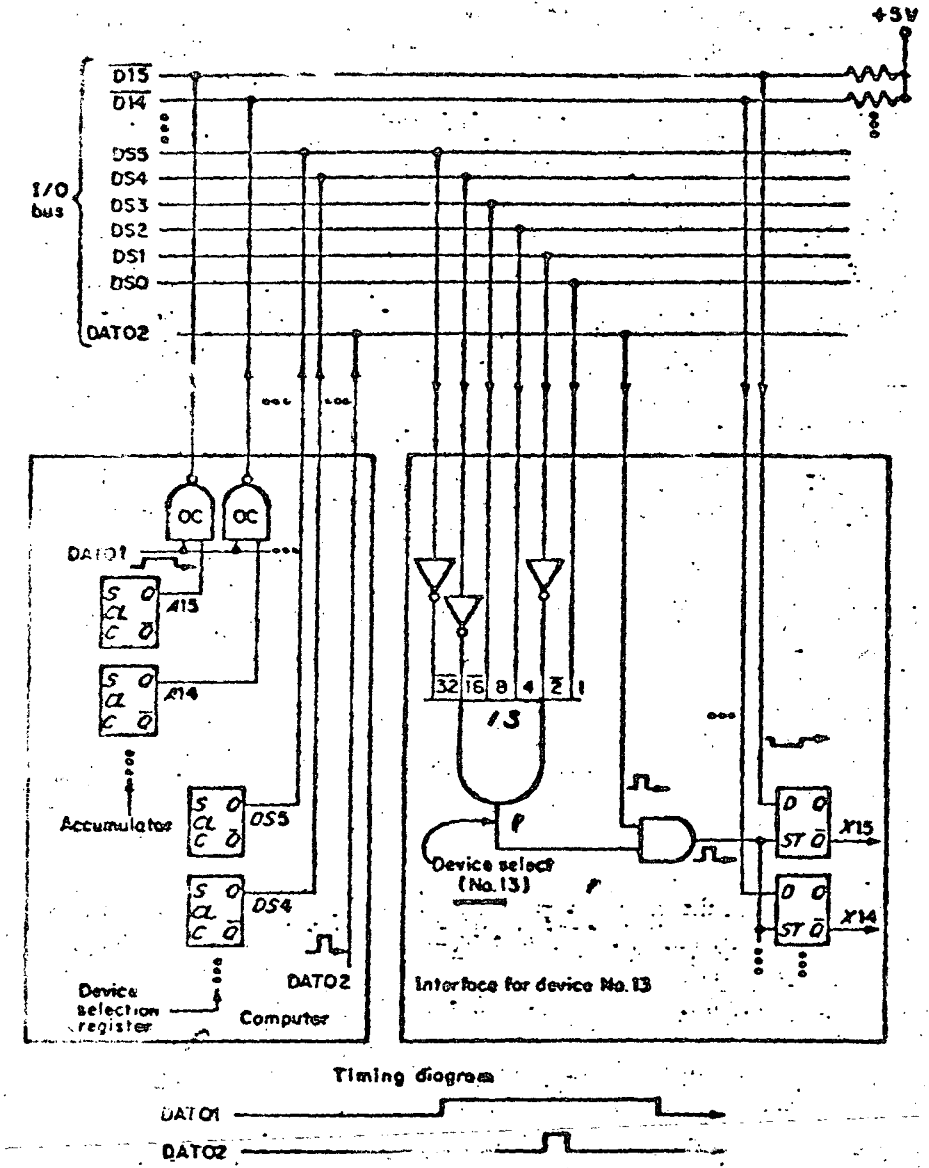


Fig. 6-32 Transferring data from the computer to a device interface.

USUALMENTE SE GENERA UN PULSO DE CONTROL ANGOSTO DATOP, EL CUAL OCURRE DESPUES DE DATO1 y ANTES DE DATO2

LO CUAL SE UTILIZA EN UN 'PRESET' DE LAS LINEAS DE DATOS QUE VAN AL REGISTRO DE F.F. (VER SIGUIENTE FIGURA) EL PULSO DATOP ES 'AND' CON EL DISPOSITIVO SELECCIONADO, PARA FORMAR UN PULSO EL CUAL HACE UN PRESET DIRECTO EN TODOS LOS F.F. DESPUES DATO2 ES 'AND' CON EL DISPOSITIVO SELECCIONADO y CON CADA LINEA DE DATOS PARA TENER UN CLEAR DIRECTO EN CADA F.F.

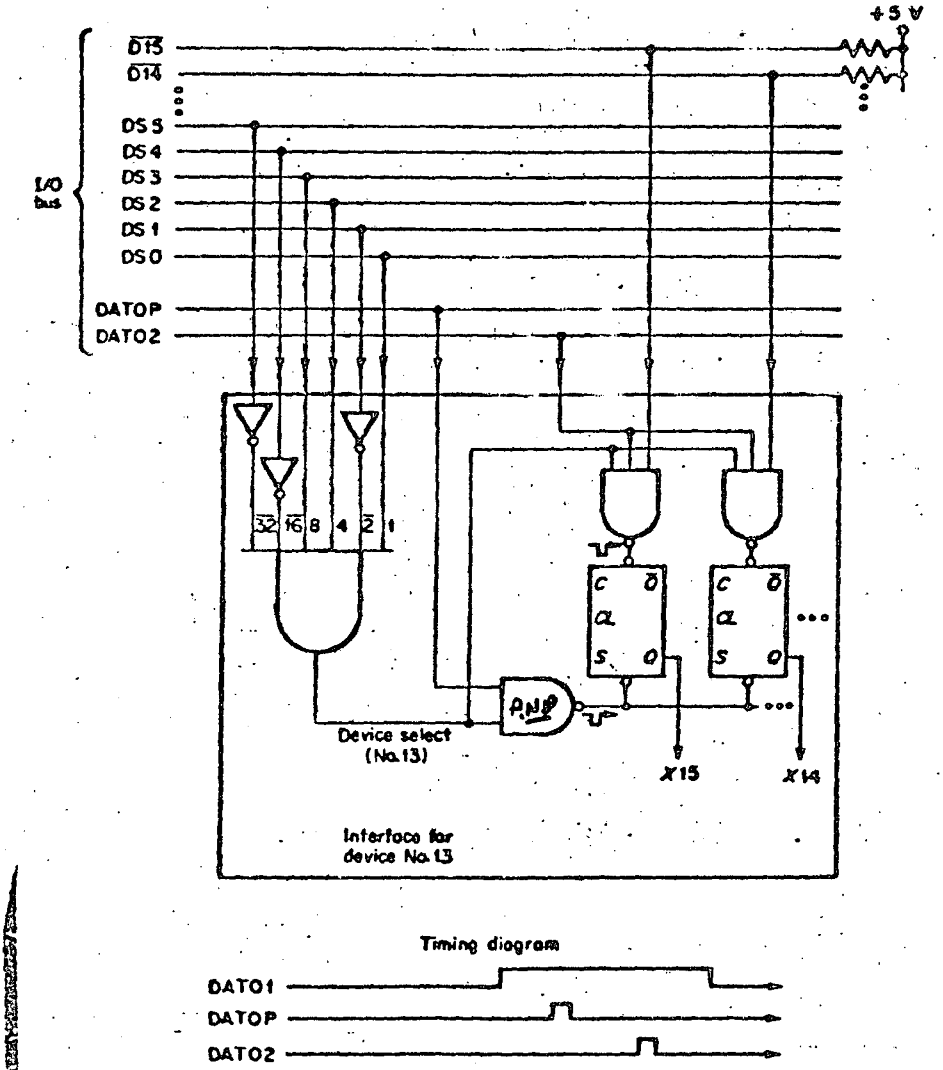


Fig. 6-33 Two-step presetting of a device interface register.

LA TRANSFERENCIA DE DATOS DE UN REGISTRO A UN ACUMULADOR EN LA COMPUTADORA SE ILUSTRA A CONTINUACION.

EL REGISTRO SELECCIONADOR DE DISPOSITIVO EN LA COMPUTADORA MANDA UN PULSO DE CONTROL DAT 11 A TODAS LAS INTERFACES.

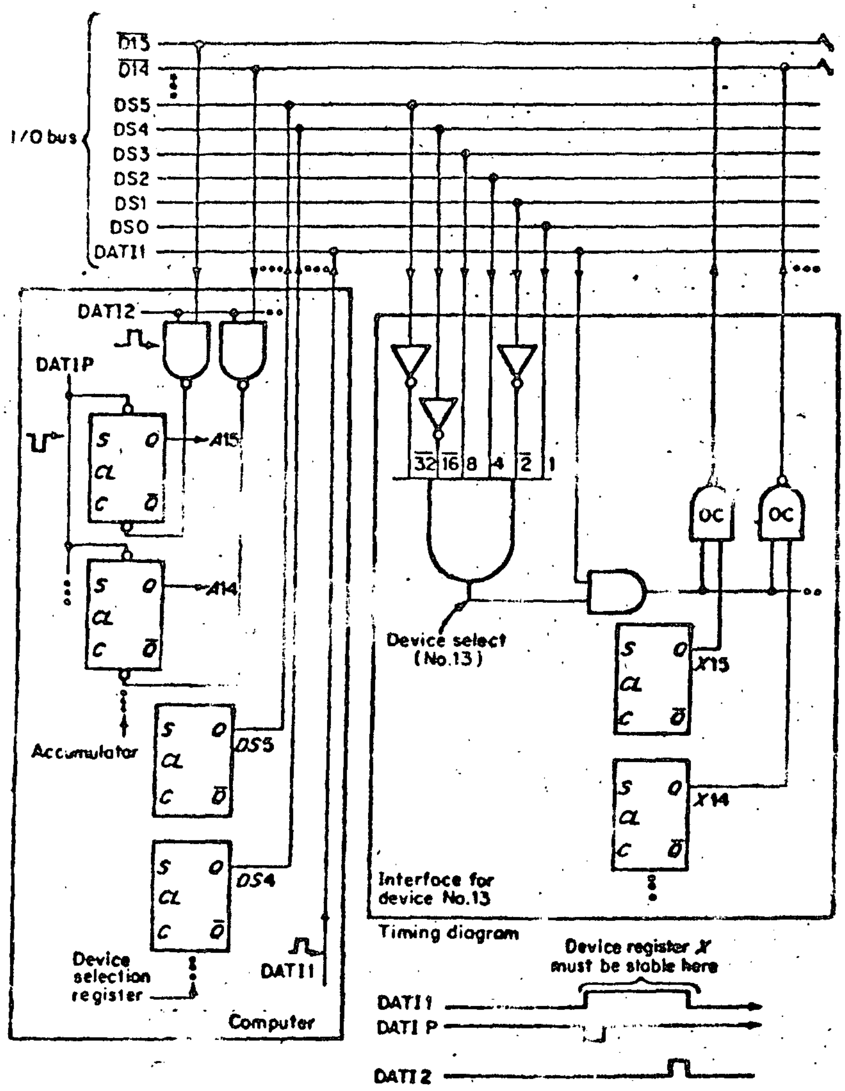


Fig. 6-34 Transferring data from a device interface register to an accumulator in the computer.

EL SISTEMA 'RUDIMENTARIO' SE VERA COMO SE MUESTRA EN LA SIGUIENTE FIGURA

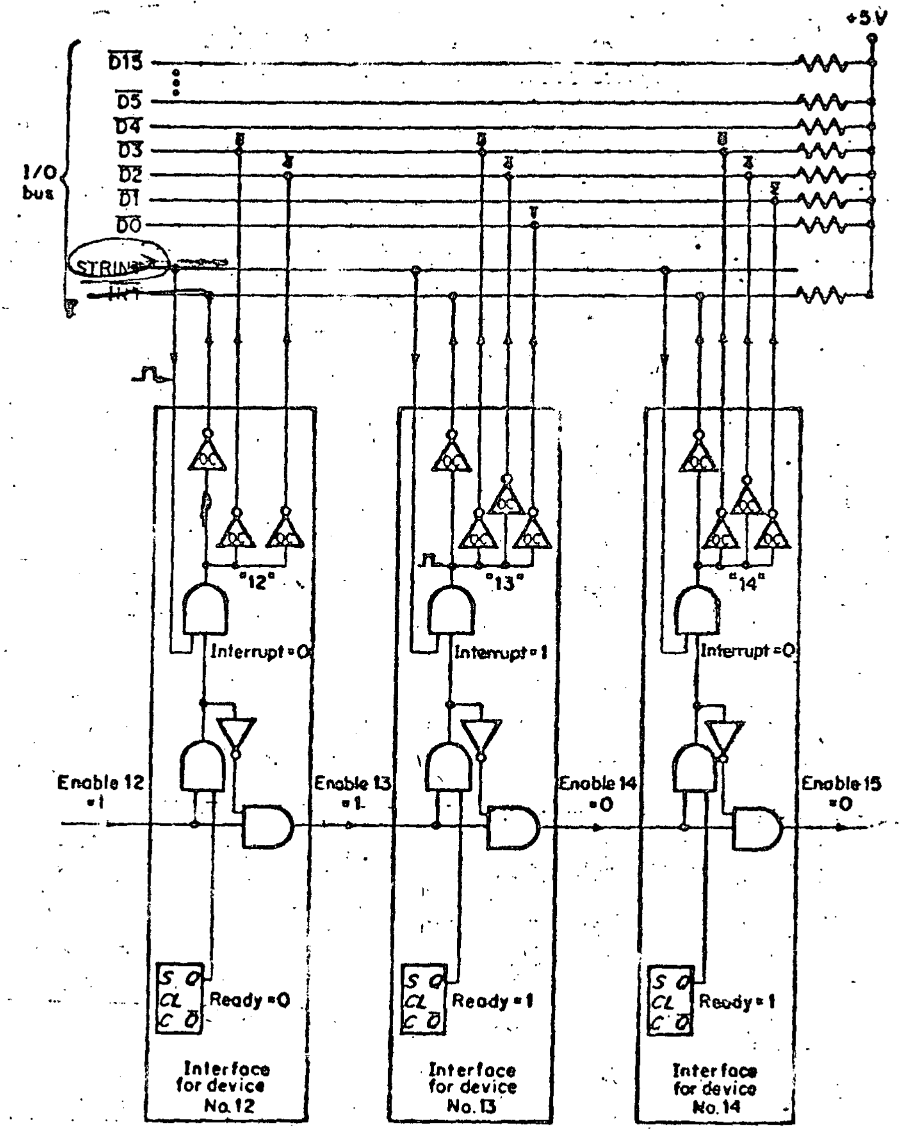


Fig. 6-37 Rudimentary priority system.

COMUNMENTE VARIOS DISPOSITIVOS SUELEN INTERRUMPIR A LA VEZ UNA FORMA DE RESOLVER ESTE PROBLEMA ES UTILIZANDO UN SISTEMA DE PRIORIDADES. ESTE SISTEMA NOS DEBE DE ASEGURAR QUE SOLAMENTE UN DISPOSITIVO SE DEBE RECONOCER A LA VEZ. LA COMPUTADORA INICIA SU TAREA DANDO SERVICIO AL DISPOSITIVO DE MAYOR PRIORIDAD, PARA LO CUAL DEJA LA SEÑAL DE INTERRUPCION DE ESE DISPOSITIVO PRENDIDA, QUITANDO LAS SEÑALES DE INTERRUPCION DE LOS DISPOSITIVOS DE MENOR PRIORIDAD.

UNA FORMA DE RESOLVER EL PROBLEMA ANTERIOR ES UTILIZANDO UNA MASCARA DE PRIORIDADES. EN LA CUAL CADA DISPOSITIVO ~~A UNO~~ DE LOS 16 GRUPOS DE PRIORIDADES. SI HAY 2 DISPOSITIVOS CON LA MISMA PRIORIDAD UNO NO PUEDE INTERRUMPIR A OTRO

SUPONGAMOS QUE EL DISPOSITIVO #13 SE LE ASIGNO PRIORIDAD DEL GRUPO 5. CUANDO ESTE DISPOSITIVO HAGA UNA INTERRUPCION, LO PRIMERO QUE HACE LA COMPUTADORA, AL INICIAR INICIAR LA SUBRUTINA DE SERVICIO ES LEER LA 'MASCARA DE PRIORIDADES' DE ESTE DISPOSITIVO LA CUAL SE ENCUENTRA EN MEMORIA Y LA COLOCA EN UN ACUMULADOR, ESTA MASCARA PUEDE TENER APAGADOS LOS DISPOSITIVOS EN LOS GRUPOS 11, 10, 5, Y 1, LOS OTROS DISPOSITIVOS SE QUEDAN 'ACTIVOS' Y PUEDEN INTERRUMPIR AL #13

sTema
de
. opción

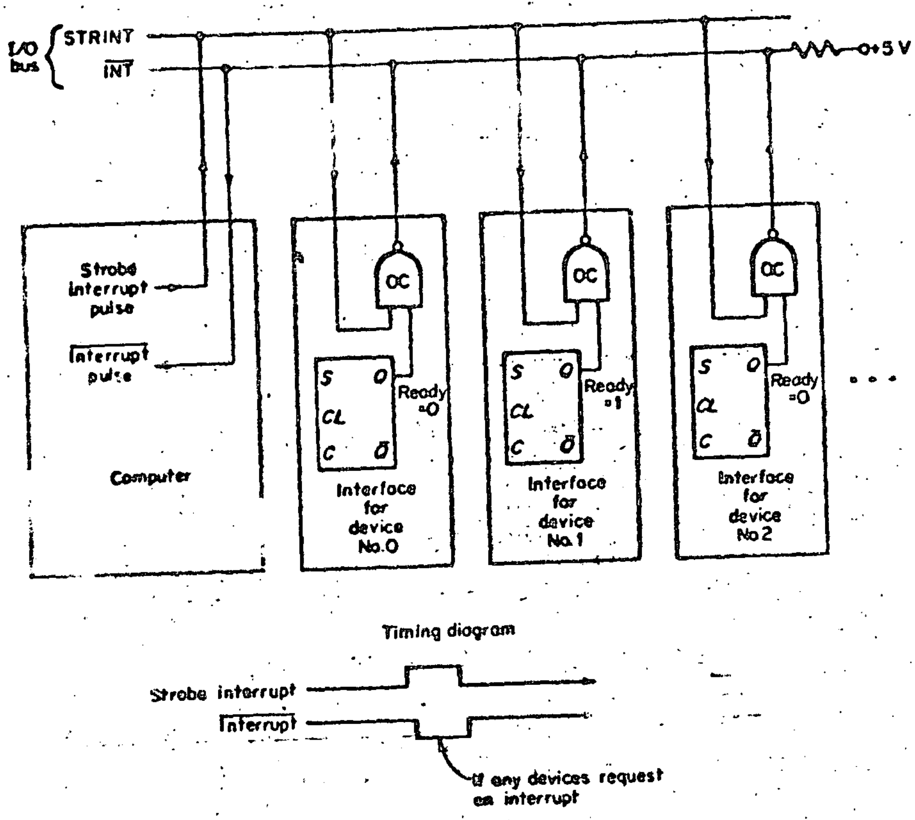


Fig. 5-35 Rudimentary interrupt system.

IDENTIFICACACIÓN DEL DISPOSITIVO QUE INTERRUMPIO

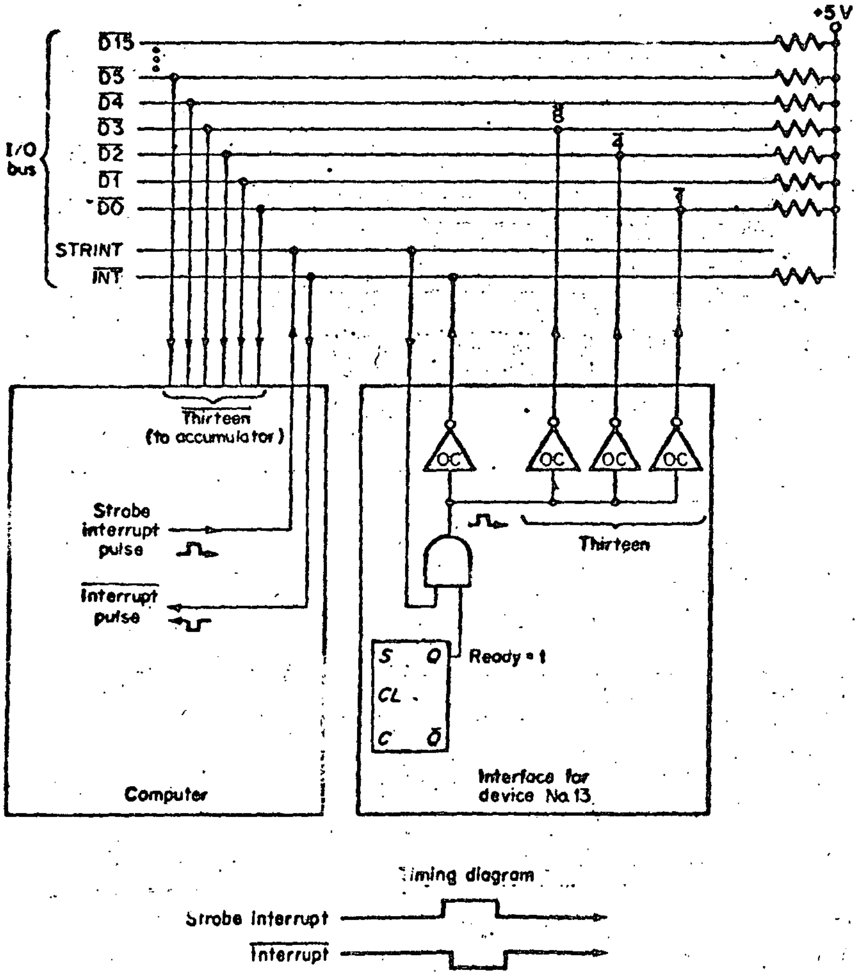
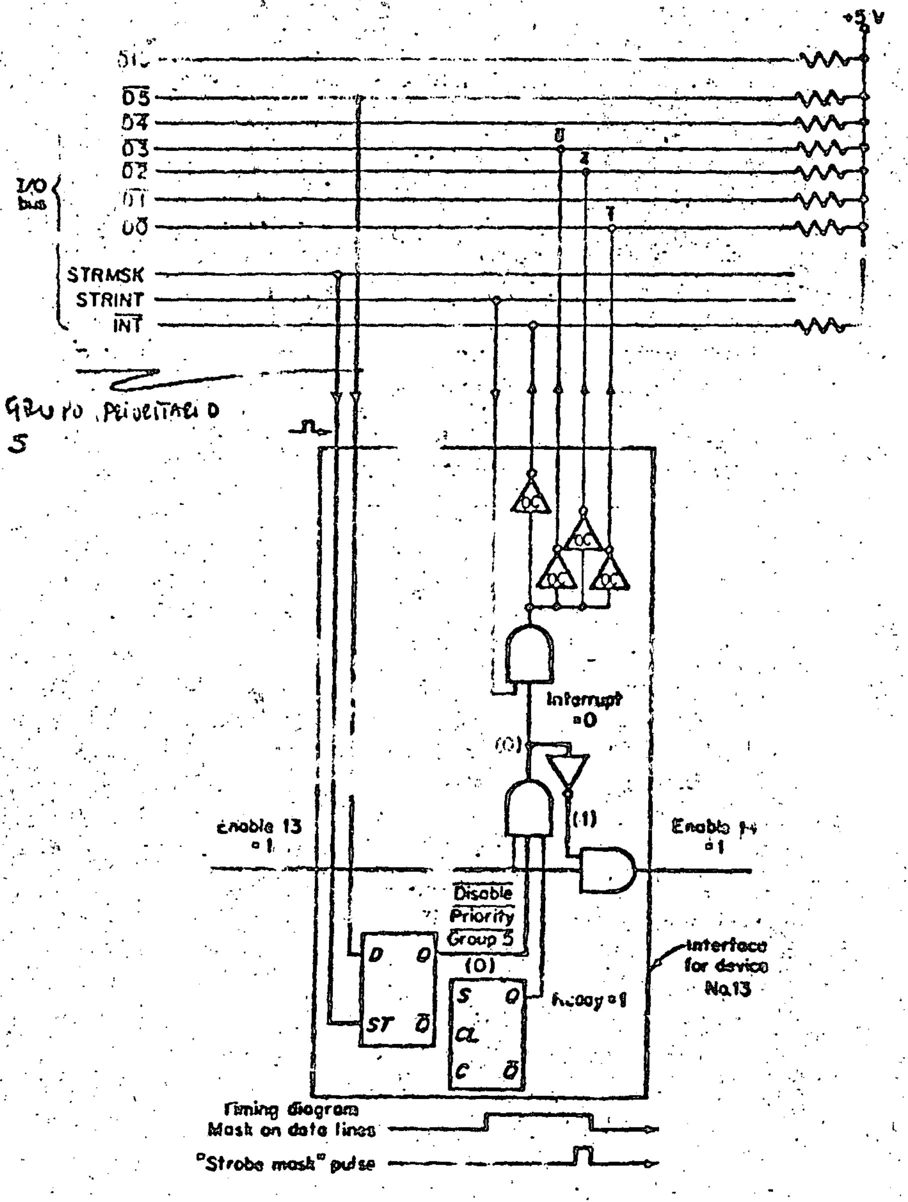


Fig. 6-36 Identifying an interrupting device.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	1	1	0	0	0	0	1	0	0	0	1	0

Fig. 6-38 Priority mask.

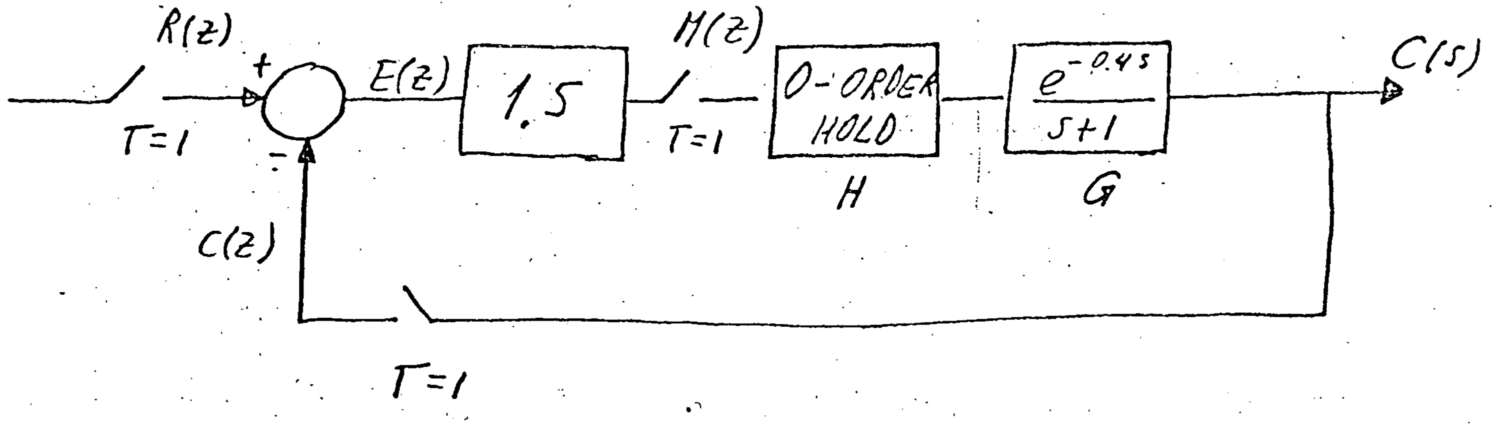


GRUPO PRIORITARIO
5

Fig. 6-39 Setting priorities with a mask.

EJEMPLO

TRANSFORMADA Z MODIFICADA



$$\frac{C(z)}{R(z)} = \frac{1.5\ HG(z)}{1+1.5\ HG(z)}$$

SI $H(s) = \dfrac{1-e^{-sT}}{s}$

$G(s) = \dfrac{e^{-0.4s}}{s+1}$

$$HG(z) = \mathcal{Z}\left[\frac{1-e^{-sT}}{s}\cdot\frac{e^{-0.4s}}{s+1}\right]$$

$$HG(z) = (1-z^{-1})\,\mathcal{Z}\left[\frac{e^{-0.4s}}{s(s+1)}\right]$$

$$HG(z) = (1-z^{-1})\,\mathcal{Z}m\left[\frac{1}{s(s+1)}\right].$$

DONDE $m = 1-\theta/T = 1-0.4/1 = 0.6$

⇨ $$HG(z) = (1-z^{-1})\,\mathcal{Z}m\left[\frac{1}{s}-\frac{1}{s+1}\right]$$

$$HG(z) = (1-z^{-1})\left[\frac{z^{-1}}{1-z^{-1}} - \frac{e^{-mT}z^{-1}}{1-e^{-T}z^{-1}}\right]$$

$$HG(z) = z^{-1}\left[\frac{(1-e^{-mT})+z^{-1}(e^{-mT}-e^{-T})}{1-e^{-T}z^{-1}}\right]$$

$$HG(z) = \frac{z^{-1}(0.45 + 0.182 z^{-1})}{1 - 0.368 z^{-1}}$$

$$\Rightarrow \quad \frac{C(z)}{R(z)} = \frac{1.5 z^{-1}(0.45 + 0.182 z^{-1})}{1 - 0.368 z^{-1} + 1.5 z^{-1} (0.45}$$

$$\frac{C(z)}{R(z)} = \frac{z^{-1}(1.125 + 0.455 z^{-1})}{1 + 0.757 z^{-1} + 0.455 z^{-2}}$$

$$\Rightarrow \quad C_n + 0.757 C_{n-1} + 0.455 C_{n-2}$$

$$= R_{n-1}\, 1.125 + 0.455 R_{n-2}$$

Directorio de Asistentes al Curso: INGENIERIA DE CONTROL DIGITAL DE PROCESOS Y SUS APLICACIONES (del 11 de Noviembre al 8 de Dic.78)

1. NICOLAS CALVA TAPIA
Nicolás Bravo 17-1
Martín Carrera, Z.P.14

2. ALEJANDRO ESTRADA GARCIA
S.A.R.H.
Dir.Gral. de Grande Irrigación
Subdirección de Promoción y Programas
Reforma 35-11°
México 1, D.F.
Tel. 591.03.83

Niño Perdido 537-4
Col. Narvarte
Z.P.12
Tel. 538.17.93

3. JOAQUIN FERRER B.
Constituyentes 900
Lomas Altas
Z.P.7
Tel. 570.20.33

4. ALBERTO AUGUSTO FUENTES HERRERA
PEMEX
Analista Programador
Av.Marina Nal. 329
México ,D.F.
Tel. 250.36.22

Necasa 125
Col. Industrial
Z.P.14
Tel. 537.04.21

5. OSCAR GOJON ZORRILLA
INSTITUTO DE INVESTIGACIONES ELECTRICAS
Shakesperare 6
México 5, D.F.
Tel. 525.35.77

Delfines 19
Fuentes de Satélite, Edo. de Méx.

6. RAFAEL HAM YEE
FACULTAD DE INGENIERIA
AYUDANTE DE PROFESOR
U.N.A.M.

Heróes Ferrocarrileros 12-5
Col. Guerrero
México 3, D.F.

7. PABLO JARDON HERRERA
CIA. DE LUZ Y FUERZA DEL CENTRO S.A.
MELCHOR OCAMPO 171
MEXICO, D.F.
TEL. 518.00.80 Ext.213

Hidalgo 28
Col. Tlacopac
México 20, DF

8. FRANCISCO LARA OCOA
INST. DE INVESTIGACIONES BIOMEDICAS
INVESTIGADOR
U.N.A.M.

Barandales 58
Jardines del Sur
México, D.F.
Tel. 548.40.73

9. JUAN MANUEL LICONA DELEON
S.A.R.H.
DEPARTAMENTO ELECTROMECANICA
IGNACIO RAMIREZ 20
MEXICO, D.F.
TEL. 566.37.55

Cerro de San Andrés 197
Campestre Churubusco
México 21, D.F.
Tel. 549.42.35

10. EDMUNDO OLLERVIDES VALDES
INST. DE INVESTIGACIONES
ELECTRICAS
INTERNADO PALMIRA
CUERNAVACA, MORELOS
TEL. 4.20.88

Potrero Verde 300 E-303
Fracc. Jacarandas
Cuernavaca, Mor.

11. LUIS REYNOSO J.
HOYEYWELL S.A. DE C.V.
AV. CONSTITUYENTES 900
LOMAS ALTAS
MEXICO, D.F.
TEL. 570.20.33

Aver 17
Col. Vallejo
México, D.F.
TEL. 355.74.49

12. LUIS R. RIGO LEMINI
INST. MEX. DEL PETROLEO
AV. 100 METROS NO.152
MEXICO 14, D.F.
TEL. 567.66.00

Monte Albán 17
Col. Narvarte
México 12, D.F.

13. SERGIO ROMERO PAREDES
S.T.C. METRO
Delicias No.67
México, D.F.
Tel. 521.86.20 Ext.505

Av. Univ. No.2014 E. Argentina D-104
Col. Romero de Terreros
México 21, D.F.
Tel. 658.01.96

14. SALVADOR SAHAGUN VALDIVIA
ESC. SUP. DE ING. MEC. Y ELEC.
UNIDAD PROFESIONAL ZACATENCO
MEXICO 14, D.F.

Av. del Taller Retorno 38 Casa 24-A
Col. Jardín Balbuena
México 9, D.F.
Tel. 552.23.59

15. SERGIO A. SANTIAGO RIVERA
POLICODOR MANTENIMIENTO ELECTRONICO
Marsella 89
México 6, D.F.
Tel. 533.23.85

Hidalgo 216-201
Iztapalapa
México 13, D.F.

16. ALEJANDRO SERRANO FELIX
FAC. DE ING.
U.N.A.M.
MEXICO 20, D.F.

Toribio Medina 108
Col. Algarin
México, D.F.
Tel. 530.53.03

17. FRANCISCO DANIEL SORIA VILLEGAS
FACULTAD DE INGENIERIA
PROF. TIEMPO COMPLETO
U.N.AM.
MEXICO 20, D.F.

Calle Sn. Jorge Manzana 824 Lote 2
Col. Sta. Ursula Coapa
México 22, D.F.

18. JOSE LUIS VAZQUEZ G
CERVECERIA MOCTEZUMA S.A.
Sur 10 y Pte. 9
Orizaba, Ver.
Tel. 530.33

Av. Ote. 8 No. 845
Orizaba, Ver.
Tel. 53716

19. ALEJANDRO VILLA Y VILLA
S.A.R.H.
SUB. DE PROYECTOS DE LA DIR.
GENERAL DE GRANDE IRRIGACION
SIERRA GORDA NO.23
MEXICO 10, D.F.
TEL. 540.00.52

Acueducto 13
Col. Los Fresnos
Naucalpan, Edo. de Méx.
Tel. 572.53.35

20. ANGEL YAÑEZ HERNANDEZ
S.A.R.H.
SUBSECRETARIA DE INFRAESTRUCTURA HIDRAULICA
DIRECCION GRAL. DE CAPTACION Y CONDUCCIONES DE AGUA
IGNACIO RAMIREZ NO.20
MEXICO, D.F.
TEL. 566.37.55

Valle del Mexquital 168
Col. Valle de Aragón
Edo. de México