INSCRIPCIONES

CENTRO DE EDUCACION CONTINUA DE LA DIVISION DE ESTUDIOS SUPERIORES DE LA FACULTAD DE INGENIERIA, U. N. A. M.

Palacio de Minería Calle de Tacuba No. 5
México 1, D F.

Horario de oficinas:
Lunes a viernes de 9 a 18 h

Cuota de inscripción $ 4,000.00

La cuota de inscripción incluye:
- una carpeta con las notas de los profesores
- bibliografía sobre el tema
- servicio de cafetería
- comidas

Requisitos

- Pagar la cuota de inscripción o traer oficio de la empresa o institución que ampare su inscripción, a más tardar una semana antes del inicio del curso
- Llenar la solicitud de inscripción
- Entregar copia de la cédula profesional

Para mayores informes hablar a los teléfonos

521-40-20 521-73-35 512-31-23

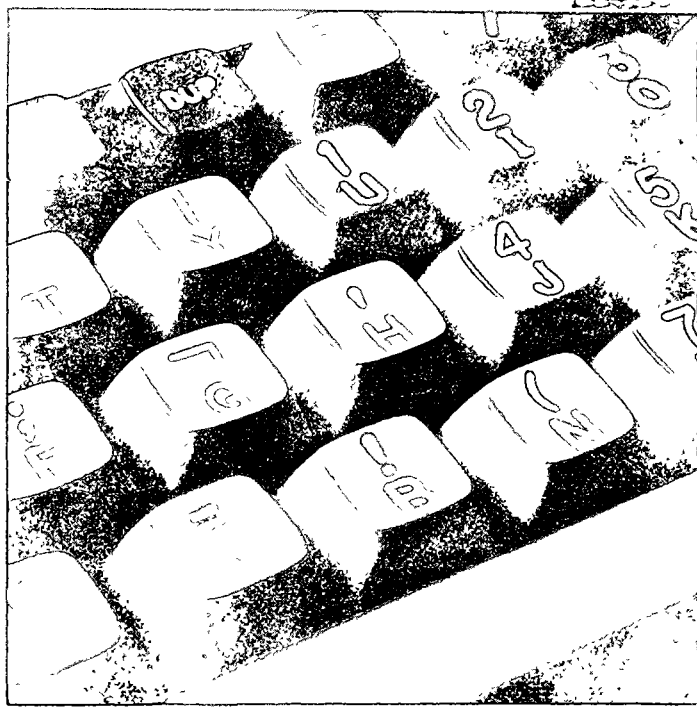
CONSTANCIA DE ASISTENCIA

Las autoridades de la Facultad de Ingeniería de la U N A M., otorgaran una constancia de asistencia a los participantes que concurran regularmente y que realicen los trabajos que se les asignen durante el curso

centro de educación continua
división de estudios superiores
facultad de ingeniería, u n a m

Palacio de Minería
Calle de Tacuba No. 5
México 1, D.F.

77



INGENIERIA DE CONTROL DIGITAL DE PROCESOS Y APLICACIONES

Duración: 44 h
Fechas: 7, 8, 14, 15, 21, 22, 28 y 29 de octubre
Horario: viernes de 17 a 21 h; sábados de 9 a 13 y de 14 a 18 h

Coordinadores: Dr. Víctor Gerez Greiser y M en C. Marcial Portilla Robertson

En colaboración con la Asociación de Ingenieros Universitarios Mecánicos Electricistas, A C

centro de educación continua
división de estudios superiores
facultad de ingeniería, u n a m

PRESENTACION DEL CURSO

Hoy en día se presenta con mayor frecuencia la necesidad de controlar complejos sistemas. Esta función de control tiene que realizarse empleando dispositivos digitales para poder monitorear y controlar varias variables simultáneamente, y lograr de esta forma minimizar costos de producción, garantizando una adecuada continuidad en el servicio y una calidad adecuada en el servicio o producto.

OBJETIVOS

Se pretende que con este curso se adquieran los conocimientos necesarios para entender el funcionamiento de estos sistemas, se puedan modificar las estrategias de control en sistemas ya operando y se puedan diseñar nuevos sistemas.

A QUIEN VA DIRIGIDO

Este curso está dirigido a profesionales que deseen conocer las bases y fundamentos de la metodología que se emplea para implementar sistemas de control digital directo tanto en la industria de procesos como de manufactura y eléctrica de servicio público.

TEMARIO

1. INTRODUCCION

   Control digital directo y control convencional.
   Adquisición de datos.
   Control digital directo
   Control supervisorio.
   Control jerárquico.

2. HARDWARE DEL EQUIPO DIGITAL.

   Arquitectura de las minicomputadoras.
   Equipos de entrada y salida.
   Interfases.

3. SOFTWARE

   El ensamblador.
   Lenguajes de programación.
   El ejecutivo.

4. CONTROL SUPERVISORIO.

   Modelos de control supervisorio.
   Técnicas de optimización.
   Aplicación a una columna de destilación.

5. CONSIDERACIONES EN EL DOMINIO DE LA FRECUENCIA.

   La transformada z.
   Muestreadores.

6. ALGORITMOS DE CONTROL.

   Diseño de algoritmos.
   Técnicas de ajuste.
   Selección del tiempo de muestreo.
   Técnicas de identificación.
   Técnicas avanzadas de control.
   Control óptimo.

7. APLICACIONES.

   Aplicación en la industria eléctrica de servicio público.
   Aplicaciones en la industria del vidrio.
   Aplicaciones en la industria siderúrgica.
   Aplicaciones en la industria del cemento.
   Aplicaciones en la industria del papel.

PROFESORES

DR. VICTOR GEREZ GREISER

M. en C. MAURICIO MIER

M. en C. MARCIAL PORTILLA ROBERTSON

INGENIERIA DE CONTROL DIGITAL DE PROCESOS Y APLICACIONES

(Octubre 197)

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	Algoritmos de Control	Dr. Víctor Gerez Greiser
Oct. 22	9 a 13 h	Algoritmos 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

ING. RAFAEL LOPEZ LOPEZ

'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

OCTUBRE 1977.

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

EL PROBLEMA DEL CONTROL DE PROCESOS

Ejemplo: SISTEMA ELECT. DE POT.

Función del Sistema: Suministrar:

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

con Restricciones sobre:

Voltaje V

Frecuencia f

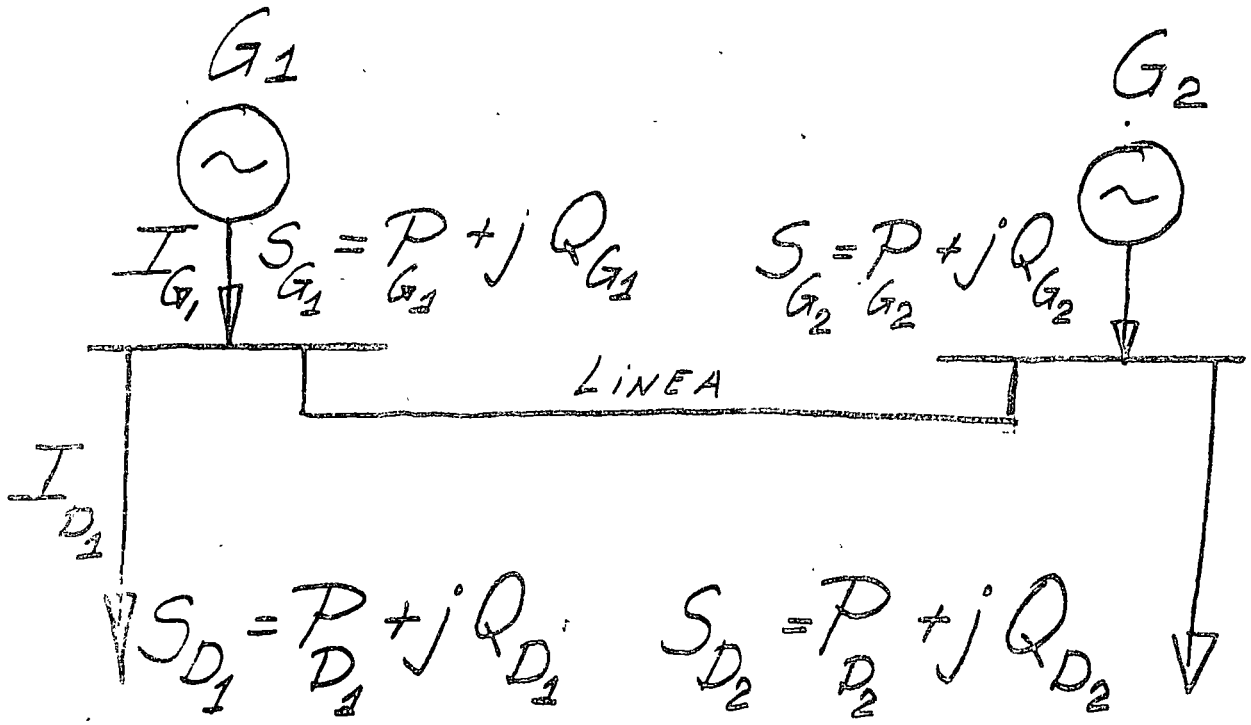
Problema de operación:

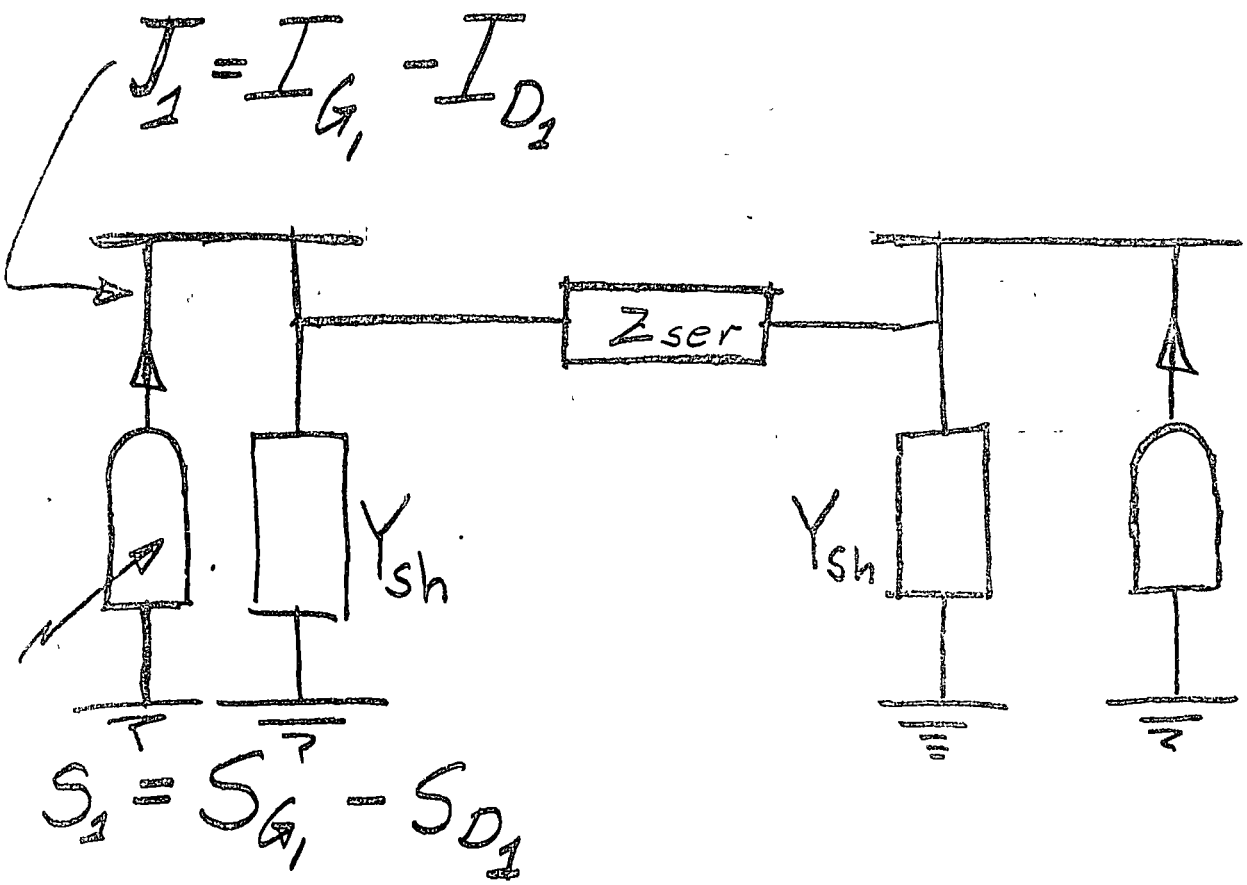
Modelar el sistema

Desarrollar estrategias de operación optima

Implementarlas (Control)

MODELADO DEL SISTEMA:





Net bus power:

$$S_1 = P_1 + jQ_1$$

$$S_2 = P_2 + jQ_2$$

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

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

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

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

$$S = V I^*$$

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

(4)

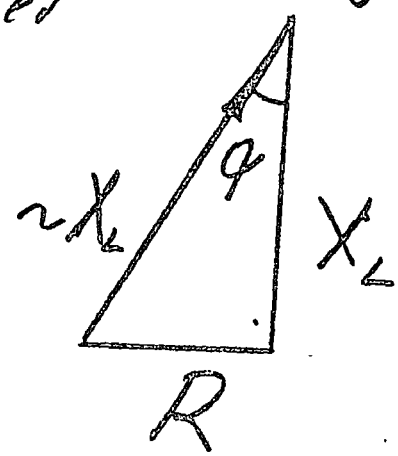
$$\frac{S_1^*}{V_1^*} = V_1 Y_{sh} + \frac{V_2 - V_1}{Z_{ser}} \quad (2)$$

otra para bus (2)

Simplificaciones:

$$Y_{sh} = \frac{f}{X_c}$$ capacitivo serie (3)

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



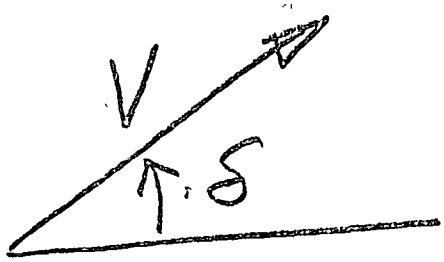
$X_L \gg R$

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

Tensiones de Bus: (5)

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

$$V_2 = |V_2| \angle \delta_2$$



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
$$\delta_1 - \delta_2 = \delta_{12} = \delta$$
6) Doce variables
Cuatro ecuaciones
⇓
Hay que predefinir 8

Clasificacion de variables

a) Fuera de control o disturbios

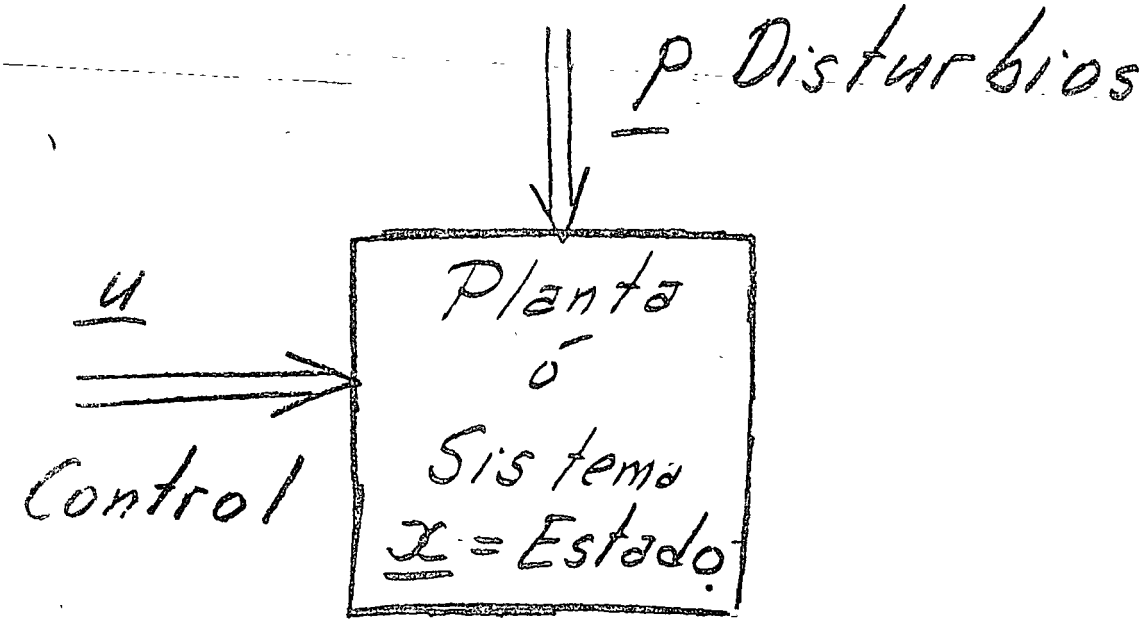
$$\underline{p} = \begin{bmatrix} P_{D_1} \\ Q_{D_1} \\ P_{D_2} \\ Q_{D_2} \end{bmatrix}$$

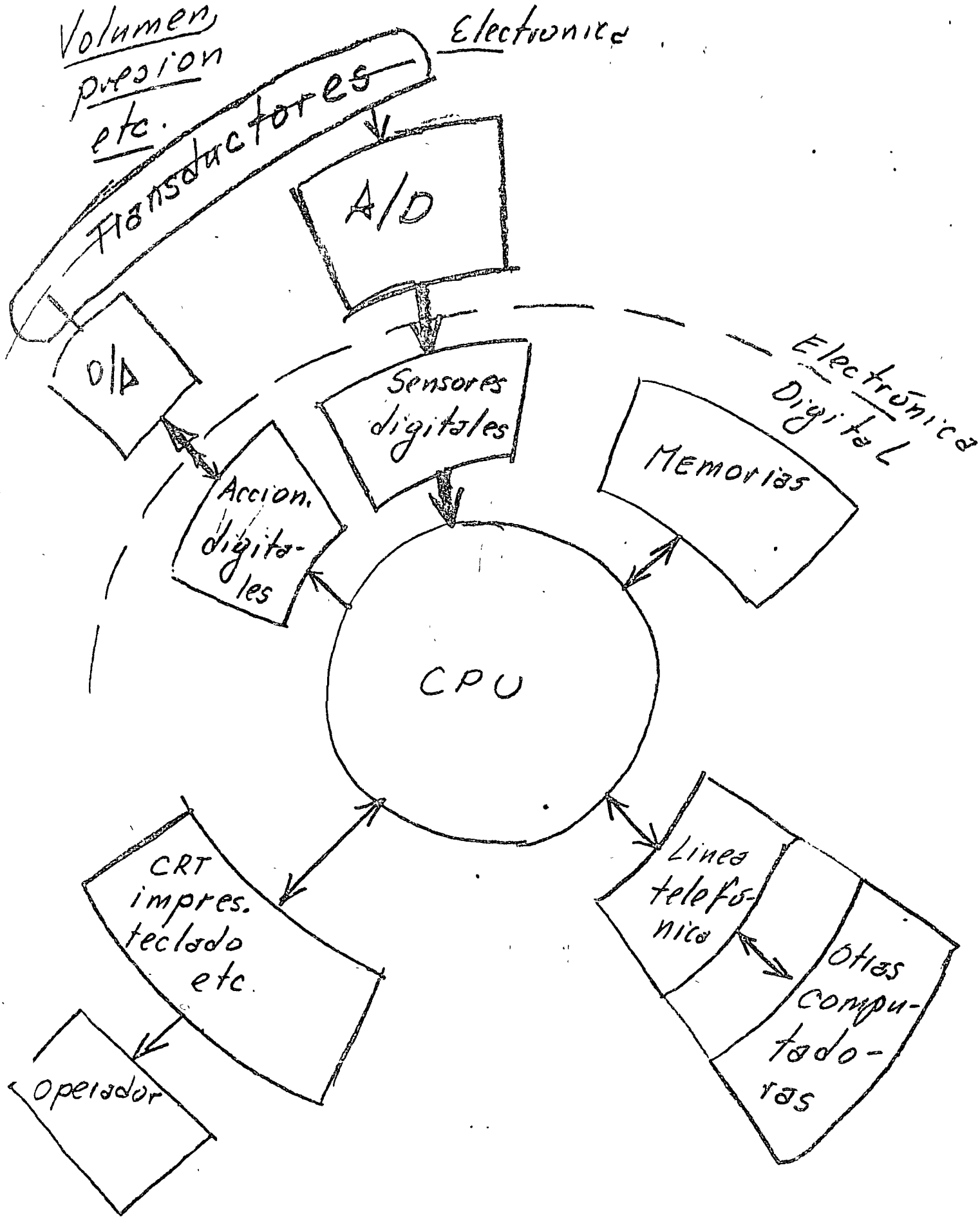
b) Control o Manipuladas

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

c) De estado

$$\underline{x} = \begin{bmatrix} \delta_1 \\ |V_1| \\ \delta_2 \\ |V_2| \end{bmatrix}$$




Volumen
presion
etc.
Transductores
Electronica
A/D
D/A
Sensores digitales
Accion. digitales
Memorias
Electrónica Digital
CPU
CRT impres. teclado etc
Operador
Linea telefonica
Otras computadoras

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

INGENIERIA DE CONTROL DIGITAL DE PROCESOS Y APLICACIONES

H A R D W A R E

S O F T W A R E

DR. VICTOR GEREZ GRELSER

M. EN C. MARCIAL PORTILLA ROBERTSON

OCTUBRE DE 1977.

PALACIO DE MINERIA
Tacuba 5, primer piso. México 1, D F.

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.

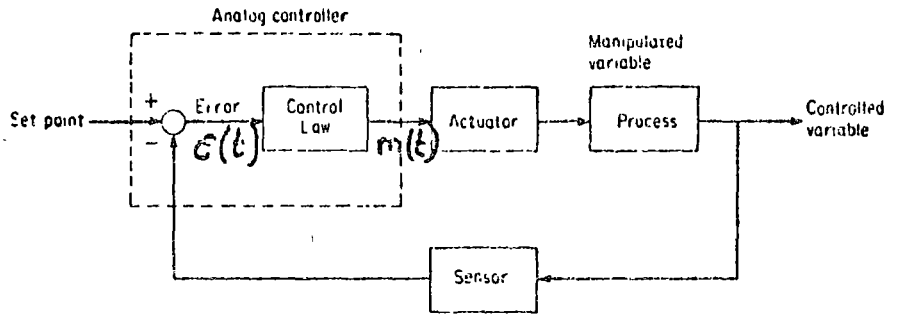
3. Variables controladas. Estas variables son las que determinan la operación de la planta. Son aquellas para las cuales se diseña una estrategia de control con objeto de mantenerlas dentro de ciertos límites. En nuestro ejemplo de sistema eléctrico de potencia son éstas la tensión y la frecuencia.

4. Variables intermedias. En diferentes puntos del proceso aparecen 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



SISTEMA REALIMENTADO.

de un sistema de control es la realimentación.

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.

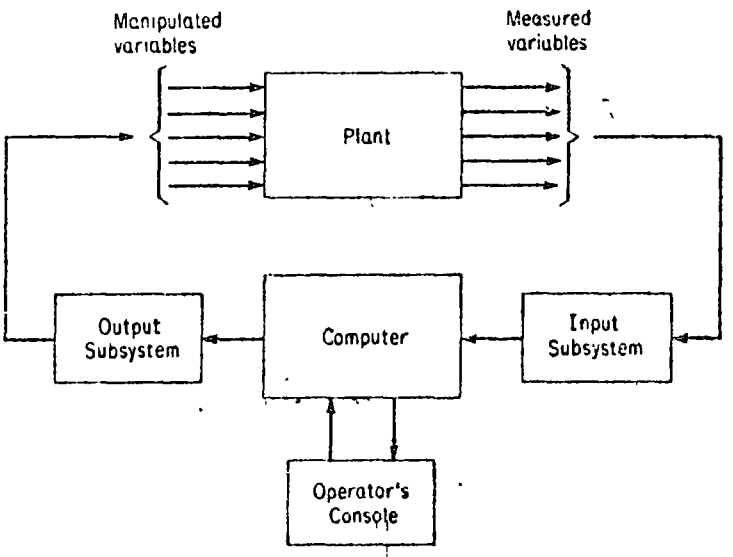
Los registros que genera la computadora son sin embargo impor-

tantes 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.

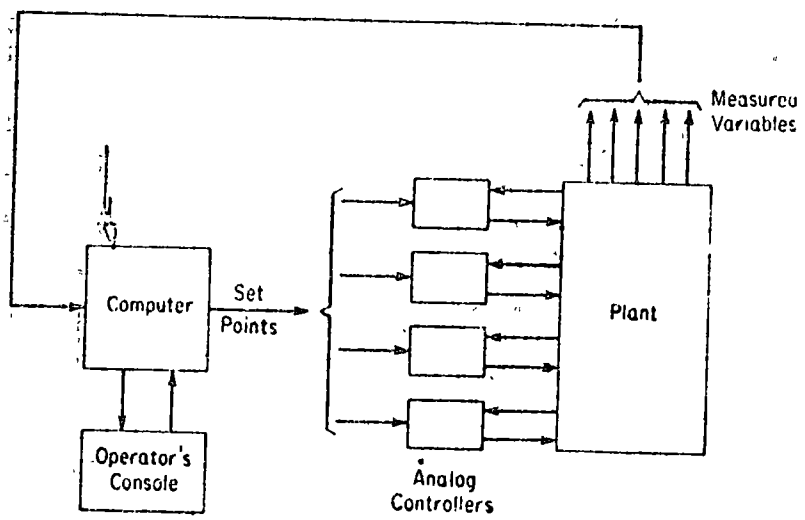
Si puede justificarse la adquisición del equipo digital por razones adicionales a las de implementación de una ley de control proporcional, integral y diferencial, debemos mencionar que empleando técnicas digitales es posible obtener con el algoritmo anterior mejor respuesta que con su versión analógica.

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.

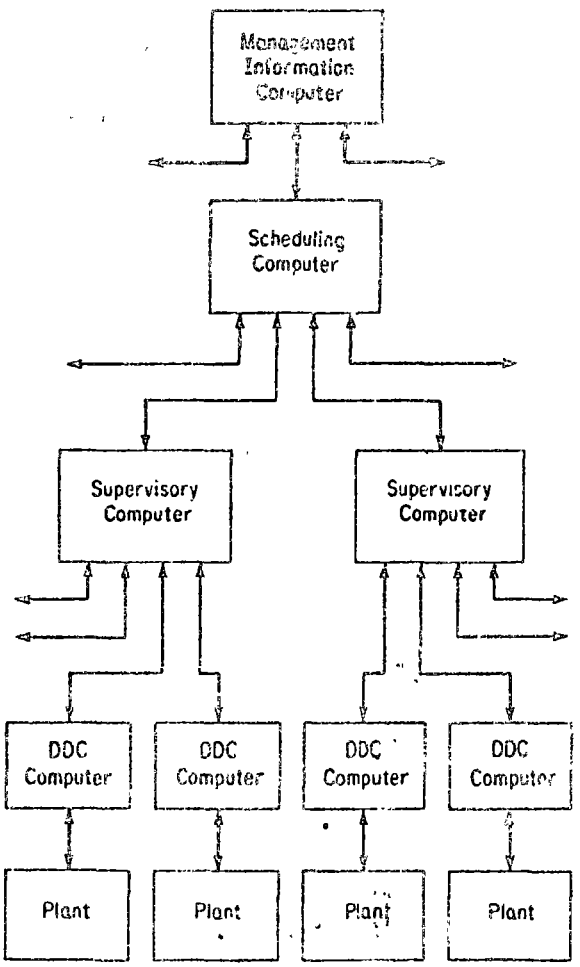
Debemos hacer incapié que la limitación principal para implementar este tipo de control es la disponibilidad de un buen modelo matemático de la planta o sistema que se desea controlar.



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.

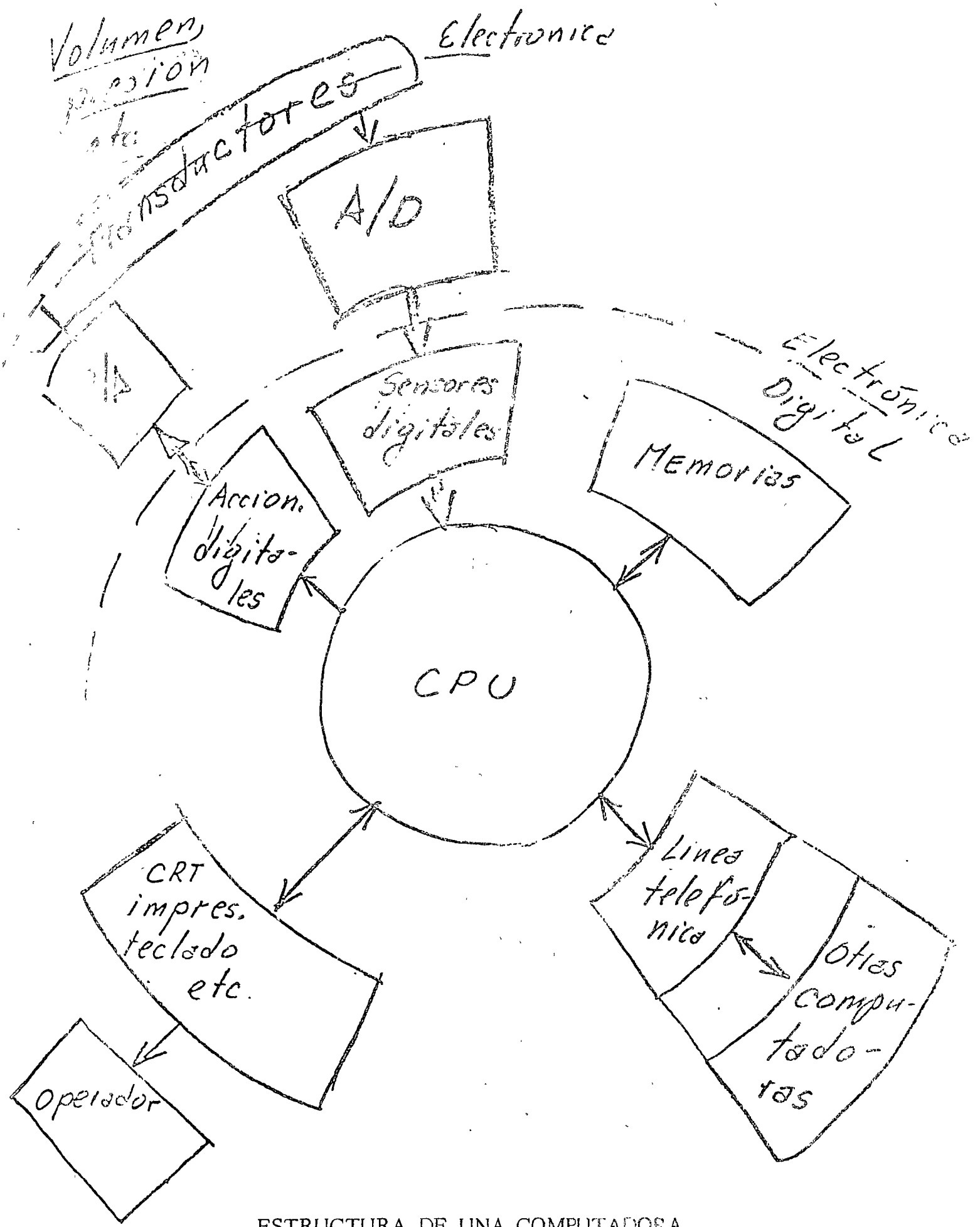
LA COMPUTADORA DIGITAL COMO ELEMENTO DE CONTROL.

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

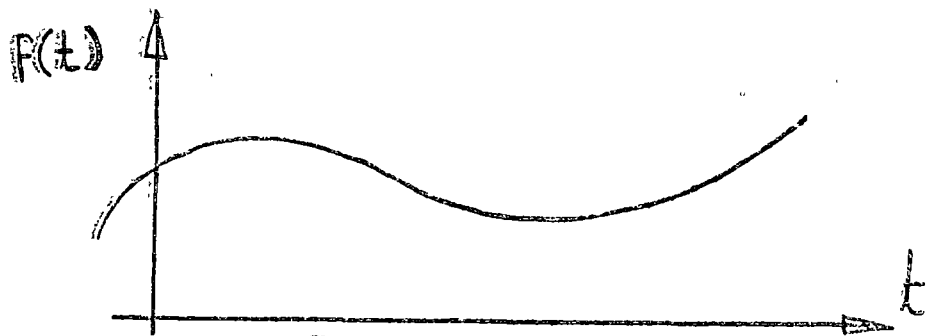


ESTRUCTURA DE UNA COMPUTADORA

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

SEÑALES ANALOGICAS Y SEÑALES DIGITALES.

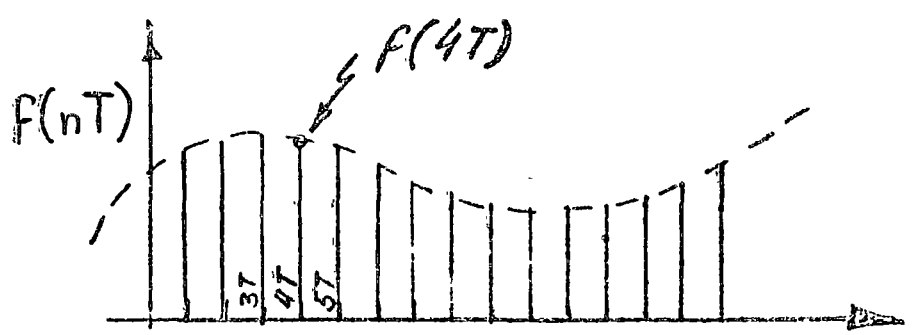
La figura muestra una señal analógica que exceptuando momentos de conexión o desconexión generalmente es contínua.



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.



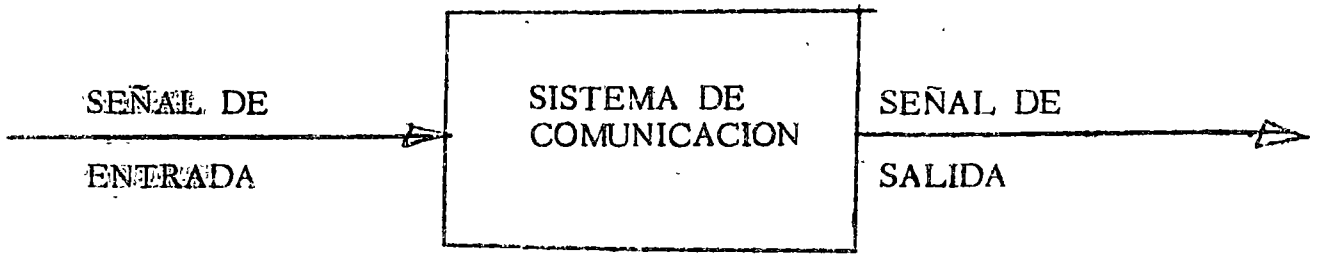
SEÑAL DISCRETA.

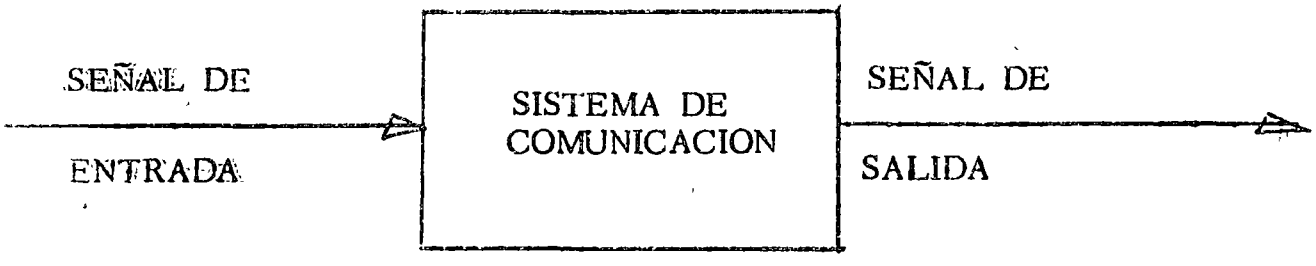
Un convertidor analógico digital transforma estos valores discretos en valores binarios, octales o de alguna otra base según el sistema digital que se estuviese empleando.

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 receptor 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.

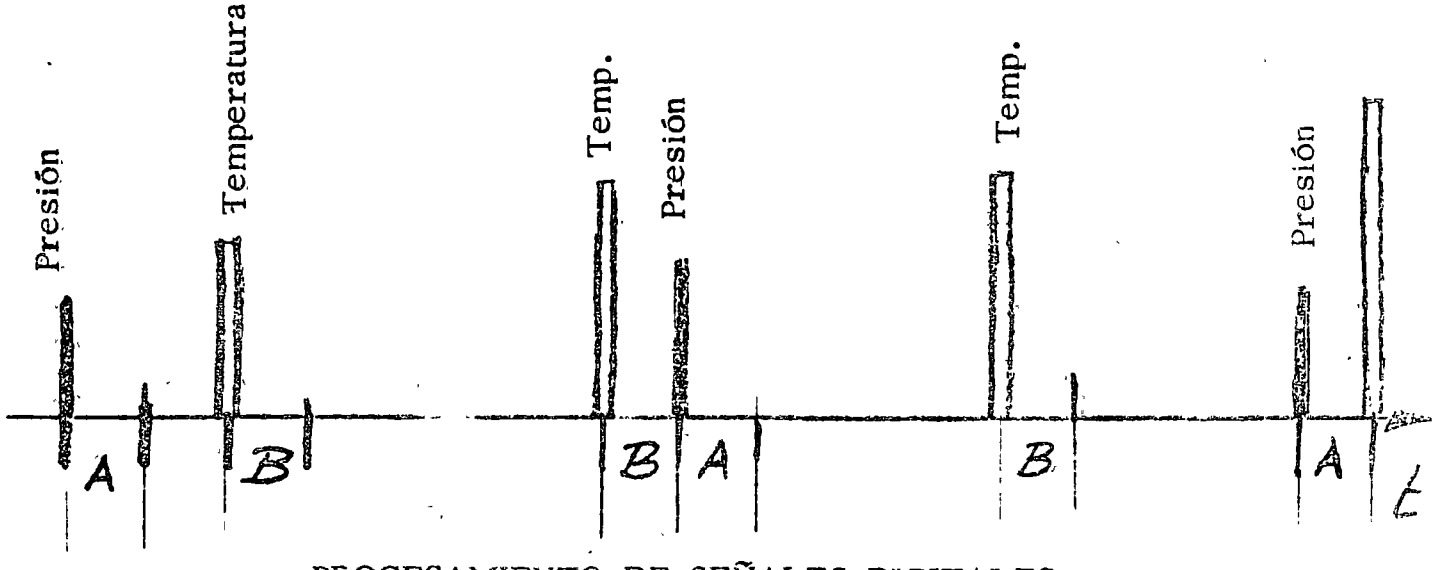
Como en la computadora además se trabaja a una enorme velocidad muy superior a la de muestreo entre operación de muestreo existe tiempo para realizar cálculos e inclusive muestrear otras cantidades.





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.

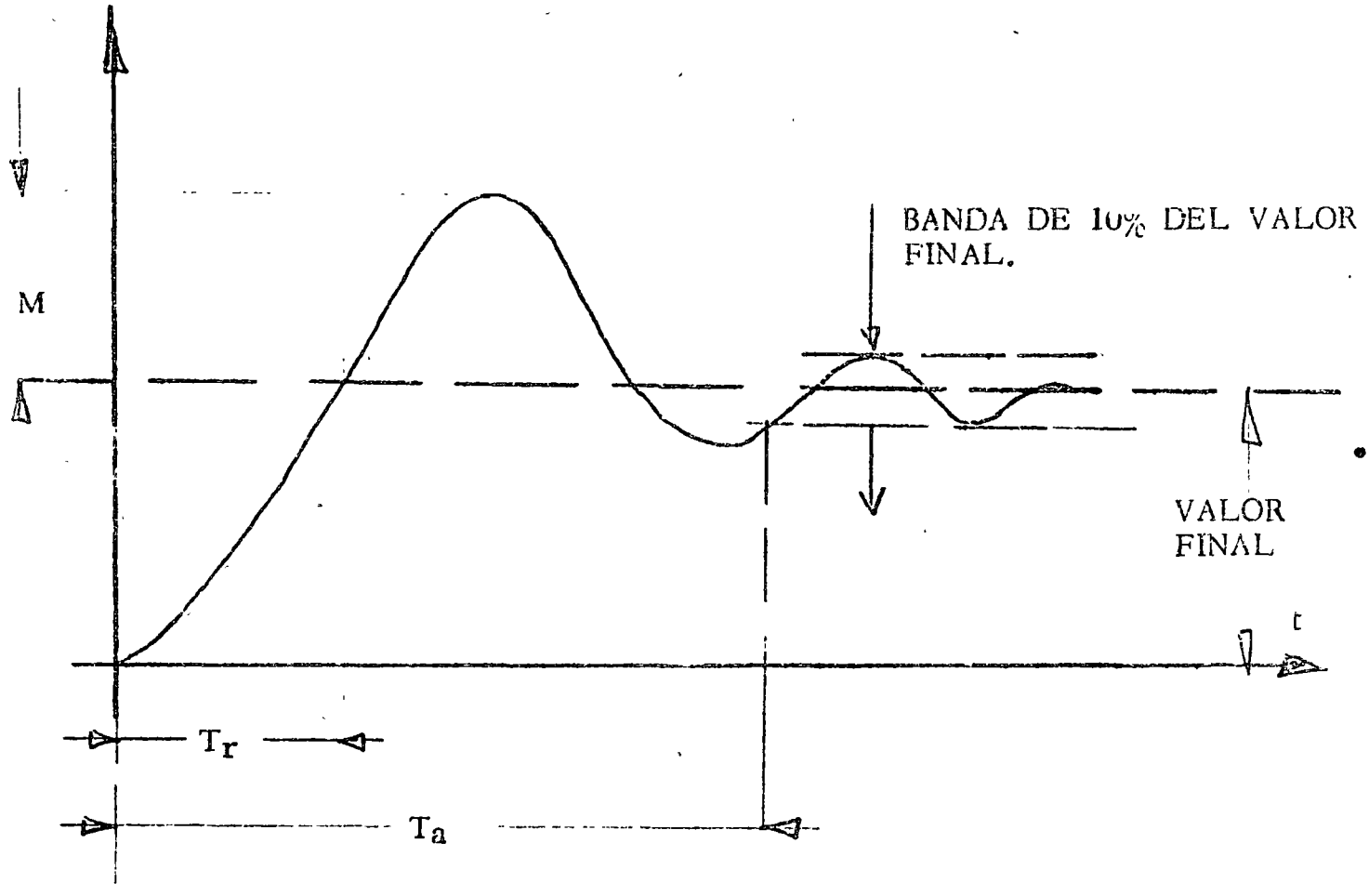
A = Intervalo de tiempo que se requiere para procesar la informaciónsobre presión y enviar una señal de telecomando
B = Intervalo correspondiente a la señal de temperatura.

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 difícil 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 mi-

nimo 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 continuo 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 aplicación de 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
Potencia Reactiva Q
} que demandan los usuarios

con Restricciones sobre:

Voltaje V

Frecuencia f

Problema de operación:

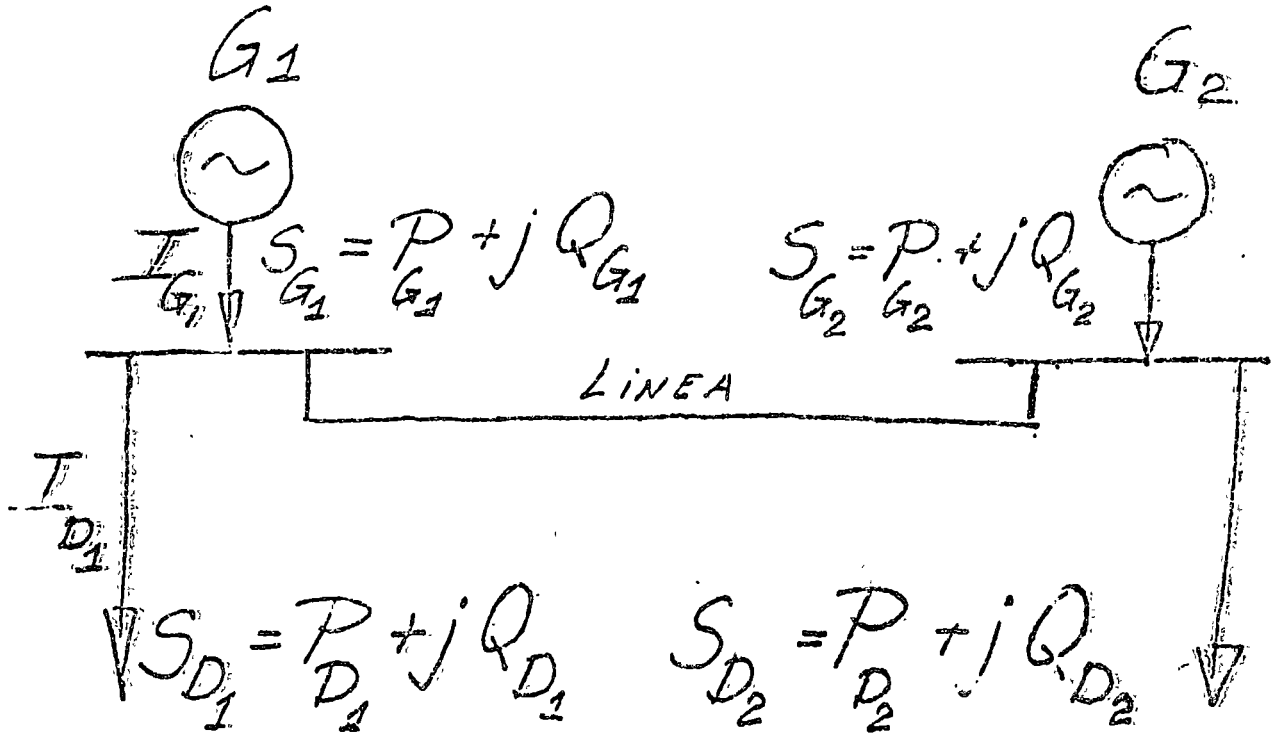
Modelar el sistema

Desarrollar estrategias de operación optima

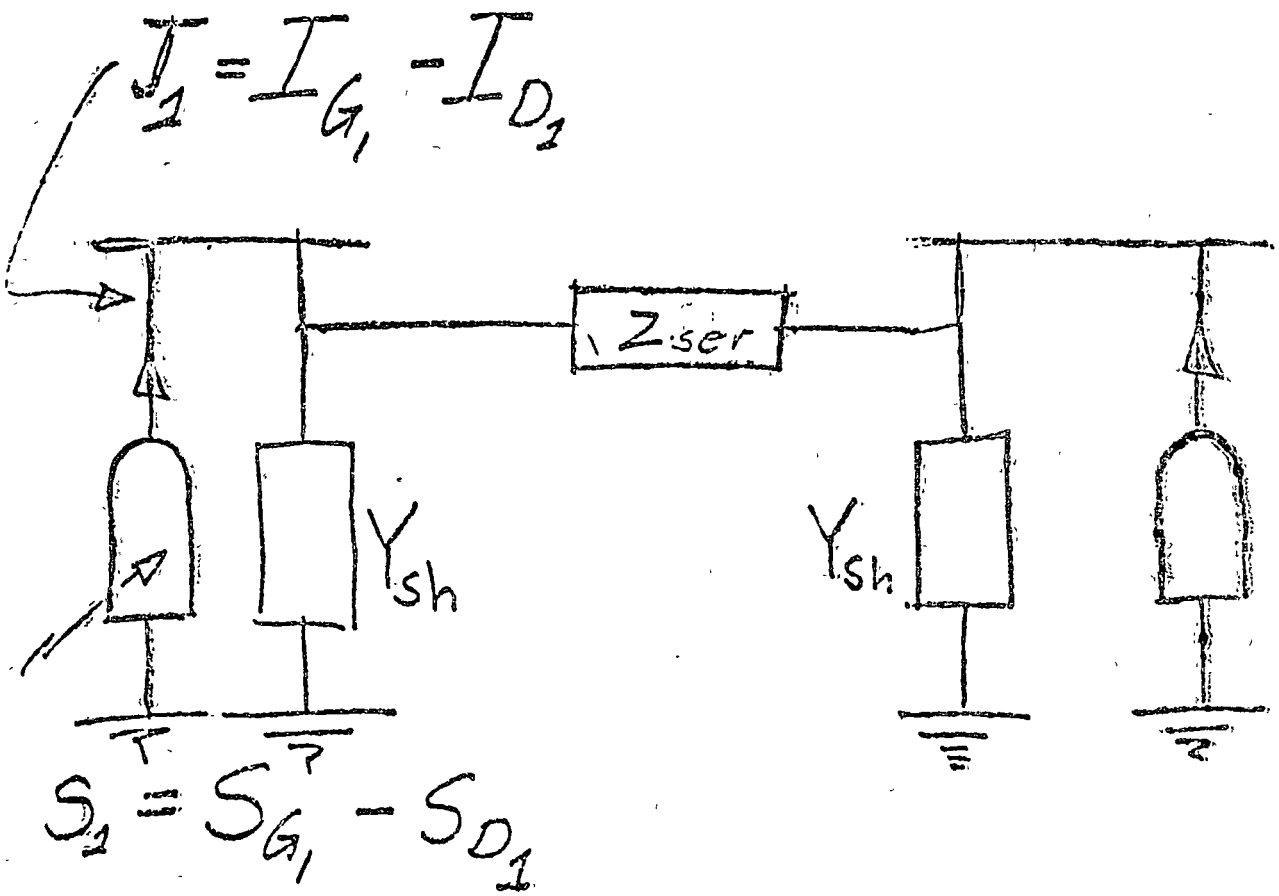
Implementarlas (control)

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 ...rencia neta del bus (lugar donde se interconectan generadores cargas y ...neas.

$$S_1 = P_1 + jQ_1$$

$$S_2 = P_2 + jQ_2$$

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

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

Potencia Real Generada =

Consumo + Pérdidas

$(f = cst.)$

Potencia Reactiva Generada =

Consumo + Pérdidas

$(V = cst.)$

POTENCIA COMPLEJA: $S = V I^*$

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

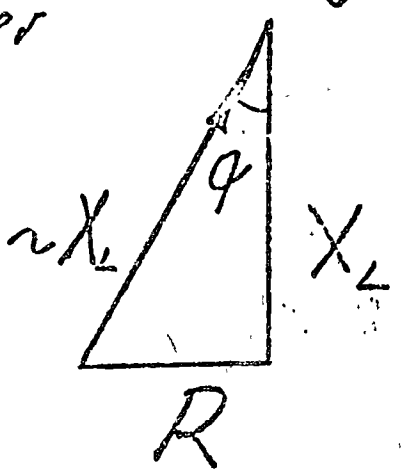
$$\frac{S_1^*}{V_1^*} = V_1 Y_{sh} + \frac{V_2 - V_1}{Z_{ser}} \quad (2)$$

otra para bus (2)

Simplificaciones:

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

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



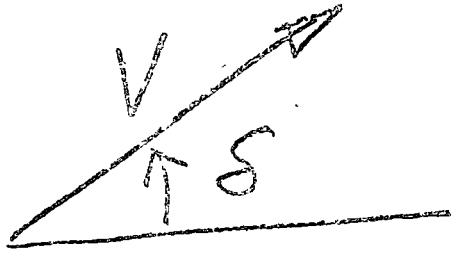
$X_L \gg R$

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

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) ⟹

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 $\mathcal{L}$ (Estado estable)
5) Siempre aparece
$$\delta_1 - \delta_2 = \delta_{12} = \delta$$
6) Doce variables
Cuatro ecuaciones
⇓
Hay que predefinir $\mathcal{E}$

Clasificación de variables

a) Fuera de control o disturbios

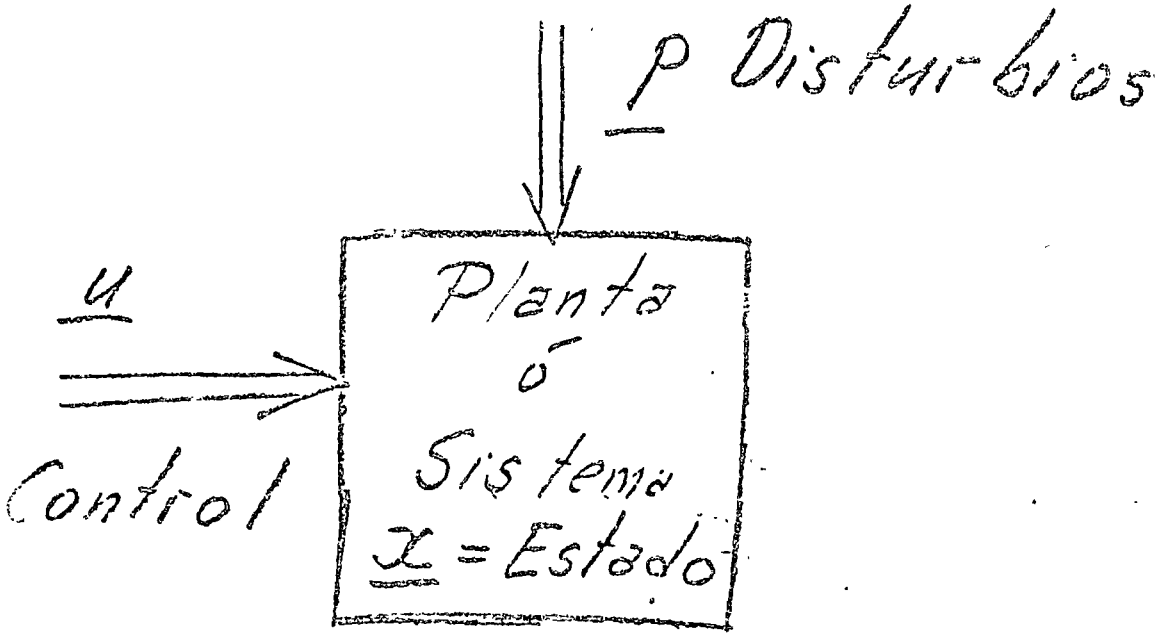
$$\underline{p} = \begin{bmatrix} P_{D_1} \\ Q_{D_1} \\ P_{D_2} \\ Q_{D_2} \end{bmatrix}$$

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

$$\underline{x} = \begin{bmatrix} \delta_1 \\ |V_1| \\ \delta_2 \\ |V_2| \end{bmatrix}$$

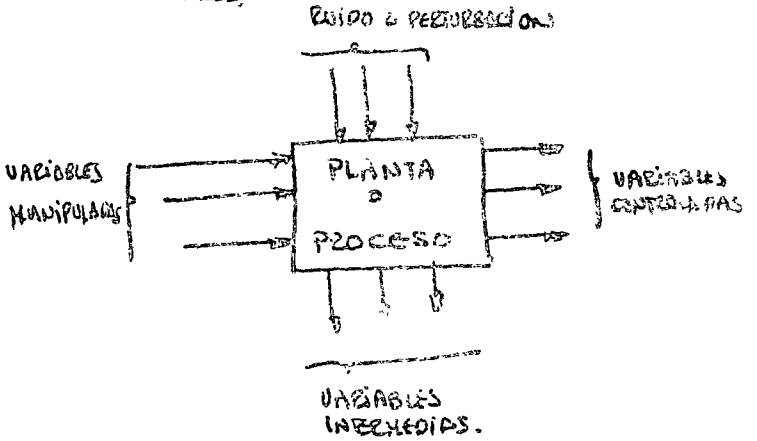


CONTROL DIGITAL DIRECTO

EXISTE UNA GRAN DISCREPANCIA ENTRE LA TEORIA DE CONTROL Y LAS APLICACIONES DE ESTA TEORIA. EN ESTA ULTIMA PARTE DEL CURSO NOS ABOCAREMOS 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 A CABO CON EXITO ① y ②. EN TODO PROCESO EXISTEN LOS SIGUIENTES TIPOS DE VARIABLES



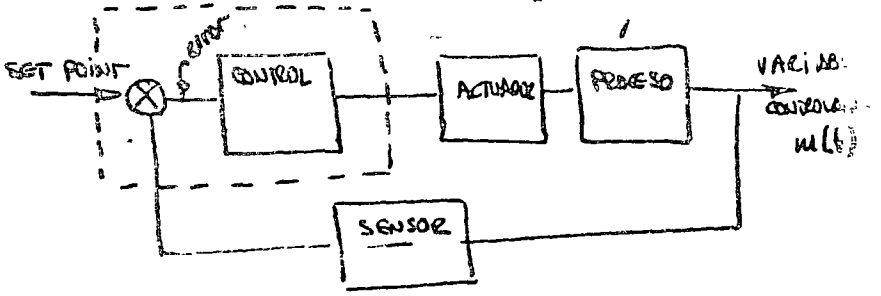
1.- VARIABLES MANIPULADAS.— ESTAS VARIABLES SON LA ENTRADA A DICHO PROCESO: VAPOR, AGUA, MATERIA PRIMA ETC. Y CUYOS VALORES DE ENTRADA NECESITAN SER MANIPULADOS

2.- RUIDO O PERTURBACION.— ESTAS VARIABLES AFECTAN LA OPERACION DEL PROCESO Y NO ESTAN SUJETAS A CONTROL. ES — CUALQUIER SEÑAL ALEATORIA, INHERENTE EN EL MATERIAL 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 MANTENERLAS ESTAS VARIABLES DENTRO DE SU SET POINT O RANGO DE ACCION

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

SISTEMAS DE CONTROL CONVENCIONALES.



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

② F. COREN
[illegible] MINICOMPUTER APPLICATIONS

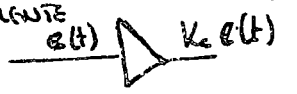
COMO YA SE VIO EN CAPITULOS PASADOS EXISTEN TRES ACCIONES BASICAS DE CONTROL (Y SUS COMBINACIONES), ESTAS ACCIONES SON:

CONTROL PROPORCIONAL

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

K_c = GANANCIA PROPORCIONAL

ESTE USUALMENTE SE UTILIZA UN AMPLIFICADOR PARA CONTROLAR PROPORCIONALMENTE

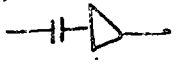


CONTROL DERIVATIVO

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

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

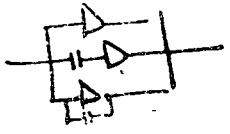
$$m(t) = \frac{K_c}{T_i} \int_0^t e(\sigma)\, d\sigma$$

T_i = TIEMPO DE INTEGRACION O "RESET"

CONTROL PID

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

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 VALORES DE K_c T_i, T_d SON AJUSTADOS POR PRUEBA Y ERROR



EN GENERAL EL 75% DE LAS APLICACIONES UTILIZAN CONTROLES PI DEBIDO A LA DIFICULTAD DE AJUSTAR EL CONTROL PID

EN PLANTAS EXISTEN DESDE UNOS CUANTOS CONTROLES PI HASTA MILES DE ESTOS. DESDE LOS 50 LOS CONTROLES NEUMATICOS FUERON CAMBIADOS POR CONTROLES ELECTRICOS Y ELECTRONICOS, Y EN LOS 70 POR MINICOMPUTADORAS.

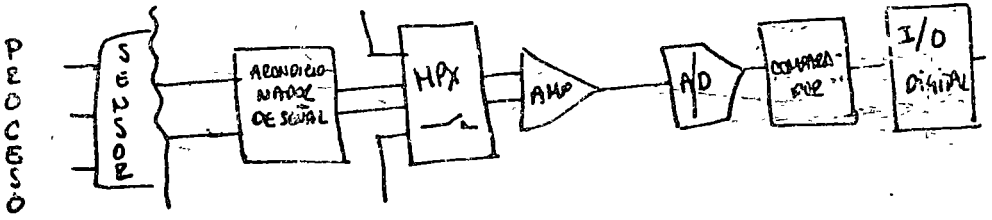
HAY QUE HACER NOTAR QUE EL EQUIPO AUXILIAR PUEDE LLEGAR A COSTAR HASTA EL 80% DEL COSTO TOTAL DEL SISTEMA.

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



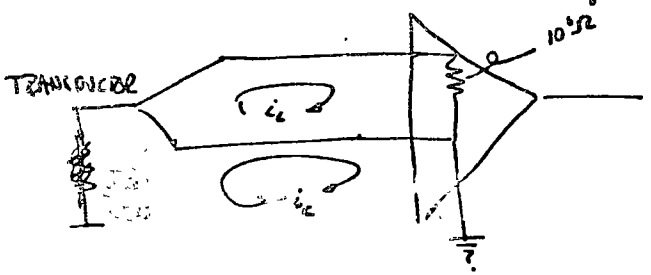
LAS SEÑALES SE CLASIFICAN COMO SIGUE:

1.- DE BAJO NIVEL (≤ 100 MICROVOLTS μV) ESTAS SEÑALES SE RECIBEN DE TERMOPARES, TERMOMETROS DE RESISTENCIA, MEDIDORES DE COMPRESION DEFLEXION ETC.

2.- DE ALTO NIVEL (≥ 10V) 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 DISTORCION Y REQUIEREN ESPECIAL CUIDADO. LAS PUNTAS DEL TERMO-PAR GENERALMENTE ESTAN RECUBIERTAS Y PROTEGIDAS, ESTAS NO DEBEN ESTAR CERCA DE CIRCUITOS R-C O MOTORES Y GENERADORES GRANDES. SE DEBE TENER ESPECIAL CUIDADO EN EL ATERRIZAJE POR SE RECOMIENDA HACER EL ATERRIZAJE EN UN PUNTO Y ESE ES LA COMPUTADORA

CUANDO LA IMPEDANCIA EN LOS AMPLIFICADORES ES MUY GRANDE (10^6 Ω), MUY POCA ONDA CORRIENTE RECIBIRA, 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 UN FILTRO RC. SI LA SEÑAL ES OTRA ESTE ACONDICIONADOR GENERALMENTE TRANSFORMA EN UN VOLTAJE ANTES DE ENTRAR AL MULTIPLEXOR.

MULTIPLEXOR .— EL MULTIPLEXOR ES UN MECANISMO EL CUAL CONECTA UNA DE VARIAS SEÑALES DEL CONVERTIDOR A/D A LA COMPUTADORA (" MUESTREANDO "). PARA LAS SEÑALES DE ALTO NIVEL SE UTILIZAN MULTIPLEXORES ELECTRONICOS, DEBIDO CON MUESTREOS MAYORES DE 10,000 PUNTOS POR SEGUNDO. PARA LAS SEÑALES BAJAS SE UTILIZAN RELAYS MECANICOS, PUES LA DISTORSION DEL EFECTO 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 2084 PUNTOS O PUERTOS, Y SU MUESTREO ES SECUENCIAL. (ES MEJOR QUE EL MAS RAPIDO MUESTREO QUE EL CPU).

AMPLIFICADORES . LOS AMPLIFICADORES "ESCALAN" LA SEÑAL DEL PROCESO (+ o -) CON LA DEL CONVERTIDOR A/D (TIPICAMENTE 1 VOLT.)

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 ALCANCE UN VALOR CERCA DESPUES DE QUE FUE APLICADA UNA NUEVA SEÑAL SE LLAMA "TIEMPO DE ASENTAMIENTO" Y LOS CONVERTIDORES A/D ELECTRONICOS TIENEN UN TIEMPO DE ASENTAMIENTO $\leq 40\ \mu seg$.

COMPARADOR :— EL COMPARADOR QUITA CHAMBA AL CPU, EVITANDO QUE SE DISTRAIGA HACIENDO TAREAS QUE PUEDAN SER EJECUTADAS POR FUERA. COMPARA LA SEÑAL DE ENTRADA CON UNA SEÑAL LIMITE (HIGH, O LOW).

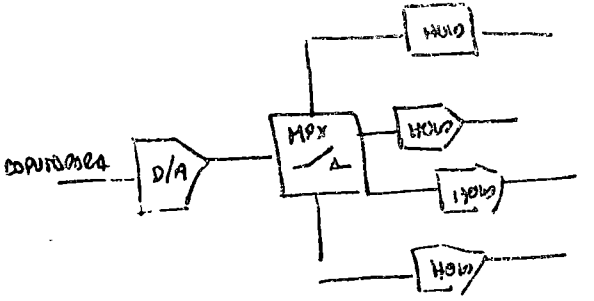
LOS COMPARADORES RESULTAN MUY UTILES EN SISTEMAS QUE HACEN MUESTREOS SECUENCIALES Y QUE TIENEN ~~FEE~~ CANALES DE ACCESO (?) DIRECTO DE MEMORIA, PUES GUARDAN LOS DATOS EN LOCALIDADES PRE ASIGNADAS DE MEMORIA

SI LA SEÑAL SALE DE LOS LIMITES (HIGH O LOW) SE LE LLAMA LA ATENCION AL CPU MEDIANTE 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 LEERA 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 MUX PARA OBTENER VARIAS SALIDAS DE UN CONVERTIDOR D/A. UTILIZA UN "HOLD" PARA RETENER EL VALOR ENTRE MUESTREOS



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

3.- CONTACTOS ON-OFF ~~(START)~~ SON UTILIZADOS PARA ARRANCAR O PARAR BOMBAS, MOTORES ETC, AUNQUE 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, PEDIR MAS DATOS PARA IMPRESION, ETC.

2.- INTERRUPCIONES DEL RELOJ.— ESTAS SE PRESENTAN CUANDO SE REQUIERE UNA SINCRONIA ENTRE LAS OPERACIONES DEL SISTEMA Y EL MUNDO EXTERIOR. LAS INTERRUPCIONES DEL RELOJ SE DAN A CIERTOS INTERVALOS DE TIEMPOS REGULARES. (EJ: 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 A CABO ALGUNA TAREA ESPECIFICA EN CONDICIONES ESPECIALES

EN LOS SISTEMAS EN DONDE SE UTILIZA COO 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 ESPECIFICADO.

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 INSTRUCCION DE UNA INSTRUCCION LLAMADA "RUTINA DE SERVICIO DE INTERRUPCION" (ISR). UNA VEZ SATISFECHA LA INTERRUPCION SE, SE RESTAURAN LOS CONTENIDOS DE LOS REGISTROS RETRABAJO (WR) Y EL CONTENIDO DEL REGISTRO DE DIRECCION (AR) 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 numero 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.

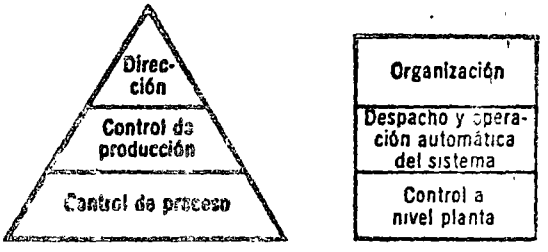


Fig. 2.1.1 Estructura jerárquica del control.

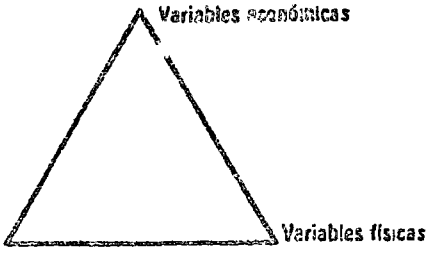
*Pueden distinguirse, en general, tres funciones de control a diferentes niveles. En el primer nivel están aquellas funciones asociadas con el control de las unidades de manufactura, que en el caso de la industria eléctrica corresponden a las plantas generadoras. En el segundo nivel, las funciones de 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

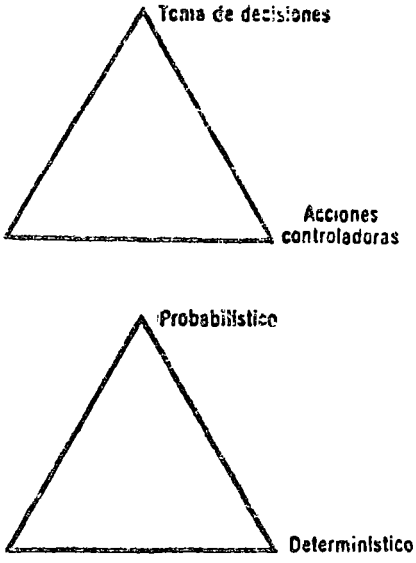
*Paralelamente a las jerarquías señaladas en el nivel de control, al ir hacia el vértice de la pirámide se puede identificar una jerarquía de funciones de control: regulación, optimización, adaptación y organización automática.

*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.

*Jerarquización { de Nivel / de Tiempo / de Modo

*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.

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:

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

Nivel 1 Plantas generadoras

Nivel 2 Sistemas individuales

Nivel 3 Sistema interconectado

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

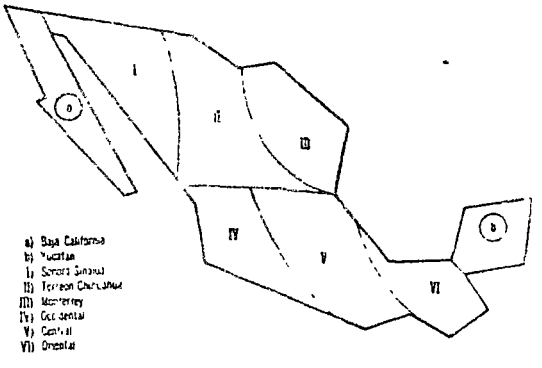


Fig. 2.2.1 Subdivisión, en sistemas regionales, de República Mexicana.

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

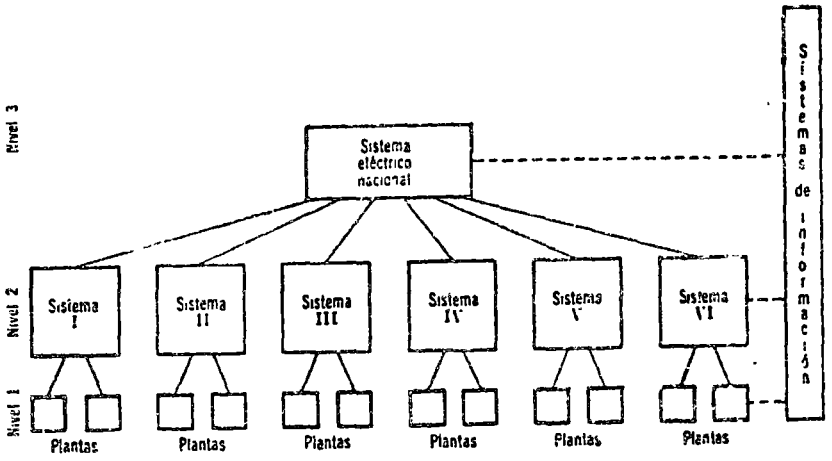


Fig. 2.2.2 Jerarquización del sistema eléctrico nacional por plantas, sistemas regionales y sistema interconectado.

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.

*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.

*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

*Subdivisión del nivel por tensiones de transmisión.

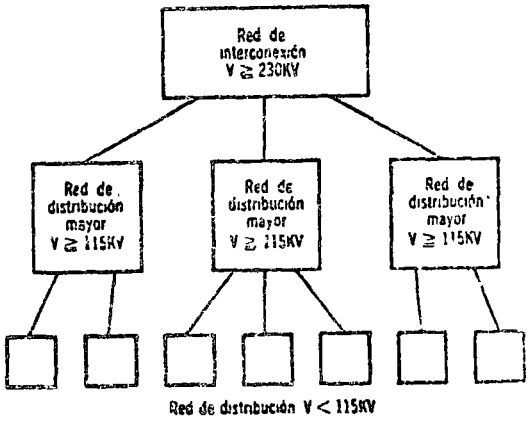


Fig. 2.2.3 Jerarquización del sistema eléctrico nacional por niveles de tensiones de transmisión

*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.

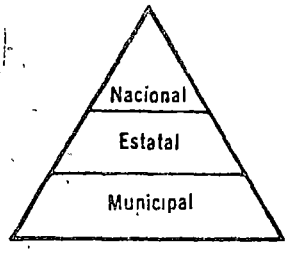


Fig. 2.2.4 Pirámide jerárquica administrativa de la educación

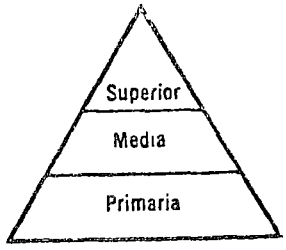


Fig. 2.2.5 Pirámide jerárquica de grado de la educación.

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 ($x \leq 24$ 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.

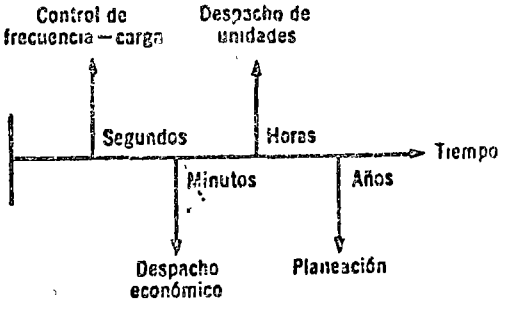


Fig. 2.2.6 Jerarquización por tiempo.

*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

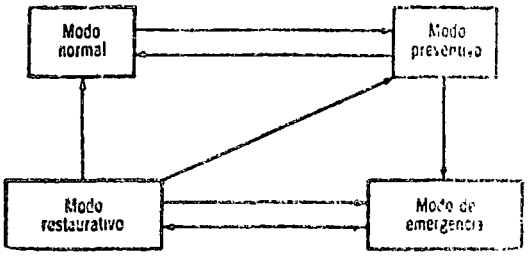


Fig. 2.2.7 Jerarquización de los modos de operación de un sistema eléctrico de potencia.

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.

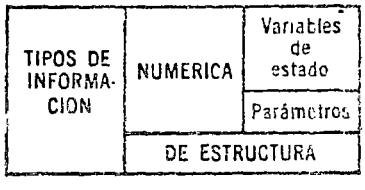


Fig. 2.4.1 Tipos de información.

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

*Formas de información { Inherente / Disponible

a) Inherente

b) Disponible de inmediato.

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:

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

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).

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.

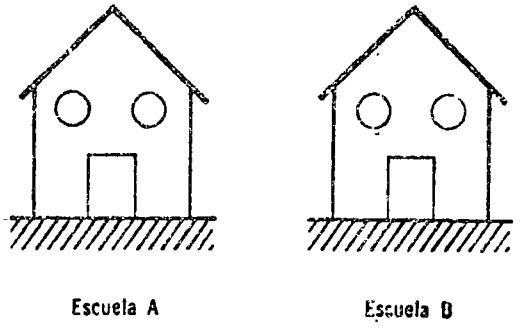


Fig. 2.4.3 Dos escuelas.

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

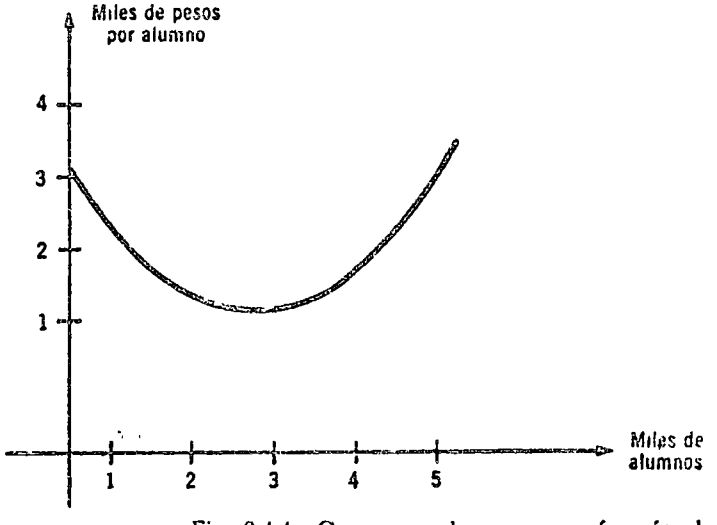


Fig. 2.4.4 Costo por alumno como función de la población.

*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:

$$2\,000 \times 1\,250 + 4\,000 \times 1\,500 = 8\,500\,000$$

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:

$$3\,000 \times 1\,000 + 3\,000 \times 1\,000 = 6\,000\,000$$

*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

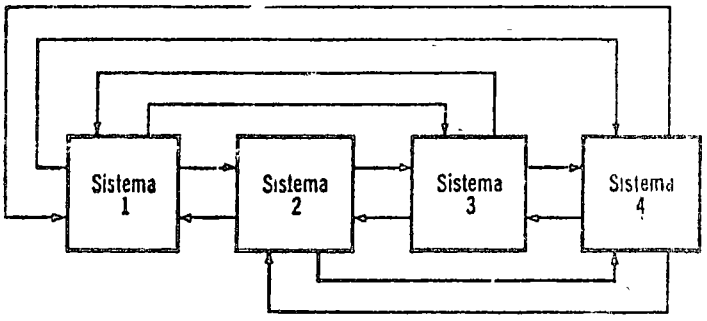


Fig. 2.4.5 Intercambio de información directa.

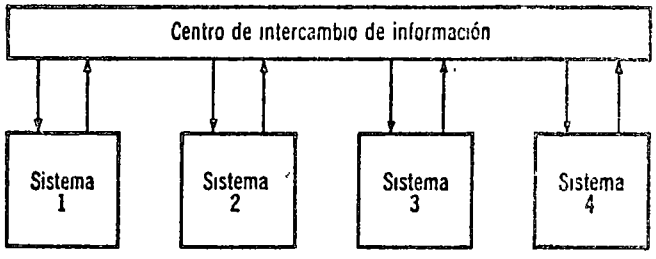


Fig. 4.2.6 Intercambio de información mediante centro de información.

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

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

*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 "*bi*nary digi*t*." 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

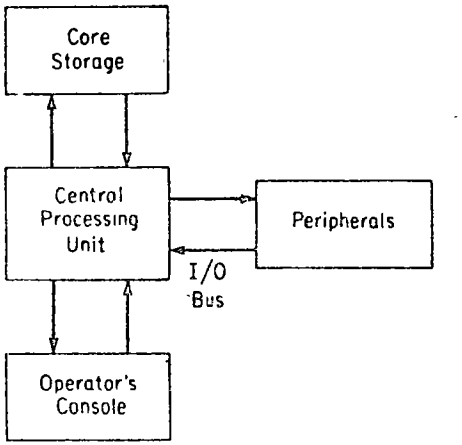


FIG. 2-1. Schematic representation of a computer.

primary functions are the following:

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 com-

monly 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. 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

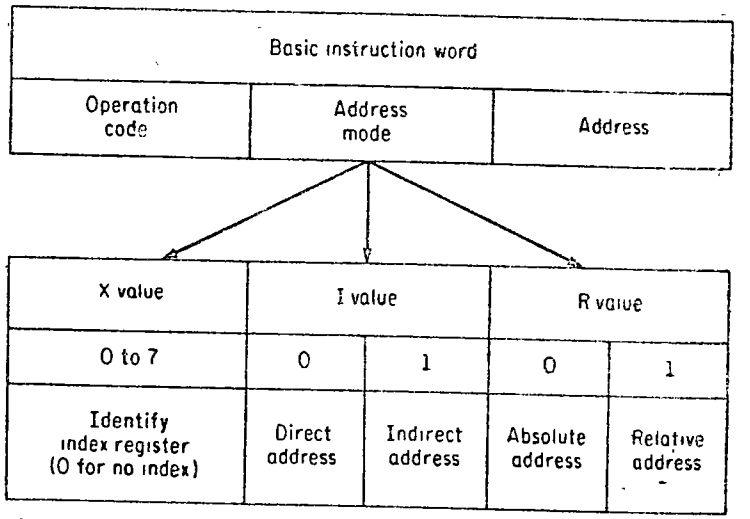


FIG. 2-2. Format of a single-address instruction. (Reproduced by permission from Ref. 1.)

code, the address mode, and the address itself. The purpose of these fields are as follows:

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 or to 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 fre-

quently 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

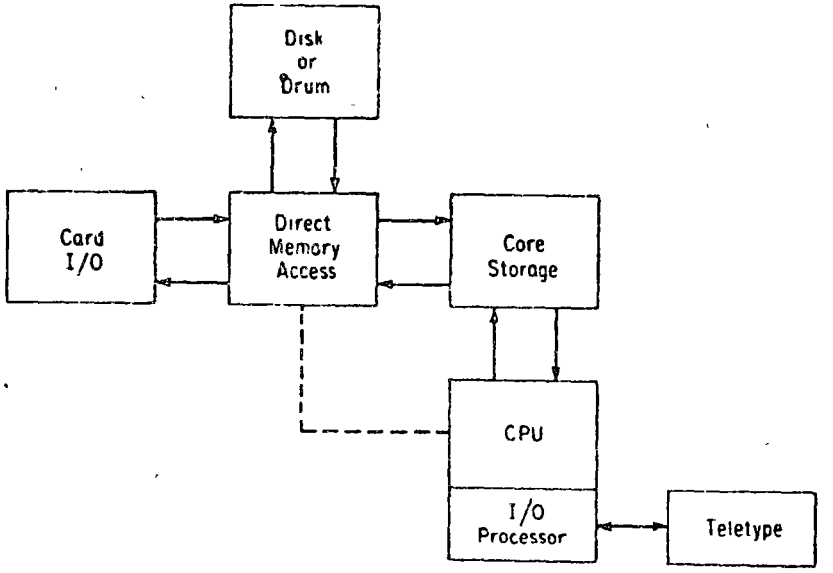


FIG. 2-3. Direct memory access channel.

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.

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 addi-

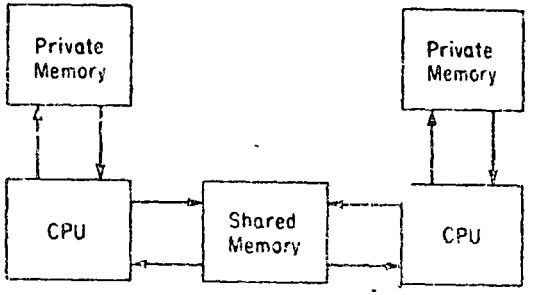


FIG. 2-4. Multiple processor configuration.

tion 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.

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

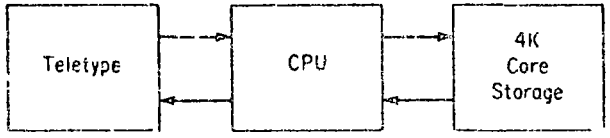


FIG. 2-5. Configuration of a minicomputer.

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.

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

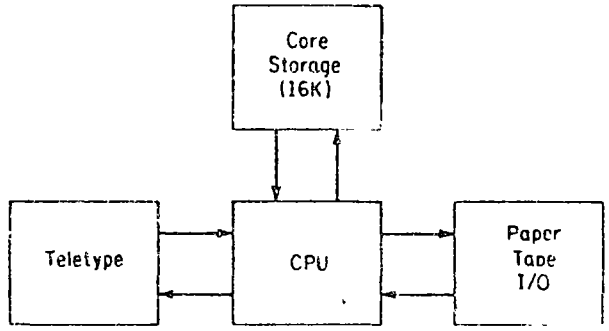


FIG. 2-6. Typical configuration of a direct digital control computer.

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.

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

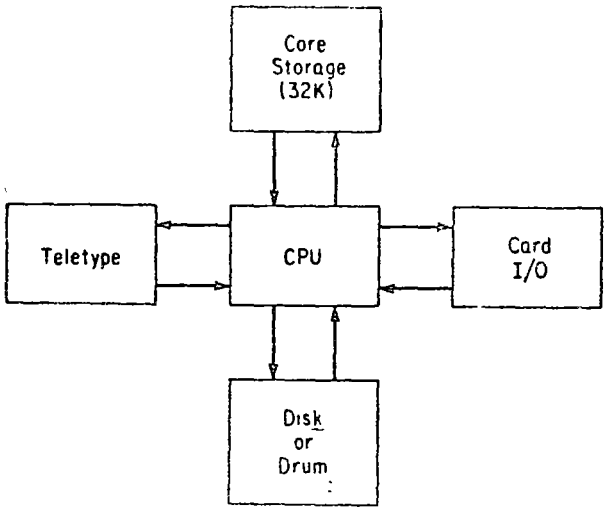


FIG. 2-7. Supervisory system

e such as Fortran. Program development and compiling could · on-line. The operating system would transfer programs from ; 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. .t, 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

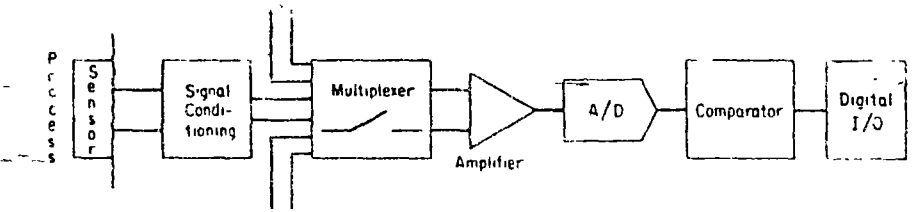


FIG. 2-8. Analog input system.

values from the process. These signals can be classified as follows:

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 susceptible 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

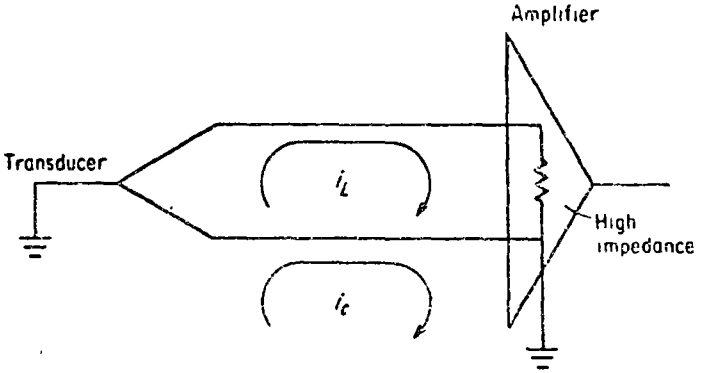


FIG. 2-9. Circuit susceptible to common-mode noise

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.

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 elements depending upon the sensor itself. When the output of the transducer 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 voltage signal, the signal conditioner generally transforms it to a voltage signal prior to the multiplexer. For example, if the output of the transducer is a current signal, the signal conditioner generally contains a resistor across which the voltage input to the multiplexer is taken.

Multiplexer. The multiplexer provides the mechanism by which

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

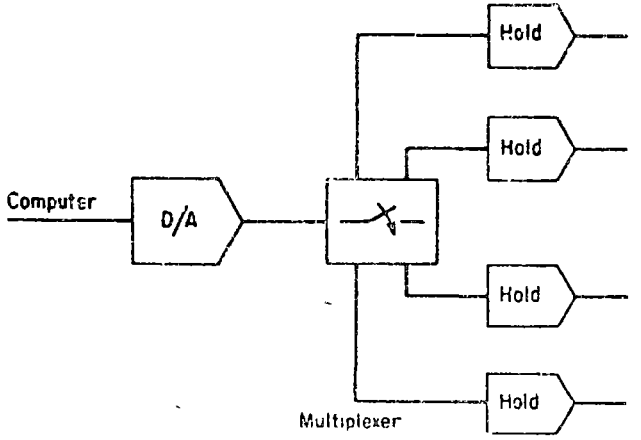


FIG. 2-10. Multiplexing the output of a D/A converter

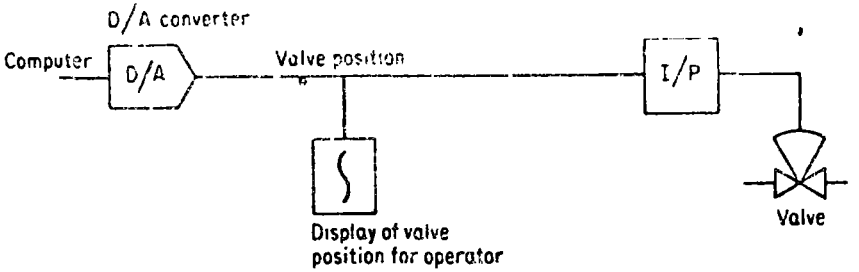
the hold circuits to maintain the value between samples, the economics tend to favor individual D/A converters.

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.

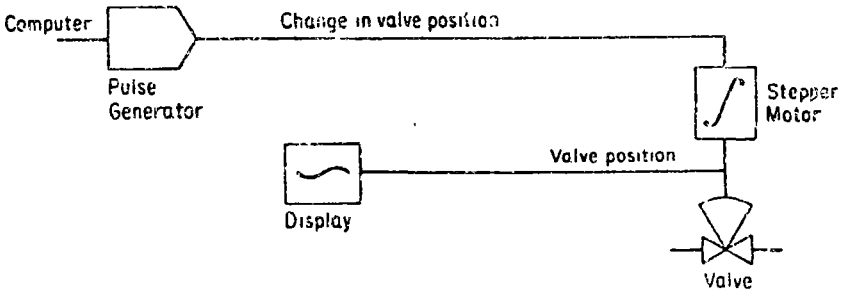
To illustrate the use of these devices, consider the output of a quantity such as a valve position or set point for an analog controller. Perhaps the most direct approach is to use a D/A converter as illustrated in Fig. 2-11a. Pertinent points are:

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.

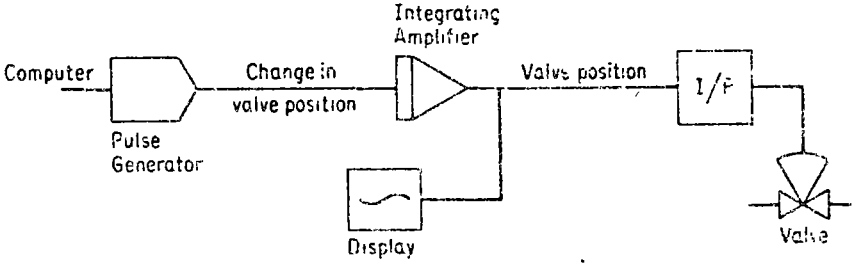
Alternatively, a pulse generator can be used in the configuration in



(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 n necessary that the computer be able to ascertain the va position (unless it must be verified that the desired change w 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 system 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 of adequate documentation requires considerable effort, but is a task that must be undertaken while preparing the programs themselves. That is, it is not feasible to delay preparation of documentation until the programming task is completed.

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

1. A written statement of the function this program is to perform, as well as the details of the approach used to perform 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 coverage.

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

2-14 FOREGROUND/BACKGROUND OPERATION

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

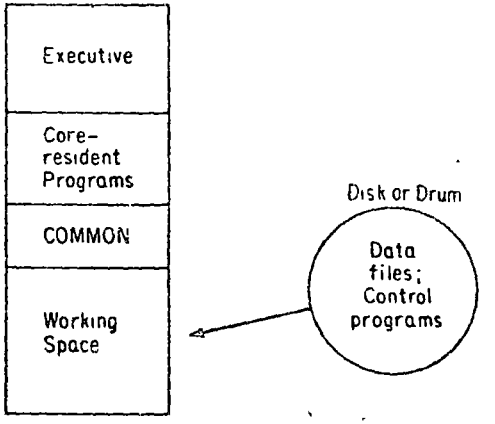


FIG. 2-12. Disk-oriented operating system.

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

When the execution of a control program residing on disk is 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*.

Multiprogramming systems generally check QUEUE both upon completing execution of a control program and upon completion of

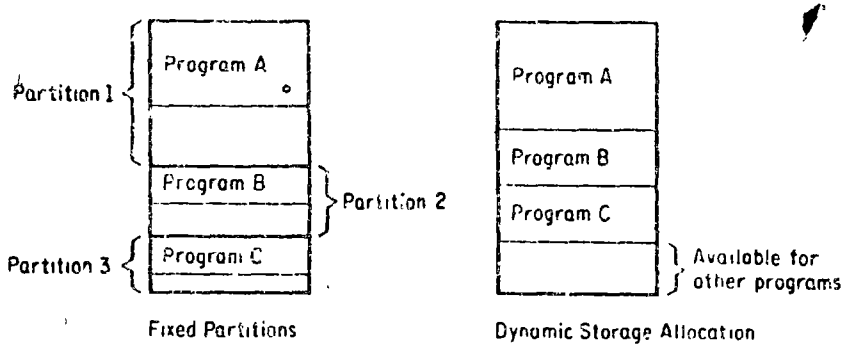


FIG. 2-13. Core allocation in multiprogramming systems

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

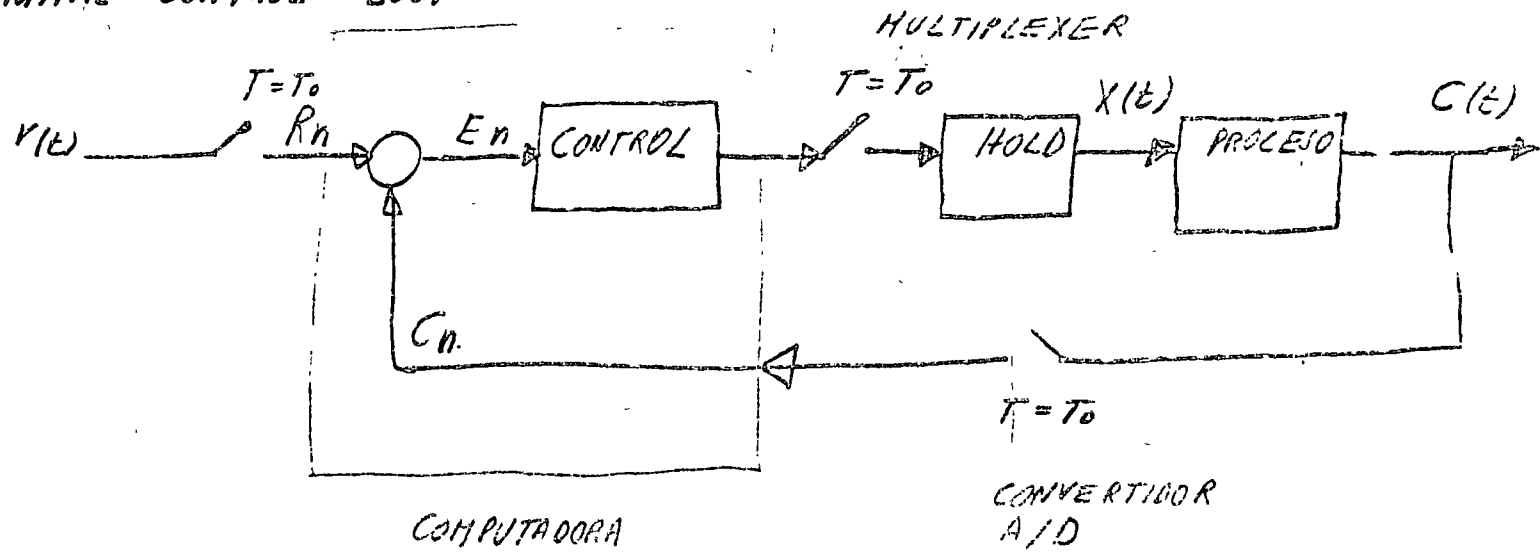
INGENIERIA DE CONTROL DE PROCESOS Y APLICACIONES

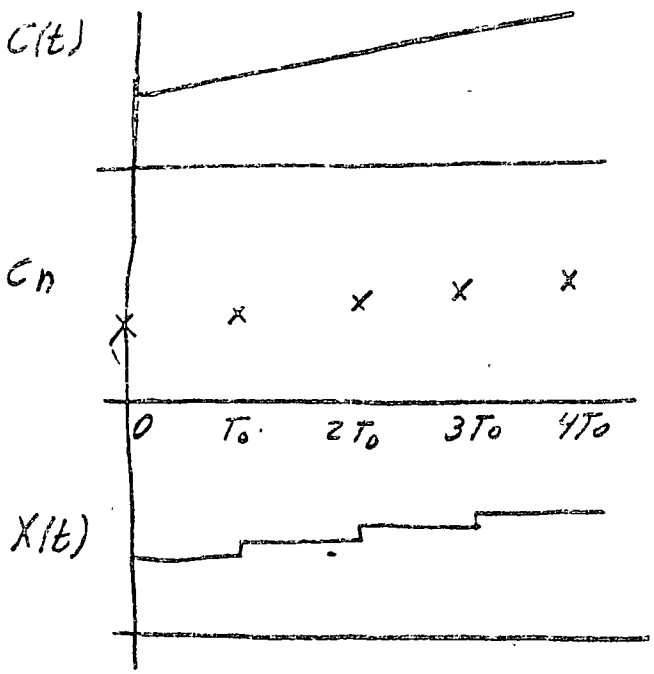
TEMA: CONSIDERACIONES EN EL DOMINIO DE LA FRECUENCIA

M. en C. ALBERTO MAURICIO

MIER MUTH

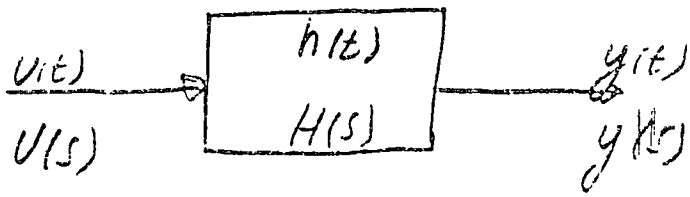
DIGITAL CONTROL LOOP



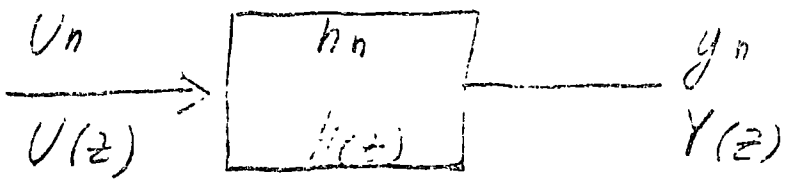


$$E_n = R_n - C_n$$

ANALISIS MATEMATICO DE DIGITAL CONTROL LOOPS



CONTINUOS



DISCRETOS

S. CONTINUOS

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

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

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

S. DISCRETOS TRANSFORMADA Z

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

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

$$= x_0 + x_1 z^{-1} + x_2 z^{-2} + \cdots$$

$$= x(0) + x(T) z^{-1} + x(2T) z^{-2} + \cdots$$

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

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

donde z = VARIABLE COMPLEJA

PROPIEDADES

i) TRANSFORMACION LINEAL

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

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

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

(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

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

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

$$= F(z/a)$$

(iv) MULT. POR UNA EXPONENCIAL

$$\mathcal{Z}\{e^{-at} U(t)\} = \sum_{n=0}^{\infty} e^{-anT} z^{-n} U(nT)$$

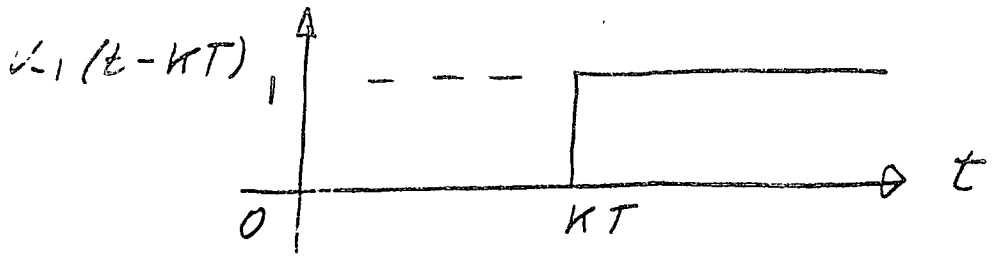
$$= \sum_{n=0}^{\infty} \left(e^{aT} z\right)^{-n} U(nT)$$

$$= F_1(e^{aT} z)$$

$$= \frac{1}{1 - (z e^{aT})^{-1}}$$

$$= \frac{1}{1 - e^{-aT} z^{-1}} \qquad |z^{-1}| < e^{aT}$$

V) RETRASOS EN EL TIEMPO



$$\mathcal{Z}\{f(t-kT)\,u_{-1}(t-kT)\} = \sum_{n=0}^{\infty} f(nT-kT)\,u_{-1}(nT-kT)\,z^{-n}$$

Si $m = n-k$

$$= \sum_{m=-k}^{\infty} f(mT)\,u(mT)\,z^{-(m+k)}$$

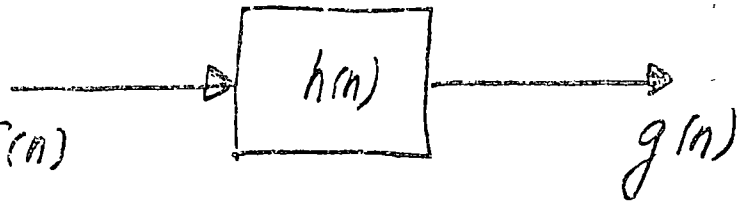
$$= z^{-k} \sum_{m=-k}^{\infty} f(mT)\,u(mT)\,z^{-m}$$

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

$$= z^{-k} F(z)$$

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

Ejemplo



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

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

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

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

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

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

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

b) $g_n = F_n * h_n$

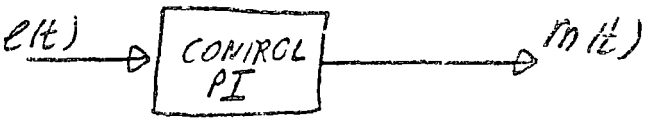
$$= \sum_{k=-\infty}^{\infty} F(n-k)\, h(k)$$

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

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

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

Z TRANSFORM OF DIFFERENCE EQUATIONS



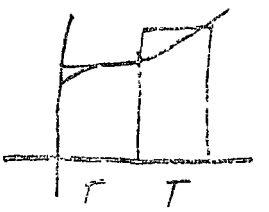
$$m(t) = K_c \left[e(t) + \frac{1}{T_i} \int e(t)\, dt \right] + M_0$$

K_C = GANANCIA

M_0 = VALOR INICIAL DE M_0

T_i = RESET TIME

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



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

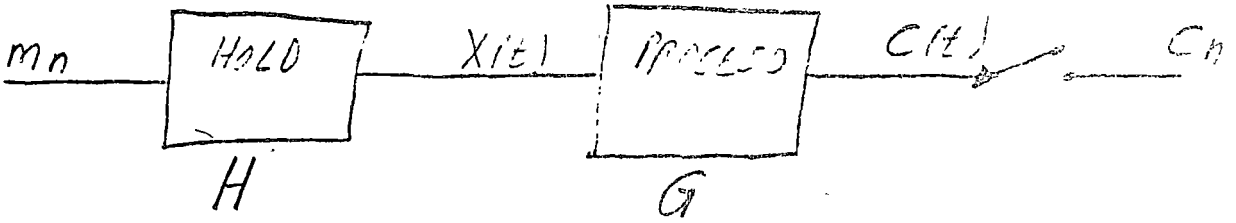
$$\Rightarrow \quad m_n - m_{n-1} = K_c \left[e_n - e_{n-1} \right] + \frac{K_c T}{T_i} e_n$$

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

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

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

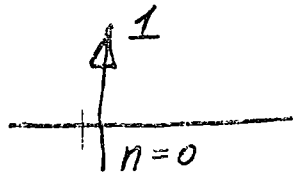
PULSE TRANSFER FUNCTIONS



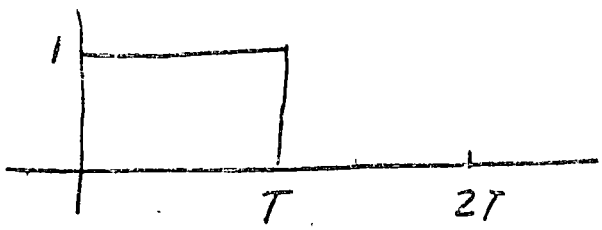
$$C(z) = [HG](z) \cdot M(z)$$

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

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



ZERO ORDER HOLD



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

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

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

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

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

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

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

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

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

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

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

$$= \frac{}{s}$$

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

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

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

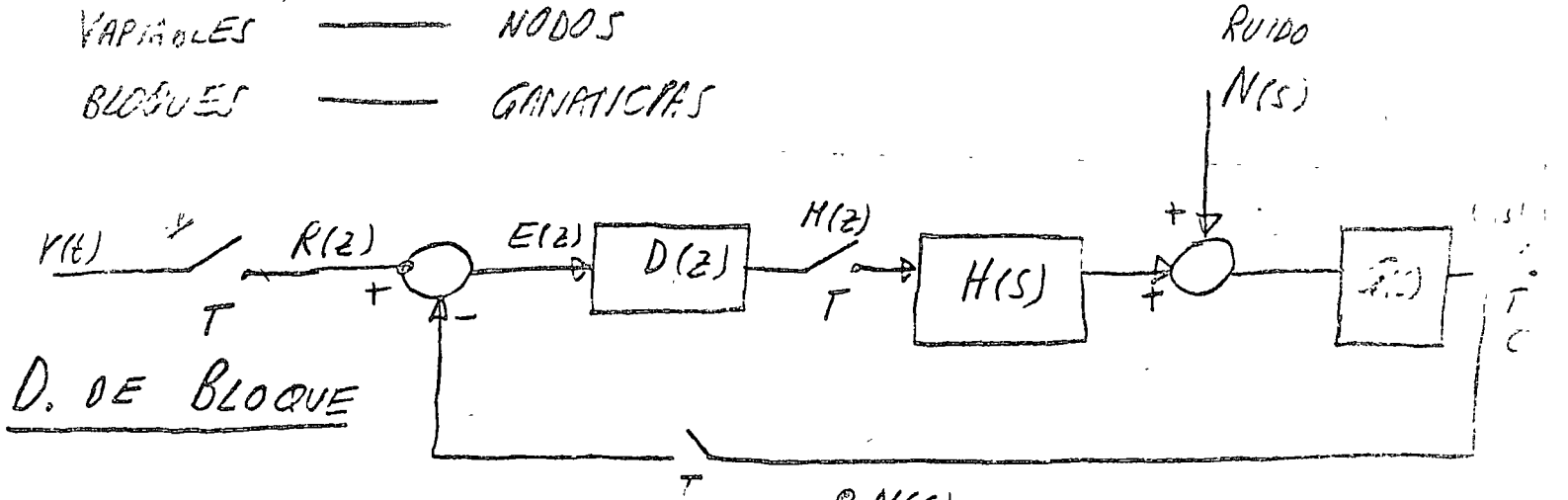
$$c_n - e^{-T}c_{n-1} = m_{n-1} - e^{-T}m_{n-1}$$

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

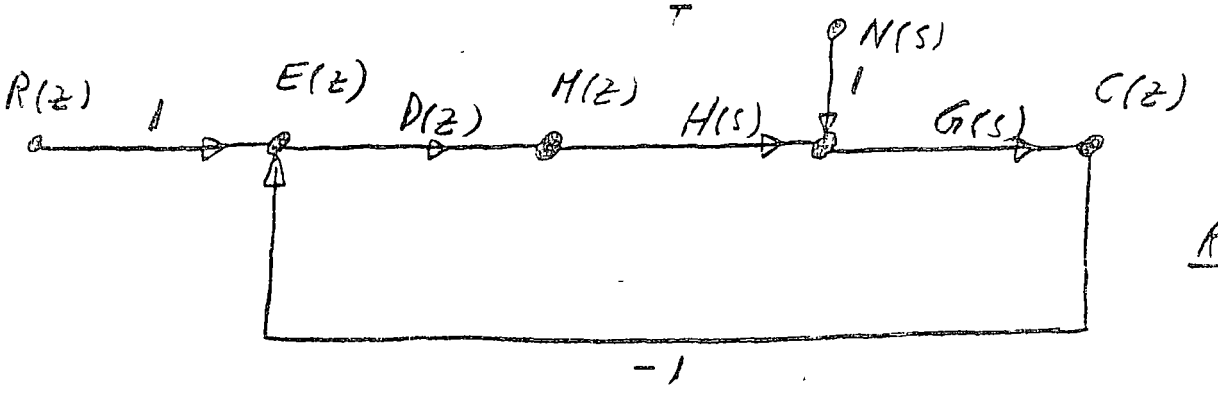
ANALISIS DE DIAGRAMAS DE BLOQUE.

VARIABLES ——— NODOS

BLOQUES ——— GANANCIAS



D. DE BLOQUE



REDGRAMA

PARA SISTEMAS LINEALES

⇒ SUPERPOSICION ES APLICABLE

$$C(z) = F_1(R(z), N(s))$$

NODO — 1 VARIABLE DEL SISTEMA

RAMA (ARISTA) — REPRESENTA RELACION CAUSA - EFECTO

NODO FUENTE

NODO POZO

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

REOGRAMA — NOTACION GRAFICA PARA DESCRIBIR CONJUNTOS DE RELACIONES LINEALES

TRAYECTORIA — SUBGRAFICA DE UN REOGRAMA 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 TRAYECTORA

GANANCIA DE LA MALLA — Π DE LAS TRANSMISIONES DE LA MALLA

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

Δ_K COFACTOR DE T_K = Δ DEL REOGRAMA QUE QUEDA AL ELIMINAR T_K

TRAYECTORIA

T_K GANANCIA DE LA TRAYECTORIA

MALLA SIMPLE

GMK 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

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

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

$N(s) = 0$

$r(t) = U_{-1}(t)$ ESCALÓN

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

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

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

$$= 0 \qquad nT < 0$$

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

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

$$= Tz\left(z^{-2} + 2z^{-3} + 3z^{-4} + \cdots\right.$$

$$= -zT\frac{d}{dz}\left(z^{-1} + z^{-2} + z^{-3} + \cdots\right.$$

$$= -Tz\frac{d}{dz}\left(z^{-1}(1 + z^{-1} + z^{-2} + \cdots\right.$$

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

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

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

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

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

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

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

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

TRANSFORMADAS INVERSAS.

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

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

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

EXAMPLE

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

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

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

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

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

$$= -6$$

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

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

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

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

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

$$-\alpha = \ln 0.5$$

$$\alpha = 0.694$$

ESTABILIDAD

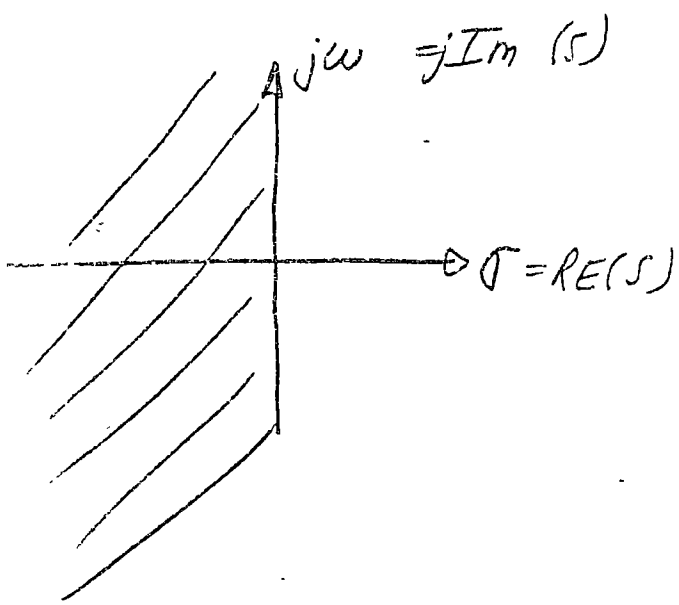
$$z = e^{TS}$$

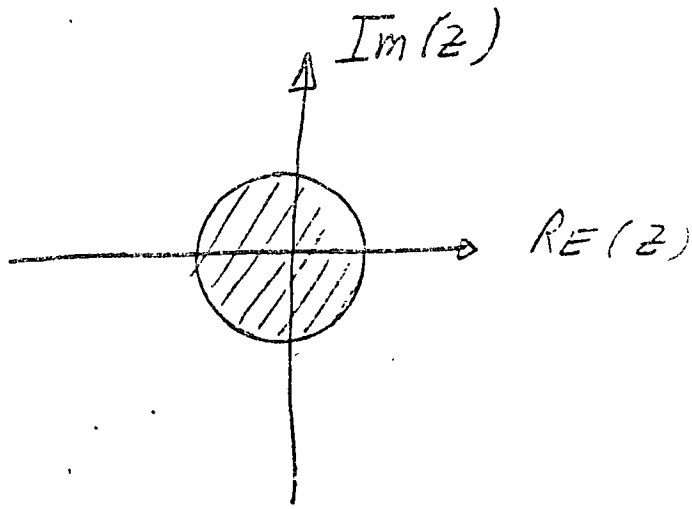
$$\ln z = TS$$

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

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

$\Longrightarrow$ $|Z| < 1$



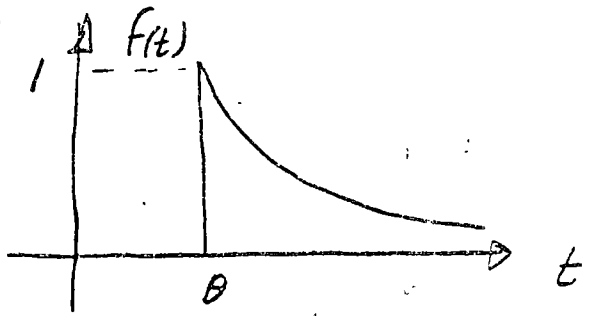


SYSTEMS WITH DEAD TIME

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

donde $m = 1 - \theta/T$

$mT = T - \theta$



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

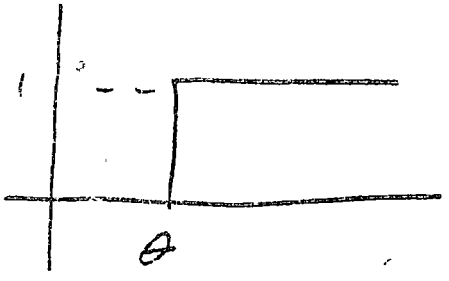
$$F(z) = \sum_{n=0}^{\infty} e^{-a(nT-\theta)} u_{-1}(nT-\theta)\, z^{-n}$$

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

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

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

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



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

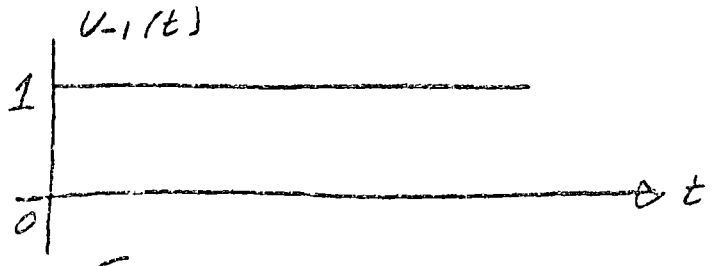
$$= \sum_{n=0}^{\infty} u_{-1}(nT-\theta)\, z^{-n}$$

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

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

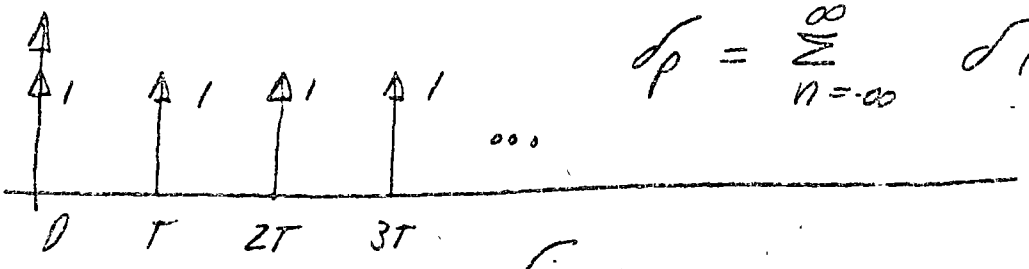
VER EJEMPLOS LIBRO 103-107

FREQUENCY SPECTRUM

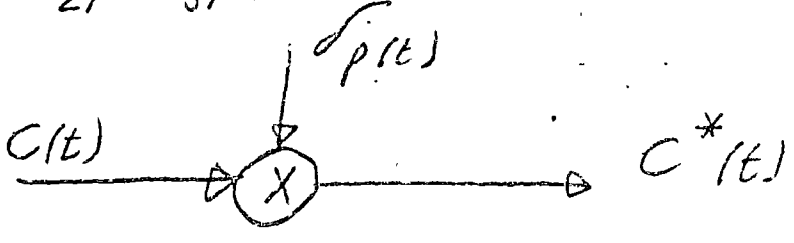


$$\delta(t) = \frac{d}{dt}\, u_{-1}(t)$$

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

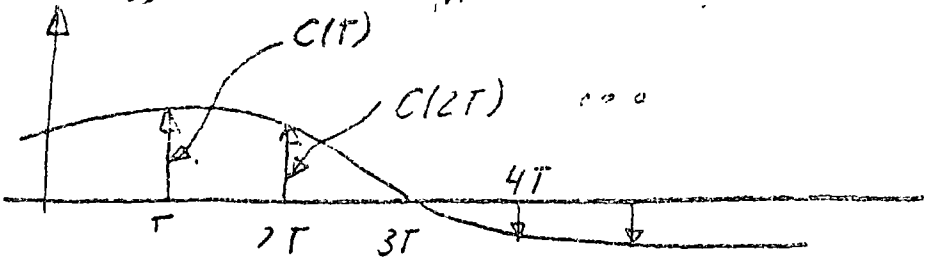


$$\delta_p = \sum_{n=-\infty}^{\infty} \delta(t - nT)$$



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

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



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

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

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

Si $z = e^{Ts}$

$$= \mathcal{Z}\,(c(t))$$

$$c^*(t) = \delta_{p(t)} \cdot c(t)$$

$$\delta_{p(t)} = \sum_{n=-\infty}^{\infty} C_n\, e^{\pm jn 2\pi F_0 t} \qquad \text{SERIE DE FOURIER}$$

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

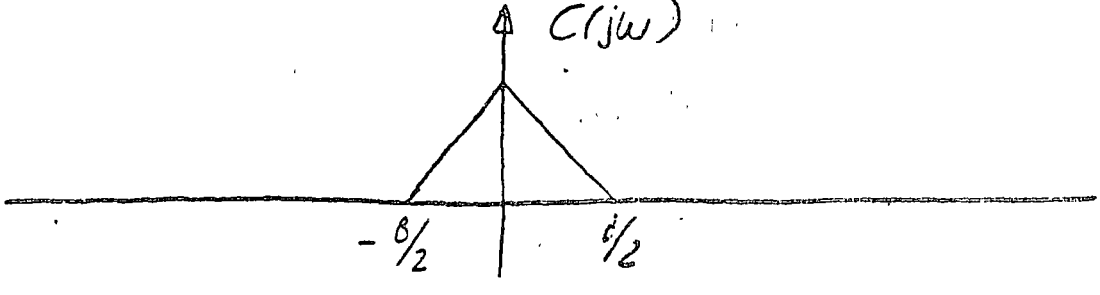
$$C_n = \frac{1}{T}$$

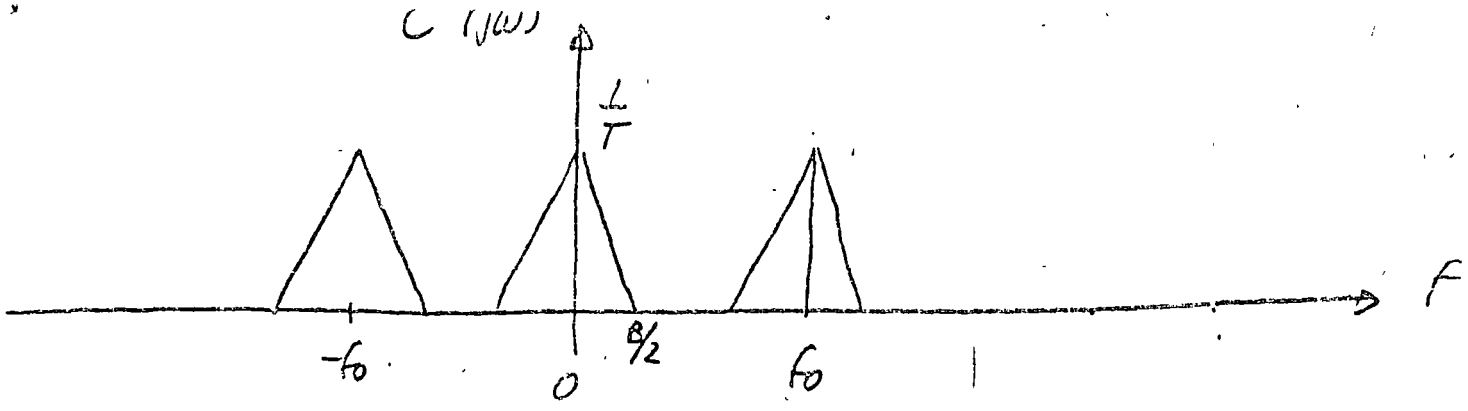
$$\Rightarrow \delta_{p(t)} = \sum_{n=-\infty}^{\infty} \frac{1}{T}\, e^{\pm j2\pi n F_0 t}$$

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

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

$$= \frac{1}{T} \sum_{n=-\infty}^{\infty} C(jw - j2\pi n f_0)$$





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

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

DATA HOLDS

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

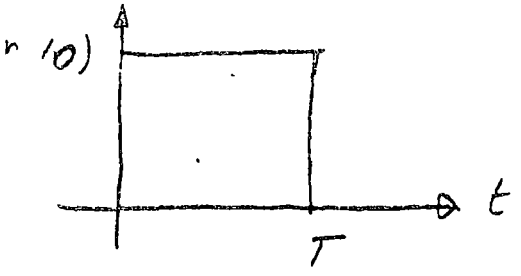
1) FACILIDAD PARA RECONSTRUIR LA SEÑAL
2) EFECTO QUE PRODUCE SOBRE LA MALLA DE CONTROL
3) ASPECTOS PRACTICOS SOBRE SU CONSTRUCCION

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

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

ZERO ORDER HOLD

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



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

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

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

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

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

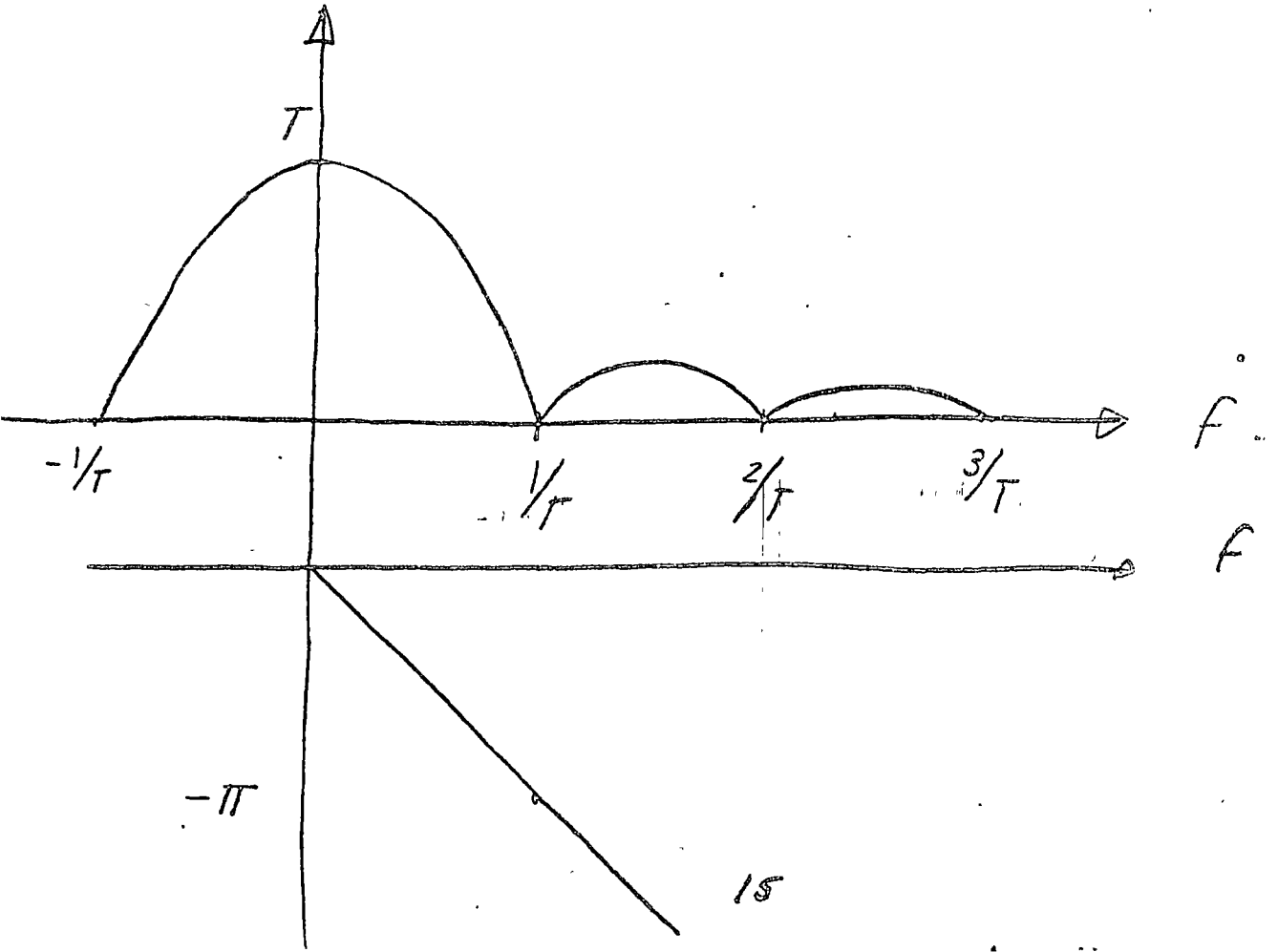
$$M(j\omega) = Te^{-j\omega T/2}\,\frac{\mathrm{sen}\,\omega T/2}{\omega T/2}$$

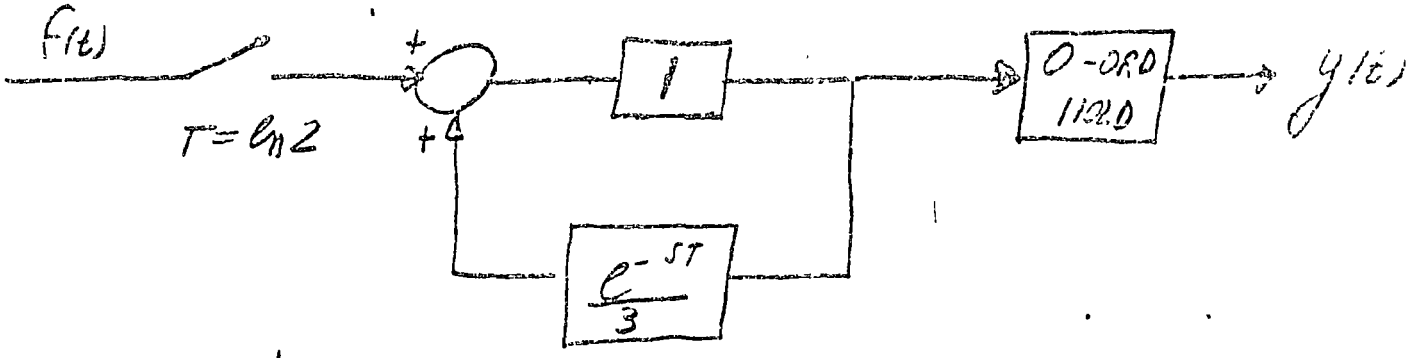
$$M(f) = T\frac{\mathrm{sen}\,\frac{2\pi f T}{2}}{\frac{2\pi f T}{2}}e^{-j\omega T/2} = T\,\mathrm{sinc}\,fT\,e^{-j\pi Tf}$$

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

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

$$\phi = -\pi Tf$$





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

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

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

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

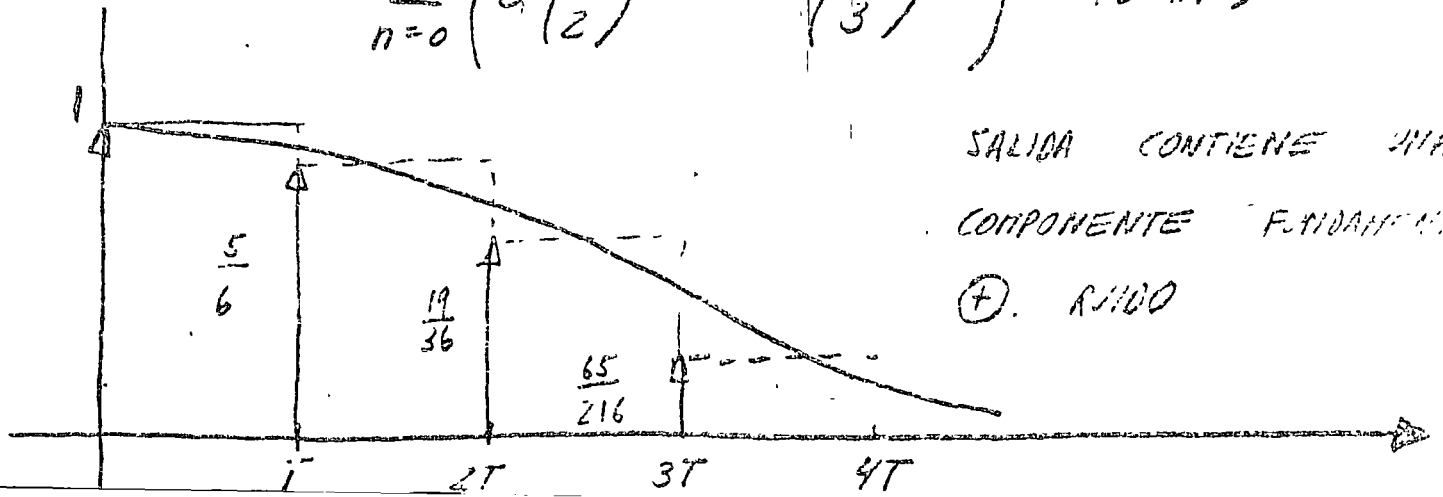
$$= \frac{1}{1-\frac{1}{2}z^{-1}}$$

INPUT TO ZERO ORDER HOLD

$$\left(\frac{1}{1-\frac{1}{2}z^{-1}}\right) \delta \left(\frac{1}{1-e^{-ST}/3}\right) = \left(\frac{1}{1-\frac{1}{2}z^{-1}}\right)\left(\frac{1}{1-\frac{z^{-1}}{3}}\right)$$

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

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 :

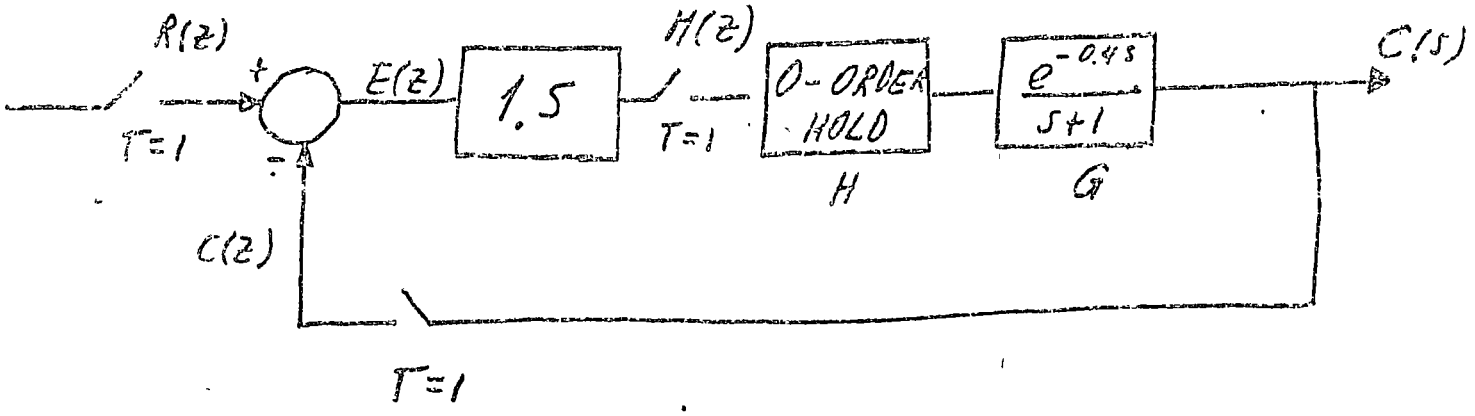
1) DIGITAL SIGNAL PROCESSING
ALAN V. OPPENHEIM / RONALD W. SCHAFER
PRENTICE HALL

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

INGENIERIA DE CONTROL DE PROCESOS Y APLICACIONES

TEMA: TECNICAS DE IDENTIFICACION

M. EN C. RAFAEL LOPEZ

OCTUBRE DE 1977.

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.

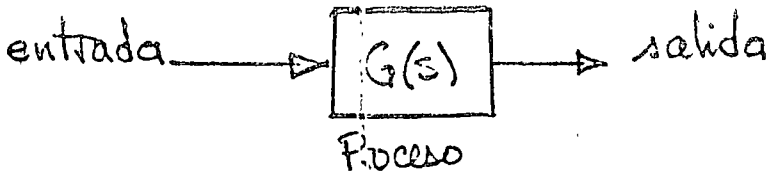
INTRODUCCION

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:

MODELO
- Estructura
- Valor de los parámetros

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

Parámetros :
- K ganancia
- θ tiempo de retraso
- τ constante de tiempo

Supongamos que en alguna forma se ha determinado que :

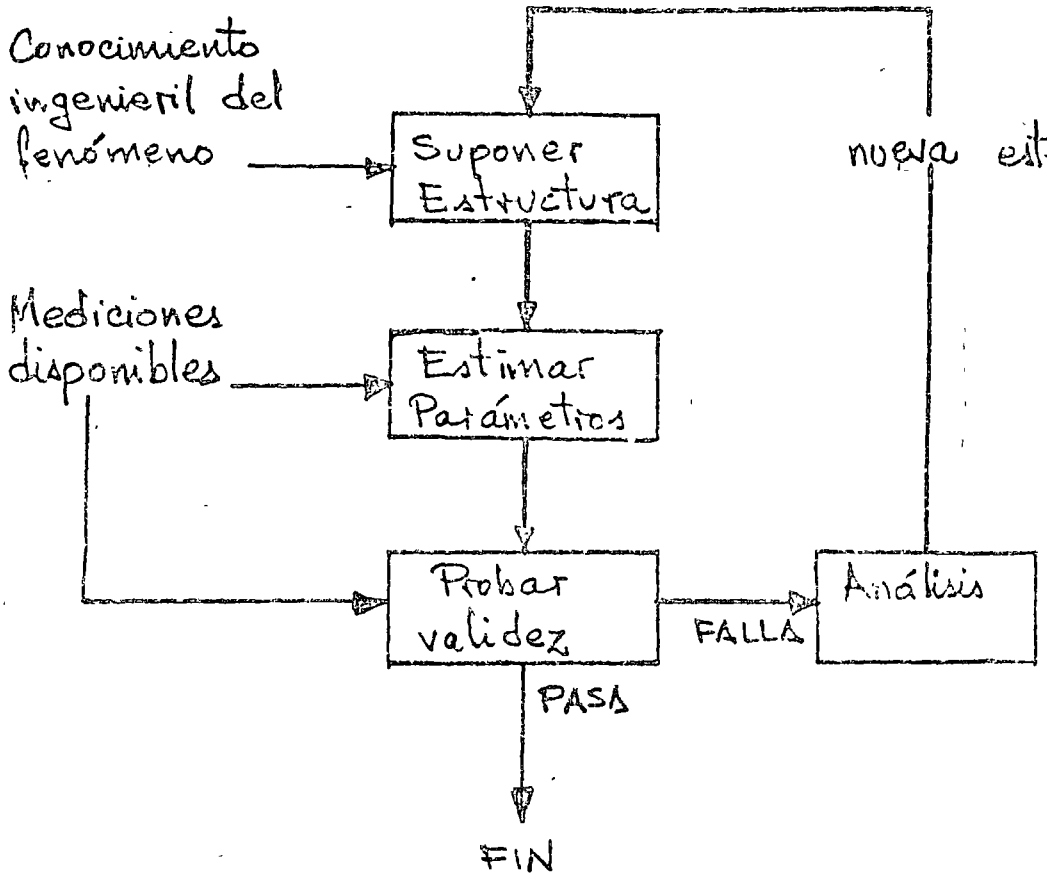
$K = 3$
$\theta = 1.5 \text{ seg}$
$\tau = 2 \text{ seg}^{-10}$

El MODELO del proceso es entonces

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

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 ... 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.

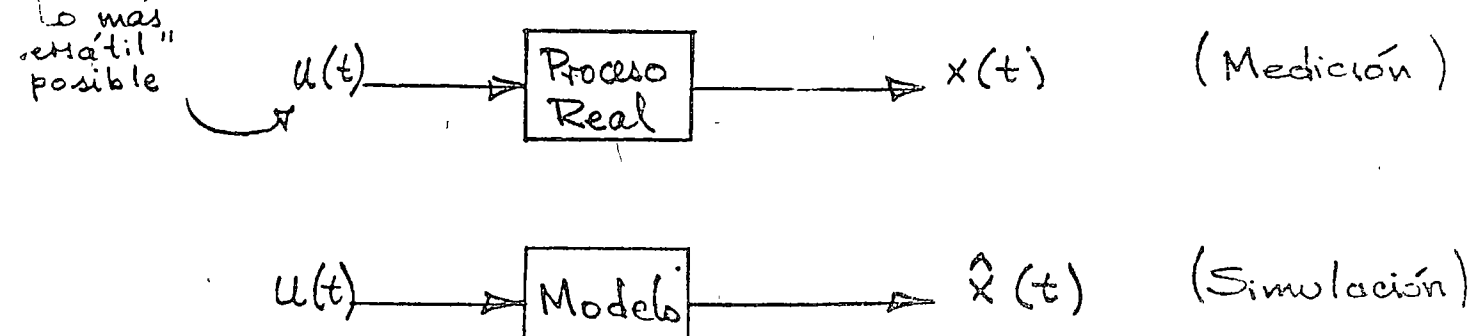
Se hará énfasis en la utilidad de los modelos discretos, que proporcionan métodos rápidos y eficientes en línea.

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



Problema

Encontrar valores de los parámetros en forma tal que una función del error

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

sea mínima.

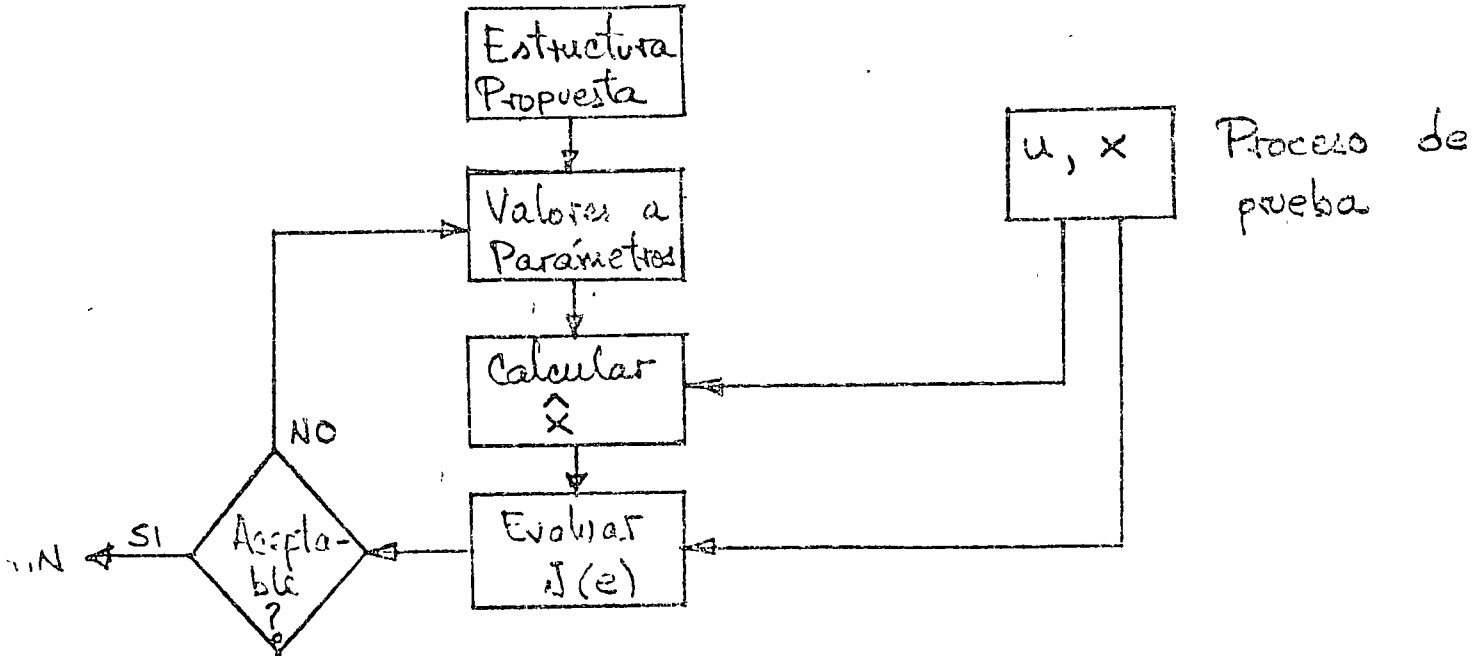
En tiempo discreto se tendrían secuencias x_i, $\hat{x}_i$ y el error

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

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 todos 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:

- Modelo continuo → Discretizar → Modelo Discreto

Modelo → Transformada z → Modelo Discreto

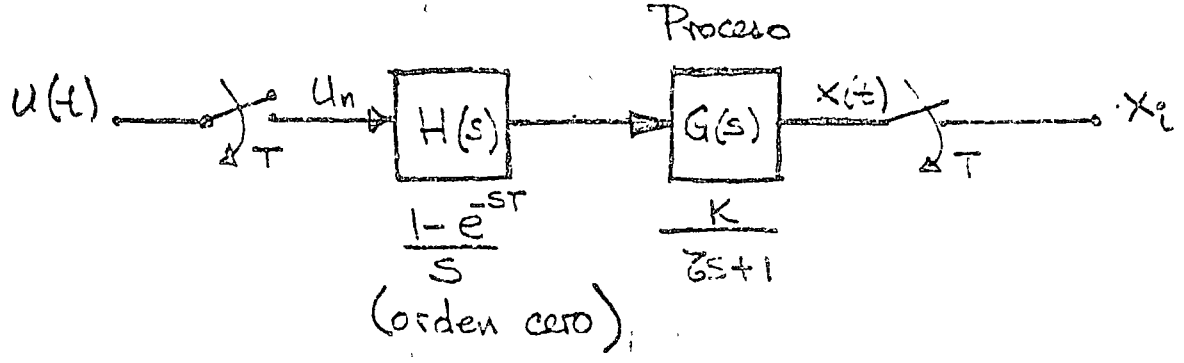
Ventajas del modelo discreto

- Más fácil de resolver (Ω_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 z óptimos

Alternativas:

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

$$\hat{x}_{i+1} = a \hat{x}_i + b u_i + D$$ (independiente del proceso real x)

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

porque produce una regresión lineal.

Mínimos cuadrados:

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

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

↑ Note que es intuitivamente correcto.

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

Regresión lineal

$$a \sum x_i^2 + b \sum x_i u_i + D \sum x_i = \sum x_{i+1} x_i$$

$$a \sum x_i u_i + b \sum u_i^2 + D \sum u_i = \sum x_{i+1} u_i$$

$$a \sum x_i + b \sum u_i + ND = \sum x_{i+1}$$

N = numero de puntos

↖ Tres ecuaciones con

— Otra forma de obtener el mismo modelo:

DIFERENCIAS FINITAS:

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

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

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

(Válido si T es pequeño)

— Simplificación (no estimar D)

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

— Otra alternativa:

$$\bar{x}_{i+1} = a \bar{x}_i + b \bar{u}_i + D$$

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

$$\bar{x}_i = \frac{1}{i} \sum_{j=0}^{i} x_j \quad ; \quad \bar{u}_i = \frac{1}{i} \sum_{j=0}^{i} u_j$$

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

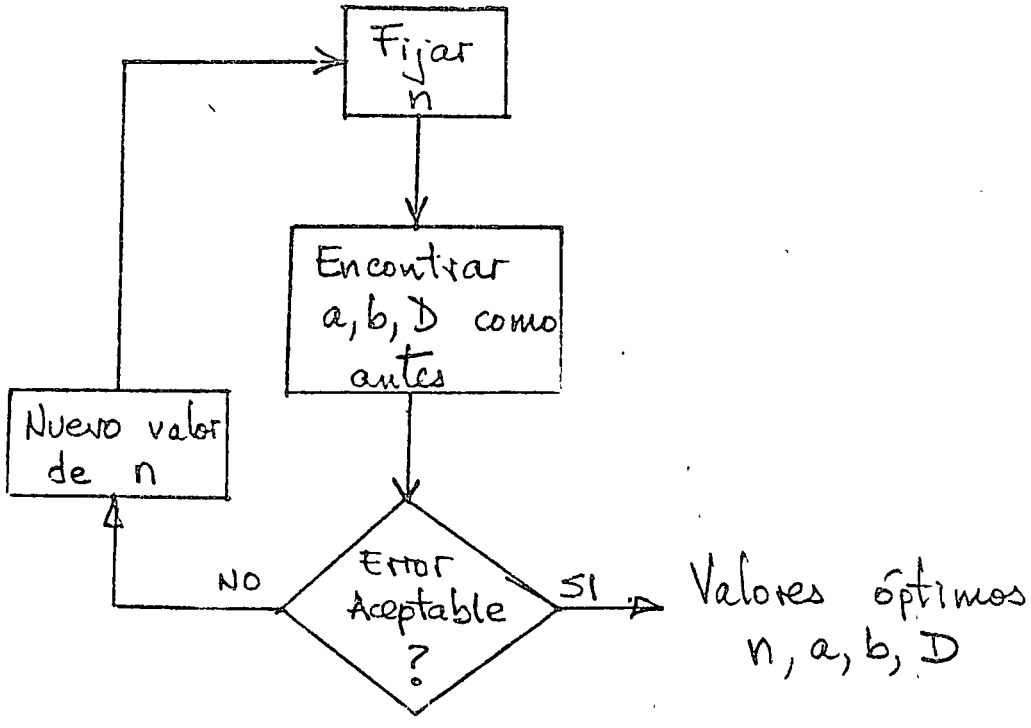
SISTEMAS CON RETRASO $e^{-\theta s}$

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

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

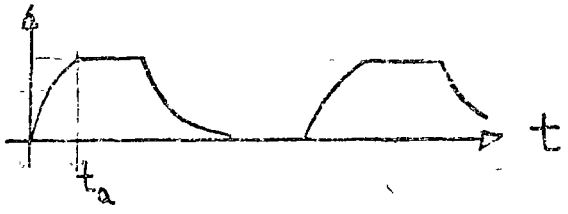
$$X_{i+1} = a X_i + b U_{i-n} + D$$

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
$x_{i+1} = ax_i + bu_i + D$	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

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

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

$$\Rightarrow x_i = a x_{i-1} - a_2 x_{i-2} + b_1 u_{i-(n+1)} + b_2 u_{i-(n+2)}$$

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

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

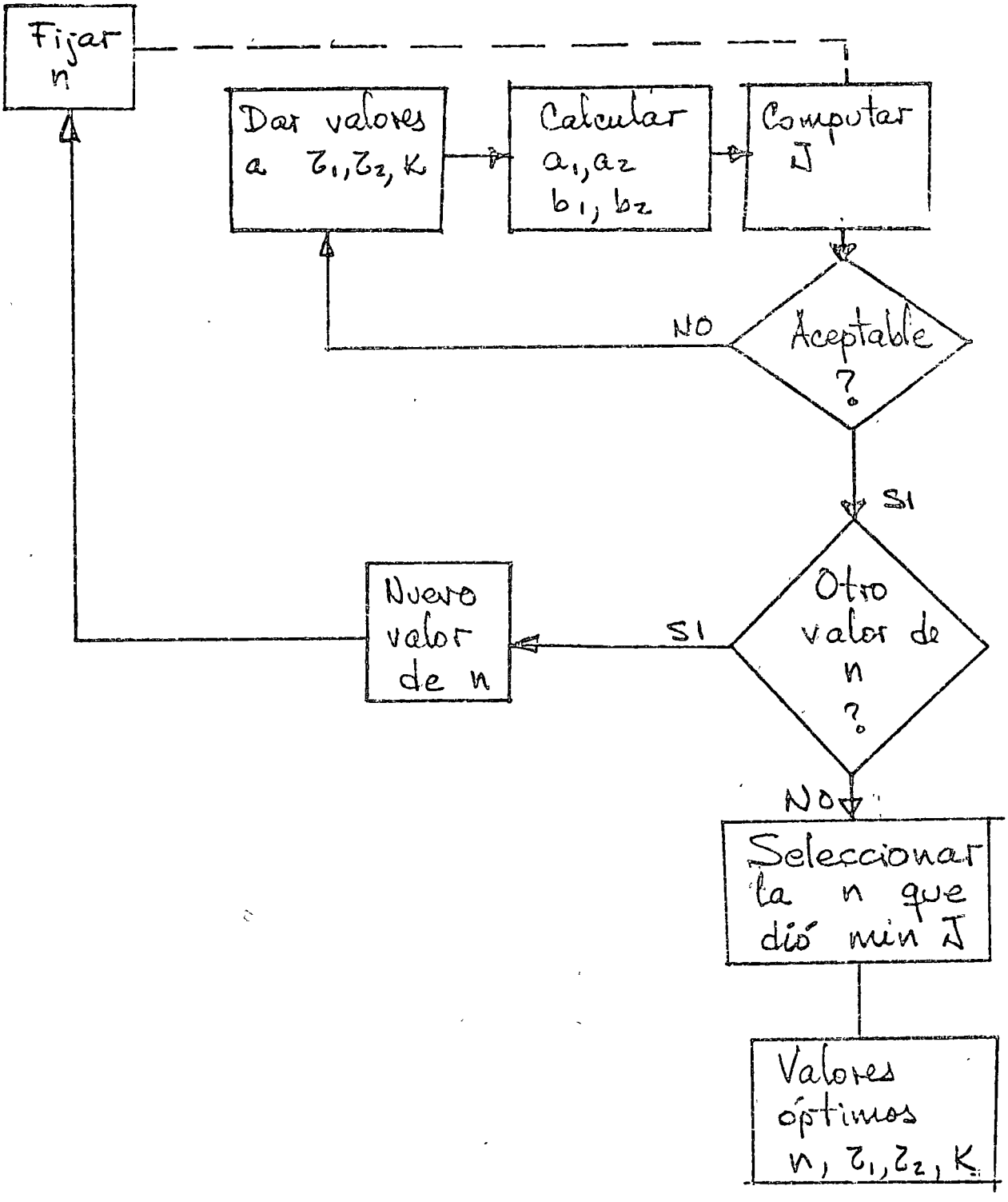
$$= \frac{1}{N} \sum_{i=1}^{N} (x_i - a x_{i-1} + a_2 x_{i-2} - b_1 u_{i-(n+1)} + b_2 u_{i-(n+2)})^2$$

Minimizar J con respecto a $n; \tau_1, \tau_2, K$ ← 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 un conjunto de valores de n sobre los

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



EVALUACION DE $\mathcal{J}(e)$

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 $\mathcal{J}(e)$

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

En $\mathcal{J}$ aparecen tres tipos de factores:

- Términos en x_i (salida) tales como:

  Σx_i^2, $\Sigma x_i x_{i-1}$, $\Sigma x_i x_{i-2}$ 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 $\sum u^2_{i-n-1}$, $\sum u_{i-n-2} u_{i-n-1}$, etc.

Aunque estrictamente estos términos dependen de n (que es uno de los parámetros que se está buscando), se pueden también considerar como constantes y evaluar a priori, si se toma en cuenta que, para valores grandes de N:

$$\sum u^2_{i-n-1} \simeq \sum u^2_{i-n-2} \simeq \sum u^2_{i-n-3} \cdots$$

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

- Términos en xu tales como

$$\sum u_{i-n-1} x_i \,, \quad \sum u_{i-n-2} x_i \,, \text{ etc.}$$

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:

$$S_{n+1} = \sum U_{i-n-1} X_{i-2}$$

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

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

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

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

y, si por ejemplo hubiese 10 posibles valores para n, sería necesario almacenar

en memoria un arreglo de

$10 \times 5 = 50$ elementos.

VENTANA EXPONENCIAL

Como se ha visto es necesario calcular varios factores del tipo

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

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

Comparemos (I) con la recursión

$$\bar{X}_i^2 = \alpha\, X_i^2 + (1-\alpha)\, \bar{X}_i^2$$

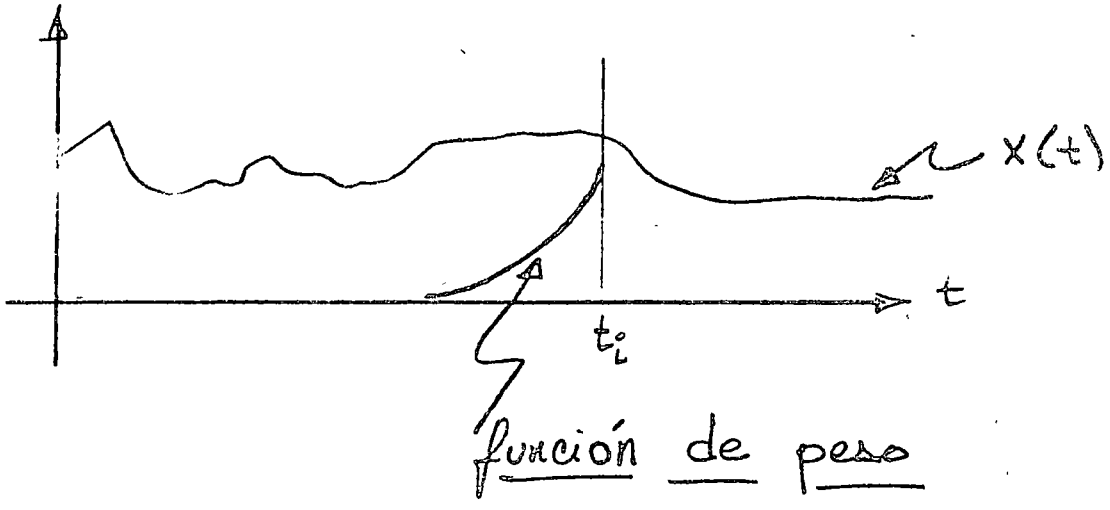
$$\bar{X}_1^2 = X_1^2 \qquad (II)$$

(II) se conoce como ventana exponencial (o media exponencial) y presenta varias ventajas sobre (I):

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

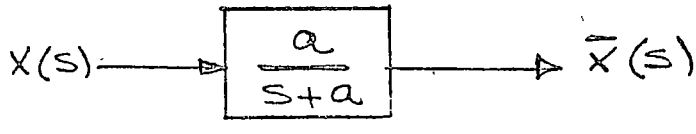
continuidad en los datos, permitiendo su actualización cada vez que se obtiene un nuevo dato

Gráficamente, el efecto de II sobre la información es :



función de peso

II es equivalente a un filtro



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

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

TECNICAS DE IDENTIFICACION

M. EN C. RAFAEL LOPEZ

15 OCTUBRE, 1977

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.

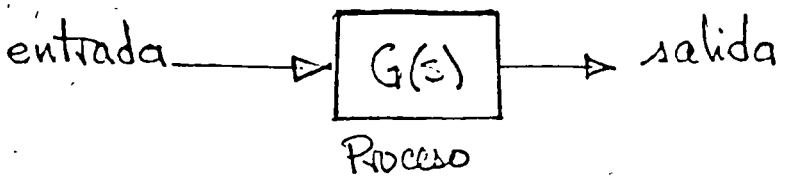
INTRODUCCION

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:

MODELO
- Estructura
- Valor de los parámetros

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

Parámetros :
- K ganancia
- θ tiempo de retraso
- τ constante de tiempo

Supongamos que en alguna forma se ha determinado que :

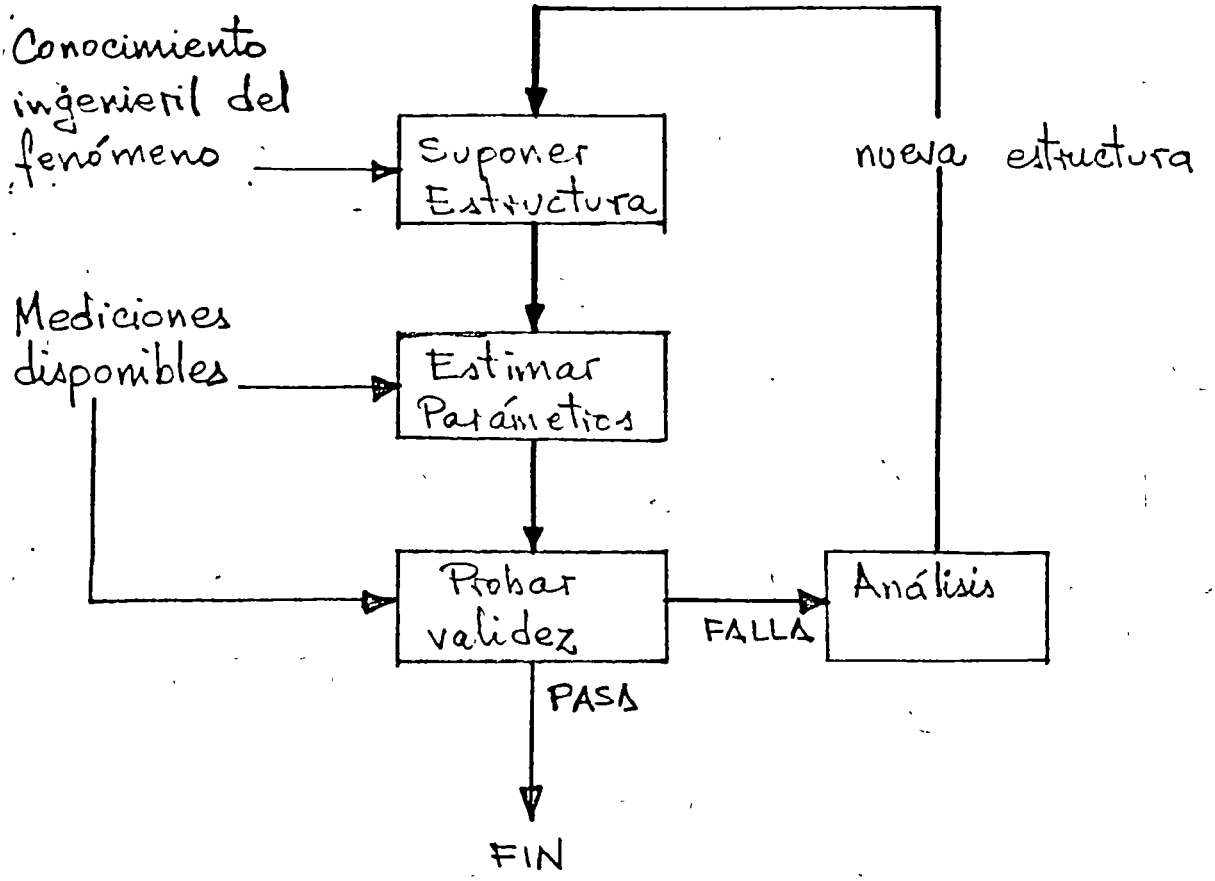
$K = 3$
$\theta = 1.5 \text{ seg}$
$\tau = 2 \text{ seg}^{-10}$

El MODELO del proceso es entonces

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

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.

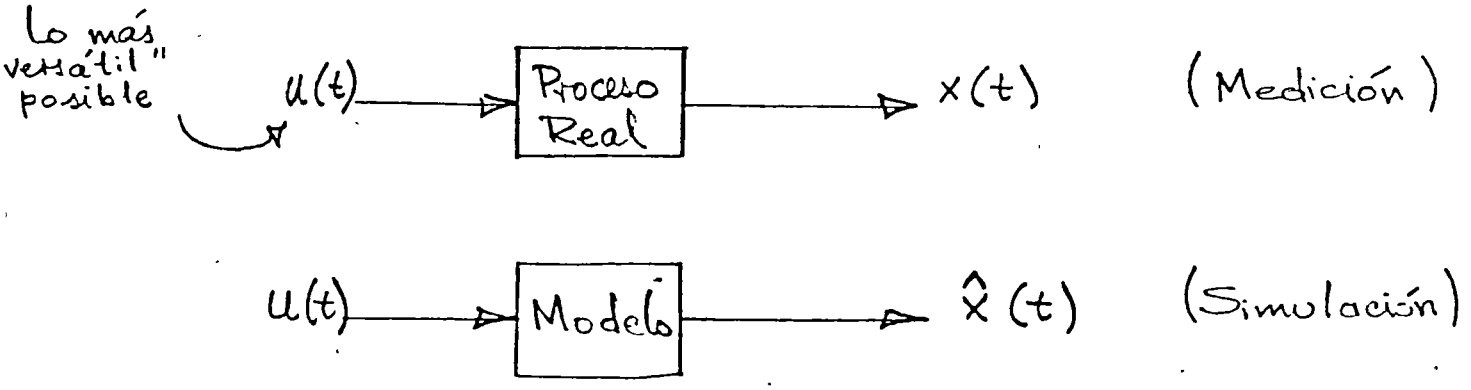
Se hará énfasis en la utilidad de los modelos discretos, que proporcionan métodos rápidos y eficientes en línea.

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



Problema

Encontrar valores de los parámetros en forma tal que una función del error

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

sea mínima.

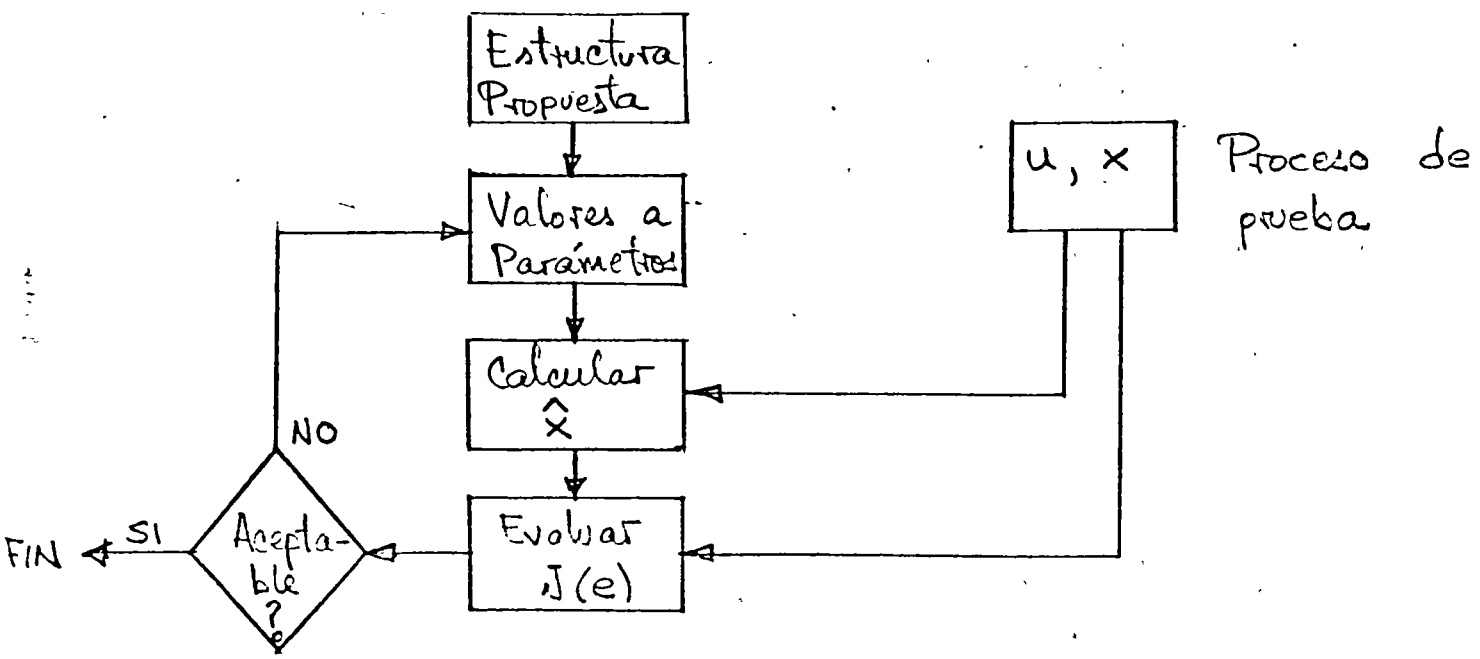
En tiempo discreto se tendrían secuencias x_i, $\hat{x}_i$ y el error

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

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 todos 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:

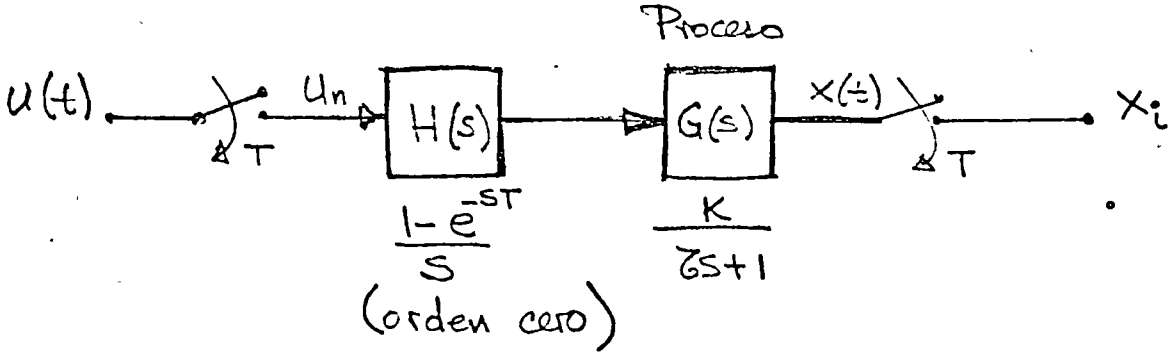
- Modelo continuo → (Discretizar) → Modelo Discreto ← Transformada z
- Modelo discreto

- Ventajas del modelo discreto
  - Más fácil de resolver $(\hat{x}_n)$
  - En algunos casos permite usar regresión linea

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 identificació en línea.

EJEMPLO



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

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

$x_{i+1} = a x_i + b u_i + D$ ← sesgo (bias)

Problema de identificación (estimación):

Encontrar K, Θ y ζ óptimos.

Alternativas:

$$\hat{X}_{i+1} = aX_i + bU_i + D$$

$$\hat{X}_{i+1} = a\hat{X}_i + bU_i + D$$ (independiente del proceso real x)

Usaremos $\hat{X}_{i+1} = aX_i + bU_i + D$

porque produce una regresión lineal.

Mínimos cuadrados:

$$\min_{a,b,D} \sum e_i^2 = \min \sum (X_{i+1} - \hat{X}_{i+1})^2$$

$$= \min_{a,b,D} \sum (X_{i+1} - aX_i - bU_i - D)^2$$

↑ Note que es intuitivamente correcto.

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

Regresión lineal

$$\begin{cases} a\sum X_i^2 + b\sum X_i U_i + D\sum X_i = \sum X_{i+1} X_i \\ a\sum X_i U_i + b\sum U_i^2 + D\sum U_i = \sum X_{i+1} U_i \\ a\sum X_i + b\sum U_i + ND = \sum X_{i+1} \end{cases}$$

N = número de puntos

↖ Tres ecuaciones con tres incógnitas

= Otra forma de obtener el mismo modelo:

DIFERENCIAS FINITAS:

$$\tau \dot{x} + x = Ku + D' \Rightarrow \tau \frac{x_{i+1} - x_i}{T} + x_i = K u_i + D'$$

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

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

(Válido si T es pequeño)

= Simplificación (no estimar D):

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

= Otra alternativa:

$$\bar{x}_{i+1} = a \bar{x}_i + b \bar{U}_i + D$$

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

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

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

SISTEMAS CON RETRASO

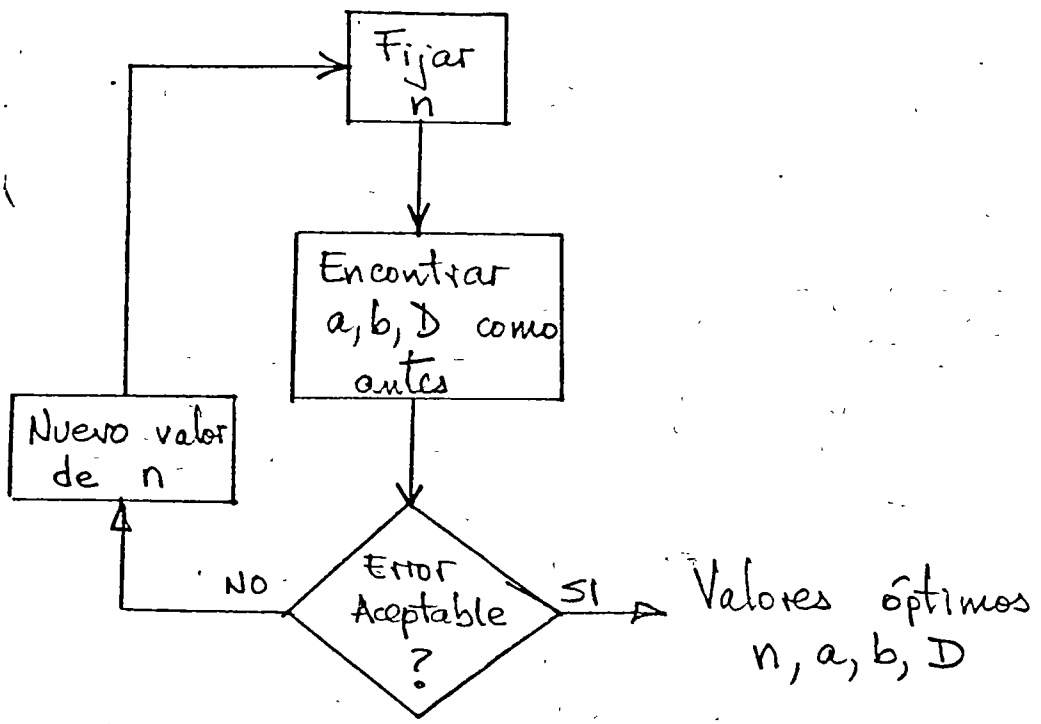
$e^{-\theta s}$

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

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

$$X_{i+1} = a X_i + b U_{i-n} + D$$

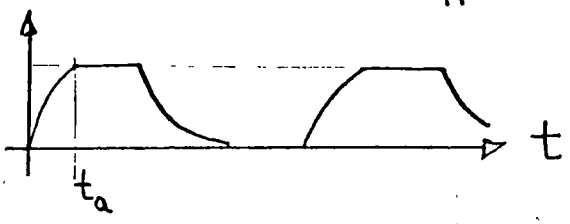
Ya no es posible usar regresión lineal. Por ello se propone el siguiente método:



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 d filtrado previo d datos
$x_{i+1} = ax_i + bu_i + D$	muy pequeños. Especialmente si t_f y τ pequeños	pequeños indepenc. 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

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

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

$$\Rightarrow x_i = a x_{i-1} - a_2 x_{i-2} + b_1 u_{i-(n+1)} + b_2 u_{i-(n+2)}$$

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

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

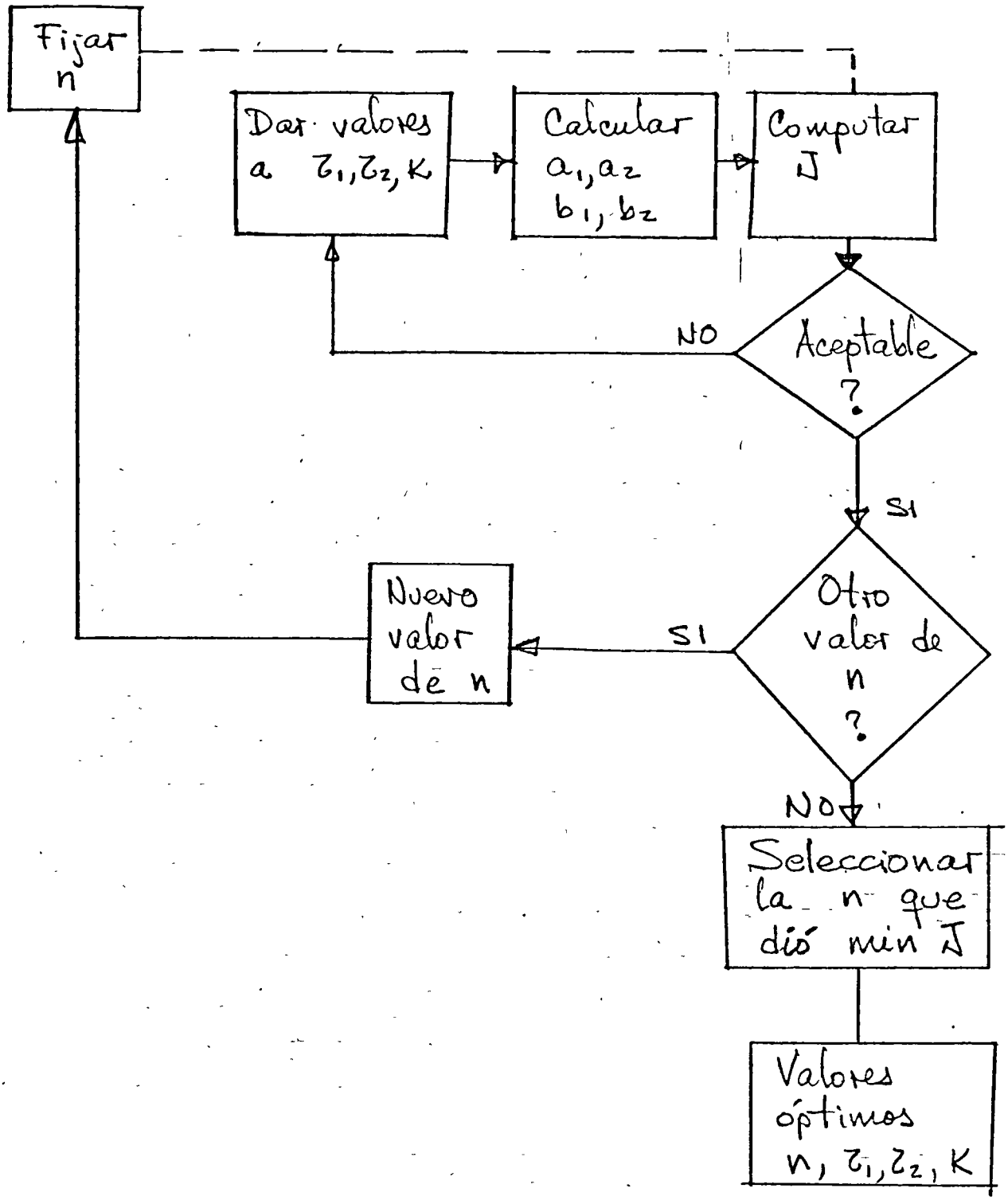
$$= \frac{1}{N}\sum_{i=1}^{N} (x_i - a x_{i-1} + a_2 x_{i-2} - b_1 u_{i-(n+1)} + b_2 u_{i-(n+2)})^2$$

Minimizar J con respecto a $n; \tau_1, \tau_2, K$ ← 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 un conjunto de valores de n sobre los cuales se efectuará la búsqueda.

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



EVALUACION DE $\mathcal{J}(e)$

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 $\mathcal{J}(e)$. En esta sección se discute la evaluación de $\mathcal{J}$ enfocada a cálculos en línea. Se busca, por tanto, obtener un mínimo de operaciones necesarias, y esto se logra identificando factores constantes en $\mathcal{J}$, es decir, que no dependen de los parámetros sobre los que se está optimizando. En $\mathcal{J}$ aparecen tres tipos de factores:

- Términos en x (salida) tales como:

  Σx_i^2, $\Sigma x_i x_{i-1}$, $\Sigma x_i x_{i-2}$ 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 constantes y evaluar a priori, si se toma en cuenta que, para valores grandes de N:

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

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

- Términos en xu tales como

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

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

$10 \times 5 = 50$ elementos.

VENTANA EXPONENCIAL

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

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

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

Comparemos (I) con la recursión

$$\bar{X}_i^2 = \alpha\, X_i^2 + (1-\alpha)\, \bar{X}_i^2$$

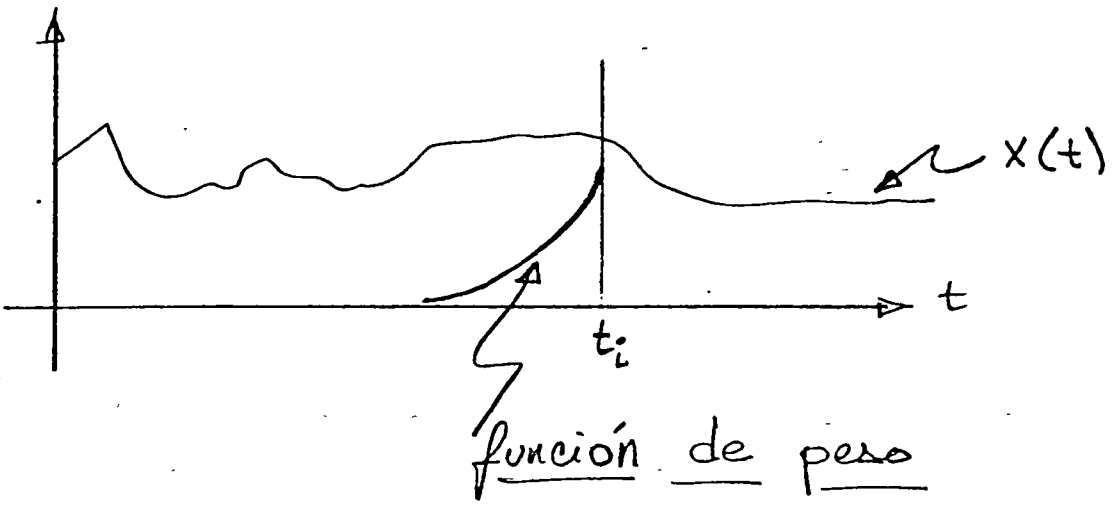
$$\bar{X}_1^2 = X_1^2 \qquad (II)$$

(II) se conoce como ventana exponencial (o media exponencial) y presenta varias ventajas sobre (I):

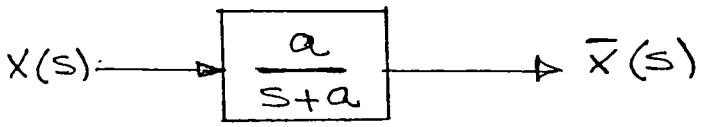
- No hay problemas de overflow
- Se da mas peso a los datos mas recientes
- Se computa recursivamente y ofrece

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

Gráficamente, el efecto de Π sobre la información es:



Π es equivalente a un filtro



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

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

CONTROL OPTIMO

DR. VICTOR GEREZ GREISER

OCTUBRE DE 1977.

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

CONTROL SUPERVISORIO

El más usado en la actualidad Controles analógicos realizan las funciones de control al primer nivel. La computadora calcule 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.

$$M = M(x_1, x_2, \ldots, x_n) \tag{6.1.1}$$

*Sujeta a las siguientes restricciones.

*Restricciones.

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

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

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

*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.

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

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.

*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 programación lineal es la más empleada.

*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.

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

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

*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 ($x_1, x_2, \ldots x_n$). 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:

*Espacio de N dimensiones.

*Coordenadas de un punto = variables $(x_1, x_2, \ldots, x_n)$

$$x_1 + x_2 \leq 4 \qquad (6.1.4)$$
$$2x_1 + x_2 \leq 6 \qquad (6.1.5)$$
$$x_1 \geq 0 \qquad (6.1.6)$$
$$x_2 \geq 0 \qquad (6.1.7)$$

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.

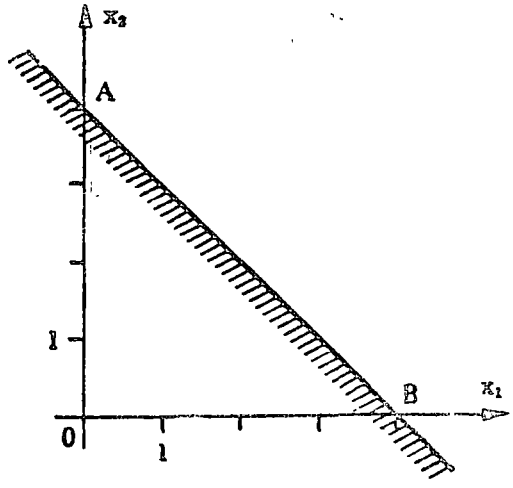


Fig. 6.1.1. Zona donde se cumple la restricción $x_1 + x_2 \leq 4$.

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.

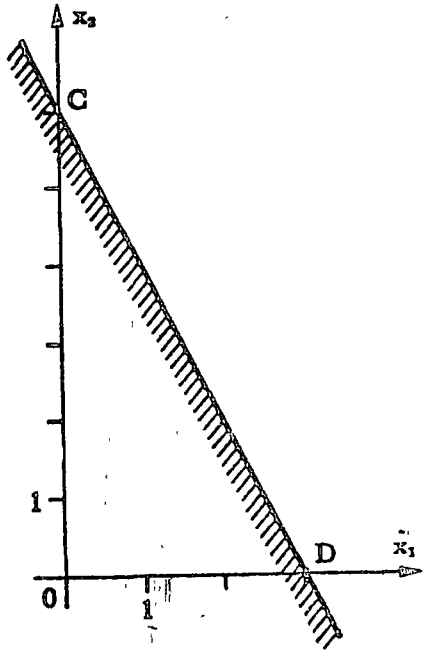


Fig. 6.1.2. Zona donde se cumple la restricción $2x_1 + x_2 \leq 6$.

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.

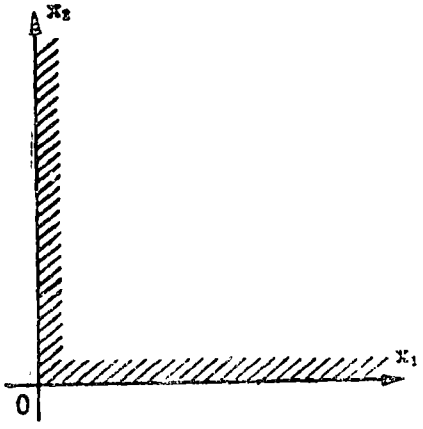


Fig. 6.1.3 Región donde se cumplen las rectricciones $x_1 > 0$ y $x_2 > 0$.

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.

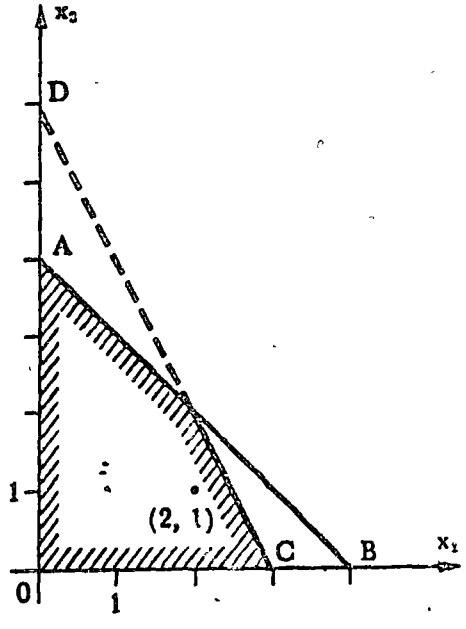


Fig. 6.1.4 Región donde se cumplen las restricciones $x_1 + x_2 \leq 4$, $2x_1 + x_2, \leq 6$.

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:

$$\begin{array}{rl} 2 + 1 & \leq 4 \\ 2.2 + 1 & \leq 6 \\ 2 & \geq 0 \\ 1 & \geq 0 \end{array}$$

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 } Funciones unimodales

B. Métodos de búsqueda multidimensional (varias variables dependientes)

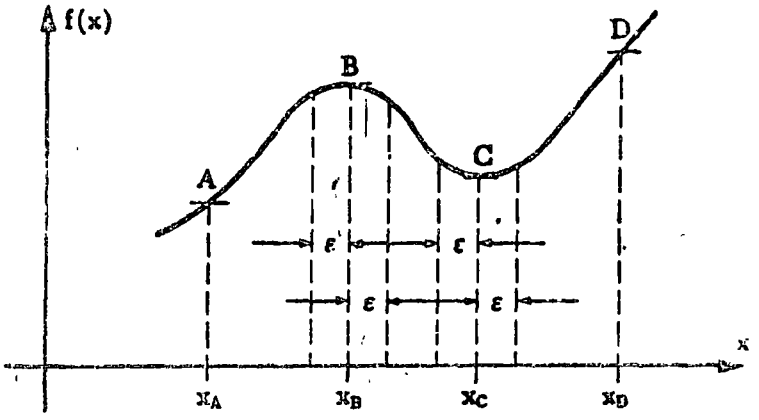
- a). Métodos simultáneos
  - 1. Búsqueda exhaustiva
  - 2. Búsqueda aleatoria
- b). Métodos secuenciales
  - 1. Búsqueda de rejilla
  - 2. Búsqueda univariada
  - 3. Métodos de gradiente
  - 4. Métodos de Fletcher-Powell
  - 5. Búsqueda de patrón.

} Funciones comparables } Funciones unimodales

*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.

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].

*La búsqueda directa encuentra máximos mínimos globales.
□Los métodos de diferenciación encuentran máximos o mínimos locales.



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.

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.

*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.

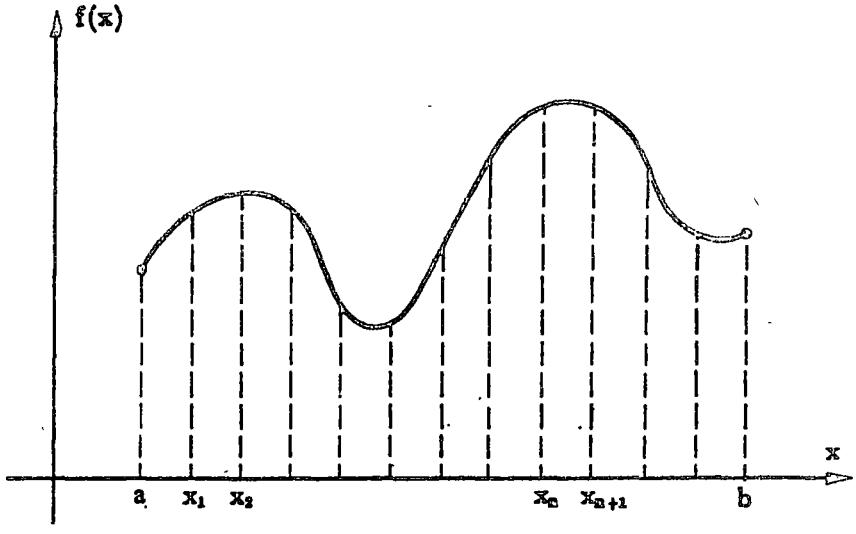
*Búsqueda secuencial → el siguiente valor x depende de valores previos de f(x).

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.

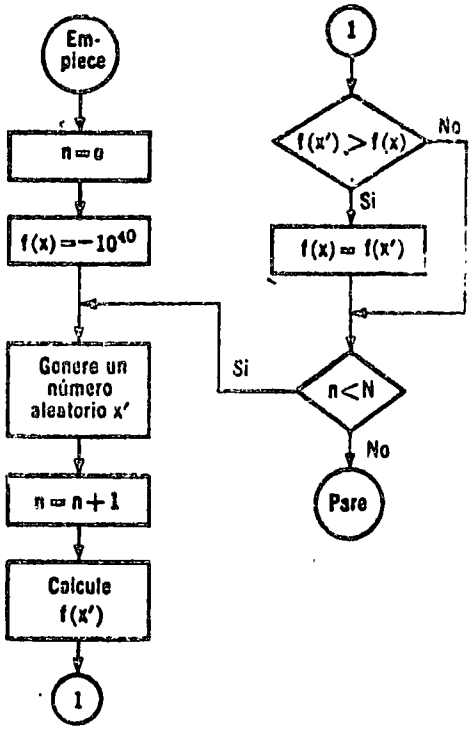


Fig. 6.2.4 Diagrama de bloque para el método de búsqueda aleatoria.

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:

$$y = -0.4\,x^3 + 4x$$

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:

$$\text{máx: } f(x) \;\Big|\; \begin{matrix} = 10 \\ x = 5 \end{matrix}$$

Tabla 6.2.3 Evaluación del máximo global de $y = -0.4x^2 + .4x$ 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 son ineficientes.

*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.

A continuación se estudian dos métodos de búsqueda simultánea, el de trisección y el de Fibonacci.

*Los métodos de búsqueda secuenciales requieren que la función sea unimodal.

*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 $x = x_0$ 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.

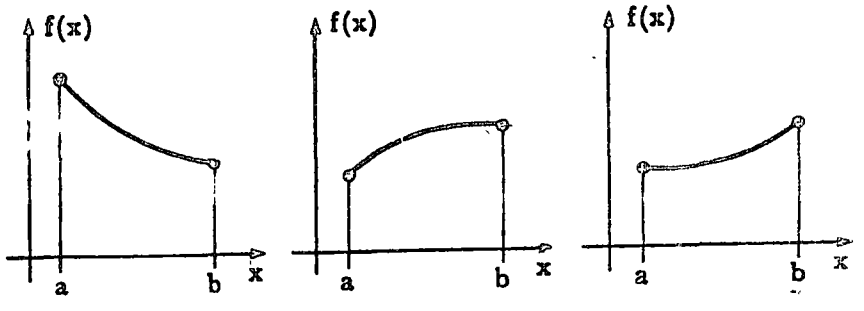


Fig. 6.2.5 Tres funciones unimodales en el intervalo [a, b]

*En el método de la trisección se subdivide el intervalo en 3 partes iguales.

*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

evalúa la función al centro del 1er. y 3er. intervalos (puntos x_1 y x_2), tal como muestra la figura 6.2.6. Los valores calculados de la función se comparan.

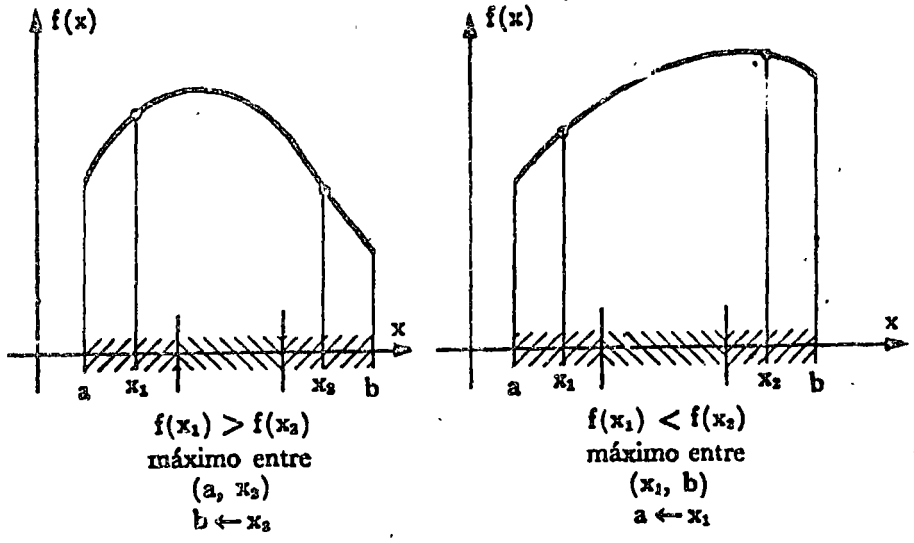


Fig. 6.2.6 Primer paso en la búsqueda del máximo por el procedimiento de trisección.

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.

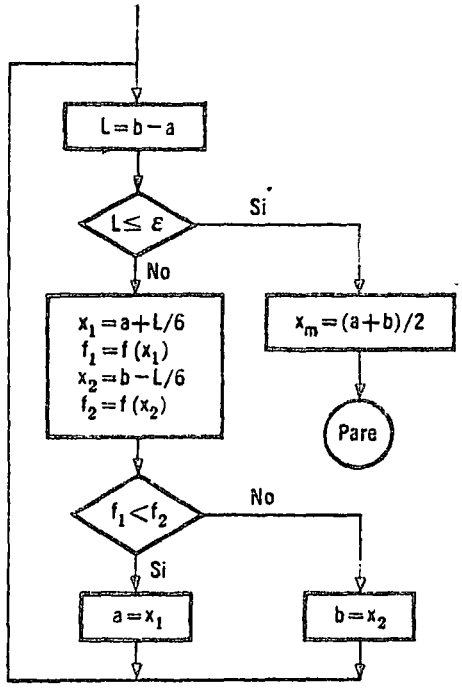


Fig. 6.2.7 Diagrama de flujo para la búsqueda de un máximo por el procedimiento de trisección.

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, popularizo 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 Fi-

bonacci para buscar el máximo o mínimo global en el intervalo [a, b] de una función *unimodal.*

Sea L_1 la longitud del intervalo [a, b]:

*Empleo de los números de Fibonacci en un proceso de búsqueda secuencial.

$$L_1 = b - a$$

*Al iniciarse el procedimiento de búsqueda se calcula la función unimodal f(x) en los dos puntos siguientes:

*1er. cálculo funcional

$$x_1 = a + \Delta_2$$
$$x_2 = b - \Delta_2 \qquad (6.2.27)$$

donde Δ_2 es igual a:

$$\Delta_2 = L_1 \frac{F_{n-2}}{F_n} \qquad (6.2.28)$$

Obsérvese que el cociente de los números de Fibonacci en la relación es:

$$\frac{F_{n-2}}{F_n} < \frac{1}{2} \qquad (6.2.29)$$

por lo que el intervalo Δ_2 definido por la relación (6.2.28) cumple con:

$$\Delta_2 \leq \frac{b-a}{2} \qquad (6.2.30)$$

*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 $\Delta_2 \leq \frac{b-a}{2}$ se descarta cualquiera de los dos intervalos siguientes:

*Se descarta por ser

$$\Delta_2 \leq \frac{b-a}{2}$$

$$(a,\ a+\Delta_2)$$
$$\text{ó}$$
$$(b-\Delta_2,\ b)$$

La figura 6.2.8 aclara este primer paso para un posible caso.

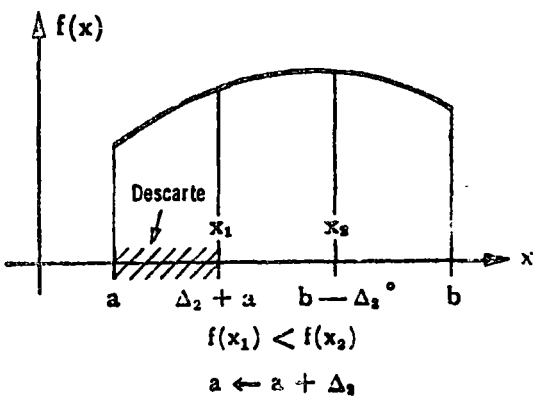


Fig. 6.2.8 Primer paso en una búsqueda secuencial.

*Observe que el intervalo en el qué 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.

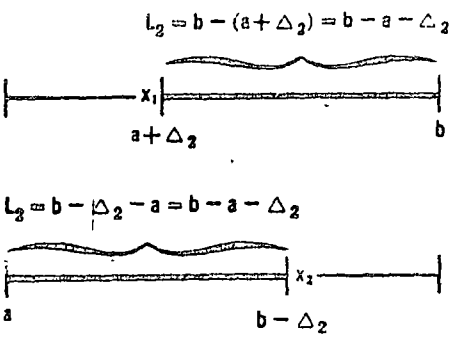


Fig. 6.2.9 Intervalos residuales L_2 después del 1er. paso.

Teniendo presente el valor de Δ_2:

$$\Delta_2 = L_1 \frac{F_{n-2}}{F_n} \qquad (6.2.28)$$

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

$$F_n = F_{n-1} + F_{n-2}$$

$$F_n - F_{n-2} = F_{n-1}$$

$$L_2 = L_1 \frac{F_{n-1}}{F_n} \qquad (6.2.31)$$

*A continuación se define de manera similar una distancia Δ_3 para dividir el intervalo que ha quedado después del 1er paso

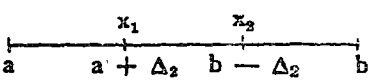
*Para dividir intervalo residual

$$\Delta_3 = L_2 \frac{F_{n-3}}{F_{n-1}} \qquad (6.2.32)$$

*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:



$$x_2 - x_1 = b - \Delta_2 - (a - \Delta_2)$$

$$= b - a - 2\Delta_2$$

$$= L_1 - 2\Delta_2$$

como $\Delta_2 = L_1 \frac{F_{n-2}}{F_n}$

$$x_2 - x_1 = L_1 \left(1 - 2\frac{F_{n-2}}{F_n}\right)$$

$$= L_1 \frac{F_n - 2F_{n-2}}{F_n}$$

$$= L_1 \frac{F_n - F_{n-2} - F_{n-2}}{F_n}$$

$$= L_1 \frac{F_{n-1} - F_{n-3}}{F_n}$$

y por la forma de generación de los números de Fibonacci

$$x_3 - x_1 = L_1 \frac{F_{n-3}}{F_n} \qquad (6.2.33)$$

pero de la relación (6.2.31)

$$L_2 = L_1 \frac{F_{n-1}}{F_n} \qquad (6.2.31)$$

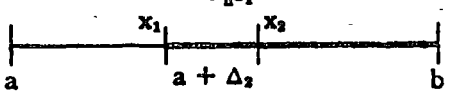
o

$$L_1 = L_2 \frac{F_n}{F_{n-1}}$$

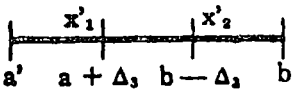
sustituyendo en (6.2.33) se obtiene

$$x_3 - x_1 = L_2 \frac{F_{n-3}}{F_{n-1}} = \Delta_3 \qquad (6.2.34)$$

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 + Δ_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 + Δ_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 $f(x'_2)$.



Intervalo residual a' ← a + Δ_2



Intervalo para el segundo paso

$\Delta_3 = x_2 - x_1$

$x'_1 = x_2$

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 − Δ_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:

$$L_3 = L_1 \left(\frac{F_{n-2}}{F_n} \right) \qquad (6.2.35)$$

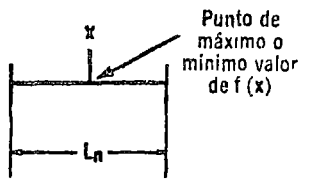
Continuando con este procedimiento, puede llegarse después de n pasos a la relación

$$L_n = L_1 \left(\frac{F_o}{F_n} \right) \qquad \frac{L_n}{L_1} = \left(\frac{F_o}{F_n} \right) \tag{6.2.36}$$

*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.

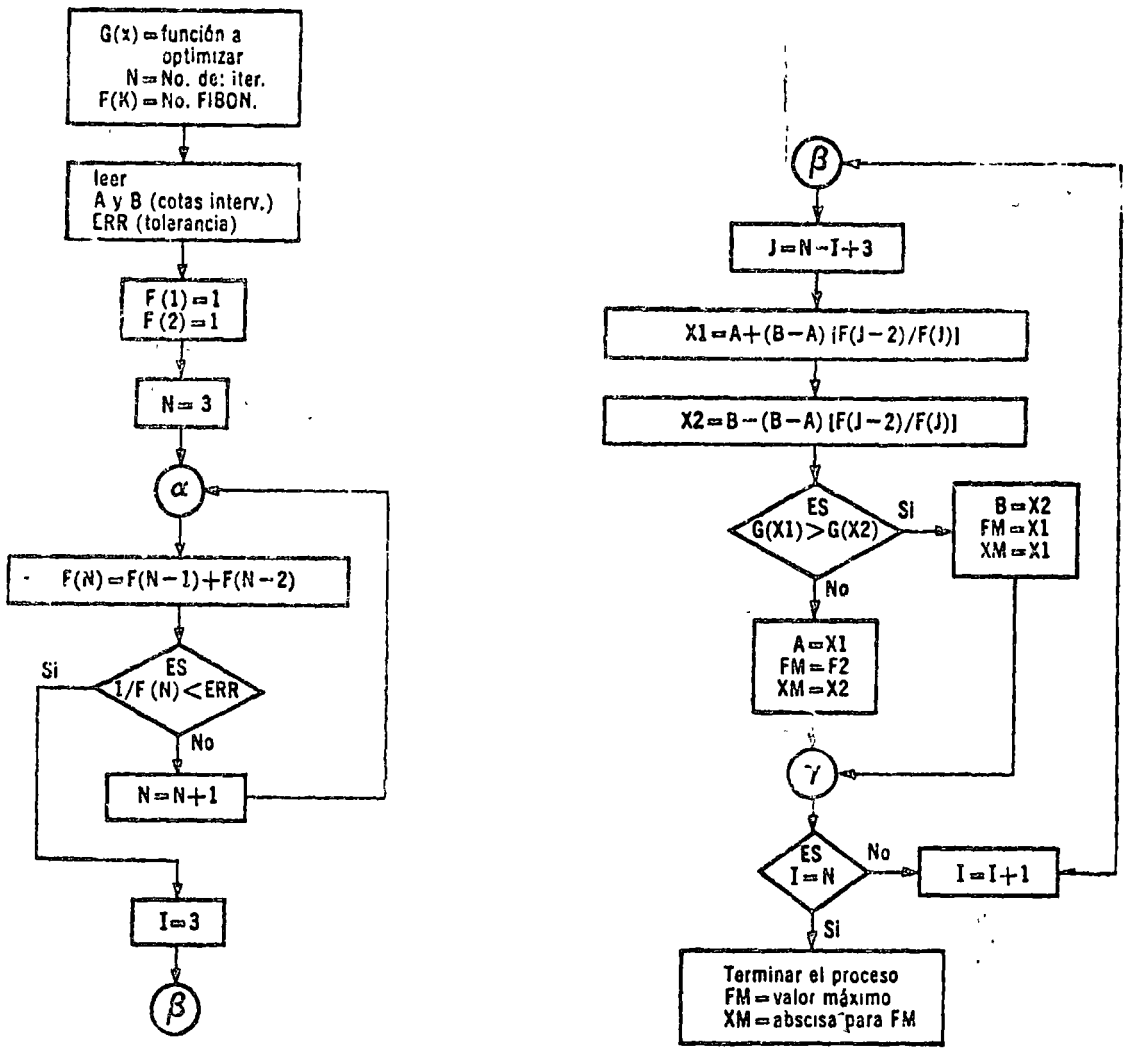


Fig. 6.2.11 Diagrama de flujo para la búsqueda de un máximo global de una función números de Fibonacci.

Tabla 6.2.5 Resultados del programa de maximización A.15 con la función $y = -0.4\,x^2 + 4x$.

	F_N	DELTA	$F(x_1)$	$F(x_2)$
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:

$$M = M(x_1, x_2, \ldots, x_n) \tag{6.1.1}$$

que hay que maximizar o minimizar, *se empieza encontrando para un punto $x_o = (x_{10}, x_{20}, \ldots x_{no})$ 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.

$$M(x_{10}, x_{20}, \ldots, x_{no}) < M(x_{10} + \Delta x_1, x_{20} + \Delta x_2, \ldots, x_{no} + \Delta x_n) \tag{6.3.1}$$

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:

$$\left.\frac{\delta M}{\delta x_1}\right|_{\underline{x}_0} \cong \frac{M(x_{10} + \Delta x_1, x_{20}) - M(x_{10}, x_{20})}{\Delta x_1}$$

$$\cong \frac{\Delta M'}{\Delta x_1} \qquad (6.3.2)$$

La figura 6.3.1 ilustra la evaluación de esta derivada. En esta figura

$$\text{tg } \alpha = \left.\frac{\delta M}{\delta x_1}\right|_{\underline{x}_0} = \frac{\Delta M'}{\Delta x_1}$$

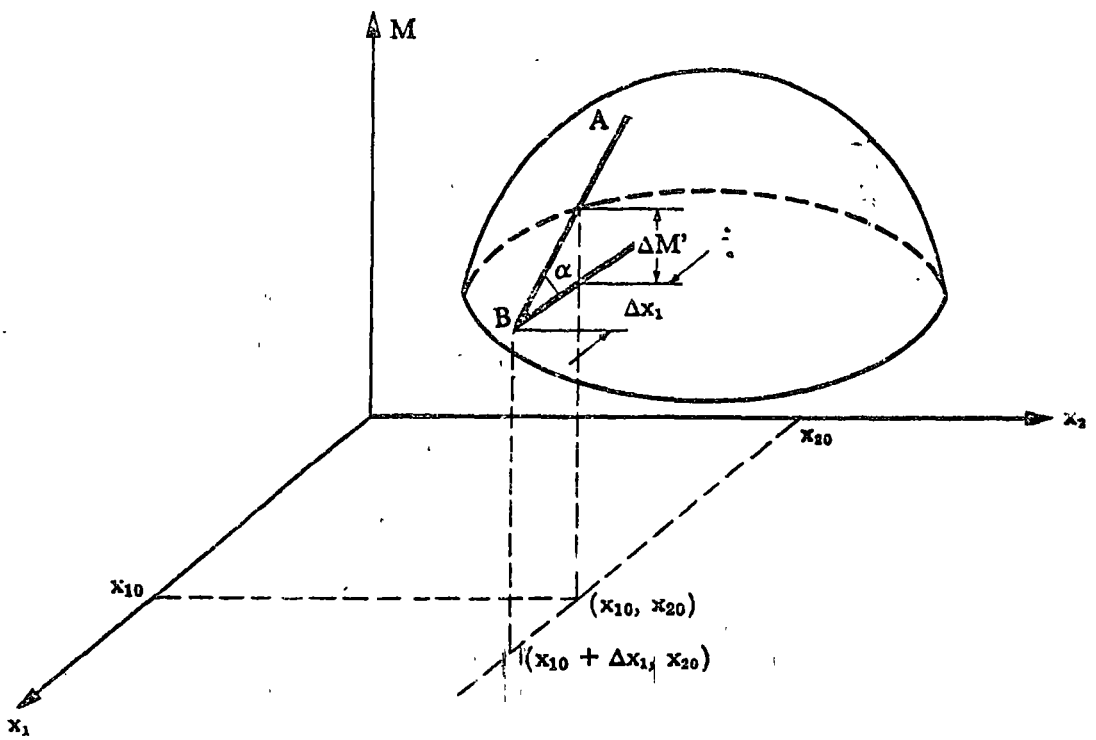


Fig. 6.3.1 Cálculo de la derivada parcial $\frac{\delta M}{\delta x_1}$

Para el cálculo de la derivada parcial $\frac{\delta M}{\delta x_2}$ se emplea la siguiente relación:

$$\left.\frac{\delta M}{\delta x_2}\right|_{x_0} \cong \frac{M(x_{10}, x_{20}+\Delta x_2) - M(x_{10}, x_{20})}{\Delta x_2} \cong \frac{\Delta M''}{\Delta x_2} \tag{6.3.3}$$

La figura 6.3.2 ilustra el cálculo de esta derivada parcial. En esta última figura

$$\operatorname{tg}\beta = \left.\frac{\delta M}{\delta M}\right|_{x_0} = \frac{\Delta M^{M''}}{\Delta x_2}$$

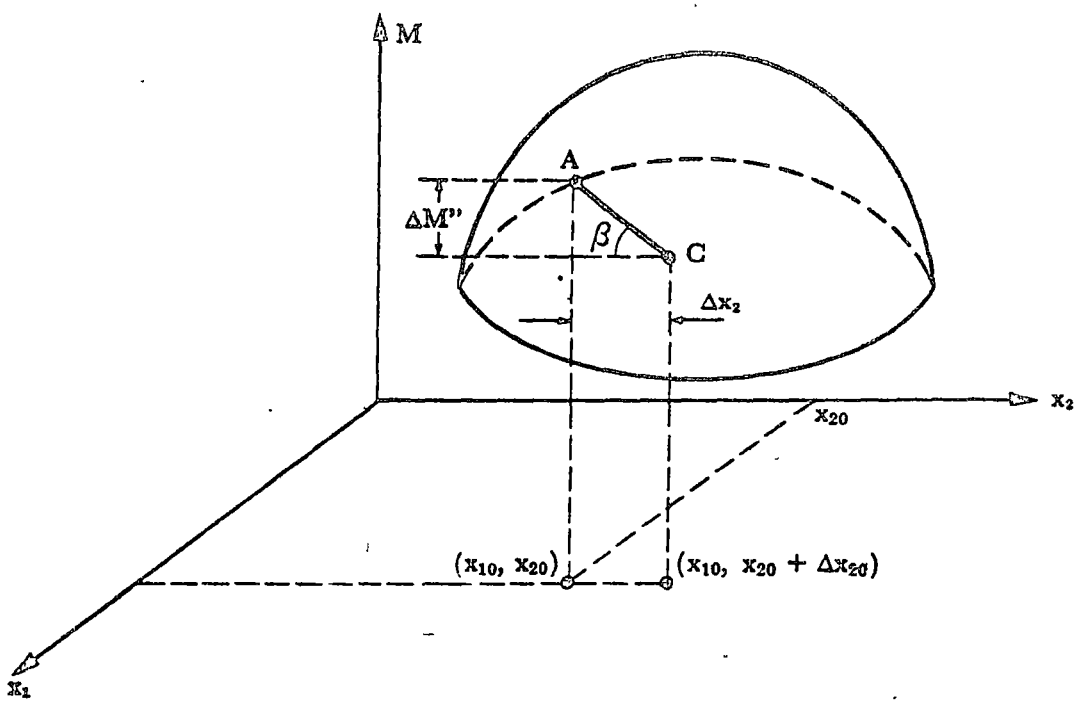


Fig. 6.3.2 Cálculo de la derivada parcial $\frac{\delta M}{\delta x_2}$

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}).

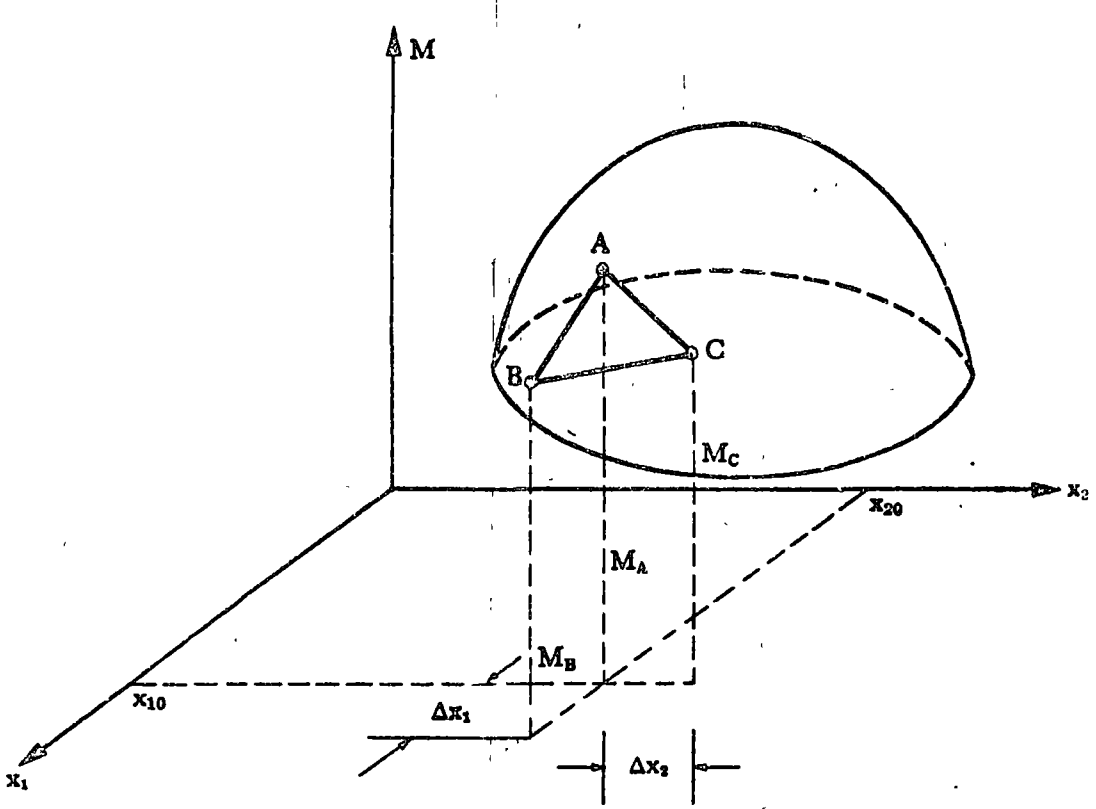


Fig. 6.3.3 Determinación del plano tangente a una superficie.

A continuacion 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:

$$m_1 = \left.\frac{\delta M}{\delta x_1}\right|_{\underline{x}_0} \tag{6.3.8}$$

De la formula (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:

$$m_2 = \left.\frac{\delta M}{\delta x_2}\right|_{\underline{x}_0} \tag{6.3.9}$$

*Para no tener que evaluar en la relación (6.3.4) la constante m_0 conviene emplear como variable independiente los incrementos de la función $M(x_1, x_2)$.

*Evítese el cálculo de m_0

Para un punto D en la vecindad del punto A, con coordenadas $x_{10} + \Delta x_1$, $x_{20} + \Delta x_2$ se tiene:

$$M_D = m_0 + m_1 (x_{10} + \Delta x_1) + m_2 (x_{20} + \Delta x_2) \tag{6.3.10}$$

Restando esta ecuación de la correspondiente al punto A se tiene:

$$\Delta M = M_D - M_A = m_1 \Delta x_1 + m_2 \Delta x_2 \tag{6.3.11}$$

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:

$$\Delta M = \left.\frac{\delta M}{\delta x_1}\right|_{\underline{x}_0} \Delta x_1 + \left.\frac{\delta M}{\delta x_2}\right|_{\underline{x}_0} \Delta x_2 \tag{6.3.12}$$

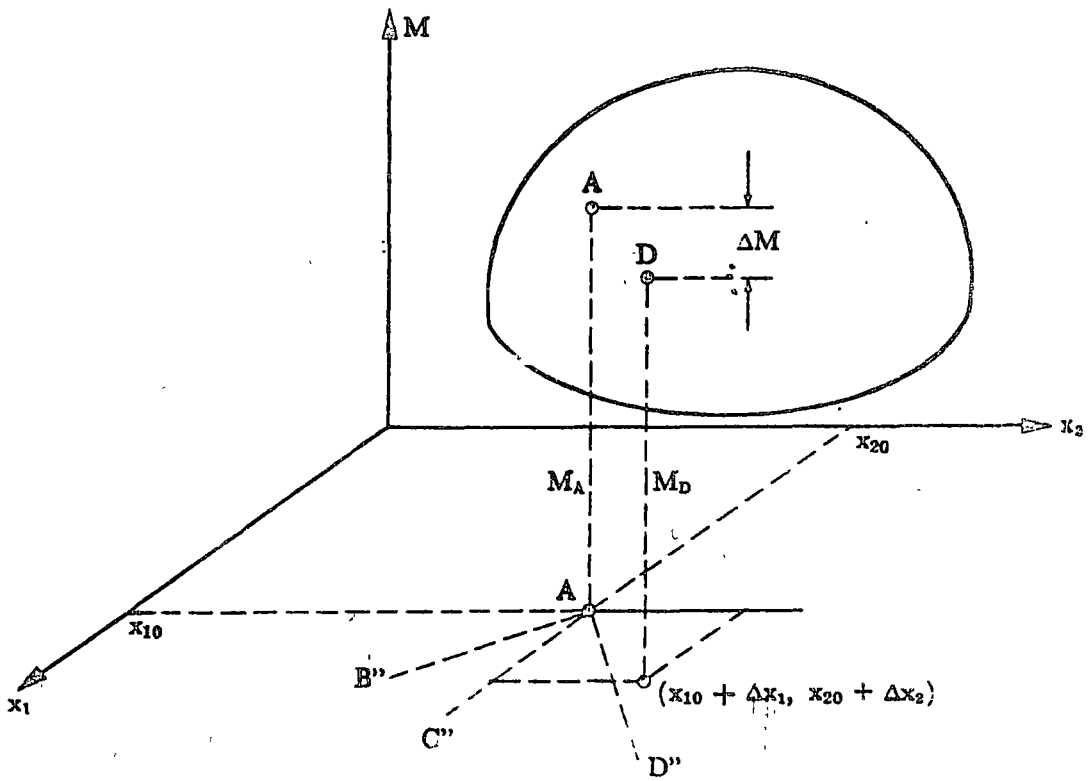


Fig. 6.3.4 Ilustración del cálculo del incremento de la función $M(x_1, x_2)$.

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.

$$\nabla M = \left(\frac{\delta M}{\delta x_1} \quad \frac{\delta M}{\delta x_2} \right) \qquad (6.3.13)$$

*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.

$$\Delta M\ (\Delta x_1, \Delta x_2) = \nabla M\Big|_{\underline{x}_0} \Delta \underline{x} \qquad (6.3.15)$$

Esta 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)

$$\Delta M\Big|_{\underline{x}_0} = \nabla M\Big|_{\underline{x}_0} \Delta \underline{x} \qquad (6.3.15)$$

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 $M\ (x_0)$

$$M\ (x_{10} + \Delta x_1, x_2 + \Delta x_2, \ldots, x_n + \Delta x_n) = M\ (x_{10}, x_{20}, \ldots, x_n) + \Delta M\Big|_{\underline{x}_0}$$

donde $\Delta M\Big|_{\underline{x}_0}$ está dado por la relación (6.3.15).

*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 $M(\underline{x})$

con respecto a la i' si una variable está dada por:

*Para evaluar el gradiente hay que calcular derivadas parciales.

$$\frac{\delta M}{\delta x_i} = \frac{M(x_{10}, x_{20}, \ldots, x_{i-1,0}, x_{i0} + \Delta x_i, x_{i+1,0}, \ldots x_n) - M(x_{10}, \ldots X_{i0}, \ldots x_{n0})}{\Delta x_i} \qquad (6.3.16)$$

Esta fórmula señala que la i'sima derivada parcial puede obtenerse calculando el incremento de la función $M(\underline{x})$ 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

*Dada la función $y = x_1^2 x_2^3$ calcule el valor de la función para

$$\underline{x}_0 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

y para

$$\underline{x} = \begin{bmatrix} 1.05 \\ 1.1 \end{bmatrix}$$

empleando la relación (6.3.15) y directamente por sustitución.

$y = x_1^2 x_2^3$

calcule

$y\big|_{(1,1)}$ y $y\big|_{(1.05,1)}$

El valor de la función en el punto $\underline{x}_0 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$ es:

Solución

$$y\Big|_{(1,1)} = 1^2\, 1^3 = 1 \qquad (6.3.17)$$

El valor de las derivadas parciales de la función puede calcularse empleando la relación (6.3.16). Para la 1er. derivada parcial se tiene:

$$\left.\frac{\delta y}{\delta x_1}\right|_{\underline{x}_0} = \frac{(1.05)^2(1)^3 - (1)^2(1)^3}{0.05} = 2.05 \qquad (6.3.18)$$

y para la 2da. derivada parcial:

$$\left.\frac{\delta y}{\delta x_2}\right|_{\underline{x}_0} = \frac{(1)^2(1.1)^3 - (1)^2(1)^3}{0.1} = 3.31 \qquad (6.3.19)$$

Estos valores son solamente aproximados, el valor exacto de estas derivadas es:

$$\left.\frac{\delta y}{\delta x_1}\right|_{\underline{x}_0} = 2x_1 x_2^3 = 2.1.1^3 = 2$$

$$\left.\frac{\delta y}{\delta x_2}\right|_{\underline{x}_0} = x_1^2 3x^2 = 1^2 3.1^2 = 3$$

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

$$\underline{x} = \begin{bmatrix} 1.05 \\ 1.1 \end{bmatrix}$$

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.

3.3.2. 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 $M(\underline{x})$ varía 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.

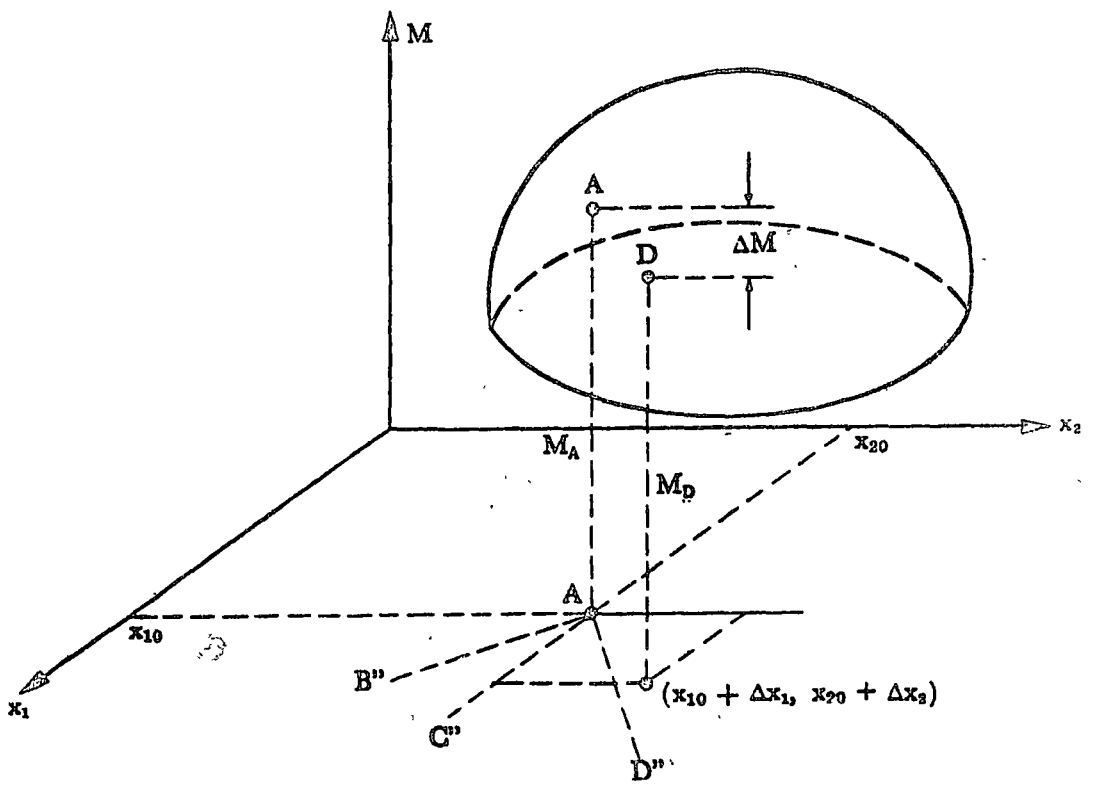


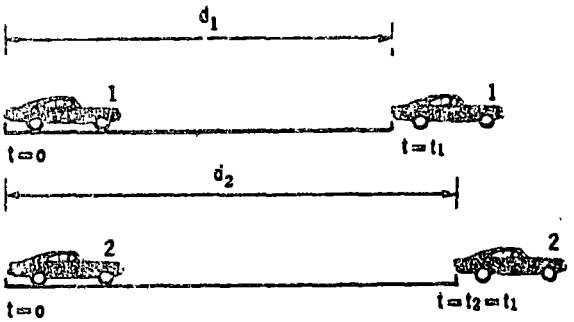
Fig. 6.3.4 Ilustración del cálculo del incremento de la función $M(x_1, x_2)$ (repetición).

*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 x sobre este círculo y la

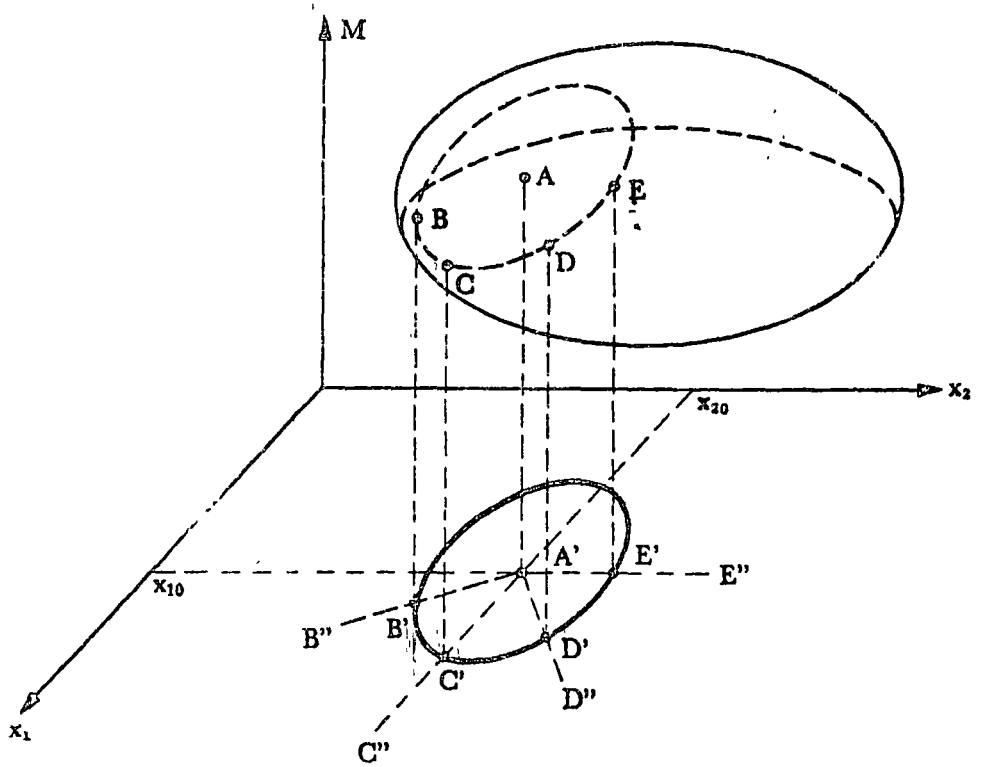


Fig. 6.3.6 Búsqueda por gradiente.

magnitud de $\underline{x}_o$ difieren precisamente en el radio r del círculo, como ilustra la figura 6.3.7.

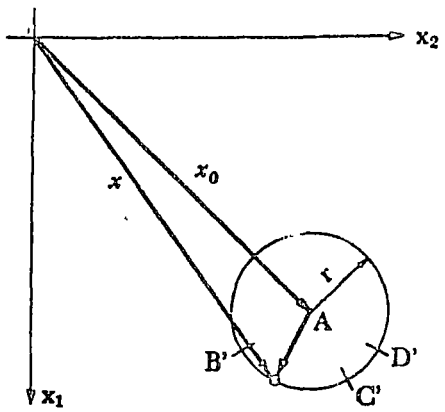


Figura 6.3.7 Variaciones del vector x.

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

$$\underline{x} = \begin{vmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_n \end{vmatrix}$$

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:

$$L(x_1, x_2, \lambda) = \nabla M\, \Delta x - \lambda \{\Delta x^2 - r^2\}$$

$$= \left.\frac{\delta M}{\delta x_1}\right|_{\underline{x}_0} \Delta x_1 + \left.\frac{\delta M}{\delta x_2}\right|_{\underline{x}_0} \Delta x_2 - \quad \lambda \{\Delta x_1^2 + \Delta x_2^2 - r^2\}$$

Igualando a cero las derivadas parciales se tiene

$$\frac{\delta L}{\delta \Delta x_1} = \left.\frac{\delta M}{\delta x_1}\right|_{\underline{x}_0} - 2\lambda\, \Delta x_1 = 0$$

$$\frac{\delta L}{\delta \Delta x_2} = \left.\frac{\delta M}{\delta x_2}\right|_{\underline{x}_0} - 2\lambda\, \Delta x_2 = 0$$

$$\frac{\delta L}{\delta \lambda} = -\Delta x_1^2 - \Delta x_2^2 + r^2 = 0$$

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:

$$\Delta \underline{x} = \frac{1}{2\lambda} \nabla M \Big|_{\underline{x}_0}^{T} \qquad (6.3.22a)$$

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 $\underline{z}_0$ para inicializar la búsqueda.

Paso 2:

Evalúe el gradiente en ese punto empleando las relaciones (6.3.13a) y (6.3.16).

$$\nabla M\Big|_{\underline{x}_o} = \left(\frac{\delta M}{\delta x^1}, \frac{\delta M}{\delta x_2}, \ldots, \frac{\delta M}{\delta x_n} \right)_{\underline{x}_o} \qquad (6.3.13a)$$

$$\frac{\delta M}{\delta x_i}\Big| = \frac{M(x_{1o}, \ldots x_{i-1,o}, x_{io} + \Delta x_i, x_{i+1,o}, \ldots x_{no}) - M(x_{1o}, \ldots, x_{io}, \ldots x_{no})}{\Delta x_i} \qquad (6.3.16)$$

Paso 3:

Calcule el incremento de la variable independiente de acuerdo con la relación

$$\Delta \underline{x} = \varrho \nabla M\Big|_{\underline{x}_o}^{T} \qquad (6.3.22b)$$

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 $\underline{x}_1$, para el cual se tiene que

$$M(\underline{x}_o) < M(\underline{x}_1)$$

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:

Dè 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:

$$\mathbf{x} = \begin{bmatrix} 1 \\ -1 \end{bmatrix}^0$$

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

$$\mathbf{x}_0 = \begin{bmatrix} 0.1 \\ 0.1 \end{bmatrix}$$

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 $M(\underline{x}_0) = M(\underline{x}_1)$ puede suceder que el máximo (o mínimo) se encuentre entre los dos puntos, o que entre $\underline{x}_0$ y $\underline{x}_0$ la función no cambie para seguir creciendo posteriormente.

*El máximo o mínimo de la función puede encontraste entre el punto $\underline{x}_0$ y el $\underline{x}_1$.

Para determinar si se ha presentado el 2do. caso, puede hacerse una búsqueda incrementando el valor de ϱ_0 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 $\underline{x}_0$ 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:

$$y(x_1, x_2) = 50\,|x_2| + 15\,\{x_1^2 + (x_2 - 300)^2\}^{1/2} + 50\,\{x_1^2 + (1300 - x_2)^2\}^{1/2} +$$
$$15\,\{(1300 - x_1)^2 + (900 - x_2)^2\}^{1/2} + 20\,\{(1300 - x_1)^2 + x_2^2\}^{1/2}$$

* Ref. 1, Cap. 10.

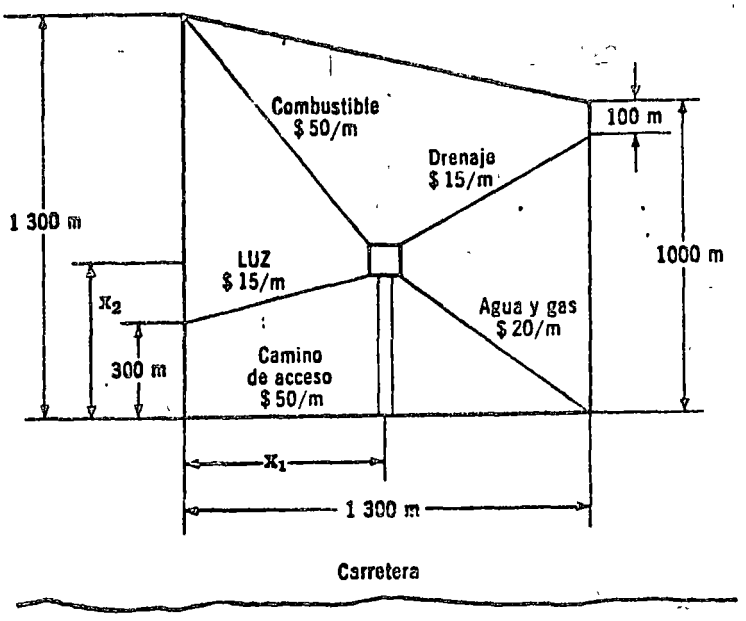


Fig. 6.3.8 Plano de localización de la fábrica del ejemplo 6.3.3.

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

Q_0	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

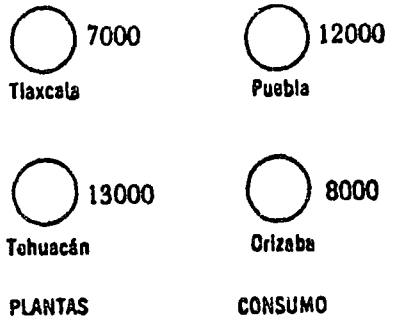


Tabla 6.5.1 Costos de transporte en el ejemplo 6.4.1.

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:

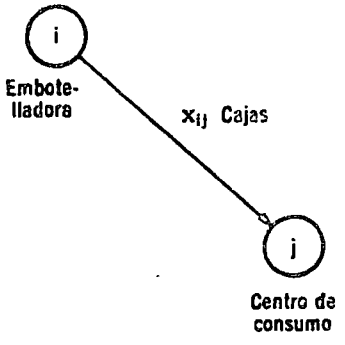
1) Cada embotelladora no puede enviar más cajas que el máximo que puede producir.
2) Cada centro de consumo puede obtener tantas cajas como puede consumir.
3) Deben minimizarse los gastos de transporte.

Para plantear este problema en el marco de las ecuaciones (6.1.1) y (6.1.2).

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

$$C_i = C_i(x_1, x_2, \ldots, x_n) \geq 0 \text{ para } i = 1,2, \ldots p$$
$$C_i = C_i(x_1, x_2, \ldots, x_n) \leq 0 \text{ para } i = p+1, \ldots r$$
$$C_i = C_i(x_1, x_2 \ldots, x_n) = 0 \text{ para } i = r+1, \ldots n \qquad (6.1.2)$$

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:

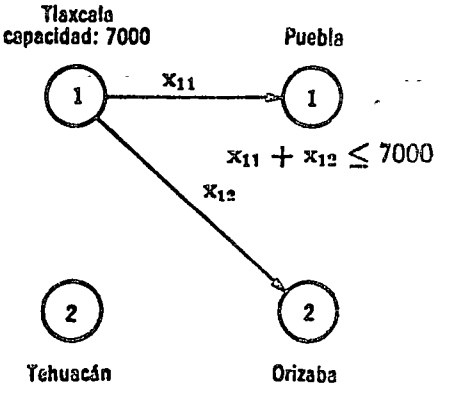


Fig. 6.5.1 Cajas enviadas desde la embotelladora en Tlaxcala.

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.

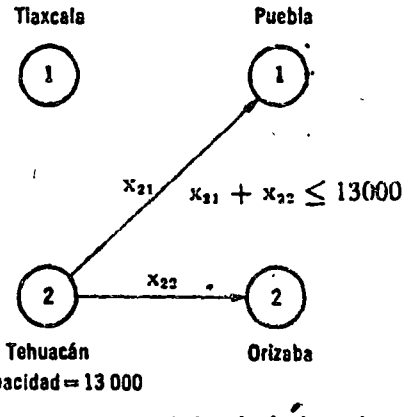


Fig. 6.52 Cajas enviadas desde la embotelladora en Tehuacán.

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:

$$x_{11} + x_{21} \geq 12000 \qquad (6.5.3)$$

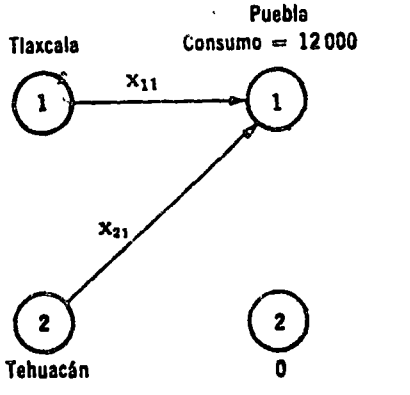


Fig. 6.5.3 Cajas recibidas en Puebla.

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.

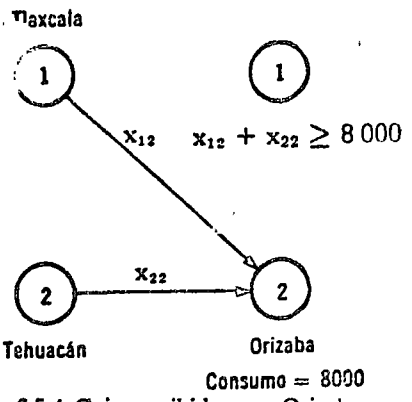


Fig. 6.5.4 Cajas recibidas por Orizaba.

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:

$$M = 0.8\,x_{11} + 1\,x_{21} + 1.3\,x_{21} + 0.9\,x_{22} \tag{6.5.5}$$

Debe además imponerse la siguiente condición:

$$x_{i}i \geq 0\ \forall_i,\ \forall i \tag{6.5.6}$$

ya que no tendrán significado valores negativos de envíos de cajas.

En resumen puede decirse que el problema consiste en minimizar la función objetivo.

$$M = 0.8\,x_{11} + 1.\,x_{21} + 1.3\,x_{12} + 0.9\,x_{22} \tag{6.5.5}$$

Sujeto a las restricciones

$$\begin{aligned} x_{11} + x_{12} &\leq 7{,}000 && (6.5.1)\\ x_{21} + x_{22} &\leq 13{,}000 && (6.5.2)\\ x_{11} + x_{21} &\geq 12{,}000 && (6.5.3)\\ x_{12} + x_{22} &\geq 8{,}000 && (6.5.4) \end{aligned}$$

$$x_{ij} \geq 0,\ \forall_i \text{ y } \forall_j. \tag{6.5.6}$$

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:

*Variables del problema x_{11}, x_{12}, x_{21} y x_{22}

Ejemplo 6.5.2

*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

*x_1 camionetas de 2 ton.. x_2 camionetas de 4 ton.

$$m = 2x_1 + 4x_2 \qquad (6.5.7)$$

y el problema consiste en maximizar dicha expresión.

*Además la compañía tiene las siguientes restricciones:

*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:

*Mantenimiento:
24 días mecánico/mes.

$$x_1 + 4x_2 \leq 24 \qquad (6.5.8)$$

*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:

*2da. Andenes de carga:
9 andenes.

$$x_1 + x_2 \leq 9 \qquad (6.5.9)$$

*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

*3ra. Cargado:
21 personas.

$$3x_1 + x_2 \leq 21 \qquad (6.5.10)$$

*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:

*Ultima:
no negatividad.

$$x_1, \geq 0;\ x_2 \geq 0 \qquad (6.5.11)$$

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.

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.

°Operacion 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

$$x_i,\ i = 1, 2, \ldots n \text{ y } x'_i,\ i = 1, 2, \ldots$$

*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:

$$m = c_1 x_1 + c_2 x_2 + \ldots + c_n x_n \qquad (6.5.16a)$$

sujeto a las siguientes restricciones:

y satisfaga:

$$ai_1 x_1 + a_{12} x_2 + \ldots + a_{in} = b_i \quad i = 1, 2, \ldots p$$

$$a_{i1} x_1 + a_{i2} x_2 + \ldots + a_{in} x_n \leq b_i \quad i = p + 1, \ldots r$$

$$a_{i1} x_1 + a_{i2} x_2 + \ldots + a_{in} x_n \geq b_i \quad i = r + 1, \ldots m$$

$$x_j \geq 0 \quad j = 1, 2 \ldots n \qquad (6.5.16b)$$

*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

$$a_{q1} x_1 + a_{q2} x_2 + \ldots + a_{qn} x_n + x_{n+q} = b_q$$

puede convertirse en una igualdad introduciendo una variable positiva x_{n+q} llamada de holgura. En efecto:

*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

la introducción de la variable de holgura positiva x_{n+q}, convierte la desigualdad en una igualdad, ya que:

$$a_{q1}\,x_1 + a_{q2}\,x_2 + \ldots + a_{qn}\,x_n - x_{n+q} = b_q$$

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_j 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:

$$n \equiv -m$$
$$\max: n = -3x_1 - 5x_2$$

la función objetivo se convierte en:

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.

* La estructura del problema de programación lineal se presta para el empleo de la notación matricial. Si se definen *la matriz de coeficientes estructurales

*Formulación matricial
*Coeficientes estructurales

$$\underline{A} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & \cdots & & \\ \vdots & & & \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \quad (6.5.17)$$

* los vectores de actividades:

*Actividades

$$\underline{x} = \begin{bmatrix} x_1 \\ x_n \\ \vdots \\ x_n \end{bmatrix} \quad (6.5.18)$$

de *costos

*Costos

$$\underline{c} = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix} \quad (6.5.19)$$

y *de restricciones

*Restricciones

$$\underline{b} = \begin{bmatrix} b_1 \\ b_2 \\ \\ b_n \end{bmatrix} \quad (6.5.20)$$

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

$$\underline{A}\,\underline{x} \leq \underline{b} \quad (6.5.22)$$

$$\underline{x} \geq 0 \quad (6.5.23)$$

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:

$$\max: m = 2x_1 + 4x_2 \quad (6.5.$$

Sujeto a las restricciones

$$x_1 + 4x_2 \leq 24 \quad (6.5.8)$$
$$x_1 + x_2 \leq 9 \quad (6.5.9)$$
$$3x_1 + x_2 \leq 21 \quad (6.5.10)$$
$$x_1, x_2 \geq 0$$

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.

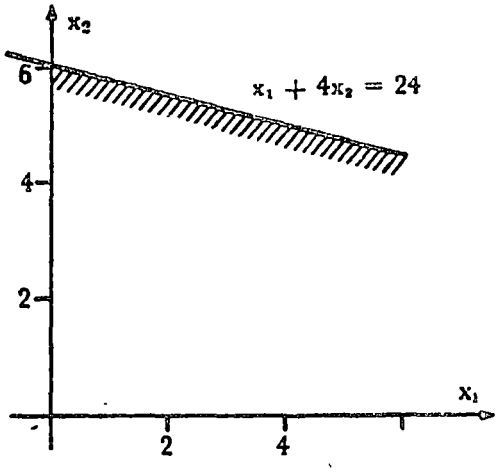


Fig. 6.5.5 Zona con restricción $x_1 + 4x_2 \leq 24$.

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).

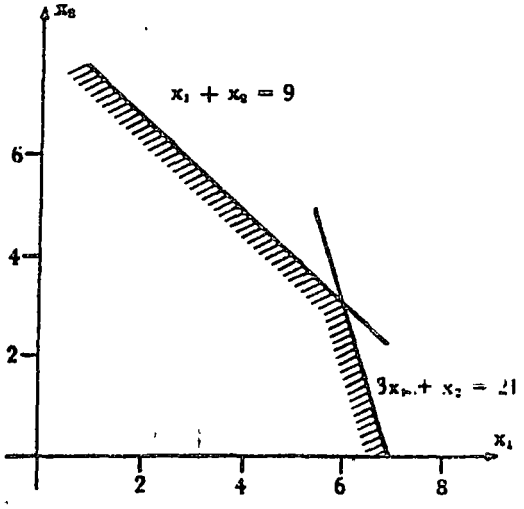


Fig. 6.5.6 Zona con restricciones $x_1 + x_2 \leq 9$ y $3x_1 + x_2 \leq 21$.

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.

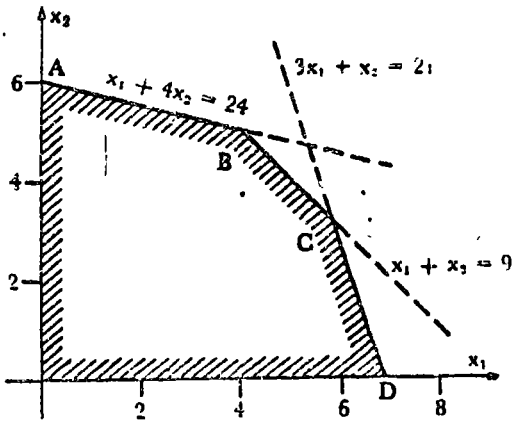


Fig. 6.5.7 Zona de soluciones factibles del ejemplo 6.5.2.

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.

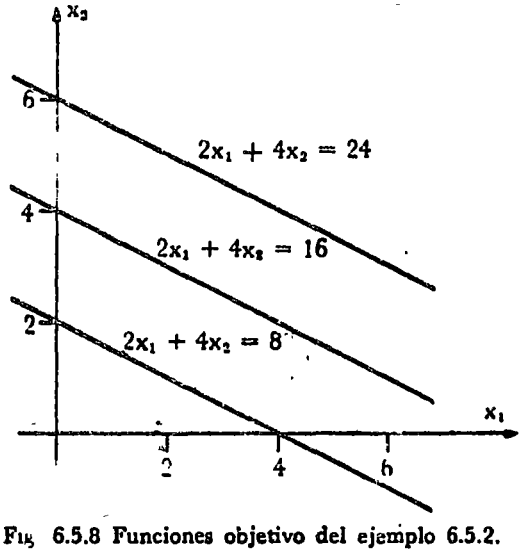


Fig. 6.5.8 Funciones objetivo del ejemplo 6.5.2.

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.

$$x_1 + 4x_2 \leq 24 \text{ (a)}$$
$$x_1 + x_2 \leq 9 \text{ (b)}$$
$$3x_1 + x_2 \leq 21 \text{ (c)}$$
$$x_1 \geq 0 \text{ (d)}$$
$$x_2 \geq 0 \text{ (e)}$$

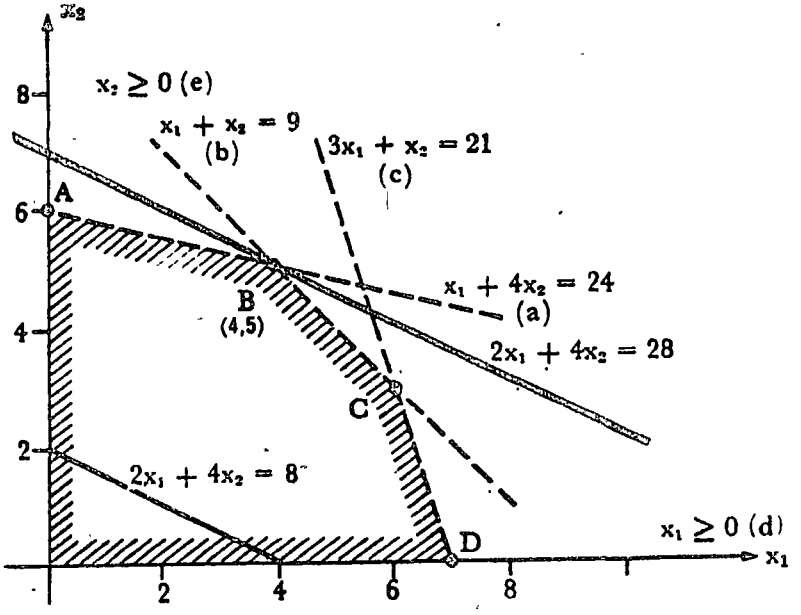


Fig. 6.5.9 Ilustración de la solución gráfica del problema de programación lineal.

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.

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.

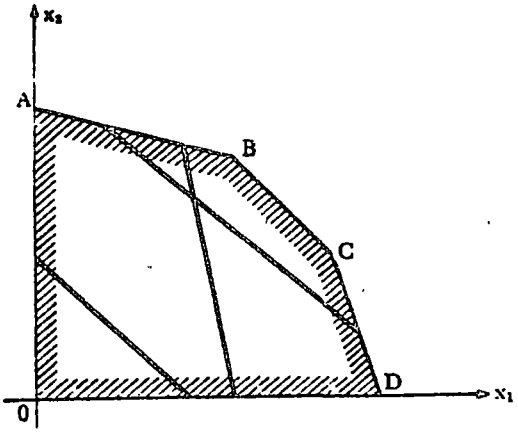


Fig. 6.5.10 Zona convexa de soluciones factibles.

*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

*Método gráfico para problemas con dos variables.

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, $\tilde{\underline{x}}$ 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 $\tilde{\underline{x}}$.

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

$$r = r\ (\underline{x}, \underline{D}, \tilde{\underline{x}})$$

5). Una transformación T, univaluada que relaciona los estados finales, con los estados iniciales, y las variables de decisión.

$$\underline{x} = T\ (\underline{x}, \underline{D}) \qquad (6.6.2)$$

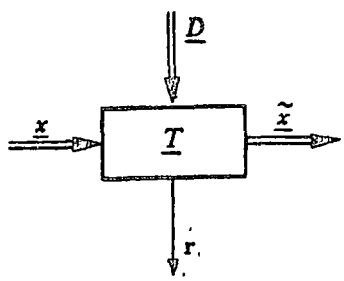


Fig. 6.6.1 Modelo de un problema de toma de decisión.

$$r = r\ (\underline{x}, \underline{D}, \underline{T}\ (\underline{x}, D))$$

*Función de beneficio

$$r = r'\ (\underline{x}, \underline{D}) \qquad (6.6.3)$$

$$\underline{x} = T'\ (\underline{x}, \underline{D})$$

$$r = r\ (T^{o}\ (\tilde{\underline{x}}, \underline{D}), D\ \tilde{\underline{x}})$$

$$r = r''\ (\tilde{\underline{x}}, \underline{D}) \qquad 6.6.4)$$

*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:

En el problema de *estado final* x, debe determinarse el máximo (o mínimo) del beneficio como función del estado final, es decir:

Con objeto de facilitar la presentación del material subsecuente e ilustrar la naturaleza de estos problemas, conviene introducir algunos símbolos:

•Optimización de estado inicial $\underline{x}$

$$f(\underline{x}) = \max_{\underline{D}} r' (\underline{x}, \underline{D}) \qquad (6.6.5)$$

(6.6.5)

•Optimización de estado final $\underline{x}$

$$f(\underline{\tilde{x}}) = \max_{\underline{D}} r'' (\underline{\tilde{x}}, \underline{D}) \qquad (6.6.6)$$

•Símbolos empleados en programación dinámica.

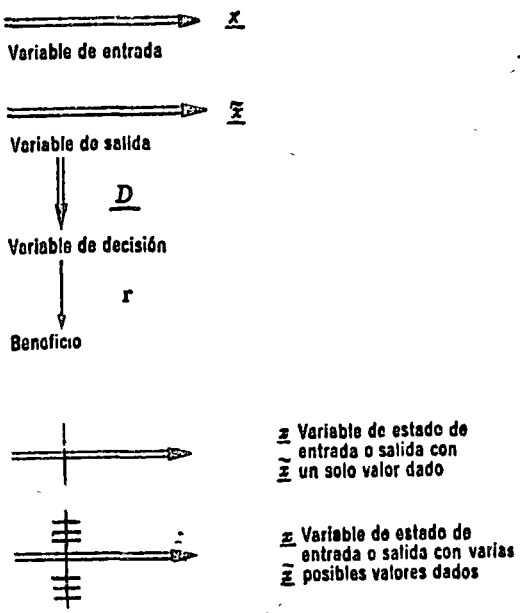
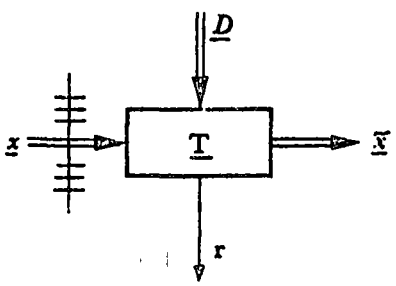


Fig. 6.6.2 Símbolos en problemas de programación dinámica.

Usando estos símbolos el problema de valor inicial puede simbolizarse:



$$f(\underline{x}) = \max_{\underline{D}} r (\underline{x}, \underline{D})$$

Fig. 6.6.3 Problema de valor inicial.

y el de valor final

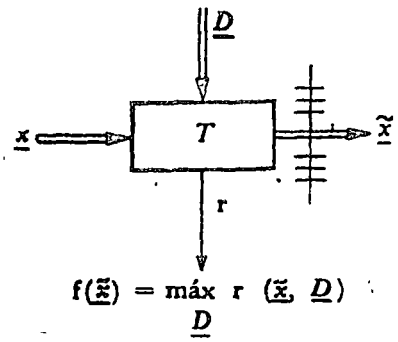


Fig. 6.6.4 Problema de valor final.

*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. ⟹
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.

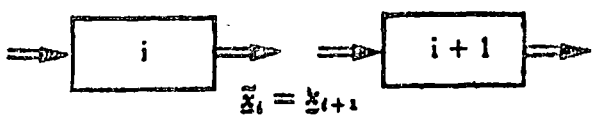
*Se asignan recursos a un proyecto a la vez.

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.

**Gere, 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.

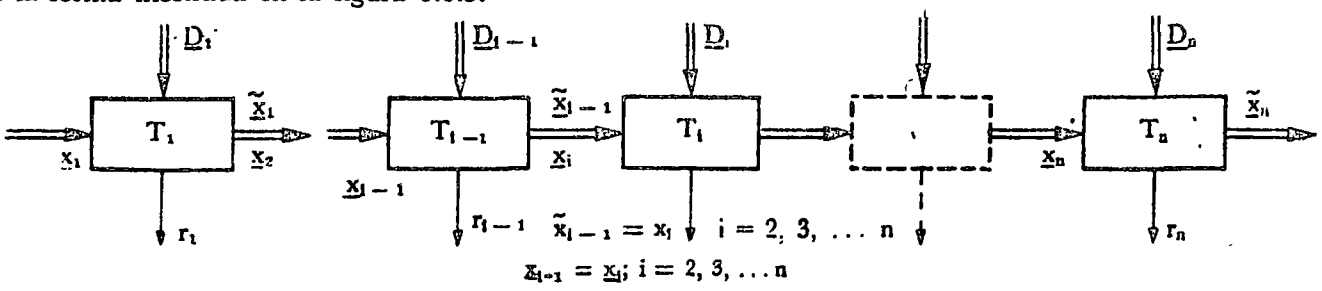


Fig. 6.6.5 Estructura en serie.

A continuación se hace una presentación formal del principio de optimalidad y se deduce la fórmula recursiva para resolver este tipo de problemas.

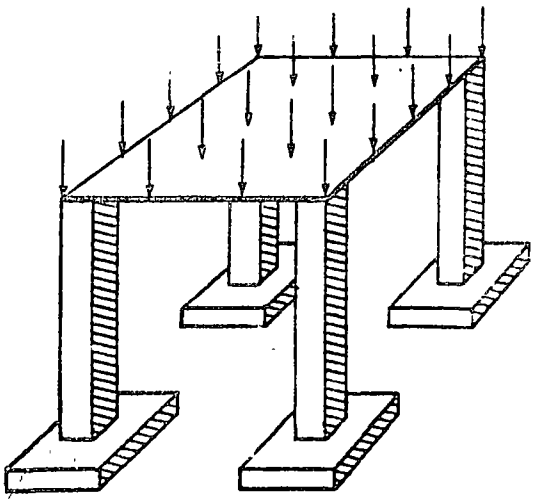
6.6.3 Principio de optimalidad

*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 $\underline{x}$, 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).

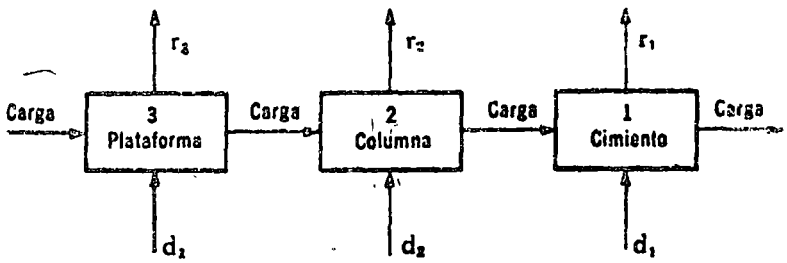
La figura 6.6.6a muestra una plataforma que debe soportar una carga dada de ωkg/m². El objetivo del problema es diseñar una plataforma, las columnas de soporte y los cimientos necesarios para soportar el peso minimizando el costo de la obra. Para aplicar la técnica de la programación dinámica a este 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 dinamica.

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.

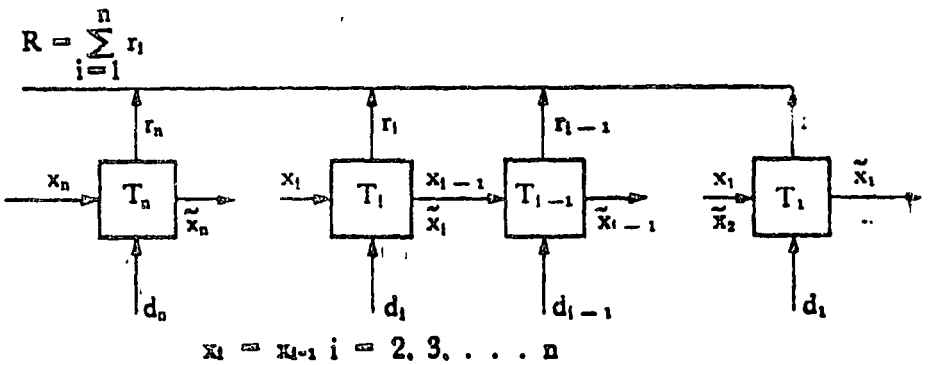


Fig. 6.6.7 Estructura secuencial de n pasos.

*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:

$$R = \sum_{i=1}^{n} r_i\,(x_i,\ d_i) \qquad (6.6.8)$$

*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 $\tilde{x}_2$, se tiene:

$$x_1 = \tilde{x}_2$$

$$r_1 = r_1\,(\tilde{x}_2,\ d_1)$$

pero

$$\tilde{x}_2 = T_2\,(x_2,\ d_2)$$

sustituyendo esta relación en la anterior

$$r_1 = r_1\,(T_2\,(x_2,\ d_2,\ d_1) \qquad (6.6.10)$$

y como

$$x_2 = \tilde{x}_3$$

y

$$\tilde{x}_3 = T_3\,(x_3,\ d_3)$$

se tiene al sustituir en (6.6.10)

$$\tilde{r}_2 = r_1\,(T_2\,(T_3\,(x_3,\ d_3),\ d_2),\ d_1)$$

Siguiendo con esta sustitución se obtiene:

$$r_1 = r_1\,[T_2\,(T_3\,[T_4 \ldots \{T_n\,(x_n,\ d_n),\ d_{n-1}\}, \ldots]\ d_2)\ d_1] \qquad 6.6.11)$$

*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 esta 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

$$\max_{d_1, d_2} \{(r_1, d_1) + r_2(x_2, d_2)\}$$

*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.

$$f_n(x_n) = \max_{d_n} \{r_n(x_n, d_n) + f_{n-1}(T_n(x_n) d_n))\} \qquad (6.6.15)$$

*Toda esta deducción puede por lo tanto resumirse en las siguientes ecuaciones de recursión para el problema de programación dinámica:

*Fórmula de recursión.

$$f_i(x_i) = \max_{d_i} Q_i(x_i, d_i) \qquad i = 1, 2, \ldots n$$

$$Q_i(x_i, d_i) = r_i(x_i, d_i) \qquad i = 1 \qquad (6.6.16)$$

$$Q_i(x_i, d_i) = r_i(x_i, d_i) + f_{i-1}(T_i(x_i, d_i)$$

$$i = 2, 3, \ldots n$$

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.

*Maximización del beneficio.

*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.

*Tres unidades industriales.

*x_1 nivel de inversión en industria i'sima y $g_i(x_i)$ su beneficio.

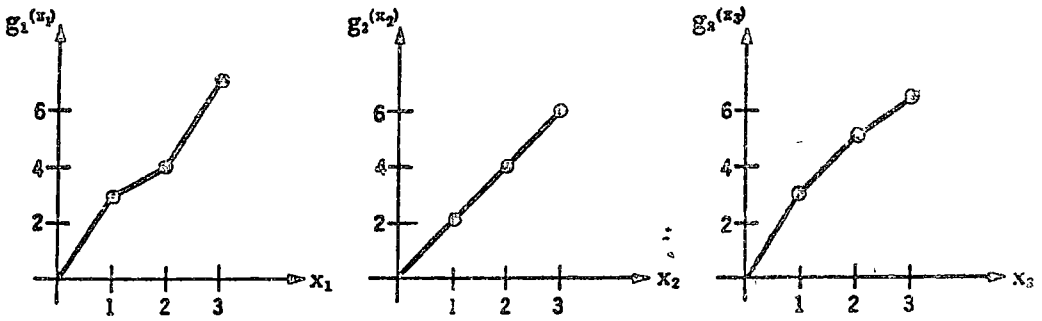


Fig. 6.6.8 Funciones de beneficio del ejemplo 6.6.1.

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.

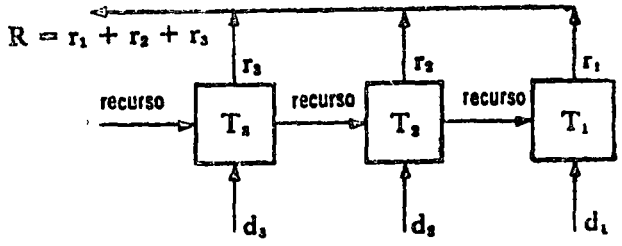


Fig. 6.6.9 Diagrama de bloque del ejemplo 6.6.1.

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

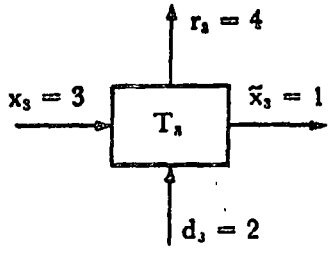
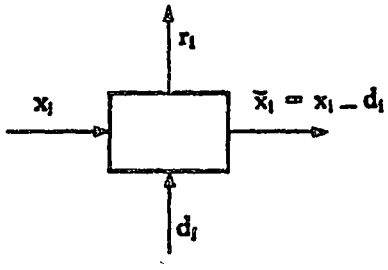


Fig. 6.6.10 Ejemplo de asignación de recursos al proyecto 3.

En este ejemplo, la transformación tiene esta forma simple $\tilde{x}_1 = x_1 - d_1$ 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

$$x_1 - d_1 \geq 0 \quad \text{ó} \quad x_1 \geq d_1$$

es decir, no se puede gastar en un proyecto más de los recursos disponibles.

*La función de beneficio $r_1(x_1, d_1)$ en este caso solamente depende de la decisión que se tome, es decir:

*Función de beneficio

$$r_1(x_1, d_1) = g_1(d_1)$$

*La fórmula de recursión para la solución del problema es

*Fórmula de recursión

$$Q_i(x_i, d_i) = r_i(x_i, d_i) + f_{i-1}(T_i(x_i, d_i)) \quad (6.6.17)$$

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

$$Q_i(x_i, d_i) = g_i(d_i) + f_{i-1}(x_i - d_i) \quad (6.6.18)$$

*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 $$f_3(x_3) = \max_{d_3} \{g_3(d_3) + f_2(x_3) - d_3\}$$

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^*	Val. de var. de decs. que prod. el ópt. 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.

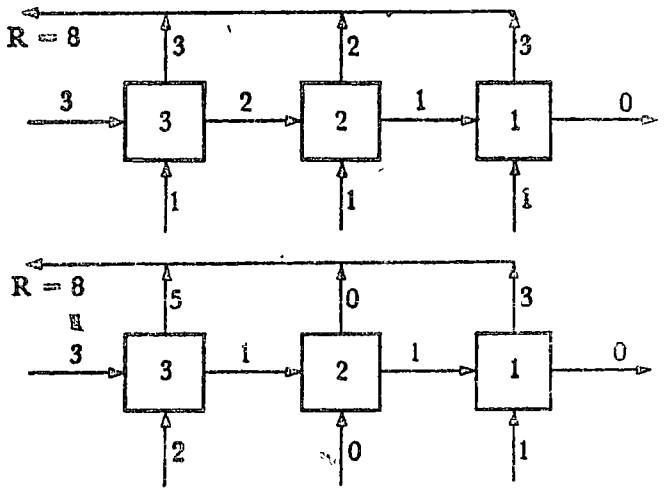


Fig. 6.6.11 Asignación óptima de recursos al proyecto del ejemplo 6.6.1

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		

*El principio de optimalidad reduce el número de alternativas a explorar.

*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.

Empezando asignando recursos al proyecto 1, se tienen las alter-

nativas 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

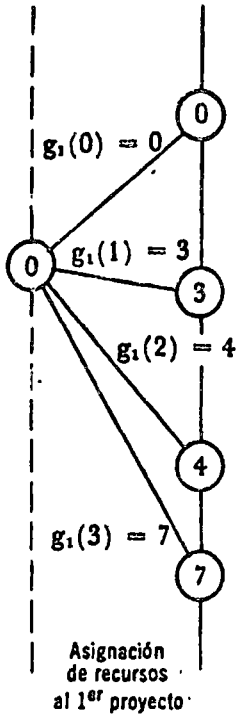


Fig. 6.6.12 Arbol de combinaciones para la asignación de 3 unidades al 1er. proyecto del ejemplo 6.6.1.

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

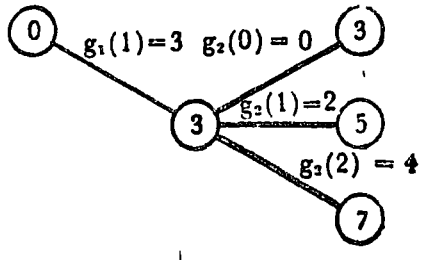


Fig. 6.6.13 Arbol con algunas posibles asignaciones de recursos al 2do. proyecto.

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.

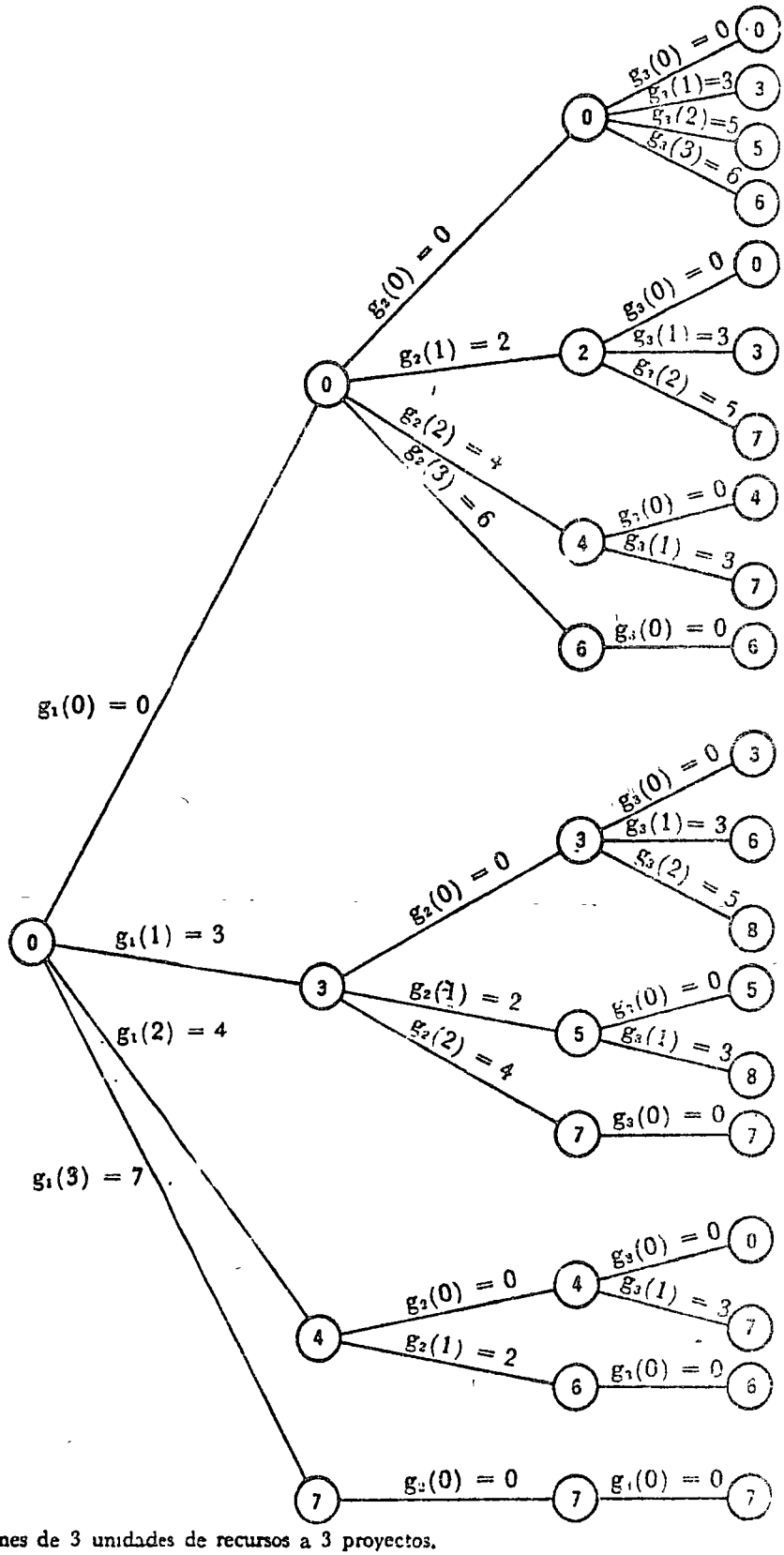


Fig. 6.6.14 Arbol de todas las posibles combinaciones de 3 unidades de recursos a 3 proyectos.

Este árbol muestra de inmediato las dos estrategias óptimas que aparecen en la figura 6.6.15

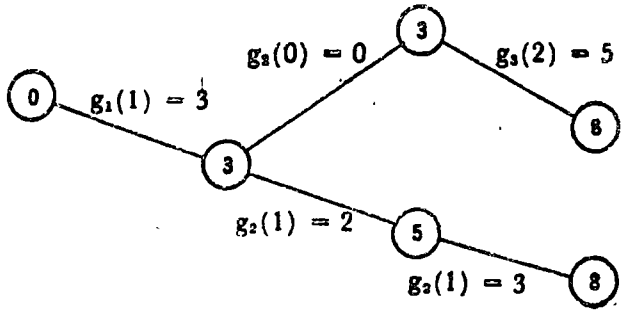


Fig. 6.6.15 Asignación óptima de recursos al proyecto del ejemplo 6.6.1.

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 anteripres. 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.*

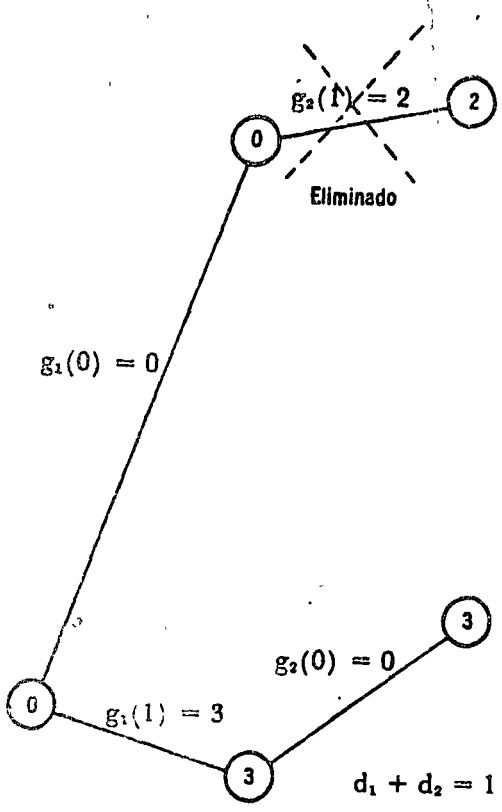


Fig. 6.6.16 *a* Asignación de una unidad de recurso en 2 etapas.

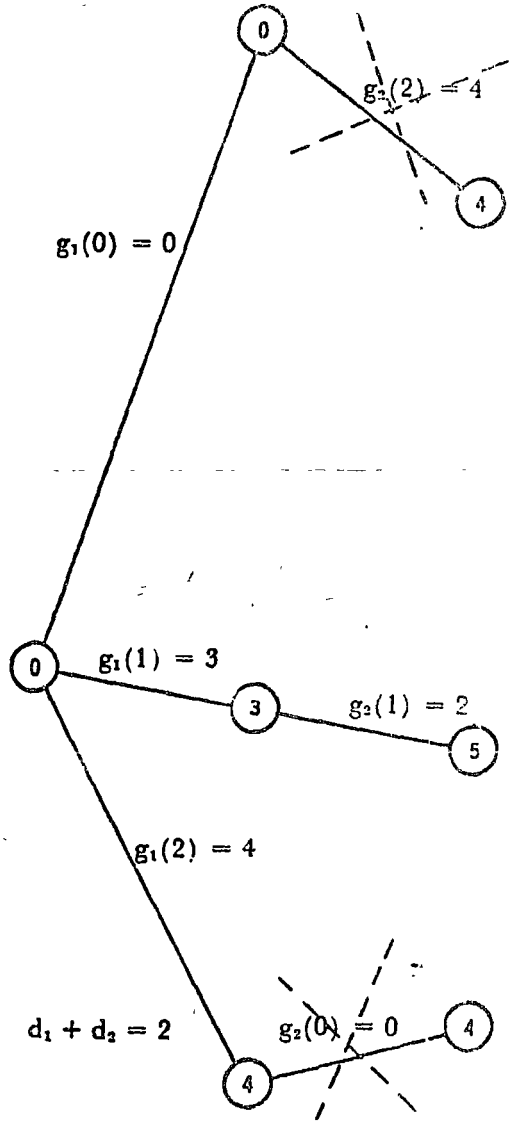


Fig. 6.6.16 *b* Asignación de 2 unidades de recurso en dos etapas.

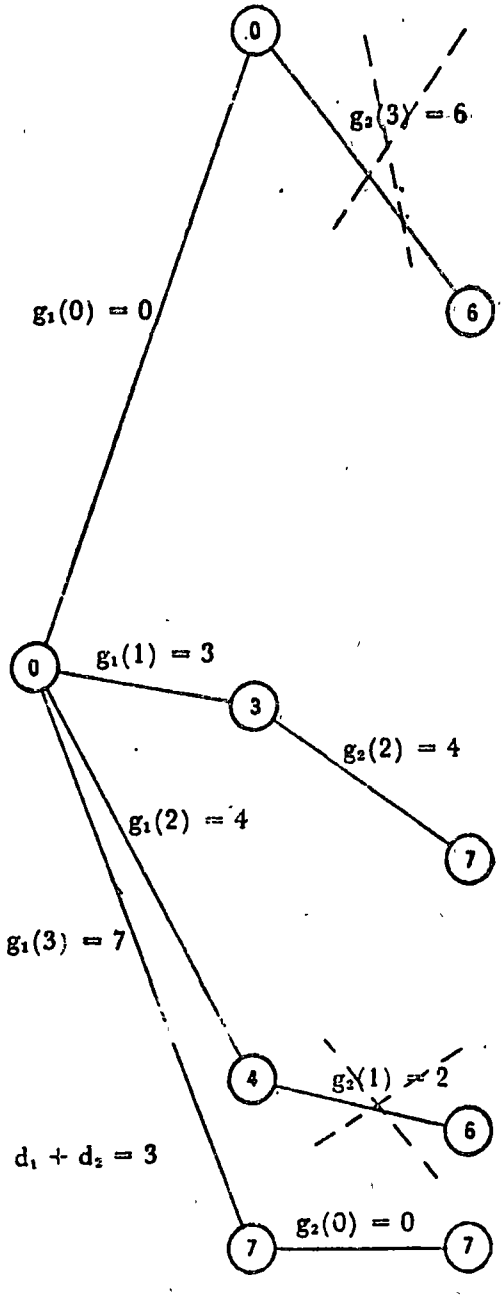


Fig. 6.6.16 *c* **Asignación de 3 unidades de recurso en 3 etapas.**

La eliminación de estas alternativas reduce la búsqueda a los casos que muestra el árbol de la figura 6.6.17 con trazo grueso.

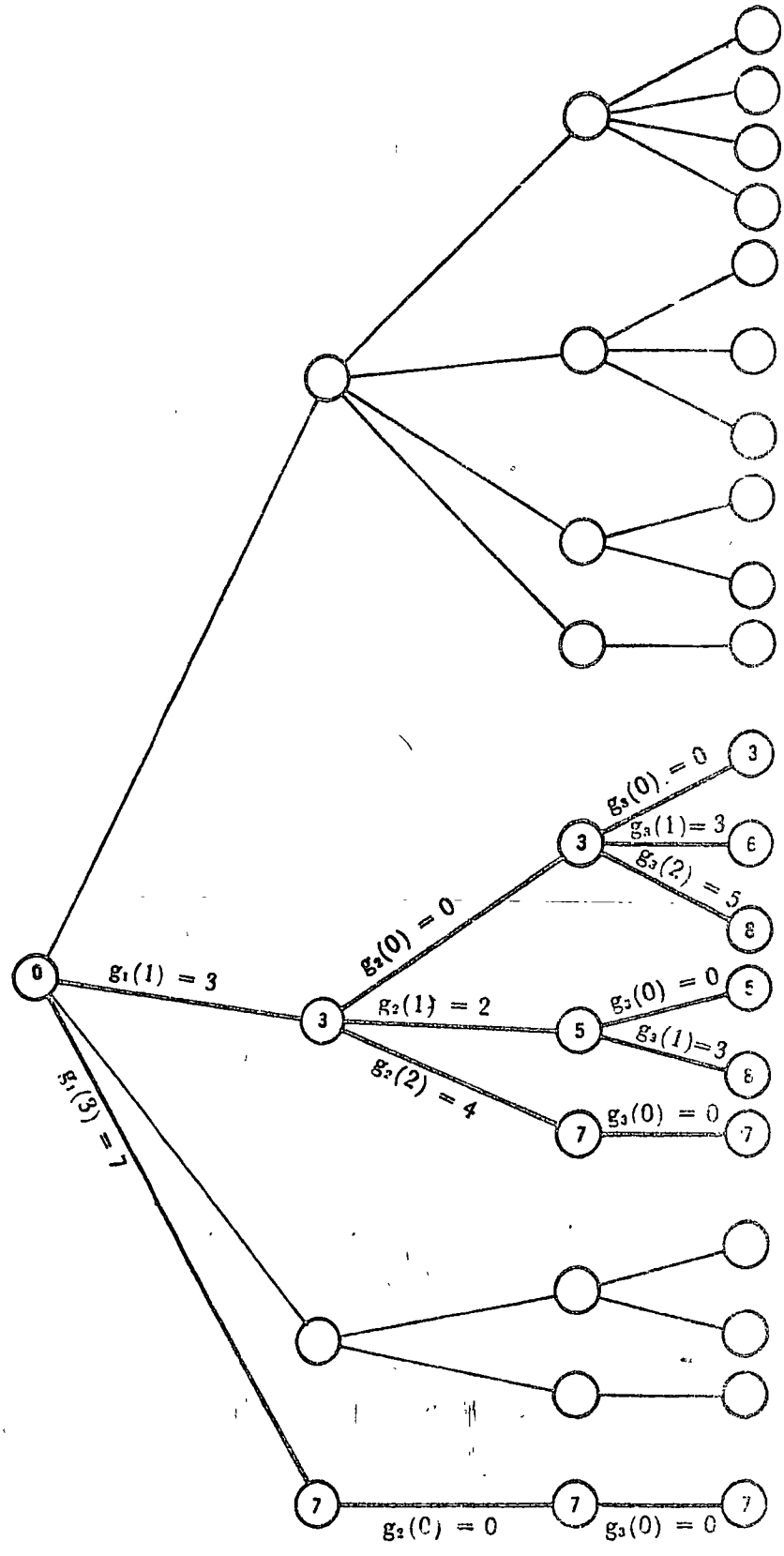


Fig 5.5.17 Reducción de alternativas a explorar.

*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

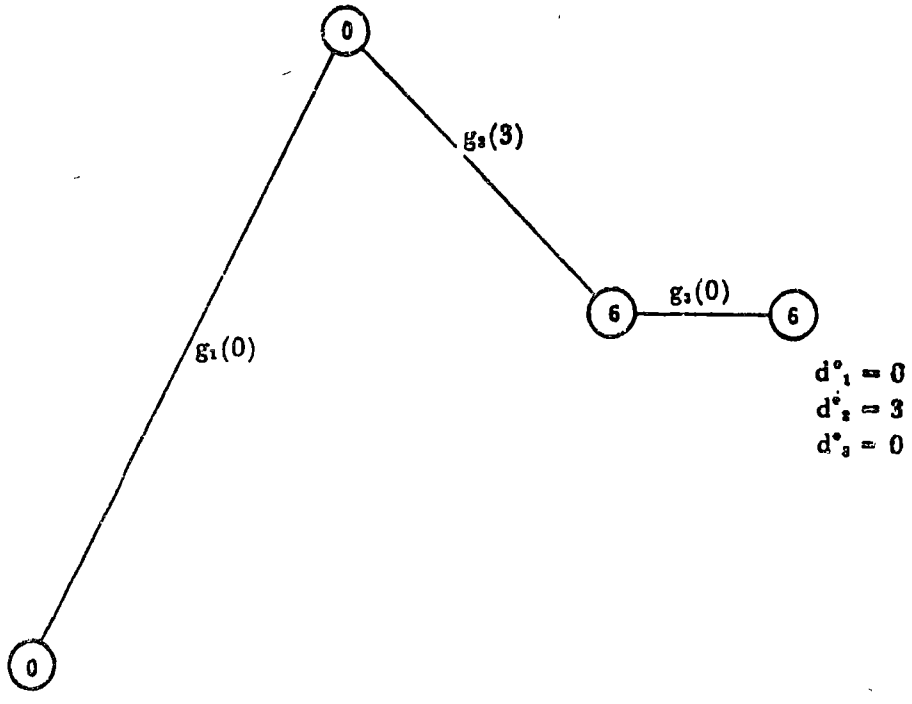


Fig. 6.6.18 Secuencia de decisiones que llevan a un beneficio determinado.

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.

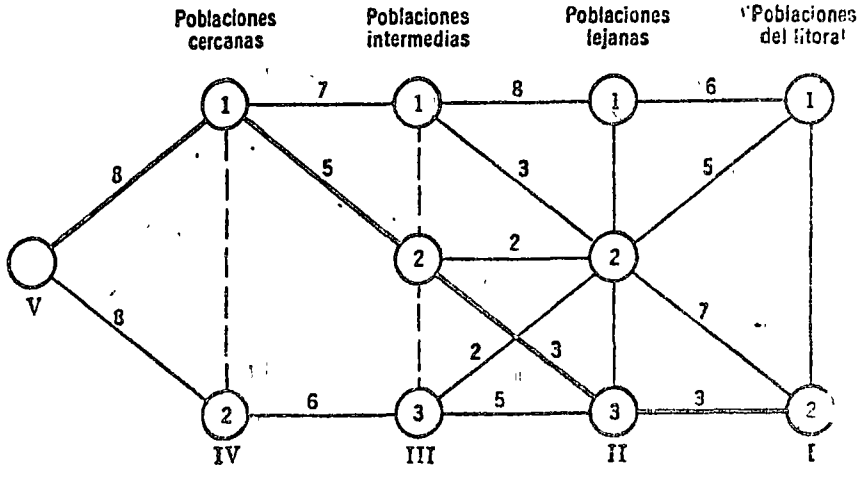
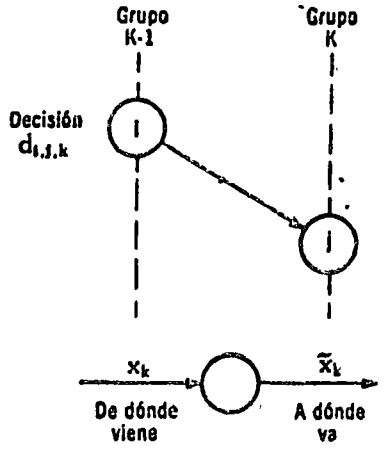


Fig. 6.6.19 Red de caminos entre la localidad V y puertos de un litoral.

*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 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
II_1	1 1 1	6	1	6	1 1 1
II_2	2 1 1 2 2 1	5 7	1 2	5	2 1 1
II_3	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}°
III_1	1 1 2	8	1	1	9	8	2 1 1	1 2 2
	1 2 2	3	2	5	8			
III_2	2 2 2	2	2	5	7	6	3 2 1	2 3 2
	2 3 2	3	3	3	6			
III_3	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_2	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^*	d_{II}^*	d_{III}^*
IV_1	1 1 3	7	1	8	15	11	321	232	123
	1 2 3	5	2	6	11				
IV_2	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^*	d_{II}^*	d_{III}^*	d_{IV}^*
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:

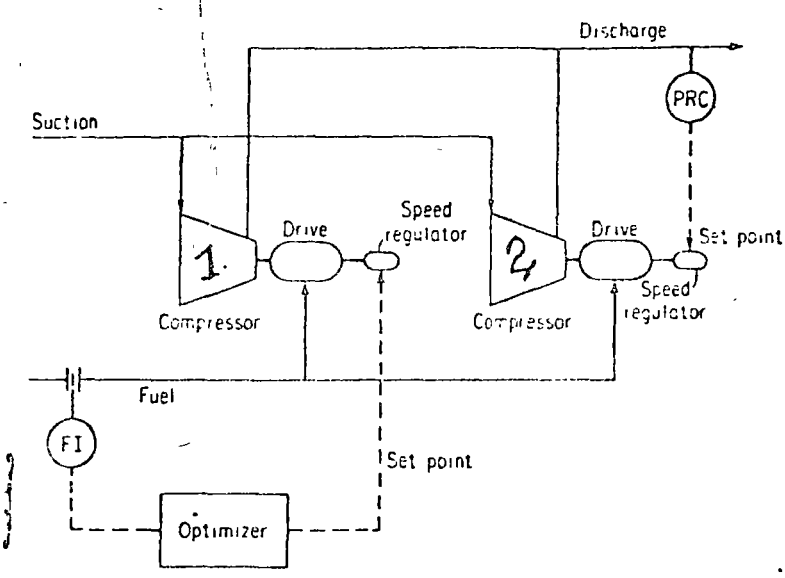
Objetivo. Regular la velocidad de los accionamientos para mantener la presión de descarga, gastando mínimo combustible

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



Estrategia:

1- Estimar la velocidad requerida según el requerimiento

2- Cambie la velocidad del compresor 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 alcanzado se repite el procedimiento en (2)

dirección.

Es un problema de búsqueda unidireccional.

La técnica 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 ura adecuada solución a los problemas que se presentan en su operación, es preciso acudir a la metodología más avan-

zada 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 y 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.

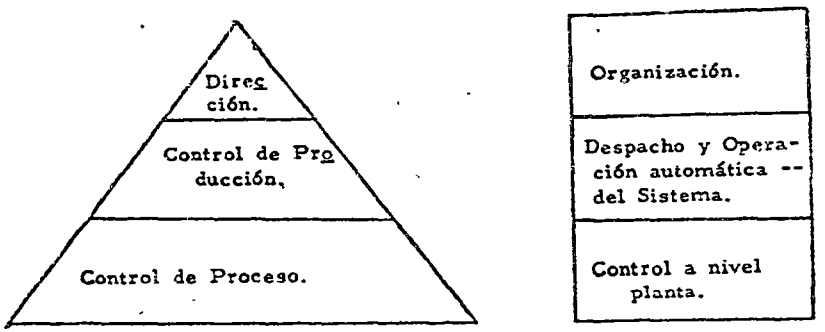


Fig. 1. - Estructura jerárquica del control.

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 gufan 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 ma-- yor · capacidad. Para mantener con estas unidades una adecuada confiabili - dad del servicio dentro de límites económicos, es necesario interconectar los sistemas. La interconexión presenta además beneficios adicionales derivados de otros aspectos del aprovechamiento ecònómico del sistema.

Debido al crecimiento de la demanda y a la interconexión de los diferentes subsistemas eléctricos, la complejidad del sistema va en aumen to, 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 tie nen 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 siste - ma 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.

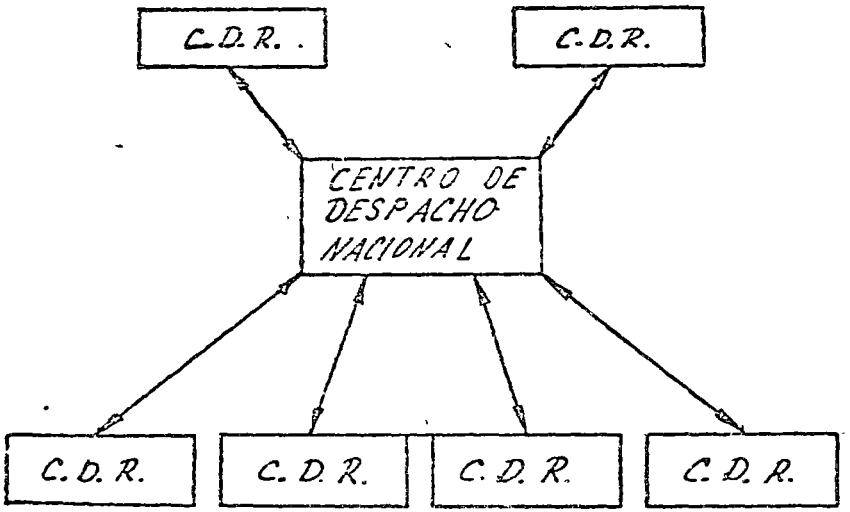
5. - ESTRUCTURA DEL SISTEMA.

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 presen ta respecto al analógico varias ventajas. Su mayor flexibilidad permite imple mentar mejores esquemas de control en tiempo real. Además, puede emplear se la computadora trabajando en tiempo compartido para realizar cálculos de apoyo a la operación del sistema.

Debido a la continua aparición de mejores técnicas de con-- trol, la flexibilidad de un sistema digital permite su implementación con cam- bios mínimos en el equipo (hardware).

El tamaño de la República y la distribución geográfica no uni forme de los centros de carga y de los recursos de generación han determina- do 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 ener gía 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.

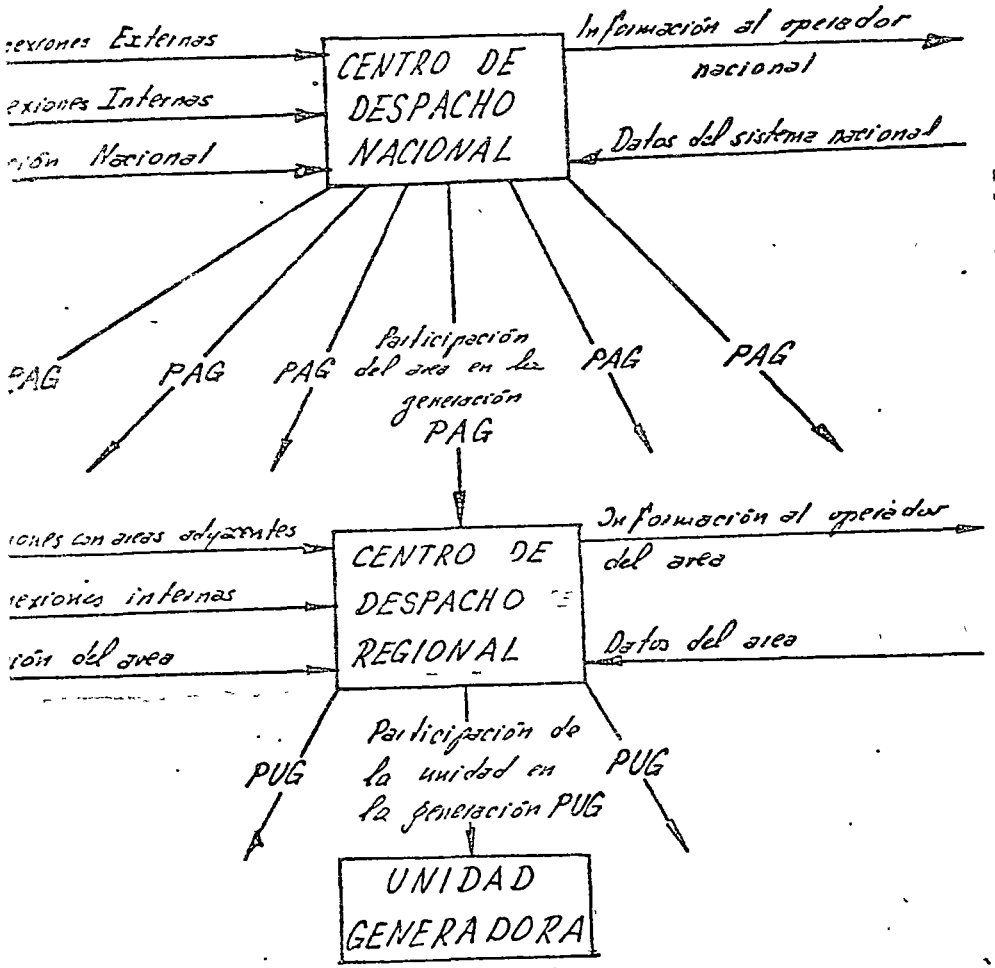


Fig 3. - Esquema del sistema de control propuesto.

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. --

##

Programa de Cómputo Crítico

	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 diagnóstico				F-L		F-L

Clave: E L Cálculo en 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 sleccionada de gran cantidad de información sobre el sistema de potencia, lo cual es imprácticо, 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 contrapartes 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.

##

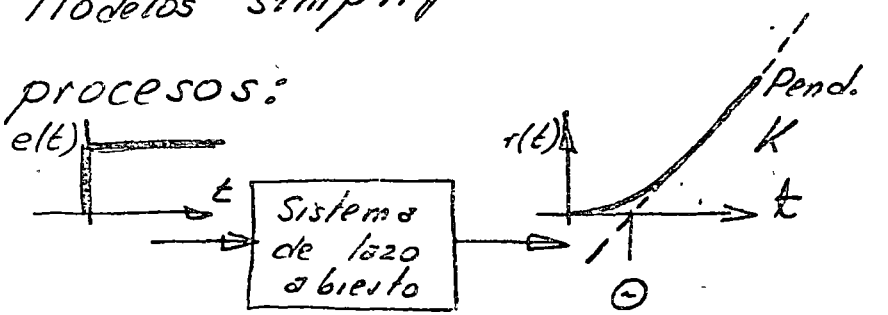
5. - Instrumentos Registradores y Diagrama Mímico. Se ha señalado en el inciso anterior la ventaja de los despliegues visuales controlados por computadora respecto a los despliegues de datos convencionales. A pesar de estas ventajas varios centros de despacho modernos cuentan con los despliegues en TRC 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. - Terminales Remotas. 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.

ALGORITMOS DE CONTROL

Diseño
- Transformada Z
- Técnicas Convencionales

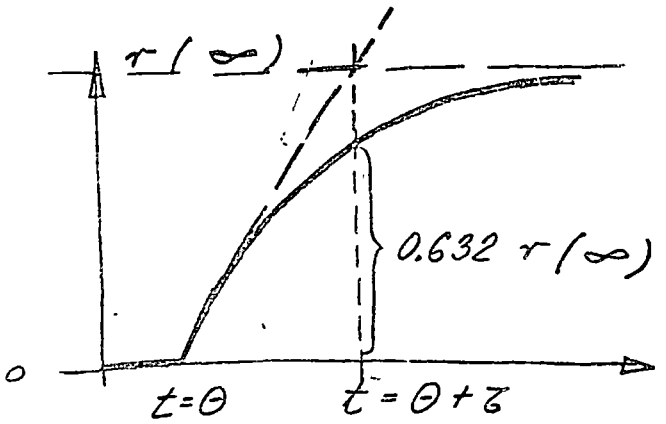
Modelos simplificados de procesos:



Función de transferencia?

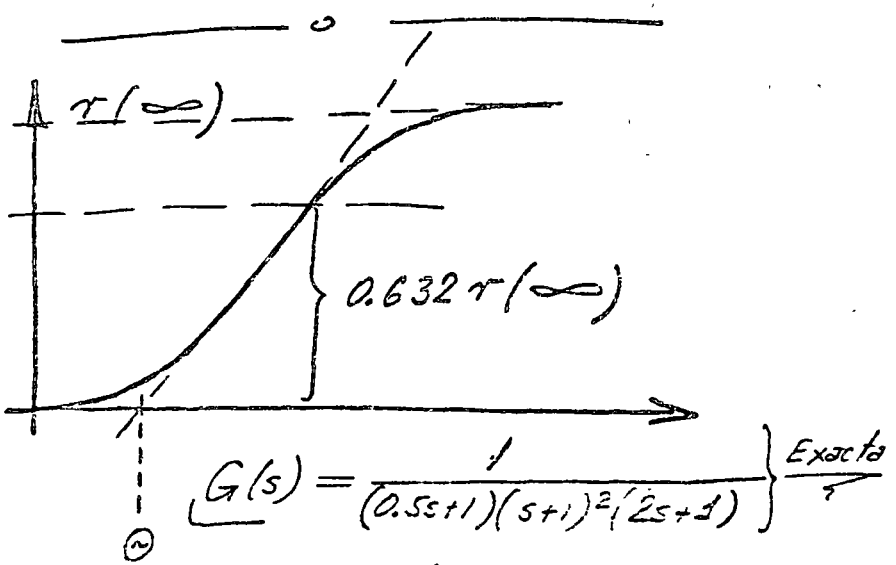
Sistema no autoregulado $r(\infty) \to \infty$

$$G(s) = \frac{Ke^{-\theta s}}{s}$$



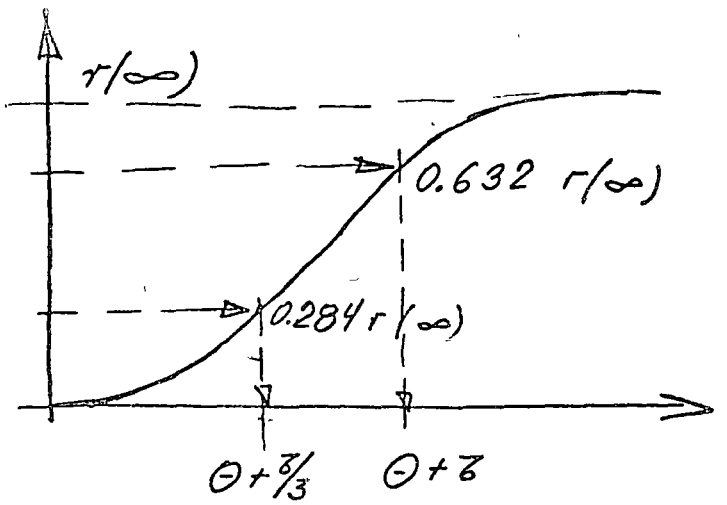
Sistema autoregulado (Primer orden con atraso + tiempo muerto)

$$G_m(s) = \frac{Ke^{-\theta s}}{\tau s + 1}$$



$$G(s) = \frac{1}{(0.5s+1)(s+1)^2(2s+1)}$$ Exacta?

Sistema autoregulado de orden superior



Ejemplo $2.88 = \Theta + \tau/3$ } 2 ecs.
$4.80 = \Theta + \tau$ } 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:

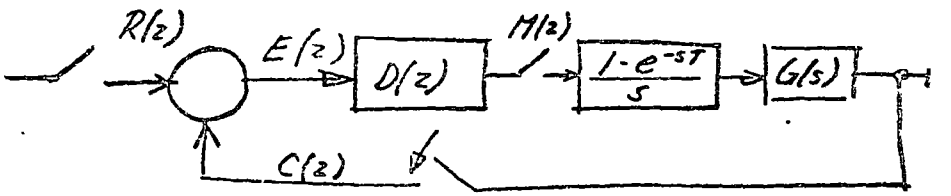
$$G(s) = \frac{Ke^{-\Theta s}}{\tau s + 1} \xrightarrow{\mathcal{L}^{-1}}$$

$$r(t) = E_m\left(1 - e^{-(t-\Theta)/\tau}\right), \quad t > L$$

$t = \Theta + \tau/3 \quad r = 0.284\, r(\infty)$

$t = \Theta + \tau \quad r = 0.632\, r(\infty)$

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 muertos, ninguna $D(z)$ puede eliminarlos, en la especificación de $\frac{C(z)}{R(z)}$ debe haber z^{-N}

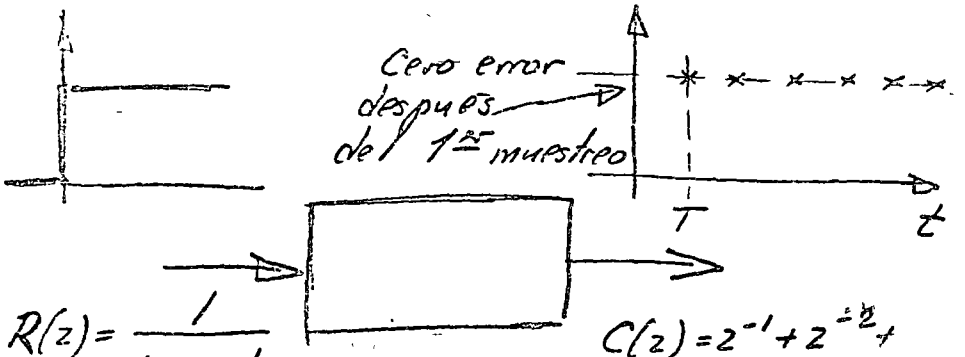
Algoritmos deadbeat:

Especificaciones:

Tiempo de asentamiento finito -

Tiempo de respuesta mínimo

Error $(\infty) = 0$

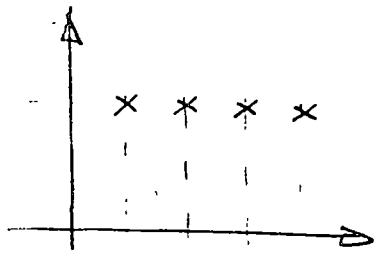


$$R(z) = \frac{1}{1 - z^{-1}}$$

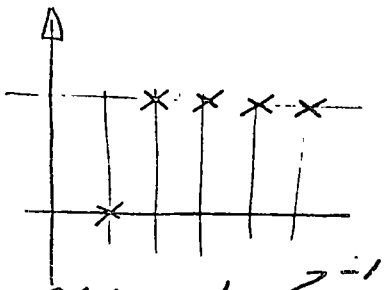
$$C(z) = z^{-1} + z^{-2} + \cdots = \frac{z^{-1}}{1 - z^{-1}}$$

Sea $HG(z) = \frac{z^{-1}(0.65 + 0.12z^{-1})}{1 - 0.22z^{-1}}$

Nota: Corresponde al sistema especificado en pág 4 $T = 5$



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



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

atraso

Método de Dahlin.

$$C(s) = \underbrace{\frac{e^{-\Theta s}}{(\lambda s + 1)}\;\frac{1}{s}}$$

respuesta buscada

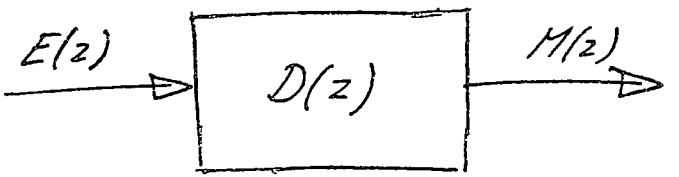
$C(T) = C(\infty)$ es muy estricto.

$$C(z) = \frac{(1 - e^{\frac{T}{\lambda}})\,z^{-N-1}}{(1 - z^{-1})(1 - e^{\frac{T}{\lambda}} z^{-1})}$$

* $\Theta = NT + \Theta' \qquad \Theta' < 1$

Sustituyendo en pg 5 *

$$D(z) = \frac{1 - 0.22 z^{-1}}{z^{-1}(0.65 + 0.12 z^{-1})}\;\frac{z^{-1}}{1 - z^{-1}}$$



$$D(z) = \frac{M(z)}{E(z)} =$$

Solución:

$$m_n = \frac{e_n - 0.22\,e_{n-1} + 0.53\,m_{n-1} + 0.12\,m_{n-2}}{0.65}$$

Observación: Si $T = 1$ y como $\Theta =$

$C(T) \neq C(\infty)$

Especifique: $C(0) = C(T) = 1$

$C(nT) = C(\infty)$.

$$R(z) = \frac{1}{1-z^{-1}} \quad \text{(escalón)}$$

$$\frac{C(z)}{R(s)} = \frac{1 - e^{\frac{T}{\lambda}} z^{-N-1}}{1 - e^{\frac{T}{\lambda}} z^{-1}}$$

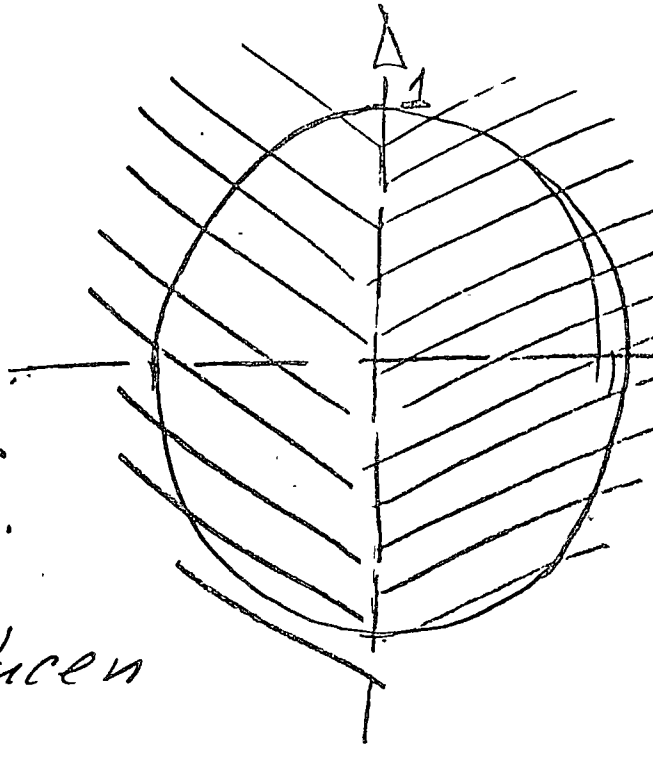
↓

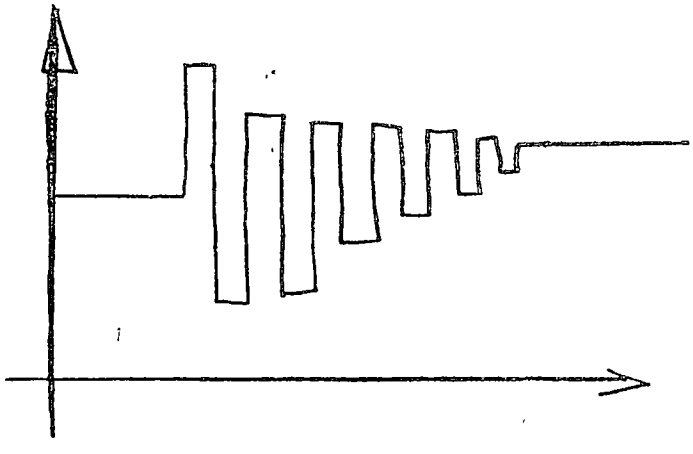
igual que en el caso anterior.

Trace la respuesta y observe que λ es "mejor"

CAMPANEO

La localización de polos y ceros de una función determina este fenómeno.-

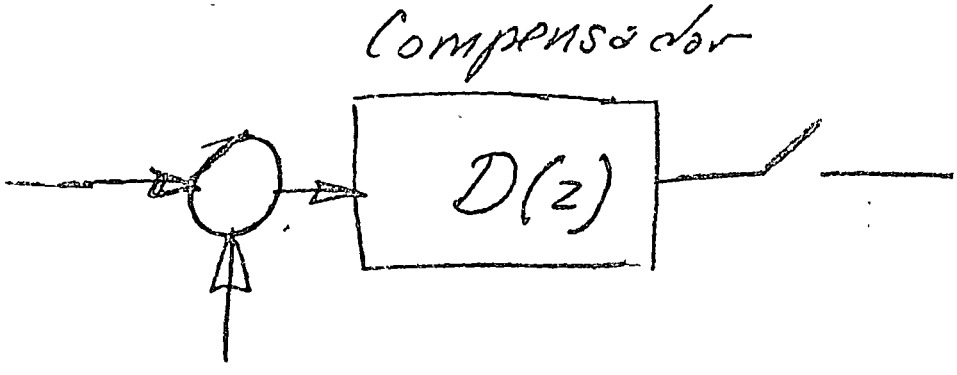




Ejemplo de campaneo.

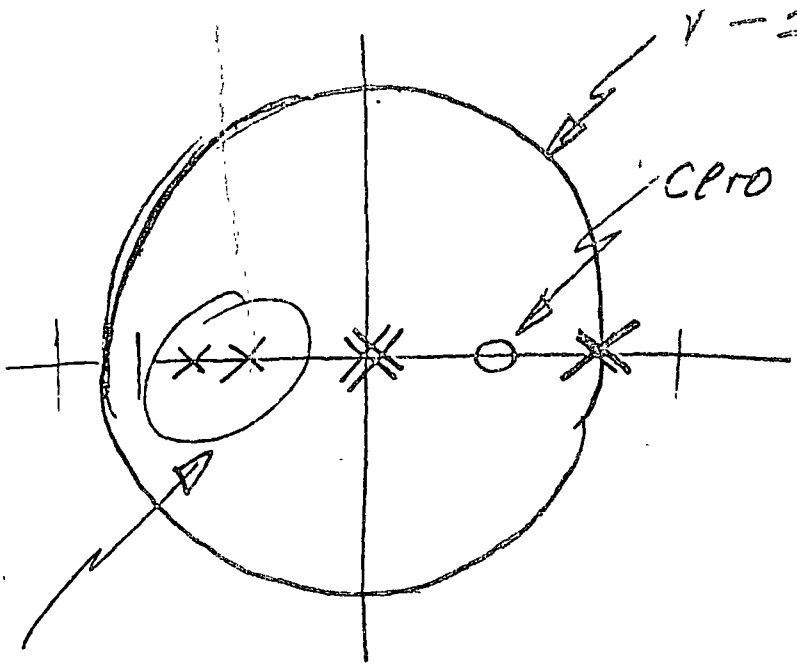
Observación. Puede no existir en la señal de salida, pero sí en alguna intermedia.

Ejemplo.



$$D(z) = \frac{1.96z^2(z-0.741)}{(z-1)(z+0.392)(z+0.738)}$$

Diseñado por el método de Dahlin (Sección anterior)



Remedio:

Probe eliminando el <u>mas negativo</u> ajustando la ganancia

Ganancia ($t \to \infty$) correspondiente a cada termino se obtiene sustituyendo $z = 1$

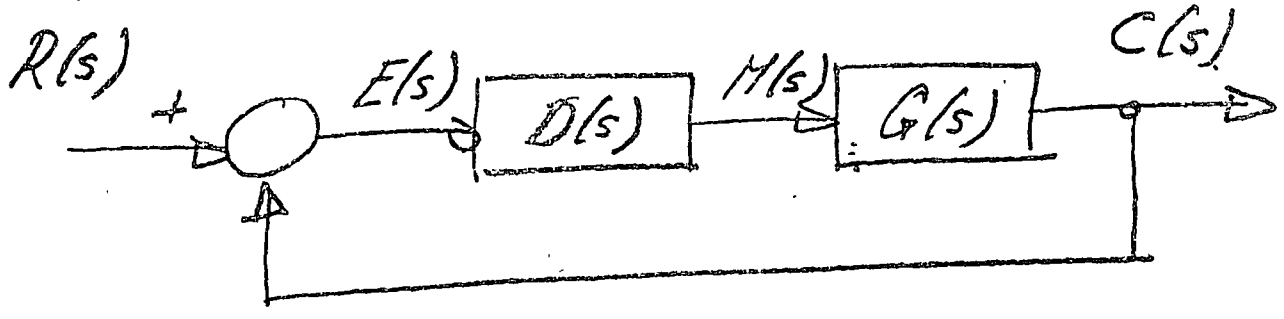
$$D(z) = \frac{1.96\, z^2 (1 - 0.741 z^{-1})}{(1 - z^{-1})(1 + 0.392 z^{-1})(1 + 0.738)}$$

EQUIVALENTE DISCRETO DE UN CONTROL ANALOGI-CO

Sea $G(s) = \dfrac{Ke^{-\theta s}}{\tau s + 1}$ → (1)

$$D(s) = \frac{M(s)}{E(s)} \cdot \frac{1}{G(s)} \cdot \frac{C(s)/R(s)}{1 - C(s)/R(s)} \quad (2)$$

controlador



Especificación:

$$\frac{C(s)}{R(s)} = \frac{e^{-\theta s}}{\lambda s + 1} \quad (3)$$

(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

$\theta = NT$

$$\lambda \frac{m_n - m_{n-1}}{T} + m_{n-1} - m_{n-N-1} = \left[\tau \frac{e_n - e_{n-1}}{T} + e_{n-1}\right]/K \quad *$$

↓

despejar: m_n

Nota: Solo emplear primeras diferencias para aproximar $\frac{d}{dt}$

si $T \leq 0.2\tau$ $\lambda \gg 2.5T$

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

CONTROL DIGITAL DE PROCESOS

CONTROL ADAPTIVO

Y

CONTROL OPTIMO

M. en C. Rafael López

Octubre 28, 1977

PALACIO DE MINERIA
Tacuba 5, primer piso. México 1, D. F.

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 sistemas 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)

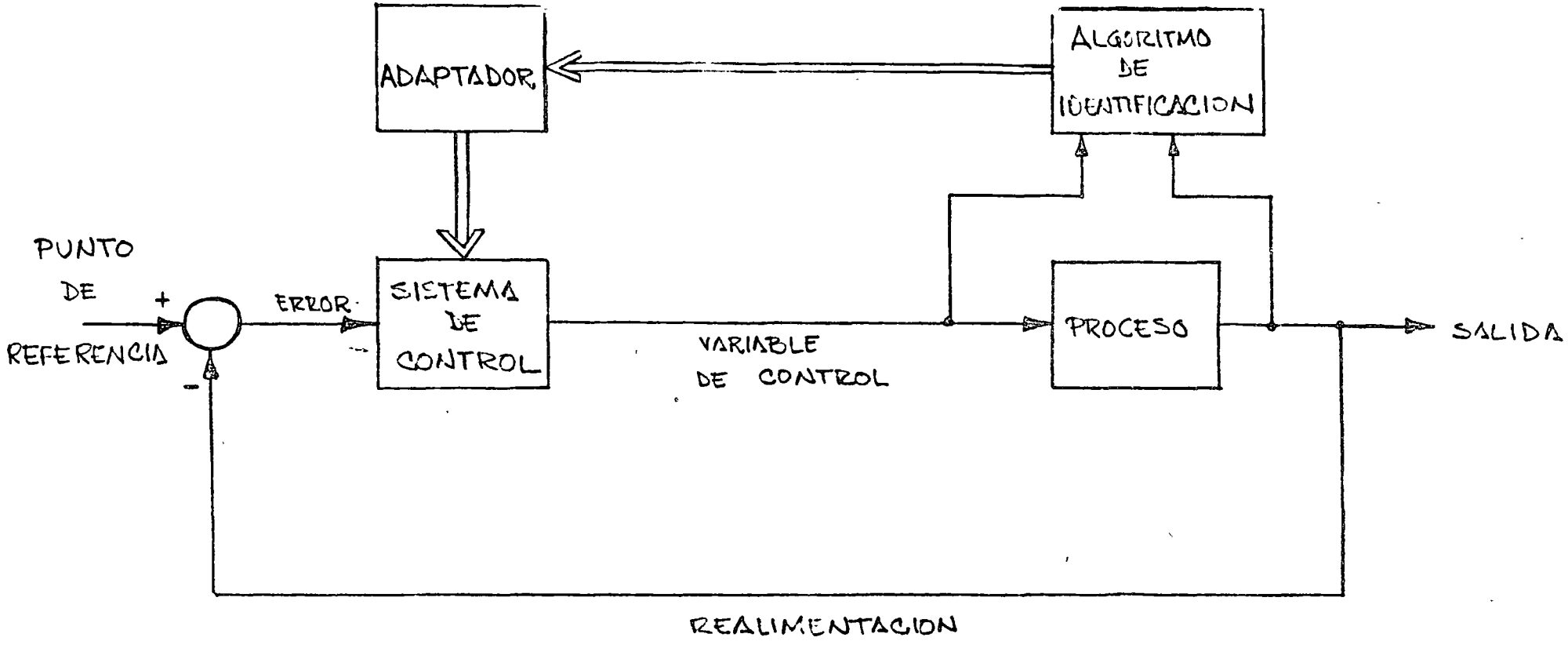
En ambos casos, el resultado neto es que los valores de los parámetros del modelo que se está utilizando para describir la planta cambian continuamente de valor. Es por ello deseable implantar algoritmos de control que tomen en cuenta este hecho y produzcan un comportamiento satisfactorio del sistema 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.

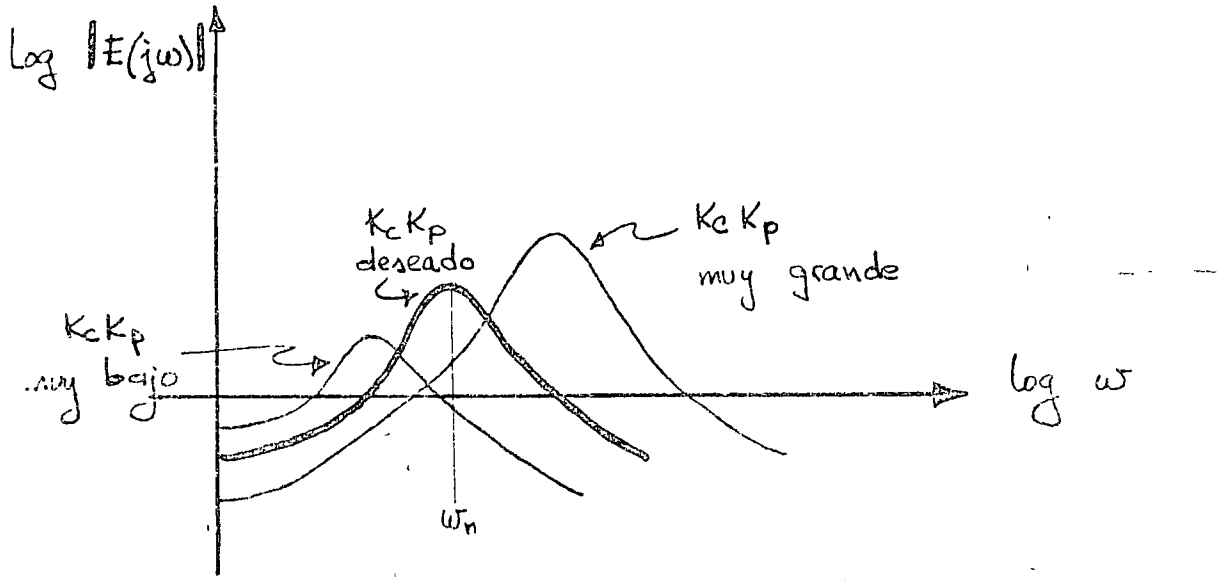
AJUSTE ADAPTIVO DE LA GANANCIA.

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,

$$E(s) = \frac{1}{1 + K_c K_g G_c(s) G_p(s)} R(s)$$

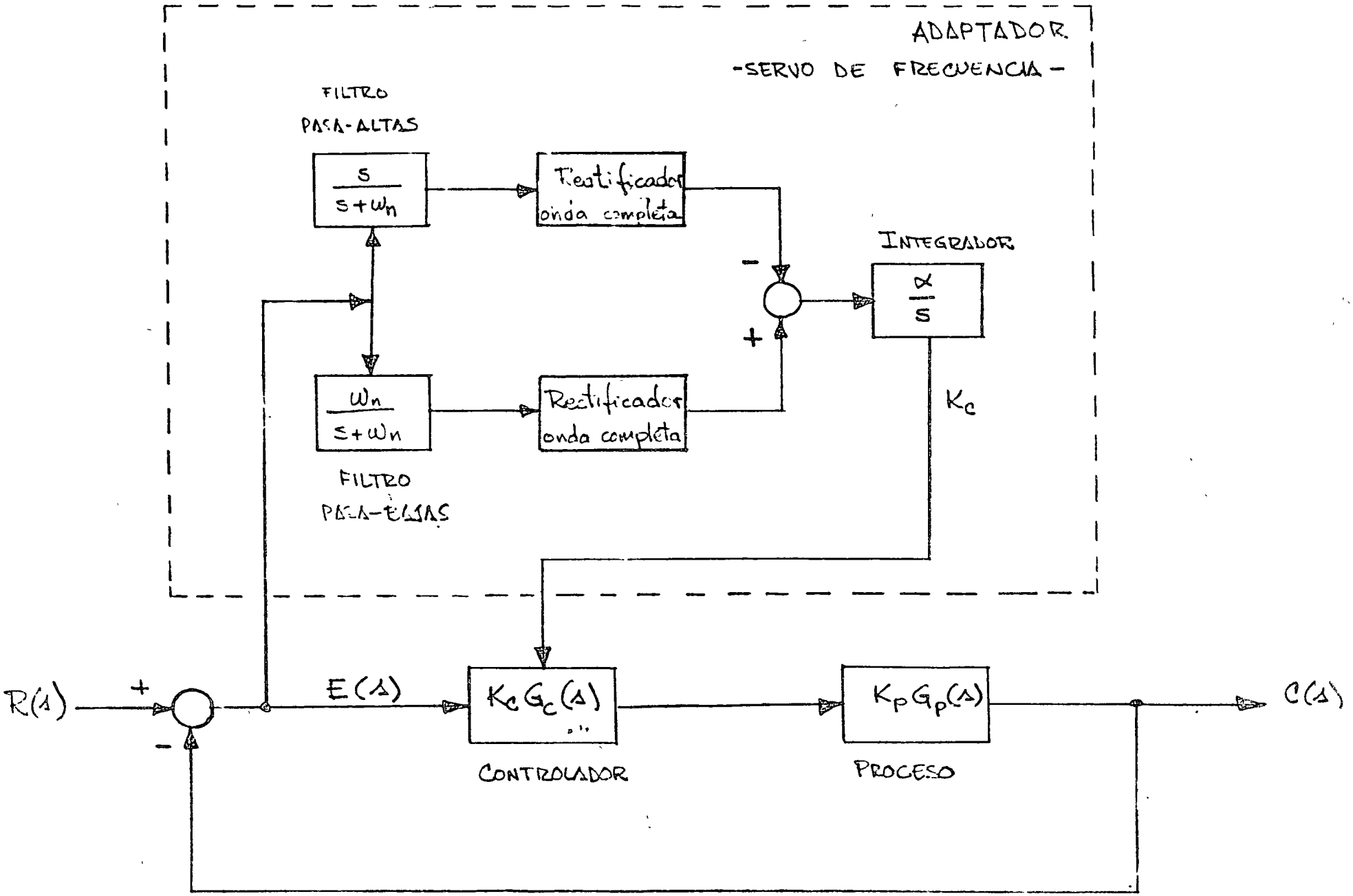
Haciendo $R(s) = \frac{1}{s}$ (entrada escalón) es posible obtener la traza de Bode de $E(j\omega)$ para distintos valores del producto $K_c K_g$,

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 servo mecanismo 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

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)

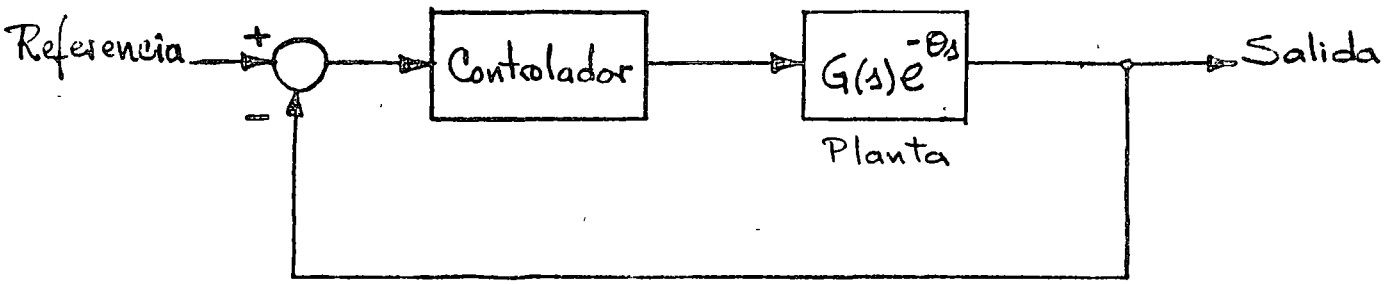
3- Para obtener la ganancia del controlador utilizar, en vez del integrador, una forma más general (PD, PI, PDI)

COMPENSADOR DE RETRASO (TIEMPO MUERTO)

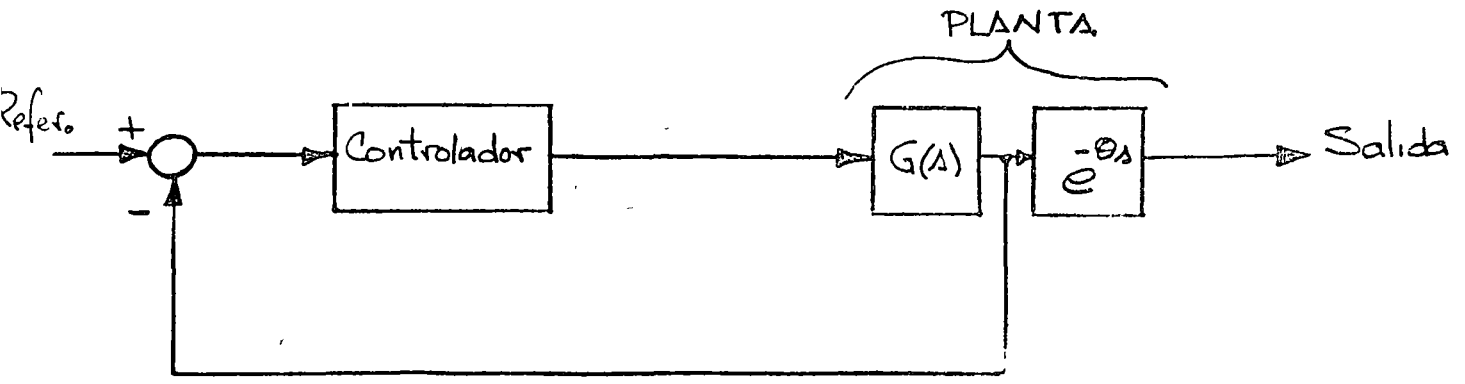
Cuando el proceso que se desea controlar tiene un retraso apreciable

$$G_p(s) = G(s)\, e^{-\theta s}$$

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 $t_1 + \theta$.



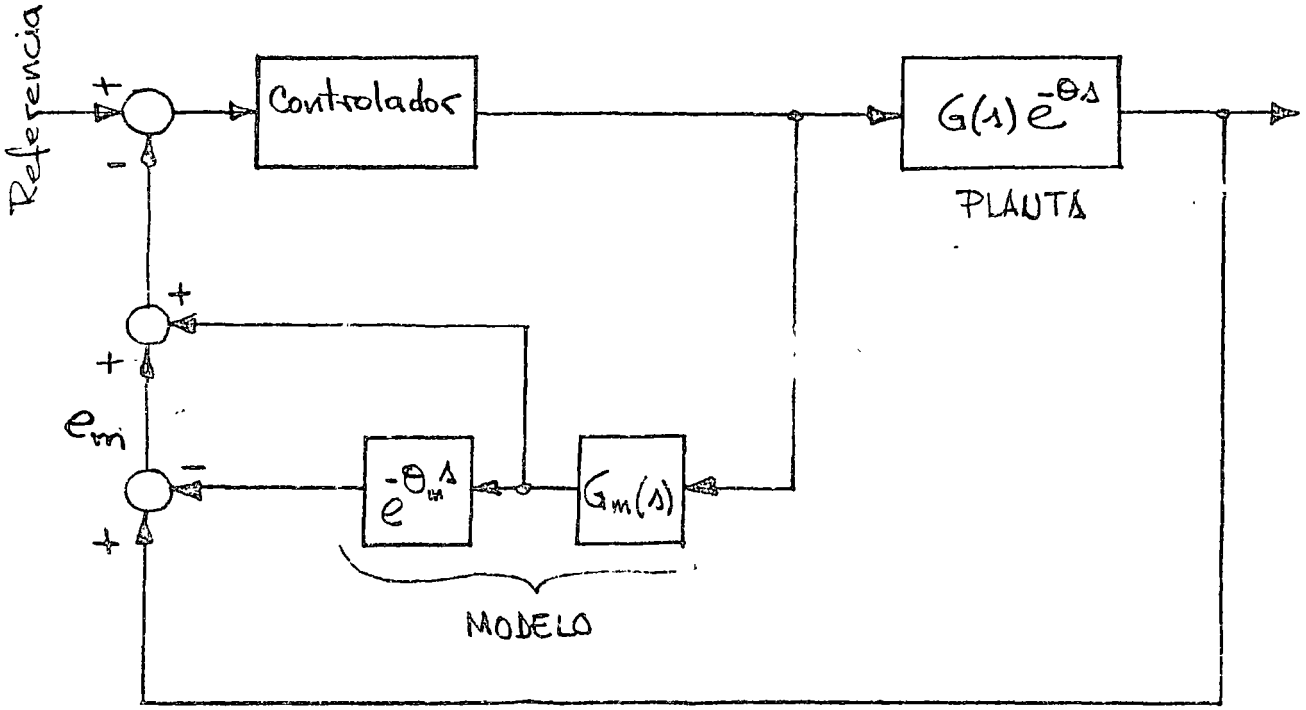
Sería desable tomar la señal "antes" que se retrasara y realimentarla



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

debería 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.

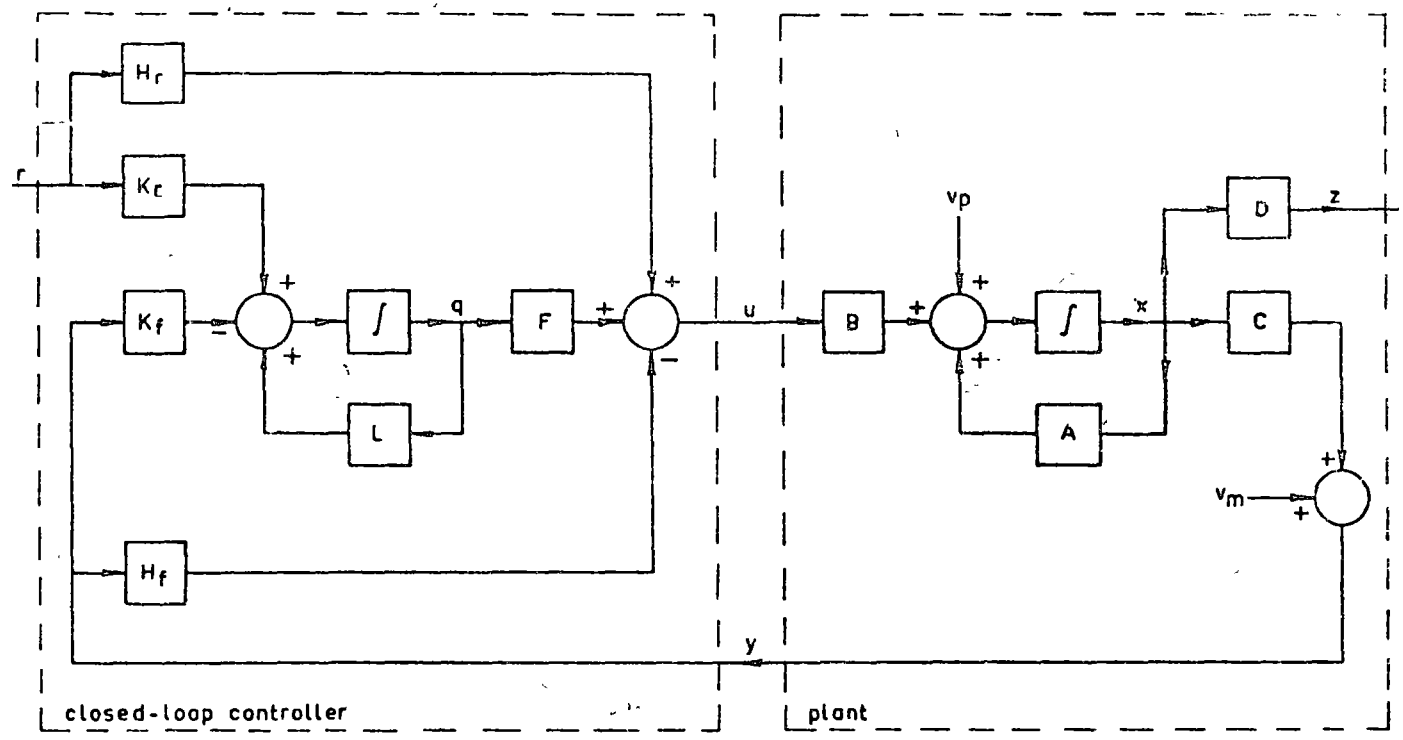


Fig. 2.7. A closed-loop control system.

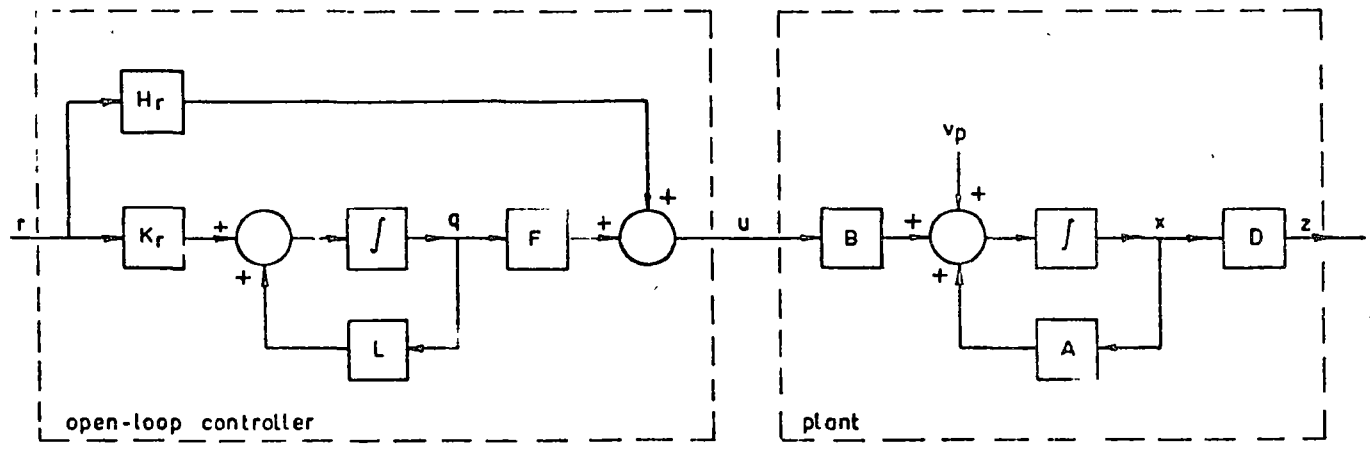


Fig. 2.8. An open-loop control system.

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:

$$\dot{\underline{x}}(t) = \underline{f}(\underline{x}(t), \underline{u}(t), t)$$

$\underline{x}(t)$ 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 $\underline{x}(t_o)$ y las entradas para tiempos posteriores, $\underline{u}(t)$, $t \geq t_o$, 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 $\underline{y}(t)$ 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

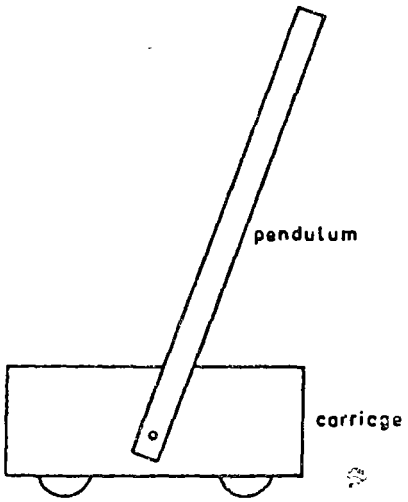


Fig. 1.1. An inverted pendulum positioning system.

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

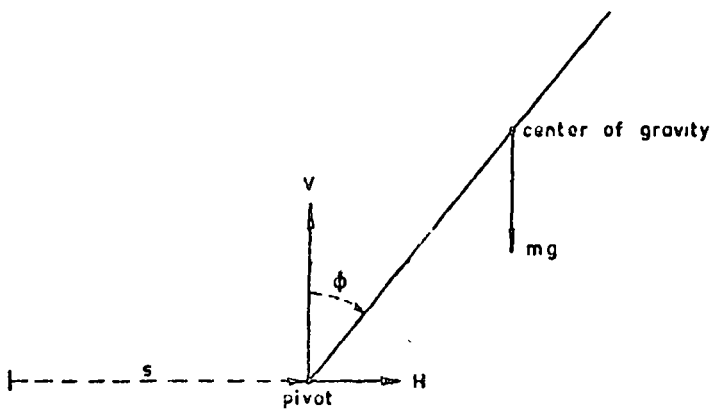


Fig. 1.2. Inverted pendulum: forces and displacements.

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:

$$\ddot{\phi}(t) - \frac{g}{L'}\phi(t) + \frac{1}{L'}\ddot{s}(t) = 0. \qquad \text{1-23}$$

We choose the components of the state $x(t)$ as

$$\begin{aligned} \xi_1(t) &= s(t), \\ \xi_2(t) &= \dot{s}(t), \\ \xi_3(t) &= s(t) + L'\phi(t), \\ \xi_4(t) &= \dot{s}(t) + L'\dot{\phi}(t). \end{aligned} \qquad \text{1-24}$$

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

$$\begin{aligned} \dot{\xi}_1(t) &= \xi_2(t), \\ \dot{\xi}_2(t) &= \frac{1}{M}\mu(t) - \frac{F}{M}\xi_2(t), \\ \dot{\xi}_3(t) &= \xi_4(t), \\ \dot{\xi}_4(t) &= g\phi(t) = \frac{g}{L'}[\xi_3(t) - \xi_1(t)]. \end{aligned} \qquad \text{1-25}$$

In vector notation we write

$$\dot{x}(t) = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{F}{M} & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -\frac{g}{L'} & 0 & \frac{g}{L'} & 0 \end{pmatrix} x(t) + \begin{pmatrix} 0 \\ \frac{1}{M} \\ 0 \\ 0 \end{pmatrix} \mu(t), \qquad \text{1-26}$$

where $x(t) = \text{col}\,[\xi_1(t), \xi_2(t), \xi_3(t), \xi_4(t)]$.

Later the following numerical values are used:

$$\begin{aligned} \frac{F}{M} &= 1\ \mathrm{s}^{-1}, \\ \frac{1}{M} &= 1\ \mathrm{kg}^{-1}, \\ \frac{g}{L'} &= 11.65\ \mathrm{s}^{-2}, \\ L' &= 0.842\ \mathrm{m}. \end{aligned} \qquad 1\text{-}27$$

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

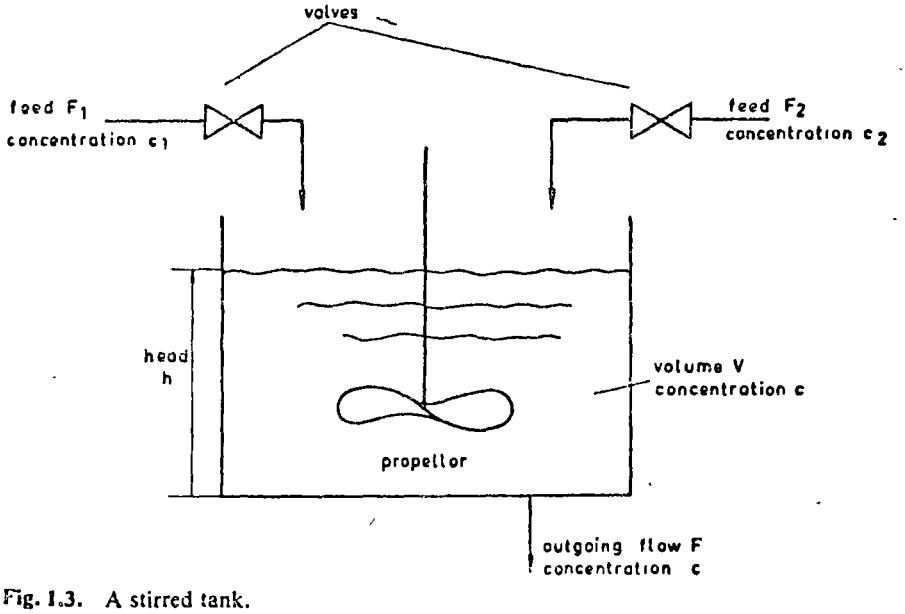


Fig. 1.3. A stirred tank.

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

$$\dot{x}(t) = \begin{pmatrix} -\frac{1}{2\theta} & 0 \\ 0 & -\frac{1}{\theta} \end{pmatrix} x(t) + \begin{pmatrix} 1 & 1 \\ \frac{c_1 - c_0}{V_0} & \frac{c_2 - c_0}{V_0} \end{pmatrix} u(t), \tag{1-43}$$

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

$$y(t) = \begin{pmatrix} \frac{1}{2\theta} & 0 \\ 0 & 1 \end{pmatrix} x(t), \tag{1-45}$$

where $y(t) = \text{col}\,[\eta_1(t), \eta_2(t)]$. We use the following numerical values:

$$\begin{aligned} F_{10} &= 0.015\ \text{m}^3/\text{s}, \\ F_{20} &= 0.005\ \text{m}^3/\text{s}, \\ F_0 &= 0.02\ \text{m}^3/\text{s}, \\ c_1 &= 1\ \text{kmol}/\text{m}^3, \\ c_2 &= 2\ \text{kmol}/\text{m}^3, \\ c_0 &= 1.25\ \text{kmol}/\text{m}^3, \\ V_0 &= 1\ \text{m}^3, \\ \theta &= 50\ \text{s}. \end{aligned} \tag{1-46}$$

This results in the linearized system equations

$$\begin{aligned} \dot{x}(t) &= \begin{pmatrix} -0.01 & 0 \\ 0 & -0.02 \end{pmatrix} x(t) + \begin{pmatrix} 1 & 1 \\ -0.25 & 0.75 \end{pmatrix} u(t), \\ y(t) &= \begin{pmatrix} 0.01 & 0 \\ 0 & 1 \end{pmatrix} x(t). \end{aligned} \tag{1-47}$$

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

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t). \end{aligned} \tag{1-48}$$

Let us define a transformed state variable

$$x'(t) = Tx(t), \tag{1-49}$$

where T is a constant, nonsingular transformation matrix. Substitution of $x(t) = T^{-1}x'(t)$ into 1-48 yields

$$\begin{aligned} T^{-1}\dot{x}'(t) &= AT^{-1}x'(t) + Bu(t), \\ y(t) &= CT^{-1}x'(t), \end{aligned} \tag{1-50}$$

or

$$\begin{aligned} \dot{x}'(t) &= TAT^{-1}x'(t) + TBu(t), \\ y(t) &= CT^{-1}x'(t). \end{aligned} \tag{1-51}$$

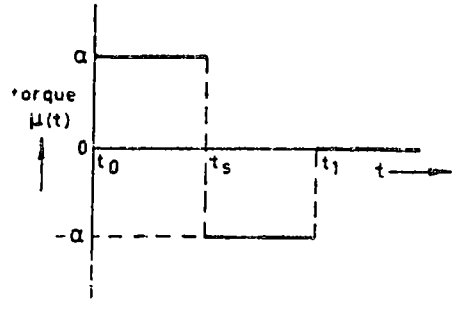


Fig. 1.12. Input torque for satellite repositioning.

(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 $x(t_0) = \text{col}\ (\phi_0, 0)$ to the state $x(t_1) = \text{col}\ (\phi_1, 0)$, 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.

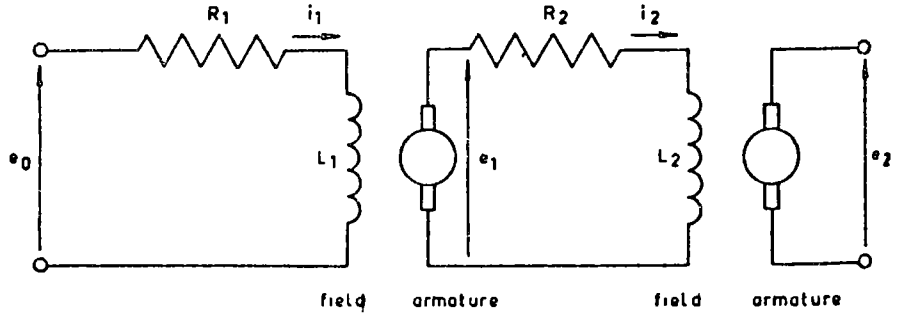


Fig. 1.13. Schematic representation of an amplidyne.

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:

$$\begin{aligned} R_1/L_1 &= 10\ \text{s}^{-1}, & R_2/L_2 &= 1\ \text{s}^{-1}, \\ R_1 &= 5\ \Omega, & R_2 &= 10\ \Omega, \\ k_1 &= 20\ \text{V/A}, & k_2 &= 50\ \text{V/A}. \end{aligned} \qquad \text{1-577}$$

(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.

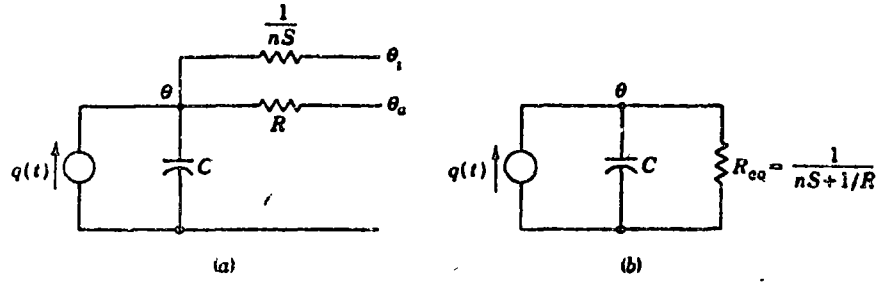


FIGURE 2-27
(*a*) Thermal network of an electric water heater; (*b*) simplified network.

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:

$$P_L = P_1 - P_2 \qquad (2\text{-}96)$$

The flow of fluid through an inlet orifice is given by[10]

$$q = ca\sqrt{2g\frac{\Delta p}{\omega}} \qquad (2\text{-}97)$$

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

$$q = C_x x \qquad (2\text{-}98)$$

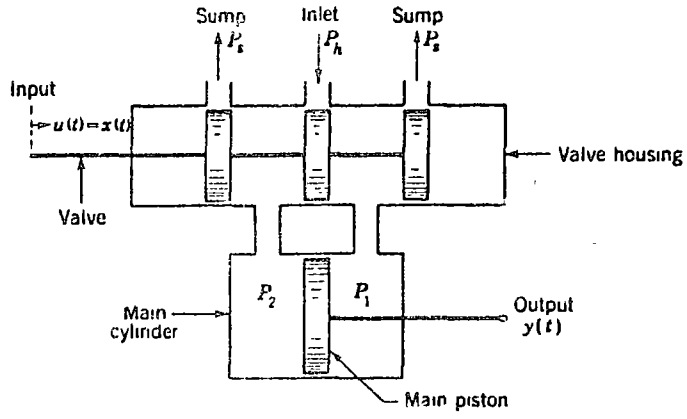


FIGURE 2-28
Hydraulic actuator.

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

$$q = C_b\, Dy \qquad (2\text{-}99)$$

Combining the two equations gives

$$Dy = \frac{C_x}{C_b} x = C_1 x \qquad (2\text{-}100)$$

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:

$$q = f(x,P_L) \qquad (2\text{-}101)$$

The differential dq, expressed in terms of partial derivatives, is

$$dq = \frac{\partial q}{\partial x}\, dx + \frac{\partial q}{\partial P_L}\, dP_L \qquad (2\text{-}102)$$

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

$$q = \left(\frac{\partial q}{\partial x}\right)_0 x + \left(\frac{\partial q}{\partial P_L}\right)_0 P_L \qquad (2\text{-}103)$$

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

$$C_b\, Dy + \frac{V}{K_B} DP_L + (L + C_p)P_L = C_x x \quad (2\text{-}110)$$

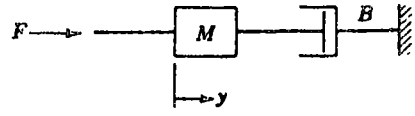


FIGURE 2-29
Load on a hydraulic piston.

The force developed by the main piston is

$$F = n_F A P_L = C P_L \qquad (2\text{-}111)$$

where n_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:

$$F = M D^2 y + B Dy = C P_L \qquad (2\text{-}112)$$

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:

$$\frac{MV}{CK_B} D^3 y + \left[\frac{BV}{CK_B} + \frac{M}{C}(L + C_p)\right] D^2 y + \left[C_b + \frac{B}{C}(L + C_p)\right] Dy = C_x x \qquad (2\text{-}113)$$

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

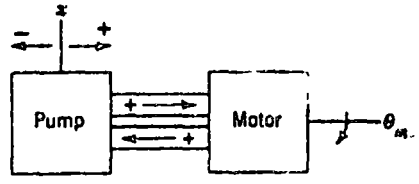


FIGURE 2-30
Hydraulic transmission.

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, $|A| = 0$. 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

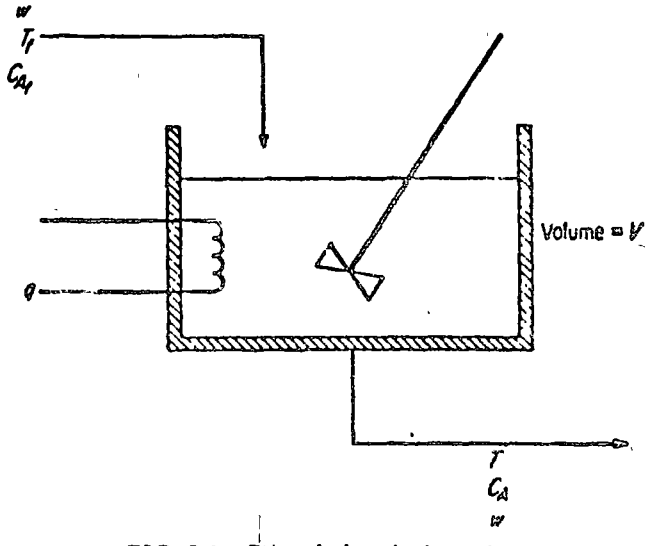


FIG. 9-2. Stirred chemical reactor.

flow unit in which the second-order reaction

$$2A \longrightarrow B$$

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 $\bar{T}$, $\bar{C}_A$, $\bar{w}$, and $\bar{q}$:

$$\frac{d\hat{T}}{dt} = -\frac{\bar{w}}{V\rho}\hat{T} + \frac{T_f - \bar{T}}{V\rho}\hat{w} + \frac{\hat{q}}{V\rho c_p} - \frac{2(\Delta H)k\bar{C}_A}{\rho c_p}\hat{C}_A \tag{9-9a}$$

$$\frac{d\hat{C}_A}{dt} = -\frac{\bar{w}}{V\rho}\hat{C}_A + \frac{C_{A_f} - \bar{C}_A}{V\rho}\hat{w} - 2k\bar{C}_A\hat{C}_A \tag{9-9b}$$

where $\hat{C}_A = C_A - \bar{C}_A$
$\hat{T} = T - \bar{T}$
$\hat{w} = w - \bar{w}$
$\hat{q} = q - \bar{q}$

Equation 9-9 may be conveniently represented by the following matrix differential equation:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \tag{9-10}$$

where

$$\mathbf{x}(t) = \begin{bmatrix} \hat{T} \\ \hat{C}_A \end{bmatrix} \quad \text{state vector}$$

$$\mathbf{u}(t) = \begin{bmatrix} \hat{q} \\ \hat{w} \end{bmatrix} = \text{manipulated inputs}$$

$$\mathbf{A} = \begin{bmatrix} -\dfrac{\bar{w}}{V\rho} & -\dfrac{2(\Delta H)k\bar{C}_A}{\rho c_p} \\ 0 & -\dfrac{\bar{w}}{V\rho} + 2k\bar{C}_A \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} \dfrac{1}{V\rho c_p} & \dfrac{T_f - \bar{T}}{V\rho} \\ 0 & \dfrac{C_{A_f} - \bar{C}_A}{V\rho} \end{bmatrix}$$

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 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- $$\dot{p}(t) = -\frac{\partial H}{\partial x(t)}$$

$$\dot{x}(t) = \frac{\partial H}{\partial p(t)}$$

$x(t_0) = x_0$ $\quad x(t_1) = x_1$ (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

$x(t_0) = x_0$

$p(t_1) = 0$ $\quad$ t_1 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 u as a function of time. For process systems, feedback control, i.e., control in which u is given as a function of the state 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

$$U_{\min} \leq u \leq U_{\max}$$

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 $c(0) = r_0$; $\dot{c}(0) = 0$).

$r < r_0$

$0 < \tau_2/\tau_1 \leq 0.9$

$$\left[\frac{(K-k) - (r_0/K_p - k)\exp(-t_1/\tau_2)}{(K - r/K_p)}\right]^{\frac{\tau_2}{\tau_1}} = \frac{(K-k) - (r_0/K_p - k)\exp(-t_1/\tau_1)}{(K - r/K_p)}$$

$0.9 < \tau_2/\tau_1 \leq 1$

$$\left[\frac{r_0/K_p - k}{K-k} - \exp\left(\frac{t_1}{\tau_2}\right)\right] \ln\left[\frac{(r_0/K_p - k) - (K-k)\exp(t_1/\tau_2)}{r_0/K_p - K}\right] + \frac{t_1}{\tau_2}\exp\left(\frac{t_1}{\tau_2}\right) = 0$$

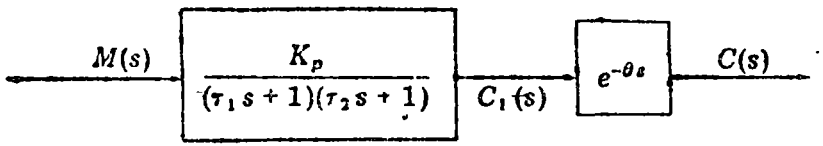
$0 < \tau_2/\tau_1 \leq 1$

$$t_2/\tau_2 = \ln\left[\frac{(r_0/K_p - k) - (K-k)\exp(t_1/\tau_2)}{r/K_p - K}\right]$$

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.

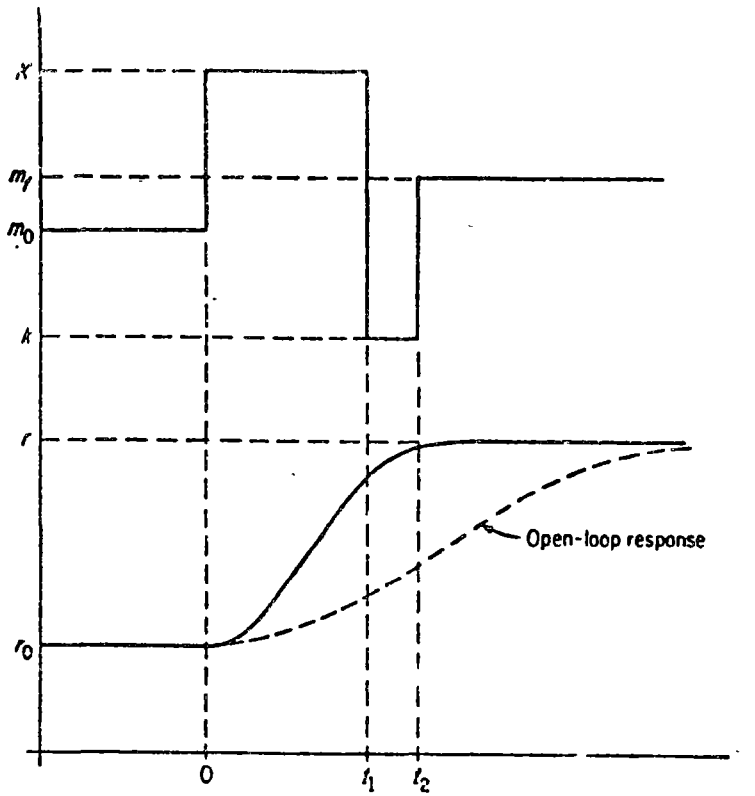


FIG. 9-3. Optimal response to a change in set point.

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

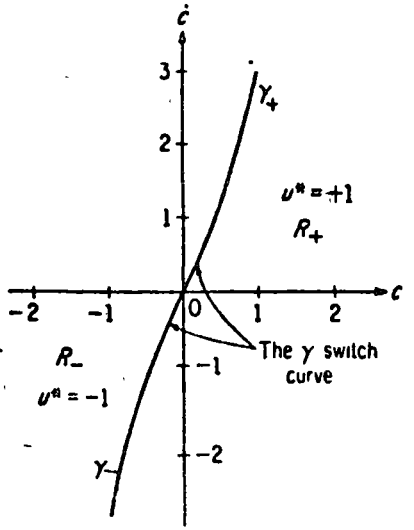


FIG. 9-4. Switching curve for a two-time-constant plant. (Reproduced by permission from M. Athans and P. L. Falb, *Optimal Control*, McGraw-Hill Book Company, New York, 1966.)

$c(t)$, $\dot{c}(t)$ 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

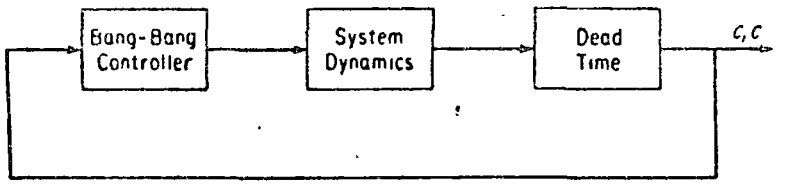


FIG. 9-5. Bang-bang control loop for systems with dead time.

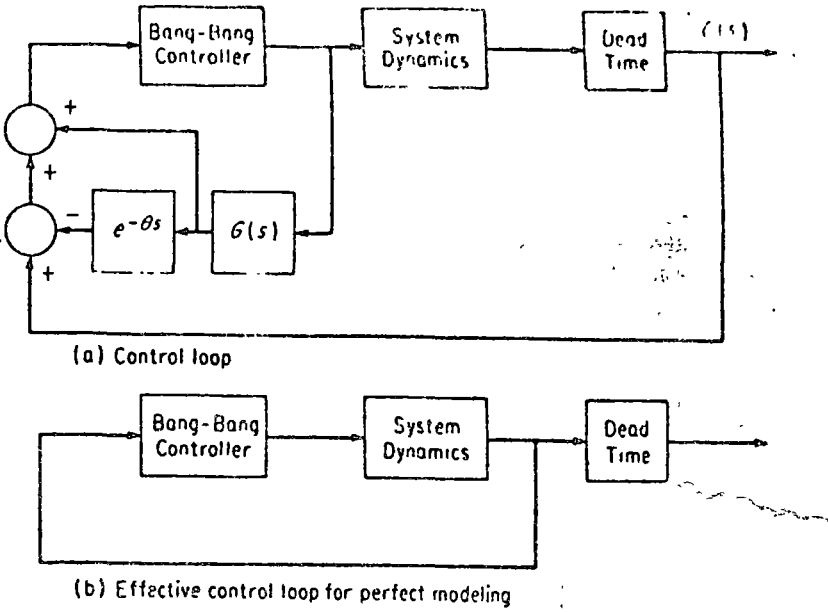


FIG. 9-6. Bang-bang controller coupled with the dead-time compensator.

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.

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

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \tag{9-22}$$

$$\mathbf{x}(0) = \mathbf{x}_0 \tag{9-23}$$

It is desired to control the system in such a manner as to minimize the cost functional

$$J = \frac{1}{2}\int_0^T [\mathbf{x}^T(t)\mathbf{Q}\mathbf{x}(t) + \mathbf{u}^T(t)\mathbf{R}\mathbf{u}(t)]\,dt \tag{9-24}$$

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.

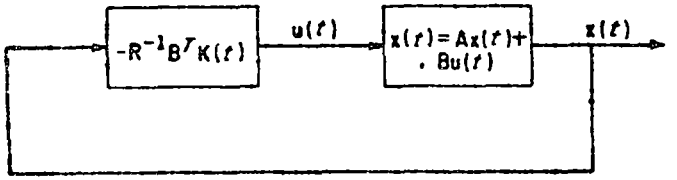


FIG. 9-7. Optimal controller.

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 $p(t)$ and Eq. 9-22 for $x(t)$ 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 $0 \leq t \leq T$.

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.

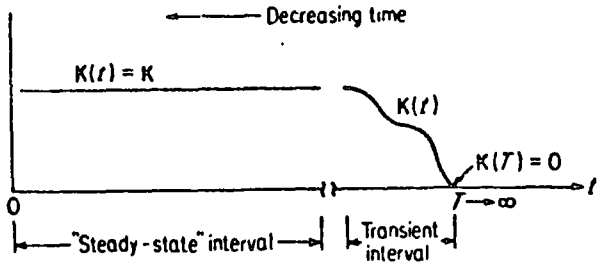


FIG. 9-8. A loose interpretation of the constant matrix K. As $T \longrightarrow \infty$, the "transient interval" tends to infinity and the "steady-state interval" occupies all finite times. (Reprinted by permission from M. Athans and P. L. Falb, *Optimal Control*, McGraw-Hill Book Company, New York, 1966.)

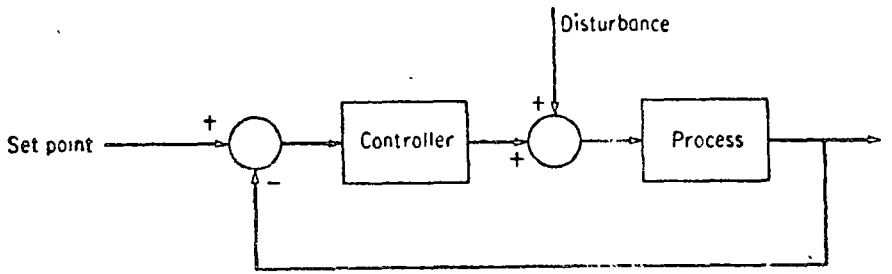


FIG. 9-9. Conventional representation of control loop

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

$$\frac{dx_1(t)}{dt} + x_1(t) + x_f = u_1(t) + u_f$$

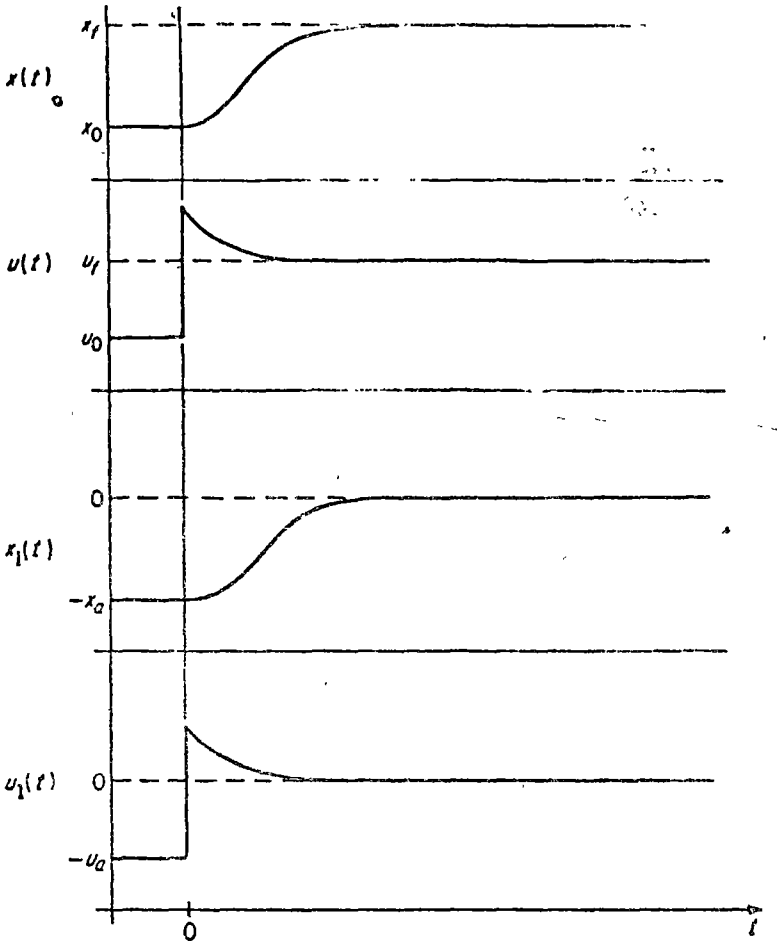


FIG. 9-10. Control and response for first-order system.

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

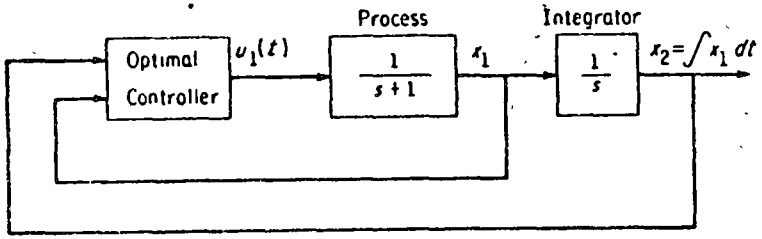


FIG. 9-11. A possible means for introducing an integral term into the optimal control law.

performance functional must be of the form

$$J = \int_0^\infty [q_1 x_1^2(t) + q_2 x_2^2(t) + u^2(t)]\,dt \qquad (9\text{-}39)$$

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.

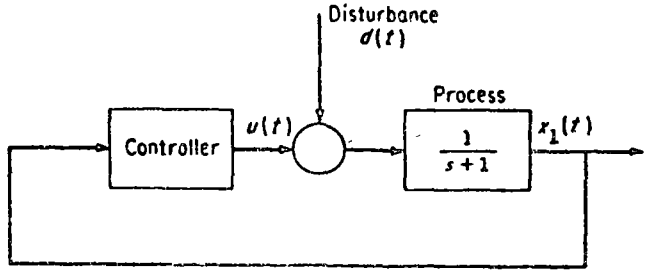
9-9 OPTIMAL CONTROL TO DISTURBANCE CHANGES

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

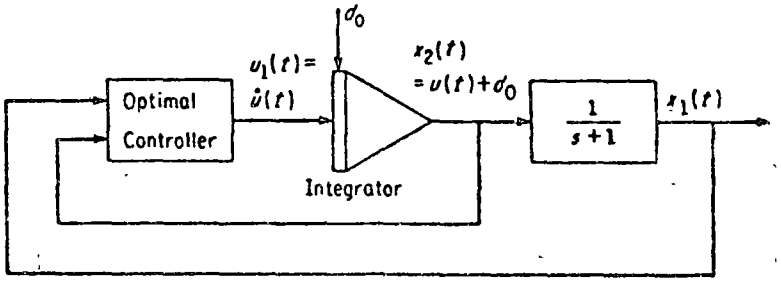
$$\dot{x}_1(t) = -x_1(t) + d(t) + u(t) \qquad (9\text{-}40)$$

$$x_1(0) = 0 \qquad (9\text{-}41)$$

Note that the disturbance appears as an input along with the control $u(t)$. To be cast into the optimal control formulation, we must trans-



(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.

form the problem in such a manner that the disturbance appears a an initial condition.

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 su 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.

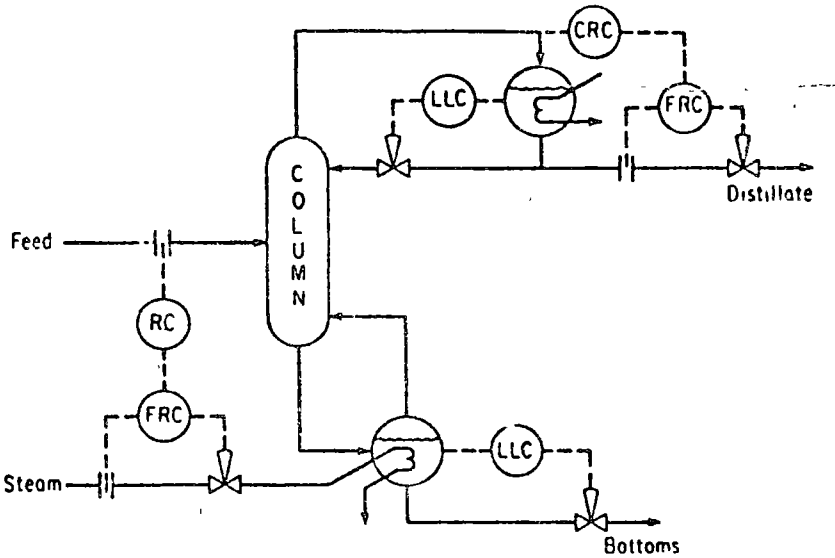


FIG. 9-13. Control scheme for distillation column.

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:

$$Y_D(s) = \frac{-0.0140}{0.55s + 1} D(s) + F_e(s)$$

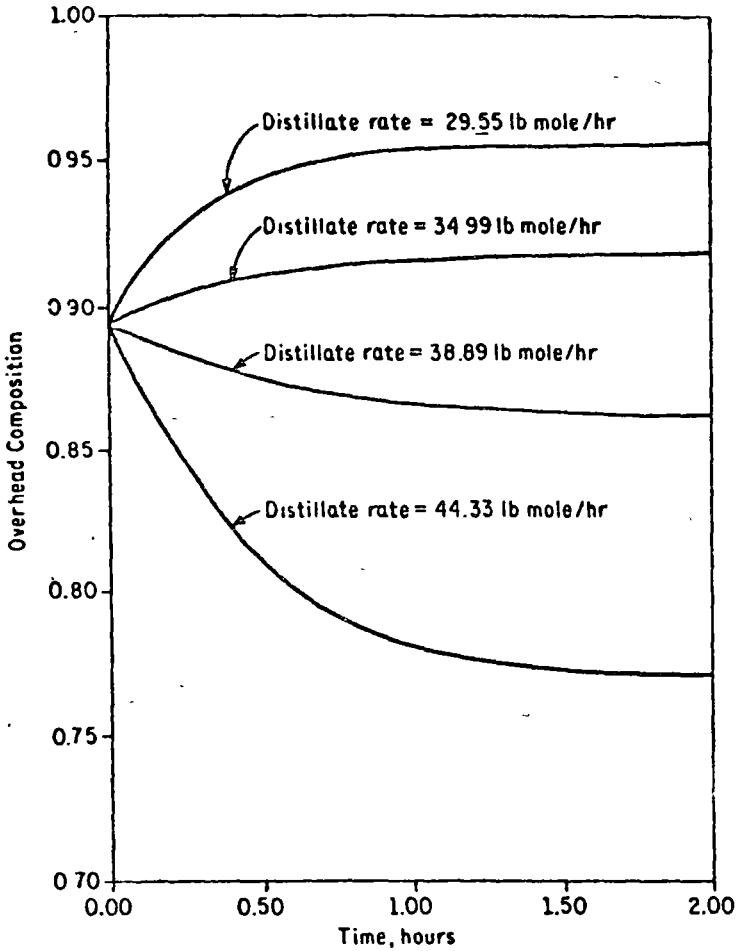


FIG. 9-14. Open-loop responses to several step changes in distillate rate.

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

$$J(D) = \int_0^\infty \{[y_{D_{set}} - y_D(t)]^2 + r\dot{D}^2\}\, dt$$

The cost $J(D)$ consists of two parts. The first part $[y_{D_{set}} - y_D(t)]^2$ penalizes for deviations of the controlled variable y_D from its desired value $y_{D_{set}}$. The second part $\dot{D}^2$ penalizes for changes in the manipulated variable D.

Applying the method presented previously gives the following control law:

$$D(t) = K_2 \int_0^t [y_{D_{set}} - y_D(\tau)]\, d\tau + K_1[y_{D_{set}} - y_D(t)] + D_0$$

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.

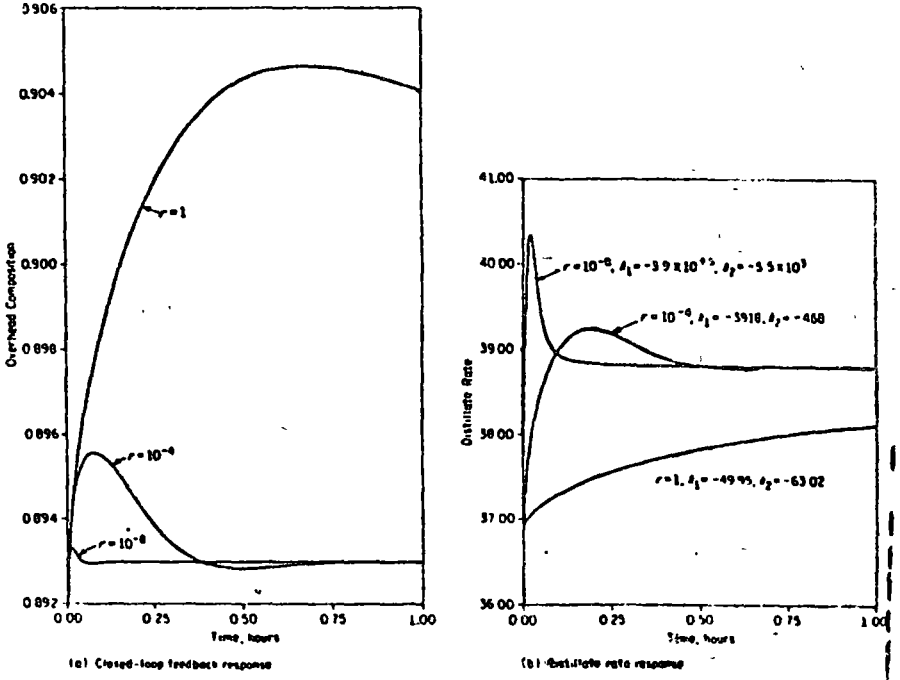


FIG. 9-15. Effect of r on the performance of the control system.

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