ECO - Apuntes y ejercicios resueltos

Por
Pascal

UNIVERSIDAD COMPLUTENSE
MADRID

Grado en Ingeniería Informática
Facultad de Informática

ECO - Apuntes y ejercicios resueltos

Madrid, 2018–2019

Sobre TEFLON

Teflon(cc0 1.0(documentación) MIT(código))es una plantilla de LaTeX creada por David Pacios Izquierdo con fecha de Enero de 2018. Con atribuciones de uso CC0.

Esta plantilla fue desarrollada para facilitar la creación de documentación profesional para Trabajos de Fin de Grado o Trabajos de Fin de Máster. La versión usada es la 1.3.

V:1.3 Overleaf V2 with pdfLaTeX, margin 1in, NO-bib

Contacto
Autor: David Pacios Izquiero
Correo: dpacios@ucm.es
ASCII: asciifdi@gmail.com
Despacho 110 - Facultad de Informática

Índice general

Tema 1

Rendimiento

1.1. Teoría previa

A la hora de evaluar un sistema podemos utilizar las siguientes métricas:

- **Rendimiento**: es la cantidad de trabajo realizado por un sistema informático. Puede medirse en función de los siguientes factores.
  - Tiempo de respuesta.
  - Productividad.
  - Utilización.
- **Fiabilidad**: estabilidad y consistencia.
  - Probabilidad de fallo.
  - Tasa de fallos o tiempo medio entre fallos.
  - Tiempo medio de reparación.
  - Disponibilidad.
- **Coste**
  - Dinero.
  - Energía.

1.1.1. Mejora de las prestaciones

Un sistema tarda un tiempo $T_{original}$ en ejecutar un programa. Mejoramos el sistema acelerando k veces uno de sus componentes, que se utiliza durante una fracción f del tiempo $T_{original}$

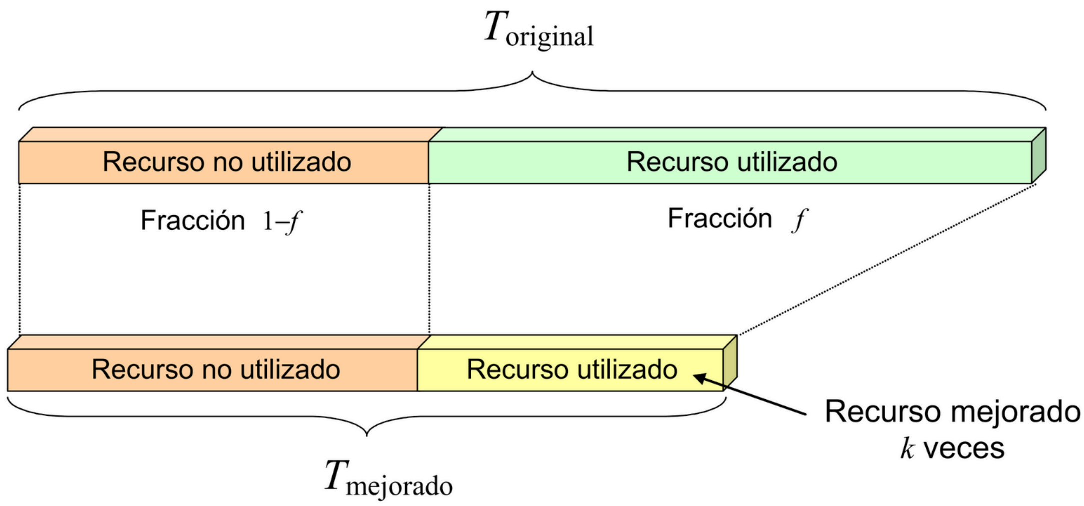


Figura 1.1: Estación de servicio

Ley de Amdahl

La ley de Amdahl permite calcular la aceleración A (speedup) del sistema completo después de acelerar k veces un componente que se usa una fracción f del tiempo.

$$A = \frac{1}{(1-f) + (\frac{f}{k})}$$

1.2. Ejercicios resueltos

1.2.1. Ejercicio 1

Dos computadores A y B ejecutan un programa P en 35 y 87 segundos respectivamente. El coste del computador A es de 710 euros y el del computador B de 650. ¿Cuál de ellos representa una mejor relación entre prestaciones y coste?

Tiempo:
$A \to 35s$
$B \to 87s$
Coste:
$A \to 710$€
$B \to 650$€
Primero comparamos el tiempo:

$$k = \frac{\max}{\min} = \frac{87}{35} = 2{,}49 = 1 + \frac{149}{100}$$

A es un 149 % más rápido que B. Ahora comparamos el coste:

$$\frac{\max}{\min} = \frac{710}{650} = 1{,}09 = 1 + \frac{9}{100}$$

A es un 9 % más caro.

$$\frac{\text{rendimiento}}{\text{coste}} \begin{cases} A : \dfrac{1/35}{710} = 4{,}024 \cdot 10^{-5} \\ B : \dfrac{1/87}{650} = 1{,}768 \cdot 10^{-5} \end{cases}$$

Por tanto A representa una mejor relación rendimiento coste.

1.2.2. Ejercicio 2

Un procesador se utiliza el 83 % del tiempo en funcionamiento de un sistema informático. Si se sustituye por uno nuevo 2.5 veces más rápido ¿Cuál será el rendimiento conseguido? **Nota**: Dar dicho rendimiento como aceleración global del SI.

f=0.83
k=2.5

$$A = \frac{1}{1 - f + \left(\dfrac{f}{k}\right)} = \frac{1}{0{,}17 + \dfrac{0{,}83}{2{,}5}} = 1{,}99 = 1 + \frac{99}{100}$$

Se consigue una mejora del 99 %.

1.2.3. Ejercicio 3

Se desea mejorar el rendimiento de un computador mediante la compra de una unidad de coma flotante, lo que permitirá reducir a la mitad el cálculo de operaciones aritméticas.

Habitualmente se ejecuta una aplicación que dedica el 60 % del tiempo a la realización de cálculo aritmético. Sin unidad de coma flotante se tarda en ejecutar el programa principal 12 segundos. ¿Cuánto tiempo tardará con la nueva unidad de coma flotante?

f=0.6
k=2
T_0 =12s

$$A = \frac{1}{1 - f + \left(\frac{f}{k}\right)} = \frac{1}{0{,}4 + \left(\frac{0{,}6}{2}\right)} = 1{,}43 = 1 + \frac{43}{100}$$

Mejora del 43 %.

$$A = \frac{T_{\text{original}}}{T_{\text{mejorado}}} \rightarrow T_m = \frac{T_0}{A} = \frac{12}{1{,}43} = 8{,}4s$$

1.2.4. Ejercicio 4

Con el objetivo de mejorar el rendimiento de un computador se dispone de dos opciones diferentes:

a). Ampliar la memoria principal, con un coste de 250 euros consiguiendo que el 50 % de los programas (todos ellos semejantes en tiempo de ejecución) se ejecuten tres veces más rápido.

Opción 1:
250 €
f=0.5
k=3

$$A_1 = \frac{1}{0{,}5 + \frac{0{,}5}{3}} = 1{,}5 \rightarrow 50\,\% \quad \text{de mejora}$$

b). Cambiar la placa base, con un coste de 150 euros, y con lo que el 70 % de los programas (todos ellos semejantes en tiempo de ejecución) se ejecutarán en la mitad de tiempo.

Opción 2:
150 €
f=0.7
k=2

$$A_2 = \frac{1}{0{,}3 + \frac{0{,}7}{2}} = 1{,}54 \rightarrow 54\,\% \quad \text{de mejora}$$

¿Cuál es la mejor opción en cuanto a prestaciones y coste?

La opción 2 es mejor: tiene un mayor rendimiento y un menor coste (es más barata).

1.2.5. Ejercicio 5

Un programa tarda en ejecutarse un total de 124 segundos. Durante este tiempo el procesador está ejecutando tres tipos diferentes de instrucciones: aritméticas de enteros, salto y coma flotante. La proporción del tiempo de ejecución en que se emplea cada tipo es del 28, 40 y 32 % respectivamente. Se pide:

a). Calcular el incremento de prestaciones si se mejoran un 15 y un 45 % las instrucciones de aritmética entera y de salto respectivamente.

$f_1 = 0{,}28$ $\quad f_2 = 0{,}4$
$k_1 = 1{,}15$ $\quad k_2 = 1{,}45$

Usamos la formula de Amdahl generalizada:

$$A = \frac{1}{(1 - \sum f) + \sum \left(\frac{f}{k}\right)} = \frac{1}{0{,}2 + \frac{0{,}28}{1{,}15} + \frac{0{,}4}{1{,}45}} = 1{,}19 \rightarrow \text{mejora} \quad 19\,\%$$

b). Determinar cuánto se tienen que mejorar las operaciones de coma flotante si queremos rebajar el tiempo de ejecución original hasta los 95 segundos solo con esta mejora.

T_m =95s
$f_3 = 0{,}32$

$$A = \frac{124s}{95s} = 1{,}31s$$

$$A = \frac{1}{1 - f_3 + \left(\frac{f_3}{k_3}\right)} \rightarrow k_3 = 3{,}72 \rightarrow \text{mejora} \quad 272\,\%$$

1.2.6. Ejercicio 6

Un computador sin memoria caché ejecuta un programa en 180 segundos. Si se incorpora una memoria de este tipo que funcionan 15 veces más rápidamente que la memoria principal, calcúlese el porcentaje de accesos a memoria que deben satisfacerse por la memoria caché para que el programa consiga ejecutarse en 80 segundos.

k=15
T_m = 80s
Nos están pidiendo calcular la f de la ley de Amdahl, así que despejando nos queda:

$$f = \frac{k(A-1)}{A(k-1)} = 0{,}60$$

1.2.7. Ejercicio 7

Una aplicación tarda en ejecutarse en una máquina secuencial 120 segundos. El administrador del sistema ha comprobado que el 70 % de las tareas de la aplicación podrían ser paralelizadas para su ejecución en un sistema multiprocesador en la nube. Se pide:

a). ¿Qué aceleración experimentará la ejecución de la aplicación si el administrador contrata un sistema de 16 procesadores idénticos para la parte paralelizable? ¿En cuánto tiempo se ejecutará la aplicación? ¿Cuál será la aceleración si contrata 20, 50, 100 procesadores?

f=0.7
k=16
t_A =120s

$$A = \frac{1}{1 - f + \frac{f}{k}} = \frac{1}{(1_k - 0{,}7) + \frac{0{,}7}{16}} = 2{,}91$$

Calculamos de la misma manera los resultados para las otras 3 situaciones, y nos queda:

$$\begin{cases} k = 20 \rightarrow A = 2{,}98 \\ k = 50 \rightarrow A = 2{,}4 \\ k = 100 \rightarrow A = 3{,}25 \end{cases}$$

b). ¿A qué tiempo de ejecución podría reducirse el cómputo sin reprogramar la aplicación?

Calculamos el límite de A suponiendo infinitos procesadores:

$$\lim_{k \to \infty} A = 3{,}3$$

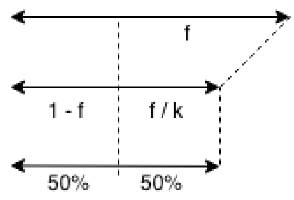
Y sacamos el tiempo mejor:

$$t_{\text{mej}} = \frac{t_0}{A} = \frac{120}{3{,}3} = 36s$$

1.2.8. Ejercicio 8

Durante la fase de diseño de un computador se ha introducido una mejora que incrementa en un factor k=10 el rendimiento local de un cierto recurso. Esta mejora ha permitido reducir el tiempo de ejecución original de un programa. Una vez introducida la mejora se ha podido estimar que ésta se emplea durante el 50 % del tiempo mejorado.

a). ¿Cuál es el porcentaje del tiempo original de ejecución del programa afectado por la mejora?



Podemos igualar las dos mitades y sacamos el valor de f:

$$1 - f = \frac{f}{k} \rightarrow 10 - 10f - f = 0 \rightarrow f = 0{,}9090$$

b). ¿Cuál es la aceleración global conseguida?

$$A = \frac{1}{1 - 0{,}909 + \left(\frac{0{,}909}{10}\right)} = 5{,}5$$

1.2.9. Ejercicio 9

Consideremos un computador dedicado exclusivamente a la gestión de reserva de billetes en un aeropuerto regional donde el procesador actual, que funciona a 66 MHz, es el cuello de botella, con una utilización media del 84 % (el 16 % restante se usa en tareas de entrada/salida). Se pide determinar cuál de las dos opciones siguientes para reemplazar el procesador presenta la mejor relación entre prestaciones y coste.

En ambos casos se supone que el computador admite los dos procesadores y que no hace falta recompilar la aplicación de reserva de billetes.

a). Procesador Bart a 550 MHz, con un precio de 480 euros.

Opción a:

$$k_1 = \frac{\max}{\min} = \frac{550}{66} = 8{,}3$$

$$C_1 = 480\text{euros}$$

$$A_1 = \frac{1}{0{,}16 + \frac{0{,}84}{8{,}3}} = 3{,}83$$

$$\frac{A_1}{C_1} = 0{,}0079$$

b). Procesador Homer a 500 MHz, con un precio de 270 euros.

Opción b:

$$k_2 = \frac{500}{60} = 7{,}57$$

$$C_2 = 270\text{euros}$$

$$A_2 = \frac{1}{0{,}16 + \dfrac{0{,}84}{7{,}57}} = 3{,}69$$

$$\frac{A_2}{C_2} = 0{,}013$$

Puesto que $\frac{A_2}{C_2} > \frac{A_1}{C_1}$, el procesador Homer es mejor.

1.2.10. Ejercicio 10

Un computador ejecuta una aplicación de gestión de base de datos. Cada transacción con esta base de datos tarda un total de 12 segundos. Un análisis llevado a cabo mediante un monitor de ejecución de programas ha permitido averiguar que el 78 % del tiempo que tarda cada transacción se debe al acceso del subsistema de discos, mientras que el resto del tiempo se emplea en operaciones del procesador.

a). Calcúlese el nuevo tiempo de respuesta de una transacción si el subsistema de discos se sustituye por uno nuevo cuatro veces más rápido.

$$A = \frac{1}{1 - f + \left(\dfrac{f}{4}\right)} = 2{,}414$$

$$A = \frac{T_A}{T_D} \rightarrow T_D = \frac{T_A}{A} = 4{,}98s$$

b). Repítase el apartado anterior suponiendo que el subsistema de discos se sustituye por uno nuevo seis veces más rápido.

$$A = \frac{1}{1 - f + \left(\dfrac{f}{6}\right)} = 2{,}85$$

$$T_D = \frac{T_A}{A} = 4{,}21$$

c). Determínese la mejora del rendimiento obtenida en cada una de las actualizaciones anteriores.

$$\frac{4{,}98}{4{,}21} = 1{,}18$$

1.2.11. Ejercicio 11

Un programa de simulación se ejecuta en 280 segundos. El 70 % del tiempo utiliza el procesador y el resto utiliza el disco.

a). Determina el tiempo de ejecución con un procesador tres veces más rápido.

k=3
A=1.875

$$T_D = \frac{T_A}{A} = 149{,}33$$

b). Calcula la fracción del tiempo mejorado de ejecución durante el cual se utiliza el nuevo procesador. Haz un análisis del fenómeno observado.

$$280 \cdot 0{,}3 = 84$$

$$T_{\text{mej}} = 149{,}33 - 84 = 65{,}33$$

$$x = \frac{65{,}33 \cdot 100}{144{,}31} = 43{,}75\,\%$$

c). A raíz del resultado obtenido en el apartado anterior, si hubiéramos de mejorar este sistema actualizado, ¿sobre qué componente del mismo deberíamos incidir?

Deberíamos cambiar el dispositivo que más frecuencia de uso tenga.

1.2.12. Ejercicio 12

Sean dos factores de mejora sobre un sistema k_1=5 y k_2=3. Suponiendo que ambas mejoras no se pueden hacer de manera simultánea, calcula la relación entre las fracciones de tiempo f_1 y f_2 durante las cuales se han de emplear las mejoras para que ambas obtengan la misma aceleración global.

$$\frac{1}{1 - f_1 + \frac{f_1}{5}} = \frac{1}{1 - f_2 + \frac{f_2}{3}}$$

$$1 - f_1 + \frac{f_1}{5} = 1 - f_2 + \frac{f_2}{3}$$

$$f_2 = \frac{12}{10} f_1$$

1.2.13. Ejercicio 13

El tiempo medio de respuesta de un sitio web es de 15 segundos. El 55 % de este tiempo es utilizado por el subsistema de discos, mientras que el resto es utilizado por el procesador. Se pretende reducir este tiempo por debajo de los 10 segundos. Indica cuál de las siguientes opciones es mejor:

a). Adquirir un nuevo procesador un 50 % más rápido.

k=2.5
A=1.17

$$T_D = \frac{T_A}{A} = 12{,}75s$$

b). Sustituir el subsistema de discos por uno 2,5 veces más rápido que el actual.

$T_{\text{disco}} = 8{,}28$
$T_{\text{proc}} = 6{,}75$
k=2.5
A=1.49

$$T_D = \frac{T_P}{A} = 10{,}06s$$

Ninguno de los dos mejora.

1.2.14. Ejercicio 14

Un sistema multiprocesador está configurado con un número fijo de 6 procesadores y la aceleración conseguida con un programa es igual a 3.

a). ¿Cuál es la fracción paralelizable f del programa?

Despejamos f de la formula de Amdahl:

$$3 = \frac{1}{1 - f + \left(\frac{f}{6}\right)} \rightarrow f = 0{,}8 \rightarrow 80\,\%$$

b). Si la versión secuencial del programa se ejecuta en 325 segundos, ¿en cuánto tiempo lo hará con la actual configuración del multiprocesador?

$$T_A = 325 \rightarrow T_0 = \frac{T_A}{A} = \frac{325}{3} = 108{,}3s$$

c). ¿Podría reducirse el tiempo de ejecución por debajo de 55 segundos ampliando el número de procesadores hasta 32?

$$t_{\text{mej}} = T_A \cdot \left(0{,}2 + \frac{0{,}8}{32}\right) = 73{,}13s$$

No se consigue.

d). ¿Y con 16 procesadores y reduciendo a la mitad la fracción no paralelizable?

La fracción no paralelizable es $1 - f = 0{,}2$, por tanto si la reducimos a la mitad su valor será 0,1, y por tanto $f = 0{,}9$.

$$t_{\text{mej}} = 325 \cdot \left(0{,}1 + \frac{0{,}9}{16}\right) = 50{,}78s$$

Sí, se consigue.

Tema 2

Monitores

2.1. Teoría previa

Para comprobar el correcto funcionamiento de un sistema, nos interesa medir las variables siguientes:

- Tiempo que tarda en producirse un evento.
- Número de eventos que ocurren en una unidad de tiempo T

2.1.1. Atributos de las herramientas de medida

- Interferencia o **sobrecarga** (overhead)

$$\text{Sobrecarga} = \frac{\text{T. ejecución monitor}}{\text{Intervalo de medida}}$$

- Coste

2.1.2. Promedios de carga l_1, l_2, l_3

Los promedios de carga l_1, l_2, l_3 corresponden a los valores de carga de 1, 5, y 15 minutos antes.

Tupla más inestable $(\alpha, 0, 0, .., 0, \beta, 0, 0, 0, \gamma)$
Más estable $(\alpha, \alpha, \alpha, ...\alpha, \beta, \beta, \beta, \beta, \gamma)$
$\gamma = L_1$

$$\frac{\beta + 0 + 0 + 0 + \gamma}{5} = L_2$$
$$\frac{\alpha + 0 + ... + 0 + \beta + 0 + 0 + 0 + \gamma}{15} = L_3$$

2.2. Ejercicios resueltos

2.2.1. Ejercicio 1

En un sistema informático la ejecución de un monitor está constituida por 150 instrucciones máquina. El procesador tiene una velocidad de 75 MIPS. Se pide:

a). Calcular el periodo de muestreo para no superar una sobrecarga del 5 %.

$$\text{overhead} = \frac{\text{tiempo ejecución monitor}}{\text{intervalo de medida}}$$

Calculamos el tiempo de ejecución que es:

$$\frac{150 \text{ ins}}{75 \cdot 10^6 \text{ ins/s}} = 2 \cdot 10^6 s$$

Y ahora sacamos el periodo de muestreo (o intervalo de medida) máximo:

$$0{,}05 \geq \frac{2 \cdot 10^{-6}}{T}; \quad \frac{2 \cdot 10^6}{0{,}05} \geq T;$$

$$40 \cdot 10^6 \geq T$$

b). Con el periodo mínimo anterior, calcular el tamaño del fichero de resultados generado tras un periodo de 4 horas si en cada activación el monitor genera una palabra de 16 bits.

pal= 16 bits
4 horas
$t = 40 \cdot 10^{-6}$

$$4 \text{ horas} = 4 \cdot 60 \cdot 60 = 14400s$$

$$\frac{14400}{40 \cdot 10^{-6}} = 360 \cdot 10^6 \quad \text{veces que se activa el monitor en 4 horas}$$

$$360 \cdot 10^6 \cdot 16 = 686{,}65\text{MB}$$

2.2.2. Ejercicio 2

El resultado de la ejecución de vmstat 3 16 en un sistema informático con sistema operativo Linux, produce una columna r 8 1 5 2 2 2 2 4 2 2 3 6 8 3 3 3 y además se conoce que la suma de la columna u_s es 1227, la suma de la columna s_y es 155, la suma de la columna b_i es 85, la suma de la columna b_0 es 19. También se conocen los datos iniciales siguientes: u_s=7, s_y=5, b_i=10, b_0=4.
Se pide:

a). ¿Cuál es el periodo de medida?

15 = 16-1, que son los intervalos intermedios.

$$3 \cdot 15 = 45$$

b). ¿Cuál es el número medio de procesos en espera para ser ejecutados?

Observemos que para calcular la media desechamos siempre el primer valor.

$$\vec{r} = \frac{1}{15}\sum_{2}^{16} r = \frac{48}{16} = 3{,}2$$

Hay 3.2 procesos.

c). ¿Cuál es la utilización media del procesador en modo usuario?

Observemos que para calcular la media desechamos siempre el primer valor.

$$\vec{u_s} = \frac{1}{15} \cdot (1227 - 7) = \frac{1220}{15} = 81{,}33\,\%$$

d). Calcular la sobrecarga media del procesador producida por el sistema operativo.

Observemos que para calcular la media desechamos siempre el primer valor.

$$\vec{s_y} = \frac{1}{15} \cdot (155 - 5) = \frac{150}{15} = 10\,\%$$

e). ¿Cuál ha sido en media la cantidad de bloques transferidos con los dispositivos de bloques?

Observemos que para calcular la media desechamos siempre el primer valor.

$$\vec{b_i} = \frac{1}{15}\sum_{1}^{16} bi = \frac{1}{15}(85 - 10) = \frac{75}{15} = 5$$

$$\vec{b_0} = \frac{1}{15}\sum_{2}^{16} bo = \frac{15}{15} = 1$$

2.2.3. Ejercicio 3

Tras ejecutar la orden *top* de Linux en un computador, se obtienen los siguientes datos:

1:27pm up 1day 1:11, 3 users, load average: 2.46, 0.80, 0.28
CPU states: 82.5 % user, 0.5 % system, 17.0 % nice
Mem: 256464 k av, 251672 k used, 4792 k free

Se pide:

a). ¿Cuánta memoria física tiene el computador?

256464 k

b). Porcentaje de memoria física en uso actual.

Used/total=98.131 %

c). Uso medio del procesador.

como idle=0 % $\rightarrow$ user+system+nice=100 %.

d). ¿Cuál es el valor más inestable posible para la carga media obtenida?

Recordemos que:
Tupla más inestable $(\alpha, 0, 0, .., 0, \beta, 0, 0, 0, \gamma)$
Más estable $(\alpha, \alpha, \alpha, ...\alpha, \beta, \beta, \beta, \beta, \gamma)$
Y que:
$\gamma = L_1$

$$\frac{\beta + 0 + 0 + 0 + \gamma}{5} = L_2$$

$$\frac{\alpha + 0 + ... + 0 + \beta + 0 + 0 + 0 + \gamma}{15} = L_3$$

Entonces:

$$\left.\begin{array}{l} L_1 = 2{,}46 \\ L_2 = 0{,}8 \\ L_3 = 0{,}28 \end{array}\right\} \left.\begin{array}{l} \gamma = 2{,}46 \\ \beta = 1{,}54 \\ \alpha = 0{,}2 \end{array}\right\} (0.2, 0, 0..,0, 1.54, 0, 0, 0, 2.46)$$

2.2.4. Ejercicio 4

Un sistema informático tiene instalado SAR y se activa cada 20 minutos tardando 450 ms en ejecutarse. En cada activación construye un registro de datos con la información obtenida y lo añade al histórico el día DD correspondiente. Se pide:

a). Calcular la sobrecarga que produce la monitorización.

$$\text{sobrecarga} = \frac{\text{tiempo}}{\text{periodo}} = \frac{0{,}45}{20 \cdot 60} = 0{,}000375$$

b). Calcular el tamaño del directorio var/log/sa a lo largo de dos semanas si el registro generado en cada activación ocupa 3 KB.

Hallamos el número de veces que se activa el monitor en dos semanas, y lo multiplicamos por el tamaño de cada activación (3K):

$$3K \cdot \frac{24 \cdot 60 \cdot 14 \text{ min}}{20 \text{ min}} = 3024K$$

c). Si el volumen máximo del directorio var/log/sa es de 150 MB, ¿Cuántos ficheros históricos saDD se pueden almacenar?

Los ficheros saDD almacenan la información de un solo día, por tanto

$$\frac{150 \cdot 1024 \text{ K}}{\frac{24 \cdot 60 \text{ min}}{20 \text{ min}} \cdot 3 \text{ K}} = 711 \text{ ficheros}$$

2.2.5. Ejercicio 5

Un monitor software se activa cada 10 minutos y tarda 300 ms en ejecutarse en cada activación. Se pide:

a). Estimar la sobrecarga producida al sistema informático.

$$\text{sobrecarga} = \frac{\text{tiempo}}{\text{periodo}} = \frac{0{,}3}{600} = 5 \cdot 10^{-4}$$

b). Tamaño del directorio de datos a lo largo de tres semanas sabiendo que en cada activación se añaden 3KB.

Activacion= 3KB
6 activaciones/hora

$$3 \cdot 7 \cdot 24 \cdot 6 = 3024$$

$$3024 \cdot 3 = 9072KB$$

c). Tamaño del directorio de datos si en cada activación se almacena el doble de cantidad de KB que en la grabación anterior y sabiendo que en la primera activación se añaden 3KB. ¿Cuál sería el tamaño de disco necesario para almacenar el histórico semanal? Dar una cantidad aproximada en TB.

Tenemos la definición recursiva DR:

$$x_n = 3KB$$
$$x_{n+1} = 2 \cdot x_n$$

Por inducción sacamos la definición explícita:

$$x_n = 3 \cdot 2^n$$

Y sacamos la serie:

$$\sum_0^{1007} x_i = 3\sum_0^{1007} 2^n = 3\left[\frac{2^{1007+1} - 2^0}{2-1}\right] = 3 \cdot (2^{1008} - 1)\text{KB}$$

Concluimos que es demasiado.

2.2.6. Ejercicio 6

Un monitor A se activa cada 20 minutos y tarda 500 ms en ejecutarse. En cada activación recoge datos que almacena en un archivo de tamaño variable entre 2 y 3 KB. Otro monitor B se activa cada vez que ocurre un evento cuya probabilidad es de 0.75 en cualquier periodo de 10 minutos. Cuando esto ocurre, se graba un archivo de tamaño 10 KB y cuando no ocurre, se graba la situación en una palabra de tamaño 8 bits. La ejecución del monitor B es de 2 segundos. Se pide:

a). Estimar la sobrecarga producida al sistema informático por cada uno de los dos monitores A y B.

$$\text{overhead (sobrecarga)}_A = \frac{\text{tiempo}}{\text{periodo}} = \frac{0{,}5}{1200} = 4{,}16 \cdot 10^{-4} = 0{,}41\,\%$$

$$\text{sobrecarga}_B = \frac{2}{10 \cdot 60} \cdot 0{,}7 = 0{,}24s$$

b). Tamaño del directorio de datos a lo largo de una semana para cada caso.

Monitor A
Archiva 504 veces por semana.
Tamaño $= [2 \cdot 504, 3 \cdot 504] = [1008, 1512]$.

Monitor B:

$$\frac{10080}{10} = 1008 \begin{cases} 0{,}75 \to 756 \cdot 10\text{KB} \\ 0{,}25 \to 252 \cdot 8\text{bits} \end{cases}$$

Suma media ponderada 7560.24 KB.

c). ¿Cuál sería el monitor más adecuado para implantar en un sistema informático supuesto que los datos recogidos son de calidad (información) similar?

El A es muchísimo mejor.

2.2.7. Ejercicio 7

Analiza el comportamiento que se infiere de la siguiente Load Average:

6.85 ; 7.37 ; 7.83.

Determina la muestra de más estacionaria y la más variable en los últimos 15 minutos acorde con los datos proporcionados. Calcula la varianza máxima y proporciona la desviación típica máxima como un indicador de alerta para el sistema informático.

Tenemos un sistema no estable:

$$\left.\begin{array}{l} l_1 = 6{,}85 \\ l_2 = 7{,}37 \\ l_3 = 7{,}83 \end{array}\right\} \begin{array}{l} L_2 = \frac{1}{5}\sum_1^{15} x_i \\ \\ L_3 = \frac{1}{15}\sum_1^{15} x_i \end{array}$$

Recordemos que:
Tupla más inestable $(\alpha, 0, 0, .., 0, \beta, 0, 0, 0, \gamma)$
Más estable $(\alpha, \alpha, \alpha, ...\alpha, \beta, \beta, \beta, \beta, \gamma)$
Y que:
$\gamma = L_1$

$$\frac{\beta + 0 + 0 + 0 + \gamma}{5} = L_2$$

$$\frac{\alpha + 0 + ... + 0 + \beta + 0 + 0 + 0 + \gamma}{15} = L_3$$

Entonces:

$$\left.\begin{array}{l} \gamma = L_1 \\ \frac{\beta + 0 + 0 + 0 + \gamma}{5} = L_2 \\ \frac{\alpha + 0 + 0... + 0 + \beta + 0, 0, 0 + \gamma}{15} = L_3 \end{array}\right. \rightarrow \begin{array}{l} \beta = 29{,}9 \\ \alpha = 80{,}7 \\ \gamma = 6{,}85 \end{array}$$

La más inestable: (80.7,0...0,29.9,0,0,0,6.85).
La más estable: (8.07,8.07,...8.07,7.47,7.47,7.47,7.47).

Varianza máxima:

$$\text{VAR} = E[x^2] - E[x]^2 = \frac{80 \cdot 7^2 + 29 \cdot 9^2 + 6 \cdot 85^2}{15} - \left(\frac{80 \cdot 7 + 29 \cdot 9 + 6 \cdot 85}{15}\right)^2 = 435{,}38$$

2.2.8. Ejercicio 8

Se sabe que la sobrecarga de un monitor software sobre un computador es del 4 %. Si el monitor se activa cada 2 segundos.

a). ¿Cuánto tiempo tarda el monitor en ejecutarse por cada activación?

$$0{,}04 = \frac{x}{2} \rightarrow x = 0{,}08s$$

b). En cada activación se almacena un registro de 5 bytes, ¿cuánto ocupará el directorio de datos tras dos semanas sabiendo que actualmente almacena un histórico de tres semanas más?

Activacion = 5B
2+3=5 semanas = 3024000 segundos.

$$\frac{3024000}{2} = 1512000 \quad \text{activaciones en 5 semanas}$$

$$1512000 \cdot 5 = 7560000 \text{ B} = 7{,}21 \text{ MB}$$

c). Se pretende reducir el overhead en un 500 % sin modificar el monitor, ¿cada cuánto tiempo deberán programarse las instancias?

$$1 + \frac{500}{100} \cdot \frac{0{,}04}{6} \geq \frac{0{,}08}{7} \rightarrow f \geq 12s$$

2.2.9. Ejercicio 9

a). Se ha ejecutado la siguiente orden en un sistema: **$ time quicksort**

```
real  0m 40.2s
user  0m 17.1s
sys   0m 3.2s
```

Indicar si el sistema está soportando mucha o poca carga. Razonar la respuesta.

Recordemos que:

- real: tiempo total usado por el sistema (tiempo de respuesta).
- user: tiempo de CPU ejecutando en modo usuario.
- sys: tiempo de CPU en modo supervisor, ejecutando código del núcleo.

Tiempo de espera = real-(user+sys), por tanto mucha, porque prácticamente el doble está ocupada por otros elementos.

b). El monitor sar se activa cada 15 minutos y tarda 750 ms en ejecutarse por cada activación. Calcular la sobrecarga que genera este monitor sobre el sistema.

$$\text{sobrecarga} = \frac{750 \cdot 10^{-3}}{15 \cdot 60} = 0{,}08\,\%$$

c). La información generada en cada activación ocupa 8192 bytes, ¿cuántos ficheros históricos del tipo saDD se pueden almacenar en /var/log/sa si se dispone únicamente de 200 MB de capacidad libre?

$$24 \cdot 4 = 96 \text{ activaciones/día}$$

$$\frac{200 \cdot 1024 \text{ bytes}}{96 \cdot 8192 \text{ bytes}} = 266{,}66 \rightarrow 266{,}66 \text{ ficheros históricos}$$

2.2.10. Ejercicio 10

El día 8 de octubre se ha ejecutado la siguiente orden en un sistema: **$ ls /var/log/sar**

```
-rw-r--r-- 1 root root 3049952 Oct 6 23:50 sa06
-rw-r--r-- 1 root root 3049952 Oct 7 23:50 sa07
-rw-r--r-- 1 root root 2372320 Oct 8 18:40 sa08
```

¿Cada cuánto tiempo se activa el monitor **sar** instalado en el sistema? ¿Cuánto ocupa el registro de información almacenada cada vez que se activa el monitor?

Hay 144 tramos de 10 minutos en 1 día.
20.68KB lo que ocupa el registro en cada activación.

Tema 3

Análisis comparativo

3.1. Teoría previa

3.1.1. Índices de rendimiento

- **CPI**

$$\text{CPI} = \frac{\text{Ciclos}}{\text{Instruccion}}$$

- **MIPS**

$$\text{MIPS} = \frac{\text{N}^{\text{o}}\ \text{Instrucciones}}{\text{T. ejecución} \cdot 10^6} = \frac{\text{Freq de reloj}}{\text{CPI} \cdot 10^6}$$

- **MFLOPS**

$$\text{MFLOPS} = \frac{\text{N}^{\text{o}}\ \text{instrucciones FP}}{\text{T. ejecución} \cdot 10^6}$$

3.1.2. ¿Cómo expresar el rendimiento?

- **Media aritmética**

$$x_a = \frac{1}{n}\sum_{i=1}^{n} x_i$$

- **Media armónica**

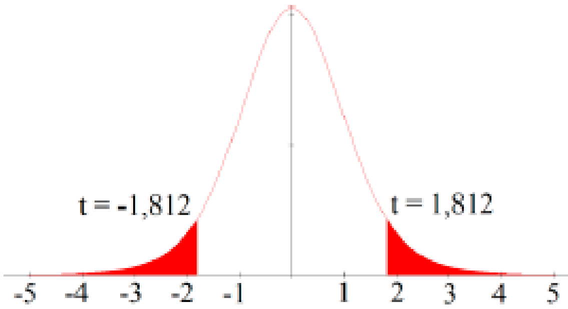
$$x_f = \frac{n}{\sum_{i=1}^{n} \frac{1}{x_i}}$$

- **Media geométrica**

$$x_g = (\prod_{i=1}^{n} x_i)^{\frac{1}{n}}$$

3.1.3. Diferencias estadísticamente significativas

TABLA
DISTRIBUCIÓN t DE STUDENT



Ejemplos:
Para n-1 = 10 grados de libertad
$P(t > 1,812) = 0,05$
$P(t < -1,812) = 0,05$

n-1 \ α	0,25	0,2	0,15	0,1	**0,05**	0,025	0,01	0,005	0,0005
1	1,0000	1,3764	1,9626	3,0777	6,3138	12,7062	31,8205	63,6567	636,6192
2	0,8165	1,0607	1,3862	1,8856	2,9200	4,3027	6,9646	9,9248	31,5991
3	0,7649	0,9785	1,2498	1,6377	2,3534	3,1824	4,5407	5,8409	12,9240
4	0,7407	0,9410	1,1896	1,5332	2,1318	2,7764	3,7469	4,6041	8,6103
5	0,7267	0,9195	1,1558	1,4759	2,0150	2,5706	3,3649	4,0321	6,8688
6	0,7176	0,9057	1,1342	1,4398	1,9432	2,4469	3,1427	3,7074	5,9588
7	0,7111	0,8960	1,1192	1,4149	1,8946	2,3646	2,9980	3,4995	5,4079
8	0,7064	0,8889	1,1081	1,3968	1,8595	2,3060	2,8965	3,3554	5,0413
9	0,7027	0,8834	1,0997	1,3830	1,8331	2,2622	2,8214	3,2498	4,7809
10	0,6998	0,8791	1,0931	1,3722	**1,8125**	2,2281	2,7638	3,1693	4,5869
11	0,6974	0,8755	1,0877	1,3634	1,7959	2,2010	2,7181	3,1058	4,4370

Supongamos la ejecución de n programas en dos máquinas A y B. ¿Son significativas las diferencias obtenidas? Para saberlo utilizamos el intervalo de confianza:

$$\bar{x} \pm t_{1-\frac{\alpha}{2},n-1} \cdot \frac{s}{\sqrt{n}} \tag{3.1}$$

Donde:

- t es el valor obtenido de la tabla T-student.
- s es la desviación típica.
- α es el nivel de confianza.

3.2. Ejercicios resueltos

3.2.1. Ejercicio 1

Un programa ejecuta un total de 120×10^6instrucciones. De ellas, el 75 % se ejecutan en 3 ciclos de reloj, mientras que el resto lo hace en 5 ciclos. Tras medir el tiempo de ejecución de este programa mediante la orden time del sistema operativo se ha obtenido la siguiente información:

```
real  0m 84s
user  0m 34s
sys   0m 1s
```

Calcular el número medio de ciclos por instrucción (CPI) obtenidos por el programa, la frecuencia del procesador y los MIPS.

$$120 \cdot 10^6 \begin{array}{l} \rightarrow 75\,\% \rightarrow 3 \quad \text{ciclos} \\ \rightarrow 25\,\% \rightarrow 5 \quad \text{ciclos} \end{array}$$

$$\text{CPI} = \frac{(3 \cdot 0{,}75 \cdot 120 \cdot 10^6) + (5 \cdot 0{,}25 \cdot 120 \cdot 10^6)}{120 \cdot 10^6} = 3 \cdot 0{,}75 + 5 \cdot 0{,}25 = 3{,}5$$

$$\text{CPI} = \frac{T_{\text{ej}\cdot F_q}}{\text{NºI}} \rightarrow F_q = \frac{\text{CPI} \cdot \text{NºI}}{T_{\text{ej}}} \rightarrow F_q = \frac{\text{CPI} \cdot \text{NºI}}{T_{\text{ej}}} = \frac{3{,}5 \cdot 120 \cdot 10^6}{(34+1)} = 12\text{Mhz}$$

$$\text{MIPS} = \frac{F_q}{\text{CPI} \cdot 10^6} = 3{,}42$$

3.2.2. Ejercicio 2

Un estudio llevado a cabo mediante un monitor de ejecución de programas ha permitido cuantificar el tiempo medio de ejecución de las instrucciones que emplea una aplicación informática. Esta aplicación se ha ejecutado en dos procesadores P y Q, con el mismo juego de instrucciones y se ha obtenido el siguiente resultado:

Tipo de Instrucción	**Frecuencia en %**	**Tiempo en P en ns**	**Tiempo en Q en ns**
Memoria	25	70	72
Comparación	35	32	27
Salto	25	13	10
Otras	15	18	12

a). Calcular el tiempo medio de ejecución de una instrucción en cada procesador y utilizar el resultado para cuantificar la mejora conseguida por el procesador más rápido.

$$TP = 0{,}25 \cdot 70 + 0{,}35 \cdot 32 + 0{,}25 \cdot 13 + 0{,}15 \cdot 18 = 34{,}65$$
$$TQ = 0{,}25 \cdot 72 + 0{,}35 \cdot 27 + 0{,}25 \cdot 10 + 0{,}15 \cdot 12 = 31{,}75$$

$$\text{ratio} = \frac{TP}{TQ} = 1 + \frac{9{,}13}{100}$$

Q es 9.13 veces más rápido que R.

b). Determinar el nuevo tiempo medio de ejecución de una instrucción en el procesador si un nuevo diseño consigue que todas las instrucciones se ejecuten un 15 % más rápidamente.

$$T_{\text{mej}}R = \frac{34{,}65}{1{,}15} = 30{,}13$$

$$T_{\text{mej}}Q = \frac{31{,}75}{1{,}25} = 27{,}60$$

3.2.3. Ejercicio 3

Considérese un programa de cálculo numérico que se ejecuta en dos minutos y hace las operaciones de coma flotante que se indican en la tabla ¿Cuál es el rendimiento conseguido por el computador con este programa de cálculo atendiendo a los MFLOPS? ¿Y con los MFLOPS normalizados?

Operación	**Cantidad**	**Operaciones normalizadas**
ADD	78×10^6	1
SQRT	29×10^6	3
COS	13×10^6	8
EXP	42×10^6	12

Sin normalizar:

$$\text{MFLOPS} = \frac{\text{n}^{\text{o}}\ \text{inst}}{t \cdot 10^6} = \frac{120}{160} = 1{,}35\ \text{MFLOPS}$$

Normalizado:

$$\text{MFLOPS} = \frac{78 \cdot 10^6 + 29 \cdot 10^6 \cdot 3 + 13 \cdot 10^6 \cdot 8 + 42 \cdot 10^6 \cdot 12}{120 \cdot 10^6} = 6{,}442\ \text{MFLOPS}$$

3.2.4. Ejercicio 4

El rendimiento de un programa que implementa un algoritmo numérico varía de acuerdo con las distintas secciones de código. En concreto, la generación de resultados del algoritmo se distribuye de acuerdo con los siguientes MFLOPS:

Porcentaje de resultados	**MFLOPS**
30 %	1
20 %	10
50 %	100

Calcular el valor medio de los MFLOPS obtenidos por el algoritmo ¿Cómo se distribuye el tiempo de ejecución en función de los MFLOPS?

$$x = \frac{1}{\sum \frac{wi}{xi}}$$

$$\frac{1}{\frac{0{,}3}{1} + \frac{0{,}2}{10} + \frac{0{,}5}{100}} = 3{,}08 \text{ MFLOPS}$$

$$30\,\% \to \frac{0{,}3}{0{,}325} = 0{,}923 \to 92{,}3\,\% \quad \text{se ejecuta 1 MFLOP}$$

$$20\,\% \to \frac{0{,}02}{0{,}325} = 0{,}062 \to 6{,}2\,\% \quad \text{a 10 MFLOP}$$

$$50\,\% \to \frac{0{,}005}{0{,}325} = 0{,}015 \to 1{,}5\,\% \quad \textbf{a 100 MFLOPS}$$

3.2.5. Ejercicio 5

Calcula los índices de rendimiento SPEC int_base2000 y SPECint2000 de los sistemas A y B a partir de las siguientes medidas:

Programa	**Referencia (Base & Peak)**	**A Base Run Time**	**A Peak Run Time**	**B Base Run Time**	**B Peak Run Time**
P1	2100	456	440	420	415
P2	2400	792	780	810	805
P3	3200	820	796	816	715

Observación: SPECint_base2000 (SPECint2000) se calcula como 100 x MediaGeométrica(ratios Ri/Ti) siendo Ri el valor de referencia y Ti el valor de Base Run Time (Peak Run Time).

Media geométrica:

$$X_t = \sqrt[n]{\prod_1^n xi} = \prod_1^n xi^{\frac{1}{w}}$$

$\mathrm{A}_{\text{base}} \to x_1 = \dfrac{210}{456} = 4{,}61;\ x_2 = \dfrac{2500}{792} = 3{,}02;\ x_3 = \dfrac{3200}{820} = 3{,}9;\ n = 3.$

Solución $\mathrm{A}_{\text{base}} = \mathrm{SPEC}_A = 100 \cdot \sqrt[B]{4{,}61 \cdot 3{,}02 \cdot 3{,}9} = 100 \cdot \sqrt{4{,}61 \cdot 3{,}02 \cdot 3{,}9}$

$\mathrm{SPEC}_B = 100 \cdot \sqrt[3]{58{,}03} > 100\sqrt[3]{54{,}29}$

B es mejor que A.

3.2.6. Ejercicio 6

Calcula los MFLOPS promedio de un sistema a partir de la siguiente tabla, que muestra el tiempo de ejecución y el número de operaciones de coma flotante de cuatro programas:

Programa	Tiempo (s)	Operaciones
P1	878	230×10^9
P2	491	210×10^9
P3	375	151×10^9
P4	427	120×10^9

$$P1 = \frac{230 \cdot 10^9}{878 \cdot 10^6} = 267$$

$$P2 = \frac{210 \cdot 10^9}{491 \cdot 10^6}$$

$$P3 = \frac{151 \cdot 10^9}{375 \cdot 10^6}$$

$$P4 = \frac{120 \cdot 10^9}{427 \cdot 10^6}$$

$$\text{MFLOPS}(P1, P2, P3, P4) = \frac{(230 + 210 + 150 + 120) \cdot 10^9}{(878 + 491 + 575 + 427)} = 327{,}5\text{MFLOPS}$$

$$X_H = \frac{4}{\frac{1}{267} + \frac{1}{P2} + \frac{1}{P3} + \frac{1}{P4}} = 328\text{MFLOPS}$$

3.2.7. Ejercicio 7

La página oficial de SPEC muestra los siguientes resultados de rendimiento para dos sistemas informáticos obtenidos mediante el benchmark CPU2000:

Sistema	Modelo	SPECint_base	SPECint2000
A	Altos G5350 (AMD Opteron 246)	1347	1438
B	Altos G5350 (AMD Opteron 254)	1788	1918

a). ¿Cuál de los dos sistemas presenta mejor rendimiento? Cuantifique numéricamente la mejora.

$$\text{SPEC}_{\text{int}}\text{\_base2000} = \frac{1788}{1347}$$

$$\text{SPECint2000} = \frac{1918}{1438}$$

b). A la vista de los resultados anteriores, ¿afecta al rendimiento de ambos sistemas la optimización realizada por el compilador en las pruebas?

$$\frac{\text{max}}{\text{min}} = 7{,}2\,\% \quad \text{mejor con optimización}$$

c). ¿En qué medida se reflejará en los resultados anteriores una mejora importante en la unidad de coma flotante del procesador?

Los índices SPEC solo se basan en aritmética entera.

d). ¿Cuál de los dos sistemas ejecutará el benchmark Whetstone más rápidamente?

3.2.8. Ejercicio 8

Tenemos dos propuestas para actualizar los equipos informáticos. El precio es de 1300 euros y 1450 euros respectivamente para los equipos A y B. Para tomar una decisión han decidido hacer una prueba con las dos alternativas y han ejecutado en cada una de las opciones los 8 programas que utilizan habitualmente, obteniendo los tiempos de ejecución que se muestran en la tabla (en segundos):

Programa	**Modelo A**	**Modelo B**
1	23.6	24.0
2	33.7	41.6
3	10.1	8.7
4	12.9	13.5
5	67.8	66.4
6	9.3	15.2
7	47.4	50.5
8	54.9	52.3

Determínese con un nivel de confianza del 99 % si existen diferencias significativas entre las alternativas A y B y en caso afirmativo indicar cuál es la mejor opción. más rápidamente?

di=Ai-Bi	di^2
-0.4	0.46
-7.9	62.11
+1.4	1.96
-0.6	0.36
+1.4	1.94
-5.9	34.81
-3.1	9.61
2.6	6.46
-12.5	118.03
$\vec{x}$=-2.562	114.63

Paso 1:

$$\vec{x} = -1{,}56$$

$$s^2 = E[di^2] - E[di]^2 = 14{,}753 - (-1{,}56) = 12{,}31$$

$$s = \sqrt{12{,}31} = 3{,}5$$

Paso 2:
Tabla de la t_7:

$$\alpha = 0{,}01 \rightarrow \frac{\alpha}{2} = 0{,}05$$

$$t = 3{,}4975$$

Paso 4:

$$\vec{x} \pm t_{\frac{\alpha}{2}+7} \frac{5}{\sqrt{n}} = -1{,}56 \pm 3{,}4995 \cdot \frac{3{,}5}{\sqrt{8}} = (-5{,}8897, 27697)$$

Paso 5:
Como 0 $\in$ al intervalo, no hay diferencia estadístico o entre A y B, pero esta desplazado al lado negativo. Por lo que para esta muestra el sistema A es levemente mejor.

3.2.9. Ejercicio 9

En la tabla se indican los tiempos de ejecución de dos computadores A y B de un conjunto de programas de prueba para aritmética entera (gcc, latex, gzip, bzip2) y aritmética de coma flotante (float, trilog, savage, linpack, whetstone). Determinar a nivel α=0.05 si hay diferencias significativas 1. en total, 2. en aritmética entera y 3. en coma flotante:

Programa	**Modelo A**	**Modelo B**
gcc	58.2	43.6
latex	32.1	21.3
gzip	42.9	32.8
bzip	11.0	8.2
float	54.2	40.2
trilog	46.6	45.1
savage	49.3	46.3
Linpack	25.8	32.8
whetstone	52.0	58.3

di=Ai-Bi	di^2
14.6	213.16
10.8	116.64
10.1	102.01
2.8	7.84
14	196
1.5	2.25
3	9
-7	49
-6.3	39.69
$\sum$43.5	735.59
$\vec{x}$=4.833	$\vec{x}$=81.732

$$\vec{x} = 4{,}833$$

$$s^2 = 81{,}732 - 4{,}833 = 76{,}898$$

$$s = \sqrt{76{,}89} = 8{,}769$$

$$\vec{x} \pm t_{\frac{\alpha}{2}+7} \frac{s}{\sqrt{n}} = 4{,}833 \pm 2{,}365 \cdot \frac{8{,}769}{\sqrt{9}} = (-2{,}079, 11{,}745)$$

Total intervalo→ No diferencia significativa.
ENT:

$$\vec{x} = 9{,}575$$

$$s^2 = 109{,}91 - 9{,}575 = 100{,}33$$

$$s = 10{,}01$$

$$(-2{,}26, 21{,}41) \to \text{No diferencia}$$

FLOAT:

$$\vec{x} = 1{,}04; s^2 = 59{,}188 - 1{,}04 = 58{,}148$$

$$s = 7{,}625$$

$$(-7{,}02, 9{,}10) \to \text{No diferencia}$$

3.2.10. Ejercicio 10

Con el objetivo de analizar el rendimiento de diseño de la memoria cache, se ha medido el tiempo de ejecución de una serie de programas de prueba. Estos tiempos, expresados en segundos, son los siguientes:

Programa	**Con 4 bloques de 64 palabras**	**Con 8 bloques de 32 palabras**
P1	228	182
P2	213	181
P3	198	226
P4	239	122
P5	217	105
P6	245	198

Determina si las diferencias observadas son significativas con nivel de significación del 95 % y, en caso afirmativo, calcula la mejora conseguida en el rendimiento debido al diseño más adecuado de la memoria cache.

di	di^2
46	2116
32	1024
-28	784
117	13689
112	12544
47	2209
$\vec{x}$=54.33	$\vec{x}$=5394.33

Paso 1:

$$s^2 = 5394 - 54{,}33 = 5340$$

$$s = 73{,}075$$

Paso 2:

$$\alpha = 0{,}05; \frac{\alpha}{2} = 0{,}025 \rightarrow t = 2{,}365$$

Paso 3:

$$\vec{x} \pm t \cdot \frac{5}{\sqrt{n}} = (-16{,}22, 122{,}884) \rightarrow \text{cae}$$

Tema 4

Análisis Operacional y Flujo

4.1. Teoría previa

4.1.1. Estación de servicio

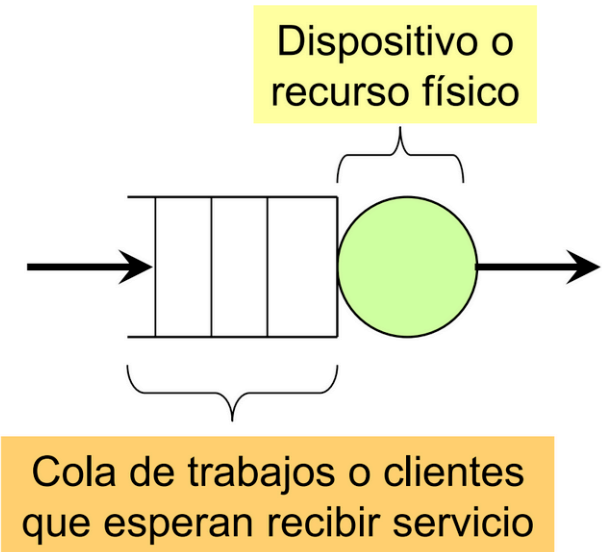


Figura 4.1: Estación de servicio

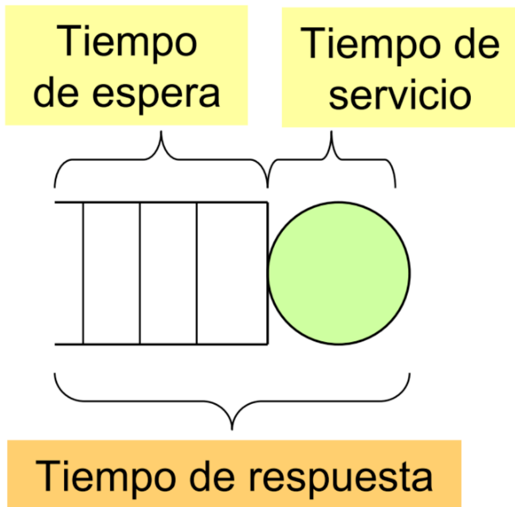


Figura 4.2: Tiempos

Una **estación de servicio** es un objeto abstracto compuesto por un servidor y una cola de espera. En ella los trabajos o tareas entrantes (A) hacen cola ordenadamente para ser procesadas y transformadas en tareas de salida (C).

Por ejemplo una impresora puede tener varios trabajos para imprimir en cola, que de forma ordenada va transformando en impresiones en papel.

4.1.2. Nomenclatura y variables

- A: tareas entrantes.
- C: tareas salientes.
- B: tiempo ocupado (busy).
- T: tiempo de medición.
- λ: tasa de llegadas.
- X: productividad.
- U: utilización.
- S: tiempo de servicio / ejecución.
- V: número de accesos.
- R: tiempo de respuesta / acceso.
- W: tiempo de espera (wating).
- N: número de trabajos (en cola y en servicio) en una estación.
- Q: número de trabajos en espera (queued).
- D: demandas de servicio.
- Z: Tiempo de reflexión.
- **HFE**: Hipótesis de flujo equilibrado.

4.1.3. Fórmulas y leyes

$$\lambda = \frac{A}{T}[t/s]$$

$$X = \frac{C}{T} = \frac{1}{D}[t/s]$$

$$E_{rel} = \frac{|A - C|}{C}$$

$$E_{abs} = |A - C|$$

$$U = \frac{B}{T} = X \cdot S[s]$$

$$S = \frac{U}{X}[s/t]$$

$$S = \frac{B}{C}[s/t]$$

$$S = T_{posic} + T_{latencia} + T_{transf}[s/t]$$

$$R = \frac{B}{T}[s]$$

$$R = \frac{1}{n}\sum_{i=1}^{n} R_i[s]$$

$$R = \sum_{i=1}^{n} V_i \cdot R_i = \sum_{i=1}^{n} D_i[s]$$

$$R = \sum_{i=1}^{n} C_i - A_i[s]$$

$$W = R - S[s]$$

$$X(N) \leq min(\frac{N}{D + Z}, \frac{1}{D_b})$$

$$D = V \cdot S$$

$$R(N) \geq max(D, N \cdot D_b - Z)$$

Hipótesis de flujo equilibrado

La Hipótesis de flujo equilibrado lo que supone es que la cantidad de tareas entantes es igual a la cantidad de tareas salientes: $A = C$.
Esto implica que la tasa de llegadas coincide con la productividad: $\lambda = X$

Error cuadrático medio

Cuando queremos calcular el error que supone asumir HFE con respecto a **varios periodos** de medición (n) debemos usar el error cuadrático medio.

$$ECM = \frac{1}{n}\sum_{i=1}^{n}(A_i - C_i)^2$$

Ley de Little

Relaciona el número de trabajos en el sistema con el tiempo de permanencia y su productividad o tasa de llegada.

$$N = \lambda \cdot R = X \cdot R$$

La ley de little también se puede aplicar a programas en cola:

$$N = \lambda \cdot W = X \cdot W$$

Ley de flujo forzado

Establece que el flujo a través de un determinado dispositivo determina el flujo en cualquier otro dispositivo. La ley es válida solo si también lo es la HFE.

$$X = V_i \cdot X_0$$

donde X es la productividad del sistema, X_0 es la productividad del dispositivo y V_i es la razón de visitas al dispositivo.

Cuello de botella

En el contexto de esta asignatura llamaremos cuello de botella al dispositivo de un sistema que posea mayor demanda (D).

4.2. Ejercicios resueltos

Los ejercicios del 1 al 8 son de sistemas interactivos y los ejercicios del 9 al 12 son sobre sistemas transaccionales, siendo los dos últimos sobre grafos.

4.2.1. Ejercicio 1

El disco de un computador se ha monitorizado durante un periodo de 30 segundos. Durante este tiempo han llegado 11 peticiones y han acabado 12. Se sabe que el disco ha estado vacío durante 2.5 segundos y se ha podido medir el tiempo de respuesta de 9 peticiones. Estos tiempos, expresados en segundos, son:
8.2, 9.1, 2.3, 5.9, 2.0, 6.2, 4.1, 6.5, 7.3.
Se pide calcular:

a). La exactitud con que se cumple la **hipótesis del flujo equilibrado** de trabajos de acuerdo a los datos. ¿Cómo se calcularía si se hubieran monitorizado 10 periodos de 30 segundos?

> Lo que nos estan pidiendo es el error que tienen los resultados obtenidos con respecto a **HFE**.
> Existen dos tipos de errores:
>
> - **Error relativo**: normalmente lo calcularemos siempre con respecto a la salida (dividimos por C_i):
> $E_r = \dfrac{|A_i - C_i|}{C_i} = \dfrac{1}{12} = 0{,}083$
>   El error relativo es del 8,3 %
> - **Error absoluto**: $E_a = |A_i - C_i| = 1$
>
> Para los casos en los que queremos medir varios periodos debemos recurrir al **error cuadrático medio**. $ECM = \dfrac{1}{n}\sum_{i=1}^{n}(A_i - C_i)^2 = \dfrac{1}{10}\sum_{i=1}^{10}(A_i - C_i)^2$

b). La tasa de llegadas de peticiones al disco y el tiempo entre llegadas.

> $\lambda = \dfrac{A}{T} = \dfrac{11}{30} = 0{,}367[t/s]$; T entre llegadas $= \dfrac{1}{\lambda} = \dfrac{30}{11} = 2{,}73[s]$

c). La productividad del disco.

> $X = \dfrac{C}{T} = \dfrac{12}{30} = 0{,}4[t/s]$

d). La utilización del disco.

En este caso, el tiempo de ocupación lo obtenemos de restar el tiempo de medición menos el tiempo que ha estado vacío:
$B = T - T_{vacio} = 30 - 2{,}5 = 27{,}5$

$U = \dfrac{B}{T} = \dfrac{27{,}5}{30} = 0{,}917 = 91{,}7\,\%$

e). El tiempo medio de servicio del disco.

$S = \dfrac{U}{X} = \dfrac{0{,}917}{0{,}4} = 2{,}29[s/t]$

f). El tiempo medio de espera en cola.

Primero calculamos el tiempo de respuesta con la fórmula del sumatorio.
$R = \frac{1}{n}\sum_{i=1}^{n} R_i = \frac{1}{9}\sum_{i=1}^{9} R_i = 5{,}73[s]$
Y sustituimos
$W = R - S = 5{,}73 - 2{,}29 = 3{,}44[s]$

4.2.2. Ejercicio 2

En un sistema cliente-servidor se considera que las transacciones usan 4ms de procesador en el cliente, 6ms de procesador en el servidor y cada transacción visita la unidad de disco un total de 12 veces, es decir, se leen 12 bloques (de 1024 bytes) del disco del servidor. De las características técnicas del disco se sabe que el tiempo medio de posicionamiento es de 8ms, la latencia media es 3.6ms y el tiempo de transferencia es $42{,}66x10^{-6}$ segundos. Se pide:

a). El tiempo de servicio de las transacciones en los procesadores del cliente y del servidor, expresadas en segundos.

De los datos del enunciado podemos sacar la siguiente tabla que representa una transacción:

Dispositivo	**Vi**	**Si(ms)**
Cliente	12	4
Servidor		6

Como nos piden S en segundos, simplemente tenemos que convertir:
$S_{cliente} = 4 \cdot 10^{-3}[s]$; $S_{servidor} = 6 \cdot 10^{-3}[s]$

b). El tiempo de servicio S del disco en un caso general, es decir teniendo en cuenta el tiempo de posicionamiento, la latencia y el tiempo de transferencia.

$S = T_{posic} + T_{latencia} + T_{transf} = 8 \cdot 10^{-3} + 3{,}6 \cdot 10^{-3} + 0{,}04266 \cdot 10^{-3} = 11{,}64266 \cdot 10^{-3}[s]$

c). El tiempo de una transacción suponiendo dos casos: que los bloques estén grabados en pistas diferentes (peor caso) o situados de forma consecutiva (mejor caso).

En todas las transacciones debemos de tener en cuenta el tiempo de transferencia.
En el **caso peor** todas las transacciones implican posicionamiento y latencia, por tanto:
$12 \cdot S = 12 \cdot 11{,}64266 \cdot 10^{-3} = 139{,}7119 \cdot 10^{-3}[s]$
En el **caso mejor** solo deberá posicionarse en el primer acceso, por tanto:
$12 \cdot T_{transf} + (T_{posic} + T_{latencia}) = 12 \cdot 0{,}04266 \cdot 10^{-3} + (8 \cdot 10^{-3} + 3{,}6 \cdot 10^{-3})[s]$

d). ¿Qué componentes del tiempo de servicio del disco influyen más en el rendimiento? ¿Por qué son importantes las operaciones de compactificación de discos?

Los componentes que más influyen son el tiempo de posicionamiento y el tiempo de latencia.
Para reducir los tiempos de acceso a la información.

4.2.3. Ejercicio 3

Durante una sesión de medida de media hora, un monitor software ha extraído las siguientes variables operacionales básicas de un servidor web:

Variable	**Valor**
A	364 peticiones
C	359 peticiones
B	23 minutos

Calcula las siguientes variables operacionales deducidas del servidor web: tasa de llegadas, productividad, utilización y tiempo medio de servicio.

$$\lambda = \frac{A}{T} = \frac{364}{30 \cdot 60} = 0{,}202[t/s];\ X = \frac{C}{T} = \frac{359}{30 \cdot 60} = 0{,}199[t/s]$$
$$U = \frac{B}{T} = \frac{23 \cdot 60}{30 \cdot 60} = 0{,}766[s];\ S = \frac{U}{X} = \frac{0{,}766}{0{,}199} = 3{,}844[s/t]$$

4.2.4. Ejercicio 4

Un procesador recibe una media de 5 programas por segundo. Cada programa experimenta un tiempo medio de ejecución de 0.14 segundos y un tiempo medio de respuesta de 8 segundos. Se pide:

- Uso medio del procesador.

El enunciado nos dice que: $S = 0{,}14[s]$; $A = 5[t]$; $R = 8[s]$
Sabemos que:
$S = \frac{U}{X}$; $X = \frac{C}{T}$; $\lambda = \frac{A}{T}$
Como no tenemos C, podemos asumir **HFE**, por tanto:
$S = \frac{U}{\lambda}$, y calculamos:

$\lambda = \frac{A}{T} = \frac{5}{1} = 5[t]$ al ser una media, ponemos $T = 1$

$S = \frac{U}{\lambda}$; $U = S \cdot \lambda = 0{,}14 \cdot 5 = 0{,}7 = 70[\%]$

- Tiempo medio de espera en la cola del procesador.

$W = R - S = 8 - 0{,}14 = 7{,}86[s]$

- Número medio de programas en la cola de espera del procesador.

Aplicamos la ley de little: $N = X \cdot W = 5 \cdot 7{,}86 = 39{,}3[t]$

4.2.5. Ejercicio 5

Después de monitorizar el procesador de un servidor web durante un periodo de 30 segundos, se sabe que ha sido utilizado durante 27 segundos. También se han contabilizado 74 llegadas y 72 salidas de peticiones. Se pide:

a). Porcentaje de error cometido al asumir la hipótesis del flujo equilibrado.

Calculamos el error relativo: $E_r = \frac{|A_i - C_i|}{C_i} = \frac{2}{72} = 0{,}027 = 2{,}7[\%]$

b). Tasa de llegadas al procesador.

$\lambda = \frac{A}{T} = \frac{74}{30} = 2{,}46[t/s]$

c). Utilización del procesador.

$U = \frac{B}{T} = \frac{27}{30} = 0{,}9[s] = 90[\%]$

d). Si cada trabajo realiza una media de cuatro visitas al procesador, ¿cuál es la productividad del servidor web?

Para hallar la productividad de un dispositivo del sistema debemos aplicar la ley de flujo forzado: $X_i = V_i \cdot X_0$

Por tanto $X_0 = \dfrac{X}{V_i} = \dfrac{2{,}46}{4} = 0{,}6$

e). Se han monitorizado 3 periodos adicionales de 30" y se han obtenido los siguientes datos:

- Llegadas: 74, 75, 77
- Salidas: 73, 72, 70
- Busy: 29, 26, 25

Calcular el ECM de la hipótesis del flujo equilibrado, y los resultados de los apartados b-d para estos nuevos datos.

$$ECM = \frac{1}{n}\sum_{i=1}^{n}(A_i - C_i)^2 = \frac{1}{4}\cdot(2^2 + 1^2 + 3^2 + 7^2) = \frac{63}{4} = 15{,}75$$

- $\lambda = \dfrac{300}{120} = 2{,}5[t/s]$
- $U = \dfrac{107}{120} = 0{,}89 = 89[\%]$
- $X_0 = \dfrac{X}{V_i} = \dfrac{2{,}5}{4} = 0{,}625$

4.2.6. Ejercicio 6

Tenemos una unidad de disco duro cuya controladora dispone de cierta cantidad de memoria caché. El tiempo medio de acceso a la controladora es de 0.1ms, el tiempo medio de posicionamiento del disco es de 5ms, la latencia rotacional media es de 6ms y el tiempo medio de transferencia es de 0.3ms.

Sabiendo que la memoria caché tiene una probabilidad de acierto del 95 %, se pide determinar el tiempo medio de servicio de la unidad de disco. Ídem en el caso de que la probabilidad de acierto sea del 98 %, 80 % y del 70 %.

Para $P = 0{,}95$: $S = 0{,}1 + (5 + 6 + 0{,}3)\cdot 0{,}05 = 0{,}665[ms]$
Para $P = 0{,}98$: $S = 0{,}1 + (5 + 6 + 0{,}3)\cdot 0{,}02 = 0{,}326[ms]$
Para $P = 0{,}80$: $S = 0{,}1 + (5 + 6 + 0{,}3)\cdot 0{,}2 = 2{,}36[ms]$
Para $P = 0{,}70$: $S = 0{,}1 + (5 + 6 + 0{,}3)\cdot 0{,}3 = 3{,}49[ms]$

4.2.7. Ejercicio 7

Considera un modelo cerrado de sistema informático con los siguientes parámetros:

Dispositivo	**Vi**	**Ri(ms)**
Procesador	7	4,3
Disco 1	2	1,5
Disco 2	4	2,3

Determina el tiempo medio de respuesta de una petición a este sistema informático. Si el número medio de peticiones activas en el sistema es 80, ¿cuál es la tasa de llegadas que soporta?

Tiempo medio de respuesta $R = \sum_{i=1}^{3} V_i \cdot R_i = 42{,}3[ms]$

Ley de little: $N = \lambda \cdot R$; $\lambda = \dfrac{N}{R} = \dfrac{80}{0{,}0423} = 1891{,}25[t/s]$

4.2.8. Ejercicio 8

Durante un periodo de medida de 5 segundos se ha obtenido la siguiente información sobre los instantes de llegada y de salida de peticiones a un disco:

Petición	1	2	3	4	5	6	7	8	9	10
Llegada	-	-	-	0.34	1.76	2.21	3.84	4.39	4.77	4.90
Salida	0.98	1.82	2.10	2.58	2.95	3.12	4.24	4.66	-	-

a). ¿Con qué precisión se cumple la ley del flujo equilibrado?

$$E_r = \frac{|A - C|}{C} = \frac{|7 - 8|}{8} = 0{,}125 = 12{,}5[\%]$$

b). ¿Cuál es la productividad y el tiempo medio de respuesta del disco?

$X = \dfrac{C}{T} = \dfrac{8}{5} = 1{,}6[t/s]$

En este caso, para calcular R, debemos quedarnos con las peticiones de las que conocemos todos sus datos (en este caso 4, 5, 6, 7 y 8), que son 5 peticiones.

$R = \dfrac{1}{5} \cdot \sum_{i=4}^{8} C_i - A_i = 1{,}002[s]$

c). ¿Cuántas peticiones soporta de media el disco?

Ley de little: $N = X \cdot R = 1{,}6 \cdot 1{,}002 = 1{,}6032[t]$

d). ¿Cuál es la utilización y el tiempo de servicio del disco?

Recordemos que: $S = \frac{B}{C}$
Para obtener el tiempo ocupado (B) restamos el tiempo total de medición (T) menos el tiempo en el que no se ha estado ejecutando nada, de manera que:
$B = 5 - ((4{,}77 - 4{,}66) + (4{,}39 - 4{,}24) + (3{,}84 - 3{,}12)) = 4{,}02[s]$
Entonces: $S = \frac{4{,}02}{8} = 0{,}5025[s]$

4.2.9. Ejercicio 9

Considera un modelo abierto de sistema informático con los siguientes parámetros:

Dispositivo	**Vi**	**Si[s]**
Procesador (1)	17	0.03
Disco 2	6	0,04
Disco 3	10	0,04

El tiempo medio entre llegadas de clientes es de 0,6 segundos. Se pide calcular:

a). La tasa de llegadas al sistema.

Al tratarse de tiempo medio, usamos 1 como valor de A:
$\lambda = \frac{A}{C} = \frac{1}{0{,}6} = 1{,}66[t/s]$

b). Las demandas de servicio de los dispositivos.

$D = V \cdot S$

Dispositivo	**Vi**	**Si[s]**	**D**
Procesador (1)	17	0.03	0.51
Disco 2	6	0,04	0.24
Disco 3	10	0,04	0.4

c). El dispositivo cuello de botella.

El procesador es el cuello de botella, por que es el dispositivo con mayor demanda. $D_b = 0{,}51[s]$

d). El tiempo mínimo de respuesta del sistema informático.

$R = \sum_{i=1}^{n} D_i = 1{,}15[s]$

e). La productividad de los dispositivos del sistema.

Utilizamos la ley de flujo forzado: $X = V_i \cdot X_0$;
$X_0 = \frac{X}{V_i} = [HFE]\frac{\lambda}{V_i} = \frac{1{,}66}{V_i}$

Dispositivo	**Vi**	**Si(s)**	**D**	X_0
Procesador (1)	17	0.03	0.51	0.09
Disco 2	6	0,04	0.24	0.27
Disco 3	10	0,04	0.4	0.16

f). El valor máximo de la tasa de llegadas que soporta el sistema.

$$\lambda_{max} = X_{max} = \frac{1}{D_{max}} = \frac{1}{0{,}51} = 1{,}961[t/s]$$

g). El tiempo de respuesta de cada dispositivo.

$R_i = (N_i + 1) \cdot S_i$

$$\begin{cases} R_1 = 0{,}2[s] \\ R_2 = 0{,}06[s] \\ R_3 = 0{,}12[s] \end{cases}$$

h). El tiempo de respuesta del sistema informático.

$R = \sum_{i=1}^{n} R_i \cdot V_i = 5[s]$

i). El número de trabajos que hay en el sistema.

$N = \lambda \cdot R = 1{,}6 \cdot 5 = 8.\overline{3}[t]$

4.2.10. Ejercicio 10

Considera la siguiente parametrización del modelo de un sistema informático interactivo con 25 usuarios y un tiempo medio de reflexión de 6 segundos:

Dispositivo	**Vi**	**Si[s]**
Procesador (1)	4	0.5
Disco 2	3	0,75

a). Identifica el cuello de botella.

$D = V \cdot S$

Dispositivo	Vi	Si[s]	D(s)
Procesador (1)	4	0.5	2
Disco 2	3	0,75	2.25

El Disco 2 es el dispositivo con mayor demanda, por lo que es el cuello de botella. $D_b = 2{,}25[s]$

b). Determina el tiempo mínimo de respuesta y la productividad máxima del sistema.

Tiempo mínimo de respuesta: $R_{min} = \sum_i^n D_i = 2 + 2{,}25 = 4{,}25[s]$

La productividad máxima $X_{max} = \frac{1}{D_b} = \frac{1}{2{,}25} = 0.\overline{4}[s]$

c). Calcula los límites del tiempo de respuesta y de la productividad.

Con los límites del tiempo de respuesta nos están pidiendo el tiempo de respuesta máximo:
$R(N) \geq max(D, N \cdot D_b - Z) \geq max(4{,}25, 25 \cdot 2{,}25 - 6) \geq 50{,}25[s]$

El límite de la productividad se refiere a la productividad mínima:
$X(N) \leq min(\frac{N}{D+Z}, \frac{1}{D_b}) \leq min(\frac{6}{4{,}25+6}, \frac{1}{2{,}25}) \leq 0{,}44[t/s]$

d). Calcula el punto teórico de saturación. A la vista de su valor, ¿el sistema se encuentra sometido a baja o alta carga?

$N^{\cdot} = \lceil \frac{D+Z}{D_b} \rceil = \lceil \frac{4{,}25+6}{2{,}25} \rceil = \lceil 4{,}55 \rceil = 5$
Alta carga, ya que actualmente tiene 25 usuarios.

e). Calcula el tiempo medio de respuesta del sistema.

$R = N \cdot D_b - Z = 25 \cdot 2{,}25 - 6 = 50{,}25[s]$

4.2.11. Ejercicio 11

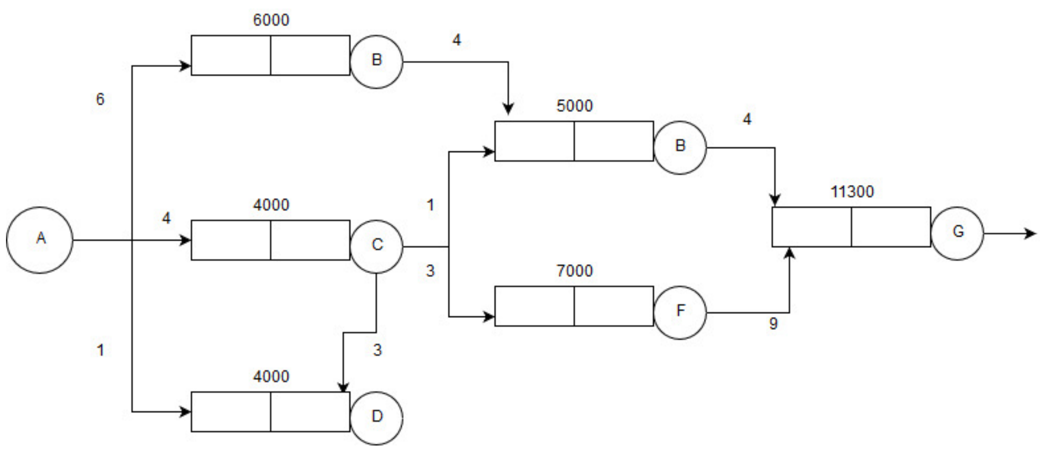
Consideremos un sistema informático con carga transaccional compuesto por 6 estaciones de trabajo descritas a continuación:

- Las estaciones de servicio B, C, D, son localizadores de vuelos cuyas colas tienen capacidad para gestionar 6000, 4000 y 5000 transacciones respectivamente. Una transacción realizada en C puede requerir un refinamiento que le deriva al localizador de vuelos D. En este caso, el servidor D tiene un máximo de aceptación de 3000 transacciones (de las 5000).

- Las estaciones de servicio E y F son gestores de pago del vuelo localizado. El gestor E admite un máximo de 5000 transacciones en cola mientras que F admite 7000. Por razones de distribución de tareas del SI el gestor E admitirá un máximo de 4000 peticiones desde el servidor B y 1000 desde el servidor C. El gestor F, por su parte, admitirá hasta 3000 solicitudes de C y el resto de D.
- La estación de servicio G, genera todas las peticiones de impresión de billetes electrónicos y tiene una capacidad de cola de 13000 transacciones, de las cuales tiene limitado a 4000 las que vienen de E.

Se pide:

a). Representar la red con un grafo al que se le debe añadir un nodo A que represente el nodo fuente y considerar G como único nodo sumidero de la red.



Camino incremental: $\mu_1[t, s] = A - B - E - G \rightarrow \alpha_1 = 4$

i	A	A	A	B	C	C	C	D	E	F	G
i	B	C	D	E	D	E	F	F	G	G	A
f	4			4					4		
c	6	4	1	4	3	1	3	4	4	9	8

b). Calcular el número máximo de transacciones realizables en cada vuelta a la red (unidad de tiempo) de forma que se cumplan las restricciones impuestas en la gestión del SI

4.2.12. Ejercicio 12

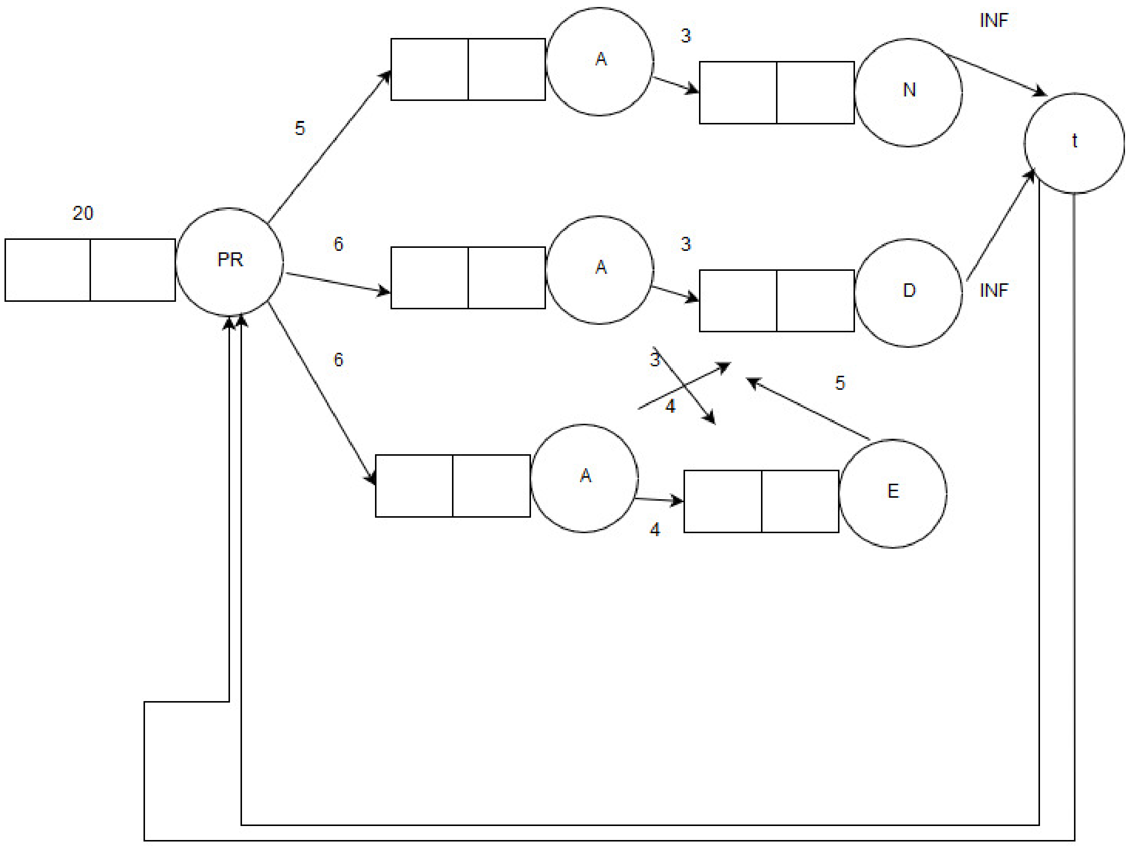
Francisco Gutiérrez (Paco) es el nuevo administrador informático de la oficina de correos de Ciudad Universitaria quiere agilizar el sistema de gestión de y para ello ha realizado un estudio que describe a continuación:

- Existe un “punto de recepción” (PR) general por donde acceden todo tipo de tareas. El propósito de este “punto de recepción es clasificar las tareas recibidas y redirigirlas a uno de los tres procesadores A, B o C en función del propósito de cada tarea: envíos nacionales (A), envíos en Europa (B), resto de envíos (C). La cola del PR tiene una capacidad máxima de 20.000 tareas diarias pero por cuestiones técnicas tiene restringida la admisión de tareas en el sistema a 16.000.
- El procesador A, que realiza el registro de entrada del paquete, tiene una cola limitada a 5000 tareas. Tras ejecutarse la tarea en este procesador, se produce una solicitud emisión e impresión del recibo en la estación de servicio N (Nacional), cuya capacidad de recepción es de 3.000 tareas.
- El procesador B, con funciones similares a A, tiene una cola limitada a 6.000 tareas. Tras el registro del paquete pueden ocurrir dos cosas: que la tarea sea redirigida a la estación de servicio D donde se produce la impresión del recibo o bien que sea redirigida a la estación de servicio E (inspecciones especiales) donde se realiza una toma de datos adicional. La capacidad de la cola de la D desde el procesador B está limitada a 8.000 tareas mientras que la capacidad de la cola de la E desde el procesador B está limitada a 3.000.
- Análogamente a lo ocurrido con el procesador B ocurre con C. Esta vez la capacidad de la cola del procesador C es de 6.000 tareas, de ellas las estaciones D y E tienen restringido a un máximo de 4000 la recepción.
- La estación de servicio E, inspección especial, recoge tareas procedentes del procesador B y tareas procedentes del procesador C. Tras la gestión de E, si no se cumplen ciertos requisitos, la tarea puede ser redirigida a la estación de servicio PR (hasta un máximo de 4.000) o enviada a la estación de servicio D para la impresión del recibo (hasta un máximo de 5.000).

Paco ha sido informado por Genaro (el anterior administrador) de cierta información útil a tener en cuenta antes de la reconfiguración de la red:

- Se sabe que a la red llega una media de 12.000 tareas diarias, de las cuales el 50 % son de tipo B y de las cuales el 50 % requieren ser inspeccionadas.
- El 25 % de las tareas que llegan al sistema son de tipo A.

Se pide dibujar la red y encontrar el flujo máximo (en media diaria) teniendo en cuenta toda la información.
¿Qué alternativas de mejora se pueden plantear teniendo en cuenta estos datos?



i	PR	PR	PR	A	B	B	C	C	N	D	E	E	T
j	A	B	C	N	D	E	D	E	t	t	D	PR	PR
f	3	5	6	3	3	3	4	2	3	12	5	-	15
c	5	6	6	3	8	3	4	4	ínf	ínf	5	4	ínf

$\mu_1 = PR - B - E - D - t; \alpha_1 = 3 \rightarrow \alpha_1 =$ minimo flujo máximo del camino
$\mu_2 = PR - B - D - T; \alpha_2 = 3$
$\mu_3 = PR - A - N - 3; \alpha_3 = 3$
$\mu_4 = PR - C - E - D - T; \alpha_4 = 2$
$\mu_5 = PR - C - D - T; \alpha_5 = 4$

15=flujo máximo

Tema 5

Fiabilidad

5.1. Teoría previa

La **fiabilidad** es la probabilidad de que un sistema siga funcionando después de un tiempo T.

5.1.1. Nomenclatura y variables

- $R(t)$: Fiabilidad.
- $F(t)$: Distribución.
- $f(t)$: Densidad de fallos.
- $\lambda(t)$: Riesgo / tasa de fallos.
- $E(t)$: Vida media.

5.1.2. Fórmulas

$$f(t) = F'(t)$$

$$\lambda(t) = \frac{f(t)}{R(t)}$$

$$R(t) = e^{-\int_0^t \lambda(s)ds}$$

$$R(t) = 1 - F(t)$$

$$E(t) = \int_0^{\infty} t \cdot f(t)dt$$

$$R(t) = P(T > t)$$

Distribución de Weibull

$$T = W(k, \lambda)$$

$$R(t) = e^{-(\frac{t}{\lambda})^k}$$

$$E(t) = \lambda\Gamma(1 + \frac{1}{k})$$

Sistemas en serie

$$R_s(t) = \prod_{i=1}^{k} R_i(t)$$

$$\lambda_s(t) = \prod_{i=1}^{k} \lambda_i(t)$$

Sistemas en paralelo

$$R_p(t) = 1 - \prod_{i=1}^{k} F_i(t)$$

5.1.3. Sistemas k out of n

$$R_{koon}(t) = \sum_{j=k}^{n} \binom{n}{j} R(t)^j (1 - R(t))^{n-j}$$

5.2. Ejercicios resueltos

5.2.1. Ejercicio 1

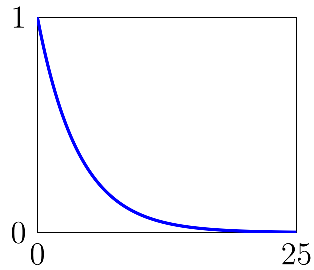
La duración de un disco duro de acuerdo con el fabricante tiene la distribución F donde la duración se mide en años. Un administrador de red quiere comprar esos discos para ser usados en dos proyectos. Para el proyecto A necesita 100 discos y para el proyecto B necesita 1000.

$$F(t) = 1 - e^{-0{,}25t}$$

Se pide:

a). Obtención de la función de fiabilidad de un disco y su representación aproximada.

$R(t) = 1 - F(t) = e^{-t/4} \quad \forall t \geq 0$



b). Esperanza de vida o duración media de un disco.

Sabemos que: $E(t) = \int_0^\infty t \cdot f(t)dt$
Y que $f(t) = F'(t) = -R'(t)$

Entonces: $f(t) = -(-\frac{1}{4} \cdot e^{-t/4}) = \frac{1}{4}e^{-t/4}$

Y por tanto la esperanza de vida es: $E(t) = \int_0^\infty \frac{t}{4}e^{-t/4}dt = 4$

c). El SCRUM Master ha estimado una duración de 3.5 años para el proyecto A y solicita el envío de un lote de 100 discos. ¿Cuál es el porcentaje de discos que aguantarán funcionando todo el proyecto?

Recordemos que $R(t) = P(T > t)$

$P(T > 3{,}5) = R(3{,}5) = e^{-3{,}5/4} = 0{,}4168 = 41{,}68\,\%$
Entonces el $41{,}68\,\%$ de $100 = 41{,}68$

d). Para prevenir problemas, el administrador de la red ha decidido comprar algunos discos más y tenerlos en reserva, pues el fabricante suele tardar unos 3 meses en realizar un nuevo envío. ¿Qué cantidad de discos extra debe comprar para asegurar que se pueda hacer el repuesto sin necesidad de esperar la entrega del fabricante?

$P(3meses > t) = P(\frac{1}{4} > t) = 1 - R(1/4) = F(1/4) = 1 - e^{-(0{,}25^2)} = 0{,}0606 = 6{,}06\,\% \rightarrow 7$ discos.

e). El proyecto B es un alojamiento de datos masivos de una red social. Para configurar el sistema el administrador necesita conocer lo siguiente:

- ¿Qué duración t_1 no será superada por el $1\,\%$ de ellos?
- ¿Qué duración t_1 será superada por el $95\,\%$ de ellos?

- $P(T > t_0) \leq 0{,}99$; $R(t_0) \leq 0{,}99$; $e^{-t_0/4} \leq 0{,}99$; $-t_0/4 \leq ln(0{,}99)$; $-t_0 \leq 4 \cdot ln(0{,}99)$; $t_0 \geq 4 \cdot ln(0{,}99)$; $t_0 \geq 0{,}040$
- $P(T > t_1) \leq 0{,}95$; $R(t_1) \leq 0{,}95$; $e^{-t_1/4} \leq 0{,}95$; $-t_1/4 \leq ln(0{,}95)$; $-t_1 \leq 4 \cdot ln(0{,}95)$; $t_1 \geq 4 \cdot ln(0{,}95)$; $t_1 \geq 0{,}2051$

5.2.2. Ejercicio 2

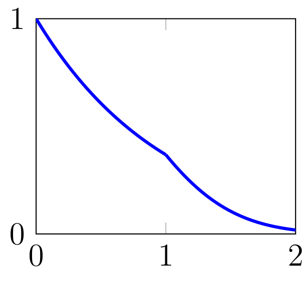
Un sistema informático posee una función de fiabilidad de

$$R(t) = \begin{cases} e^{-t} & 0 < t < 1 \\ e^{-t^2} & 1 \leq t \end{cases}$$

donde la duración está expresada en años. Se pide:

a). Representar la fiabilidad y calcular la función de densidad de fallos.

La función de fiabilidad $R(t)$ se representa:



Y la función de densidad de fallos es:

$$f(t) = -R'(t) = \begin{cases} e^{-t} & 0 < t < 1 \\ 2t \cdot e^{-t^2} & 1 \leq t \end{cases}$$

b). La tasa de fallos. ¿Es constante en cualquier intervalo?

Recordemos que la tasa de fallos $\lambda(t) = \dfrac{f(t)}{R(t)}$, por tanto:

$$\lambda(t) = \begin{cases} 1 & 0 < t < 1 \\ 2t & 1 \leq t \end{cases}$$

Es constante en el intervalo (0,1).

c). Probabilidad de que el sistema falle durante el primer año.

$P(1 < t) = 1 - R(1) = 1 - \dfrac{1}{e} = 0{,}6321$

d). Probabilidad de que el sistema falle entre el segundo y el tercer año.

$P(2 < t < 3) = R(2) - R(3) = e^{-4} - e^{-9} = 0{,}01819$

5.2.3. Ejercicio 3

Sea T = tiempo hasta que se produce el primer intento de intrusión por determinados hackers en un sistema informático. Se conoce la función de riesgo

$$\lambda(t) = \begin{cases} 0 & 0 < t < 1 \\ \dfrac{2}{t} & t \geq 1 \end{cases}$$

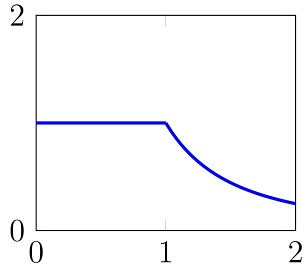
donde el tiempo se define en días. Se pide:

a). Comprobar que se trata de una función de riesgo real calculando y analizando la fiabilidad del modelo.

Sabemos que $R(t) = e^{-\int_0^t \lambda(s)ds}$.

- $0 < t < 1 : R(t) = e^{-\int_0^t 0ds} = e^{-0} = 1$
- $t \geq 1 : R(t) = e^{-\int_0^t \lambda(s)ds} = ... = 1/t^2$

$$R(t) = \begin{cases} 1 & 0 < t < 1 \\ \dfrac{1}{t^2} & t \geq 1 \end{cases}$$



b). Calcular el tiempo disponible para efectuar una transacción electrónica de fundamental seguridad para la empresa, desde que se pone en funcionamiento el SI.

Buscamos un $t_0 : P(T > t_0) = 1$
Queremos el máximo $t_0 : R(t_0) = 1$, por tanto $t_0 = 1$

c). Calcular la vida fiable de la variable asociada a un nivel de fiabilidad de 0.95 ¿Cada cuánto tiempo se recomienda cambiar la contraseña del sistema para garantizar dicha fiabilidad?

$$P(T > t_1) \geq 0{,}95; \quad R(t_1) \geq 0{,}95; \quad \frac{1}{t_1^2} \geq 0{,}95; \quad t_1 \leq 1{,}025;$$

5.2.4. Ejercicio 4

El departamento de I+D de una empresa, ha estimado que la tasa de fallos de un procesador de última generación es

$$\lambda(t) = \begin{cases} \dfrac{1}{1+t} & 0 < t < 1 \\ 0{,}5 & t \geq 1 \end{cases}$$

Sabiendo que el tiempo se mide en cientos de miles de horas, se pide:

a). Calcular la función de distribución y la función de fiabilidad.

Primero calculamos la función de fiabilidad $R(t) = e^{-\int_0^t \lambda(s)ds}$.

- Si $0 < t < 1$: $R(t) = e^{-\int_0^t \frac{1}{1+s}ds} = e^{-ln(1+t)} = \dfrac{1}{1+t}$
- Si $t \geq 1$: $R(t) = \frac{1}{2}e^{\frac{1-t}{2}}$

Por tanto:

$$R(t) = \begin{cases} \dfrac{1}{1+t} & 0 < t < 1 \\ \frac{1}{2}e^{\frac{1-t}{2}} & t \geq 1 \end{cases}$$

Y sabiendo que $F(t) = 1 - R(t)$, calculamos la función de distribución:

$$F(t) = \begin{cases} 1 - \dfrac{1}{1+t} & 0 < t < 1 \\ 1 - (\frac{1}{2}e^{\frac{1-t}{2}}) & t \geq 1 \end{cases}$$

b). Calcular la vida media de dichos procesadores.

La vida media se define como: $E(t) = \int_0^\infty t \cdot f(t)dt$
Por lo que primero necesitamos conocer la densidad de fallos $f(t)$.

$$f(t) = -R'(t) = \begin{cases} \dfrac{1}{(1+t)^2} & 0 < t < 1 \\ \frac{1}{4}e^{\frac{1-t}{2}} & t \geq 1 \end{cases}$$

Y ahora calculamos $E(t)$

$$E(t) = \int_0^\infty t \cdot f(t)dt = \int_0^1 \frac{t}{(1+t)^2}dt + \int_1^\infty \frac{t}{4}e^{\frac{1-t}{2}}dt = 1{,}643$$

5.2.5. Ejercicio 5

Para diseñar un sistema de comunicaciones se dispone de 10 componentes idénticos e independientes. El administrador quiere calcular la esperanza de vida de distintos diseños:

a). Diseño en serie de 10: Suponiendo que las componentes tienen distribución exponencial y una duración media de cada componente de 8000 horas, que es lo que se ha estimado a partir de una muestra anterior.

Si la esperanza de vida media de 1 componente son 8000 horas, la esperanza de vida de 10 componentes en serie será $E(S_{10}) = \dfrac{8000}{10} = 800$ horas.

b). Diseño en serie de 10 tras un estudio más minucioso que revela que la duración de los componentes sigue realmente una distribución de Weibull con $k = 2$ y $\lambda = 10$.

Sabemos que: $T = W(k = 2, \lambda = 10)$ y que
$E(t) = \lambda\Gamma(1 + \frac{1}{k})$
$R(t) = e^{-(\frac{t}{\lambda})^k}$
Recordamos que la función gamma $\Gamma(n) = (n-1)!$

Primero sacamos la fiabilidad y la vida media para un componente:

- $R(t) = e^{-(\frac{t}{\lambda})^k} = e^{-(\frac{t}{10})^2}$
- $E(t) = \lambda\Gamma(1 + \frac{1}{k}) = 10\Gamma(1 + \frac{1}{2}) = 10\frac{\sqrt{\pi}}{2}$

Ahora hacemos los cálculos para todo el sistema, con 10 dispositivos en serie:

- $R_{s10}(t) = \prod_{i=1}^{10} R_i(t) = \prod_{i=1}^{10} e^{-(\frac{t}{10})^2} = e^{-10(\frac{t^2}{100})} = e^{-(\frac{t^2}{10})} = e^{-(\frac{t^2}{\sqrt{10}})^2}$

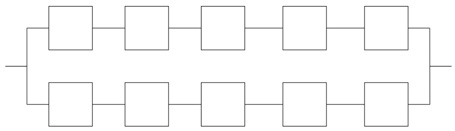
De aquí deducimos que: $W(k = 2, \lambda = \sqrt{10})$

- $E_{s10}(t) = \lambda\Gamma(1 + \frac{1}{k}) = 10\Gamma(1 + \frac{1}{2}) = \sqrt{10} \cdot \frac{\sqrt{\pi}}{2} = 2{,}8025$

c). Diseño en paralelo de dos series de 5 con la distribución anterior. En este caso, en lugar de la esperanza de vida se pide:

a) Calcular la probabilidad de que el sistema siga funcionando al cabo de 500 horas.

b) Calcular la probabilidad de que el sistema siga funcionando al cabo de 5000 horas.



Primero calculamos la fiabilidad para una serie de 5:

$$R_{s5} = \prod_{i=1}^{5} R_i(t) = \prod_{i=1}^{5} e^{-(\frac{t}{10})^2} = e^{-5\cdot\frac{t^2}{100}} = e^{-(\frac{t}{\sqrt{20}})^2}$$

Ahora hallamos la fiabilidad para el sistema en paralelo, usando la fórmula correspondiente:

$$R_{p2s5} = 1 - \prod_{i=1}^{n} F_i(t) = 1 - \prod_{i=1}^{n} 1 - R_i(t) = 1 - (1 - R_{s5}(t))^2 = 1 - (1 - e^{\frac{t^2}{20}})^2$$

Ya podemos calcular las probabilidades del sistema completo:

a) $P(T > 500) = R_{p2s5}(1/2) = 1 - (1 - e^{\frac{-1}{80}})^2 = 0{,}9998$

b) $P(T > 5000) = R_{p2s5}(5) = 1 - (1 - e^{\frac{-25}{20}})^2 = 0{,}4909$

5.2.6. Ejercicio 6

El proceso de arranque de cierto tipo de procesadores invadido por un virus informático presenta una duración hasta fallo (en minutos) modelizada mediante la siguiente distribución:

$$F(t) = \begin{cases} \frac{1}{10}(t^3 + t) & 0 < t < 2 \\ 1 & t \geq 2 \end{cases}$$

Sabiendo que el sistema de arranque normal tarda 180 segundos, se pide:

a). Hallar la función de fiabilidad del proceso de arranque y la función de riesgo asociada.

$$R(t) = 1 - F(t) = \begin{cases} 1 - \frac{1}{10}(t^3 + t) & 0 < t < 2 \\ 0 & t \geq 2 \end{cases}$$

Recordemos: $\lambda(t) = \dfrac{f(t)}{R(t)}$
Sacamos la funcion de densidad $f(t)$:

$$f(t) = F'(t) = \begin{cases} \frac{1}{10}(3t^2 + 1) & 0 < t < 2 \\ 0 & t \geq 2 \end{cases}$$

Y la función de riesgo es:

$$\lambda(t) = \frac{f(t)}{R(t)} = \begin{cases} \dfrac{3t^2 + 1}{10 - t^3 - t} & 0 < t < 2 \\ 0 & t \geq 2 \end{cases}$$

b). Calcular el porcentaje de procesadores que completará el proceso de arranque con éxito.

El proceso de arranque tarda 3 minutos.
$P(T > 3) = R(3) = 0$

c). Para resolver el problema, la organización ha decidido instalar un antivirus que tarda 1 minuto en eliminar el virus atacante. Si la organización posee 2000 equipos, ¿cuántos de ellos se espera que realicen bien el proceso de arranque?

$P(T > 1) = R(1) = \dfrac{4}{5}$; Que son 1600 ordenadores.

Tema 6

Caracterización de la carga

6.1. Introducción

6.1.1. Modelado de la carga

A la hora de evaluar un sistema es muy complicado utilizar la carga real.

- Depende de muchos parámetros
- Varía a lo largo del tiempo
- Interacciona con el sistema informático
- Resulta complicado reproducirla

El modelado de la carga es obtener una descripción cuantitativa para tomar medidas de rendimiento y tomar decisiones sobre el ajuste

6.1.2. Representatividad de la carga

Los modelos de carga son aproximaciones que representan una abstracción del trabajo que se pretende representar.

La representatividad de la carga es una medida de la similitud entre el modelo y la carga real.

Descripciones de carga de trabajo

- **Descripción orientada al negocio (usuario)**: Se describe en términos empresariales como, por ejemplo, el número de empleados o clientes, la facturación, etc. Se conocen como unidades naturales o de predicción natural.
- **Descripción funcional (software)**: Se describen los programas, comandos, peticiones... que constituyen la carga de trabajo.
- **Descripción orientada a los recursos (hardware)**: Se describe el consumo de recursos del sistema durante la carga, por ejemplo, uso de CPU, operaciones de disco, ocupación de memoria, etc.

6.1.3. Componentes y parámetros

- **Componentes básicos de la carga**
    - Unidades genéricas de trabajo que provienen de fuentes externas (entidades que realizan peticiones de servicio al sistema).
    - Dependen de la naturaleza del servicio que provee el sistema.
        - Aplicaciones (e-mail, edición, programación...), peticiones HTTP, sesiones de conexión de usuarios, transacciones...
- **Parámetros de carga**
    - Se usan para modelar o caracterizar la carga.
    - Se deben elegir parámetros dependientes de la carga, no dependientes del sistema.
        - Instrucciones, tamaño de paquetes de red, patrones de referencia a páginas...

6.1.4. Modelos de carga

- Un modelo de carga debe ser representativo, compacto y reproducible.
- Los modelos naturales se construyen usando componentes básicos de la carga real o, al menos, trazas de la ejecución de la carga real.
- Los modelos artificiales o sintéticos no usan componentes básicos de la carga real de trabajo.
    - Los modelos ejecutables, por ejemplo benchmarks, son programas que cargan al sistema con un trabajo similar al que quieren reproducir
    - Los modelos no ejecutables describen una serie de valores paramétricos que reproducen el mismo uso del sistema que la carga real.

6.2. Metodología de caracterización

Usualmente, la caracterización de la carga de un sistema se realiza siguiendo los siguientes pasos:

6.2.1. 1. Elección del objetivo de estudio de carga

6.2.2. 2. Identificación de los componentes básicos de la carga

Los componentes son las unidades de trabajo que dependen de la naturaleza del servicio que provee el sistema.

Cada componente de la carga se caracteriza por

- Parámetros de intensidad de la carga.
- Parámetros de demanda.

6.2.3. 3. Elección de los parámetros característicos de los componentes

6.2.4. 4. Recolección de datos

Hay que recolectar los datos cuando se produce la carga que nos interesa. Se asignan valores a cada componente del modelo de carga.

a). Identificar las ventanas temporales que definen las sesiones de medida.

b). Monitorizar y medir las actividades del sistema durante la ventanas temporales definidas.

c). A partir de los datos recogidos, asignar valores a los parámetros de caracterización de cada componente de la carga.

6.2.5. 5. Fraccionamiento de la carga de trabajo

La carga real puede verse como una colección heterogénea de componentes. Las técnicas de fraccionamiento dividen la carga de trabajo en series de clases de tal forma que sus poblaciones contengan componentes homogéneos.

6.2.6. 6. Cálculo de los parámetros de clase

Hay diversas técnicas para el cálculo de los valores que representan cada clase.

a). **Utilización de medias.** La media aritmética es inadecuada cuando la varianza es alta. En general, si la carga es homogénea se pueden utilizar medias, en cualquier otro caso, puede ser peligroso.

   Para cargas heterogéneas se utiliza el agrupamiento, que determina grupos de cargas similares.

b). **Especificación de la dispersión.**

c). **Histogramas de uno o múltiples parámetros.**

d). **Análisis de componentes principales.**

e). **Modelos markovianos.**

f). **Agrupamiento (clustering)**: Se aplica cuando se dispone de un gran número de componentes. A continuación estudiaremos dos algoritmos utilizados para realizar el agrupamiento.

6.3. Algoritmos de agrupamiento

Los algoritmos de agrupamiento sirven para detectar diferentes clases de carga de trabajo especialmente cuando las cargas tienen varianzas grandes y/o son cargas heterogéneas. Buscamos agrupar las componentes en grupos, de manera que las componentes que forman un grupo sean similares entre sí, pero distintas con respecto al resto de grupos.

Supongamos un ejemplo en el que se han realizado unas mediciones sobre 5 programas, en los que hemos medido el tiempo que han utilizado para entrada-salida, y el tiempo de CPU.

Queremos agrupar las cargas en 2 grupos, para ello vamos a estudiar los siguientes algoritmos.

6.3.1. Árbol de extensión mínima (MST)

El algoritmo MST agrupa en cada etapa los datos en clusters hasta obtener el número deseado.

1) Empezar con k=n clases.

2) Para toda clase i, encontrar el centroide de la clase.

3) Para toda clase i y j, calcular la matriz de distancia de pares (i, j) entre centroides de esas clases.

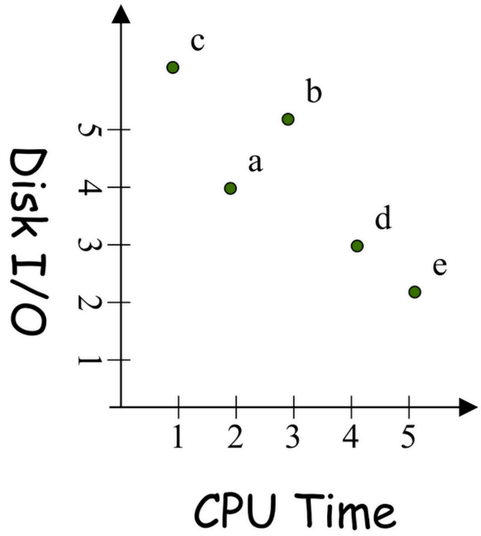
4) Encontrar la distancia mínima en la matriz y fusionar las clases entre las cuales esa distancia es mínima.

5) Repetir 2 a 4 hasta que todos los componentes pertenezcan a la misma clase.

1.- Empezamos con 5 clases.

Iteración 1

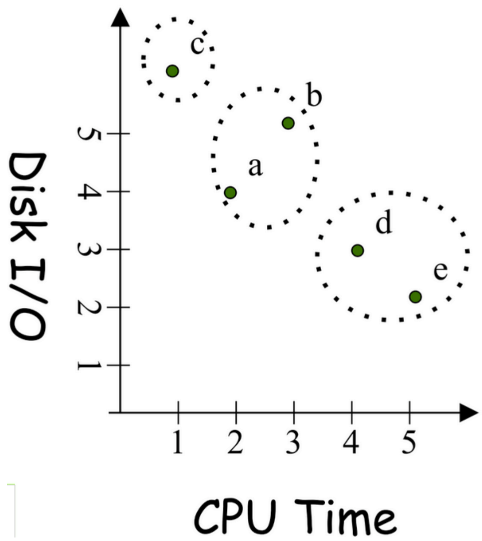
2.- En este caso los centroides son los propios puntos, puesto que son el centro del grupo (de sí mismos).



3.- Calculamos la distancia entre las clases.

	Program				
Program	A	B	C	D	E
A	0	$\boxed{\sqrt{2}}$	$\sqrt{5}$	$\sqrt{5}$	$\sqrt{13}$
B		0	$\sqrt{5}$	$\sqrt{5}$	$\sqrt{13}$
C			0	$\sqrt{18}$	$\sqrt{32}$
D				0	$\boxed{\sqrt{2}}$
E					0

4.- Obtenemos la distancia mínima y agrupamos en clusters. En este caso la distancia de A-B es igual a la distancia E-D ($\sqrt{2}$), que es la mínima, por lo que las agrupamos. C se queda suelto.

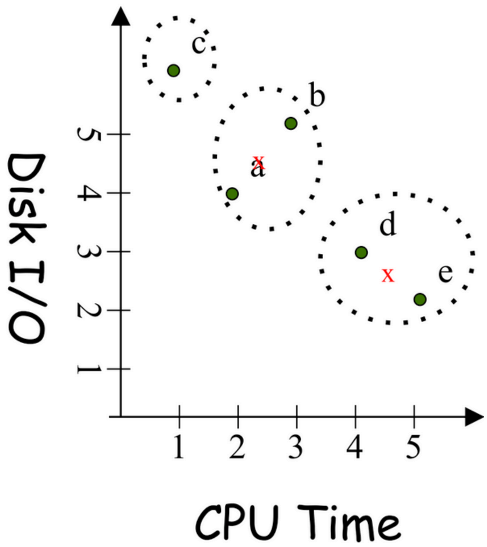


Si quisiéramos hacer 3 grupos ya habríamos acabado. Como queremos hacer 2 grupos seguimos con el algoritmo.

Iteración 2

2.- Calculamos los centroides de los grupos.

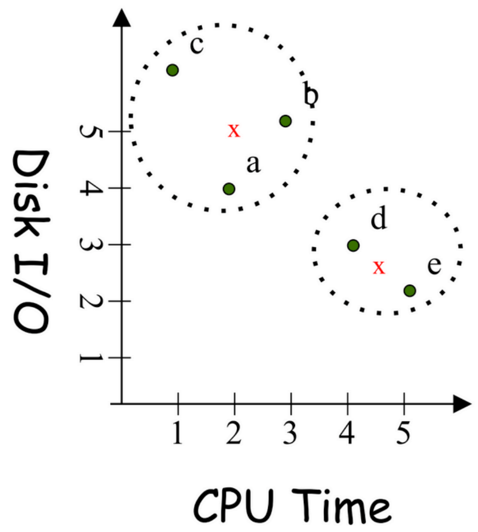
- El centroide de AB es (2+3)/2, (4+5)/2 = 2.5, 4.5
- El de DE es 4.5, 2.5



3.- Calculamos las distancias entre centroides.

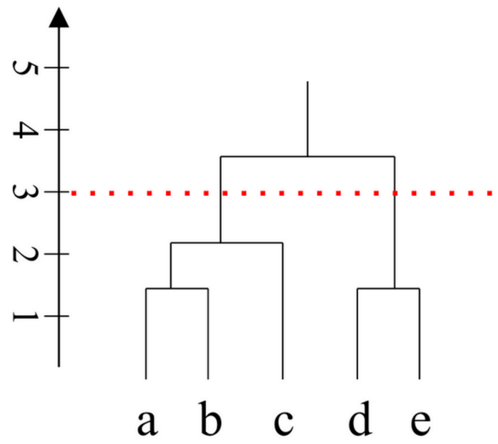
	Program		
Program	AB	C	DE
AB	0	$\boxed{\sqrt{4.5}}$	$\sqrt{10.25}$
C		0	$\sqrt{24.4}$
DE			0

4.- Observamos que la menor distancia es la del grupo **AB** con **C**, así que agrupamos.



Ya tenemos los 2 grupos formados así que podemos parar aquí.

Este sería el árbol de extensión media resultante.

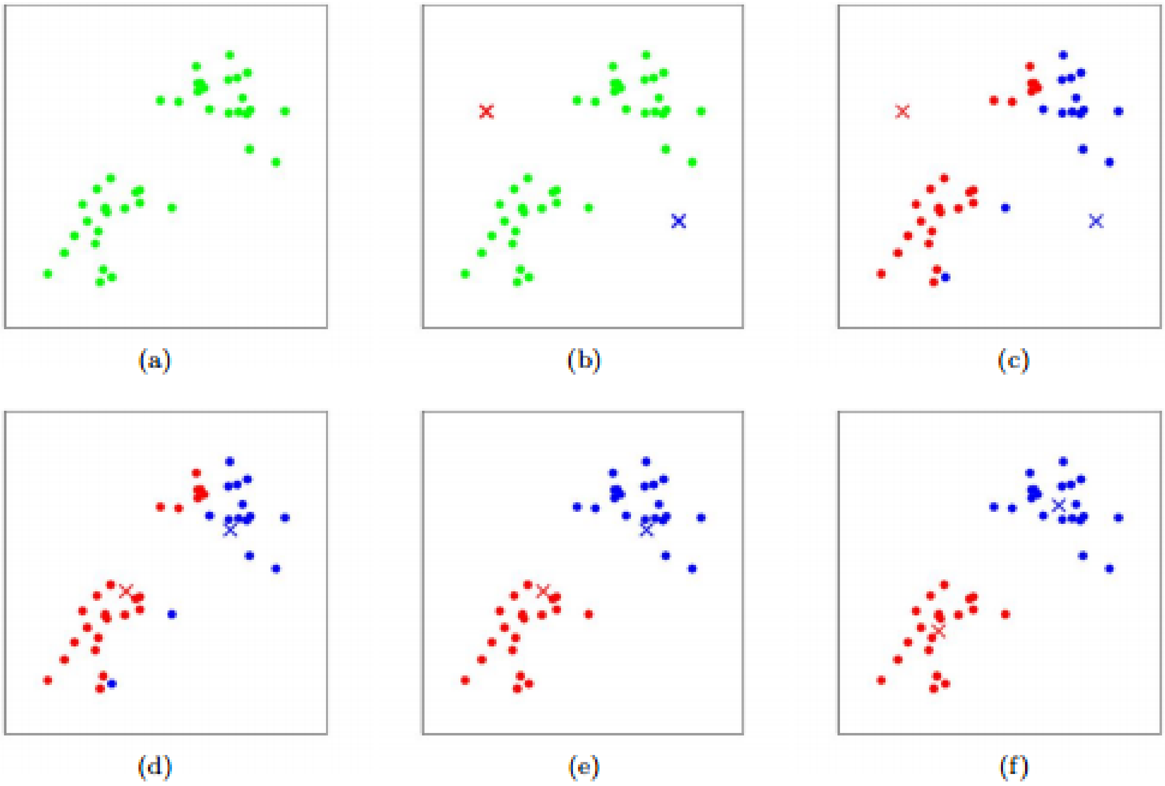


6.3.2. K-Mean

La desventaja del algoritmo K-means es que hay que conocer de antemano el número de cargas diferentes de trabajo que hay en el sistema.

Video https://www.youtube.com/watch?v=w_aUCJHRv0Y

1) Escogemos aleatoriamente los centroides.
2) Asignamos cada componente al grupo o centroide más cercano.
3) Calculamos los centroides de los grupos.
4) Repetir 2 y 3 hasta que dejen de existir variaciones entre iteraciones (hasta llegar a un resultado estable).



PASCAL
MAYO 2018
Ult. actualización 28 de mayo de 2019
LaTeX lic. LPPL & powered by TEFLON CC-ZERO