

Interpretación de dos algoritmos EDF *on-line* para la optimización de sistemas distribuidos de tiempo real

Juan M. Rivas, y J. Javier Gutiérrez

Grupo de Ingeniería Software y Tiempo-Real, Universidad de Cantabria
{rivasjm, gutierjj}@unican.es

Resumen. Los planificadores EDF (*Earliest Deadline First*) *on-line* calculan los plazos de planificación en tiempo de ejecución y se utilizan habitualmente en sistemas de tiempo real laxo. En este trabajo se propone una interpretación de dos de estos algoritmos de planificación, EQS (*Equal Slack*) y EQF (*Equal Flexibility*), para su adaptación a sistemas distribuidos de tiempo real estricto en los que la asignación de parámetros de planificación se realiza *off-line*, es decir, antes de la ejecución del sistema. La adaptación propuesta permite asignar plazos de planificación en sistemas planificados por EDF, pero también permite asignar prioridades en sistemas planificados por prioridades fijas. Los resultados obtenidos en la evaluación de los algoritmos propuestos los colocan como los más adecuados para planificar sistemas en los que los plazos son superiores a los periodos, circunstancia que suele ser habitual en los sistemas distribuidos, y secuencias largas de tareas y mensajes en respuesta a los eventos.

Palabras clave: tiempo real, sistemas distribuidos, optimización, prioridades fijas, EDF.

1 Introducción¹

La política de planificación EDF (*Earliest Deadline First*) continúa captando la atención de investigadores en los ámbitos académico e industrial, en especial por los beneficios que se pueden obtener en el incremento del uso de los recursos procesadores. Podemos encontrar planificadores EDF en (1) lenguajes de tiempo real como Ada [1] o Java (RTSJ [2]), (2) en sistemas operativos de tiempo real como SHaRK [3], ERIKA [4], y OSEK/VDX (implementado en el nivel de aplicación) [5], (3) en redes de comunicaciones de tiempo real como el bus CAN [6] u otras de propósito general [7], o (4) en software de intermediación de tiempo real como RT-CORBA [8].

Por otra parte, en la actualidad es común encontrarnos con sistemas de tiempo real constituidos por varios procesadores conectados mediante una o varias redes de comunicación, por lo que las aplicaciones que se ejecutan en ellos tienen una naturaleza distribuida. Incluso los actuales procesadores con varios núcleos (en particular los *many-core*) permiten concebir las aplicaciones como si fueran distribuidas. En estos sistemas

¹ Este trabajo ha sido financiado en parte por el Gobierno de España en el proyecto TIN2014-56158-C4-2-P (M2C2).

se pueden tener secuencias de tareas y mensajes que se ejecutan en diferentes procesadores, o son enviados por diferentes redes de comunicaciones en respuesta a los eventos que se generan en el entorno.

Sobre los sistemas distribuidos planificados por EDF se suelen aplicar técnicas de asignación de plazos de planificación que normalmente hacen una distribución de los requisitos temporales en función de los tiempos de ejecución de peor caso [9][10][11]. En un trabajo reciente [12] demostramos que en algunos casos la aplicación de estas técnicas a EDF obtenía unos resultados muy pobres en cuanto a la utilización que se podía alcanzar en los recursos procesadores, y propusimos nuevas técnicas de asignación de plazos que rompían con la regla de que la suma de los plazos asignados a una secuencia de tareas debía cumplir con el requisito temporal establecido.

En este trabajo, nos centramos en dos algoritmos de planificación, EQS (*Equal Slack*) y EQF (*Equal Flexibility*), que fueron propuestos en [13] para la asignación *on-line* de plazos de planificación en sistemas de tiempo real laxo, y que se han seguido utilizando con este fin [14][15][16]. Estos algoritmos tampoco cumplen con la mencionada regla de suma de plazos de planificación, y el objetivo es hacer una interpretación de los mismos para obtener nuevos métodos de asignación *off-line* de parámetros de planificación, que puedan aplicarse por tanto a sistemas de tiempo real estricto. Aunque estos algoritmos están pensados para EDF, en este trabajo se estudiará también su comportamiento en la asignación a planificadores de prioridades fijas (FP, *Fixed Priority*).

El documento queda organizado de la siguiente manera. En el apartado 2 se describe el modelo de sistema distribuido utilizado en este trabajo. En el apartado 3 se hace un breve repaso de las técnicas de asignación de parámetros de planificación en sistemas distribuidos con las que se compararán los resultados. El apartado 4 trata sobre el proceso de interpretación de los algoritmos EQS y EQF de asignación de plazos de planificación para su adaptación como técnicas de asignación *off-line*. El estudio del rendimiento de estos dos algoritmos con respecto a las técnicas existentes se presenta en el apartado 5. Por último, en el apartado 6 se plantean las conclusiones de este trabajo.

2 Modelo del sistema

Se utiliza el modelo MAST [17][18] de sistemas distribuidos, que está alineado con el estándar MARTE [19] del OMG (*Object Management Group*). El sistema está compuesto por varios recursos procesadores (CPUs y redes de comunicaciones) que pueden usar planificación FP o EDF. El modelo de tareas está compuesto por flujos de principio a fin (flujos e2e) distribuidos, que son activados de manera periódica o esporádica con un intervalo mínimo entre llegadas. Cada flujo e2e T_i está compuesto por un conjunto N_i de actividades que pueden ser tareas ejecutando en un procesador o mensajes enviados a través de la red de comunicaciones. Cada actividad se activa cuando finaliza la ejecución de la anterior y está estáticamente asignada a un procesador o red. Con este modelo, el análisis de planificabilidad de los mensajes en las redes es similar al de las tareas en los procesadores.

La Figura 1 muestra un ejemplo de un flujo e2e sencillo con tres actividades, cada una ejecutando en un recurso procesador (PR_k) distinto. La llegada del evento externo

e_i activa el flujo $e2e$ con su periodo T_i . Cada actividad τ_{ij} tiene asociado un tiempo de ejecución de peor caso (C_{ij}), y un tiempo de ejecución de mejor caso (C_{ij}^b). Los requisitos de tiempo que se consideran son los plazos de principio a fin, D_i , medidos desde la llegada del evento externo hasta que finaliza la última actividad del flujo $e2e$. Los plazos pueden ser mayores que los periodos. Cada actividad puede tener asignado un offset (Φ_{ij}), que indica la cantidad mínima de tiempo que debe transcurrir desde la llegada del evento externo hasta que la actividad pueda activarse.

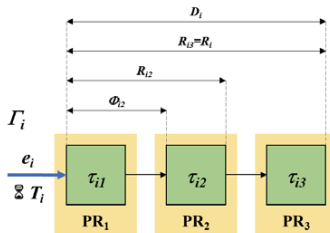


Fig. 1. Flujo de principio a fin (F_i) con 3 actividades

Para cada actividad τ_{ij} se define su tiempo de respuesta de peor caso (o una cota superior de éste) R_{ij} como la cantidad de tiempo máximo que tarda la actividad en ejecutarse contado desde que llega el evento externo. Similarmente se define el tiempo de respuesta de mejor caso (o una cota inferior de éste) R_{ij}^b . Estos valores de los tiempos de respuesta se obtienen tras la aplicación de una técnica de análisis de planificabilidad. Si los tiempos de respuesta de peor caso son menores o iguales que sus plazos asociados, se dice que el sistema es planificable.

Debido a las relaciones de precedencia en los flujos $e2e$, las activaciones pueden tener *jitter* de activación (J_{ij}), que es la variación máxima que puede experimentar la actividad τ_{ij} entre sus instantes de activación. Similarmente, el evento externo puede tener *jitter*.

Cada actividad se planifica utilizando su parámetro de planificación, que puede ser una prioridad fija (P_{ij}) para planificación FP, o un plazo de planificación para planificación EDF. Los plazos de planificación son valores utilizados para planificar, pero no representan un requisito temporal. Dependiendo de si el sistema distribuido tiene sus relojes sincronizados o no, distinguimos dos tipos de plazos de planificación:

- Plazos globales de planificación (SD_{ij}), que toman como referencia el instante de activación del flujo $e2e$. Hacer uso de este tipo de plazos de planificación requiere sincronización en los relojes, y por ello a este tipo de planificación lo llamamos GC-EDF (*Global-Clock* EDF).
- Plazos locales de planificación (Sd_{ij}), que toman como referencia la propia activación de la actividad. Este esquema no requiere sincronización en los relojes, y por lo tanto lo llamamos planificación LC-EDF (*Local-Clock* EDF).

Además del modelo, MAST también proporciona un conjunto de herramientas de análisis y optimización de sistemas distribuidos [20]. Las técnicas de análisis determinan la planificabilidad de los sistemas calculando tiempos de respuesta de peor caso, que se comparan con los plazos impuestos. Para sistemas distribuidos, en MAST se implementa el análisis holístico para FP [21], GC-EDF [22] y LC-EDF [10]. En MAST también se ofrece una variedad de análisis basados en *offsets* que reducen el pesimismo del análisis holístico, y que se implementan para planificación FP [23][24][25], GC-EDF [26] y LC-EDF [27]. Por último, en MAST también se implementan diferentes técnicas de asignación de parámetros de planificación que serán utilizadas en este trabajo para la evaluación de los algoritmos propuestos. Las técnicas de asignación disponibles se tratan en el siguiente apartado.

3 Asignación de parámetros de planificación

Las técnicas de asignación de parámetros de planificación son de vital importancia en el desarrollo de sistemas de tiempo real. Debido al elevado número de combinaciones de actividades y parámetros de planificación que pueden probarse en un sistema distribuido, es trivial comprobar que encontrar la asignación óptima es un problema NP-difícil [28][29]. Por lo tanto, se hace necesario el uso de técnicas no óptimas, pero que sean tratables computacionalmente.

Para la asignación de prioridades fijas se propuso el algoritmo HOPA [30] (*Heuristic Optimized Priority Assignment*), que es un algoritmo heurístico que hace uso de los tiempos de respuesta de peor caso para asignar y optimizar prioridades fijas en sistemas distribuidos. HOPA es capaz de planificar un mayor número de sistemas, y en menos tiempo, que otros algoritmos de propósito general como el templado simulado [30].

En cuanto a sistemas EDF, Liu [9] destaca los siguientes algoritmos de reparto del plazo de principio a fin: UD, en el que los plazos de planificación son iguales a los de principio a fin; ED, en el que los plazos de planificación son iguales a los de principio a fin, pero se restan los tiempos de ejecución de peor caso de las actividades posteriores en el flujo e2e; PD, en el que los plazos de principio a fin se reparten de manera proporcional a los tiempos de ejecución de cada actividad; y NPD, que es similar a PD, pero en el que además se tiende a dar plazos más largos en aquellos procesadores más cargados.

Con el objetivo de mejorar la capacidad de planificación de estos algoritmos de reparto, y utilizando como base el algoritmo HOPA, se definió el algoritmo HOSDA para la asignación de plazos de planificación en sistemas GC-EDF [31] y LC-EDF [10]. Al igual que HOPA, utiliza los tiempos de respuesta de peor caso obtenidos por alguna técnica de análisis de planificabilidad para asignar y optimizar plazos de planificación.

Estas técnicas que se han destacado están diseñadas para trabajar únicamente en sistemas distribuidos con una única política de planificación. Para superar esta limitación, se propuso el algoritmo heurístico HOSPA [11], que es una fusión entre HOPA y HOSDA para asignar tanto prioridades fijas como plazos de planificación en sistemas distribuidos en los que conviven diferentes políticas de planificación (sistemas heterogéneos). En [11] además se introduce el concepto de plazo virtual, que es un número

intermedio que se transformará en el parámetro de planificación correspondiente: prioridad fija para FP, o plazo de planificación para EDF. Este plazo virtual se puede utilizar para aplicar los algoritmos de reparto de plazos en sistemas con políticas para las cuales no fueron diseñados. HOSPA será uno de los algoritmos utilizados en este trabajo como referencia para la evaluación de las nuevas propuestas.

4 Interpretación de los algoritmos EQS y EQF para la asignación *off-line* de parámetros de planificación

El objetivo es interpretar los algoritmos EQS (*Equal Slack*) y EQF (*Equal Flexibility*) propuestos en [13] para que puedan producir plazos virtuales fijos que luego se puedan transformar en plazos de planificación locales, globales o en prioridades fijas. Así pues, como primer paso estudiamos la formulación original de estos algoritmos, que producen plazos globales de planificación, y la adaptamos al modelo de flujo e2e que utilizamos.

El algoritmo EQS asigna los plazos dividiendo equitativamente el *slack* entre las actividades del flujo e2e, entendiendo *slack* como la diferencia entre el plazo y el tiempo de respuesta de peor caso. Como *a priori* no se conocen los tiempos de respuesta, utiliza los tiempos de ejecución de peor caso como una estimación optimista de este tiempo de respuesta de peor caso. La ecuación de EQS en su formulación original se muestra a continuación [13]:

$$dl(T_i) = ar(T_i) + pex(T_i) + \frac{dl(T) - ar(T_i) - \sum_{j=i}^m pex(T_j)}{m-i+1} \quad (1)$$

donde los términos en los que se expresa tienen los siguientes significados de acuerdo a los conceptos definidos en nuestro modelo:

- $dl(T_i)$: plazo absoluto asignado a la actividad i -ésima del flujo e2e.
- $ar(T_i)$: instante de activación de la actividad i -ésima del flujo e2e.
- $pex(T_i)$: tiempo de ejecución de peor caso de la actividad i -ésima del flujo e2e.
- $dl(T)$: plazo absoluto de principio a fin del flujo e2e.
- m : número de actividades en el flujo e2e.

El algoritmo EQS original descrito en la ecuación (1) está pensado para su ejecución *on-line*, y asigna plazos globales a las actividades teniendo en cuenta los instantes de activación de éstas. Sin embargo, en nuestro modelo pensado para herramientas de asignación *off-line*, estos instantes de activación de las actividades se desconocen. Para adaptar la ecuación (1) a la formulación del modelo MAST que usamos, asumimos que los instantes de activación de las actividades ocurren en el instante cero, lo que significa que los plazos virtuales (Vd) obtenidos toman como referencia la activación del flujo e2e (plazos globales de planificación). Así, el algoritmo adaptado se muestra en la siguiente ecuación:

$$Vd_{ij} = C_{ij} + \frac{D_i - \sum_{k=j}^{N_i} C_{ik}}{N_i - j + 1} \quad (2)$$

El algoritmo EQF original también se basa en el reparto del *slack* (definido de la misma manera que en EQS), pero lo realiza de manera proporcional a los tiempos de ejecución de cada actividad. Para ello define un nuevo concepto llamado “flexibilidad” (*flexibility*), que es la ratio entre el *slack* y el tiempo de respuesta (o una estimación de éste). La ecuación de EQF en su formulación original se describe a continuación [13]:

$$dl(T_i) = ar(T_i) + pex(T_i) + [dl(T) - ar(T_i) - \sum_{j=i}^m pex(T_j)] * \left[\frac{pex(T_i)}{\sum_{j=i}^m pex(T_j)} \right] \quad (3)$$

donde los términos en los que se expresa son los utilizados en EQS.

Adaptamos la ecuación original de EQF para la asignación de plazos virtuales de planificación definidos de acuerdo con nuestro modelo. Para ello seguimos el mismo criterio que aplicamos a EQS, asumiendo la activación de las actividades en el instante cero. La formulación de EQF adaptada al modelo MAST de flujo e2e se muestra en la siguiente ecuación:

$$Vd_{ij} = C_{ij} + \left[D_i - \sum_{k=j}^{N_i} C_{ik} \right] \left[\frac{C_{ij}}{\sum_{k=j}^{N_i} C_{ik}} \right] \quad (4)$$

En realidad, las adaptaciones que proponemos de EQS y EQF obtienen plazos virtuales de planificación que razonablemente sólo estarían indicados para su uso con planificación GC-EDF. Así pues, es necesaria una interpretación de estos plazos virtuales para su transformación en parámetros de planificación [11]. Para planificación FP, los plazos virtuales se transforman en prioridades fijas utilizando el criterio *Deadline Monotonic* en cada recurso procesador. Para planificación GC-EDF y LC-EDF, los plazos virtuales se usan directamente como plazos de planificación. En el caso de LC-EDF, se están utilizando plazos locales de planificación que incumplen la regla mencionada en la introducción relativa a que la suma de los plazos locales de planificación en un flujo e2e cumplan con el requisito temporal impuesto. Como se demuestra en [12], esta regla que parece muy razonable no es en absoluto necesaria, e incluso tiene consecuencias negativas en el rendimiento que se puede obtener con la planificación LC-EDF.

5 Evaluación

En este apartado, evaluamos el rendimiento de la interpretación de EQS y EQF propuesta cuando se aplica en sistemas distribuidos de tiempo real cuya planificación se realiza *off-line*. Para ello, se hace una comparación con la aplicación de las técnicas de asignación que hemos introducido en el apartado 3, para las diferentes políticas de planificación que tratamos en el apartado 2: FP, GC-EDF y LC-EDF. En esta evaluación utilizamos la herramienta GEN4MAST [32], que permite generar un conjunto de pruebas lo suficientemente amplio para obtener resultados estadísticamente relevantes, y permite además la aplicación automática de las técnicas de asignación bajo estudio.

Las características del conjunto de sistemas sintéticos generado se plantean con dos objetivos principales: (1) abarcar un conjunto amplio de sistemas lo suficientemente complejos como para dar validez a los resultados, y (2) que el estudio requiera un tiempo total de ejecución razonable. Bajo estas dos premisas, todos los sistemas generados poseen las siguientes características básicas:

- Poseen 10 flujos $e2e$ y 5 recursos procesadores.
- Los flujos $e2e$ no recorren el mismo recurso procesador en más de una ocasión. Esto sólo es posible si el número de actividades en el flujo $e2e$ es menor o igual que el número de recursos procesadores. Cuando esta condición no se cumple, la localización de las actividades se realiza de forma aleatoria.
- Los periodos se seleccionan de forma aleatoria en el rango [100,1000] utilizando una distribución de probabilidad logarítmica-uniforme [33].
- Los tiempos de ejecución de peor caso de las actividades se calculan con el algoritmo *UUnifast* [34]. Los tiempos de ejecución de mejor caso son iguales a cero. Todos los recursos procesadores tienen la misma utilización.

A partir de estas características básicas, generamos el conjunto de sistemas de estudio variando otras tres características fundamentales de éstos:

- Plazos de principio a fin: se generan sistemas de forma que la ratio D_i/T_i vaya tomando todos los valores en el conjunto {1,2,4,6,8,10,12,14,16,18,20}. Estas variaciones se hacen con sistemas con 10 actividades por flujo $e2e$.
- Número de actividades en los flujos $e2e$ con valores en el conjunto {4,6,8,10,12,14,16,18,20}. Estas variaciones se hacen con plazos de principio a fin $D_i=N_i^*T_i$ para sistemas FP y LC-EDF, y $D_i=5*T_i$ para sistemas GC-EDF, donde N_i y T_i son respectivamente el número de actividades y el periodo de los flujos $e2e$.
- Series de utilizaciones medias del sistema en el rango [10, 96](%), con variación del 1%. Esta generación de series se realiza para todos los sistemas.

Para obtener resultados estadísticamente relevantes, se generan 30 sistemas con cada una de las combinaciones de las características evaluadas. Una vez generado el conjunto de sistemas, se aplican sobre éste las técnicas de asignación de parámetros de planificación que se han citado previamente: UD, ED, PD, EQS, EQF y HOSPA. Estudiaremos por separado las tres políticas de planificación (FP, LC-EDF y GC-EDF) y en todos los casos utilizaremos el análisis holístico [21][22][10] para determinar la planificabilidad del sistema. Cómo métrica para comparar el rendimiento de las diferentes técnicas, nos centraremos en observar cuál es la utilización máxima planificable (UMP) alcanzada por cada una de las técnicas de asignación utilizada. Una UMP mayor indica que la asignación de parámetros de planificación fue capaz de encontrar una solución planificable para sistemas con mayor carga computacional.

En la Figura 2 se muestra la UMP media obtenida por las diferentes técnicas cuando se utilizan para asignar prioridades en planificación FP. En el estudio de la UMP en función de los plazos de principio a fin (Figura 2a) observamos dos zonas diferenciadas. En primer lugar, para plazos de principio a fin pequeños ($D_i=T_i$), las mejores técnicas son HOSPA y PD, que consiguieron planificar sistemas que en promedio tenían un 30%

de utilización. En esta zona, EQS, EQF, ED y UD obtienen un rendimiento claramente inferior, con alrededor de un 20% de UMP media. Por otro lado, a medida que los plazos de principio a fin aumentan, la UMP media de EQS y EQF crece a un ritmo superior que el del resto de técnicas. Observamos que a partir de aproximadamente $D_f=4*T_s$, EQF adelanta a HOSPA, convirtiéndose en la técnica que mayores UMP medias obtiene.

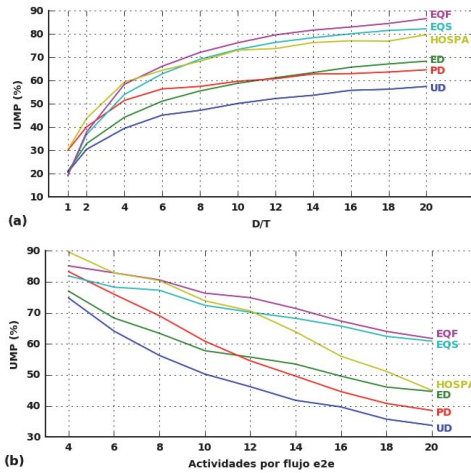


Fig. 2. UMP media (%) para sistemas planificados con FP: (a) diferentes plazos de principio a fin; (b) diferentes números de actividades por flujo e2e.

Si observamos los resultados para sistemas con diferentes números de actividades por flujo e2e (Figura 2b), también se distinguen dos zonas. Mientras que para 4 actividades/flujo e2e el algoritmo HOSPA es el que mayor UMP media obtuvo (90%), a partir de 6 actividades/flujo EQF adelanta a HOSPA en términos de UMP media. La diferencia de rendimiento reportada entre EQF y HOSPA aumenta a medida que los flujos e2e son más largos. Para flujos e2e con 20 actividades, EQF posee una UMP media 17% superior a la obtenida por HOSPA.

En la Figura 3 se muestran los resultados de las diferentes técnicas bajo estudio cuando se utilizan para asignar plazos globales de planificación para GC-EDF. En este caso, y a diferencia de lo observado previamente para FP, se comprueba que el rendimiento de EQS y EQF se ve claramente superado por HOSPA, tanto para diferentes plazos de principio a fin (Figura 3a), como de números de actividades por flujo e2e (Figura 3b). Cabe notar que para los resultados de la Figura 3b se está utilizando un

plazo de principio a fin $D_i=5*T_i$. Esta modificación se realiza a la vista del rendimiento de HOSPA y PD para plazos de principio a fin superiores a $D_i=4*T_i$ (ver Figura 3a), en el que consiguen planificar sistemas con hasta el 96% de utilización (la máxima generada). Este rendimiento se asemeja al de la planificación EDF en sistemas monoprocesadores y tareas independientes.

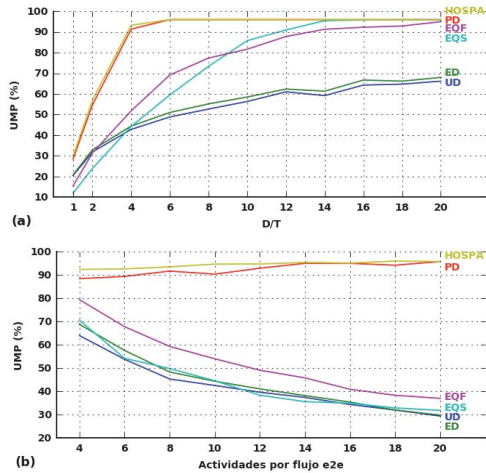


Fig. 3. UMP media (%) para sistemas planificados con GC-EDF: (a) diferentes plazos de principio a fin; (b) diferentes números de actividades por flujo e2e.

Por último, en la Figura 4 se evalúa el rendimiento de las diferentes técnicas bajo estudio cuando asignan plazos locales de planificación para LC-EDF. En este caso, los algoritmos HOSPA y PD se modifican utilizando el criterio LC-EDF-GSD propuesto en [12] para asignar plazos locales de planificación con valores equivalentes a plazos globales. El motivo es comparar las nuevas propuestas con las que obtienen los mejores resultados, que como se demuestra en [12] consiste en el uso de este criterio para la asignación de plazos locales de planificación en LC-EDF. A la vista de los resultados mostrados en la figura, y al contrario de lo observado para planificación GC-EDF, el algoritmo EQF aquí posee un rendimiento que en promedio supera al de las técnicas existentes, aventajando a HOSPA en hasta un 7% en términos de UMP media. La superioridad de EQF en UMP media se mantiene en todos los casos estudiados, salvo para plazos de principio a fin iguales a los periodos (Figura 4a), en el que todas las técnicas obtienen una UMP media en torno al 20%.

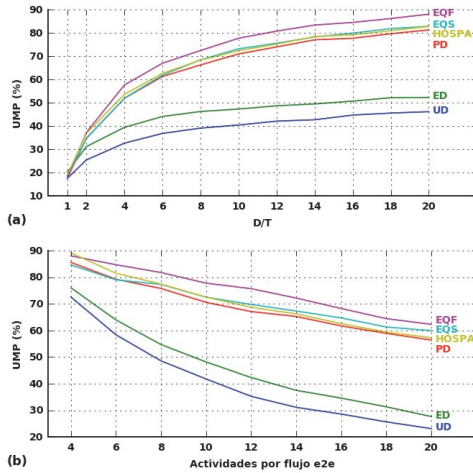


Fig. 4. UMP media (%) para sistemas planificados con LC-EDF: (a) diferentes plazos de principio a fin; (b) diferentes números de actividades por flujo e2e.

6 Conclusiones

En este trabajo se ha presentado la adaptación como técnica *off-line* de dos algoritmos de asignación de plazos de planificación, EQS y EQF, que en su origen fueron diseñados para la planificación EDF *on-line* de sistemas de tiempo real laxo. Esta adaptación permite su aplicación a sistemas distribuidos de tiempo real estricto planificados por prioridades fijas, o EDF tanto con relojes locales como con reloj global. En todos los casos la planificabilidad de los sistemas resultantes se puede comprobar mediante la aplicación de las técnicas de cálculo de tiempos de respuesta correspondientes.

Se ha evaluado el rendimiento de las adaptaciones propuestas de ambos algoritmos, comparándolos con las técnicas de asignación de parámetros de planificación existentes. Como resultado de esta evaluación, se ha comprobado cómo EQF, y en menor medida también EQS, poseen una capacidad de planificación equiparable, y en muchas situaciones superior, al algoritmo heurístico HOSPA para sistemas con planificación FP y LC-EDF. De los resultados obtenidos cabe destacar lo siguiente:

- Es sorprendente que estos algoritmos obtengan los peores resultados para la planificación GC-EDF que es para la que fueron diseñados como técnicas *on-line*.

- Se dispone de dos nuevos algoritmos con los que optimizar la asignación de prioridades en FP, o de plazos locales de planificación en LC-EDF, para sistemas distribuidos con plazos superiores al periodo o con flujos e2e largos.
- Los algoritmos propuestos no son iterativos, por lo que sus resultados se obtienen muy rápidamente en comparación con cualquier otra opción iterativa como HOSPA.

Como conclusión general, se observa que en la asignación de parámetros de planificación ninguno de los algoritmos es el mejor en todas las situaciones. Por este motivo, en la optimización de un sistema, quizá la solución pase por probar todos los algoritmos que no son iterativos, y si no se encuentra una solución, probar entonces los iterativos con las mejores soluciones encontradas por los anteriores como asignación inicial.

7 Bibliografía

1. ISO/IEC, 2012. Ada 2012 Reference Manual. Language and Standard Libraries - International Standard ISO/IEC 8652:2012(E) (2012).
2. RTSJ (Real-Time Specification for Java) home page, <http://www.rtsj.org>
3. S.Ha.R.K. (Soft Hard Real-Time Kernel) home page, <http://shark.sssp.it/>
4. ERIKA Enterprise, Evidence home page, <http://www.evidence.eu.com/>
5. C. Diederichs, U. Margull, F. Slomka, and G. Wirrer, "An application-based EDF scheduler for OSEK/VDX". Design, Automation and Test in Europe, DATE '08 , págs. 1045-1050 (2008)
6. P. Pedreiras, and L. Almeida, "EDF message scheduling on controller area network". Computing & Control Engineering Journal 13(4), págs. 163-170 (2002).
7. M. Di Natale, and A. Meschi, "Scheduling Messages with Earliest Deadline Techniques". Real-Time Systems 20(3), págs. 255-285 (2001).
8. OMG (Object Management Group), Realtime Corba Specification. v1.2, 2005. <http://www.omg.org/spec/RT/1.2/>
9. J.W.S. Liu, "Real-time systems". Prentice Hall (2000).
10. J.M. Rivas, J.J. Gutiérrez, J.C. Palencia and M. González Harbour, "Optimized Deadline Assignment and Schedulability Analysis for Distributed Real-Time Systems with Local EDF Scheduling". Proc. of the 8th International Conference on Embedded Systems and Applications (ESA), Las Vegas (Nevada, USA), págs. 150-156 (2010).
11. J.M. Rivas, J.J. Gutiérrez, J.C. Palencia, Michael González Harbour, "Schedulability analysis and optimization of heterogeneous EDF and FP distributed real-time systems". Proc. of the 23rd Euromicro Conference on Real-Time Systems, Porto, págs. 195-204 (2011).
12. J.M. Rivas, J.J. Gutiérrez, J.C. Palencia and M. González Harbour, "Deadline Assignment in EDF Schedulers for Real-Time Distributed Systems". IEEE Transactions on Parallel and Distributed Systems, Vol. 26(10), págs. 2671-2684 (2015).
13. B. Kao and H. Garcia-Molina, "Deadline Assignment in a Distributed Soft Real-Time System". IEEE Transactions on Parallel and Distributed Systems, vol. 8, no. 12 (1997).
14. J. Yu and R. Buyya, "A taxonomy of workflow management systems for grid computing". Journal of Grid Computing, vol. 3, no. 3-4, págs. 171-200 (2005).
15. Z. Wu, X. Liu, Z. Ni, D. Yuan and Y. Yang, "A market-oriented hierarchical scheduling strategy in cloud workflow systems". The Journal of Supercomputing, vol. 63, no. 1, págs. 256-293 (2013).

16. J.H. Son and M. Ho Kim, "Improving the performance of time-constrained workflow processing". *Journal of Systems and Software*, vol. 58, no. 3, págs. 211-219 (2001).
17. Michael González Harbour, J.J. Gutiérrez, J.C. Palencia, J.M. Drake, "MAST: Modeling and Analysis Suite for Real Time Applications". *Proc. of the 13th Euromicro Conference on Real-Time Systems*, Delft (The Netherlands), págs. 125-134 (2001).
18. Michael González Harbour, J.J. Gutiérrez, J.M. Drake, P. López, J.C. Palencia, "Modeling distributed real-time systems with MAST 2". *Journal of Systems Architecture*, vol 56, no. 6, Elsevier, págs. 331-340 (2013).
19. Object Management Group, "UML profile for MARTE: Modeling and Analysis of Real Time Embedded Systems, version 1.1". *OMG document formal/2011-06-02* (2011).
20. MAST, página web <http://www.mast.unican.es>
21. K.W. Tindell and J. Clark, "Holistic Schedulability Analysis for Distributed Hard Real-Time Systems". *Microprocessing and Microprogramming*, vol. 50, no. 2-3 (1994).
22. M. Spuri, "Analysis of Deadline Scheduled Real-Time Systems". *Research Report RR-772*, INRIA, France (1996).
23. K.W. Tindell, "Adding Time-Offsets to Schedulability Analysis". *Department of Computer Science, University of York, Technical Report YCS-221* (1994).
24. J. Mäki-Turja and M. Nolin, "Efficient implementation of tight response-times for tasks with offsets". *Real-Time Systems Journal* 40(1), págs. 77-116 (2008).
25. J.C. Palencia and M. González Harbour, "Exploiting Precedence Relations in the Schedulability Analysis of Distributed Real-Time Systems". *Proc. of the 20th Real-Time Systems Symposium (RTSS)*, págs. 328-339 (1999).
26. J.C. Palencia and M. González Harbour, "Offset-Based Response Time Analysis of Distributed Systems Scheduled under EDF". *Proc. of the 15th Euromicro Conference on Real-Time Systems (ECRTS)*, Porto (Portugal), págs. 3-12 (2003).
27. U. Díaz-de-Cerio, J.P. Uribe, M. González Harbour and J.C. Palencia, "Adding precedence relations to the response-time analysis of EDF distributed real-time systems". *Proc. of the 22nd International Conference on Real-Time Networks and Systems*, págs. 129-138 (2014).
28. A. Burns, "Scheduling Hard Real-Time Systems: A Review". *Software Engineering Journal* vol. 6, no. 3, págs. 116-128 (1991).
29. K.W. Tindell, A. Burns and A.J. Wellings, "Allocating Real-Time Tasks. An NP-Hard Problem Made Easy". *Real-Time Systems Journal*, vol. 4, no. 2, págs. 145-165 (1992).
30. J.J. Gutiérrez and M. González Harbour, "Optimized priority assignment for tasks and messages in distributed hard real-time systems". *Proc. of the 3rd Third Workshop on Parallel and Distributed Real-Time Systems (WPDRTS)*, Santa Barbara (California, USA), págs. 124-132 (1995).
31. J.M. Rivas and J.J. Gutiérrez (Director), "Algoritmo de asignación de plazos globales en sistemas distribuidos de tiempo real con planificación EDF: comparativa de estrategias de planificación". *Tesis de Máster, Universidad de Cantabria* (2009).
32. J.M. Rivas, J.J. Gutiérrez and M. González Harbour, "GEN4MAST: A Tool for the Evaluation of Real-Time Techniques Using a Supercomputer". *Proc. of the 3rd International Workshop on Real-Time and Distributed Computing in Emerging Applications (REACTION)* (2014)
33. P. Emberson, R. Stafford and R.I. Davis, "Techniques for the synthesis of multiprocessor tasksets". *Proc. of the 1st International Workshop on Analysis Tools and Methodologies for Embedded and Real-time Systems (WATERS)*, Brussels, págs. 6-11, (2010).
34. E. Bini and G.C. Buttazzo, "Measuring the performance of schedulability tests". *Real-Time Systems*, vol 30, no. 1-2, págs. 129-154 (2005).