

# Control de los efectos del retraso en sistemas de tiempo real planificados con EDF\*

José L. Lorente, José Carlos Palencia  
Grupo de Computadores y Tiempo Real  
Universidad de Cantabria  
39006 Santander  
[lorentejl@unican.es](mailto:lorentejl@unican.es), [palencij@unican.es](mailto:palencij@unican.es)

## Resumen

La planificación dinámica en los sistemas de tiempo real está en la actualidad en creciente desarrollo, tanto desde un punto de vista teórico como práctico. La causa de este creciente auge es el mayor rendimiento de los sistemas con planificación dinámica frente a los sistemas con planificación estática. En sistemas sencillos con tareas periódicas ha habido ya muchos estudios con resultados satisfactorios; sin embargo, en sistemas más complejos, como por ejemplo sistemas distribuidos, la planificación dinámica presenta todavía algunas carencias. En particular, la activación retrasada de tareas periódicas conlleva la pérdida de planificabilidad del sistema, tanto de las tareas que sufren retraso como del resto de tareas. Surge así la necesidad de crear nuevos métodos de planificación o de adaptar a prioridades dinámicas algunos desarrollados previamente para planificación estática. En este artículo veremos un nuevo algoritmo de planificación, basado en el servidor de reserva de ancho de banda CBS, que hace que disminuya la interferencia de la tarea planificada con este servidor sobre el resto de tareas, pudiendo hacer planificables sistemas que antes no lo eran.

## 1. Introducción

El algoritmo de planificación basado en prioridades dinámicas EDF (Earliest Deadline First) fue definido en 1973 por Liu y Layland [4] y es óptimo para sistemas monoprocesadores con tareas expulsables y periódicas con plazos iguales a los periodos [4] [2]. Así, muchos sistemas no planificables bajo prioridades fijas sí lo son con prioridades dinámicas.

(\*) Trabajo financiado por la Comisión Interministerial de Ciencia y Tecnología del gobierno español, proyecto TIC2002-04123-C03-02

Para estos sistemas, podemos garantizar el cumplimiento de los plazos si la utilización del sistema es igual o inferior a la unidad:  $\sum(C_i/T_i) \leq 1$ .

Sin embargo, para sistemas más complejos no existe un test de planificabilidad exacto y, por ello, se han desarrollado técnicas pesimistas de análisis de tiempos de respuesta de peor caso [10][7].

Si bien los sistemas planificados mediante EDF tiene mayor rendimiento que los planificados con prioridades fijas (RM) [4], en muchos casos, sistemas con utilización alta de procesador y activaciones retrasadas de tareas, no son planificables, perdiendo el plazo alguna de las tareas. La activación retrasada de tareas periódicas no sólo influye en el tiempo de respuesta de peor caso de la propia tarea, sino que también aumenta la interferencia sobre el resto de tareas del sistema (debido a que dos activaciones consecutivas pueden ocurrir en un intervalo menor que el periodo de activación). En estos sistemas, aunque la activación de una tarea es periódica, puede sufrir un retraso en cada una de sus activaciones, debido a que puede transcurrir un cierto tiempo entre el instante teórico en que se debería producir el evento periódico y el instante en el que se produce realmente. Si bien este retraso puede ser pequeño en sistemas monoprocesadores, su estudio es útil para el análisis de sistemas multiprocesadores y distribuidos, donde el retraso es debido fundamentalmente a que la activación de una tarea depende de la finalización de la tarea previa en la cadena de respuesta o de la llegada de un mensaje de otro procesador [11]. Dado que este retraso puede ser muy grande, los efectos sobre la planificabilidad del sistema pueden ser drásticos, por lo que esta activación retrasada es uno de los principales problemas en los sistemas multiprocesadores y distribuidos [6][11].

En este artículo vamos a ver un método para reducir la interferencia que una tarea con retraso produce sobre el resto de tareas, analizando tam-

bién el impacto sobre la propia planificabilidad de la tarea controlada. Este algoritmo está basado en los servidores de reserva de ancho de banda. Aunque, en principio, estos servidores fueron pensados para la ejecución de tareas aperiódicas, es posible, con alguna modificación, utilizarlos para el control de las tareas periódicas con activación retrasada. Los servidores de reserva de ancho de banda para planificación de tareas no periódicas en sistemas EDF son de dos tipos: unos adaptados directamente de RM y otros desarrollados para EDF sin equivalencia en prioridades fijas. Los servidores DPE (Dynamic Priority Exchange) [3], DSS (Dynamic Sporadic Server) [8][9] son ejemplos de servidores adaptados de RM. Algunos de los nuevos servidores para EDF son el TBS (Total Bandwidth Server) [8][9] y el algoritmo CBS (Constant Bandwidth Server) [1], que es el que pensamos que, con alguna modificación, mejor se adapta a nuestros propósitos. Recientemente se ha propuesto un nuevo servidor, IRIS (Idle-Time Reclaiming Improved Server) [5] similar al CBS, por lo que pensamos que también puede servir para el control de la activación retrasada de tareas, aunque todavía debe ser evaluado.

Este artículo está estructurado en las siguientes secciones: en la sección 2 introducimos el modelo computacional, en la sección 3 indicamos las reglas que definen el comportamiento del servidor CBS. A continuación, en la sección 4 se muestran los problemas del servidor CBS para el control de la activación retrasada de tareas, por lo que, en la sección 5, presentaremos una modificación del CBS (al que denominaremos CBS Modificado, CBSM). En la sección 6, calcularemos los tiempos de respuesta de peor caso de sistemas planificados con estos servidores y, por último, daremos las conclusiones y el trabajo futuro.

## 2. Modelo computacional

El modelo computacional que utilizaremos en este artículo es el de un sistema compuesto por un conjunto de tareas ejecutando en un único procesador (aunque el comportamiento sería similar en un sistema distribuido). Cada tarea  $\tau_i$  se activa a intervalos regulares con un periodo de activación  $T_i$ , un tiempo de ejecución de peor caso  $C_i$  y un plazo máximo de ejecución (deadline)  $D_i$ , medido respecto a la activación de la tarea. El retraso

máximo (o jitter) en la activación de la tarea  $\tau_i$  viene denotado por  $J_i$ . Denotaremos como  $\tau_{ik}$  a la  $k$ -ésima activación de  $\tau_{ik}$ , cuyo tiempo de activación ocurre en  $f_{ik}$ , su liberación ocurre en  $g_{ik} \in [f_{ik}, f_{ij} + J_i]$  su plazo absoluto en  $d_{ik} = f_{ik} + D_i$  y su tiempo de respuesta en  $r_{ik}$ . El sistema será planificable si  $\forall i, k \ r_{ik} \leq d_{ik}$ . Es decir si  $\forall i \ R_i = \max\{r_{ik} - f_{ik}\} \leq D_i$ .

La planificación de las tareas es dinámica y está basada en la asignación de prioridades según los plazos de ejecución, de forma que ejecuta antes aquella tarea activa que tenga su plazo más cercano (EDF, Earliest Deadline First). Además, cada tarea puede experimentar un bloqueo debido a recursos compartidos con otras tareas del sistema, siendo este bloqueo máximo  $B_i$ .

Adicionalmente, alguna de las tareas periódicas puede estar controlada por un servidor CBSM, que definiremos en los siguientes apartados.

## 3. Definición del servidor CBS

El servidor CBS [2] es un planificador de reserva de ancho de banda utilizado para tareas aperiódicas en sistemas con planificación dinámica (EDF). Las reglas que rigen su comportamiento y su notación son las siguientes:

- 1) El planificador CBS se caracteriza por una capacidad  $C_s$ , cuyo valor máximo es  $Q_s$ , y por un periodo  $T_s$ . La razón  $U_s = Q_s/T_s$  se define como la utilización del servidor.
- 2) En cada instante de tiempo el servidor tiene un plazo asociado  $d_s^k$ , que irá variando según el estado del sistema, y donde el superíndice  $k$  indica el número de veces que se ha revaluado. La posición del primer plazo es  $d_s^0 = 0$ .
- 3) En cada instante, la tarea servida por el CBS tiene asignada de forma dinámica un plazo igual al del servidor. La posición de éste respecto al del resto de tareas determina su prioridad en el sistema.
- 4) Cuando una tarea es ejecutada bajo el servidor, la capacidad del servidor  $C_s$  disminuye en la misma cantidad que el tiempo que ejecute la tarea.
- 5) Cuando  $C_s = 0$ , la capacidad del servidor es recargada al máximo valor  $Q_s$  y el plazo se reevalúa al valor  $d_s^{k+1} = d_s^k + T_s$ .
- 6) El servidor se dice activo en un instante  $t$  si la tarea servida por él se encuentra ejecutan-

do o pendiente de ejecutar. En caso contrario se dice que el servidor está libre.

- 7) Cuando una acción llega al servidor se encola según una teoría predicativa (FIFO)
- 8) Estando el servidor libre y con un plazo  $d_s^k$ , si una activación ocurre en el instante  $r_i$  cumpliéndose  $C_s \geq (d_s^k - r_i) \cdot U_s$ , entonces la capacidad del servidor se recarga al valor máximo ( $C_s = Q_s$ ), y el plazo se modifica al valor  $d_s^{k+1} = d_s^k + T_s$ . En cualquier otro caso la activación se planifica con los anteriores plazo y capacidad del servidor.

#### 4. Limitaciones del servidor CBS en el control del retraso.

Sea un sistema de tiempo real compuesto por  $n$  tareas periódicas ( $\tau_1, \tau_2, \dots, \tau_n$ ). Queremos disminuir mediante el servidor CBS el efecto que el retraso de una de estas tareas,  $\tau_i$ , produce sobre las restantes tareas. Para este fin controlamos la ejecución de la tarea con un servidor CBS. Cuando  $\tau_i$  es servida con un servidor CBS la prioridad de la tarea en el sistema viene determinada por el plazo del servidor, y no por su plazo. La tarea será planificable si ejecuta antes de su plazo, con independencia de la posición del plazo del servidor.

Si queremos servir la tarea  $\tau_i$  con un CBS, tendremos que ajustar los parámetros del servidor acorde con los de la tarea. Surgen así dos posibilidades:

- 1) El periodo del servidor igual al periodo de la tarea ( $T_s = T_i$ ) y la capacidad igual a la ejecución ( $Q_s = C_i$ ).
- 2) El periodo del servidor igual al plazo de la tarea ( $T_s = D_i$ ) y la capacidad igual a la ejecución ( $Q_s = C_i$ ).

Veamos como, en cualquiera de los dos casos, si el periodo de la tarea es menor que su plazo ( $T_i < D_i$ ), el servidor no nos proporciona los resultados deseados. Esto nos planteará la necesidad de modificar el servidor para que sea aplicable a cualquier tarea con independencia de la relación entre su periodo y su plazo.

Nota: en los diagramas de ejecución que aparecerán a continuación diferenciaremos entre el instante de activación de la tarea y el de liberación. El primero ocurre cuando llega el evento periódico, mientras que el instante de liberación es cuando empieza a estar dispuesto a ejecutar (des-

pués de sufrir su posible retraso). Identificaremos con flechas discontinuas hacia arriba las activaciones ( $\uparrow$ ) y continuas ( $\uparrow$ ) los instantes de liberación. Cuando coinciden activación y liberación (no hay retraso) sólo se representará la liberación. Diferenciaremos los plazos generados por el servidor mediante flechas discontinuas hacia abajo ( $\downarrow$ ) y los plazos de ejecución de las tareas mediante flechas continuas ( $\downarrow$ ).

#### 4.1 Tareas con plazos menores que el periodo ( $D_i < T_i$ )

##### 4.1.1. Tarea $\tau_i$ servida CBS con periodo $T_s = T_i$ .

Supongamos un sistema sencillo con dos tareas periódicas con los siguientes parámetros:

Tarea	T	D	J	C
$\tau_1 = \tau_1$	8	6	2	2
$\tau_2$	5	4	1	2

Ejecutaremos la primera tarea con y sin servidor CBS de parámetros  $T_s = T_1 = 8$  y  $Q_s = C_1 = 2$ . Comprobaremos que si  $\tau_1$  se ejecuta con el servidor se reduce su interferencia sobre  $\tau_2$ .

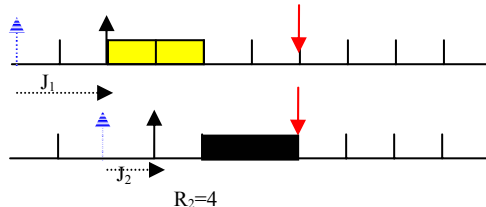


Figura 1. Tiempo de respuesta de  $\tau_2$  cuando  $\tau_1$  ejecuta sin servidor

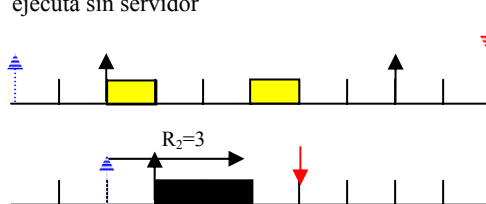


Figura 2. Tiempo de respuesta de  $\tau_2$  cuando  $\tau_1$  ejecuta con CBS ( $T_s = 8, Q_s = 2$ )

Si planificamos la tarea periódica con  $T_s=T_i$ , la tarea  $\tau_i$  se comporta para el resto de tareas como otra tarea periódica con los mismos parámetros  $T_i$ ,  $C_i$ ,  $J_i$ , pero plazo  $D_i'=D_i+(T_i-D_i)+J_i=T_i+J_i$ , que es mayor que su plazo ( $D_i'>D_i$ ) y por tanto puede disminuir la interferencia sobre las demás tareas periódicas.

4. 1.2. Tarea  $\tau_i$  servida con un servidor CBS con periodo  $T_s=D_i$

Vamos a ver como se comporta una tarea  $\tau_i$  con parámetros  $T_i=6$ ,  $D_i=4$  y  $J_i=3$  cuando ejecuta con y sin servidor CBS de periodo  $T_s=D_i=4$ . El estudio de esta situación es más complejo que el caso anterior, pues al planificar con el plazo, las activaciones y los plazos tendrán diferente periodicidad.



Figura 3. Patrón de activaciones tarea sin servidor

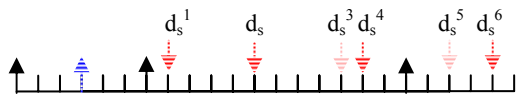


Figura 4. Patrón de activaciones tarea planificada con CBS ( $T_s=D_i=4$ )

La interferencia de peor caso sobre otra tarea  $\tau_j$  ocurre cuando las liberaciones de  $\tau_i$  se producen los más próximas posible entre sí. Esto sucede cuando la primera liberación ocurre con máximo retraso, y las siguientes sin retraso. Cuando planificamos con el servidor CBS, el primer plazo generado por el servidor  $d_s^1$  se sitúa en el instante  $T_s=D_i$ . A partir de él, los plazos ocurrirán de forma periódica cada  $T_s=D_i$  hasta que, debido a que las activaciones ocurren con mayor periodo que los plazos, llega un momento en que el plazo  $d_s^3$  cumple la condición  $(d_s^3-r_3)<T_s$ . En este instante por la regla 8 del CBS ( $C_s=C_i>(d_s-r_s)\cdot U_s$ ) se modifica el plazo situándose en  $d_s^4=r+T_s$ . A partir de entonces por la misma razón, los plazos  $d_s^5, d_s^6$ , etc son modificados. Así los plazos de planificación de la tarea serán  $d_s^1, d_s^2, d_s^4, d_s^6$ , etc.

Si comparamos la posición de los plazos de la tarea con y sin servidor, se cumple que el primer plazo se retrasa  $J_i$  unidades de tiempo. Si se cumple que  $J_i-(D_i-T_i)>0$  el segundo plazo se retrasa este valor, así sucesivamente hasta la activación

$n$ -ésima que cumpla  $J_i-n(T_i-D_i)>0$ . A partir de esa activación, los sucesivos plazos ocurren en la misma posición que cuando la tarea no era servida con un CBS. Al igual que cuando  $T_s=T_i$  la interferencia de la tarea  $\tau_i$  sobre el resto de tareas puede disminuir, al alejar los plazos (aunque en este caso no todos ellos).

4.2 Tareas con plazos mayores que el periodo ( $D_i>T_i$ )

4.2.1 Tarea  $\tau_i$  servida CBS con periodo  $T_s=T_i$ :

Veamos el comportamiento con un ejemplo sencillo de dos tareas periódicas planificadas en EDF con los siguientes parámetros:

Tarea	T	D	J	C
$\tau_1=\tau_i$	3	6	2	2
$\tau_2$	5	4	1	2

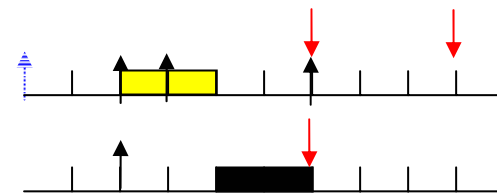


Figura 5. Tiempo de respuesta de  $\tau_2$  cuando  $\tau_1$  ejecuta sin servidor

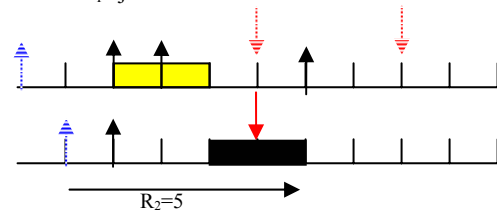


Figura 6. Tiempo de respuesta de  $\tau_2$  cuando  $\tau_1$  ejecuta con CBS ( $T_s=3, Q_s=2$ )

En este caso, la tarea planificada con servidor CBS se comporta como una tarea periódica con los mismos parámetros  $C_i$ ,  $T_i$ ,  $J_i$ , excepto  $D_i'=D_i+J_i-(D_i-T_i)$ . Dependiendo de los valores de  $J_i$ ,  $T_i$  y  $D_i$ ,  $D_i'$  puede ser mayor o menor que el plazo original. Para disminuir la interferencia, nos interesa que  $D_i'>D_i$  y por tanto, que  $J_i-(D_i-T_i)>0$ . En nuestro ejemplo,  $J_i-(D_i-T_i)=-1<0$  y por tanto disminuimos el plazo, obteniendo el efecto no deseado (de hecho, como se ve en el ejemplo,

aumentamos en una unidad el tiempo de respuesta de  $\tau_2$ ).

4.2.2 Tarea  $\tau_i$  servida con CBS con periodo  $T_s=D_i$ :

Al igual que en el apartado 4.1.2 el comportamiento es más complejo que en el caso anterior, pues las activaciones y los plazos tendrán diferente periodicidad. Sea una tarea  $\tau_i$  con parámetros  $T_i=4$ ,  $D_i=6$ ,  $J_i=1$ , veamos su comportamiento si se ejecuta con el servidor CBS.



Figura 7. Patrón de activaciones tarea sin servidor

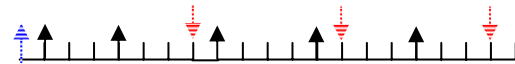


Figura 8. Patrón de activaciones tarea con CBS.

Encontramos que el plazo en cada activación se aleja una distancia  $(D_i-T_i)$  respecto a la activación anterior. La primera activación tiene un plazo  $d_1=D_i+J_i$  y las sucesivas cumplen  $(d_n-f_{in})=D_i+J_i+n(D_i-T_i)$ . Este alejamiento del plazo no tiene límite y puede producir un tiempo de respuesta para la tarea planificada demasiado grande (al tener plazos muy alejados de su activación).

Como conclusión de estos ejemplos, podemos considerar que cuando el periodo de la tarea es menor que el plazo ( $D_i>T_i$ ) el servidor CBS original no proporciona los resultados deseados, tanto al planificar con el periodo como con el plazo. Necesitamos modificar el CBS para que produzca los resultados deseados.

5. El servidor CBS Modificado para el control del retraso.

Como hemos visto, mediante el CBS no conseguimos los resultados esperados para el control del retraso. Surge así la necesidad de modificar el servidor acorde nuestros intereses. El nuevo servidor se denominará CBSM (CBS Modificado) y se diferencia del servidor CBS en los siguientes aspectos:

- 1) El servidor CBSM viene definido por tres parámetros en vez de dos, son el periodo  $T_s$ , el plazo  $D_s$ , y la capacidad máxima  $Q_s$ .
- 2) La utilización del servidor es  $U_s=Q_s/D_s$ .
- 3) Cuando se agota la capacidad de ejecución del servidor se rellena la capacidad del servidor al valor máximo,  $C_s=Q_s$  y se modifica el plazo al instante  $d_s^{i+1}=d_s^i+T_s$ .
- 4) Cuando se activa una tarea, estando el servidor en estado no activo y se cumple la condición  $U_s(d_i-r_i) \leq C_s$ , entonces se rellena la capacidad a  $Q_s$ , y se reevalúa el plazo  $d_s^i=r_i+D_s$

Se cumple que si  $T_s=D_s$ , la planificación con CBSM y CBS produce los mismos resultados, ya que las nuevas reglas no modifican las originales del servidor CBS.

Para estudiar el comportamiento del nuevo servidor CBSM introduciremos el siguiente teorema:

*Teorema 1:* Sea  $\tau_i$  una tarea periódica planificada con un servidor CBSM con parámetros  $Q_s=C_i$ ,  $T_s=T_i$  y  $D_s=D_i$ . Si la tarea ejecuta  $C_i$  en cada  $k$ -ésima activación, el plazo del servidor se sitúa en el intervalo  $d_s \in [f_{ik}+D_i, f_{ik}+D_i+J_i]$  siendo  $f_{ik}$  el instante de activación de  $\tau_i$ . El valor  $d_s = f_{ik} + D_i$  ocurrirá cuando la activación  $k$ -ésima y todas sus anteriores ocurran sin retraso y el valor  $d_s = f_{ik} + D_i + J_i$  cuando esta activación o alguna de sus anteriores se libere tras sufrir el retraso máximo,  $J_i$ .

*Demostración:*

Sea  $k_1$  el retraso de la primera activación, esto es  $g_{i1}=0+k_1$ . Por definición del CBSM  $d_s^0=0$  de manera que debido a la tercera propiedad se reevalúa el plazo del servidor pasando a ser  $d_{i1}=d_s^1=k_1+D_s$ . El valor del plazo respecto a la activación en  $f_{i1}=0$  es  $d_s^1-f_{i1}=D_s+k_1$ . Dado que la capacidad inicial es  $C_i$ , cuando se ejecuta el tiempo de ejecución de peor caso se agota la capacidad reevaluándose de nuevo el plazo  $d_s^2=D_s+k_1+T_s$  y rellenándose la capacidad.

La segunda activación ocurre en  $f_{i2}=T_i$ . Cuando se libera la siguiente activación puede ocurrir:

- 1) Se libera con retraso  $k_2$  menor a  $k_1$ , entonces  $g_{i2}=T_s+k_2 < T_s+k_1$ , así  $(d_s^2-g_{i2}) > D_s=U_s \cdot C_s$ , y el plazo no se modifica con lo que la segunda activación tiene un plazo  $d_{i2}=d_s^2=D_s+k_1+T_s$ . Respecto a la activación  $f_{i2}$  situado a una distancia  $(d_s^2-f_{i2})=(D_s+k_1+T_s)-T_s=D_s+k_1$ .
- 2) Se libera con retraso mayor o igual a  $k_1$ , entonces  $g_{i2}=T_s+k_2 \geq T_s+k_1$  de manera que  $(d_s^2-$

$g_{i2} \leq D_s = U_s \cdot C$ , por lo que el plazo se reevalúa y la segunda activación tiene un plazo  $d_{i2} = d_s^3 = g_{i2} + D_s = D_s + T_s + k_2$ , con lo que el plazo relativo a la activación es  $(d_{i2} - f_{i2}) = (D_s + T_s + k_2) - T_s = D_s + k_2$

Podemos englobar los dos resultados en una sola expresión:  $(d_{i2} - f_{i2}) = D_s + \max(k_1, k_2)$ .

De nuevo, en la situación de peor caso se ejecuta un tiempo  $C_i$  que hace que se agote la capacidad y que el nuevo plazo sea desplace  $T_s$  respecto al anterior. Para la tercera activación  $\tau_{i3}$  tendremos, por tanto, la misma situación que en  $\tau_{i2}$

Repetiendo este proceso para el resto de activaciones, tenemos que para la  $n$ -ésima activación la distancia entre su plazo respecto y su activación será:  $(d_{in} - f_{in}) = D_s + \max(k_1, k_2, \dots, k_n)$

A partir de la expresión anterior el mayor valor del plazo ocurre cuando  $\max(k_1, \dots, k_n) = J_i$ , es decir cuando alguna de las activaciones anteriores o ella misma ha sufrido un retraso  $J_i$ .

Por otro lado, el plazo más próximo a la activación se produce cuando  $\max(k_1, \dots, k_n) = 0$ , esto ocurre cuando ninguna activación anterior a ella ni ella misma han sufrido retraso alguno, ( $f_{ik} = g_{ik}$ )

□

Un concepto importante a la hora de determinar el tiempo de respuesta de peor caso de una tarea es el de instante crítico. Éste ocurre cuando la tarea bajo análisis se activa de forma que su tiempo de respuesta es máximo. Para determinar la situación que conduce a ese instante crítico, introduciremos los siguientes teoremas:

**Teorema 2:** La mayor interferencia que sufre una tarea  $\tau_i$  debido a otra tarea periódica  $\tau_j$  ocurre cuando se dan las dos condiciones siguientes:

- Cuando, conocido el plazo de la tarea  $\tau_j$  aumentamos el número de activaciones de  $\tau_j$  con plazos entre  $t=0$  (comienzo periodo de ocupación) y el instante en el que está situado el plazo de la tarea  $\tau_i$ .
- Las liberaciones de la tarea  $\tau_j$  con plazo menor que el de la tarea de estudio estén lo más próximas posibles al comienzo del periodo de ocupación ( $t=0$ ).

**Demostración:**

- Por definición de EDF el número de activaciones que pueden interferir a una tarea  $\tau_i$  son las que poseen un plazo menor al de esta tarea.

- Cuanto antes ocurran las liberaciones de la tarea  $\tau_j$ , más posibilidades tienen de interferir a  $\tau_i$ , ya que si las alejamos puede que alguna activación no interfiera a la ejecución de  $\tau_i$ , al liberarse después de finalizar la ejecución de ésta.

□

**Teorema 3:** La peor contribución de una tarea  $\tau_i$  planificada con CBSM a la ejecución de otra tarea  $\tau_a$ , ocurre cuando la primera activación de  $\tau_i$ ,  $\tau_{i0}$ , se libera en el comienzo del periodo de ocupación ( $g_{i1} = t=0$ ) después de sufrir el máximo retraso. El resto de activaciones se producen de forma que estén lo más próximas al comienzo del periodo de ocupación

**Demostración:**

Veamos por separado las distintas condiciones del teorema:

- $\tau_i$  se libera en  $t=0$ . Supongamos que se libera:
  - En  $t'$  respecto el inicio del periodo de ocupación ( $t=0$ ). Si acercamos  $t'$  a  $t=0$  conseguimos aproximar los plazos y las activaciones al comienzo del periodo de ocupación. Por el teorema anterior, produciría mayor interferencia.
  - Se libera antes del periodo de ocupación; en ese caso el periodo de ocupación comenzará antes.
- Se libera con máximo retraso,  $J_i$ : si se libera con retraso  $k$ ,  $f_{i0} = -k$ , la posición del plazo respecto a su activación puede oscilar entre  $(d_{i0} - f_{i0}) \in [D_i + k, D_i + J]$  dependiendo de las activaciones anteriores (teorema 1). La mayor interferencia producida por  $\tau_i$  al resto de tareas ocurrirá cuando más cerca está su deadline de su liberación en  $t=0$  (comienzo del periodo de ocupación), esto ocurre cuando  $(d_{i0} - f_{i0}) = D_i + k$ , y por tanto  $d_{i0} = D_i$  independiente del retraso  $k$ . Esta situación ocurre cuando los retrasos anteriores iguales o menores al de la primera activación (teorema 1). Si la primera activación sufre máximo retraso  $J_i$ , nos aseguramos que sus activaciones anteriores no sufren mayor retraso y así  $d_{i0} = D_i$

Por otro lado siempre que las activaciones posteriores se produzcan con menor o igual retraso que sus anteriores el patrón de plazos es una secuencia periódica de periodo  $T_i$  respecto a  $t=0$ . Si alguna activación se produce con mayor retraso que todas las anteriores plazo se aleja respecto al patrón periódico, de

forma que, conforme al teorema 2, no genera situación de peor caso. Así la, mayor interferencia producida por  $\tau_j$  servida con el servidor CBS ocurre cuando el retraso de la primera activación es mayor o igual que el de las siguientes (esto ocurrirá siempre si el primer retraso es  $J_i$ ).

Por otro lado, cuanto mayor sea el valor del retraso inicial más próximas ocurrirán las liberaciones respecto al comienzo del periodo de ocupación, y mayor interferencia puede producir a  $\tau_a$  por el teorema 2.

- 3) Las activaciones  $\tau_{i-p}$  que se activan en  $f_{i-p} \in [f_{i0} - J_i, 0)$  se liberan en  $t=0$ : Si se liberaran antes parte de su ejecución se puede producir antes de  $t=0$  disminuyendo así la interferencia producida sobre  $\tau_a$ . Si se liberaran después de  $t=0$ , entonces se alejan del comienzo del periodo de ocupación y pueden causar menor interferencia (teorema 2).
- 4) Las activaciones  $\tau_{ip}$  que se activan después del periodo de ocupación lo hacen sin retraso ( $f_{ip} = g_{ip}$ ). Si se retrasaran alejarían las relajaciones del comienzo del periodo de ocupación, y por el teorema 2 pueden disminuir su interferencia

□

*Ejemplo:* Sea una tarea  $\tau$  planificada por servidor CBSM con  $T_i=5, D_i=6$  y  $J_i=6$ :

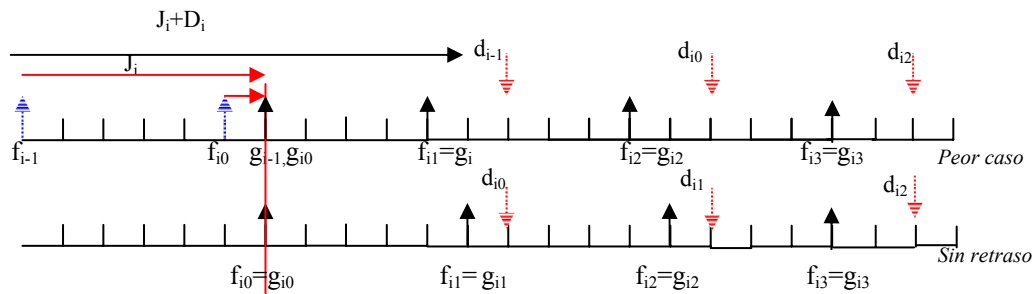


Figura 9. Patrones de activaciones de peor caso y sin retraso de tarea ejecutando con CBSM

Podemos ver en esta figura como la interferencia de peor caso producida por la tarea  $\tau_i$  planificada con CBSM es la misma que la producida por una tarea periódica con mismo periodo, retaso y tiempo de ejecución pero plazo  $D_i+J_i$ , pudiendo reducir así la interferencia sobre el resto de tareas. Por tanto el servidor CBSM disminuye la interfe-

rencia de una tarea sobre el resto, independientemente de los parámetros de dicha tarea.

### 6. Tiempo de respuesta de peor caso de sistemas planificados en EDF con servidores CBSM.

A partir de las ecuaciones descritas por Palencia y González para planificación EDF [7], y aplicando que la contribución de peor caso de las tareas servidas con CBSM es equivalente a la de una tarea con los mismos T,J,C pero plazo  $D+J$ , podremos determinar el tiempo de respuesta de peor caso de las tareas, tanto de las periódicas servidas con CBSM como las que no. Las ecuaciones que determinan los tiempos de respuesta de peor caso de  $\tau_a$  son las siguientes:

- Contribución de peor caso cada tarea,  $\tau_i$ , al periodo de ocupación de longitud  $t$  cuando el plazo de  $\tau_a$  ocurre en  $D$ :

$$W_i(t, D) = \min \left( \left\lceil \frac{t + J_i}{T_i} \right\rceil, \left\lfloor \frac{J_i + D - d_i}{T_i} \right\rfloor + 1 \right) C_i \quad (1)$$

(Cuando  $\tau_i$  no servida con CBSM)

$$W_i(t, D) = \min \left( \left\lceil \frac{t + J_i}{T_i} \right\rceil, \left\lfloor \frac{D - d_i}{T_i} \right\rfloor + 1 \right) C_i \quad (2)$$

(Cuando  $\tau_i$  está servida con CBSM)

- Posibles posiciones de los plazos de la activaciones de la tarea  $\tau_a$  respecto el comienzo del periodo de ocupación:

$$\psi = \cup \{ (p-1)T_i - J_i + d_i^* \mid \forall p = 1 \dots \left\lceil \frac{L + J_i}{T_i} \right\rceil, \forall i \quad (3)$$

$$d_i^* = \{ d_i \text{ si no CBSM } d_i + J_i \text{ si es CBSM} \},$$

donde también se han incluido los plazos generados por la propia tareas  $\tau_a$ .  $L$  es el periodo de ocupación que luego calcularemos

- Cálculo del instante de liberación  $A$  de la primera activación de  $\tau_a$  en el periodo de ocupación y posición de su plazo, para cada  $\tau_{ap}$ :

$$\psi_x = \{ \psi_x \in \psi : (p_a - 1)T_a - J_a + d_a \leq \psi_x < p_a T_a - J_a + d_a \} \quad (4)$$

$$A = \psi_x - (p_a - 1)T_a - J_a + d_a \quad (5)$$

- Posición del deadline respecto  $A$

$$D^A(p_a) = A - J_a + (p_a - 1)T_a + d_a \quad (6)$$

Si la tarea de estudio está planificada con un servidor CBSM  $d_a \rightarrow d_a + J_a$

- Cálculo de tiempo de ocupación del procesador (iterativo):

$$W_a^{A(n+1)}(p) = pC_A + \sum_{\forall i \text{ periódica} \neq a} W_i(W_a^{A(n)}, D_a^A(p)) \quad (7)$$

- Tiempo de respuesta de la activación  $p_a$  de la tarea  $\tau_a$  con primera activación en  $t=A$ .

$$R^A(p_a) = W^A(p_a) - (A - J_a) + (p_a - 1)T_a \quad (8)$$

- Tiempo de respuesta de por caso de  $\tau_a$ :

$$R_a = \max(R_a^A(p)) (\forall p = 1 \dots \left\lceil \frac{L}{T_a} \right\rceil, \forall A \in \psi^*) \quad (9)$$

- Cálculo de la longitud del periodo de ocupación:

$$L^{(n+1)} = \sum_{\forall i} \left\lceil \frac{L^{(n)}}{T_i} \right\rceil C_i, \text{ con } L^{(0)} = \sum_{\forall i} C_i \quad (10)$$

## 7. Conclusiones y trabajo futuro.

En este artículo hemos desarrollado un nuevo método de planificación (servidor CBSM) que disminuye las interferencias debidas a la activación retrasada de las tareas. Además hemos obtenido las expresiones analíticas que nos permiten calcular los tiempos de respuesta exactos de peor caso de las tareas de forma que podemos aplicar un test de planificabilidad sobre el sistema planificado con el nuevo servidor. También hemos analizado el impacto del servidor sobre la ejecución de la propia tarea servida.

Si bien el efecto de la activación retrasada en sistemas monoprocesadores generalmente no es grande, su estudio es muy interesante puesto que nos permitirá extender el análisis a sistemas mul-

tiprocesadores y distribuidos, donde el efecto de la activación retrasada sí que puede tener un impacto drástico sobre la planificabilidad del sistema.

Como trabajo futuro puede estar el desarrollar un método que nos permita determinar en qué condiciones la introducción de un servidor CBSM puede convertir un sistema no planificable en planificable, y la realización práctica de este análisis a sistemas de tiempo real. Otra línea de trabajo futuro es extender el servidor CBSM a sistemas distribuidos con transacciones. Por último, sería interesante el estudio de otros servidores en el control del retraso (como por ejemplo, el servidor IRIS) comparando el resultado entre ambos.

## 8. Referencias

- [1] L. Abeni. "Server Mechanisms for Multimedia Applications". Technical Report RETIS TR98-01, Scuola Superiore S. Anna, Pisa, Italy, 1998.
- [2] Ausdley N y Burns A.: "Real-Time system Scheduling". Department of Computer Science, University of York, Technical Report YCS\_164, 1991.
- [3] J.P. Lehoczky, L. Sha, J.K. Strosnider, "Enhanced Aperiodic Responsiveness in hard Real-time Environments". Proc of Real-time systems Symposium, pag 261-270, 1987
- [4] Liu C.L. and Layland J.W.: "Scheduling Algorithms for Multi-Programming in a Hard Real-Time Environment". Proc. Of Journal of Association for Computing Machinery, vol,20,nº1, pag 46-61, 1973.
- [5] L.Marzario, G. Lipari, P. Balbastre, A. Crespo "IRIS: a new reclaiming algorithm for server-based real-time systems" Real-time Application Symposium (RTAS 04) Toronto(Canada), 2004.
- [6] Palencia J.C., González M. "Schedulability Analysis for Task with Static and Dynamic offsets". 18<sup>th</sup> IEEE Real-Time Systems Symposium 1998
- [7] Palencia J.C., González M. "Response Time Analysis of Distributed Systems Scheduled under EDF Journal of Embedded Computing, Cambridge International Science Publishing, 2003.
- [8] M.Spuri, G. Buttazzo. "Efficient Aperiodic Service under Earliest Deadline Scheduling". 15<sup>th</sup> IEEE Real-Time systems Symposium, 2-11, 1984.
- [9] M.Spuri, G. Buttazzo, F. Sensini. "Robust Aperiodic Scheduling under Dynamic Priority Systems". 16<sup>th</sup> IEEE Real-Time Systems Symposium, pag 210-219, 1995..
- [10] M. Spuri. "Holistic Analysis of Deadline Scheduled Real-Time Distributed Systems". RR-2873, INRIA, France, 1996.
- [11] Tindell K "Guaranteed Message Latencies for Distributed Safety-Critical Hard Real-Time Control Networks". Department of Computer, University of York, Technical Report YCS\_229, 1994