MODELADO Y EVALUACION DEL RENDIMIENTO DE COMPUTADORAS

MATERIAL DIDACTICO DE APOYO

DR. RAMON PUIGJANER TREPAT

OCTUBRE, 1 9 7 9

1. INTRODUCCION.

La evaluación del rendimiento de un sistema informático es algo que en la actualidad no tiene una respuesta concreta y única como sucede cuando nos preguntan, por ejemplo, cuál es el rendimiento de un motor eléctrico. Sin embargo es necesario darle una respuesta y ésta podría ser :

La medida de cómo un software determinado está utilizando el hardware con una determinada combinación de programas.

En los momentos actuales, esta medida no se concreta en un valor único, sino al contrario, es un conjunto de valores que varían, puede decirse, de una instalación a otra en función de la configuración concreta y del uso a que se destina la máquina. Sin embargo, antes de plantearnos qué y cómo hay que medir, hemos de justificar la necesidad de la evaluación del rendimiento de un sistema informático.

1.1. Necesidad de la evaluación de un sistema informático.

Esta necesidad ha aparecido como una consecuencia natural del aumento de potencia y complejidad de los sistemas informáticos.

Los primeros ordenadores estaban concebidos para que los utilizara el propio programador que, personalmente, controlaba lo que sucedía en el ordenador mientras se ejecutaba el programa.

Eran los tiempos en que prácticamente no existía el software y en que las decisiones fundamentales de evaluación y comparación se referían a la longitud de la palabra, al conjunto de instrucciones y a su implantación, al ciclo de base de la CPU o al tiempo de ejecución de una instrucción característica (normalmente, la instrucción de sumar).

La aparición de las ayudas software y de periféricos cada vez más sofisticados, el aumento de la dimensión de las memorias, las unidades centrales más complejas (multiprocesadores, "pipelines", etc.) con sistemas de interrupciones muy sofisticados, han hecho que la evaluación del rendimiento se fuera convirtiendo en un cuerpo de doctrina en el que no sólo se ha de considerar el hardware sino también las facilidades proporcionadas por el software al acercar la máquina a los usuarios teniendo en cuenta, por el contrario, el "overhead" (es decir, gastos generales de máquina) que lleva asociado todo software. Estas consideraciones nos hacen comprender que la evaluación del rendimiento no es tarea sencilla, ya que ha de tener en cuenta muchos y variados aspectos del hardware, el software y las aplicaciones que se han de llevar a término en el sistema informático.

Un primer motivo para evaluar el rendimiento de los sistemas informáticos es el poder compararlos, en general a priori, cuando se trata de seleccionarlos de cara a adquirir uno. Es, posiblemente, el caso en que con más frecuencia aparece la necesidad de evaluar el rendimiento de los sistemas informáticos.

No obstante, hay otros casos que, aunque menos frecuentes, justifican también la necesidad de disponer de técnicas de evalua

ción del rendimiento. Tal es el caso de cuando tratamos de dictaminar si el uso de una máquina es el correcto para aumentar la capacidad de un sistema ("tuning") o para mantener el uso de un sistema dentro de unas características especificadas, o si tratamos de predecir qué efecto tendrá sobre el sistema la implantación de nuevas aplicaciones, intentando prever con tiempo las ampliaciones necesarias.

1.2. Tipos de medidas.

Toda clasificación será siempre difícil y más aún en un tema aún no estabilizado del todo, como es el que nos ocupa. A pesar de ello, una posible clasificación, aunque no exhaustiva ni perfecta, podría ser clasificar las medidas en estáticas y dinámicas según la incidencia que tienen las aplicaciones en la medida realizada.

1.2.1. Medidas estáticas.

1.2.1.1. Mix.

Miden básicamente el rendimiento interno de la CPU en número de instrucciones ejecutadas por unidad de tiempo o a la inversa, el tiempo medio de ejecución de una instrucción. Es una manera frecuente de comparar las potencias de las unidades centrales.

Fabricantes, asociaciones de usuarios y consultores han establecido diferentes ponderaciones de los tiempos de ejecución de las instrucciones (o funciones) según sus frecuencias de aparición.

El principal inconveniente del mix se deriva de querer ser a la vez demasiado general y simplista, ya que no tienen, ni

pueden tener en cuenta ni la longitud de las posiciones de memoria ni la estructura de las instrucciones por un lado, ni por otro, tiene en cuenta el software asociado al hardware

1.2.1.2. Kernels.

Los programas Kernel generalizan el procedimiento del mix a nivel de funciones completas que intervienen con frecuencia en todas las aplicaciones.

Estos programas pueden valorar el comportamiento de un ordenador frente a un problema típico (clasificación, inversión de una matriz, etc.) o pueden usarse para tratar de caracterizar la carga normal de una instalación (benchmark sintético).

1.2.2. Medidas dinámicas.

1.2.2.1. Benchmarks:

Es un método bastante frecuente de comparación de máquinas frente a una carga característica de instalación concreta, determinar exactamente la carga característica y cómo los programas aprovechan las peculiaridades de un software determinado.

Variantes de este sistema se usan para contrastar monitores y para validar modelos.

1.2.2.2. Monitores.

Los monitores son las herramientas de medición que permiten analizar el comportamiento de todos los elementos de un sistema informático por medio de realizaciones de hardware, software o mixtas.

Son herramientas imprescindibles para evaluar el comportamiento de un sistema existente a pesar de sus limitaciones y las perturbaciones que pueden introducir en el sistema medido.

Aparte de su utilización directa para tomar medidas de un sistema existente, permiten determinar la aproximación de un benchmark a la carga real, obtener datos para la construcción de modelos y su validación posterior.

1.2.2.3. Modelos.

Es la herramienta que hay que utilizar cuando se trata de evaluar el comportamiento de un sistema en el que hay algún elemento (hardware o software) que no está instalado.

En general se fundamentan en la teoría de colas pudiéndose considerar las colas en forma individual o unidas, en redes abiertas o cerradas.

El tratamiento de estos modelos se pueden hacer aprovechado los métodos analíticos de la teoría de colas o por medio de la simulación. La limitación de los primeros métodos es su incapacidad para tratar en forma exacta determinadas estructuras y comportamientos de colas que existen en los modelos de los ordenadores. El segundo de los métodos no tiene estas limitaciones, pero, en general es más caro de cálculo que el primero.

No obstante la principal dificultad de estos métodos es la obtención de datos lo suficientemente precisos como para poder construir el modelo con el grado de aproximación que se exige.

1.3. Magnitudes a medir.

La evaluación del comportamiento de un sistema informático se lleva a cabo por medio de un conjunto de parámetros cuantitativos, que son las medidas del comportamiento. Estas medidas las podemos agrupar en dos clases, las que hacen referencia al comportamiento del hardware y del software del ordenador y las que hacen referencia a cómo el usuario ve el comportamiento del sistema.

Hay que tener en cuenta, sin embargo, que lo que hay que medir varía de una instalación a otra, ya que los parámetros que caracterizan el comportamiento de un sistema dependen del uso al que esté destinado, es decir de las necesidades de los usuarios del sistema. La lista de parámetros que se expone a continuación no pretende ser exhaustiva ni, por otra parte, que se tomen en todas las instalaciones todas las medidas que se citan aquí.

1.3.1. "Throughput".

Es la cantidad de trabajo útil por unidad de tiempo con una carga determinada (normalmente se mide en trabajos o transacciones por unidad de tiempo).

1.3.2. Capacidad.

Máxima cantidad de trabajo útil que puede realizar por unidad de tiempo con una carga determinada.

1.3.3. Tiempo de respuesta.

Es el tiempo transcurrido entre la entrega de un trabajo o una

transacción a la máquina y la recepción del resultado o la respuesta.

1.3.4. Factor de multiprogramación del tiempo transcurrido.

Es la relación entre el tiempo de respuesta de un trabajo en multiprogramación y el tiempo de respuesta del mismo trabajo en monoprogramación.

1.3.5. Factor de ganancia.

Es la relación entre el tiempo total necesario para ejecutar un conjunto de trabajos en multiprogramación y el tiempo total necesario para ejecutar el mismo conjunto secuencialmente.

1.3.6. Solapamiento de componentes.

Es el porcentaje del tiempo en que dos o más componentes funcionan simultáneamente.

1.3.7. Factor de utilización de un componente.

Es el porcentaje del tiempo durante el cual un componente está siendo utilizado.

1.3.8. "Overhead".

Es el porcentaje de tiempo de CPU en que ésta está ejecutando código del sistema operativo.

1.3.9. Frecuencia de fallo de página.

Es el número de fallos de página por unidad de tiempo, en un sistema de memoria visual.

1.4. Magnitudes que caracterizan la carga.

En caso de tomar medidas de un sistema existente se nos plantea el problema de cuando podemos considerar que el funcionamiento del sistema es el característico.

Cuando se nos plantea la construcción de un modelo se nos plantea el problema de determinar cuáles son los datos que caracterizan la carga que queremos que soporte el sistema.

Si debemos plantear un benchmark deberemos determinar cuáles son los programas que nos representan significativamente la carga del sistema.

El común denominador de estos problemas reside en la determinación de las magnitudes que determinan la carga del sistema.

En el apartado 4 se tratan con detalle los métodos de caracterización de la carga, pero a continuación se definen algunos términos utilizados en la caracterización de la carga.

1.4.1. Tiempo de CPU por trabajo.

Es el tiempo de CPU necesario para un trabajo.

1.4.2. I/O por trabajo.

Es el número total de operaciones de entrada/salida que requiere un trabajo.

1.4.3. Tiempo de servicio por I/O.

Es el tiempo necesario para procesar una operación de entrada/salida.

1.4.4. Tiempo entre llegadas.

Es el tiempo entre dos requerimientos sucesivos para un servicio del sistema.

1.4.5. Prioridad.

Es la que el usuario asigna a cada uno de los trabajos.

1.4.6. Memoria necesaria.

Es la cantidad de memoria que requiere un trabajo para ser ejecutado.

1.4.7. Medida del conjunto de trabajo.

Es el número de páginas de un trabajo que se han de mantener en memoria principal.

1.4.8. Localidad de las referencias.

Es el tiempo en el que todas las referencias a memoria hechas por un trabajo permanecen dentro de una página o conjunto de páginas.

Si consideramos la ejecución de un programa como una sucesión de referencias a memoria, la localidad del programa será tanto más grande cuanto más tiempo estemos dentro de la página o conjunto de páginas considerado. Esta es la aparente contradicción de medir una magnitud ligada al espacio (la localidad) por medio del tiempo.

1.4.9. Tiempo de respuesta del usuario.

Es el tiempo que el usuario de un terminal de un sistema interactivo necesita para generar una nueva pregunta (es el tiempo de pensar y teclear).

1.4.10. Número de usuarios simultáneos.

Es el número de usuarios interactivos que trabajan simultáneamente en un momento dado.

1.4.11. Intensidad del usuario.

Es la relación entre el tiempo de proceso por requerimiento y el tiempo de respuesta del usuario.

1.5. Magnitudes para controlar el rendimiento.

Hasta ahora nos hemos preocupado de qué hay que medir y de cómo podemos escoger cuándo hay que medir, pero, ¿ Qué hemos de hacer si el rendimiento del sistema no nos satisface ? ¿ Cuáles son las teclas que podemos mover para mejorar el comportamiento ?

Las modificaciones que podemos introducir en nuestro sistema para mejorar su comportamiento las podremos hacer a todos los niveles que influyan en el comportamiento y que son :

- Configuración del sistema.
- Políticas de los programas del sistema operativo.
- Eficiencia de procesador del conjunto de instrucciones.
- Velocidad de los componentes hardware.

Las acciones sobre estas variables las podemos conseguir,

entre otras, por las varias maneras que se citan a continuación :

1.5.1. Ajuste de los parámetros de control del sistema.

1.5.1.1. Medida del quantum.

Es el quantum de tiempo en que se asigna CPU de un sistema de tiempo compartido a los diferentes trabajos.

1.5.1.2. Prioridad interna.

Es la prioridad que se fundamenta en las demandas de servicio de un trabajo y en los servicios ya recibidos.

1.5.1.3. Grado de multiprogramación.

Es el número de trabajos que están simultáneamente en memoria principal y por lo tanto que tienen opción a utilizar la CPU.

1.5.1.4. Medida de la partición de memoria.

Es la cantidad de memoria principal asignada a un solo trabajo.

1.5.1.5. Medida de la ventana.

Es el intervalo de tiempo durante el cual se toman medidas para determinar el conjunto de trabajo de un trabajo en un entorno de memoria virtual.

Evidentemente según la duración del tiempo durante el cual tomamos medidas para determinar el conjunto de trabajo, éste varía y por lo tanto el valor medio de aquella magnitud estará

afectado por los valores que intervengan en su cálculo.

1.5.1.6. Máxima frecuencia de paginación permitida.

Es la frecuencia máxima de paginación permitida en un sistema paginado interactivo.

1.5.1.7. Indice de supervivencia de páginas.

Es el número de ráfagas de CPU recibidas por un programa antes de que saque de la memoria principal una página no referenciada.

1.5.1.8. Número de usuarios simultáneos.

Es el número máximo de usuarios de terminal permitidos en el sistema.

1.5.2. Cambio o modificación de las políticas de gestión de recursos.

Esta posibilidad es accesible sólo en algunos sistemas operativos ya que normalmente en nuestro país la programación del sistema no llega a modificar el sistema operativo que suministra el fabricante de hardware.

1.5.3. Equilibrado de la distribución de cargas.

Es necesario que los diferentes componentes del sistema tengan una carga equilibrada, lo que se puede conseguir por medio de cambios en la asignación de los dispositivos periféricos a los canales o en la asignación de los archivos a los dispositivos físicos de almacenaje, cambios en la distribución de los componentes software en la jerarquía de memoria del sistema (rutinas transitorias convertidas en residentes, etc.) etc.

1.5.4. Sustitución o modificación de los componentes del sistema.

Básicamente consiste en la modificación de la configuración del sistema ampliándola o cambiándola para hacer desaparecer los cuellos de botella que se hayan detectado.

1.5.5. Modificación de los programas.

Se pueden hacer también modificaciones en los programas, especialmente en los sistemas de memoria virtual, para mejorar las propiedades de localidad y disminuir la paginación necesaria para acabar la ejecución.

1.6. Relaciones fundamentales (análisis operacional) (DEN78)

Estas relaciones deducidas por Buzen y Denning tienen por objeto llegar a las mismas relaciones que las que se deducen a partir de la teoría de colas pero partiendo de hipótesis operacionales, es decir, comprobables por medida.

1.6.1. Variables operacionales.

Las variables operacionales pueden ser básicas, que se miden directamente durante el periodo de observación, o deducidas, que se calculan a partir de las básicas.

La figura 1.1 nos muestra un sistema de un solo servidor con una fila de espera con las cuatro magnitudes básicas :

T longitud del periodo de observación.
A número de llegadas producidas durante el periodo de observación.

B cantidad de tiempo total durante el cual el sistema está ocupado durante el periodo de observación ($B \leq T$).

C número de terminaciones de servicio producidas durante el periodo de observación.

Cuatro magnitudes deducidas importantes son :

$\lambda = A/T$, la frecuencia de llegada (trabajos/seg.)

$X = C/T$, la frecuencia de salida (trabajos/seg.)

$U = B/T$, la utilización (fracción de tiempo que el sistema está ocupado)

$S = B/C$, el tiempo medio de servicio por trabajo acabado.

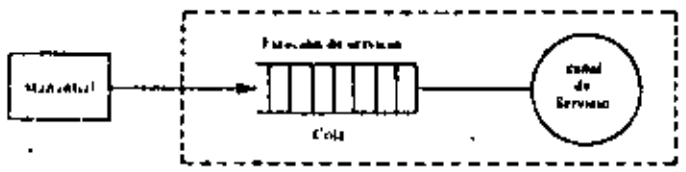


fig. 1.1.-

1.6.2. Leyes operacionales.

Es fácil ver que las magnitudes deducidas satisfacen la ecuación.

$$U = XS$$

que constituye la ley operacional de la utilización.

Supongamos además, que el número de llegadas es igual al número de terminaciones durante el periodo de observación, es decir

$$A = C$$

Esta suposición se denomina equilibrio del flujo de trabajos e implica que

$$\lambda = X$$

Esta suposición , que no es siempre cierta, se cumple normalmente en periodos de observación largos ya que la proporción de trabajos no terminados es pequeña, (A-C)/C. En cualquier caso esta hipótesis es comprobable. Si admitimos esta hipótesis, es fácil ver que

$$U = \lambda S$$

que constituye un ejemplo de teorema operacional.

1.6.3. Medidas operacionales en redes.

La figura 1.1 mostraba un modelo de un solo recurso. Este modelo podemos emplearlo para representar un dispositivo de entrada-salida o una CPU, por ejemplo. El modelo completo de un sistema informático puede desarrollarse conectando modelos tales como el de la fig 1.1 del mismo modo que los dispositivos se conectan en un sistema informático real.

1.6.3.1. Tipos de redes.

Al establecer un modelo por conexión de K dispositivos supondremos que un trabajo circula por la red desde su entrada a su salida, esperando en las colas y recibiendo servicio en los distintos dispositivos.

Se supone además que no existe solape entre los servicios que recibe un trabajo en distintos dispositivos, lo cual es practicamente cierto siempre; y que no existe interacción en las condiciones de servicio de los distintos dispositivos, lo cual se cumple con menos frecuncia que la suposición anterior.

Diremos que un trabajo está en el dispositivo i, si está esperando

en la cola o recibiendo servicio. Indicaremos por n_i el número de trabajos en el dispositivo i, y por $N=n_1+n_2+\ldots+n_K$ el número total de trabajos en el sistema. La frecuencia de salida del sistema X_0 es el número de trabajos que salen del sistema por segundo. Si el sistema es abierto X_0 es conocido y N varía cuando un trabajo entra o sale del sistema. Si el sistema es cerrado N es fijo y se obtiene uniendo en el modelo la entrada con la salida.

En un sistema abierto se supone que X_0 es conocido y se trata de caracterizar la distribución de N. El análisis de un sistema cerrado empieza con N dado y trata de determinar X_0 como el flujo existente en la conexión entre la entrada y la salida. En ambos casos se trata de determinar las longitudes de la cola y los tiempos de respuesta de los dispositivos.-

La figura 1.2 nos muestra una red de servidor central cerrada, en la que un nuevo trabajo entra en el sistema tan pronto como termina un trabajo activo. Este comportamiento es típico de un sistema trabajando en batch. El "througput" del sistema en estas condiciones nos lo indicará X_0.

Los sistemas de tiempo compartido que se utilizan desde terminales también pueden representarse por redes cerradas (figura 1.3). El modelo está separado en dos subredes abiertas : el subsistema central, que consta de los dispositivos de entrada salida y las CPU, y el sistema de terminales. Cada terminal está manejado por un usuario que alterna periodos de pensar y esperar. En un periodo de pensar el usuario está contemplando el resultado de su requisición anterior y preparando la siguiente y, por lo tanto, el subsistema central no efectúa ningún trabajo para él. Cuando transmite una requisición el usuario entra en un periodo de espera

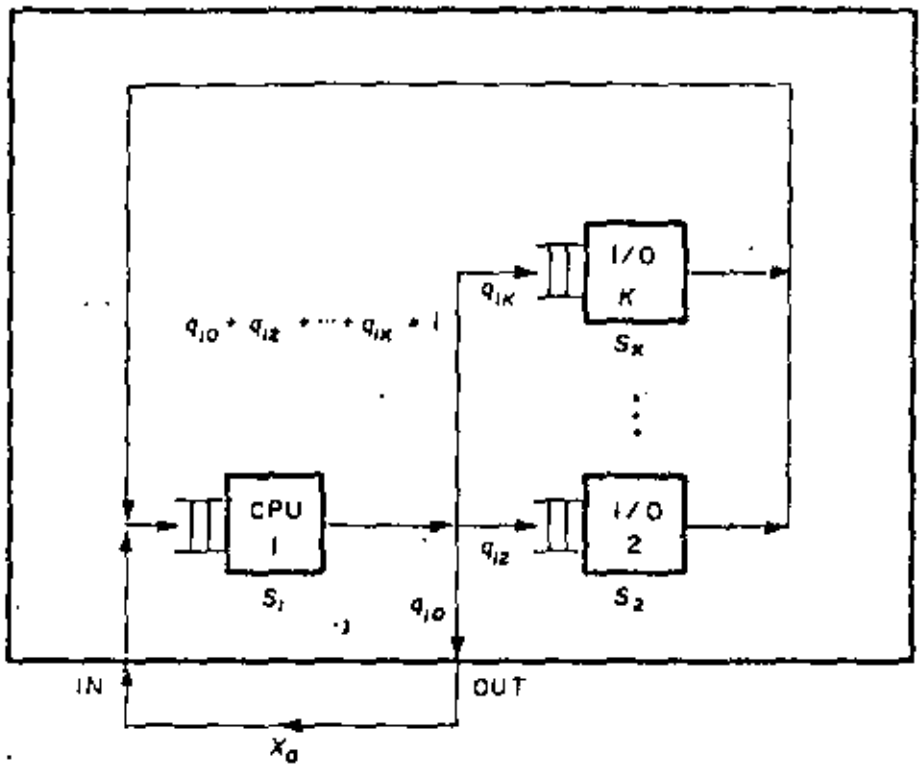


Figura 1.2.-

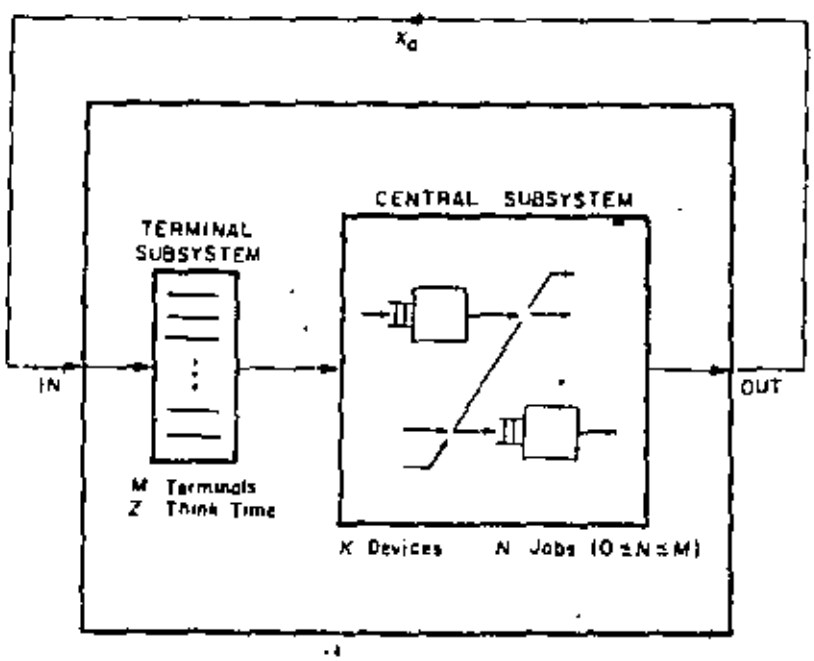


Figura 1.3.-

en el que permanece mientras el subsistema central efectúa el trabajo solicitado. El tiempo que el usuario gasta en un periodo de pensar lo denominaremos tiempo de pensar y lo representaremos por Z. El tiempo medio que un usuario gasta en un intervalo de espera se denomina tiempo de respuesta (del subsistema central) y lo representaremos por R. Puesto que los usuarios piensan independientemente, el tiempo de pensar Z será independiente del número de terminales M. Sin embargo, puesto que los trabajos solicitados desde los terminales compiten por los recursos del subsistema central, R es función de M.

Es posible definir también sistemas mixtos que son abiertos para algunas cargas y cerrados para otras.

1.6.3.2. Magnitudes operacionales básicas.

Supongamos que se ha medido el sistema durante un periodo de observación de T segundos y que se recogen para cada dispositivo i, i=1,2,...K, los siguientes datos :

- A_i número de llegadas.
- D_i tiempo de ocupación total (tiempo durante el que $n_i > o$)
- C_{ij} número de veces que un trabajo requiere servicio del dispositivo j inmediatamente después de terminar el servicio en el dispositivo i.

Estas magnitudes son similares a las que hemos especificado en 1.6.1. para el servidor de la figura 1.1. Si consideramos el mundo exterior como dispositivo 0, podemos definir también

A_{oj} número de trabajos cuya primera requisición de servicio se produce en el dispositivo j.

C_{io} número de trabajos cuya última requisición de servicio se produce en el dispositivo i.

Supondremos $C_{oo}=0$. Sin embargo C_{ii} puede ser mayor que 0 para $i \neq o$. El número de trabajos terminados en el dispositivo i es :

$$C_i = \sum_{j=o}^{K} C_{ij} \quad , \; i = 1, \ldots, K.$$

Los números de llegadas y salidas del sistema son, respectivamente,

$$A_o = \sum_{j=1}^{K} A_{oj} \quad , \quad C_o = \sum_{i=1}^{K} C_{io}$$

Si el sistema es cerrado $A_o = C_o$

Las magnitudes operacionales deducidas las definiremos ahora

$U_i = B_i/T$, utilización del dispositivo i

$S_i = B_i/C_i$, tiempo medio de servicio del dispositivo i.

$X_i = C_i/T$ frecuencia de salida del dispositivo i.

$q_{ij} = C_{ij}/C_i$ si $i = 1, \ldots, K$.

$= A_{oj}/A_o$ si $j = 0$.

la frecuencia de direccionamiento, es decir la fracción de trabajos que se dirigen al dispositivo j al terminar el servicio en el dispositivo i.

$$\left(\sum_{i=1}^{K} X_i \; q_{io} \right.$$

Observaremos además que $X_o, X_1, \ldots X_k$ no pueden ser consideradas como "throughputs" ya que no hemos establecido ninguna hipótesis de flujo equilibrado de trabajos.

Las razones de visita, que expresan el número medio de requisiciones de un dispositivo por trabajo, pueden calcularse siempre en forma única a partir de las ecuaciones de equilibrio de flujo de trabajos. Definamos

$$V_i = X_i/X_o$$

donde V_i es el flujo de un trabajo a través del dispositivo i respect a flujo de salida del sistema. Nuestras definiciones implican que $V_i = C_i/C_o$. Puesto que V_i puede interpretarse como el número medio de visitas al dispositivo i por trabajo, podemos denominarla razón de visita.

La relación $X_i = V_i X_o$ es la ley operacional denominada ley del flujo forzado, que establece que el flujo en cualquier parte del sistema determina el flujo en todas partes del sistema.

Sustituyendo X_i por $V_i X_o$ en las ecuaciones de equilibrio del flujo, obtenemos las ecuaciones de las razones de visita.

$$V_o = 1.$$
$$V_j = q_{oj} + \sum_{i=1}^{K} V_i\, q_{ij} \;, \quad j = 1...K.$$

que son K +1 ecuaciones con K +1 incógnitas de las que siempre puede obtenerse una solución única.

Para determinar el tiempo de respuesta medio por trabajo, R, para un sistema abierto o cerrado, debemos aplicar la ley de Little al sistema total.

$$R = \bar{N}/X_o$$

donde $\bar{N} = \bar{n}_1 + \ldots + \bar{n}_K$. Si se desconocen $\bar{N}$ o X_o podemos usar un metodo alternativo. Puesto que $\bar{n}_i = X_i R_i$ (ley de Little aplicada al dispositivo i) y

$X_i = V_i X_o$ (ley del flujo forzado), tenemos

$$\frac{\bar{n}_i}{X_o} = V_i R_i$$

de donde podemos obtener

$$R = \sum_{i=1}^{K} V_i \quad R_i$$

que es la ley del tiempo de respuesta general, que se cumple incluso si el flujo de trabajos del sistema no es equilibrado.

La ley de Little puede usarse para calcular el tiempo de respuesta R del subsistema central de la figura 1.3 en un sistema de tiempo compartido. El tiempo medio de un ciclo completo de pensar y esperar es Z+R. Cuando el flujo es equilibrado X_o indica la frecuencia de realización de estos ciclos. Por la ley de Little $(Z+R)X_o$ debe ser el número total de usuarios que estén en un ciclo pensar-esperar. Por consiguiente :

$$N = (Z+R)X_o$$

de donde $R = \frac{N}{X_o} - Z.$

que es la fórmula del tiempo de respuesta de un sistema interactivo.

Ejemplo : Consideremos un sistema mixto como el de la figura 1.5 con las siguientes características :

40 terminales
Tiempo de pensar : 15 segundos.
Tiempo de respuesta interactivo : 5 segundos.
Tiempo medio de servicio del disco : 40 mseg.
Cada trabajo interactivo genera 10 accesos al disco.
Cada trabajo batch genera 5 accesos al disco
Utilización del disco : 90%
Deseamos calcular el "throughput" del sistema batch y estimar luego una cota inferior del tiempo de respuesta interactivo suponiendo que se triplica el "throughput" de batch.

La fórmula del tiempo de respuesta interactivo nos da el "throughput"

Este principio es una buena aproximación en periodos de observación suficientemente largos para que la diferencia entre entradas y salidas, $A_i - C_i$, sea pequeña en comparación con C_i. Será exacto si $n_i(o) = n_i(T)$, es decir eligiendo como punto final del periodo de observación un instante en que cada servidor se halle en el mismo estado que en el instante inicial (que es la idea subyacente en el metodo de los "puntos de regeneración" para realizar simulaciones).

Si se cumple el principio antes enunciado podemos considerar X_i como el "throughput" del dispositivo i. Expresando en forma de ecuación este principio podemos escribir

$$C_j = A_j = \sum_{i=o}^{K} C_{ij}, \quad i = 0, \ldots K.$$

Teniendo en cuenta la definición de $q_{ij} = C_{ij}/C$,

$$C_i = \sum_{i=o}^{K} q_{ij}$$

y empleando la definición $X_i = C_i/T$, obtenemos

$$X_j = \sum_{i=o}^{K} X_i \quad q_{ij}, \quad i = 0, \ldots k$$

que son las ecuaciones del equilibrio del flujo de trabajos.

Si la red es abierta, el valor de X_o se especifica externamente y las ecuaciones tienen una solución única para las incógnitas X_i Sin embargo, si la red es cerrada, X_o es desconocido inicialmente y las ecuaciones no tienen una solución única puesto que puede comprobarse que hay K ecuaciones independientes con K+1 incognitas A pesar de ello contienen información de considerable valor.

Por otro lado la ley de utilización

$$U_i = X_i S_i$$

se cumple para cada dispositivo.

Aunque n_i indica el número de trabajos en el dispositivo i, que varía a lo largo del periodo de observación, y para ponerlo de manifiesto lo representaremos por $n_i(t)$. La figura 1.4 nos muestra un ejemplo de tal evolución. Si representamos por W_i el área comprendida por la función $n_i(t)$, podremos escribir que el número medio de trabajos en el dispositivo i, $\bar{n}_i$, será

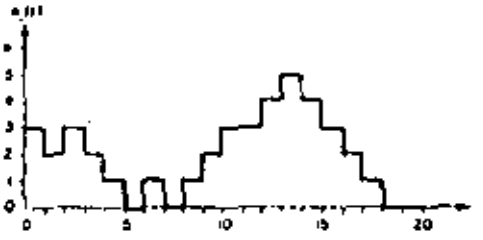


Figura 1.4.-

$$\bar{n}_i = W_i/T$$

El tiempo medio de respuesta del dispositivo i, R_i, está relacionado también con W_i, puesto que esta magnitud puede considerarse como el número de "trabajos x segundo" acumulados en el dispositivo i. R_i se define como el tiempo promedio de permanencia por trabajo completado, es decir,

$$R_i = W_i/C_i$$

Una consecuencia inmediata de estas definiciones es la ley operacional $\bar{n}_i = X_i R_i$

denominada ley de Little.

1.6.4. Análisis del flujo de trabajos.

Admitimos para cada dispositivo i el principio del equilibrado del flujo de trabajos, es decir, que para cada dispositivo i, X_i es igual a la frecuencia total de entrada i.

1.6.5. Análisis de los "cuellos de botella".

Se trata aquí del comportamiento asintótico del "throughput" y del tiempo de respuesta de sistemas cerrados cuando aumenta el número de trabajos en el sistema N. Supondremos que las razones de visita y los tiempos medios de servicio son invariantes con N.

Observemos que la relación de las frecuencias de terminación para dos dispositivos cualesquiera es igual a la relación de sus razones de visita

$$\frac{X_i}{X_j} = \frac{V_i}{V_j}$$

puesto que $U_i = X_i S_i$, podemos establecer que

$$\frac{U_i}{U_j} = \frac{V_i S_i}{V_j S_j}$$

Si el dispositivo i se satura, su utilización alcanza el 100% es decir $U_i = 1$ y, por lo tanto,

$$X_i = 1/S_i$$

Si el subíndice b indida el dispositivo que se satura al crecer N, corresponderá a aquel que tenga un producto $V_i S_i$ mayor puesto que las relaciones de utilizaciones vienen fijadas por esos productos. Es decir

$$V_b S_b = \max (V_1 S_1, \ldots V_K S_K)$$

Si N se hace grande, observaremos $U_b = 1$, y $X_b = 1/S_b$, puesto que $X_o/X_o = 1/V_b$, implica que $X_o = \frac{1}{V_b S_b}$

que es el máximo valor posible del "throughput" del sistema. Puesto que $V_i S_i$ es el total de todas las requisiciones al dispositivo i, la suma

$$R_o = V_1 S_1 + \ldots + V_K S_K$$

interactivo.

$X'_o = N/(R+Z) = 40/(15+5) = 2$ trabajos/seg.

Si el subíndice i hace referencia al disco podemos escribir.

$X_i + X'_i = V_i/S_i = 0,9/0,04 = 22,5$, accesos/seg.

La ley del flujo forzado implica que el componente interactivo es,

$X'_i = V'_i \; X_o = 10,2 = 20$ accesos/seg.

por lo que el componente batch es

$X_i = 22,5 - 20 = 2,5$ accesos/seg.

Usando de nuevo la ley del flujo forzado, hallamos que el "throughput" batch es

$X_o = X_i/V_i = 2,5/5 = 0.5$ trabajos/seg.

Consideremos ahora el efecto de triplicar el "throughput" batch. Si X_o pasa a ser 1,5 trabajos/seg. sin cambiar V_i, entonces $V_i \; X_o = 7,5$ accesos/seg. Suponiendo que el incremento de "throughput" no cambia su tiempo de servicio, la máxima frecuencia de terminación en el disco es $1/S_i = 25$ accesos/seg. esto implica que la frecuencia de terminación de la carga interactiva, X'_i, no puede exceder 25-7,5 = 17,5 accesos/seg. Por consiguiente

$$X'_o = X'_i/V'_i \qquad 17,5/10 = 1,75 \text{ trabajos/seg.}$$

y

$$R' = N/X'_o - Z \qquad 40/1,75 - 15 \gg 7,9 \text{ seg.}$$

Por lo tanto triplicar el "throughput" batch provoca un aumento del tiempo de respuesta interactivo de 2,9 segundos.

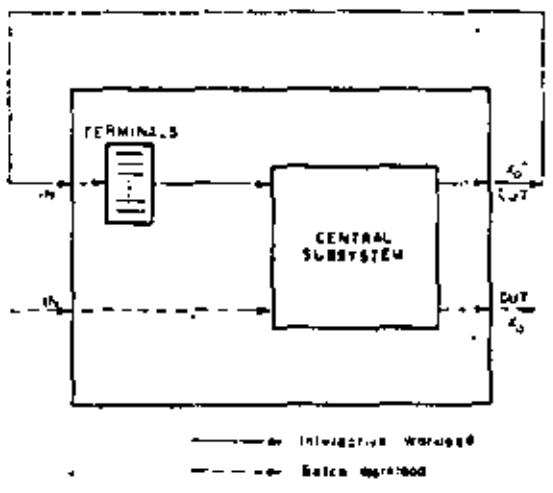


Fig. 1.5.-

donde N_1, es el punto de corte de la asíntota inclinada con el eje de abscisas.

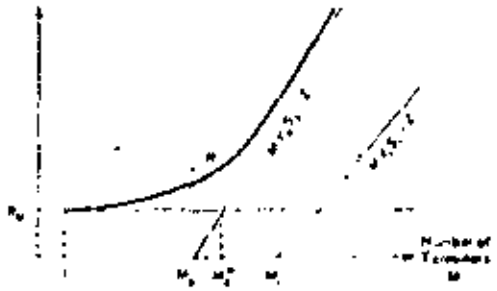


Figura 1.7.-

Ejemplo 2.- Consideremos el sistema de la figura 1.8.

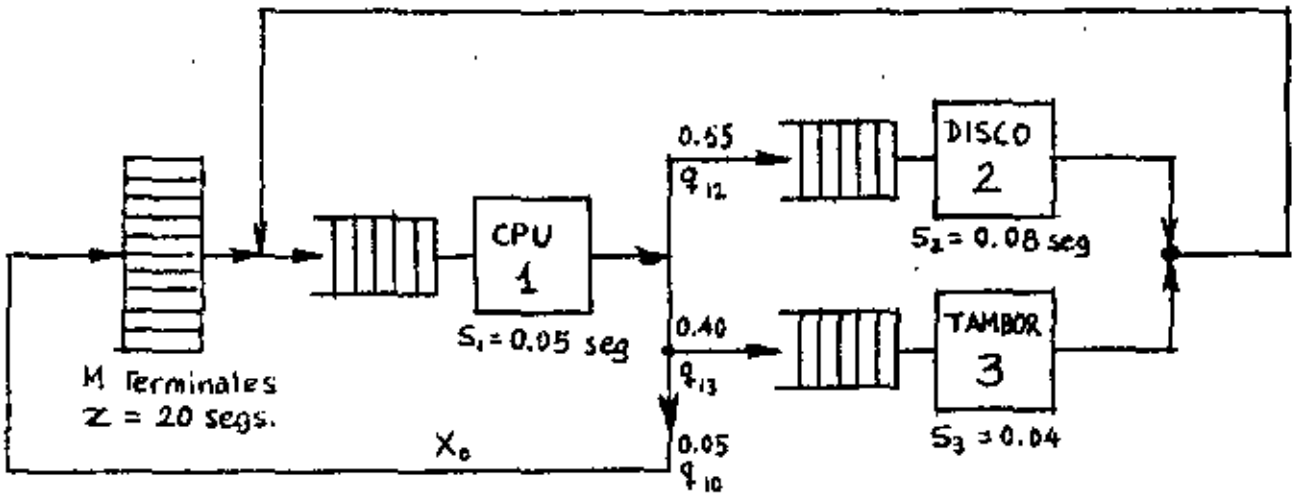


Figura 1.8.-

Las ecuaciones de las razones de visita son

$$V_o = 1 = 0,05\ V_1$$
$$V_1 = V_o + V_2 + V_3$$
$$V_2 = 0,55\ V_1$$
$$V_3 = 0,40\ V_1$$

cuya solución es

$$V_1 = 20 \qquad V_2 = 11 \qquad V_3 = 8$$

Los productos $V_i\ S_i$ son

que ignora los retrasos en las colas indica el menor valor posible del tiempo de respuesta. De hecho R_o es el tiempo de respuesta cuando N=1; lo cual implica $X_o=1/R_o$ cuando N=1. Estas propiedades se resumen en la figura 1.6

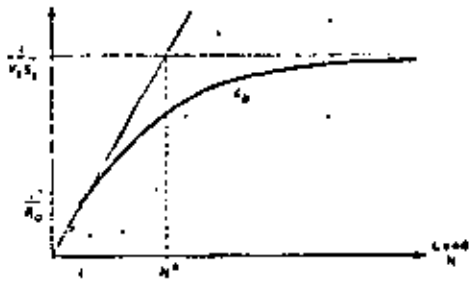


Figura 1.6.-

El punto de ruptura de las dos asíntotas, N^*, es decir el número de trabajos a partir del cual ya no crece el "throughput", es

$$\frac{1}{R_o} N^* = \frac{1}{V_b S_b}$$

de donde

$$N^* = \frac{V_1 S_1 + \ldots + V_K S_K}{V_b S_b}$$

Estos resultados pueden extenderse a sistemas con terminales como el de la figura 1.3. Para M terminales y tiempo de pensar Z, el tiempo de respuesta es $R = M/X_o - Z$. Cuando M=1, R debe ser R_o. Puesto que X_o no puede exceder $1/V_b S_b$, tenemos

$$R \geq MV_b S_b - Z \geq MV_i S_i - Z, \quad i = 1, \ldots, K.$$

Cuando M crece, R se aproxima a la asíntota $MV_b S_b - Z$ tal como muestra la figura 1.7. El punto de intersección de ambas asíntotas corresponde a

$$R_o = M_b^* V_b S_b - Z$$

de donde

$$M_b^* = \frac{R_o + Z}{V_b S_b} = N^* + M_b$$

$V_1 S_1 = 20 \times 0,05 = 1$ segs.

$V_2 S_2 = 11 \times 0,08 = 0,88$ segs.

$V_3 S_3 = 8 \times 0,04 = 0,32$ segs.

de donde el tiempo de respuesta mínimo es

$R_o = 1+0,88+0,32 = 2,2$ seg.

El máximo $V_i S_i$, es para la CPU, por lo tanto b=1, y el cuello de botella es la CPU.

El número de terminales en el periodo de pensar en saturación es

$$N_1 = \frac{Z}{V_1 S_1} = \frac{20}{1} = 20 \text{ terminales,}$$

el número de trabajos que saturan el sistema es

$$N^* = \frac{R_o}{V_1 S_1} = 2,2 \text{ trabajos,}$$

y el número total de terminales para saturar el sistema es

$$N_1^* = 20+2,2 = 22,2 \text{ terminales.}$$

Si en este sistema se midieron 0,715 trabajos/seg. y un tiempo medio de respuesta de 5,2 segundos ¿Cuál es el número de usuarios conectados durante el periodo de observación?

$$N = (R+X)/X_o = (5,2+20)/0,715 = 18 \text{ terminales.}$$

¿ Sería posible obtener un tiempo de respuesta de 8 segundos cuando hay conectados 30 usuarios ? Si no lo fuera ¿ Qué aumento de velocidad de la CPU sería necesario ?

Por el análisis asintótico, para N = 30

$R > 30.1-20 = 10$ segundos $>$ 8 segundos por lo tanto no es posible.

Si S'_1 es el tiempo de servicio de la nueva CPU, necesitamos que

$$N V_1 S'_1 - Z \leq 8$$

de donde

$$S'_1 \leq \frac{8+20}{30.20} = 0,047 \text{ segs.}$$

lo que se obtiene con un factor de aceleración de

$$\frac{S_1}{S_1} \quad \frac{0,05}{0,047} = 1.07 \text{ segs.}$$

En el nuevo sistema el cuello de botella sigue siendo la CPU, pues $V_1 S_1 = 20 \times 0,047 = 0,93$ segs. , por lo que nuestro análisis asintótico sigue siendo correcto.

¿ Sería posible obtener un tiempo de respuesta de 10 segundos con 50 usuarios conectados ? Si no lo fuera ¿ que aumento de de velocidad de la CPU sería necesario ?

Procediendo como antes, para M = 50

$$R \geq 50.1-20 = 30 \text{ segs.} > 10 \text{ segs.}$$

$$S'_1 = \frac{10+20}{50.20} = 0,03 \text{ segs.}$$

pero ahora $V_2 S_2 > V_1 S'_1$ por lo que el cuello de botella pasa a ser el disco. Para M=50.

$$R \geq 50.0,88-20 = 24 \text{ segs.} > 10 \text{ segs.}$$

por lo que no pueden conseguirse las especificaciones de trabajo solicitadas.

2. MONITORES.

2.1. Introducción.

Monitor es una palabra que define un instrumento de medida y control : en un sistema informático se utilizan monitores hardware y software para obtener medidas que de alguna manera den información sobre el comportamiento del sistema.

Dicha información puede servir para :

- asegurar una correcta operación del sistema
- aislar las fuentes de problemas actuales y futuros.
- conocer el comportamiento del sistema frente a las necesidades de servicio de los usuarios.

2.2. Conceptos de medida.

Aunque es posible conseguir programas que, ejecutándose al mismo nivel que los normales del sistema, midan algunas propiedades internas, es necesario proporcionar puntos de observación adecuados en él. Un monitor registra el comportamiento en estos puntos, detectando los cambios de estado del sistema : señalando o bien el principio o bien el final de un periodo de cierta actividad (o inactividad) en un componente (hardware, software, proceso) del sistema (tabla 2.1). A un tal cambio de estado se le llama también acontecimiento. Un acontecimiento software es un suceso relacionado con la función del programa : ocurre cuando el programa alcanza un cierto punto lógico. Un acontecimiento hardware está generado por una operación hardware.

T A B L A 2.1.-

Estados de las memorias por distintos problemas de medida (SVOB 76)

Medida	Nivel	Componentes	Estado de la memoria
utilización de la UCP y del canal	procesadores, memorias	UCP; canales	bit de estado espera/ocupado de la UCP bit de estado supervisor/problema de la UCP bits de "canal ocupado" bits de "interrupción de canal"
utilización del disco	procesadores, memorias	canal disco	bit de canal ocupado bit de interrupción de canal bit de dispositivo ocupado bit de dispositivo leyendo bit de dispositivo escribiendo bit de búsqueda (seek) bits de dirección al cilindro y registro
evaluación del tiempo de respuesta del sistema	sistema operativo	planificador	palabras de memoria representando longitudes de las colas del sistema y estado de trabajos individuales en el sistema

.../...

sigue tabla 2.1.-

eficiencia del programa	sistema operativo	rutinas del sistema, programas de utilidad, procesadores de lenguaje	bit de estado wait/busy bits del contador de instrucciones
paginación	sistema operativo	manager de memoria páginas virtuales	registros asociativos tablas de paginación bits de referencias válidas/ invalidas. bits de referencias en memoria
utilización del código máquina	procesador de instrucciones	unidad de control unidad aritmética y lógica	bits de registro de código de operación bits de control de ejecución de instrucciones.
utilización de registros	transferencia de registros	registros de trabajo y control	bit de registro válido/inválido bit de registro asignado/libre bit de control de trasferencia de registros

La forma de instrumentación depende del tipo de medida y de la información que se desea. Las categorías de medida pueden describirse como sigue :

a) Traza. Una actividad queda descrita por una secuencia de pares de instantes de entrada y de salida de la actividad. Esta información podemos obtenerla a partir de la traza de los acontecimientos registrados secuencialmente con mención del instante en que se han producido.

b) Actividad relativa es la relación entre el tiempo total en una actividad determinada y el tiempo total transcurrido. Por ejemplo

- medida de la utilización de la UCP o del canal
- medida del solapamiento entre la UCP y el canal

c) Frecuencia de acontecimiento. La frecuencia de entrada en un estado determinado se mide contando los acontecimientos que representan la entrada en ese estado y dividiendo por el tiempo total transcurrido. Por ejemplo

- frecuencia de fallo de página
- frecuencia de llamadas al supervisor
- frecuencia de acceso a un disco
- frecuencia de llamada a un procedimiento

d) Distribución de intervalos de actividad. Este tipo de medida pretende tabular los tiempos de permanencia del sistema en una actividad determinada. Por ejemplo

- la medida de la distribución de los tiempos de

UCP entre diversos procesos (incluídos los del sistema operativo) presentes en el intervalo de medida.

En los casos mencionados, se supone que es posible detectar los cambios apropiados en el estado del sistema. En el tercero, no es necesaria la detección exacta del cambio de estado : se puede tomar muestras a intervalos de tiempo independientes de la actividad medida, acumulándose el número de éxitos (observación que encuentra el sistema en el estado que se evalúa) y fracasos (observación que no encuentra el sistema en el estado que se evalúa) y luego calcular el tanto por ciento de tiempo en el estado en cuestión dividiendo el número de éxitos por el total de observaciones (éxitos+fracasos) efectuadas.

2.3. Instrumentos de medida.

Los instrumentos para evaluar el comportamiento de un sistema informático se denominan monitores.

Un monitor lleva consigo el concepto de una observación continua del comportamiento del sistema observado, por oposición a medida que induce a pensar en experimentos conducentes a analizar la influencia de una variable determinada sobre otras, lo cual requiere que las circunstancias sean reproducibles (cosa poco frecuente en sistemas informáticos).

Independientemente de la tecnología empleada en construirlos, los monitores presentan generalmente la estructura de la figura 2.1.

La conexión del monitor al sistema a medir se efectúa mediante una instrumentación adecuada que permita la observación de un conjunto específico de actividades del sistema.

El elemento de proceso selecciona un subconjunto de actividades observables para su seguimiento.

El elemento de proceso interroga el estado del sistema a medir y según las opciones de medida especificadas recoge y prepara los datos pertinentes para que el elemento de registro pueda llevar a cabo su misión.

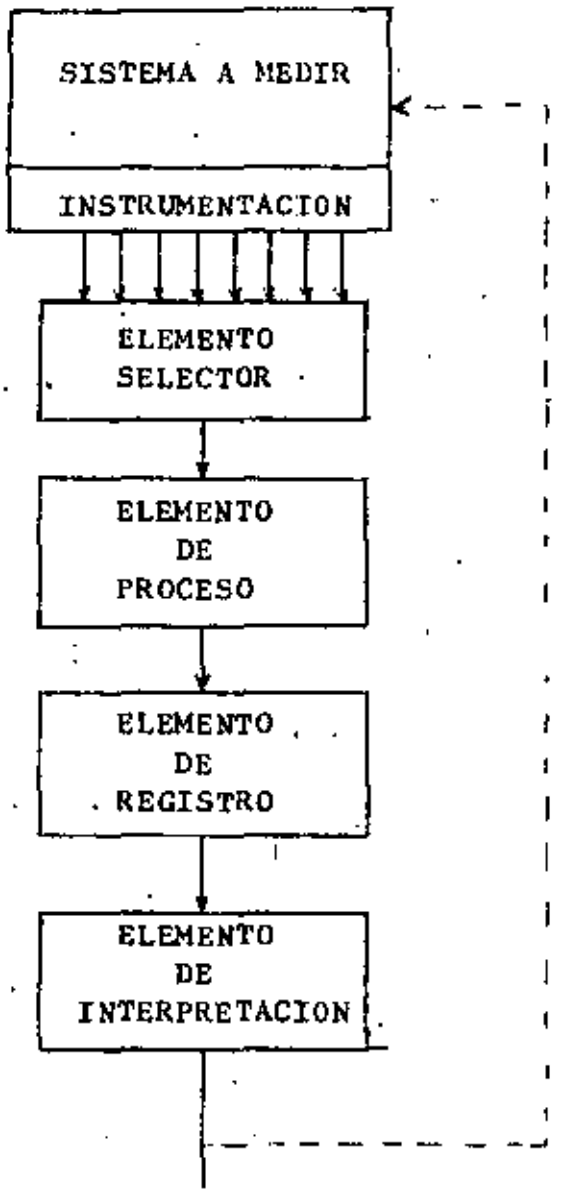


Figura 2.1.-

El elemento de interpretación analiza y sintetiza los datos acumulados por el elemento de registro y presenta los resultados de una forma inteligible. Normalmente esta interpretación es un proceso post-medida, pero puede llevarse a cabo en paralelo con ella. En este caso pueden utilizarse las medidas efectuadas en tiempo real (que presentan interesantes problemas estadísticos de estimación) para utilizarlas dinámicamente en el control del comportamiento del sistema.

2.3.1. Monitores software.

Un instrumento de medida software es un programa especial que es ejecutado por el sistema que se mide : por tanto compite activamente por los recursos del sistema y con los programas que constituyen la carga de trabajos del sistema. Claramente, la presencia del instrumento afecta la eficacia del sistema medido, en forma de carga adicional de trabajo, memoria ocupada y a veces canales y dispositivos de entrada/salida.

La implementación de un monitor software según el esquema general de la figura 2.1. es la siguiente :

Instrumentación	puntos de bifurcación muestreo interpretación
Selector	conmutador software hardware especial
Procesador	software hardware especial
Almacenamiento	memoria registros memoria externa

Analizador	programa software batch o tiempo real
Salida de datos	listado, pantalla.

Un primer método de recoger información se consigue insertando puntos de bifurcación en lugares adecuados del software del sistema que transfieren control al registrador (DENI 69, LONE 70, BAUR 73).

El muestreo se implementa a base de una rutina que genera interrupciones de tiempo y permite al registrador acceder a una "instantanea" del estado del sistema en el instante de la interrupción (HOLW 71, KOLE 71, BAUR 73, SVOB 73). La exactitud de los resultados depende del número de observaciones realizadas y de las caracteristicas del sistema a medir, por lo que una carga de trabajo no estacionaria puede proporcionar medidas poco fiables a menos que no se determinen intervalos adecuados de muestreo. Otro factor que afecta la precisión de los resultados de un monitor de muestreo software es la prioridad. A más alta prioridad, menor es la probabilidad de que quede bloqueado por otra tarea más prioritaria y por tanto mejores son los resultados. En (ROSE 78) se menciona que el monitor software para OS/370 ejecutándose como tarea más prioritaria da un 3% de discordancia respecto a las lecturas del monitor hardware. Para sistemas 370/VS, ejecutándose con la segunda prioridad más alta (la primera es la de operaciones de paginación), da una discordancia del 5%.

El método de interceptación utiliza las interrupciones sincronas que transfieren el control entre los niveles de protección de la jerarquía del sistema operativo (por ejemplo, una llamada al

supervisor). El conocimiento del ejecutivo del sistema a medir es indispensable para implementar este método.

En general, un instrumento software se implementa en distintos lenguajes. Por razones de eficiencia y por la necesidad de llegar a niveles hardware, la programación se hace en lenguaje máquina. Por otra parte, la microprogramación facilita la implementación de monitores muy potentes : las microinstrucciones acceden a muchos indicadors hardware que normalmente son inaccesibles para el lenguaje máquina implantado al nivel inmediatamente superior al de la microprogramación (PART 76, ROBE 72, SAAL 72).

2.3.2. Monitores hardware.

Un instrumento hardware es un dispositivo (externo o interno) que detecta señales electrónicas en los ciruitos del sistema. Las posibles implementaciones según el esquema de la figura 2.1., son :

Instrumentación	sondas electrónicas interfases cableadas al sistema
Selector	tablero de conexiones memoria asociativa software
Procesador	tablero de conexiones software
Almacenamiento	registros hardware memoria asociada RAM dispositivos de memoria externa
Analizador	post-proceso (batch) tiempo real
Salida de datos	impresora y/o pantalla

La función del tablero de conexiones es dar la posibilidad de efectuar operaciones booleanas sobre los bits de estado. Un estado puede interpretarse como una determinada combinación de valores on/off de los bits detectados en diferentes partes del hardware. Una serie de contadores completan esta parte del instrumento.

2.3.3. Comparación de ambos monitores.

2.3.3.1. Potencia de un monitor.

Definamos en primer lugar las magnitudes que nos permiten caracterizar la calidad de un monitor, y que son:

a) Dominio del monitor es la clase de actividades teoricamente observables con una tecnica de medida determinada. Observemos que la instrumentación facilita la aplicación de la técnica de medida a un problema concreto seleccionando del conjunto de acontecimientos medibles aquellos que son indicadores de las actividades específicas dentro del dominio del monitor.

b) Frecuencia de entrada es la máxima frecuencia a la que se pueden reconocer y registrar los acontecimientos.

c) Anchura de entrada es el número de bits de información de entrada que el monitor puede extraer y procesar cuando se produce un acontecimiento.

d) Capacidad de registro es el número de elementos de memoria disponibles para almacenar la información extraida. Es un atributo del registrador. Determina la cantidad de información que puede rete-

nerse para un proceso posterior. La actividad relativa puede medirse mediante un solo contador. La duración media de una actividad requiere dos valores: la duración acumulada de esa actividad y el número total de veces que se ha efectuado dicha actividad. Los datos necesarios para determinar la distribución de una variable puede obtenerse a partir de una traza de acontecimientos (que puede requerir millones de palabras) o a partir de los registros acumulados en una tabla de contadores (en una ocupación de memoria inferior).

e) Resolución del monitor es la del tiempo del reloj que sirve para temporizar la información. Este factor limita la precisión alcanzable en las medidas efectuadas en base al tiempo.

2.3.3.2. Limitaciones de los monitores hardware y software.

a) Dominio del monitor. Los monitores software tienen el acceso controlado a aquellos elementos de memoria que pueden leerse mediante una instrucción máquina. Pueden observar acontecimientos relacionados al hardware solo si están acompañados por una transferencia de control a una instrucción en una dirección conocida o si almacenan información identificadora que pueda ser investigada posteriormente. Por otro lado, ciertas informaciones tales como nombres de programas o de datos pueden extraerse sólo mediante un monitor software.

Los monitores hardware pueden observar cualquier elemento de memoria, siempre que no sea necesaria ninguna señal generada por el sistema para recuperar la información del estado. Un monitor hardware no puede observar directamente el contenido de una memoria de acceso directo; tiene acceso a la información almacenada sólo cuando entra o sale de la misma. Estrictamente sucede lo mismo con un monitor

software, pero este puede decidir que datos deben traerse para comprobarlos o procesarlos, mientras que el monitor hardware es sólo un observador pasivo que no tiene control sobre el sistema de memoria. Un monitor hardware puede observar acontecimientos relacionados al software solo cuando van acompañados por una transferencia de control a una dirección absoluta fija o cuando se ponen en marcha por ejecución de instrucciones especiales como llamadas al supervisor.

b) Frecuencia de entrada. Los monitores hardware tienen la capacidad de resolver acontecimientos a frecuencias elevadas (de 10 a 25 Mlz) y es una magnitud absoluta determinada por la velocidad de las sondas y de la lógica del monitor. El monitor software no puede descender más allá de su acontecimiento elemental: la ejecución de una instrucción. Por lo tanto, la máxima frecuencia de entrada del monitor software viene fijada por la máxima frecuencia de ejecución de las instrucciones descontadas las necesarias para la ejecución del monitor. Es pues una magnitud relativa.

c) Anchura de entrada. Un monitor software puede detectar los acontecimientos solo secuencialmente. Sin embargo, es capaz de parar la CPU en el sentido que interrumpe el proceso normal durante el tiempo necesario para extraer la información necesaria. La anchura de entrada es pues teoricamente ilimitada, con la única restricción del " overhead ". El monitor hardware permite la detección de acontecimientos en paralelo pero su anchura está limitada por el número de sondas disponibles.

d) Capacidad de registro. El elemento de registro primario de muchos monitores hardware es solo un conjunto de contadores

(del orden de las decenas). La capacidad de registro de los monitores hardware que usan una memoria de acceso directo es comparable a la de los monitores software. Si existe una memoria secundaria esta capacidad puede considerarse ilimitada.

e) Resolución. Los monitores hardware miden a intervalos de tiempo por muestreo del estado del sistema a frecuencia muy alta. Esta frecuencia se deduce del reloj que rige al monitor. Los monitores software usan el temporizador del sistema medido. Los sistemas antiguos tenían una resolución muy baja (del orden de los milisegundos) mientras que los más modernos tienen temporizadores más rápidos (del orden de los microsegundos).

En la tabla 2.2. se resumen los factores positivos y negativos de los monitores hardware y sftware.

TABLA 2.2.

Comparación instrumentos hardware/instrumentos software.

Factor	Instrumento software	Factor	Instrumento hardware
negativo	generalmente no portable	positivo	portables
negativo	interfiere con las operaciones del sistema medido	positivo	no utiliza los recursos del sistema.
positivo	la actividad del sistema puede (gene-	negativo	ha de registrarse la información mientras

Factor	Instrumento software	Factor	Instrumento hardware
	ralmente) interrumpirse tanto tiempo como sea preciso, a fin de dar tiempo a registrar los datos.		el sistema funciona a máxima velocidad.
positivo	puede tratar variables descriptivas (nombre del programa).	negativo	no puede hacerlo
positivo	puede acceder a cualquier dato en memoria.	negativo	generalmente no puede registrar información de una memoria, ya que sólo registra la información que pasa a través de los circuitos.
positivo	puede registrar el uso de los componentes software independientemente de su posición en memoria.	negativo	sólo puede hacerlo si las direcciones de estos componentes son fijas.
negativo	no puede observar acontecimientos en dispositivos E/S.	positivo	diferentes unidades hardware pueden registrarse simultáneamente.

Factor	Instrumento software	Factor	Instrumento hardware
negativo	la resolución del reloj registrador de tiempos no es suficiente.	positivo	reloj de alta resolución.
negativo	debe existir la sincronización entre el reloj y el acontecimiento registrado.	positivo	opera asíncronicamente.
negativo	un fallo del sistema con el proceso de medida, lo aborta	positivo	continúa aún si hay un fallo del sistema
negativo	los errores lógicos en el monitor pueden hacer caer el sistema.	negativo	las sondas pueden producir fallos en el hardware.
positivo	el monitor puede controlarse con el sistema.	negativo	el sistema medido no tiene control sobre el monitor.
negativo	requiere un íntimo conocimiento del sistema operativo.	negativo	necesita un conocimiento íntimo del hardware.

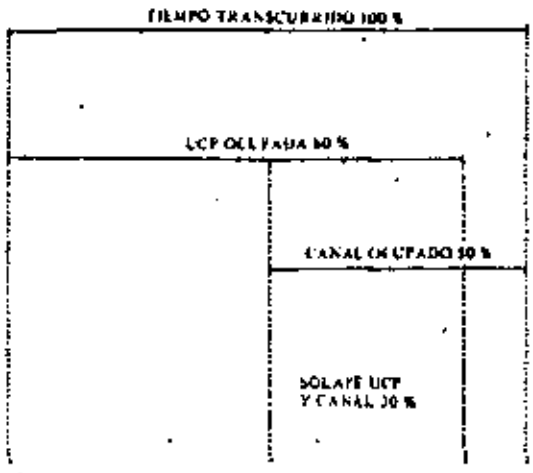
2.4. Proceso de los datos obtenidos.

Una vez efectuada la selección de los datos, éstos han de procesarse para su interpretación y análisis. Tradicionalmente, los resultados de las medidas se presentanen forma de tablas e histogramas.

Si queremos relacionar nuestras medidas con otro tipo de parámetros como por ejemplo la carga de trabajos, habremos de considerar más de un factor para interpretar los resultados. Por ejemplo, podemos combinar la actividad de la CPU con la actividad de entrada/salida a fin de encontrar una medida más útil: el grado de solape entre la entrad/salida y la CPU. Las técnicas de representación gráfica de Gantt y Kiviat ayudan a representar los perfiles de utilización del sistema, relacionando las utilizaciones de varios recursos del sistema.

2.4.1. Gráficas de Gantt.

Un ejemplo del uso de una carta de Gantt es la figura 2.2 que sigue:



Cada barra horizontal representa la utilización de un recurso individual. Las partes solapadas indican la cantidad de solapamiento en las actividades de los componentes asincronos del sistema. Un aumento en un factor de 2 en la velocidad de la CPU afecta al gráfico en la forma en que se muestra en la figura 2.3 (Ha de tenerse en cuenta que se admite que la relación entre la parte solapada y la parte de CPU no solapada no varía). La mejora en el rendimiento del sistema se define como la razón entre el tiempo total transcurrido ("elapsed time") en el sistema original y en el mejorado. En el ejemplo, la razón es de 1.33, es decir, un 30%.

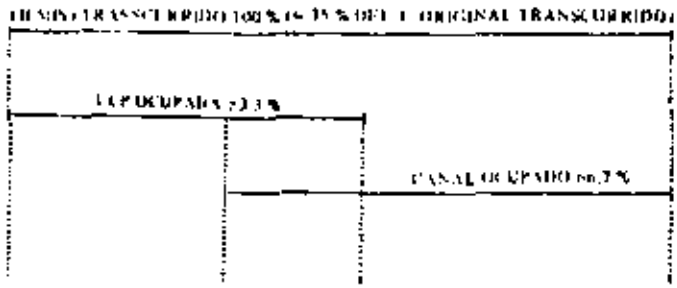
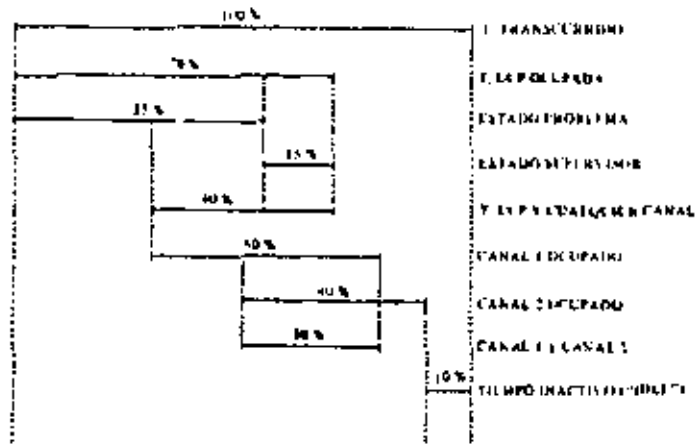


Fig. 2.3

Las reglas usadas para convertir estos perfiles son rudimentarias; deben tomarse sólo como una cruda aproximación a análisis más detallados. Una gráfica de Gantt más complicada se muestra en la figura 2.4.

Fig. 2.4.



2.4.2Gráficas de Kiviat

Otra forma de representación es la utilización de gráficos circulares, llamados de Kiviat. La figura 2.5 puede representarse circularmente asignando los tantos por ciento de cada variable respecto al total de tiempo transcurrido en el periodo de medida a cada eje del gráfico.

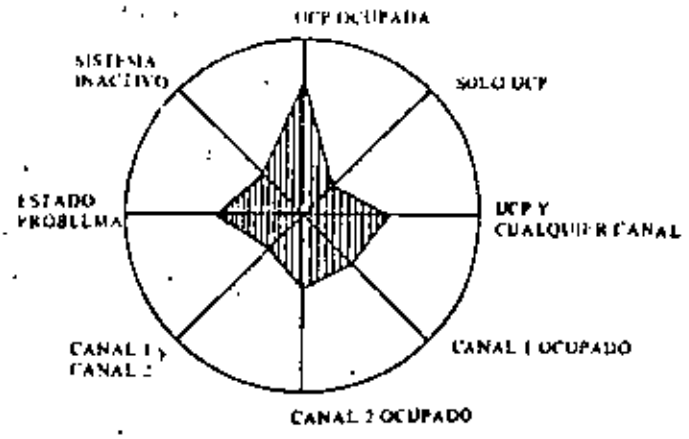


Fig. 2.5

En general se acostumbra a representar las variables "favorables" en los ejes impares y las "desfavorables" en los ejes pares: un sistema ideal tendría el aspecto de una estrella (figura 2.6)

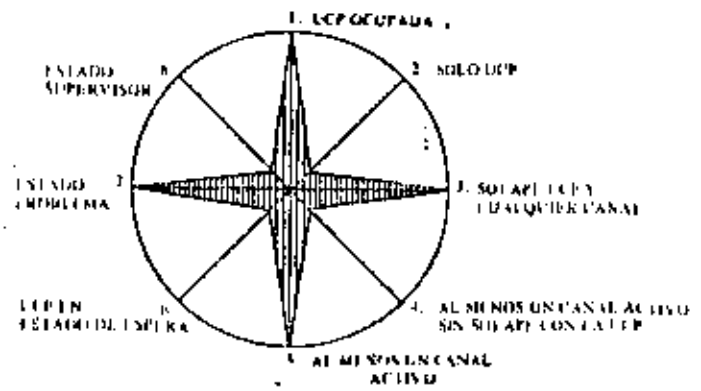


Fig. 2.6

Un sitema limitado por la CPU (CPU bound) tiene una típica gráfica de "vela de barco". Un sistema limitado por la entrada/salida (I/O bound) aparece como la " quilla de un barco " (figuras 2.7.a y 2.7.b).

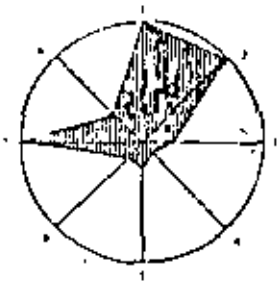


Fig. 2.7.a

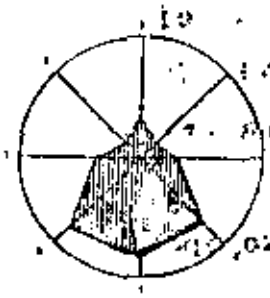


Fig. 2.7.b

El"límite de eficiencia" producido en un sitema con una excesiva paginación se muestra en la figura 2.8.

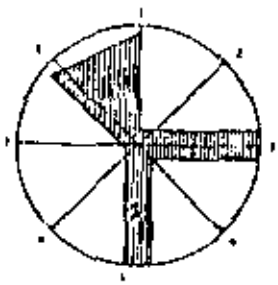


Fig. 2.8.

Los gráficos de Kiviat no pueden tomarse como índices absolutos de un sistema. Por ejemplo, puede discutirse el sentido de "malo" de una figura del tipo " vela de barco " en un entorno en que se procesa una carga de tipo científico.

Una ventaja inmediata es, por tanto, el dar idea no tanto de un índice de utilización como de los cambios que pueden producirse.

En un sistema con cargas estacionarias, las medidas han de dar figuras similares. Un cambio radical en esta figura no indica nada más que un cambio en el sistema (carga, configuración) que afectará a su grado de utilización y por tanto a su eficiencia.

Nótese que en la figura 2.5 se ha abandonado deliberadamente el concepto de ejo favorable para concentrarse en las medidas de las actividades de los canales. Este aspecto es también importante resaltarlo para estudiar problemas tales como:

a) el poco solapamiento entre canales.

b) el poco equilibrio entre las cargas de los canales.

c) la alta utilización del canal.

d) la alta espera de la CPU sin ningún canal activo.

e) el tiempo de "solo CPU" demasiado alto.

Se puede siempre escoger el subconjunto de medidas que den más detalle sobre el problema en concreto.

Finalmente puede usarse la estadística para construir un modelo de regresión que, a partir de las medidas obtenidas, nos de unas ecuaciones (una para cada medida) que expresen la eficiencia del sistema (E_i) en función de las variables independientes (Z_i) medidas y un conjunto de parámetros de regresión junto con un factor de error e_i (W A L D 73).

$$E_i = A_{io} + \sum_{j=1}^{n} A_{ij} Z_j + \sum_{k=1}^{n} \sum_{l=1}^{n} A_{ikl} Z_k Z_l + e_i$$

2.5. Monitores suministrados por los constructores (ROSE 78).

Generalmente, los monitores hardware son independientes del sistema; los monitores que los constructores pueden suministrar son en todo caso monitores software.

Serie CDC 6600:

No comercializado, aunque el diseño e implementacion de un monitor software parece estar al alcance de un buen programador de sistemas.

Serie Honeywell 6000:

Nombre: "Generalized Monitor Facility" (GMP)

Tipo: Interceptación de acontecimientos

Medidas:

- Monitor de memoria, de memoria masiva y de CPU
  - + número de trabajos en memoria
  - + utilización de la CPU
  - + tiempo de servicio de la CPU
  - + número de trabajos en cola de la CPU

- Monitor del canal
  - + tiempo de servicio
  - + longitud de las colas frente al canal
  - + tiempos de espera

- Monitor de comunicaciones
  - + actividad del terminal
  - + duración de las sesiones
  - + longitud de los mensajes
  - + tiempo en que el usuario "piensa" antes de emitir un mensaje

+ tiempos de respuesta para el terminal

Serie IBM 370

Nombre: Resource Measurement Facility (RMF) para OS/VS2 MVS

Tipo: muestreo e interceptación de acontecimientos

Medidas:

+ número de usuarios "batch" y TSO
+ utilización de la CPU
+ utilización del canal
+ actividad del canal por segundo
+ tiempo medio de servicio por canal
+ actividad de dispositivos
+ tiempo medio de servicios por dispositivo

Nombre: "Systems Management Facility" (SMF)

Tipo: interceptación de acontecimientos

Medidas:

+ tiempo CPU usado por cada programa
+ número de faltas de página por programa
+ número de operaciones por programa

Nota: el SMF está diseñado para funcionar como un sistema de contabilidad (accounting); es decir, entra y registra datos entre paso y paso de trabajo y a la terminación de un trabajo.

Nombre: "Generalized Trace Facility" (GTF)

Tipo: interceptación de acontecimientos

Medidas:

+ llamadas al supervisor
+ actividad de entrada/salida
+ paginación
+ ocupación de memoria

Serie Univac 1110

Nombre: Software Instrumentation Package (SIP)

Tipo: Interceptación de acontecimientos

Medidas:

+ utilización
+ actividad de entrada/salida
+ número de trabajos "batch", tiempo compartido
+ utilización del canal
+ interrupciones de la CPU
+ número de palabras transferidas por el canal

3. MODELOS DE SISTEMAS INFORMATICOS.

Existen numerosas circunstancias en que la evaluación del rendimiento de un sistema informático debe efectuarse mediante la construcción de un modelo. Ello será así siempre que alguno de los elementos haedware o software que componen el sistema no exista, como son los casos de instalación de un nuevo sistema o de reconfiguración (cambio de configuración hardware, introdución de una nueva aplicación, etc.) de uno existente.

3.1. Introducción a la teoría de colas.

¿Qué informático no se ha encontrado alguna vez esperando a que su listado saliera por la impresora ocupada en emitir los interminables resultados de una explotación o de otro programador?

¿Quién que se haya sentado en un terminal para trabajar en tiempo compartido en un ordenador no ha tenido que esperar a que su programa entrara en memoria o utilizara suficientemente la CPU?

¿Qué jefe de explotación no se ha visto en la necesidad de hacer esperar programas que no podían ejecutarse por falta de cintas o discos?

Todas estas esperas provocan colas dentro o fuera del sistema informático y aun antes de que existieran los ordenadores se

había desarrollado la teoría de colas para intertar analizar su comportamiento.

La teoría de colas desarrollada con anterioridad y que en sus primeros niveles no es más que un caso particular de un proceso de Markov (que es aquel en que su estado actual depende sólo del anterior y de las entradas que ha recibido y no de los demás estados precedentes; es decir es un proceso sin memoria) se ha utilizado en la construcción de modelos de sistemas informáticos debido a que realmente en ellos aparecen numerosos subsistemas en que para obtener servicio de un elemento es preciso colocarse en cola para poder alcanzarlo, como es el caso de la CPU, discos, canales, etc.

3.1.1. Componentes de un modelo de colas.

Los componentes básicos de un modelo de colas son las estaciones de servicio, las colas y los manantiales. Las estaciones de servicio se usan generalmente para modelizar los recursos solicitados por los trabajos que sometemos a un ordenador. Los trabajos se generan en los manantiales o existen en el modelo desde su creación. Cada estación de servicio puede atender sólo un número limitado de trabajos al mismo tiempo, lo que se conoce como el número de canales de la estación de servicio. Estos trabajos cuando encuentran la estación de servicio ocupada esperan hasta que les llega el turno. Cada estación de servicio tiene por lo menos una cola y con frecuencia el concepto estación engloba también la cola. Un trabajo generalmente requiere la atención de una estación

de servicio durante un cierto tiempo denominado tiempo de servicio y entra en la estación de servicio en un instante denominado tiempo de llegada del trabajo.

3.1.1.1. Características del manantial.

a) Su tipo finito o infinito; si un manantial es finito, el número máximo de trabajos generados por él, que un modelo puede contener, tiene una cota finita.

b) La distribución de los intervalos entre la generación de dos trabajos sucesivos.

c) Las demandas de cada trabajo de los servicios de cada estación de servicio del modelo; si las demandas de un determinado tipo de servicio están idénticamente distribuidas para todos los trabajos, es natural considerarlas como una característica de la correspondiente estación de servicio en vez de como una del manantial; sin embargo, puesto que representan demandas de recursos hechas por los trabajos, es más correcto pensar en ellas como características del manantial.

3.1.1.2. Características de la estación de servicio.

a) El número y la capacidad de sus colas; la capacidad de una cola es el máximo número de trabajos que puede contener.

b) El número de canales de servicio de cada una.

c) La velocidad de los servidores; es decir el número medio de trabajos que puede atender por unidad de tiempo; también se acostumbra a utilizar su inversa, o sea el tiempo medio de servicio; cuando la velocidad de servicio de la estación es fija y las demandas de servicio están idénticamente distribuidas para todos los trabajos, podemos considerar la distribución de los tiempos de servicio entre las características de la estación de servicio.

d) La disciplina de servicio, que especifica bajo qué condiciones las estaciones de servicio terminan su servicio a un trabajo que debe ser servido a partir de la cola de la estación de servicio, y lo que hace un trabajo servido incompletamente.

3.1.1.3. Interconexiones.

El modelo de colas se completa con las interconexiones entre las estaciones de servicio que especifican los caminos existentes entre ellas.

3.1.2. Procesos de nacimiento-muerte.

Esta teoría general, que vamos a describir a continuación, describe una amplia clase de sistemas de colas de los que haremos uso al establecer modelos de sistemas informáticos.

3.1.2.1. Ecuación de Kolmogorov.

Consideremos un sistema caracterizado por el número de elementos k que hay en el mismo. Estos elementos pueden llegar nacer al sistema con frecuencia λ_k que podemos considerar dependiente de k y pueden salir (morir) del sistema con frecuencia μ_k que podemos considerar dependiente de k. Tanto los nacimientos como las muertes solo pueden producirse de una en una. Admitamos además que el sistema llega a un estado estacionario donde son conocidas las posibilidades p_k de que haya k elementos en el sistema.

Evidentemente debe cumplirse que

$$\sum_k p_k = 1,$$

que

$$\lambda_{-1} = \lambda_{-2} = \ldots = 0$$

$$\mu_0 = \mu_{-1} = \ldots = 0$$

$$p_{-1} = p_{-2} = \ldots = 0$$

Al estado E_k (el sistema contiene k elementos) se produce un flujo de entrada desde los estados E_{k+1} y E_{k-1}, que dan una frecuencia global de llegada de

$$\lambda_{k-1}\, p_{k-1} + \mu_{k+1}\, p_{k+1}$$

Del estado E_k se produce un flujo de salida hacia los estados E_{k+1} y E_{k-1}, que dan una frecuencia global de salida de

$$(\lambda_k + \mu_k)\ p_k$$

Si el sistema está en equilibrio ambas frecuencias deben ser iguales, de donde se deduce inmediatamente

$$\lambda_{k-1}\ p_{k-1} + \mu_{k+1}\ p_{k+1} = (\lambda_k + \mu_k)\ p_k$$

de donde

$$0 = -(\lambda_k + \mu_k) p_k + \lambda_{k-1}\ p_{k-1} + \mu_{k+1}\ p_{k+1}$$

teniendo en cuenta las restricciones establecidas antes, tenemos para k = 0

$$0 = -\lambda_0\ p_0 + \mu_1\ p_1$$

Estas dos últimas ecuaciones se conocen como ecuaciones de Kolmogovov, que nos permiten deducir las probabilidades de los estados junto con la ecuación de que la suma de las probabilidades es igual a 1. Tenemos

$$p_1 = p_0\ \frac{\lambda_0}{\mu_1}$$

Sustituyendo en la ecuación de Kolmogovov para k = 1, tenemos

$$0 = -(\lambda_1 + \mu_1)\ \frac{\lambda_0}{\mu_1}\ p_0 + \lambda_0 p_0 + \mu_2 p_2$$

de donde

$$p_2 = p_0 \frac{\lambda_0\lambda_1}{\mu_1\mu_2} = p_1 \frac{\lambda_1}{\mu_2}$$

y en general

$$p_k = p_0 \prod_{i=0}^{k-1} \frac{\lambda_i}{\mu_{i+1}} = p_{k-1} \frac{\lambda_{k-1}}{\mu_k}$$

Sustituyendo en la ecuación de la suma de las probabilidades, podemos deducir p_0

$$p_0 = \frac{1}{1 + \sum_{k=1}^{\infty} \prod_{i=0}^{k-1} \frac{\lambda_i}{\mu_{i+1}}}$$

Estos plantcos nos lleva a procesos de Markov tanto de entrada como de salida del sistema.

3.1.2.2. Cola M/M/1.

En este tipo de cola, el más sencillo, se supone que tanto las frecuencias de llegada como de salida son independientes del estado

$$\lambda_k = \lambda \quad , \quad k = 0, 1, 2, \ldots$$

$$\mu_k = \mu \quad , \quad k = 1, 2, 3, \ldots$$

Las transiciones las podemos representar tal como en la figura 3.1.

Aplicando las condiciones impuestas a este caso a las ecuaciones de Kolmogovov, encontramos

$$p_k = p_0 \prod_{i=0}^{k-1} \frac{\lambda}{\mu} = p_0 \left(\frac{\lambda}{\mu}\right)^k$$

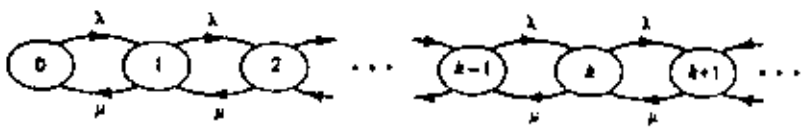


Fig. 3.1.

y

$$p_0 = \frac{1}{1+\sum_{k=1}^{\infty} \left(\frac{\lambda}{\mu}\right)^k}$$

La suma del denominador converge si $\lambda < \mu$ y en este caso

$$p_0 = \frac{1}{1+\frac{\lambda/\mu}{1-\lambda/\mu}} = 1 - \frac{\lambda}{\mu} ;$$

si denominamos $\rho = \frac{\lambda}{\mu}$, la condición de estabilidad del sistema exige que la frecuencia de salida supere a la de llegada, es decir $0 \leq \rho \leq 1$ de donde

$$p_k = (1-\rho)\rho^k \qquad k = 0,1,2,\ldots$$

En este caso puede demostrarse que la distribución de llega-

das es de Poisson y la de tiempos de servicio es exponencial.

Si deseamos calcular el número medio de elementos en el sistema, tendremos

$$\bar{N} = \sum_{k=0}^{\infty} k p_k = \sum_{k=0}^{\infty} k\,(1-\rho)\rho^{k} = (1-\rho)\sum_{k=0}^{\infty} k\rho^{k} =$$

$$= (1-\rho)\frac{\partial}{\partial\rho}\sum_{k=0}^{\infty}\rho^{k+1} = (1-\rho)\rho\frac{\partial}{\partial\rho}\sum_{k=0}^{\infty}\rho^{k} =$$

$$= (1-\rho)\rho\frac{\partial}{\partial\rho}\frac{1}{1-\rho} = \frac{\rho}{1-\rho}$$

y la variancia

$$\sigma_N^2 = \sum_{k=0}^{\infty}(k-\bar{N})\,p_k = \frac{\rho}{(1-\rho)^2}$$

Aplicando la ley de Little podemos determinar el tiempo medio de permanencia en el sistema

$$T = \frac{\bar{N}}{\lambda} = \left(\frac{\rho}{1-\rho}\right)\left(\frac{1}{\lambda}\right) = \frac{1/N}{1-\rho}$$

3.1.2.3. Colas M/M/∞ (infinito número de servidores).

Consideramos aquí el caso que puede interpretarse como un sistema que acelera su frecuencia de servicio linealmente con los clientes o como un sistema que siempre tiene un canal de servicio disponible cuando llega un cliente. En la práctica, tendremos

$$\lambda_k = \lambda \qquad , \quad k = 0,1,2,\ldots$$

$$\mu_k = k\mu \qquad , \quad k = 1,2,3,\ldots$$

El diagrama de transición de estados es el que muestra la figura 3.2.

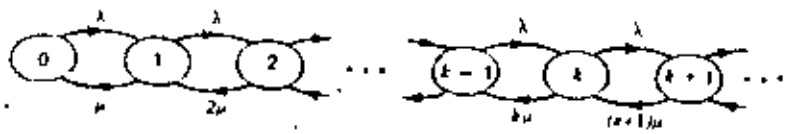


Fig. 3.2.

Sustituyendo en las ecuaciones de Kolmogovov tendremos

$$p_k = p_0 \prod_{i=0}^{k-1} \frac{\lambda}{(i+1)\mu} = p_0 \left(\frac{\lambda}{\mu}\right)^k \frac{1}{k!}$$

y

$$p_0 = \frac{1}{1 + \sum_{k=1}^{\infty} \left(\frac{\lambda}{\mu}\right)^k \frac{1}{k!}} = e^{-\lambda/\mu} = 1 - \rho$$

de donde

$$p_k = \frac{(\lambda/\mu)^k}{k!} e^{-\lambda/\mu} \qquad , \quad k = 0,1,2,\ldots$$

El número medio de elementos en el sistema será

$$\bar{N} = \frac{\lambda}{\mu}$$

y el tiempo medio de permanencia, por la ley de Little, será

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

que coincide con el de servicio, ya que, como habíamos supuesto en este sistema los clientes no tienen ninguna espera.

3.1.2.4. Cola M/M/m.

En este caso suponemos que la estación de servicio dispone de m canales, por lo tanto:

$$\lambda_k = \lambda \quad , \quad k = 0,1,2, \ldots$$

$$\mu_k = \min\,(k\mu,\ m\mu) = \begin{cases} k\mu\ , & 0 \leq k \leq m \\ m\mu\ , & k \geq m \end{cases}$$

El diagrama de transición de estados es el que muestra la fig. 3.3.

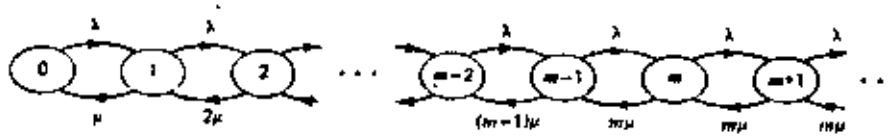


Fig. 3.3.

Debido a la estructura de μ_k, la aplicación de la ecuación de Kolmogovov debe hacerse por partes; para $k \leq m$

$$p_k = p_0 \prod_{i=0}^{k-1} \frac{\lambda}{(i+1)\mu} = p_0 \left(\frac{\lambda}{\mu}\right)^k \frac{1}{k!}$$

y para $k \geq m$

$$p_k = p_0 \prod_{i=0}^{m-1} \frac{\lambda}{(i+1)\mu} \prod_{j=m}^{k-1} \frac{\lambda}{m\mu} = p_0 \left(\frac{\lambda}{\mu}\right)^k \frac{1}{m!\,m^{k-m}}$$

Puede demostrarse que la condición de estabilidad es que $0 \leq \rho = \frac{\lambda}{m\mu} < 1$. De ahí

$$p_0 = \left[\sum_{k=0}^{m-1} \frac{(m\rho)^k}{k!} + \frac{(m\rho)^m}{m!} \frac{1}{1-\rho} \right]^{-1}$$

de donde podemos deducir los valores de p_k.

3.1.2.5. Llegadas desanimadas.

En este caso tenemos que las llegadas disminuyen a medida - que aumenta el número de elementos en el sistema. Una forma de efectuar este modelo es hacer

$$\lambda_k = \frac{\alpha}{k+1} \qquad k = 0,1,2, \ldots$$

$$\mu_k = \mu \qquad k = 1,2,3, \ldots$$

El diagrama de transición de estados es el que muestra la figura 3.4.

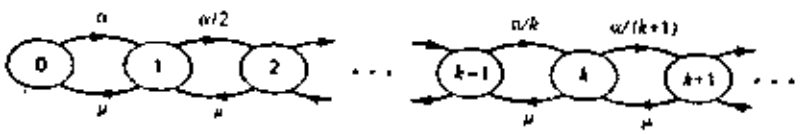


Fig. 3.4.

Aplicando la ecuación de Kolmogovov tenemos

$$p_k = p_0 \prod_{i=0}^{k-1} \frac{\alpha/(i+1)}{\mu} = p_0 \left(\frac{\alpha}{\mu}\right)^k \frac{1}{k!}$$

expresión idéntica a la obtenida en 3.1.2.3. sustituyendo λ por α ; por lo tanto podemos escribir directamente

$$p_0 = e^{-\alpha/\mu} = 1 - \rho$$

$$p_k = \frac{(\alpha/\mu)^k}{k!} e^{-\alpha/\mu}$$

$$\bar{N} = \frac{\alpha}{\mu}$$

3.1.2.6. Cola M/M/1/K: Almacenamiento finito.

En este caso consideramos un sistema que solo puede contener un número finito K de clientes. Cuando el sistema está lleno los nuevos clientes que puedan llegar se pierden. De acuerdo con ello podemos escribir

$$\lambda_k = \begin{cases} \lambda & k < K \\ 0 & k \geq K \end{cases}$$

$$\mu_k = \mu$$

El diagrama de transición de estado es el que muestra la figura 3.5.

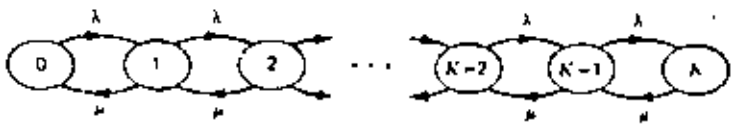


Fig. 3.5.

Aplicando la ecuación de Kolmogovov, tenemos

$$p_k = p_0 \prod_{i=0}^{k-1} \frac{\lambda}{\mu} = p_0 \left(\frac{\lambda}{\mu}\right)^k \quad , \quad k \leq K$$

$$p_k = 0 \quad , \quad k > K$$

de donde

$$p_0 = \left[1 + \sum_{k=1}^{K} \left(\frac{\lambda}{\mu}\right)^k\right]^{-1} = \frac{1 - \frac{\lambda}{\mu}}{1 - \left(\frac{\lambda}{\mu}\right)^{K+1}}$$

a partir de lo cual podemos obtener fácilmente los valores de p_k.

3.1.2.7. Cola M/M/1//M: Población finita.

En este caso el número total de posibles clientes, que constituyen la población que se dirige a nuestro sistema, es finito e igual a M. Supondremos que las llegadas serán proporcionales a los clientes que no están en nuestro sistema. Por lo tanto podremos escribir.

$$\lambda_k = \begin{cases} \lambda(M - k) & , \quad 0 \leq k \leq M \\ 0 & , \quad k > K \end{cases}$$

$$\mu_k = \mu \qquad , \quad k = 1,2,3, \ldots$$

El diagrama de transición de estados es el que muestra la figura 3.6.

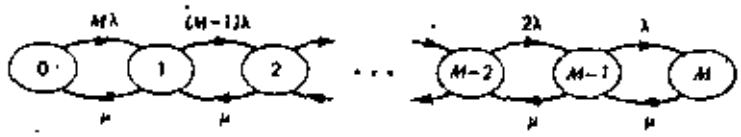


Fig. 3.6.

Aplicando la ecuación de Kolmogovov tendremos

$$p_k = p_0 \prod_{i=0}^{k-1} \frac{\lambda(M-i)}{\mu} = p_0 \left(\frac{\lambda}{\mu}\right)^k \frac{M!}{(M-k)!} \quad , \quad 0 \leq k \leq M$$

$$p_k = 0 \quad , \quad k > M$$

a partir de las cuales podemos obtener

$$p_0 = \left[\sum_{k=0}^{M} \left(\frac{\lambda}{\mu}\right)^k \frac{M!}{(M-k)!} \right]^{-1}$$

de donde podemos calcular los valores de p_k.

3.1.2.8. Cola M/M/∞//M.

Este caso es idéntico al anterior pero suponemos además que existe siempre un canal de servicio para cualquier cliente que llegue al sistema. Es decir

$$\lambda_k = \begin{cases} \lambda(M-k) & ; \quad 0 \leq k \leq M \\ 0 & ; \quad k > M \end{cases}$$

$$\mu_k = k\mu$$

El diagrama de transición de estado es el que muestra la figura 3.7.

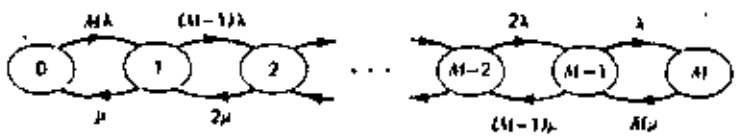


Fig. 3.7.

Aplicando las ecuaciones de Kolmogovov obtenemos

$$p_k = p_0 \prod_{i=0}^{k-1} \frac{\lambda(M-i)}{(i+1)\mu} = p_0 \left(\frac{\lambda}{\mu}\right)^k \binom{M}{k}, \quad 0 \leq k < M$$

de donde podemos obtener

$$p_0 = \left[\sum_{k=0}^{M} \left(\frac{\lambda}{\mu}\right)^k \binom{M}{k}\right]^{-1} = \frac{1}{\left(1+\frac{\lambda}{\mu}\right)^k}$$

así como los valores

$$p_k = \frac{\left(\frac{\lambda}{\mu}\right)^k \binom{M}{k}}{\left(1+\frac{\lambda}{\mu}\right)^k}$$

$$\bar{N} = \sum_{k=0}^{M} k p_k = \frac{\sum_{k=0}^{M} k \left(\frac{\lambda}{\mu}\right)^k \binom{M}{k}}{\left(1 + \frac{\lambda}{\mu}\right)^k} = \frac{M \frac{\lambda}{\mu}}{1+\frac{\lambda}{\mu}}$$

3.1.2.9. Cola M/M/m/K/M.

Este caso reune todos los aspectos tratados parcialmente en los apartados anteriores, en decir, m canales de servicio, espacio para K clientes y población total de M clientes - ($m \leqslant K \leqslant M$). Por todo ello tenemos que

$$\lambda_k = \begin{cases} \lambda(M - k) & 0 \leqslant k \leqslant K \\ 0 & k \geqslant M \end{cases}$$

$$\mu_k = \begin{cases} k\mu & 0 \leqslant k \leqslant m \\ m\mu & k \geqslant m \end{cases}$$

El diagrama de transición de estados es el que nos muestra la figura 3.8.

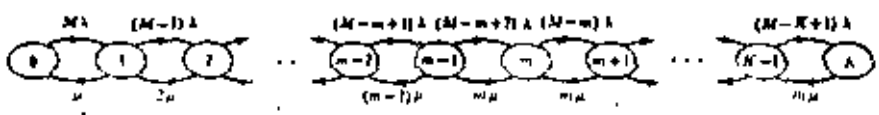


Fig. 3.8.

Aplicando la ecuación de Kolmogovov a cada uno de los intervalos de k, tenemos

$$p_k = p_0 \prod_{i=0}^{k-1} \frac{\lambda(M-i)}{(i+1)\mu} = p_0 \left(\frac{\lambda}{\mu}\right)^k \binom{M}{k} \quad 0 \leq k \leq m$$

$$p_k = p_0 \prod_{i=0}^{m-1} \frac{\lambda(M-i)}{(i+1)} \prod_{i=m}^{k-1} \frac{\lambda(M-i)}{m\mu} =$$

$$= p_0 \left(\frac{\lambda}{\mu}\right)^k \binom{M}{k} \frac{k!}{m!} m^{m-k}, \quad m \leq k \leq K$$

A partir de estas expresiones se puede determinar el valor de p_0, que tiene una fórmula bastante complicada, y a continuación recalcular los valores de p_k que nos caracterizan el comportamiento del sistema.

3.1.3. Colas $M/E_k/1$.

Las colas de este tipo son las que cumplen las hipótesis siguientes:

- Manantial infinito.

- Los intervalos de tiempo entre dos llegadas consecutivas están distribuidos según una ley exponencial de valor medio $t_m = \frac{1}{\lambda}$, (λ, número de llegadas por unidad de tiempo), cuya probabilidad tiene por expresión

  Prob. $(t \leq T) = 1 - e^{T/t_m}$

  lo cual es equivalente a que las llegadas se produzcan según un proceso aleatorio de Poisson, que significa que en un instante dado no pueden producirse dos llegadas, que el número medio de llegadas por unidad de tiempo es λ y que la probabilidad de que por unidad de tiempo se produzcan n llegadas es

$$p(n) = \frac{e^{-\lambda} \lambda^n}{n!}$$

- El tiempo de servicio es también una variable aleatoria - que puede ser desde constante hasta exponencial e hiper-- exponencial (que puede sustituirse por varias exponenciales en serie) pasando por las leyes de Erlang, cuyo grado de aleatoriedad depende de la relación entre la media y - la desviación tipo que se mide generalmente por

$$c = \frac{\sigma}{m} \quad \text{o} \quad E = \left(\frac{m}{\sigma}\right)^2$$

y cuyo valor medio es $\frac{1}{\mu}$, siendo μ entonces el número medio de trabajos que puede atender la estación de servicio por unidad de tiempo.

- La disciplina de la cola es FIFO

- Para que la cola alcance un régimen estacionario estable, y no tienda a una longitud infinita es preciso que el tiempo medio servicio sea inferior al tiempo medio entre llegadas, o lo que es lo mismo $\mu > \lambda$.

- Un solo canal de servicio

En estas circunstancias son aplicables las fórmulas de Khintchine-Pollaczek, en las que interviene el factor de utilización ρ de la estación de servicio y que podemos definir como

$$\rho = \frac{\text{tiempo total con la estación de servicio ocupada}}{\text{tiempo total disponible}}, \quad \text{o}$$

$$\rho = \frac{\text{carga total de la estación de servicio}}{\text{carga máxima posible de la estación de servicio}}\ , \text{ o}$$

ρ = número medio de llegadas (λ) x tiempo medio de servicio $\left(\frac{1}{\mu}\right)$ y que representa la probabilidad de que la estación de servicio esté ocupada.

Número medio de trabajos en la estación de servicio:

$$\bar{N} = \rho + \frac{\rho^2}{2(1-\rho)} \left(1 + \frac{1}{E}\right)$$

Tiempo medio de permanencia en la estación de servicio:

$$\bar{t} = \frac{1}{\mu}\left[1 + \frac{\rho}{2(1-\rho)} \left(1 + \frac{1}{E}\right)\right]$$

Estas fórmulas se simplifican cuando el tiempo de servicio es constante, $\sigma = 0$, y por lo tanto $E = \infty$

$$\bar{N} = \frac{(2-\rho)\rho}{2(1-\rho)}$$

$$\bar{t} = \frac{(2-\rho)}{2(1-\rho)} \quad \frac{1}{\mu}$$

y cuando la distribución de tiempos de servicio es exponencial, $\sigma = \frac{1}{\mu}$, y, por lo tanto $E = 1$

$$\bar{N} = \frac{\rho}{1-\rho}$$

$$\bar{t} = \frac{1}{1-\rho} \quad \frac{1}{\mu}$$

Estas fórmulas así como las correspondientes a las desviaciones tipo se encuentran dispuestas en gráficos en las figuras 3.9, 3.10, 3.11 y 3.12.

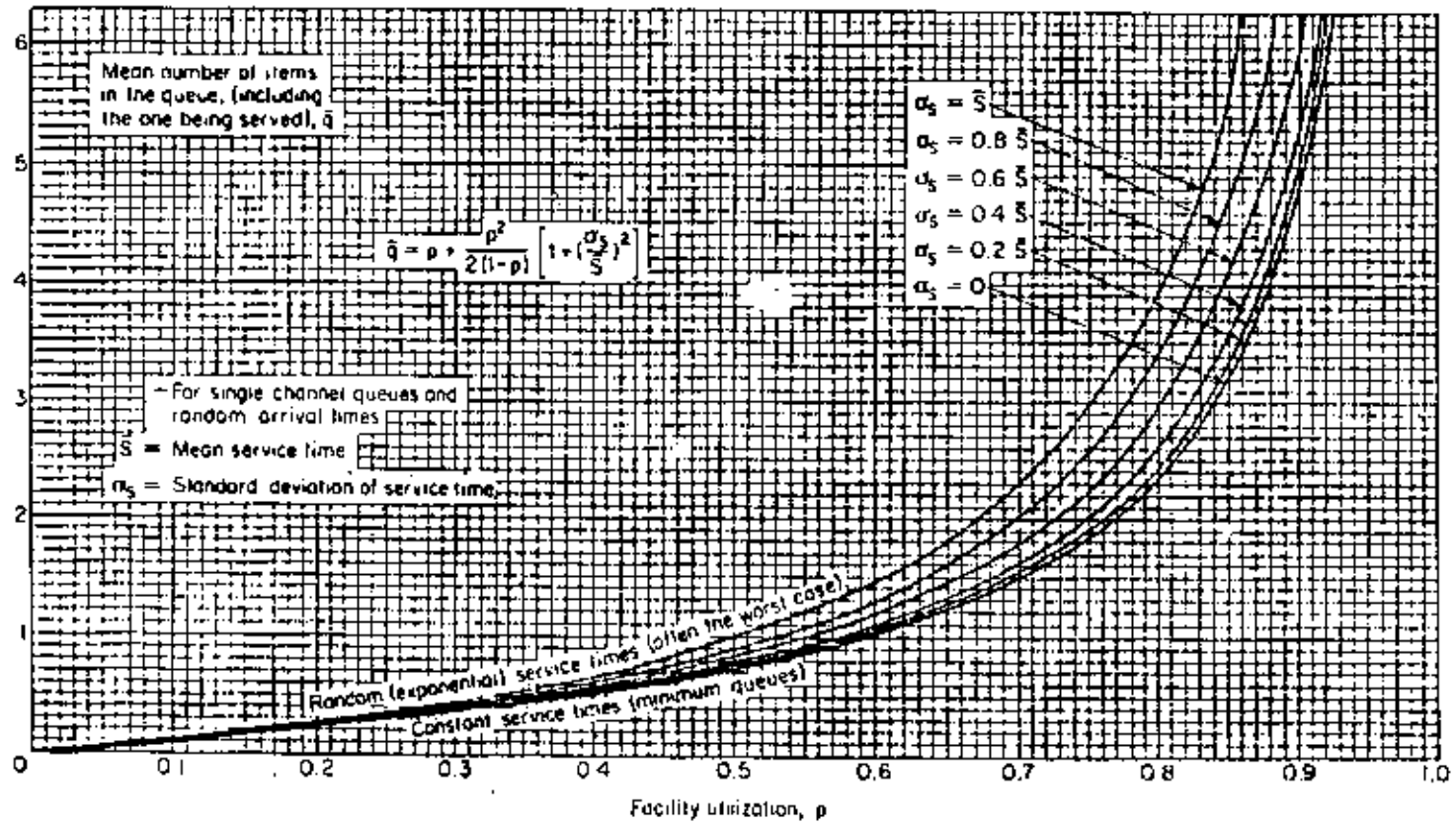


Fig. 3.9.

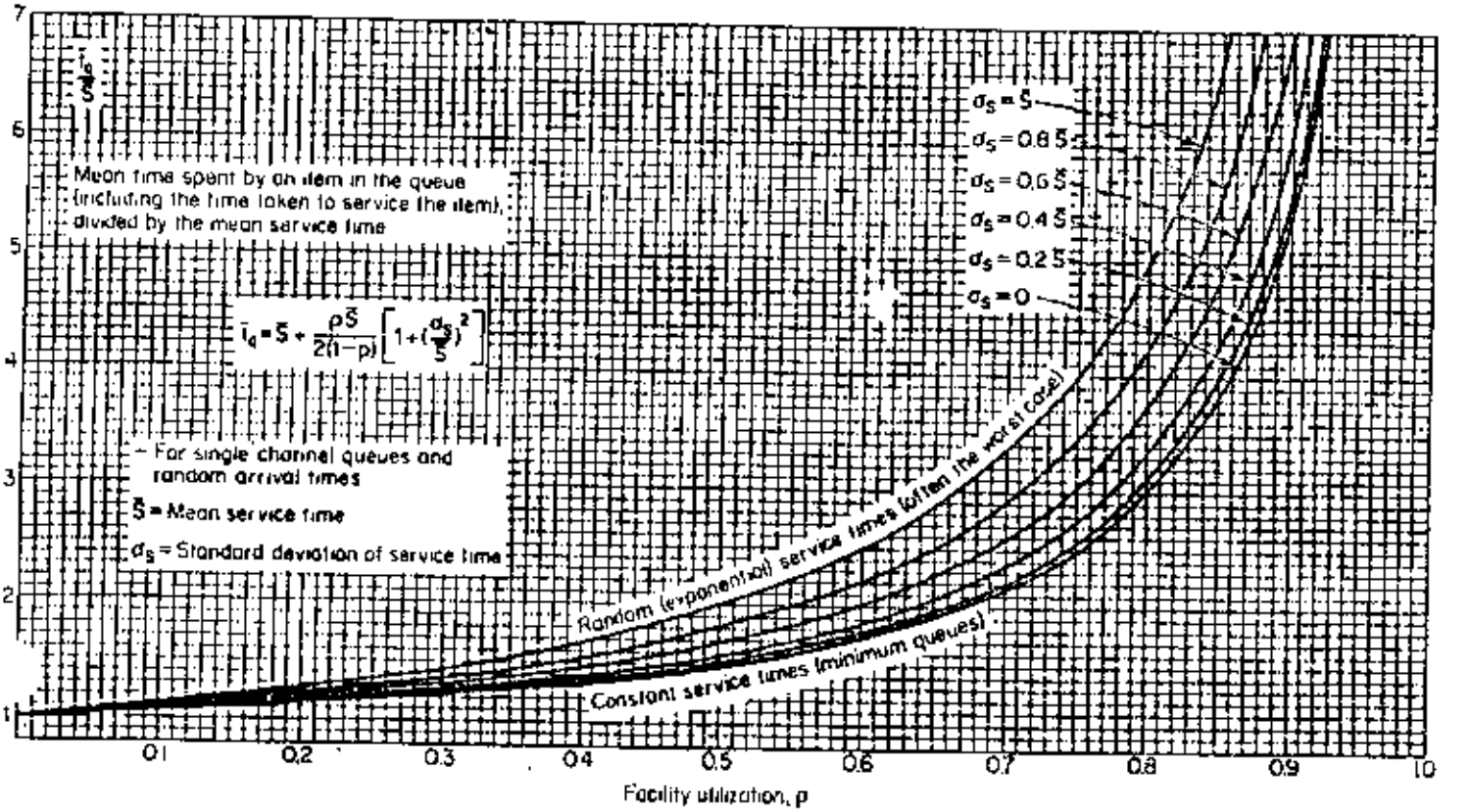


Fig. 3.10.

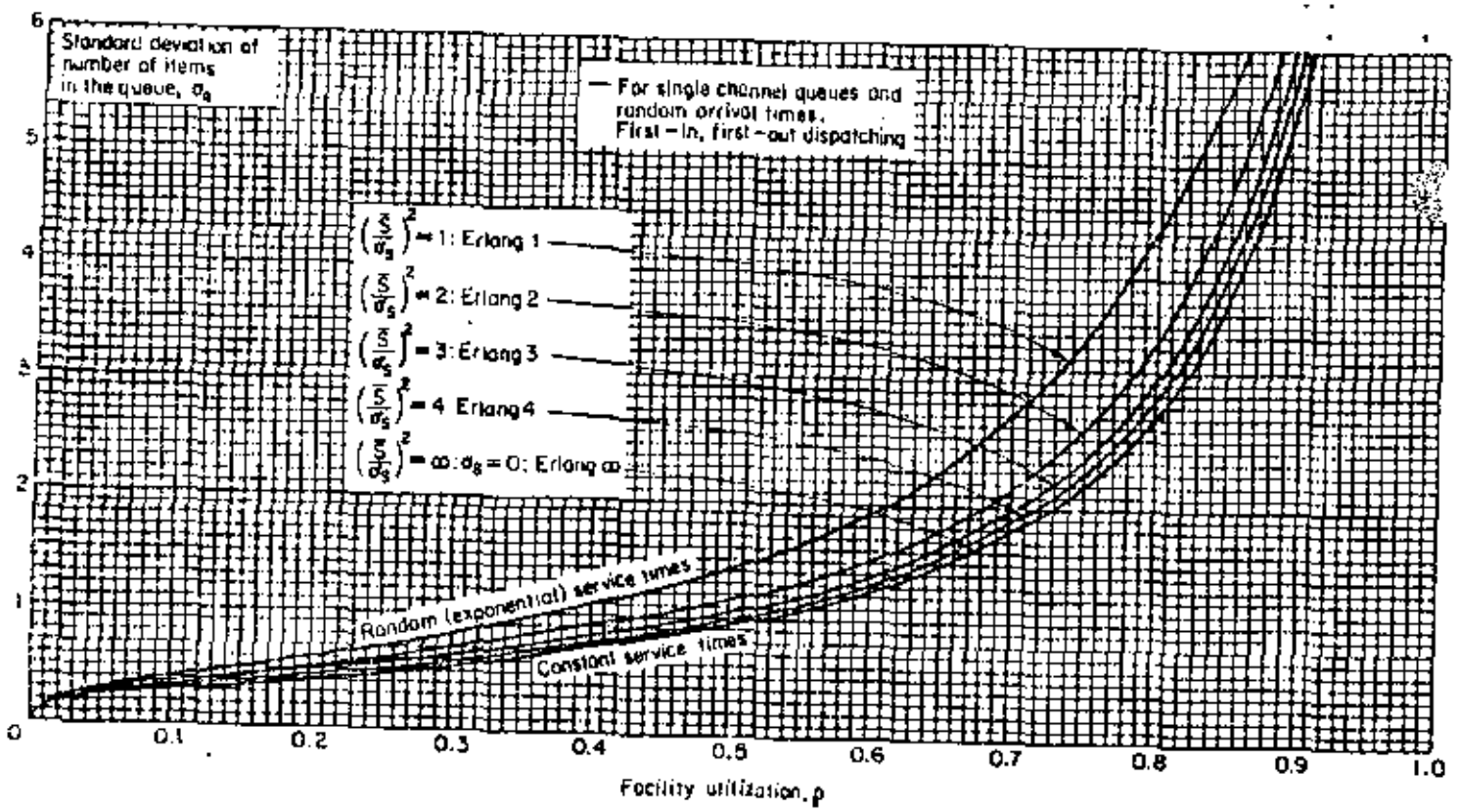


Fig. 3.11.

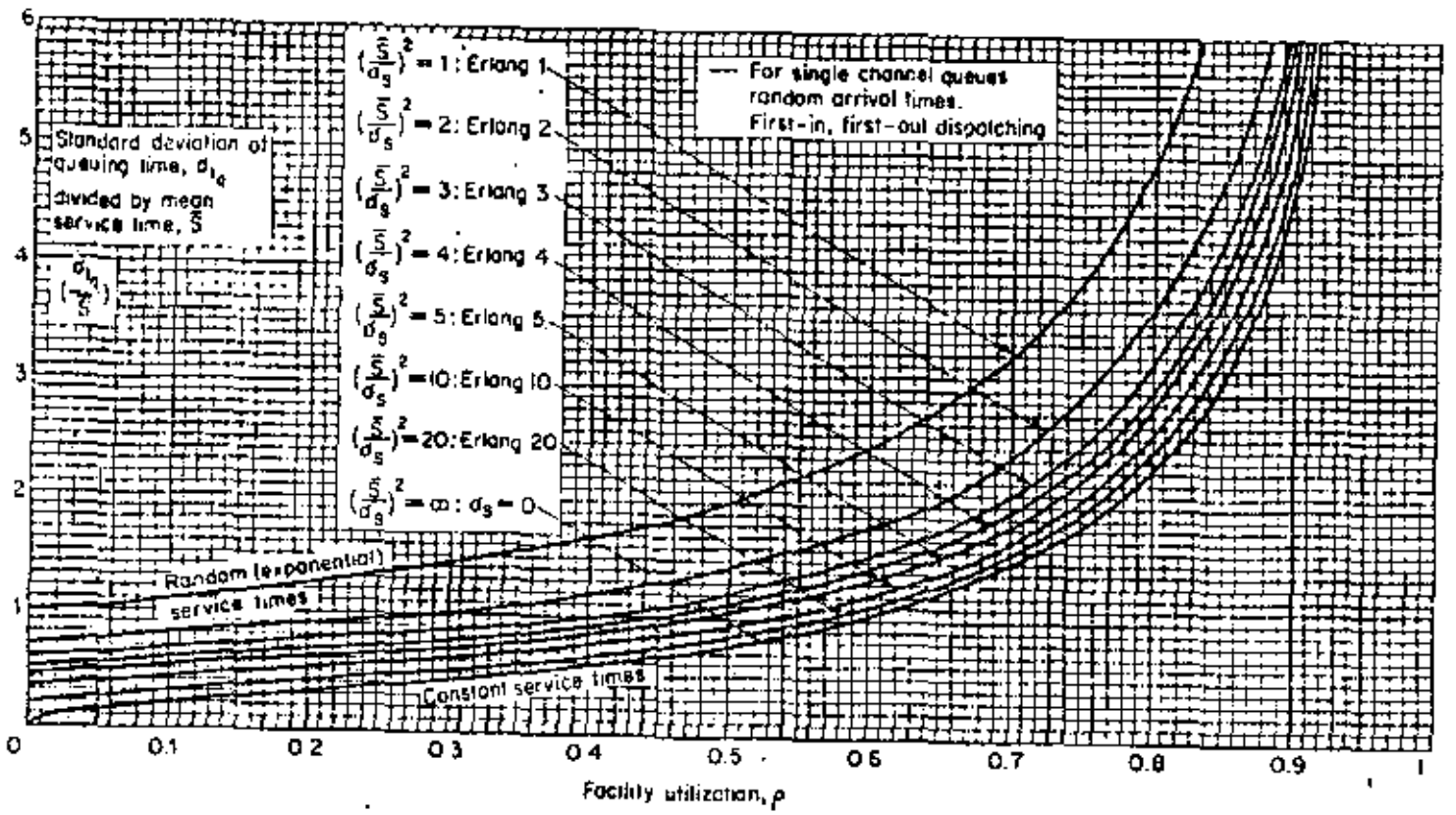


Fig. 3.12.

3.2. Modelos individuales de los subsistemas.

La modelización de sistemas informáticos puede enfocarse estableciendo modelos individuales de cada uno de los subsistemas o bien estableciendo un modelo global de todo el sistema. En este apartado analizaremos los modelos individuales de cada uno de los subsistemas, y en el 3.3 los modelos globales.

3.2.1. Tambores o discos de cabezas fijas.

Estos dispositivos sólo permiten tratar una sola entrada/salida simultánea, debido a que la unidad de control y el canal sólo permiten el paso de una de ellas. Por lo tanto, independientemente del número de ejes de que dispongamos, a efectos de su modelización es equivalente a que dispusiéramos de uno solo. El modelo está constituido por una cola que da acceso a la unidad que controla los distintos dispositivos.

El número de accesos por unidad de tiempo (λ) depende de las aplicaciones que se ejecutan en un momento dado.

El tiempo medio de servicio $\frac{1}{\mu}$ y la variancia de los tiempos de servicio se determinan teniendo en cuenta el tiempo de latencia (tiempo de espera hasta que el registro deseado pasa por debajo de la cabeza de lectura/escritura) y el de transferencia de los registros.

Esta situación puede pues modelizarse sin dificultad mediante una cola $M/E_k/1$.

Ejemplo: En un canal de I/O hay tres tambores conectados a una unidad de cotrol. El tiempo medio de transferencia, incluyendo el de latencia, es de 20 ms. con una desviación tipo de 10 ms. Los accesos se producen a razón de 30 por segundo. Determinar el tiempo medio de respuesta. ¿Qué sucedería si en vez de 30 accesos/seg. fuesen 40?

$$E = \left(\frac{20}{10}\right)^2 = 4$$

$$\lambda = 30 \text{ ac/seg.}$$

$$\frac{1}{\mu} = 20 \text{ ms} = 0,02 \text{ seg.}$$

$$\rho = 30 \times 0,02 = 0,6$$

$$\bar{t} = 0,02 \left[1 + \frac{0,6}{2(1-0,6)} \left(1 + \frac{1}{4}\right)\right] = 0,039 \text{ seg.}$$

$$\lambda = 40 \text{ ac/seg.}$$

$$\frac{1}{\mu} = 20 \text{ ms} = 0,02 \text{ seg.}$$

$$\rho = 40 \times 0,02 = 0,8$$

$$t = 0,02 \left[1 + \frac{0,8}{2(1-0,8)} \left(1 + \frac{1}{4}\right)\right] = 0,07 \text{ seg.}$$

La observación de estos resultados nos permite extraer una consecuencia de tipo general que es debida al término $1 - \rho$ en el denominador; si ρ es pequeño sus variaciones afectan relativamente poco al resultado, pero si ρ se aproxima a 1 su crecimiento provoca aumentos muy importantes en el tiempo de respuesta.

3.2.2. Discos.

De forma simplificada, el proceso de I/O en un disco se puede descomponer en las siguientes fases:

. El acceso se pone en cola para acceder al disco correspondiente. La gestión de esta cola por parte del sistema operativo se puede hacer según distintas políticas (FIFO, mínimo desplazamiento del brazo, etc.). En el tratamiento que sigue supondremos que la política es FIFO.

. Sale de la cola para lanzar el movimiento del brazo (seek) ocupando durante un tiempo despreciable la unidad de control cuando la unidad de control y el disco correspondiente están libres simultáneamente.

. Una vez acabado el desplazamiento del brazo, si el disco no tiene sistema de posicionamiento angular trata de iniciar la transferencia a través de la unidad de control, colocándose en la cola de este dispositivo. Si el disco tiene dispositivo de posicionamiento angular la solicitud de servicio a la unidad de control no se produce sino cuando pasa por debajo de la cabeza de lectura/escritura una determinada señal que está un cierto ángulo delante del registro; entonces si la unidad de control está libre adquiere servicio y se efectúa la transferencia, pero si está atendiendo otra transferencia entonces pierde una vuelta antes de volverlo a intentar, repitiéndose el proceso hasta que se puede llevar a cabo la transferencia.

. Una vez se adquiere servicio de la unidad de control, es preciso esperar que el registro llegue debajo de la cabeza de lectura/escritura (tiempo de latencia) para dejar desfilar entonces todo el registro (tiempo de transferencia). En caso de que no haya posicionamiento angular, el tiempo de latencia está uniformemente distribuído entre cero y el tiempo correspondiente a una vuelta. Si hay posicionamiento angular en vez del tiempo de latencia tenemos el tiempo correspondiente al ángulo que precede al registro.

. Una vez acabada la transferencia se liberan a la vez el disco y la unidad de control, que quedan en disposición de atender nuevos accesos.

3.2.2.1. Discos sin posicionamento angular.

El funcionamiento de un subsistema de este tipo, que acabamos de describir, podemos representarlo en el diagrama de la figura 3.13.

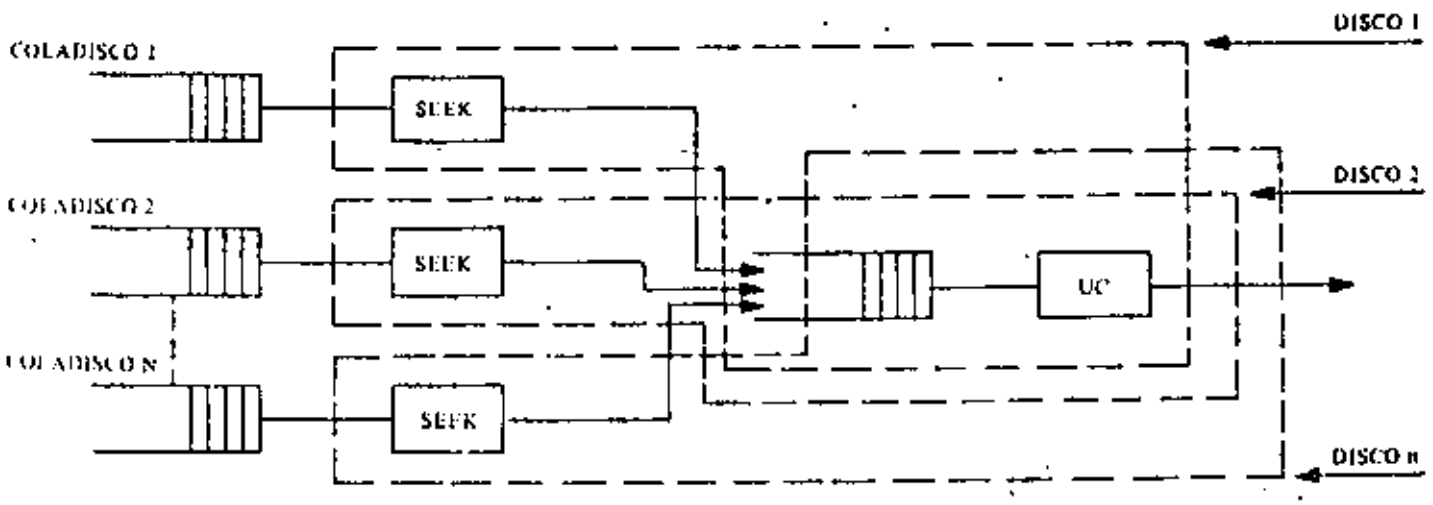


Fig. 3.13.

El cálculo del tiempo de respuesta se desglosa en las siguientes fases:

a) Cálculo para cada archivo i del tiempo medio de ocupación de la unidad de control, que es igual al tiempo de latencia, L_i más el tiempo de transferencia, T FER_i.

Los tiempos de latencia están distribuidos según una ley uniforme entre cero y el tiempo de una vuelta. Por lo tanto, el tiempo medio de latencia corresponde al de media vuelta y la varianza al cuadrado del tiempo de una vuelta dividido por doce.

El tiempo de transferencia se puede considerar igual al tiempo de una vuelta dividido por el número de registros físicos que hay en una pista o bien puede calcularse a partir de la velocidad de transferencia del disco.

b) Determinación del número medio N_i de accesos, tanto entradas como salidas, a cada archivo, i, que depende de las aplicaciones que se ejecuten en ese instante.

c) Cálculo de la ocupación de la unidad de control provocada por cada archivo y que es $N_i(\bar{L}_i + T\ FER_i)$

d) Cálculo del factor de utilización del canal, que es igual a la suma de todos los valores calculados en el apartado anterior.

$$\rho_{ch} = \sum_i (T\ FER_i + \bar{L}_i)\ N_i$$

e) Cálculo del tiempo medio del servicio del canal, que es promedio de los valores calculados en el apartado a).

$$\frac{1}{\mu_{ch}} = \frac{\sum (T\ FER_i + \bar{L}_i)\ N_i}{\sum_i N_i}$$

f) Cálculo del tiempo medio de espera en el canal, W_{ch} que es igual al tiempo medio de estancia en el sistema menos el de servicio. Hay que tener en cuenta que en este caso no son aplicables las fórmulas de Khintchine-Pollaczeck ya que el manantial no es infinito, puesto que esta cola puede tener, como máximo, tantos accesos en espera como ejes haya en el subsistema. Es preciso utilizar un modelo a base de una cola con el número de clientes finito (Apartado 3.1.2.7) fórmulas adecuadas o los gráficos que nos dan ese valor directamente en función de ρ_{ch}, de $\frac{1}{\mu_{ch}}$ y del número de mecanismos de acceso o ejes. (fig. 3.14).

g) Cálculo para cada disco j (un archivo puede estar entre varios discos, o, por el contrario, en un disco puede haber varios archivos) de las características del tiempo de servicio, que es igual al tiempo medio de desplazamiento de brazo, SK_j, más tiempo de espera en el canal, W_{ch} más el tiempo medio de latencia, L_j, más el tiempo medio de transferencia $T\ FER_j$, que depende de los archivos que haya en el disco

$$t_{Dj} = SK_j + W_{CH} + L_j + TFER_j$$

El desplazamiento del brazo depende del tipo de accionamiento y de la ocupación y la situación de los archivos en el disco.

Para calcular las características estadísticas del desplazamiento es preciso conocer además de la situación y ocupación de los archivos en el disco y la frecuencia de acceso a cada uno de ellos. Si admitimos que solo hay un archivo que ocupa N cilindros podremos decir que la probabilidad de que se produzca un desplazamiento de n cilindros, si todos son igualmente probables, vale

$$P_n = \frac{2n}{N} \cdot \frac{1}{N} + \frac{(N-2n)}{N} \cdot \frac{1}{N} \cdot 2 = \frac{2(N-n)}{N^2}$$

El primer sumando representa la probabilidad de que la posición inicial esté en los n primeros o últimos cilindros y se haga el desplazamiento citado (solo puede ser en un sentido). El segundo sumando representa la probabilidad de un desplazamiento que tenga su posición inicial entre los cilindros n y N-n, en cuyo caso el desplazamiento puede hacerse en los dos sentidos.

En consecuencia el desplazamiento medio valdrá

$$\bar{n} = \sum_{n=1}^{N} n P_n = \sum_{n=1}^{N} n \frac{2(N-n)}{N^2} = \frac{N^2-1}{3N}$$

y la variancia

$$\sigma_n^2 = \sum_{n=1}^{N} n^2 P_n - \bar{n}^2 = \sum_{n=1}^{N} n^2 \frac{2(N-n)}{N^2} - \left(\frac{N^2-1}{3N}\right)^2 = \frac{N^4+N^2-18}{18N^2}$$

Cuando el número de cilindros ocupados por el archivo, N, es grande, estos valores pueden aproximarse por

$$\bar{n} \simeq \frac{N}{3} \qquad \sigma_n^2 = \frac{N^2}{15}$$

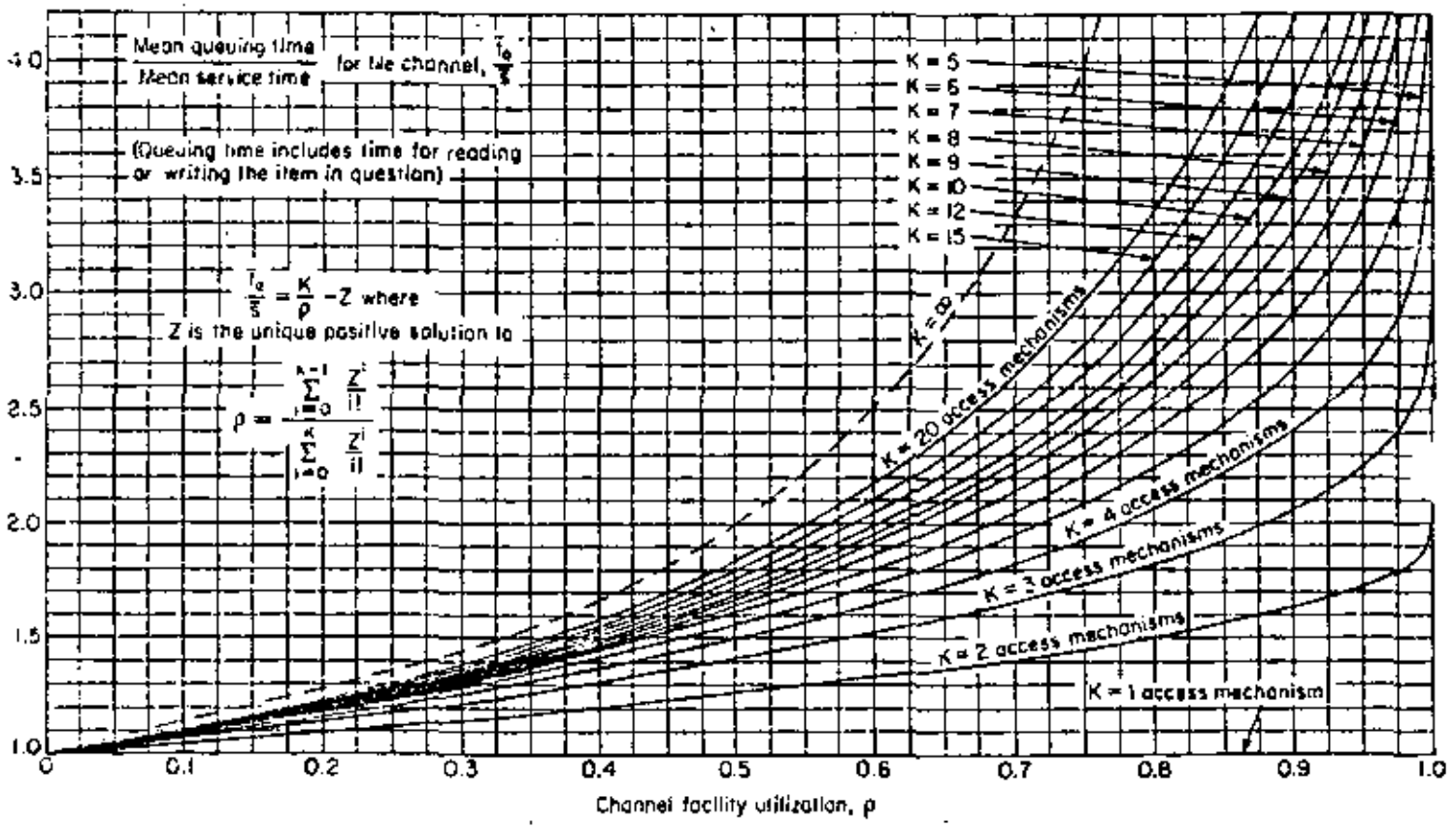


Fig. 3.14.

Aún cuando realmente este cálculo debería hacerse con los tiempos necesarios para efectuar estos desplazamientos, normalmente proporciona una aproximación suficiente la trnas-- formación en tiempo de los números de cilindros $\bar{n}$ y σ_n de -

acuerdo con las curvas que nos relacionan desplazamientos - con tiempos (fig. 3.15).

El tiempo de espera en el canal tiene un valor medio que ya hemos calculado (párrafo 3.2.2.1.f) y una varianoia que podríamos calcular a partir de la distribución de probabilidad del número de clientes en el canal, pero que, en primera - aproximación, (la que se efectua normalmente) podemos considerar igual al cuadrado de la media. Esta aproximación no -

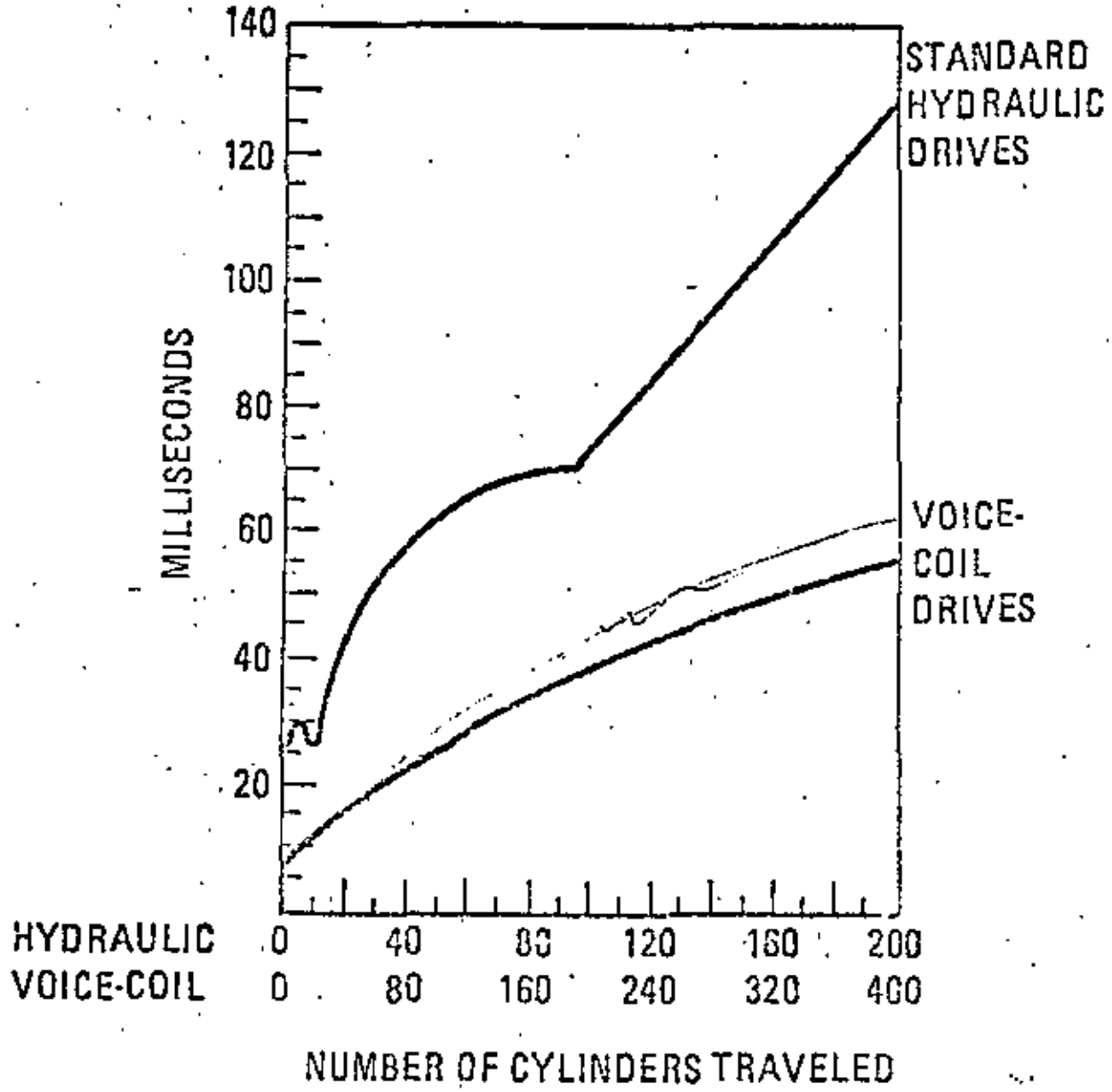


Fig. 3.15.

tiene demasiada influencia ya que normalmente este sumando representa un porcentaje pequeño del tiempo de servicio del disco.

Las características estadísticas del tiempo de latencia ya se han comentado (párrafo 3.2.2.1.a) y las del tiempo de transferencia se pueden determinar fácilmente conociendo las de los archivos que hay en el disco y las frecuencias de acceso de cada uno de ellos. Por lo tanto podemos escribir

$$\frac{1}{\mu_{Dj}} = \overline{SK}_j + \overline{W_{CH}} + \overline{L}_j + \frac{\sum_{i \in Dj} N_i \times TFER_i}{\sum_{i \in Dj} N_i}$$

$$\sigma^2_{Dj} = \sigma^2_{SK} + \overline{W}^2_{CH} + \frac{(2\overline{L}_j)^2}{12} + \left[\frac{\sum_{i \in Dj} N_i \times TFER_i^2}{\sum_{i \in Dj} N_i} - \left(\frac{\sum_{i \in Dj} N_i \times TFER_i}{\sum_{i \in Dj} N_i} \right) \right]$$

h) Cálculo, a partir de los datos obtenidos, del factor de utilización de cada disco

$$\rho_{Dj} = \frac{1}{\mu_{Dj}} \sum_{i \in Dj} N_i$$

i) Cálculo del tiempo de respuesta de cada disco por aplicación de las fórmulas de Khintchine-Pollaczeck. Hacemos sin embargo, la incorrección de suponer que las llegadas provienen de una población infinita, lo cual no es cierto, puesto que como máximo es igual al número de trabajos en ejecución en la unidad central. No obstante, si el nivel de multiprogramación es un poco elevado o si se trata de sistemas trans

accionales, los resultados acostumbran a ser suficientemente aproximados.

j) Al actuar de esta forma no hemos tenido en cuenta que para salir de la cola que da acceso al disco correspondiente - es preciso no sólo que el disco esté libre sino que también lo ha de estar la unidad de control. Puede añadirse un tiempo que corrija el tiempo de espera y que puede hacerse igual a

$$\rho_{ch} \cdot \frac{1}{2}\left[\frac{1}{\mu ch} + \frac{\sigma^2_{ch}}{1/\mu_{ch}}\right]$$

y para ser más exactos deben considerarse para cada disco - los valores referentes al canal provocados por los restantes discos, pero no él mismo. No obstante, en la mayoría de los casos, esta corrección tiene poca importancia, a no ser que se alcancen elevados factores de utilización de los discos y la unidad de control.

La interpretación de esta expresión nos dice que la probabilidad de encontrar el canal ocupado es ρ_{ch} y el tiempo medio de ocupación será la mitad del tiempo de transferencia del - canal. El término complementario es para tener en cuenta la distribución de los tiempos de transferencia.

Ejemplo: Consideremos un subsistema de tres discos donde se hallan los archivos que consulta y actualiza un sistema transaccional de tiempo real. Las transcciones llegan a razón de 30 por segundo. Los discos giran a razón de 3.600 r.p.m.

Al archivo A acceden el 60% de las transacciones de las cuales el 25% son de actualización. Hay 10 registros por pista. El disco está totalmente lleno en un 50%, lo cual hace que la media y la desviación tipo de los tiempos de seek sean 20 ms y 16 ms.

Al archivo B acceden el 40% de las transacciones sin actualización. Hay 5 registros por pista. El disco está totalmente lleno por lo que la media y la desviación tipo de los tiempos seek son 30 ms y 24 ms.

Al archivo C acceden el 20% de las transacciones que hacen en promedio tres accesos. Hay 20 registros por pista. El disco está lleno en un 30%, por lo que la media y la desviación tipo de los tiempos de seek son 15 ms y 12 ms.

Determinar los tiempos medios de acceso de cada archivo.

$N_A = 30 \cdot 0,6 \cdot 1,25 = 22,5$ accesos/seg

$N_B = 30 \cdot 0,4 \cdot 1 = 12$ accesos/seg

$N_C = 30 \cdot 0,2 \cdot 3 = 18$ accesos/seg

$L + TFER_A = 8,33 + \frac{16,67}{10} = 10$ mseg

$L + TFER_B = 8,33 + \frac{16,67}{5} = 11,67$ mseg

$L + TFER_C = 8,33 + \frac{16,67}{20} = 9,17$ mseg

$N_A(L + TFER_A) = 0,225$ seg

$N_B(L + TFER_B) = 0,140$ seg

$N_C\ (L + TFER_C) = 0,165$ seg

$\rho_{ch} = 0,225 + 0,140 + 0,165 = 0,53$

$\frac{1}{\mu_{ch}} = \frac{0,225 + 0,140 + 0,165}{22,5 + 12 + 18} = 0,0101$ seg

Para determinar el tiempo de espera en el canal podemos usar el gráfico de la figura 3.13 obteniendo

$$W_{CII} = (1,45 - 1)\ 0,0101 = 0,00455 \text{ seg}$$

$\frac{1}{\mu_A} = 20 + 4,55 + 10 = 34,55$ mseg

$\frac{1}{\mu_B} = 30 + 4,55 + 11,67 = 46,22$ mseg

$\frac{1}{\mu_C} = 15 + 4,55 + 9,17 = 28,72$ mseg

$\rho_A = 22,5 \times 0,03455 = 0,777$

$\rho_B = 12 \times 0,04622 = 0,555$

$\rho_C = 18 \times 0,02872 = 0,517$

$\sigma_A^2 = 16^2 + 4,55^2 + \frac{16,67^2}{12} + 0^2 = 299,85$ mseg^2

$\sigma_B^2 = 24^2 + 4,55^2 + \frac{16,67^2}{12} + 0^2 = 619,85$ mseg^2

$\sigma_C^2 = 12^2 + 4,55^2 + \frac{16,67^2}{12} + 0^2 = 187,85$ mseg^2

$$\bar{t}_A = 34,55 \left[1 + \frac{0,777}{2(1-0,777)} \left(1 + \frac{299,85}{34,55^2}\right)\right] = 109,83 \text{ mseg}$$

$$\bar{t}_B = 46,22 \left[1 + \frac{0,555}{2(1-0,555)} \left(1 + \frac{619,85}{46,22^2}\right)\right] = 83,38 \text{ mseg}$$

$$t_C = 28,72 \left[1 + \frac{0,517}{2(1-0,517)} \left(1 + \frac{187,85}{28,72^2}\right)\right] = 47,70 \text{ mseg}$$

Utilizando los gráficos de la figura 3.12 obtenemos las desviaciones tipo de los tiempos de respuesta.

$$\sigma_{t_A} = 2,7 \times 34,55 = 93,29 \text{ mseg}$$

$$\sigma_{t_B} = 1,35 \times 46,22 = 62,40 \text{ mseg}$$

$$\sigma_{t_C} = 1,15 \times 28,72 = 33,03 \text{ mseg}$$

La corrección debida a la espera suplementaria en la cola a causa de estar ocupada la unidad de control se calcula como sigue,

$$\rho'_A = 0,140 + 0,165 = 0,305$$

$$\frac{1}{\mu'_A} = \frac{12 \times 11,67 + 18 \times 9,17}{12 \times 18} = 10,17 \text{ mseg}$$

$$\sigma'^2_A = \frac{12\left(11,67^2 + \frac{16,67^2}{12}\right) + 18\left(9,17^2 + \frac{16,67^2}{12}\right)}{12 + 18} -$$

$$- 10,17^2 = 24,652 \text{ mseg}^2$$

$$\Delta t_A = 0,305 \cdot \frac{1}{2} \quad 10,17 + \frac{24,652}{10,17^2} = 1,92 \text{ mseg}$$

$$\rho'_B = 0,225 + 0,165 = 0,39$$

$$\frac{1}{\mu'_B} = \frac{22,5 \times 10 \times 18 \times 9,17}{22,5 + 18} = 9,63 \text{ mseg}$$

$$\sigma'^2_B = \frac{22,5(10^2 + \frac{16,67^2}{12}) + 18(9,17^2 + \frac{16,67^2}{12})}{22,5 + 18} - 9,63^2 = 23,317 \text{ mseg}^2$$

$$\Delta t_B = 0,39 \cdot \frac{1}{2}(9,63 + \frac{23,317}{9,63}) = 2,35 \text{ mseg}$$

$$\rho'_C = 0,225 + 0,14 = 0,365$$

$$\frac{1}{\mu'_C} = \frac{22,5 \times 10 + 12 \times 11,67}{22,5 + 12} = 10,58 \text{ mseg}$$

$$\sigma'^2_C = \frac{22,5(10^2 + \frac{16,67^2}{12}) + 12(11,67^2 + \frac{16,67^2}{12})}{22,5 + 12} - 10,58^2 = 23,776 \text{ mseg}^2$$

$$\Delta t_C = 0,365 \cdot \frac{1}{2}(10,58 + \frac{23,776}{10,58}) = 2,34 \text{ mseg}$$

$$\bar{t}_A = 109,83 + 1,92 = 111,75 \text{ mseg}$$

$$\bar{t}_B = 83,38 + 2,35 = 85,73 \text{ mseg}$$

$$\bar{t}_C = 47,70 + 2,34 = 50,04 \text{ mseg}$$

3.2.2.2. Discos con posicionamiento angular.

En el caso de discos con posicionamiento angular, la marcha de cálculo es muy similar, teniendo en cuenta, sin embargo, las diferencias de funcionamiento.

a) Cálculo para cada archivo i del tiempo medio de ocupación de la unidad de control, que es igual al tiempo medio de posicionamiento angular, $\overline{RPS}_i$, más el de transferencia, $TFER_i$.

El tiempo de posicionamiento angular depende del número de marcas grabadas físicamente en el disco para detectar registros y del número de registros por pista.

b) Idéntico al párrafo 3.2.2.1.b.

c) Cálculo de la ocupación de la unidad de control provocada por el archivo i, que vale en este caso

$$N_i \ (\overline{RPS}_i + TFER_i)$$

d) Cálculo del factor de utilización del canal, que es la suma de todos los valores calculados en el párrafo anterior.

$$\rho_{CH} = \sum_i (\overline{RPS}_i + TFER_i) \ N_i$$

e) No es necesario en este caso.

f) En estas circunstancias no se puede hablar propiamente de tiempo de espera, sinó de tiempo debido a vueltas perdidas. Podemos escribir que la probabilidad de no perder vuelta, p_o, vale

$$p_o = 1 - \rho_{CH}$$

y las de perder 1,2, ... vueltas, son, respectivamente

$$p_1 = \rho_{CH} \ (1 - \rho_{CH})$$

$$p_2 = \rho_{CH}^2 (1 - \rho_{CH})$$

$$p_n = \rho_{CH}^n (1 - \rho_{CH})$$

Por lo tanto, las características estadísticas de este tiempo valen

$$\overline{W}_{vp} = 2L \sum_{i=0}^{\infty} i\, p_i = 2L \sum_{i=0}^{\infty} i \rho_{CH}^i (1 - \rho_{CH}) = 2L \frac{\rho_{CH}}{1-\rho_{CH}}$$

$$\sigma_{vp}^2 = (2L)^2 \left[\sum_{i=0}^{\infty} i^2 p_i - \left(\frac{\rho_{CH}}{1-\rho_{CH}}\right)^2\right] = 4L^2 \left[\sum_{i=0}^{\infty} i^2 \rho_{CH}^i (1-\rho_{CH}) - \frac{\rho_{CH}^2}{(1-\rho_{CH})^2}\right] =$$

$$= 4L^2 \frac{\rho_{CH}}{(1-\rho_{CH})^2}$$

g) El tiempo de servicio de cada disco, j, es igual al tiempo de desplazamiento del brazo, más el tiempo de espera hasta que pasa la marca de posicionamiento angular, más el tiempo debido a vueltas perdidas, más el tiempo del ángulo de posicionamiento, más el tiempo de transferencia

$$t_{Dj} = SK_j + L_j + W_{vp} + RPS_j + TFER_j$$

Como conocemos las características estadísticas de todos los sumandos, podemos escribir

$$\frac{1}{\mu_{Dj}} = \overline{SK}_j + \overline{L}_j + \overline{W}_{vp} + \frac{\sum_{i \in Dj} N_i (RPS_i + TFER_i)}{\sum_{i \in Dj} N_i}$$

$$\sigma^2_{Dj} = \sigma^2_{SK} + \frac{(2L_j)^2}{12} + \sigma^2_{vp} + \left[\frac{\sum_{i\in Dj} N_i \times TFER_i^2}{\sum_{i\in Dj} N_i} - \left(\frac{\sum_{i\in Dj} N_i \times TFER_i}{\sum_{i\in Dj} N_i}\right)^2\right] +$$

$$+ \left[\frac{\sum_{i\in Dj} N_i \left[\frac{(RPS_{max} - RPS_{min})^2}{12} + RPS_i^2\right]}{\sum_{i\in Dj} N_i} - \left(\frac{\sum_{i\in Dj} N_i \times RPS_i}{\sum_{i\in Dj} N_i}\right)^2\right]$$

h) Idéntico al del párrafo 3.2.2.1 h.

i) Idéntico al del párrafo 3.2.2.1 i.

j) Idéntico al del párrafo 3.2.2.1 j.

Un ejemplo del cálculo de este caso lo trataremos en el apartado siguiente.

3.2.2.1. Discos con unidad de acceso dual.

En numerosos casos cuando una sola unidad de control no puede soportar el tráfico de los archivos del subsistema, en vez de partirlo en dos subsistemas, se puede colocar una segunda unidad de control sobre el mismo subsistema. Para distribuir los accesos entre ambas unidades de control se pueden adoptar diversas políticas, de las cuales las más frecuentes son:

. Considerar las dos unidades de control como una estación de servicio con dos canales.

. Considerar que se intenta acceder siempre a través de una unidad de control primaria. Si al llegar el acceso la encuentra libre todo el proceso se efectuará a través de ella. Si está ocupada se va a la secundaria que será la que atenderá todo el proceso y ante la cual permanecerá en cola si al llegar la encuentra ocupada.

En el primer caso no hace falta variar mucho las marchas de cálculo propuestas en los apartados anteriores solo hace falta considerar que la estación unidad de control tiene dos canales de servicio, que se puede tratar en forma aproximada mediante los gráficos de las figuras 3.16, 3.17 y 3.18 o en forma más exacta estableciendo un modelo de tipo M/M/2//M (apartado 3.1.2.9).

En el segundo caso es preciso determinar el porcentaje de acceso que pasa a través de cada una de las unidades de control. Este porcentaje X, se determina a partir de la ecuación

$$1 - X = \rho_{CH}\, X$$

que dice que el porcentaje de accesos que va a través de la segunda unidad de control es igual a la probabilidad de que la unidad de control primaria esté ocupada. En esta ecuación ρ_{CH} es la ocupación conjunta de las dos unidades de control. A partir de esta ecuación obtenemos:

$$X = \frac{1}{1+\rho_{CH}}$$

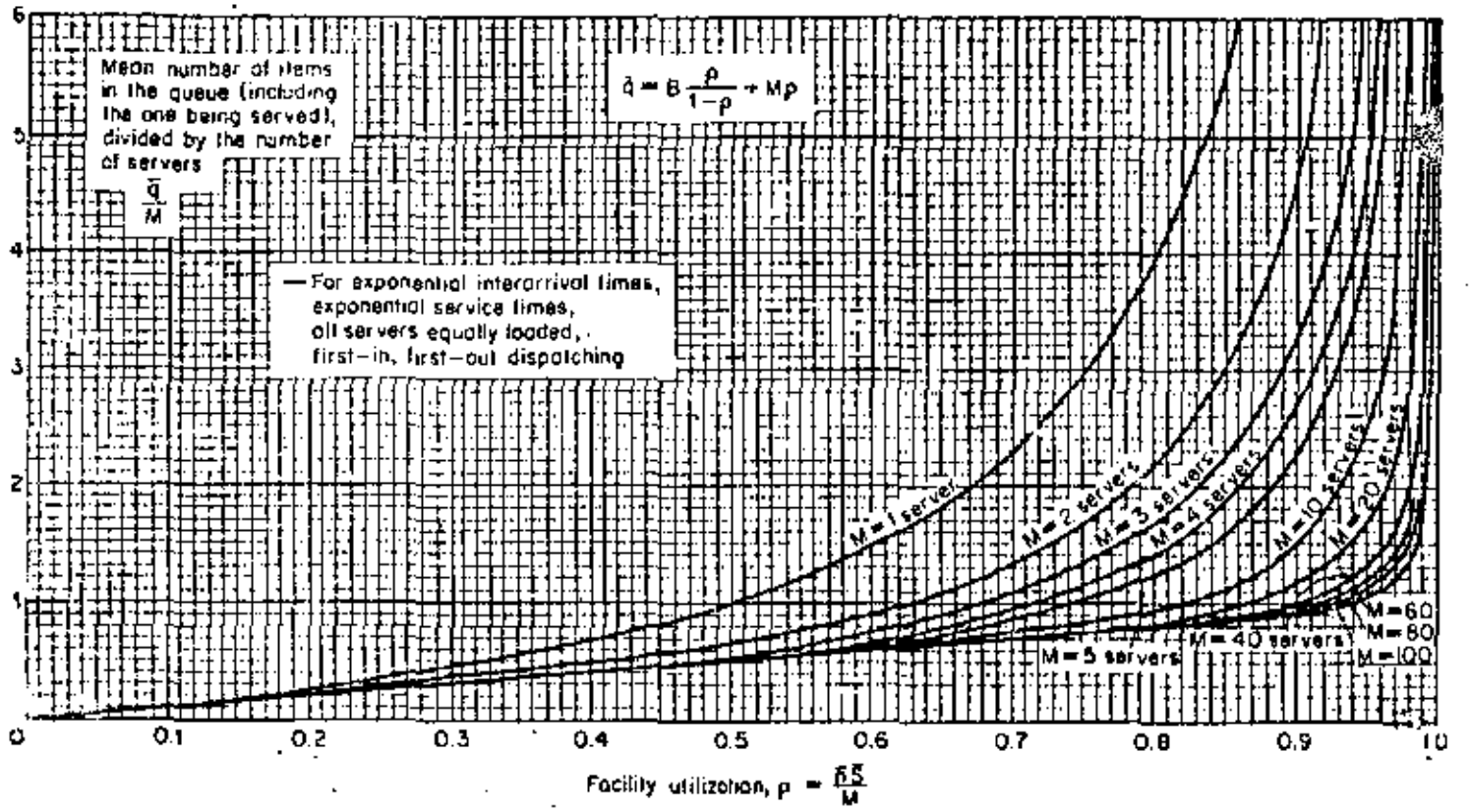


Fig. 3.16.

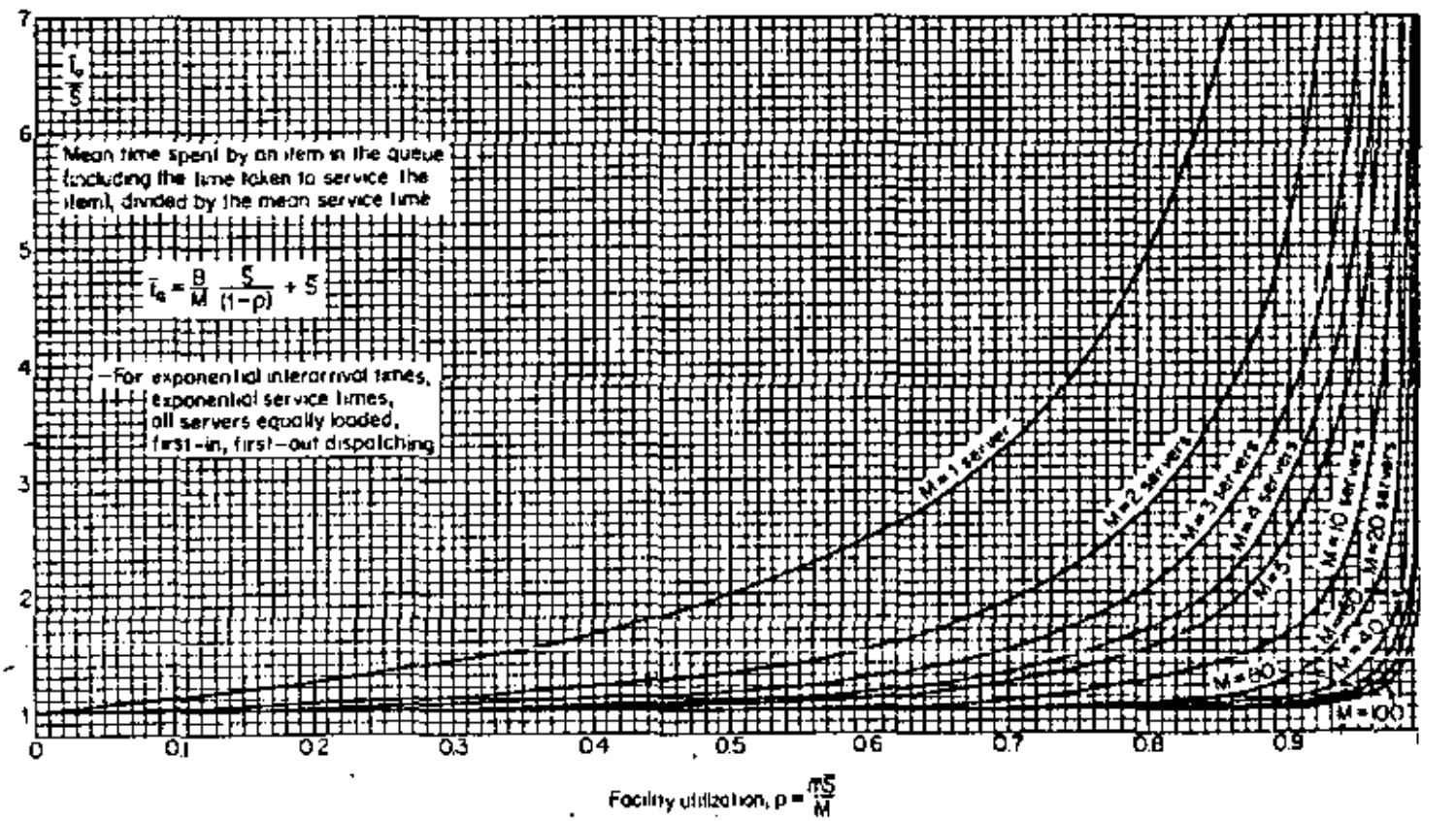


Fig. 3.17.

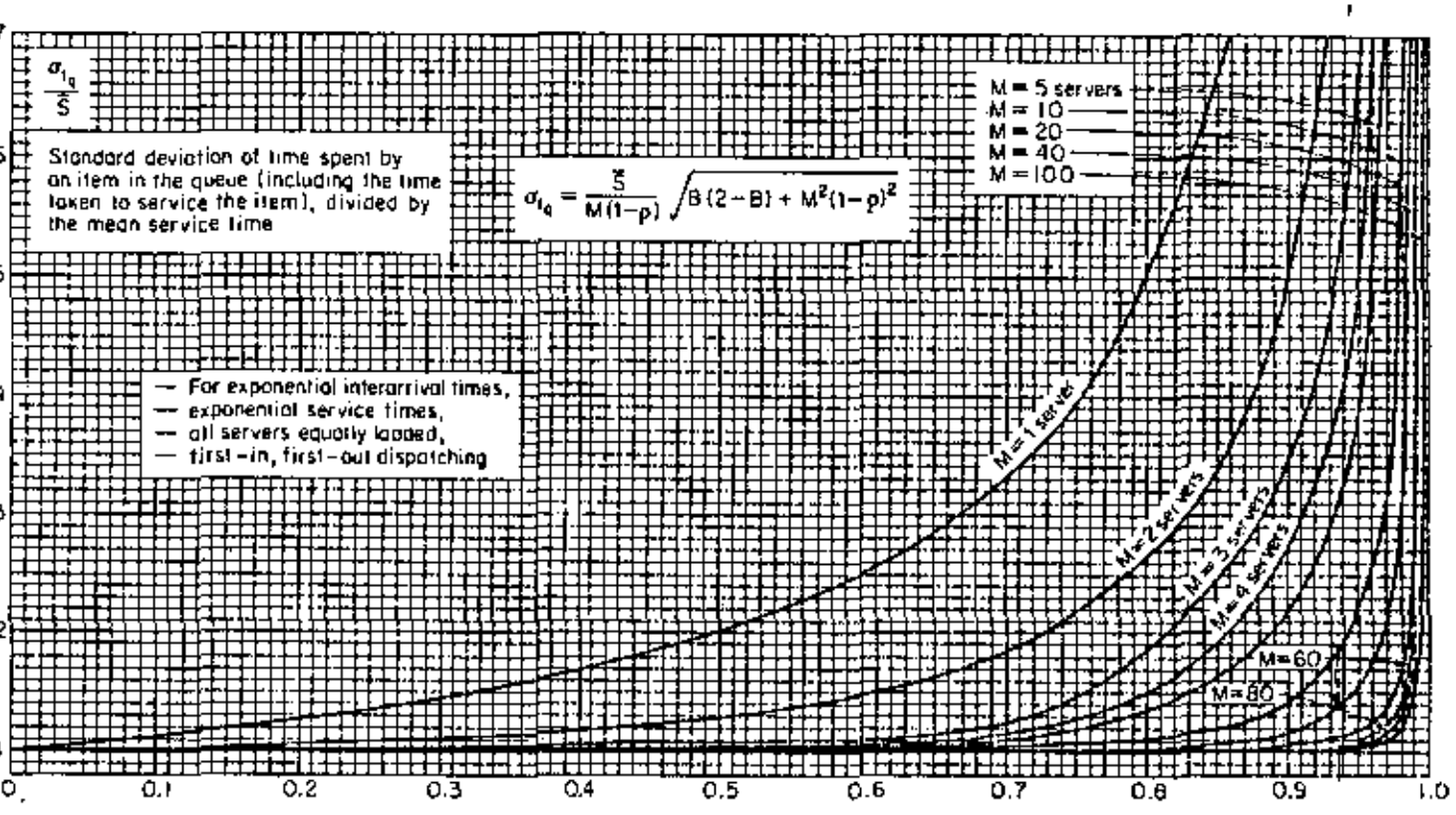


Fig. 3.18.

Con esta información podemos rehacer los cálculos establecidos en los dos apartados anteriores teniendo en cuenta el porcentaje de accesos que irá por cada unidad de control y, en consecuencia, el nivel de ocupación que tendrá cada una de ellas.

Ejemplo: Consideremos un subsistema de seis discos con acceso dual (primaria-secundaria). Al subsistema llegan 40 accesos seg uniformemente repartidos. Los discos giran a 2400 r.p.m., tienen un tiempo medio de seek de 30 mseg y una desviación tipo de 20 mseg. El tiempo de transferencia son 2,3

mseg. Los discos tiene dispositivo de posicionamiento angular con un tiempo medio de 5,68 mseg, un tiempo máximo de 6,25 mseg y un tiempo mínimo de 5,11 mseg. Determinar el tiempo medio de respuesta del subsistema.

Siguiendo las marchas de cálculo expuestas en 3.2.2.1, 3.2.2.2. y 3.2.2.3 podemos hacer:

Tiempo medio de una revolución de los discos

$$2\ L = \frac{60.1000}{2400} = 25 \text{ mseg.}$$

Ocupación conjunta de los dos canales

$$\rho_{CH} = (2,3 + 5,68).\ 40/1000 = 0,3192$$

Porcentaje de accesos por la unidad de control primaria

$$X = \frac{1}{1 + 0,3192} = 0,758$$

Ocupaciones de las unidades de control

$$\rho_1 = 0,3192 \times 0,758 = 0,2420$$

$$\rho_2 = 0,3192 \cdot (1 - 0,758) = 0,0772$$

Números medios de vueltas perdidas a través de cada unidad de control

$$V\rho_1 = \frac{0,242}{1-0,242} = 0,3192 \text{ vueltas}$$

$$V\rho_2 = \frac{0,0772}{1-0,0772} = 0,0837 \text{ vueltas}$$

Tiempo medio de espera por vueltas perdidas

$$W_{vp} = (0,758 \times 0,3192 + (1-0,758) \times 0,0837) \times 25 = 6,56 \text{ mseg.}$$

Tiempo medio de servicio de los discos

$$\frac{1}{\mu_D} = 30 + 12,5 + 6,56 + 5,68 + 2,3 = 57,04 \text{ mseg.}$$

Ocupación de los discos

$$\rho = \frac{57,04}{1000} \times \frac{40}{6} = 0,38$$

Varianza de los tiempos de servicio

$$\sigma_D^2 = 20^2 + \frac{25^2}{12} + \left[\left(\frac{0,242}{(1-0,242)^2} \cdot 25^2 + (0,3192.25)^2\right) \times 0,758 + \right.$$

$$\left. + \left(\frac{0,0772}{(1-0,0772)^2} .25^2 + (0,0837.25)^2\right)(1-0,758) - 6,56^2\right] + \frac{(6,25-5,11)^2}{12}$$

$$= 671,75 \text{ mseg}^2$$

Aplicación de las fórmulas de Khintchine-Pollaczeck

$$t = 57,04 \left[1 + \frac{0,38}{2(1-0,38)} \left(1 + \frac{671,75}{57,04^2}\right)\right] = 78,13 \text{ mseg.}$$

Desviación tipo de los tiempos de respuesta

$$\sigma = 0,8 \cdot 57,04 = 45,63 \text{ mseg.}$$

Siguiendo los procedimientos ya conocidos podríamos determinar la corrección para tener cuenta la incorrección del mode

lo en cuanto al inicio del desplazamiento del brazo.

3.2.3. Subsistemas secuenciales.

La modelización de estos subsistemas no ofrece ninguna dificultad ya que, en general, pueden representarse por la estación de servicio elemental. Además, estos subsistemas no acostumbran a ser los cuellos de botella de un sistema informático y, por lo tanto, su modelización no plantea problamas especiales.

3.2.4. CPU.

Desde los primeros ordenadores en que solo había un programa en la memoria y, por lo tanto, en la CPU no se producían nunca colas hasta los sofisticados sistemas actuales en que el algoritmo para repartir el procesador entre los distintos trabajos se adapta a la carga que está procesando hay todo un abanico de estadios intermedios. Vamos a analizar a continuación el comportamiento de la CPU en sus modos de trabajo más característicos.

3.2.4.1. Modelo batch.

Es el algoritmo de planificación más sencillo y supone que la disciplina de la cola es FIFO, es decir cuando llega un nuevo trabajo se coloca en la cola y cuando alcanza la CPU recibe en forma ininterrumpida todo el servicio requerido.

Para este modelo podemos aplicar directamente las fórmulas referentes a la cola M/M/1 o mejor a la M/M/1//M.

Utilizando la teoría que se deriva de las colas M/G/1 se puede demostrar que el tiempo medio de espera es el mismo para todos los trabajos e independiente del tiempo de servicio que tengan. En consecuencia este tipo de algoritmo no hace ninguna discriminación entre los trabajos según su tiempo de servicio, ya que todos esperan en promedio lo mismo. No se adapta, pues, a trabajos en tiempo compartido, pero es el que provoca menos "overhead".

3.2.4.2. Modelo procesador compartido.

Este modelo conocido en la literatura anglo-sajona como "processor sharing" (PS) o "round robin" (RR) supone que cuando un trabajo sale de la CPU todavía no ha terminado su servicio y vuelve a colocarse en la cola (fig. 3.19) con una probabilidad b. Por otra parte la política de asignación de la CPU dice que se asigna a cada trabajo un quantum de tiempo de servicio ininterrumpido q. El objetivo es que si hay n trabajos en la CPU cada uno de ellos reciba 1/n del tiempo disponible de la CPU por unidad de tiempo.

Aunque puede estudiarse su comportamiento a partir de la cola M/G/1, haremos una deducción algo más larga del tiempo de respuesta, pero que permite comprender mejor el mecanismo de este modelo, aunque simplificando algunas hipótesis. Por ejemplo, hay que tener en cuenta que la probabi-

lidad b no es constante, ya que para un trabajo determinado

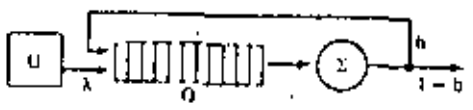


Fig. 3.19.

disminuye a medida que recibe servicio de la CPU. No obstante la supondremos constante.

Supondremos que la probabilidad de que un trabajo requiera k quanta viene expresada por la distribución geométrica

$$\text{Prob}\ (t_s = kq) = (1 - b)\ b^{k-1}$$

Además consideraremos que el estado del sistema viene representado por el número N de trabajos y que la probabilidad del estado N es p_N.

Consideremos que un trabajo que requiere k quanta llega a la CPU y encuentra j trabajos. Marquemos este trabajo que pasará k-1 veces por el camino de retorno antes de salir de la CPU. Una pasada del trabajo marcado se define como el periodo que empieza cuando llega a la cola y termina cuando sale de la estación de servicio después de recibir un quantum. El tiempo para realizar la i-esima pasada lo denominaremos τ_i.

Durante la primera pasada, el trabajo marcado tiene j trabajos por delante. Uno de ellos está recibiendo servicio por lo que abandonará la estación al cabo de α q unidades de tiempo ($0 \leq \alpha \leq 1$). Podemos escribir

$$\tau_1 = \alpha q + (j - 1) q + q$$

Las duraciones de las pasadas sucesivas son múltiplos enteros de q. Si h_i es la duración medida en número de quanta de la i-esima pasada, el trabajo marcado durante esta pasada tiene $h_i - 1$ trabajos por delante, de los cuales $b(h_i - 1)$ volverán a la cola. Observemos que este resultado se basa en la suposición de que la probabilidad de que un trabajo requiera un nuevo quantum es independiente de su historia y por consiguiente concuerda con la distribución geométrica de probabilidad y es válida ya que suponemos también que la distribución de tiempos de servicio no tiene tampoco memoria.

Durante la i-esima pasada del trabajo marcado hay en promedio $\lambda \tau_i$ llegadas. Puesto que $h_i = \bar{\tau}_i/q$, tenemos para $k \geq 3$

$$\bar{\tau}_{i+1} = b(h_i - 1)q + \lambda \bar{\tau}_i q + q = (\lambda q + b)\, \bar{\tau}_i + q(1-b)$$

o de forma equivalente

$$\bar{\tau}_{i+1} = a^{i-2}\, \bar{\tau}_2 + q(1-b)\, \frac{1 - a^{i-2}}{1 - a}$$

donde

$$a = \lambda q + b$$

Mediante el mismo enfoque podemos escribir

$$\bar{\tau}_2 = bjq + \lambda \bar{\tau}_1 \, q + q = \lambda q \, \bar{\tau}_1 + q(1+bj)$$

sustituyendo este valor, nos permite expresar el tiempo medio gastado por el trabajo marcado en el modelo cuando hay j trabajos cuando llega.

$$\bar{t}_k(j) = \sum_{i=1}^{k} \bar{\tau}_i = \bar{\tau}_1 + \frac{q(k-1)}{1-} + q \, \frac{1-a^{k-1}}{1-a}(\lambda \bar{\tau}_1 + bj - \frac{\rho}{1-\rho})$$

donde

$$\rho = \frac{\lambda q}{1-b}$$

Observemos que ρ es el factor de utilización puesto que el tiempo medio de servicio es $q/(1-b)$.

La longitud media de la cola viene dada por

$$\sum_{N=1}^{\infty} (N-1)p_N = \sum_{N=1}^{\infty} Np_N - \sum_{N=1}^{\infty} p_N = \bar{N} - \rho$$

La media de αq es $\rho q/2$, puesto que la estación de servicio está ocupada solo durante una fracción del tiempo igual a ρ. Observando que la longitud de la cola a la llegada del trabajo marcado es j-1, podemos deducir la expresión de $\bar{\tau}_1$

$$\bar{\tau}_1 = q(\bar{N} - \frac{\rho}{2} + 1)$$

Por las fórmulas de Khintchine-Pollaczech y teniendo en cuenta que la distribución geométrica tiene variancia b, podemos escribir que

$$\bar{N} = \rho + \frac{\rho^2(1+b)}{2(1-\rho)}$$

Calculando la media respecto a j de $\bar{t}_k(j)$ tenemos

$$\bar{t}_k = \sum_{j=0}^{\infty} p_j \, \bar{t}_k(j) = \bar{\tau}_1 + \frac{q(k-1)}{1-\rho} + q \, \frac{1-a^{k-1}}{1-a} \cdot (\lambda \bar{\tau}_1 + b\bar{N} - \frac{\rho}{1-\rho})$$

donde podríamos substituir los valores ya calculados y donde observaríamos que si $a < 1$, $\bar{t}_k$ depende del tiempo de servicio kq de forma que tiende a ser lineal al crecer k, con pendiente $1/(1-\rho) > 1$.

El tiempo medio de respuesta será

$$\bar{t} = \sum_{k=1}^{\infty} b^{k-1} \, (1 - b) \, \bar{t}_k$$

Si hacemos además que el quantum q tienda a cero como si la CPU procesara simultáneamente todos los trabajos, tenemos - que b y a tienden a 1, si ρ se mantiene constante. $\bar{N}$ tiende a $\rho/(1-\rho)$ y $\bar{\tau}_1$ tiende a cero. Con todo ello obtenemos

$$\overline{t(t_s)} = \frac{t_s}{1-\rho}$$

De esta fórmula deducimos que el procesador compartido favorece los trabajos con tiempos de ejecución cortos; aunque el tiempo de servicio medio coincida con el del modelo batch.

3.2.4.3. Procesador interrumpible.

Otro interesante algoritmo es el último en llegar primero en recibir servicio en que el trabajo llegado más recientemento interrupme, si ha lugar, el servicio de la CPU desplazando -

el trabajo a la cabeza de la cola hasta que o es interrumpido a su vez o termina su servicio (fig. 3.20).

En este caso podemos calcular el tiempo de respuesta condicional de un trabajo que requiera t_s segundos de servicio como la suma de su tiempo de servicio más su tiempo de espera. En promedio, este último es solo el número medio de trabajos (λt_s) que le interrumpen por el tiempo medio de respuesta de los trabajos que es $\bar{t}_s/(1-\rho)$. Por consiguiente

$$\overline{t(t_s)} = t_s + \frac{\lambda t_s \bar{t}_s}{1-\rho}$$

pero $\rho = \lambda t_s$, de donde

$$\overline{t\ (t_s)} = \frac{t_s}{1-\rho}$$

que es exactamente el mismo que en el caso anterior

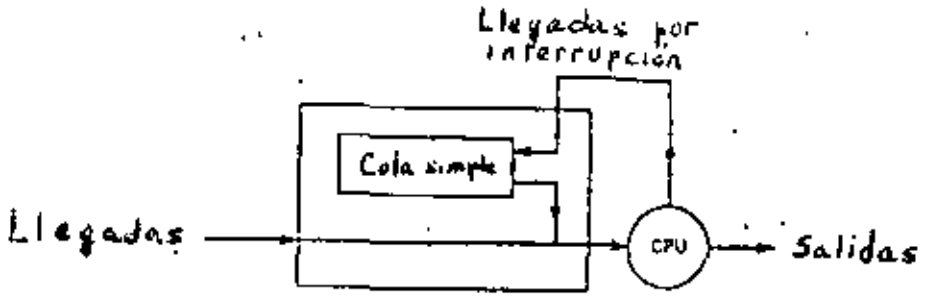


Fig. 3.20.

3.2.5. Memoria.

La modelización de la memoria es especialmente importante en el caso de los sistemas de tiempo real e interactivos,

mientras que en el caso de los sistemas batch debido a la menor variabilidad a lo largo del tiempo de la ocupación de la memoria es más fácil determinar su mapa.

Centrémonos en el caso de sistemas en tiempo real. Ante todo es preciso conocer el tiempo de permanencia de una transacción en memoria, que es igual a la suma de los tiempos de espera en la cola de la CPU, de uso de CPU y de realización de la entrada/salida. Este tiempo de presencia influye, evidentemente, en la memoria necesaria y el nivel de multiprogramación. A su vez, éste incide en la determinación de la dimensión de la cola existente en la CPU.

Evidentemente hay que considerar además el espacio ocupado por el software fijo (sistema operativo, sistema de base de datos, etc.).

Por otro lado, los sistemas de memoria virtual presentan además las complicaciones adicionales que representan la determinación del conjunto de trabajo y la frecuencia de fallo de página, que, a su vez, también están relacionados.

Las principales dificultades en el establecimiento de modelos de unidades centrales reside en la obtención de datos suficientemente correctos. Así, respecto a los programas la longitud de código ejecutado, la dimensión de los programas, y en los sistemas de memoria virtual, el conjunto de trabajo y la frecuencia de fallo de página, y respecto al procesador, el número de instrucciones ejecutadas por unidad de tiempo, no son datos fáciles de obtener. Tal vez

la forma más segura, y eso aún para cada caso concreto, es a través de medidas realizadas con monitores sobre sistemas ya existentes.

A continuación se expone un sencillo ejemplo de modelo de un conjunto CPU-memoria.

Ejemplo: disponemos de una CPU capaz de ejecutar 500.000 instrucciones por segundo. Este sistema atiende transacciones a razón de 4 por segundo, sabiendo que los accesos de I/O tienen un tiempo medio de 75 milisegundos. Determinar la memoria necesaria y el factor de multiprogramación del sistema, sabiendo que las características de las transacciones están resumidas en el cuadro que se expone a continuación

	Accesos	Ocupación (K)	Instrucciones $\times 10^3$	Frecuencia %
T1	5	20	80	37,5
T2	6	25	90	32,5
T3	8	40	110	20
T4	9	50	150	10

Al tratarse de un sistema en tiempo real donde todos los trabajos tienen accesos a la CPU de la misma duración aproximada podemos utilizar sin demasiada diferencia cualquiera de los modelos antes expuestos. En este caso utilizaremos concretamente el modelo batch.

El factor de ocupación de la CPU es el siguiente:

$$\rho = \frac{(0,375 \times 80.000 + 0,325 \times 90.000 + 0,2 \times 110.000 +}{500.000}$$

$$\frac{+ 0,1 \times 150.000)4}{500.000} = 0,77$$

Admitiendo en primera aproximación y como hipótesis muy peyorativa que se pueden aplicar a la cola de la CPU las fórmulas de Khintchine-Pollaczeck con los tiempos de servicio distribuidos exponencialmente, tendremos

$$\frac{1}{\mu} = \frac{0,375 \times 80.000 + 0,325 \times 90.000 + 0,2 \times 110.000 +}{500.000}$$

$$\frac{+ 0,1 \times 150.000}{500.000} = 0,1925 \text{ seg}$$

$$t = \frac{0,1925}{1-0,77} = 0,83696 \text{ seg}$$

El tiempo medio de espera en la cola de la CPU será

$$t_w = 836,96 - 192,5 = 644,46 \text{ mseg}$$

Los tiempos de presencia de cada una de las transacciones serán

$$t_1 = \frac{80.000}{500} + 644,46 + 5 \times 75 = 1179,46 \text{ mseg}$$

$$t_2 = \frac{90.000}{500} + 644,46 + 6 \times 75 = 1274,46 \text{ mseg}$$

$$t_3 = \frac{110.000}{500} + 644,46 + 8 \times 75 = 1464,46 \text{ mseg}$$

$$t_4 = \frac{150.000}{500} + 644,46 + 9 \times 75 = 1619,46 \text{ mseg}$$

El factor de multiprogramación es igual a

$$N = (0,375 \times 1,17946 + 0,325 \times 1,27446 + 0,2 \times 1,46446 + 0,1 \times 1,61946) \times 4 = 5,325$$

que evidentemente se redondea a uno de los enteros más próximos, por ejemplo, 6, lo cual significa que tendremos seis trabajos en memoria y por lo tanto podemos aplicar el modelo de cola M/M/1//6.

Las probabilidades de tener entre 0 y 6 trabajos en la CPU son

$$p_0$$
$$p_1 = 6\, p_0 \frac{\delta}{\mu}$$
$$p_2 = 30\, p_0 \left(\frac{\delta}{\mu}\right)^2$$
$$p_3 = 120\, p_0 \left(\frac{\delta}{\mu}\right)^3$$
$$p_4 = 360\, p_0 \left(\frac{\delta}{\mu}\right)^4$$
$$p_5 = 720\, p_0 \left(\frac{\delta}{\mu}\right)^5$$
$$p_6 = 720\, p_0 \left(\frac{\delta}{\mu}\right)^6$$

de donde

$$p_0 = \frac{1}{1+6\frac{\delta}{\mu}+30\left(\frac{\delta}{\mu}\right)^2+120\left(\frac{\delta}{\mu}\right)^3+240\left(\frac{\delta}{\mu}\right)^4+720\left(\frac{\delta}{\mu}\right)^5+720\left(\frac{\delta}{\mu}\right)^6}$$

Por otro lado podemos escribir que

$$\rho = \frac{\lambda}{\mu} = \frac{6\delta p_0+5\delta p_1+4\delta p_2+3\delta p_3+2\delta p_4+1\delta p_5+0.\delta.p_6}{\mu}$$

y sustituyendo llegamos a la ecución

$$165,6\left(\frac{\delta}{\mu}\right)^6+165,6\left(\frac{\delta}{\mu}\right)^5+82,8\left(\frac{\delta}{\mu}\right)^4+27,6\left(\frac{\delta}{\mu}\right)^3+6,9\left(\frac{\delta}{\mu}\right)^2+1,38\left(\frac{\delta}{\mu}\right)-0,77=0$$

que resolviendo obtenemos

$$\frac{\delta}{\mu} = 0,1815$$

que nos permite calcular las probabilidades

$$p_0 = 0,22973$$

$$p_1 = 0,25017$$

$$p_2 = 0,22703$$

$$p_3 = 0,16483$$

$$p_4 = 0,08975$$

$$p_5 = 0,03258$$

$$p_6 = 0,00591$$

lo cual nos da que el número medio de elementos en el sistema es

$$N = 0 \times 0,22973 + 1 \times 0,25017 + 2 \times 0,22703 + 3 \times$$

$$\times 0,16483 + 4 \times 0,08975 + 5 \times 0,03258 +$$

$$+ 6 \times 0,00591 = 1,75608$$

y el tiempo medio de permanencia en el sistema es

$$\bar{t} = \frac{1,75608}{4} = 0,43902 \text{ seg}$$

y, en consecuencia, el tiempo medio de espera será

$$t_w = 0,43902 - 0,1925 = 0,24652$$

Los tiempos medios de cada transacción serán, ahora,

$t_1 = 0,78152$ seg

$t_2 = 0,87652$ seg

$t_3 = 1,06652$ seg

$t_4 = 1,22152$ seg

A la vista de estos resultados se podría reconsiderar el factor de multiprogramación, cosa que aquí no se ha hecho.

Ahora bien, para determinar el tiempo de respuesta, es decir desde que una transacción llega hasta que termina su ejecución, hay que sumar a estos tiempos el tiempo de espera que sufren cuando encuentran ocupados los seis espacios de memoria para las transacciones. Esto puede modelizarse mediante una estación con seis canales de servicio, cuya saturación y tiempo medio de servicio son respectivamente,

$$t_m = 0,78152 \times 0,375 + 0,375 + 0,87652 \times 0,32 +$$

$$+ 1,06652 \times 0,2 + 1,22152 \times 0,1 = 0,9134 \text{ seg}$$

$$= \frac{4 \times 0,9134}{6} = 0,6089$$

y utilizando, por ejemplo el gráfico de la figura 3.14 tenemos

$t_w = (1,1 - 1) \times 0,9134 = 0,09134$ seg

con lo que los tiempos de respuesta de cada transacción son

$t_1 = 0,78152 + 0,09134 = 0,87286$ seg

$t_2 = 0,87652 + 0,09134 = 0,96786$ seg

$t_3 = 1,06652 + 0,09134 = 1,15786$ seg

$t_4 = 1,22152 + 0,09134 = 1,31286$ seg

El espacio de memoria necesario será

$$M = (0,375 \times 0,78151 \times 20 + 0,325 \times 0,87652 \times 25 + + 0,2 \times 1,06652 \times 40 + 0,1 \times 1,22152 \times 50)\ 4 = 110,5\ K$$

3.2.5. Red de comunicaciones.

La red de comunicaciones puede adoptar numerosas y variadas estructuras, por lo que es difícil indicar una metodología general para su análisis. No obstante en cada caso particular es posible establecer modelos adaptados a la estructura existente basados siempre en las colas que se produzcan en la red.

3.3. Modelos globales de sistemas informáticos.

Hasta ahora hemos estado considerando cada uno de los sub-- sistemas separadamente. No obstante es evidente que existe una interrelación entre ellos que no es posible ignorar. Es por ello que si unimos los modelos de colas estudiados hasta aquí obtenemos una red colas.

Diremos que tenemos una red de colas cuando los trabajos que salen de una estación de servicio van a parar:

. O en forma determinista a otro sistema de colas.
. O al exterior del sistema.
. O en forma aleatoria con probabilidades determinadas a uno de entre varios sistemas de colas o al exterior.

En estas redes el número de trabajos que existen en su interior puede ser fijo (red cerrada) o puede producirse un flu-

jo de trabajos que pueden entrar y salir por distintos puntos (red abierta). En ambos casos su estudio es más complejo si se quiere tratar de forma exacta. En muchos casos se puede hallar un tratamiento analítico, exacto o aproximado, del conjunto, en otros, sin embargo, es preciso recurrir a métodos de simulación.

3.3.1. Métodos analíticos exactos.

Jackson (referencia JACK 63) dio una primera forma de tratar redes de colas, que fue ampliada recientemente por Baskett, Chandy,Muntz y Palacios (BCMP) (referencia BASK 75) incluyendo todos los casos susceptibles del mismo tratamiento.

Los sistemas que consideramos contienen un número arbitrario pero finito, N, de estaciones de servicio. Hay un número arbitrario pero finito, R, de clases distintas de trabajos. Es decir, un trabajo de clase r que sale de la estación de servicio i requerirá servicio de la estación j en la clase s con una probabilidad que indicaremos por $P_{i,r;j,s}$. Esta matriz la designaremos por $P = \left[P_{i,r;j,s}\right]$ y sea n_{ir} el número de trabajos de clase r en la estación i.

Las estaciones de servicio pueden ser de los cuatro tipos siguientes

Tipo 1. La estación i tiene un solo canal con tiempo de servicio exponencial de tiempo medio $1/\mu_i(n_i)$ idéntico para todas las clases, siendo n_i ($n_i = \sum_r n_{ir}$) el número de trabajos -

en la estación y la disciplina de la cola, FIFO. (Los discos se asimilan a una estación de este tipo).

Tipo 2. La estación tiene un solo canal, la disciplina de servicio es de procesador compartido (es decir, cuando hay n trabajos en la estación de servicio, cada uno recibe servicio a razón de 1/n de segundo cada segundo) y cada clase de trabajo tiene una distribución de tiempos de servicio - que puede ser distinta y arbitraria. (Una estación de este tipo es la CPU).

Tipo 3. El número de canales en la estación de servicio es mayor o igual que el número máximo de trabajos que puede - haber en la estación en un instante cualquiera y cada clase de trabajo tiene una distribución de tiempos de servi-- cio que puede ser distinta y arbitraria (Una estación de - este tipo son los terminales interactivos).

Tipo 4. La estación de servicio tiene un solo canal, la - disciplina de cola es LIFO con interrupción del último en llegar y cada clase de trabajo tiene una distribución de - tiempos de servicio que puede ser distinta y arbitraria. - (Una estación de este tipo es la interrupción de la CPU).

El proceso de llegada a la red sigue una distribución de - Poisson de parámetro $\lambda(n)$, donde $n = \sum_i n_i$ es el número de trabajos que hay en el sistema representado por la red.

Además es preciso calcular las frecuencias efectivas de llegada de cada clase de trabajo a cada estación de servicio

e_{ir}. Ello se logra resolviendo el siguiente sistema de ecuaciones:

$$e_{js} = \sum_{r=1}^{R} \sum_{i=1}^{N} e_{ir} P_{i,r;j,s} + q_{js}$$

de donde q_{js} es la probabilidad de entrada desde el exterior en la estación j de un trabajo de clase s. Si la red es cerrada, evidentemente todas las q_{js} son nulas y nos encontramos con un sistema de ecuaciones homogéneo del que hay que hallar una solución distinta de la idénticamente nula. Este sistema de ecuaciones es equivalente (cambiando la notación) al planteado al estudiar el análisis operacional.

Para deducir los resultados es preciso plantear las ecuaciones globales de equilibrio del sistema en régimen estacionario. Es decir para todos los estados S_i

$$\sum_{\forall S_j} P(S_j) \left[\text{frecuencia de paso de } S_j \text{ a } S_i\right] =$$

$$= P(S_i) \left[\text{Frecuencia de salida de } S_i\right]$$

También pueden deducirse a partir de las ecuaciones independientes de equilibrio en las que se iguala la frecuencia de flujo de entrada en un estado por entrada de un trabajo en una estación al flujo de salida de ese estado por salida de un trabajo de esa misma estación. Es decir se plantea el equilibrio a nivel de estación. La suma de las ecuaciones independientes nos lleva a la ecuación global, es decir nos dan una condición suficiente de equilibrio.

La solución de cualquiera de estas ecuaciones nos lleva a - determinar la probabilidad del estado como producto de las probabilidades del estado de cada estación. No obstante, el enunciado completo del teorema de BCMP lleva a una expre- - sión notablemente compleja, por lo que expondremos aquí sólo las consecuencias que permiten una más fácil comprensión y aplicación.

Definiremos como estado del sistema el número de trabajos - de cada clase en cada estación de servicio. Más formalmente el estado del sistema viene dado por $(y_1, y_2, \ldots, y_N)$, donde - $y_i = (n_{i1}, n_{i2}, \ldots n_{ir})$. Sea $1/\mu_{ir}$ el tiempo medio de servicio de un trabajo de clase r en la estación i. Entonces, para una red de estaciones de servicio que puede ser abierta, cerrada o mixta y en que éstas pueden ser tipo 1,2,3 ó 4, - las probabilidades del estado en equilibrio vienen dadas - por

$$P(y_1, y_2, \ldots y_n) = Cd(n)\ g_1(y_1) g_2(y_2) \ldots g_n(y_n)$$

donde:

Si la estación es de tipo 1, entonces

$$g_i(y_i) = n_i! \left(\prod_{r=1}^{R} \frac{1}{n_{ir}!}\ e_{ir}^{\ n_{ir}} \right) \left(\frac{1}{\mu_i} \right)^{n_i}$$

Si la estación es de tipo 2 ó 4, entonces

$$g_i(y_i) = n_i! \prod_{r=1}^{R} \frac{1}{n_{ir}!} \left(\frac{e_{ir}}{\mu_{ir}} \right)^{n_{ir}}$$

Si la estación es de tipo 3, entonces

$$g_i(y_i) = \prod_{r=1}^{R} \frac{1}{n_{ir}!} \left(\frac{e_{ir}}{\mu_{ir}}\right)^{n_{ir}} ,$$

$$d(n) = \prod_{i=0}^{n=1} \lambda(i)$$

si la red es abierta y d(n) = 1 si es cerrada.

C es la constante de normalización para lograr que la suma de todas las probabilidades sea igual a 1. Esta constante de normalización puede determinarse sin ningún problema si tratamos un sistema cerrado, pues el número de estados distintos en que puede hallarse el sistema es finito y por lo tanto la suma también.

En caso de tratarse de una red abierta, solo podremos determinar sin riesgo de error de truncamiento de la suma infinita cuando esa suma tenga una expresión analítica. Tal es el caso, cuando la llegada no depende del estado del modelo y el estado del sistema viene caracterizado por $(n_1, n_2, \ldots n_n)$, es decir por el número total de trabajos n_i que hay en cada estación i. Entonces:

$P_i(n_i) = (1 - \rho_i)\rho_i^{n_i}$ si la estación es de tipo 1,2 ó 4.

$P_i(n_i) = e^{-\rho_i} \rho_i^{n_i}/n_i!$ si la estación es de tipo 3 y

donde

$$\rho_i = \sum_{r=1}^{R} \lambda e_{ir}/\mu_i \quad \text{si la estación es de tipo 1.}$$

$$\rho_i = \sum_{r=1}^{R} \lambda e_{ir}/\mu_{ir} \quad \text{si la estación es de tipo 2,3 ó 4.}$$

Aunque la utilizacion de este método solo aparece cuando se dispone de un software que permita aplicarlo pues exige una gran cantidad de cálculos vamos a hacer una aplicación directa a un par de ejemplos sencillos.

Ejemplo: Consideremos el ejemplo de la figura 3.21, en que tenemos un sistema cerrado, con dos clases de trabajos y cinco estaciones de servicio, la 1 de tipo 2 (procesador compartido) y las otras cuatro de tipo 1 (discos).

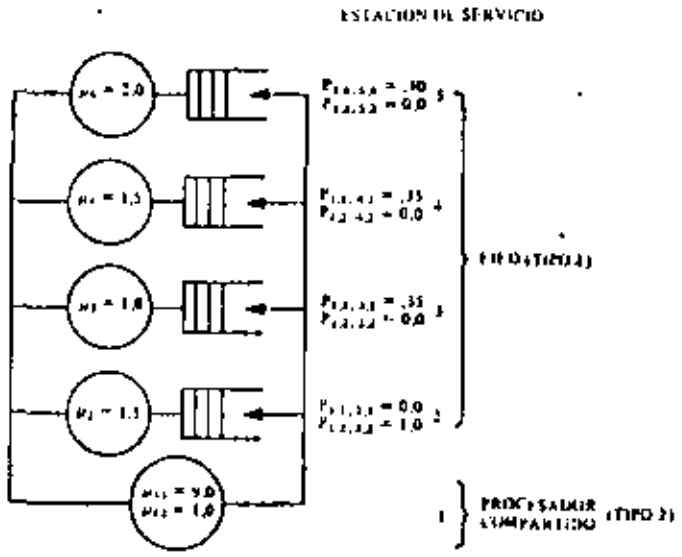


Fig. 3.21.

En función de las probabilidades de transición podemos plantear el sistema de ecuaciones homogéneo para determinar las frecuencias de visita.

$$e_{11} = e_{21}+e_{31}+e_{41}+e_{51} \qquad e_{12} = e_{22}+e_{32}+e_{42}+e_{52}$$

$$e_{21} = 0 \cdot e_{11} \qquad e_{22} = 1 \cdot e_{12}$$

$e_{31} = 0,35\ e_{11}$ $\qquad$ $e_{32} = 0 \cdot e_{12}$

$e_{41} = 0,35\ e_{11}$ $\qquad$ $e_{42} = 0 \cdot e_{12}$

$e_{51} = 0,3\ e_{11}$ $\qquad$ $e_{52} = 0 \cdot e_{12}$

de donde hallamos

$e_{11} = 1;\ e_{21} = 0;\ e_{31} = 0,35;\ e_{41} = 0,35;\ e_{51} = 0,3$

$e_{12} = 1;\ e_{22} = 1,\ e_{32} = 0\ ;\ e_{42} = 0\ ;\ e_{52} = 0.$

Si consideramos inicialmente el caso en que hay un trabajo de clase 2 y ninguno de clase 1, el sistema solo podrá estar en los estados

$S_1 = [(0,1),(0,0),(0,0),(0,0),(0,0)]$

$S_2 = [(0,0),(0,1),(0,0),(0,0),(0,0)].$

Por lo tanto

$$P(S_1) = C \cdot \left[1!\left[\frac{1}{0!}\left(\frac{1}{9}\right)^0\right]\left[\frac{1}{1!}\left(\frac{1}{1}\right)^1\right]\right]\left[0!\left[\frac{1}{0!}\,0^0\right]\left[\frac{1}{0!}\,1^0\right]\left(\frac{1}{1,5}\right)^0\right]$$
$$\left[0!\left[\frac{1}{0!}\,0,35^0\right]\left[\frac{1}{0!}\,0^0\right]\left(\frac{1}{1}\right)^0\right]\left[0!\left[\frac{1}{0!}\,0,35^0\right]\left[\frac{1}{0!}\,0^0\right]\left(\frac{1}{1,5}\right)^0\right]$$
$$\left[0!\left[\frac{1}{0!}\,0,3^0\right]\left[\frac{1}{0!}\,0^0\right]\left(\frac{1}{2}\right)^0\right] = C$$

$$P(S_2) = C \left[0!\left[\frac{1}{0!}\left(\frac{1}{9}\right)^0\right]\left[\frac{1}{0!}\left(\frac{1}{1}\right)^0\right]\right]\left[1!\left[\frac{1}{0!}\,0^0\right]\left[\frac{1}{1!}\,1^1\right]\left(\frac{1}{1,5}\right)^1\right]$$

$$\left[0!\left[\frac{1}{0!}\ 0,35^0\right]\left[\frac{1}{0!}\ 0^0\right]\left(\frac{1}{1}\right)^0\right]\left[0!\left[\frac{1}{0!}\ 0,35^0\right]\left[\frac{1}{0!}\ 0^0\right]\left(\frac{1}{1,5}\right)^0\right]$$
$$\left[0!\left[\frac{1}{0!}\ 0,3^0\right]\left[\frac{1}{0!}\ 0^0\right]\left(\frac{1}{2}\right)^0\right] = \frac{C}{1,5}$$

Para determinar la constante de normalización hacemos

$$P(S_1) + P(S_2) = 1$$

$$C + \frac{C}{1,5} = 1$$

de donde $C = 0,6$, y

$$P(S_1) = 0,6 \quad \text{y} \quad P(S_2) = 0,4$$

Los factores de utilización, es decir la probabilidad que en un dispositivo $n_i \geq 1$, serán

$$\rho_1 = 0,6, \quad \rho_2 = 0,4, \quad \rho_3 = 0, \quad \rho_4 = 0, \quad \rho_5 = 0$$

Si consideramos ahora el caso en que hay un trabajo de cada clase, el sistema solo podrá estar en los estados

$$S_1 = [(1,1),(0,0),(0,0),(0,0),(0,0)]$$
$$S_2 = [(0,1),(0,0),(1,0),(0,0),(0,0)]$$
$$S_3 = [(0,1),(0,0),(0,0),(1,0),(0,0)]$$
$$S_4 = [(0,1),(0,0),(0,0),(0,0),(1,0)]$$
$$S_5 = [(1,0),(0,1),(0,0),(0,0),(0,0)]$$
$$S_6 = [(0,0),(0,1),(1,0),(0,0),(0,0)]$$
$$S_7 = [(0,0),(0,1),(0,0),(1,0),(0,0)]$$
$$S_8 = [(0,0),(0,1),(0,0),(0,0),(1,0)]$$

Por lo tanto las probabilidades serán (solo están considerados los factores de las estaciones que tienen algún trabajo, siendo los demás 1):

$$P(S_1)=C\left[2!\left[\frac{1}{1!}\left(\frac{1}{9}\right)^1\right]\left[\frac{1}{1!}\left(\frac{1}{1}\right)^1\right]\right] 1.1.1.1 = \frac{2C}{9}$$

$$P(S_2)=C\left[1!\left[\frac{1}{0!}\left(\frac{1}{9}\right)^0\right]\left[\frac{1}{1!}\left(\frac{1}{1}\right)^1\right]\right] 1.\left[1!\left[\frac{1}{1!}0,35^1\right]\left[\frac{1}{1!}\,0^0\right]\left(\frac{1}{1}\right)^1\right] 1.1 = \frac{7C}{20}$$

$$P(S_3)=C\left[1!\left[\frac{1}{0!}\left(\frac{1}{9}\right)^0\right]\left[\frac{1}{1!}\left(\frac{1}{1}\right)^1\right]\right] 1.1\left[1!\left[\frac{1}{1!}0,35^1\right]\left[\frac{1}{0!}\,0^0\right]\left(\frac{1}{1,5}\right)^1\right].1 = \frac{7C}{30}$$

$$P(S_4)=C\left[1!\left[\frac{1}{0!}\left(\frac{1}{9}\right)^0\right]\left[\frac{1}{1!}\left(\frac{1}{1}\right)^1\right]\right] 1.1.1.\left[1!\left[\frac{1}{1!}0,3^1\right]\left[\frac{1}{0!}\,0^0\right]\left(\frac{1}{2}\right)^1\right] = \frac{3C}{20}$$

$$P(S_5)=C\left[1!\left[\frac{1}{1!}\left(\frac{1}{9}\right)^1\right]\left[\frac{1}{0!}\left(\frac{1}{1}\right)^0\right]\right]\left[1!\left[\frac{1}{0!}\,0^0\right]\left[\frac{1}{1!}\,1^1\right]\left(\frac{1}{1,5}\right)^1\right].1.1.1 = \frac{2C}{27}$$

$$P(S_6)=C1\left[1!\left[\frac{1}{0!}\,0^0\right]\left[\frac{1}{1!}\,1^1\right]\left(\frac{1}{1,5}\right)^1\right]\left[1!\left[\frac{1}{1!}0,35^1\right]\left[\frac{1}{0!}0^0\right]\left(\frac{1}{1}\right)^1\right]1.1 = \frac{7C}{30}$$

$$P(S_7)=C1\left[1!\left[\frac{1}{0!}\,0^0\right]\left[\frac{1}{1!}\,1^1\right]\left(\frac{1}{1,5}\right)^1\right] 1\left[1!\left[\frac{1}{1!}0,35^1\right]\left[\frac{1}{0!}0^0\right]\left(\frac{1}{1,5}\right)^1\right]1 = \frac{7C}{45}$$

$$P(S_8)=C1\left[1!\left[\frac{1}{0!}\,0^0\right]\left[\frac{1}{1!}\,1^1\right]\left(\frac{1}{1,5}\right)^1\right] 1.1\left[1!\left[\frac{1}{1!}0,3^1\right]\left[\frac{1}{1!}0^0\right]\left(\frac{1}{2}\right)^1\right] = \frac{C}{10}$$

Para determinar la constante de normalización haremos

$$P(S_1)+P(S_2)+P(S_3)+P(S_4)+P(S_5)+P(S_6)+P(S_7)+P(S_8) = 1$$

$$\frac{2C}{9}+\frac{7C}{20}+\frac{7C}{30}+\frac{3C}{20}+\frac{2C}{27}+\frac{7C}{30}+\frac{7C}{45}+\frac{C}{10} = 1$$

de donde $C = \frac{27}{41}$ y

$P(S_1)=0,1463$; $P(S_2)=0,2305$; $P(S_3)=0,1537$; $P(S_4)=0,0988$;

$P(S_5)=0,0488$; $P(S_6)=0,1537$; $P(S_7)=0,1024$; $P(S_8)=0,0658$

Los factores de utilización serán

$$\rho_1 = P(S_1) + P(S_2) + P(S_3) + P(S_4) + P(S_5) = 0,6781$$

$$\rho_2 = P(S_5) + P(S_6) + P(S_7) + P(S_8) = 0,3707$$

$$\rho_3 = P(S_2) + P(S_6) = 0,3842$$

$$\rho_4 = P(S_3) + P(S_7) = 0,2561$$

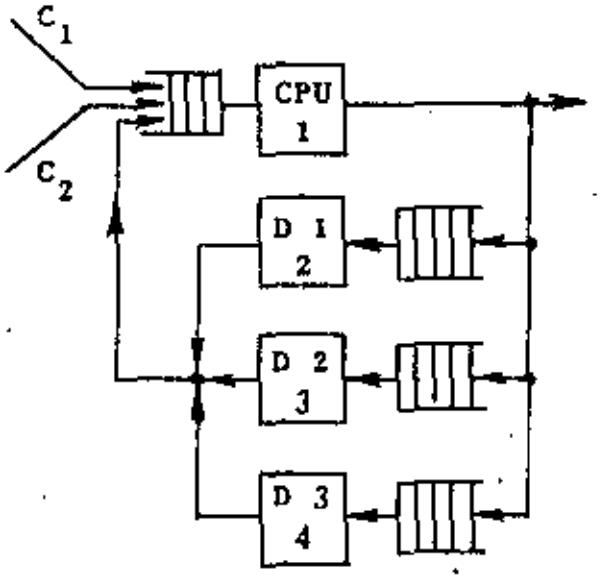
$$\rho_5 = P(S_4) + P(S_8) = 0,1646$$

En la tabla que sigue se exponén los niveles de ocupación de las cinco estaciones de servicio cuando hay un trabajo de clase 2 y un número variable de clase 1.

	1	2	3	4	5
0	0,6	0,4	0	0	0
1	0,678	0,371	0,384	0,256	0,165
2	0,720	0,352	0,606	0,404	0,260
3	0,744	0,339	0,743	0,495	0,318
4	0,759	0,330	0,831	0,554	0,356
5	0,769	0,324	0,888	0,592	0,381
6	0,775	0,321	0,926	0,617	0,397
7	0,779	0,318	0,951	0,634	0,407

Si desearamos conocer otras características de nuestro modelo lo podríamos hacer sin ninguna dificultad a partir de la probabilidad de los distintos estados.

Ejemplo: Consideramos el sistema de la figura 3.22 en que tenemos un sistema abierto con dos clases de trabajos y trece estaciones de servicio: una CPU (tipo 2) y tres grupos de 4 discos idénticos (tipo 1).



C_1 $\lambda_1 = 0,05$, $\mu_{11} = 200$

C_2 $\lambda_2 = 5$, $\mu_{12} = 300$

$\mu_2 = 25$

$\mu_3 = 20$

$\mu_4 = 15$

Fig. 3.22

$p_{11;21} = 0,45$ $p_{21;11} = 1$

$p_{11;31} = 0,36$ $p_{31;11} = 1$

$p_{11;41} = 0,18$ $p_{41;11} = 1$

$p_{12;22} = 0,25$ $p_{22;12} = 0,8$ $p_{22;11} = 0,2$

$p_{12;32} = 0,15$ $p_{32;12} = 1$

$p_{12;42} = 0,40$ $p_{42;12} = 1$

El sistema de ecuaciones para determinar las frecuencias de visita será:

$$e_{11} = e_{21} + e_{31} + e_{41} + 0,2\ e_{22} + 0,05$$

$$e_{21} = 0,45\ e_{11}$$

$$e_{31} = 0,36\ e_{11}$$

$$e_{41} = 0,18\ e_{11}$$

$$e_{21} = 0,8\ e_{22} + e_{32} + e_{42} + 5$$

$$e_{22} = 0,25\ e_{12}$$

$$e_{32} = 0,15\ e_{12}$$

$$e_{42} = 0,40\ e_{12}$$

de donde hallamos

$$e_{11} = 110 \qquad e_{12} = 20$$

$$e_{21} = 49,5 \qquad e_{22} = 5$$

$$e_{31} = 39,6 \qquad e_{32} = 3$$

$$e_{41} = 19,8 \qquad e_{42} = 8$$

A partir de ahí podemos determinar directamente los factores de utilización de cada estación

$$\rho_1 = \frac{110}{200} + \frac{20}{300} = 0,617$$

$$\rho_2 = \frac{(49,5+5)/4}{25} = 0,545$$

$$\rho_3 = \frac{(39,6+3)/4}{20} = 0,533$$

$$\rho_4 = \frac{(19,8+8)/4}{15} = 0,463$$

y el número medio de elementos en cada estación

$$\bar{n}_1 = \frac{0,617}{1-0,617} = 1,611$$

$$\bar{n}_2 = \frac{0,545}{1-0,545} = 1,198$$

$$\bar{n}_3 = \frac{0,533}{1-0,533} = 1,141$$

$$\bar{n}_4 = \frac{0,463}{1-0,463} = 0,862$$

Si suponemos además que los tiempos de servicio son todos exponenciales podemos escribir además que

$$\rho_{11} = \frac{110}{200} = 0,55 \qquad \rho_{12} = \frac{20}{300} = 0,067$$

$$\rho_{21} = \frac{49,5/4}{25} = 0,495 \qquad \rho_{22} = \frac{5/4}{25} = 0,05$$

$$\rho_{31} = \frac{39,6/4}{20} = 0,495 \qquad \rho_{32} = \frac{3/4}{20} = 0,038$$

$$\rho_{41} = \frac{19,6/4}{15} = 0,330 \qquad \rho_{42} = \frac{8/4}{15} = 0,133$$

$$\bar{n}_{11} = \frac{0,55}{1-0,617} = 1,436 \qquad \bar{n}_{12} = \frac{0,067}{1-0,617} = 0,175$$

$$\bar{n}_{21} = \frac{0,495}{1-0,545} = 1,088 \qquad \bar{n}_{22} = \frac{0,05}{1-0,545} = 0,110$$

$$\bar{n}_{31} = \frac{0,495}{1-0,533} = 1,060 \qquad \bar{n}_{32} = \frac{0,038}{1-0,533} = 0,081$$

$$\bar{n}_{41} = \frac{0,330}{1-0,463} = 0,616 \qquad \bar{n}_{42} = \frac{0,133}{1-0,463} = 0,246$$

y por la fórmula de Little los tiempos medios de permanencia en cada estación

$$\bar{t}_{11} = \frac{1,436}{110} \times 10^3 = 13,05 \text{ mseg} \qquad \bar{t}_{21} = \frac{0,067}{20} 10^3 = 3,35 \text{ mseg}$$

$$\bar{t}_2 = \frac{1,198}{(49,5+5)/4} 10^3 = 87,93 \text{ mseg}$$

$$\bar{t}_3 = \frac{1,141}{(39,6+3)/4} 10^3 = 107,14 \text{ mseg}$$

$$\bar{t}_4 = \frac{0,862}{(19,8+8)/4} 10^3 = 124,03 \text{ mseg}$$

El número medio de ciclos para los trabajos de cada clase son

$$C_1 = \frac{0,45+0,36+0,18}{1-(0,45+0,36+0,18)} = \frac{0,99}{0,01} = 99$$

$$C_2 = \frac{0,25+0,15+0,40}{1-(0,25+0,15+0,40)} = \frac{0,80}{0,20} = 4$$

Por lo tanto los tiempos medios de respuesta para cada clase de trabajo son

$$t_{R1} = (99+1)13,05+99 \frac{0,45}{0,99} 87,93+ \frac{0,36}{0,99} 107,14+ \frac{0,18}{0,99}124,03 =$$

$$= 11.351,43 \text{ mseg}$$

$$t_{R2} = (4+1)3,35+4 \frac{0,25}{0,80} 87,93+ \frac{0,15}{0,80}107,14+ \frac{0,40}{0,80}124,03 =$$

$$= 455,08 \text{ mseg.}$$

3.3.2. Métodos de simulación.

Todos los modelos que se han expuesto pueden estudiarse por simulación sometiendo el modelo al funcionamiento estipulado y obteniendo estadísticas de su comportamiento. Puede incluso mejorarse la calidad del modelo liberándonos de las restricciones impuestas para conseguir un tratamiento analítico (tipos de distribuciones, direccionamiento aleatorio de los trabajos, etc.).

3.3.2.1. Metodología.

El objetivo de la simulación con un modelo consiste en investigar el comportamiento del sistema en estado estacionario estimando las características de las variables que caracterizan la respuesta del sistema.

Para alcanzar este objetivo se presentan dos dificultades básicas, comunes a todos los procesos de simulación de sistemas discretos.

- Suponiendo que el sistema que estudiamos tenga un estado estacionario, antes de alcanzarlo pasará por un régimen transitorio determinado por las condiciones iniciales y, en general, es difícil determinar cuando se ha entrado en el estado estacionario.

- Por otro lado existe una dependencia estadística entre las sucesivas observaciones del estado estacionario que, en general, no son independientes sino que tienen una correlación apreciable.

Los métodos de análisis del estado estacionario pueden dividirse en dos clases principales:

- de diseño de la simulación de forma que se obtengan observaciones estadísticamente independientes que permitan aplicar los métodos de la estadística clásica al análisis de los resultados obtenidos por la simulación

- de análisis directo de los datos correlacionados, utilizando los métodos de análisis de series temporales.

Dentro de la primera clase hay tres procedimientos clásicos para obtener observaciones estadísticamente independientes:

a) Método de las repeticiones, que consiste en llevar a cabo K ejecuciones independientes del modelo de simulación con m observaciones en cada una. La independencia se consigue utilizando distintas sucesiones de números aleatorios en cada ejecución con el mismo estado inicial.

b) Método de las medias de los lotes que consiste en llevar a cabo una ejecución de longitud N del modelo dividiendo el conjunto de las observaciones de la ejecución en K lotes de m observaciones cada uno. Los valores medios de las observaciones de cada lote si se elige m suficientemente grande no estarán prácticamente correladas. Para que fueran independientes debería cumplirse además que estuvieran distribuidas normalmente, aunque se consideran independientes si se da solo la condición de no correlación. Con frecuencia se combinan ambos métodos.

c) Método regenerativo, que puede utilizarse si el sistema lo es y se dice que lo es cuando existe una sucesión de puntos, llamados de regeneración, tales que en ellos el modelo se halla cada vez en las mismas condiciones. La ejecución de la simulación se divide entonces en una secuencia de bloques independientes igualmente distribuidos.

El problema del método regenerativo es el de determinar si existen puntos de regeneración y para ello es preciso que el sistema vuelva con una cierta frecuencia a un estado específico y que el tiempo medio entre estos pasos sea finito. Las ventajas de este método residen en que la agrupación aleatoria de las observaciones que proporcionan los puntos de regeneración, producen bloques independientes idénticamente distribuidos desde el inicio de la simulación lo cual permite evitar los problemas de dependencia estadística entre las sucesivas observaciones y de determinación del estado estacionario, todo lo cual nos permite definir mejores estimadores.

3.3.2.2. Estimadores.

Si trabajamos a partir de las repeticiones mezcladas con las medias de los lotes consideraremos como i-esima observación de la j-esima repetición el resultado del i-esimo lote de forma que el valor medio $\bar{x}_j$ y la variancia σ_j^2 de la j-esima repetición son

$$\bar{x}_j = \frac{1}{n} \sum_{i=1}^{n} x_{ij}$$

$$\sigma_j^2 = \frac{1}{n-1} \sum_{i=1}^{n} (x_{ij} - x_j)^2$$

La combinación de los resultados de los valores medios independientes de las p repeticiones da como estimaciones del valor medio $\bar{x}$ y de la variancia σ^2.

$$\bar{x} = \frac{1}{p} \sum_{j=1}^{p} \bar{x}_j$$

$$\sigma^2 = \frac{1}{p} \sum_{j=1}^{p} \sigma_j^2$$

de donde resulta un intervalo de confianza

$$\bar{x} \pm \frac{\sigma}{\sqrt{n}} t_{n-1,\alpha/2}$$

Ahora bien, es preciso tener en cuenta que si los datos de partida de la simulación están correlados el estado estacionario corresponderá al de un proceso estocástico estacionario covariante. En esta situación el efecto de la correlación no afecta a la estimación del valor medio, pero se ha de llevar a cabo una corrección en la estimación de la variancia, teniendo en cuenta las covariancias.

Un estimador puntual de las autocovariancias entre las observaciones es

$$R_k = \frac{1}{n-k} \sum_{t=1}^{n-k} (\bar{x}_t - \bar{\bar{x}})(\bar{x}_{t+k} - \bar{\bar{x}})$$

$$\bar{\bar{x}} = \frac{1}{n} \sum_{t=1}^{n} \bar{x}_t$$

y a partir de él se puede construir el siguiente estimador no sesgado de la variancia

$$\sigma^2 = \frac{n}{n-k}\left[R_o + 2\sum_{s=1}^{k-1}\left(1 - \frac{s}{n}\right) R_s\right], \quad k < n$$

Si nos hallamos, por el contrario, en un caso regenerativo, sea $0 < E_1 < E_2 < \ldots$ la sucesión de puntos de regeneración - que nos definirán las longitudes de cada ciclo como

$$\alpha_i = E_{i+1} - E_i$$

que son el número de entidades que ha abandonado el sistema durante el i-esimo ciclo. Entonces si Y_i es la suma de los tiempos de espera en el i-esimo ciclo, un estimador del tiempo medio de espera será

$$E(w) = \frac{E(Y)}{E(\alpha)}$$

Para un total de n ciclos, obtendríamos el conjunto de observaciones

$$\{Y_1, Y_2, \ldots Y_n\} \quad y \quad \{\alpha_1, \alpha_2, \ldots \alpha_n\}$$

En general, puesto que los ciclos son independientes e idénticamente distribuidos, también los son las Y_i que acostumbran a estar fuertemente correladas con las α_i. Para estas observaciones tendríamos los estimadores

$$\bar{Y} = \frac{1}{n}\sum_{i=1}^{n} i \qquad \bar{\alpha} = \frac{1}{n}\sum_{i=1}^{n} \alpha_i$$

Un estimador puntual clásico de E(w) es

$$r_c = \frac{\bar{Y}}{\bar{\alpha}}$$

con un intervalo de confianza $r_c \pm d_c$ que es

$$d_c = t_{1-\frac{\alpha}{2}} \frac{S_c}{\bar{\alpha}\sqrt{n}}$$

de donde

$$S_c^2 = S_{11}^2 - 2r_c S_{12}^2 + r_c^2 S_{22}^2$$

siendo

S_{11}^2 = variancia de las muestras $Y_j = \frac{1}{n-1} \sum_{i=1}^{n} (Y_i - Y)^2$

S_{22}^2 = variancia de las muestras $\alpha_j = \frac{1}{n-1} \sum_{i=1}^{n} (\alpha_i - \bar{\alpha})^2$

S_{12}^2 = covariancia de las muestras $(\ _i, \alpha_i) = \frac{1}{n-1} \sum_{i=1}^{n} (\ _i - \ )(\alpha_i - \bar{\alpha})$

Este esquema supone que la dimensión de la muestra es suficientemente grande para que, según los resultados del teorema central del límite, se pueda aproximar la distribución de la muestra por una ley normal. En caso contrario r_c proporciona una estimación sesgada. Para muestras pequeñas métodos de Jacknife proporciona mejores estimadores puntuales:

$$r_j \pm d_j$$

siendo

$$r_j = \frac{1}{n} \sum_{i=1}^{n} \theta_i$$

$$\theta_i = n \left(\frac{Y}{\alpha}\right) - (n-1) \frac{\sum_{j \neq i} Y_i}{\sum_{j \neq i} \alpha_i}$$

$$y \qquad d_j = t_{n-1,\alpha/2} \frac{S_j}{\sqrt{n}}$$

$$S_j = \frac{1}{n-1} \sum_{i=1}^{n} (\theta_i - r_j)^2$$

3.3.3. Métodos aproximados.

Tanto los métodos analíticos como los de simulación requieren grandes potencias de cálculo para obtener resultados por lo que han aparecido un gran número de métodos aproximados que nos permiten obtener resultados en unos tiempos de cálculo más razonables. A continuación se expone en forma resumida el fundamento de algunos de ellos.

3.3.3.1. Método de difusión.

Este método planteado inicialmente por I. Kolbayashi y extendido luego E. Gelenbe parte de las siguientes hipótesis:

- Red abierta con M estaciones.
- R clases de clientes.
- Estaciones FIFO.
- La clase r (r = 1, ... R) tiene
  - . flujo de llegada de frecuencia $\lambda_{0,r}$
  - . coeficiente de variación entre llegadas $Ka_{0,r}$
  - . reparto $q_{i,r}$
  - . distribución general de los tiempos de servicio $G_r^i(x)$
  - . tasa de servicio en la estación i: $\mu_{i,r}$
  - . coeficiente de variación del servicio $Ks_{i,r}$
  - . direccionamiento markoviano: $p_{ir,js}$

El principio del método de cálculo se basa en las fases siguientes

a) Estimación del flujo de llegadas a la estación i para la clase r, es decir determinación de

. tasa de llegadas λ_{ir}

. coeficiente de variación del tiempo entre llegadas Ka_{ir}

b) Resolución por difusión de la cola i.

Para llevar a cabo estas dos fases es preciso llevar a cabo los siguientes pasos

a1) Cálculo de λ_{ir}

Por el principio de conservación del flujo

$$\lambda_{ir} = \lambda_{0r}\, q_{ir} + \sum_{j} \sum_{s} \lambda_{js}\, q_{js;ir} \quad ; \quad \forall i \quad \forall r$$

de donde obtenemos valores únicos de λ_{ir} pues el sistema es abierto. De ahí

$$\rho_{ir} = \lambda_{ir} / \mu_{ir}$$

$$\rho_{i} = \sum_{r} \rho_{ir}$$

$$\lambda_{i} = \sum_{r} \lambda_{ir}$$

$$\Pi_{ir} = \lambda_{ir} / \lambda_{i}$$

a2) Cálculo de Ka_{ir}

Para ello supondremos que los procesos de llegada y de salida para cada clase y cada cola son procesos de renovación. Si τ_i es el tiempo entre dos salidas consecutivas de la estación i, podemos escribir

$$\tau_i = \begin{cases} S_i & \text{con probabilidad } \rho_i \\ S_i + A_i & \text{con probabilidad } 1 - \rho_i \end{cases}$$

$$E(\tau_i) = E(S_i) + E(A_i)(1 - \rho_i)$$

$$E(\tau_i) = \sum_r \frac{\pi_{ir}}{\mu_{ir}} + \frac{1 - \rho_i}{\lambda_i}$$

$$E(\tau_i^2) = E(S_i^2) + 2(1-\rho_i)\,\lambda_i^{-1} \sum_r \frac{\pi_{ir}}{\mu_{ir}} + (1-\rho_i)\,E(\lambda_i^2)$$

Si C_i es el coeficiente de variación del tiempo entre dos salidas y K_i lo es para dos llegadas en la estación i

$$C_i = \lambda_i \sum_r \rho_{ir}(Ka_{ir}+1)/\mu_{ir} + (1-\rho_i)(K_i+1+2\rho_i) - 1$$

$$K_i = \lambda_i^{-1} \sum_{j=0}^{M} \left[(C_j - 1)\, P_{ij} + 1\right] \lambda_j\, P_{ji}$$

$$P_{ji} = \sum_r \sum_s (\lambda_{jr} / \lambda_j)\, p_{jr;is}.$$

A partir de donde podemos determinar

$$Ka_{ir} = (K_i - 1)\,\pi_{ir} + 1$$

b) La resolución por difusión de la cola i supone, mediante determinadas hipótesis, que la función f(x,1) de la densidad de probabilidad del número de elementos en la cola a lo largo del tiempo cumple la ecuación de difusión

$$-\frac{\partial f_i}{\partial t} - b_i \frac{\partial f_i}{\partial x} + \frac{1}{2}\alpha_i \frac{\partial^2 f_i}{\partial x^2} + \lambda_i P_i(t) f_i(x-1) = 0$$

$$\frac{d}{dt} P_i(t) = -\lambda_i P_i(t) + \lim_{x \to 0^+} \left[-b_i f_i + \frac{1}{2}\alpha_i \frac{\partial f_i}{\partial x} \right]$$

donde $P_i(t)$ probabilidad de sistema vacio en t

$$b_i = \lambda_i - \mu_i$$

$$\alpha_i = \lambda_i K_i + \mu_i^3 V_i$$

Si estamos en régimen permanente estacionario

$$P_i = \lim_{t \to \infty} P_i(t) = 1 - \rho_i$$

$$f_i(x) = \lim f_i(x,t) = \begin{cases} \rho_i (1 - e^{\gamma_i x})\alpha, & x \leq 0 \\ \rho_i (1 - e^{-\gamma_i x}) e^{\gamma_i x}, & x > 0 \end{cases}$$

de donde $\gamma_i = 2b_i / \alpha_i$

A partir de estas ecuaciones se obtiene

$$\bar{n}_i = \rho_i \left[1 + \frac{\mu_i K_i + K a_i}{2(1-\rho_i)} \right] \quad \text{(número medio de elementos)}$$

$T_i = \bar{n}_i / \lambda_i$ (tiempo medio de respuesta)

$W_i = W_{ir} = T_i - \mu_i^{-1}$ (tiempo medio de espera)

$T_{ir} = W_i + \mu_{ir}^{-1}$ (tiempo medio de respuesta de la clase r)

Lo cual nos permite afirmar que es un método de solución muy rápido y que da, en general, una buena aproximación.

3.3.3.2. Métodos de descomposición.

En estos métodos que se basan en los trabajos de Courtois se basan en descomponer una red cerrada en una cola y el resto en una sub-red que se reduce a una cola equivalente y se estudia entonces el sistema constituido por dos colas cerradas una sobre la otra. Este método es especialmente adecuado a las redes de tipo BCMP donde da una solución exacta o en redes quasi-descomponibles en las que existe un débil acoplamiento entre la sub-red y la cola aislada.

3.3.3.3. Métodos iterativos.

Estos se basan en los trabajos de Chandy y Marie. En ellos y para sistemas cerrados se estudia el flujo de entrada en cada cola procedente del resto de la red. A partir de él se estudia el comportamiento de red. La suma del número medio de elementos en cada estación debe coincidir con el número total de elementos en el sistema. Si no coinciden se corrige e itera hasta alcanzar la coincidencia.

4. CARACTERIZACION DE LA CARGA.

4.1. Introducción.

La eficiencia de un sistema puede discutirse sólo en el contexto de lo que se le pide que haga. Las aplicaciones del usuario una vez traducidas a programas, pueden caracterizarse por el tipo y la cantidad de recursos asignados: el total de los requerimientos de recursos representa la carga total sobre el sistema.

El método básico para medir la eficiencia de un sistema consiste en observarlo y medirlo mientras está ejecutando unos trabajos determinados. Los resultados obtenidos dependen, por tanto, de la carga de trabajo sobre el sistema; puede decirse que la eficiencia del sistema es pues su reacción ante una carga.

La carga de trabajos de un sistema es generalmente irreproducible en su composición exacta.

Aunque pueda hablarse de unas condiciones estadísticas estacionarias que permitan efectuar experimentos de evaluación o sintonización del sistema trabajando con la carga real, este tipo de medidas se enfrentan a tres clases de problemas (SVOB 76):

- dificultad en la determinación del intervalo de tiempo en el que puede considerarse estadísticamente estacionario (y por tanto reproducible) el sistema. Dicho intervalo quedará determinado por la frecuencia en la que la carga exhibe cambios significativos. Por ejemplo, procesos de periodicidad -

diaria, semanal, etc.

- nulo control sobre las condiciones del experimento, y por tanto, imposibilidad de relacionar valores de las medidas y resultados con los parámetros de la carga.

- cantidad de datos a analizar. Por ejemplo, considérese el caso de decidir como estacionaria la carga de un trimestre de explotación.

La puesta en marcha de nuevas aplicaciones y su sustitución o eliminación, los hábitos de los usuarios en cuanto a nuevas técnicas de programación, manejo de archivos, etc., hacen - que la carga de trabajos sea difícilmente reproducible aun - en intervalos largos.

Por todo ello se han desarrollado modelos que pueden usarse para caracterizar (modelizar) la carga. Las condiciones que se exigen a estas caracterizaciones son:

- poder definir cargas representativas para la evaluación comparativa de diferentes sistemas

- poder definir cargas reproducibles y controlables

- reducir la gran cantidad de datos a analizar

- proporcionar datos útiles para la modelización del sistema

Este trabajo describe un modelo de caracterización de la carga en términos de los recursos utilizados. Otros modelos que consideran la distribución temporal de las demandas a los recursos y su localización pueden estudiarse aplicando teorías de procesos estocásticos (AGRA 76).

4.2. Definición de las variables de carga (SVOB 76).

Se usan comúnmente medidas de:

- tiempo de CPU usado por el trabajo
- número de operaciones de Entrada/Salida por trabajo
- tiempo de CPU gastado en procesar una tarea "sólo-CPU"
- tiempo de Entrada/Salida gastado en procesar una tarea "sólo E/S"
- tiempo entre dos requerimientos sucesivos de servicio a un componente del sistema
- prioridad asignada al trabajo
- tiempo en el que un trabajo es incapaz de recibir servicio de la CPU
- cantidad de memoria requerida por un trabajo
- número de páginas de un trabajo que han de mantenerse en memoria ("working-set")
- tiempo en el que las referencias a memoria de un trabajo permanecen dentro del ámbito de una o más páginas
- tiempo que el usuario necesita para generar un nuevo requerimiento (transacción) desde un terminal
- tiempo de proceso por transacción/tiempo de respuesta del usuario
- número de usuarios interactivos simultáneamente en el sistema

- número de trabajos o tareas recibiendo servicio o esperando en colas
- frecuencia relativa de diferentes tipos de instrucciones que el sistema debe ejecutar ("mix").

La mayoría de las variables descritas suelen obtenerse de los sistemas de contabilidad del sistema operativo.

Si se dispone de monitores software, pueden definirse medidas más adecuadas para concentrar el análisis en zonas específicas.

4.3. Estudio estadístico de los resultados.

Una forma común de representar los resultados es utilizar histogramas o gráficos, en que un eje corresponde a valores numéricos de la variable representada y el otro número de veces que dichos valores aparecen.

Por ejemplo, sobre memoria utilizada podría obtenerse la distribución de la fig. 4.1.

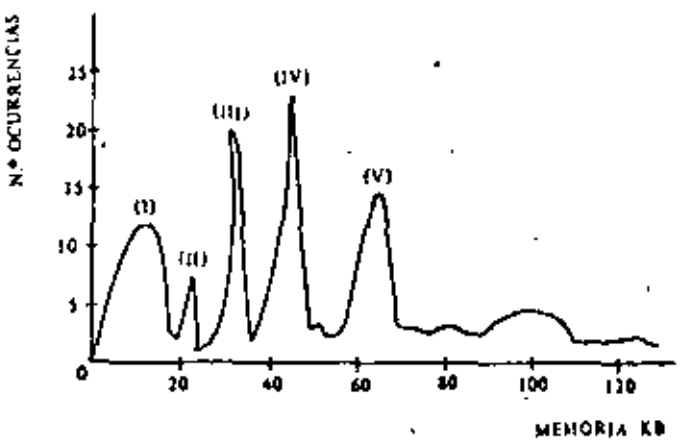


Fig. 4.1.

De la simple observación puede deducirse que en el intervalo de tiempo en el qie se han recogido las medidas, las demandas de memoria de los programas se han agrupado en cinco grupos bien diferenciados I, II, III, IV y V. La identificación de estos grupos con programas es inmediata ya que los nombres de compiladores, montador, programas de utilidad y programas de aplicación se registran normalmente junto con los datos de contabilidad.

Para cada variable estudiada puede dibujarse su histograma correspondiente (Figs. 4.2, 4.3, 4.4 y 4.5).

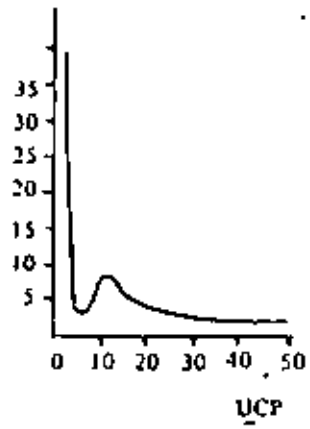


Fig. 4.2.

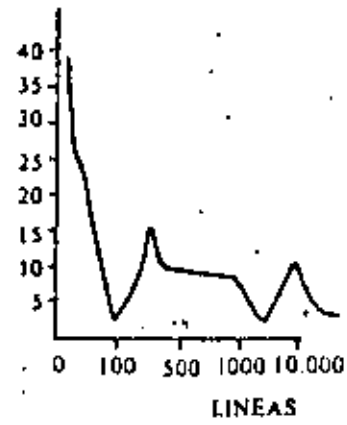


Fig. 4.3.

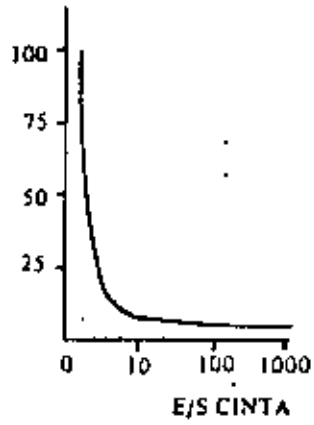


Fig. 4.4.

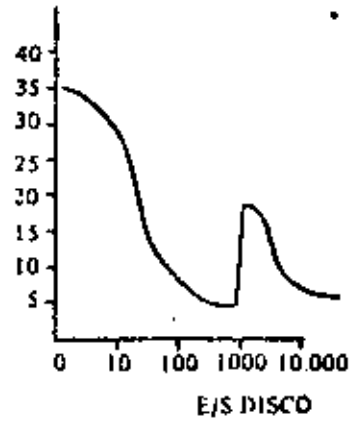


Fig. 4.5.

Puede formalizarse la situación diciendo que para cada programa, un vector de la forma:

$$R_i = \{\text{variable}_1 \ldots \text{variable}_n\{$$

caracteriza las demandas de recursos del sistema R_i recibe el nombre de vector de recursos.

El siguiente paso consiste en clasificar los programas según sus vectores; es decir, según los valores de las variables

que representan las demandas a recursos. Dos programas pertenecerán al mismo grupo si sus vectores de recursos son similares.

Formalizando aún más el problema: se trata de clasificar los vectores R_i en grupos de características similares en el espacio definido por las variables consideradas.

Para mejor comprender el método de clasificación consideremos un sencillo problema, por ejemplo, el grupo de trabajos caracterizados por la memoria ocupada y las cintas asignadas que muestra la tabla de la figura 4.6.

Número de trabajo	Memoria ocupada (K)	Cintas asignadas
1	54	1
2	88	0
3	120	2
4	110	2
5	64	0
6	68	1
7	112	1
8	56	0
9	64	1
10	90	4
11	60	1
12	110	1
13	60	0
14	116	1
15	118	2
16	150	2

Fig. 4.6.

Dada la dimensión de la muestra por simple examen llegamos a la siguiente agrupación:

1. 1,6,9 y 11 alrededor de 60 K y 1 cinta
2. 5,8 y 13 alrededor de 60 K sin cintas
3. 7,12 y 14 alrededor de 114 K y 1 cinta
4. 3,4 y 15 alrededor de 114 K y 2 cintas

siendo imposible agrupar los trabajos 2,10 y 16.

Para lograr tal agrupación existen numerosos métodos conocidos bajo el nombre genérico de métodos de "clusteving", que provocan otros, entre los cuales se halla el del escalado.

Consideremos la representación de los trabajos en la figura 4.7, en la que se han elegido escalas idénticas para ambas magnitudes, por lo que las diferencias de cintas asignadas se muestran insignificantes. Para orillar esta dificultad debemos escalar de distinto modo las dos variables que caracterizan el trabajo.

Para determinar la similaridad de los trabajos en el plano (en general en un espacio de n dimensiones) es preciso fijar una medida que nos permita discernir los trabajos que están próximos entre sí de los que puedan pertenercer a otros grupos. Según los casos pueden adoptarse distintas medidas pero en el caso que nos ocupa como medida de la similitud se utiliza la distancia euclídea entre dos puntos del espacio definido por las variables del vector R_i

$$d_{ij} = \left[\sum_{k=1}^{p} (X_{ik} - X_{jk})^2 \right]^{1/2}$$

con P = número de variables

X_{ij} = medida i-ésima de la variable j

Para P = 2 esta expresión no es más que el teorema de Pitágoras.

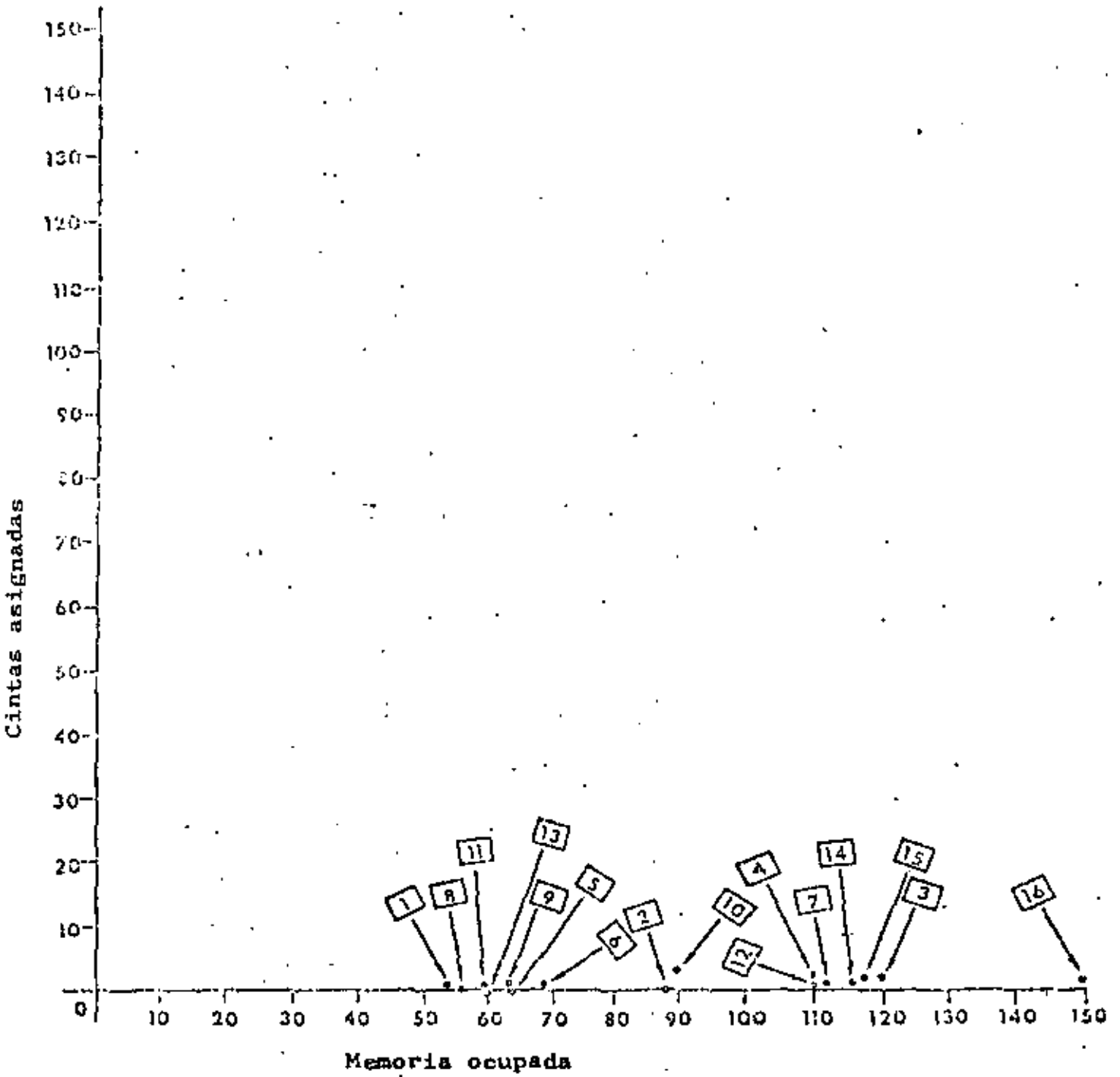


Fig. 4.7.

Si utilizamos directamente esta medida nos hallaremos con los siguientes resultados si calculamos las distancias entre los trabajos 12 y 14

$$d_{12,14} = \sqrt{(110 - 116)^2 + (1 - 1)^2} = 6$$

mientras entre los trabajos 2 y 10

$$d_{2,10} = \sqrt{(88 - 90)^2 + (0 - 4)^2} = 4,47$$

No obstante es evidente que mientras los dos primeros son muy similares los dos últimos son muy distintos. Ello es debido a usar escalas idénticas para las dos magnitudes que caracterizan cada trabajo. Es preciso pues escalarlas diferentemente, lo cual puede hacerse por distintos métodos siendo el más frecuentemente utilizado el de la normalización, es decir transformar las variables de acuerdo con

$$z_{ij} = \frac{x_{ij} - x_j}{\sigma_j}$$

donde

$$x_j = \frac{\sum_{i=1}^{N} x_{ij}}{N}$$

$$\sigma_j = \sqrt{\frac{\sum_{i=1}^{N} x_{ij}^2}{N-1} - \frac{\left(\sum_{i=1}^{N} x_{ij}\right)^2}{N(N-1)}}$$

En nuestro ejemplo obtendríamos

variable	P	$\sum_i X_{ij}$	$\sum_i X_{ij}^2$	X_i	j
memoria ocupada	16	1440	142936	90	29,82
cintas asignadas	16	19	39	1,19	1,05

En la tabla de la figura 4.8 tenemos las magnitudes características de nuestros trabajos una vez escalados

Trabajo número	memoria ocupada		cintas asignadas	
	original	escalada	original	escalada
1	54	-1.21	1	-0.18
2	88	-0.07	0	-1.13
3	120	1.01	2	0.77
4	110	0.67	2	0.77
5	64	-0.87	0	-1.13
6	68	-0.74	1	-0.18
7	112	0.74	1	-0.18
8	56	-1.14	0	-1.13
9	64	-0.87	1	-0.18
10	90	0.00	4	2.68
11	60	-1.01	1	-0.18
12	110	0.67	1	-0.18
13	60	-1.01	0	-1.13
14	116	0.87	1	-0.18
15	118	0.94	2	0.77
16	150	2.01	2	0.77

Fig. 4.8.

Las distancias antes calculadas se convierten ahora en

$$d_{12,14} = \sqrt{(0,67 - 0,87)^2 + (-0,18-(-0,18))^2} = 0,2$$

$$d_{2,10} = \sqrt{(-0,07 - 0,00)^2 + (-1,13-2,68)^2} = 3,81$$

que ya pone de manifiesto las diferencias en el sentido que deseabamos, lo cual queda corroborado en la representación gráfica de la figura 4.9.

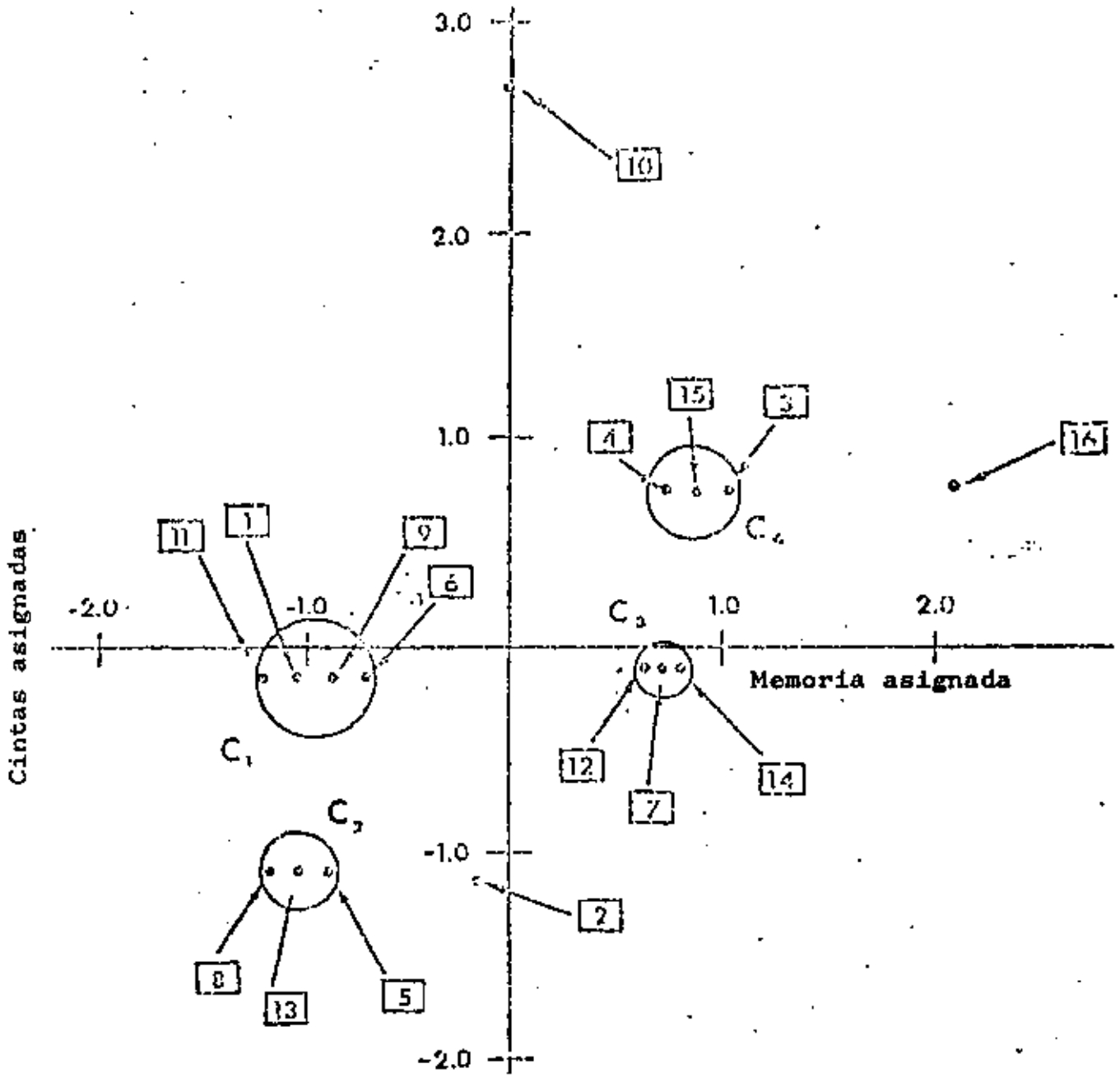


Fig. 4.9.

No obstante cuando se trata de clasificar varios centenares o millares de trabajos caracterizados por más de dos variables la representación gráfica que ahora nos ha permitido - establecer la clasificación, deja de ser útil y es preciso recurrir a otros métodos más sistemáticos.

La idea básica de estos métodos es la de asignar cada vector a la clase correspondiente de modo que se maximice la "calidad" de la partición.

El criterio más sencillo es tratar de minimizar globalmente la suma de los cuadrados de las distancias de los puntos al centro de su clase. La mejor partición, es decir la que cumple el criterio citado, se podría conseguir por la enumeración exhaustiva a partir de un conjunto de N elementos en C clases. Esto lleva a cifras del orden de $C^N/C!$ casos que - obligan a buscar otros métodos. De entre ellos existen los de tipo heurístico y los de tipo jerárquico.

En los de tipo heurístico se inicia fijando el número de clases y sus centros al azar. Los puntos se asignan a la clase cuyo centro está más próximo. El centro tiene por coordenadas

$$Y_{kj} = \frac{1}{n_k} \sum_{i=1}^{n_k} X_{ij} \quad (j = 1,2,\ldots P;\ k=1,2,\ldots C_{max})$$

de donde n_k es el número de trabajos asignados al grupo k y C_{max} el número inicial de clases.

A continuación se mueven los puntos de una clase a otra con el fin de satisfacer el criterio de minimización ya citado

$$S_C = \frac{1}{N} \sum_{k=1}^{C} \sum_{i=1}^{n_k} \sum_{j=1}^{P} (X_{ij} - Y_{kj})^2$$

En estos movimientos se cambian también los centros de las clases.

Una vez acabada la primera iteración se reduce el número de clases. El par de clases que hay que mezclar se halla buscando la combinación que minimiza el incremento de la suma de los cuadrados de las distancias de los puntos a los centros de las clases. Hecho esto se mueven los puntos y se recalculan los centros de las clases como antes. Este proceso continúa hasta alcanzar el mínimo número de clases especificados.

No obstante se plantea la determinación del número de clases adecuado que puede resolverse utilizando los test de chi-cuadrado o de Kolmogovov-Smirnov. Este número adecuado de clases C_n es el que produce una agrupación natural de los datos. Intuitivamente también podemos hallar C_n; S_c aumenta monotonamente a medida que se reduce C, aumentando lentamente cuando se acerca a C_n y creciendo rápidamente cuando nos separamos de C_n al reducir C.

En el método jerárquico se van agrupando los puntos por orden de proximidad sustituyéndolos por un nuevo punto de peso la suma de los pesos de los puntos agrupados (Partiendo de pesos unitarios o ponderados según algún criterio) y si-

tuado en el baricentro. Este criterio se sigue aplicando mientras que el radio de cada clase (distancia del centro al punto más alejado de la clase) se mantiene inferior a una determinada cantidad, que al irla variando nos permite estimar la calidad y la estabilidad de la clasificación.

Existen además muchos otros algoritmos de clasificación en ambos grupos de métodos (heurísticos y jerárquicos) que pueden hallarse en las referencias.

4.4. Definición de combinaciones de trabajos (ARTIS 1976).

En cualquier instante t, la combinación (mix) de trabajos en el sistema puede representarse por un vector M_t en el que la i-ésima componente es el número de trabajos activos pertenecientes al grupo i

$M_t = \{2,0,4\}$ indica que hay dos trabajos del grupo 1, ninguno del grupo 2 y 4 del grupo 3 (para una descripción con tres grupos) activos en el instante t.

Se definen las combinaciones predominantes P_j como aquellas que representan un conjunto de vectores M_{tj} idénticos. A cada P_j se le puede asignar un peso según la cantidad de vectores M_{tj} en el total del intervalo de evaluación. Se obtiene por tanto una descripción de los tantos por ciento en que una determinada mezcla de trabajos está cargando el sistema.

Un reducido ejemplo es el siguiente: Los registros de contabilidad expresan la hora de comienzo y final del trabajo en

una instalación que ha clasificado sus trabajos en diez clases:

Hora de comienzo	Hora de final	Nombre progr.	Grupo	Partición
090332	090341	IPTDISC	1	BG
090256	090442	DITTO	3	F2
090341	090506	LIBRARIAN	3	BG

En este caso los vectores M_t para intervalos de 20 segundos son:

$M_1 = (0,0,1,0,0,0,0,0,0,0)$

$M_2 = (1,0,1,0,0,0,0,0,0,0)$

$M_3 = (1,0,2,0,0,0,0,0,0,0)$

$M_4 = (0,0,2,0,0,0,0,0,0,0)$

$M_5 = (0,0,2,0,0,0,0,0,0,0)$

$M_6 = (0,0,2,0,0,0,0,0,0,0)$

$M_7 = (0,0,1,0,0,0,0,0,0,0)$

$P_1\ (M_1,M_7) = 2/7 = 0,285$

$P_2\ (M_2) = 1/7 = 0,142$

$P_3\ (M_3) = 1/7 = 0,142$

$P_4\ (M_4,M_5,M_6) = 3/7 = 0,430$

4.5. Bancos de ensayos (BENCHMARK).

Se define el término "banco de ensayo" como un punto de referencia a partir del cual pueden efectuarse medidas (SVOB 76). El método de la clasificación de la carga de trabajos en grupos y en mezclas de grupos define el conjunto de valores de la carga bajo los cuales opera el sistema. Un cambio en las características del software o del hardware pueden evaluarse en términos de los nuevos valores de servicio y utilización que se espera obtener. Por consiguiente, para una misma carga de trabajo, dos sistemas diferentes deben dar valores diferentes en sus medidas de servicios.

Esta ha sido y es la metodología usual seguida para la elección de ordenadores de marcas diferentes o modelos diferentes. Generalmente se seleccionan una o varias aplicaciones con un volumen promedio de datos a tratar y simplemente se ejecutan en los sistemas a comparar. Los cocientes entre valores obtenidos para las diferentes medidas son los índices de comparación deseados.

La clasificación en grupos puede ayudar en la selección de las aplicaciones significativas y volúmenes adecuados: se trata ahora de cargar el sistema bajo estudio con las mezclas de grupos que se consideren representativos. Evidentemente en sistemas con multiprogramación ello presupone que el total de los programas ejecutándose en las diferentes particiones den precisamente las mezclas con las que se intenta probar el sistema. El problema que se plantea es otra vez el de reproducir la carga.

4.1. Programas sintéticos y "mix".

La carga se ha definido como una secuencia de demandas a servicios del sistema; es decir a recursos del sistema. Para la mayoría de problemas de evaluación, el conocimiento de estas secuencias de demandas es suficiente. Un programa sintético es un programa que simula el uso de los recursos del sistema característico de un determinado grupo (ARTIS 76).

Un ejemplo sencillo del uso de programas sintéticos es el siguiente: se desea evaluar una máquina junto con su compilador FORTRAN. Para ello pueden construirse dos tipos de programas sintéticos. El primero de ellos tomará tiempos de las operaciones características de un programa Fortran: sentencias de asignación, bucles, simple, doble y triple indexado, instrucciones aritméticas en coma fija y flotante y funciones implícitas tales como la exponenciación, raíz cuadrada, seno, etc.

Pasando este programa en monoprogramación se evalúa principalmente el recurso "CPU" junto con el compilador (y su nivel de optimización).

El segundo tipo de programa sintético puede estar constituido por sentencias que expresen los cálculos que usualmente aparecen en en programas científicos: multiplicar vectores, invertir matrices, evaluar polinomios, etc.

La deseada generalización consiste en un programa que sepa simular los valores del vector de recursos de los grupos. En

cada bucle del programa se activan rutinas que efectúan las correspondientes demandas al sistema (discos, cintas, CPU,- etc.). Al finalizar cada rutina se registra el tiempo usado en procesarla. En monoprogramación, este programa registraría los tiempos característicos de cada grupo trabajando sólo en el sistema. Este sería un método sencillo pero eficaz -154- de comparación. Se define el tiempo de servicio St_i como el número de segundos totales transcurridos ("elapsed time") - para conseguir simular 1 segundo de CPU para el grupo i.

El siguiente paso es pasar un conjunto de programas sintéticos, todos ellos representantes del grupo i se obtienen - - otros nuevos valores para S.

Introduciendo un número j de programas sintéticos, todos - ellos representantes del grupo i se obtienen otros nuevos - valores para S_{ti}. Los cambios producidos y su incidencia en la utilización de los recursos del sistema se analizan con la teoría de perturbaciones.

Este método con todo y sus limitaciones (dependencia del programador, desaprovechamiento de posibilidades de determinadas arquitecturas, etc.), proporciona indicaciones mucho - más precisas que los mix que no son más que sumas ponderadas según la frecuencia de aparición de los tiempos de ejecución de cada instrucción para obtener una estimación del tiempo medio de ejecución de una instrucción (average instruction execution time, AIET) o su inversa el número de instrucciones por segundo. Este método presenta defectos graves como:

no tener en cuenta la arquitectura de la CPU (pipe line, cache, etc.) que hace que el tiempo de ejecución de una instrucción no sea constante.

-155-

- no tener en cuenta el software que utiliza el sistema.

BIBLIOGRAFIA

(AGRA 76) AGRAWALA, A.K.; MOHR, J.M.: BRYANT, R.M.: An approach to the workload characterization problem. Computer pp 18-32. June 1976.

(AGUI 79) AGUILA, J.: Modelo para evaluar el impacto de variaciones en la configuración de discos y procesadores en el rendimiento de un sistema. Actes de la Convenció Informàtica Llatina CIL 79 pp 118-133.

(ARTIS 76) ARTIS, H.P.: A technique for determining the Capacity of a computer System. Proc. of the Computer Performance Evaluation User's Group Meeting. San Diego 1976.

(BAR 79 A) BARCELO, J., PUIGJANER, R.: Utilització de la simulació com eina per a predir el funcionament d'un sistema informàtic. Actes de la Convenció Informàtica Llatina CIL 79, pp 103-117.

(BAR 79 B) BARCELO, J., PUIGJANER, R.: Tchniques for computer modelling and wocklvad characterization. Proc. of ECOMA-7.

(BASK 75) BASKETT, F.: CHANDY, K.M.: MUNTZ, R.R.: PALACIOS, F.: Open, closed and mixed networks with differents classes of customers. Journal of the A.C.M. 22,2 Abril 1975, pp 248-260.

(BAUR 73) BAUER, M.J.: McCREDIE, J.W.: "AMS": A Software Monitor for Performance Evaluation and System Control. Proc. First Annual SICME Symposium on Measurement and Evaluation, February 1973 pp. 147-160.

(BEI 78) BEIZER, B.: Micro-Analysis of Computer System Performance. Van Nostrand Reinhold 1978.

(CHAN 78) CHANDY, K.M., SAUER, CH.: Approximate methods for Analyzing queueing network models of computer systems. Computing Surveys 10.3 Septiembre 1978, pp 281-317.

(COLO 78) COLOMER, J.L.; PUIGJANER, R.: Apunts dels cursos Tècniques d'anàlisi del rendiment dels ordinadors. Escola Informàtica d'Estiu. A.T.I. 1978.

(COLO 79A) COLOMER, J.L.; PUIGJANER, R.: Que es l'avaluació del rendiment d'un sistema informàtic. NOVATICA nº 25, Enero 1979, pp. 7-10.

(COLO 79B) COLOMER, J.L.: Caracterización de la carga. NOVATICA nº 25, Enero 1979, pp. 23-26.

(COU 77) COURTOIS, P.J. Decomposability: Queueing and Computer System Applications. Academic Press 1977.

(DENI 69) DENISON, W.R. "SIRE": A TSS/360: Software Measurement Tecnique Proc. 24 th ACM National Conference 1969 pp. 229-245.

(DENN 78) DENNING. P.; BUZEN, J.: The Operational analysis of queueing network models. Computing Surveys 10.3 Septiembre 1978, pp. 225-261.

(FERR 72) FERRARI, D.: Workload characterization and Selection in Computer Performance Measurement. Computer vol. 5 nº 4 Jul-Aug. 72. pp. 18-24.

(FERR 78) FERRARI, D.: Computer Systems performance evaluation. Prentice Hall 1978.

(GELE 75) GELENBE, E.: On approximate computer system models. Journal of the ACM 22.2 Abril 1975 pp. 261-269.

(GELE 76) GELENBE, E.: A non-markovian diffusion model and its application to the approximation of queueing system behaviour Rapport de Recherche nº. 158, Marzo 1976.

(HOLW 71) HOLTWICH, G.M.: Designing a Commercial Performance Measurement System. Proc. ACM SIGOPS. Work shop on System Performance Evaluation, April 1971 pp. 29-58.

(JACK 63) JACKSON, J.R.: Hobship like queueing systems. Management Science 10(1963) pp. 131-142.

(KLEI 75) KLEINROCK, L.: Queueing systems, Vol. I, John Wiley 1975.

(KLEI 76) KLEINROCK, L.: Queueing, Vol. II, John Wiley 1976.

(KOBA 74) KOBAYASHI, H.: Application of the diffusion approximation to queueing networks I: Equilibrium queue distributions Journal of the ACM 21, 2 Abril 1974 pp. 316-328; II: Nonequilibrium distributions and applications to computer models.

Journal of the ACM 21, 3 Julio 1974, pp. 549-469.

(KOB 78) KOBAYASHI, H.: Modeling and Analysis: An Introduction to System Performance Evaluation Methodology. Addison Wesley 1978.

(KOLE 71) KOLENCE, K.W.: A software view of measurement tools. Datamation 17, 1 January 1, 1971 pp. 32-38.

(KOLE 72) KOLENCE, K.W.: Software physics and computer performance measurement. Proc. ACM Nat Conf. pp. 1020-1040.

(LONE 70) LONERGAN, R.; ANDROSCIANI, V. "SUPERMON": A software Monitor for Performance Evaluation. Stanford Computation Center,Stanford, California, Technical Memo. No. 30 January 1970.

(MAM 77) MAMRAK, S.A.; AMER, P.D.: A feature selection tool for workload characterization. Proc. of the 1977 SIGMETRICS/ CMG VIII.

(MART 67) MARTIN, J.: Design of real-time computer systems Prentice Hall 1967.

(MOR 73) MORRISON, J.E.: User program performance in virtual storage systems. IBM Systems Jornal vol. 12, 1973.

(PART 76) PARTRIDGE, D.K.; CARD, R.E.: Hardware Monitoring of Real-Time Aerospace Computer Systems. Proc. International Symposium on Computer Performance Modeling, Measurement and Evaluation, March 1976 pp. 85-101.

(PUIG 77) PUIGJANER, R.: Metodologies per a l'avaluació del rendiment dels ordinadors. Actes de la Convenció Informàtica Llatina CIL 77, pp. 103-113.

(PUIG 79A) PUIGJANER, R.: Modelos de sistemas informáticos. NOVATICA nº 25, Enero 1979, pp. 15-22.

(PUIG 79B) PUIGJANER, R.; COLOMER, J.L.: Monitores. NOVATICA, nº 25, Enero 1979, pp. 27-32.

(PUIG 79C) PUIGJANER, R; COLOMER, J.L.; BARCELO, J.: Determinació de la càrrega d'un centre de càlcul per mètodes d'agrupament. Actes de la Convenció Informàtica Llatina CIL 79, pp. 96-102.

(REI 74) REISER, M.; KOBAYASHI, H.: Accuracy of the Diffusion Approximation for Some Queueing Systems. IBM Journal of Research and Development, vol. 18 nº 2 pp. 110-124.

(ROBE 72) ROBERTS, L.W.: Performance Measurement with Microcode. Presented at the Seminar on Computer System Performance Measurements, Whippany, New Jersey, June 14-155, 1972.

(ROSE 78) ROSE, C.A.: Measurement procedure of queueing network models of computer systems. Computing Surveys 10, 3 Septiembre 1978, pp. 263-280.

(SAAL 72) SAAL, H.J.; SHUSTEK, L.J.: Microprogrammed Implementation of Computer Measurement Techniques. Proc. ACM 5 th Annual Wokshop on Microprogramming.

(SARZ 77) SARZOTTI, A.: Techniques d'evaluation et de mesure des Systèmes Informatiques, Eyrolles, 1977.

(SEBA 74) SEBASTIAN, P.R.: Hybrid Events Monitoring Instrument. Proc. Second Annual SIGMETRICS Symposium on Measurement and Evaluation, September 1974, pp. 127-139.

(SVOB 73) SVOBODOBA, L.: Online System Performance Measurement with software and Hybrid Monitors. Proc. ACM SIGOPS 4 th Symposium on Operating Systems Principles, October 1973, pp. 45-53.

(SVOB 76) SVOBODOBA, L.: Computer Performance Measurement and Evaluation Methods: Analysis and Application, American Elsevier Publishing Company, 1976.

(TRIV 78) TRIVEDI, K.S.: Analytic Modeling of Computer Systems. Computer 11, 10 Octubre 1.978.

(WALD 73) WALDBAUM, G.: Evaluating Computing System. Changes by Means of Regression Models. Proc. First Annual SICME Symposium on Measurement and Evaluation, February 1973, pp. 127-135.

DIRECTORIO DE ASISTENTES AL CURSO: MODELADO Y EVALUACION DEL RENDIMIENTO DE COMPUTADORAS, DEL 5 AL 15 DE NOVIEMBRE DE 1979.

NOMBRE Y DIRECCION	EMPRESA Y DIRECCION
1. JOSE DANIEL AYALA URANGA W. 157 Col. Moderna México 13, D. F. Tel: 5-57-71-33	COMISION FEDERAL DE ELECTRICIDAD Ródano No. 14 Col. Cuauhtémoc México 5, D. F. Tel: 5-79-74-82
2. SERGIO G. BANUET MORALES Batalla de Celaya Edif. J-A Col. Habitencial Militar	CENTRO DE SERVICIOS DE COMPUTO Ciudad Universitaria México 20, D. F. Tel: 5-50-52-15
3. JOSE ANSELMO BECERRA APONTE Andres Moline Enriquez No. 1006-C-001 San Andres Tetepilco México 13, D. F.	CIATES Presidente Carranza No. 162 Coyoacán México 21, D. F. Tel: 5-54-84-62
4. GUILLERMO CAHUE DIAZ Juan Escutia No. 177 Col. Américas Unidas México 13, D. F. Tel: 6-72-05-73	INSTITUTO DE INVESTIGACIONES ELECTRICAS Internado Palmira Col. Palmira Cuernavaca, Mor. Tel: 4-13-93
5. ERASMO CAL Y MAYOR CRUZ Museo del Vaticano No. 6 Bellavista, Sat. Edo. de México	LECHE INDUSTRIALIZADA CONASUPO,S.A. Km. 17.5 Carr. Mex. Tlalnepantla
6. ROLANDO S. CARRERA SANCHEZ	FACULTAD DE INGENIERIA, UNAM Ciudad Universitaria México, D. F.
7. JOSE RICARDO CIRIA MERCE Loma Hermosa 42-304 Col. Irrigación México 10, D. F. Tel: 5-57-41-60	FACULTAD DE INGENIERIA CENTRO DE CALCULO Ciudad Universitaria México 20, D. F. Tel: 5-50-52-15 Ext. 4150

DIRECTORIO DE ASISTENTES AL CURSO: MODELADO Y EVALUACION DEL DESEMPEÑO DE COMPUTADORAS, DEL 5 AL 15 DE NOVIEMBRE DE 1979.

	NOMBRE Y DIRECCION	EMPRESA Y DIRECCION
8.	JOSE JUAN CONTRERAS ESPINOSA Dios Flor No. 6 Sec. Los Parques Cuautitlan Izcalli	ENEP-CUAUTITLAN Cuautitlan Izcalli Edo. de México
9.	LUIS CORDERO BORBOA Vallarta No. 37 Coyoacán México 21, D. F.	FACULTAD DE INGENIERIA, UNAM Ciudad Universitaria México 20, D. F. Tel: 5-50-59-15 Ext. 3750
10.	JUVENTINO CUATE VALERDI Andorra No. 41-3 Col. Zacahuiztco México, D. F. Tel: 5-39-48-87	LIBRA ASESORES, S. A. Av. Boulevard Adolfo L. M. No. 2777 Progreso Atizapan México, D. F. Tel: 5-48-51-14
11.	ENRIQUE DUARTE N. Plaza del Carmen No. 26 La Alteña I, L. Verdes Naucalpan, México Tel: 5-62-27-50	INSTITUTO DE INVESTIGACIONES ELECTRICAS Leibnitz No. 14-7o. Piso Col. Anzurez México 5, D. F. Tel: 5-33-69-54
12.	MARGARITA ESPONDA ARGUERO Ferrocarril de Cuernavaca No. 279 Col. Anáhuac México, D. F. Tel: 5-31-58-49	INSTITUTO NACIONAL DE INVESTIGACIONES NUCLEARES Km. 36 1/2 Carretera México Toluca Tel: 5-70-14-72
13.	JORGE FERNANDEZ MORENO Heriberto Frias 1421-2 Col. del Valle México 12, D. F. Tel: 5-03-71-33	COMISION FEDERAL DE ELECTRICIDAD Ródano No. 14 Col. Cuauhtémoc México 5, D. F. Tel: 5-59-66-71
14.	ANA MARIA FLORES VELEZ Meseta No. 130 Pedregal México 20, D. F. Tel: 5-68-10-67	

DIRECTORIO DE ASISTENTES AL CURSO: MODELADO Y EVALUACION DEL RENDIMIENTO DE COMPUTADORAS, DEL 5 AL 15 DE NOVIEMBRE DE 1979.

NOMBRE Y DIRECCION	EMPRESA Y DIRECCION
15. ALBERTO GARCIA ADALID Vi[illegible]ñor No. 5 C. Geógrafos Cd. [illegible]te Tel: 5-[illegible]2-16-37	INSTITUTO DE INVESTIGACIONES ELECTRICAS Leibnitz No. 14-7o. Piso,703 Col. Condesa México 11, D. F. Tel: 5-33-69-5[illegible]
16. JORGE GARCIA CAMACHO Doctor Balmis 24 Int. 14 Col. Doctores México 7, D. F.	FACULTAD DE INGENIERIA CENTRO DE CALCULO Ciudad Universitaria México 20, D. F. Tel: 5-50-52-15 Ext. 4150
17. ISABEL A. GOMEZ JIMENEZ Brillante 26 Col. Estrella México 14, D. F. Tel: 5-37-47-59	LIBRA ASESORES, S. A. Av. Boulevard Adolfo L. M. No. 2777 Progreso Atizapan México, D. F. Tel: 5-48-51-14
18. JORGE HERNANDEZ AGUILAR Heriberto Frías 1505-202 Col. del Valle México 12, D. F. Tel: 5-59-07-41	COMISION FEDERAL DE ELECTRICIDAD Av. Toluca y Don Manuelito Col. Olivar de los Padres México 20, D. F. Tel: 5-95-54-00
19. CIPRIANA JIMENEZ C. Condominios Jardín Entrada C-204 Naucalpan, Edo. de México Tel: 5-76-59-55	P. RIWTAFORM Altamirano 90 Col. San Rafael México, D. F. Tel: 5-66-05-72
20. JOSE LANDEROS VALDEPEÑA Calle dos No. 41 Col. Independencia México 13, D. F.	ENEP-CUAUTITLAN IZCALLI Cuautitlan Izcallí México, D. F.
21. RAMON LIRA COLORADO Costarricences No. 16 Ma. G. de Ruiz México 18, D. F. Tel: 2-71-01-78	ENEP-CUAUTITLAN Cuautitlan Izcallí

DIRECTORIO DE ASISTENTES AL CURSO: MODELADO Y EVALUACION DEL RENDIMIENTO DE COMPUTADORAS, DEL 5 AL 15 DE NOVIEMBRE DE 1979.

NOMBRE Y DIRECCION	EMPRESA Y DIRECCION
22. VICTOR F. LOPEZ DE BUEN Cda. Río Churubusco No. 12-7 Col. Portales México 13, D. F. Tel: 5-39-04-60	FACULTAD DE INGENIERIA, UNAM Fray Servando y Teresa de Mier No. 77-8o. Piso Col. Obrera México 8, D. F. Tel: 7-61-40-44 Ext. 262
23. ROBERTO MALDONADO MAZA Zumpango No. 39 Col. Mirador Naucalpan, Edo. de México Tel: 5-60-86-51	CENTRO DE SERVICIOS DE COMPUTO Ciudad Universitaria México 20, D. F. Tel: 5-52-12-10 Ext. 4543
24. JOSE MARIA MENDOZA GUTIERREZ Cerro del Sombrero No. 139-3 Col. Campestre Churubusco México 21, D. F. Tel: 5-44-23-61	FACULTAD DE INGENIERIA, UNAM Ciudad Universitaria México 20, D. F. Tel: 5-50-52-15 Ext. 3750
25. ISAAC O. MEZA BAUTISTA Fisicos 312 Col. Sifón México 8, D. F. Tel: 5-82-37-87	MEXICANA DE AVIACION Aeropuerto Internacional Benito Juárez México, D. F. Tel: 7-62-89-02
26. JAIME DANIEL MORENO JIMENEZ Camino Real de Toluca No. 78 Col. Bella Vista México 18, D. F. Tel: 2-71-16-79	ENEP-CUAUTITLAN Cautitlan Izcalli
27. HERIBERTO OLGUIN ROMO Odontologia 69-401 Fracc. Copilco Universidad México 21, D. F. Tel: 5-48-18-60	FACULTAD DE INGENIERIA, CENTRO DE CALCULO Ciudad Universitaria México 20, D. F. Tel: 5-50-52-15 Ext. 4150
28. JORGE ONTIVEROS JUNCO Av. Universidad 2014 Brasil 603 Copilco San Angel México 20, D. F. Tel: 6-58-02-49	FACULTAD DE INGENIERIA, UNAM Ciudad Universitaria México 20, D. F. Tel: 5-50-52-15 Ext. 4150

DIRECTORIO DE ASISTENTES AL CURSO: MODELADO Y EVALUACION DEL RENDIMIENTO DE COMPUTADORAS, DEL 5 AL 15 DE NOVIEMBRE DE 1979

NOMBRE Y DIRECCION	EMPRESA Y DIRECCION
29. JORGE PAULIN URIOLA Darwin No. 102-3 Col. Anzurez México 5, D. F. Tel: 5-33-18-00	INGENIEROS Y CONTRATISTAS, S. A. Darwin No. 102-3 Col. Anzurez México 5, D. F. Tel: 5-33-18-00
30. RAFAEL PICENO REINA Km. 17.5 Carr. México-Tlalnepantla Edo. de México	LECHE INDUSTRIALIZADA CONASUPO, S. A. Km. 17.5 Carr. México-Tlalnepantla Edo. de México
31. GUADALUPE QUIJANO LEON Veracruz No. 46- L-16 San Jerónimo México 20, D. F.	CENTRO DE SERVICIOS DE COMPUTO Ciudad Universitaria México 20, D. F. Tel: 5-50-52-15 Ext. 4543
32. CARLOS A. RAMOS LARIOS Moras 762 Col. del Valle México 12, D. F. Tel: 5-34-91-10	FACULTAD DE INGENIERIA CENTRO DE CALCULO Ciudad Universitaria México 20, D. F. Tel: 5-50-52-15 Ext. 4150
33. ISRAEL RANGEL MEZA Dr. Vertiz 489 Int. 39 Col. Narvarte México 12, D. F.	SECRETARIA DE PROGRAMACION Y PRESUPUESTO Izazaga No. 38 México 1, D. F. Tel: 5-21-75-44
34. JUAN PABLO REYES GARCIA 3a. Cda. de San Antonio Tomatlán No. 14 Col. 7 de Julio México 9, D. F. Tel: 7-89-11-82	DIESEL NACIONAL, S. A. Av. Universidad 803 Col. del Valle México 12, D. F. Tel: 5-75-94-18
35. DANIEL RIOS ZERTUCHE O. Puebla 207 San Angel México 20, D. F. Tel: 5-48-42-69	SECRETARIA DE HACIENDA Y CREDITO PUBLICO Fray Servando No. 198 México 1, D. F. Tel: 5-42-29-26