

تحسين أداء خوارزمية الجدولة EDHS على المهام الدورية وشبه الدورية في نظم الزمن الحقيقي متعددة المعالجات

د. محمد حجازية*

رلى مريشة**

(تاريخ الإيداع 26 / 10 / 2016. قُبِلَ للنشر في 23 / 1 / 2017)

□ ملخص □

يقدم البحث طريقة جديدة لتحسين أداء خوارزمية الجدولة EDHS (الزمن الحرج الأقصر والتقسيم الأعلى أولوية) على كل من المهام الدورية وشبه الدورية في أنظمة الزمن الحقيقي متعددة المعالجات، حيث تعتبر هذه الخوارزمية من أهم خوارزميات الجدولة في الزمن الحقيقي لقدرتها على حل مشكلة الجدولة الجزئية وشبه الجزئية. شملت هذه الدراسة جدولة مهام دورية وشبه دورية ذات قيود زمنية مساوية لدورها، ومستقلة، و قابلة للمقاطعة (للتطبيع المانع-المنع) على عدة معالجات متجانسة، وتمت عملية تحليل الأداء بحساب معدل نجاح الجدولة وعدد المقاطعات وعدد الهجرات وتبديلات السياق وزمن انتظار المهام في الأرتال. تم الحصول على نتائج تطبيق هذه الخوارزمية لأول مرة على المهام الدورية وتمت مقارنة أداءها عند تطبيقها على المهام شبه الدورية، حيث يقترح البحث طريقة لتعديل عمل هذه الخوارزمية تساعد في تحسين أدائها.

الكلمات المفتاحية: خوارزمية الجدولة EDHS، المهام الدورية، المهام شبه الدورية، نظم الزمن الحقيقي، الأنظمة متعددة المعالجات

* أستاذ مساعد - قسم هندسة الحاسبات والتحكم الآلي - كلية الهندسة الميكانيكية والكهربائية - جامعة تشرين - اللاذقية - سورية.
** طالبة دكتوراه - قسم هندسة الحاسبات والتحكم الآلي - كلية الهندسة الميكانيكية والكهربائية - جامعة تشرين - اللاذقية - سورية.

Performance Improvement of EDHS Algorithm for Periodic and Sporadic Tasks on Real-Time Multiprocessor Systems

Dr. Mohammed Hijazieh*
Rula Mreisheh**

(Received 26 / 10 / 2016. Accepted 23 / 1 / 2017)

□ ABSTRACT □

The paper presents a new approach to improve the performance of EDHS scheduling algorithm (Earliest Deadline and Highest Priority Split) for periodic and sporadic tasks on real time multiprocessor systems. EDHS is one of the most important algorithms in real time multiprocessor systems and that for its ability to solve the problem of partitioning and semi-partitioning scheduling.

This paper considers the scheduling of n periodic and sporadic , in-dependent, and preempted tasks with implicit deadlines on a platform of m homogenous multiprocessor, and the performance analysis has been done by calculating the success rate, preemption count, migration count, the number of context switches, and tasks' average waiting time in queues.

The results of EDHS algorithm on periodic tasks have been obtained for the first time, and it has been compared with sporadic ones, then the paper suggests a new approach to modify EDHS scheduling way that helps to improve the performance of EDHS.

Key Words: EDHS scheduling algorithm, periodic tasks, sporadic tasks, real-time systems, multiprocessor systems.

* Associate Professor, Department of computer and automatic control Engineering, Faculty of Mechanical and electrical Engineering, Tishreen University, Lattakia, Syria.

**PhD student, Department of computer and automatic control Engineering, Faculty of Mechanical and electrical Engineering, Tishreen University, Lattakia, Syria.

مقدمة:

تعرف أنظمة الزمن الحقيقي بأنها الأنظمة التي يجب للمهام فيها أن تعطي نتائج صحيحة ضمن الزمن المحدد. وبالتالي ترتبط فعالية النظام بحسب قدرته على الاستجابة لهذه المهام على أداء الحسابات الضرورية اللازمة ضمن الزمن المحدد لكل مهمة ، حيث تعرف **المهمة Task**: بأنها وحدة التنفيذ الأساسية الخاصة التي تتم عليها عملية الجدولة حيث يتم تخصيصها في وحدة المعالجة المركزية (CPU) استناداً إلى خوارزميات الجدولة [1,2]. وفي حال ورود عدة مهام بنفس الوقت يحتاج الحاسب عندها لجدولة الحسابات للعمليات الواردة ليتمكن من الاستجابة لكل مهمة ضمن الوقت المخصص لها ، وفي حال عدم تمكن الحاسب من ذلك قد يؤدي هذا إلى عدة عواقب تتفاوت في خطورتها حسب نوع المهمة.

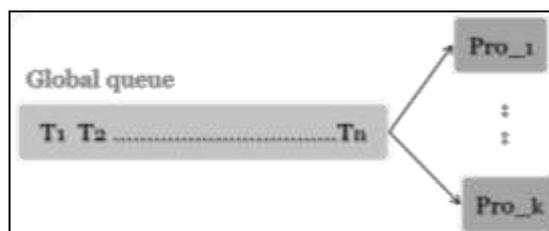
منظومة الزمن الحقيقي اللينة soft real time system تراعي القيود الزمنية المفروضة عليها ولكن إذا أخفقت استمر النظام بالعمل لكن بدون الفائدة المرجوة ولا يؤدي بالضرورة إلى نتائج كارثية ، كلما أنهت المهمة عملها بعد الموعد النهائي كلما كانت النتائج أسوأ ومثال ذلك، أجهزة الصراف الآلي ATM [1,2].

أما **منظومة الزمن الحقيقي الصلبة hard real time system** تستخدم في إدارة المهمات الحساسة التي تتطلب دقة متناهية بالعمل يجب أن تستجيب هذه المنظومة في المدة الزمنية المطلوبة حتماً، وإلا قد يؤدي ذلك إلى نتائج كارثية أو خسائر مادية وكمثال نظام التحكم في الطائرة، ونظام الكبح المضاد للإقفال ABS في السيارات. ومن هنا جاءت أهمية **المجدول** ضمن الحاسب الذي يقوم باستخدام خوارزمية جدولة معينة لتلبية وتنفيذ عمل جميع هذه المهام خلال زمن التنفيذ المحدد لها أي يتم مراعاة القيود الزمنية للمهام حيث يتم في نهاية عملية الجدولة اتخاذ قرار ببداية تنفيذ أحد هذه المهام في **الأنظمة وحيدة المعالج** وهي الأنظمة التي يتم فيها تنفيذ مهمة واحدة من رتل المهام وتبقى المهام الأخرى منتظرة حتى الانتهاء من التنفيذ ، أو يعمل المجدول على البدء بتخصيص عدة مهام على عدة معالجات في **الأنظمة المتعددة المعالجات** وهي الأنظمة التي يتم فيها تنفيذ أكثر من مهمة في الوقت نفسه وذلك من خلال توزيع المهام على المعالجات.

تقسم المعالجات المتعددة في نظم الزمن الحقيقي إلى نوعين: **معالجات متجانسة (Homogenous)**: المعالجات لها نفس الخصائص تماماً مثل تطابق عدد المسجلات وسرعة المعالجة **ومعالجات متغايرة (Heterogeneous)**: لا تكون للمعالجات نفس الخصائص تماماً بل تختلف بعدد المسجلات وسرعة المعالجة والسعة [2,3].

تقسم الجدولة في أنظمة الزمن الحقيقي متعددة المعالجات إلى ثلاثة أنواع [4]:

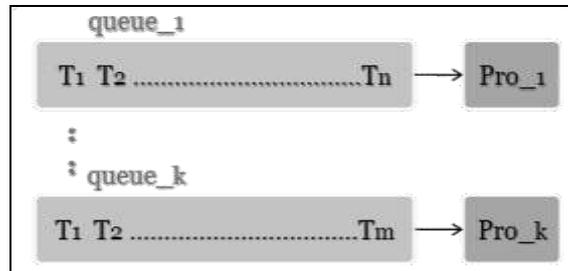
1-الجدولة العامة Global Scheduling: تتوضع كافة المهام في رتل عام واحد يتم مشاركته من قبل جميع المعالجات كما هو مبين في الشكل (1)، كلما أصبح أحد المعالجات غير مشغول أي خالي (idle) تخصص له أحد المهام من الرتل العام، ولكن من مساوئها عدد الهجرات والمقاطعات الكبير جداً الذي تسببه وبالتالي تزيد بشكل كبير من طول الجدولة.



الشكل (1) تمثيل رتل المهام وتوزعه على المعالجات في الجدولة العامة

2-الجدولة الجزئية Partitioning Scheduling: يخصص لكل معالج رتل مهام خاص به، حيث يتم

تنفيذ هذه المهام حصراً على هذا المعالج وبالتالي من مساوئ هذه الطريقة أنه في حال كانت مهمة ما منتظرة في أحد الأرتال لا يمكنها أن تنفذ على معالج آخر خالي غير مخصص لها، كما هو مبين بالشكل(2).

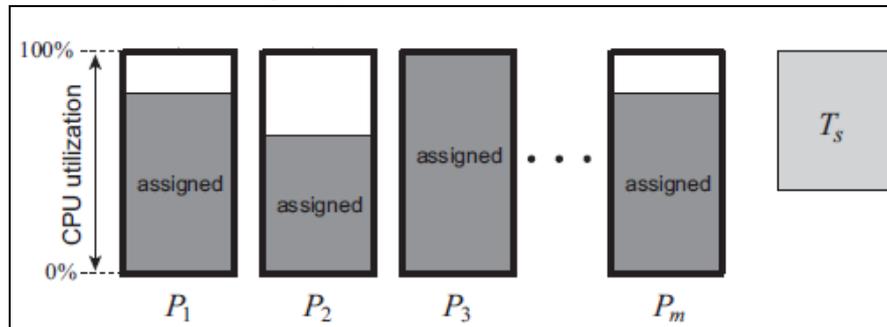


الشكل (2) تمثيل رتل المهام وتوزعه على المعالجات في الجدولة الجزئية

3-الجدولة شبه الجزئية Sime-Partitioning Scheduling: يخصص لكل معالج رتل مهام خاص به،

ولكن من الممكن لأحد المهام الخاصة بمعالج ما أن تنفذ على معالج آخر خالي حين يكون المعالج المخصص لها مشغول.

من مساوئ الجدولة شبه الجزئية (والجدولة الجزئية أيضاً) أنه في حال تم تخصيص المهام إلى الحد الأعظمي الذي يتسع له كافة المعالجات و بقيت أحد المهام طول تنفيذها أكبر من أي سعة متبقية لأي معالج عندها لن تجدول هذه المهمة. يظهر الشكل (3) المهمة T_s المتبقية من المهام التي تم تخصيصها (assigned) على المعالجات ($P_1, P_2, P_3, \dots, P_m$) وأن سعة T_s أكبر من أي سعة متبقية في أي معالج وبالتالي لا يمكن جدولتها.



الشكل (3) مشكلة طريقة الجدولة شبه الجزئية

وتنقسم أنواع المهام في الزمن الحقيقي إلى [3.4]:

(1) المهام اللادورية Aperiodic Tasks: هي المهام التي تنفذ في أوقات غير منتظمة وتحدد

خصائصها بالشكل ($WCET, D$) aperiodic task t_1 حيث:

- ($WCET$) هو زمن تنفيذ المهمة الأعظمي Worst case execution time: وهو الزمن الذي يوضع فيه

المعالج تحت تصرف المهمة.

(D) هو زمن الموعد الحرج النسبي Deadline: وهو الزمن الحدي المسموح به إنجاز المهمة.

(2) مهام دورية Periodic Tasks: هي المهام التي يتم تنفيذها بشكل منتظم في واحدة الزمن وتحدد

خصائصها بالشكل ($WCET, D, P$) periodic task t_2 حيث:

(P) هو الدور Period وهو الزمن الذي تتكرر فيه المهمة.

(3) المهام شبه الدورية Sporadic Tasks: هي المهام التي تنفذ بشكل غير منتظم ويكون لها عدة أزمنة وصول List of activation dates وتحدد خصائصها بالشكل: $t_3(WCET, D, P)$ Sporadic task، حيث يكون لهذه المهام معدل زمني محدد معروف وهو الدور P الذي يعبر عن أصغر فرق زمني بين تفعيلين (وصولين) متتالين لهذه المهمة.

أهمية البحث وأهدافه:

يهدف هذا البحث إلى تحسين أداء عمل خوارزمية EDHS بعد تطبيقها وتحليل أدائها على المهام الدورية لأول مرة ومقارنته مع نتائج أدائها على المهام شبه الدورية، حيث تعتبر هذه الخوارزمية من أهم خوارزميات الجدولة في الزمن الحقيقي لقدرتها على حل مشكلة الجدولة الجزئية وشبه الجزئية. إن عملية تحسين خوارزمية جدولة في الزمن الحقيقي تعني تحسين واحد أو أكثر من معايير قياس أداء الخوارزميات مثل رفع معدل نجاح الجدولة و تقليل كلاً من عدد المقاطعات أو تبديلات السياق أو عدد الهجرات أو معدل الانتظار الواسطي للمهام في الأرتال. إن رفع معدل نجاح جدولة المهام من أهم معايير تحليل الأداء في الزمن الحقيقي لأن الإخفاق في تحقيقه إذا لم يؤدي إلى نتائج كارثية قد يؤدي إلى فقدان المنظومة للهدف المرجو منها. إن عدد المقاطعات والهجرات وتبديلات السياق تزيد من كلفة الجدولة وتزيد أيضاً من وقت الجدولة على المعالجات ولذلك تسعى الكثير من الدراسات إلى تحسين أداء الخوارزميات لتحقيق القيود الزمنية للمهام بأقل طول وكلفة لعملية الجدولة، ومن هنا تأتي أهمية البحث بتحسين أداء خوارزمية EDHS بالنسبة لبارامترات المقارنة المدروسة.

طرائق البحث ومواده:

عند البدء بدراسة خوارزميات الجدولة يجب معرفة القواعد التي تخضع لها عملية جدولة المهام، حيث يتم حساب معدل الاستخدام $utilization$ كل مهمة (T) Task على المعالج كما يلي :

$$U_i = e_i / p_i \quad (1)$$

حيث : $i=1,2,3,\dots,n$ تمثل رقم المهمة و e_i هو زمن تنفيذ المهمة الأعظمي و p_i هو دور المهمة.

تكون مجموعات المهام الدورية قابلة للجدولة على عدة معالجات متجانسة إذا وفقط إذا:

$$U_{\max}(T) \leq 1 \quad (2)$$

$$U_{\text{sum}}(T) \leq m \quad (3)$$

أي أن يكون معدل الاستخدام $utilization$ الأعظمية للمهمة الواحدة أقل أو تساوي الواحد وأن يكون مجموع معدلات الاستخدام $utilization$ لجميع المهام الداخلة إلى الجدولة أصغر أو تساوي عدد المعالجات m [5,6].

3-1 معايير قياس أداء خوارزميات الجدولة [6,7,8]:

توجد عدة معايير تحدد كيفية قياس أداء خوارزميات الجدولة وحساب فعاليتها، ومن أهم هذه المعايير:

معدل نجاح عملية الجدولة $Success Rate$: وهي عدد المرات التي نجحت فيها الخوارزمية في تنفيذ المهام قبل الوقت النهائي المحدد لها ، كلما كان أعلى كلما كانت الخوارزمية أفضل.

عدد المقاطعات $Preemptions$: هو عدد المرات التي تقوم مهمة ذات أولوية أعلى بمقاطعة عمل مهمة أخرى ذات أولوية أقل منها يتم تنفيذها على المعالج. عندها تبقى المهمة التي تمت مقاطعتها في حالة انتظار حتى تنتهي ذات الأولوية الأعلى من التنفيذ لتعود مجدداً وتستكمل عملها على نفس المعالج. تسبب المقاطعات زيادة في كلفة وزمن عملية الجدولة وبالتالي كلما كانت أقل كلما كانت الخوارزمية أفضل.

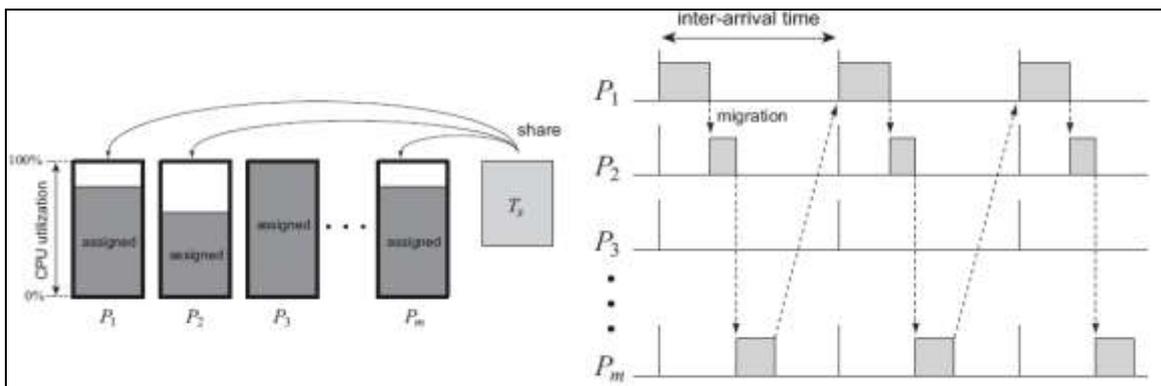
عدد الهجرات $Migrations$: عدد المرات التي تقوم مهمة ذات أولوية أعلى بمقاطعة عمل مهمة أخرى ذات أولوية أقل منها ويتم تنفيذها على المعالج، بعدها تكمل المهمة التي تمت مقاطعتها عملها على معالج آخر غير مشغول، هذا ما يدعى بعملية تبديل السياق context switch وهي تسبب أيضاً زيادة في كلفة وزمن الجدولة بسبب عملية المقاطعة.

معدل الانتظار الوسطي للمهام $average Waiting time$: وهو الزمن الذي تبقى فيه المهام منتظرة في الرتل حتى يتم تخدمها، وكلما كان هذا المعدل أقل كلما كانت الخوارزمية أفضل لأنه يزيد من طول عملية الجدولة ويقدر بالميلي ثانية (ms).

2-3 خوارزمية الجدولة EDHS (الزمن الحرج الأقصر والتقسيم للأولوية الأعلى) (أو:

(Earliest Deadline and Highest Priority Split) [9] :

تعمل هذه الخوارزمية بطريقة الجدولة شبه الجزئية وهي تتميز بقدرتها على حل مشكلة الجدولة شبه الجزئية، حيث تعمل على تقسيم المهمة التي لا تتسع بشكل كامل لأي معالج إلى أجزاء يتم تنفيذ كل جزء منها على معالج بحسب سعته أي تبدأ المهمة بالتنفيذ على أحد المعالجات وعند انتهاء سعته تهاجر إلى معالج لآخر لتكمل عملها وهكذا حتى تنتهي المهمة ويطلق عليها اسم المهمة المشتركة (Shared task). كما هو مبين بالشكل (4).



الشكل (4) طريقة عمل خوارزمية الجدولة EDHS

تسمح هذه الخوارزمية بالهجرات فقط للمهام المشتركة التي يتم توزيع عملها بين المعالجات أي المهام التي لا تتسع بشكل كامل على معالج واحد وتعطي لهذه المهام الأولوية الأعلى بين المهام المخصصة لكل معالج.

تستخدم خوارزمية EDHS خوارزمية الزمن الحرج الأقصر أولاً (EDF) Earliest Deadline First Scheduling في جدولة المهام المخصصة على المعالجات، حيث خوارزمية EDF هي [10] خوارزمية مقادة

بالأولويات وفيها الأولوية الأعلى تعطى للمهمة التي تملك الزمن الحدي الأقل، وتعطى العلاقة الرياضية لهذه الخوارزمية بالشكل التالي:

$$EDF : D_i - t \quad (4)$$

يمكن تلخيص عمل خوارزمية EDHS بالخطوات التالية:

1 - تبدأ المهام المشتركة بالعمل على المعالج الأول حتى تنتهي سعته ثم تهاجر إلى المعالج التالي وهكذا.

2 - للمهام المشتركة الأولوية العليا على المهام المخصصة لكل معالج.

3 - للمهام المشتركة فقط القدرة على الهجرة بين المعالجات حتى إتمام عملها.

4 - جدول المهام المخصصة بخوارزمية الجدولة EDF.

وتعطى المعادلتين الرياضيتين لخوارزمية EDHS بالشكل التالي:

$$c'_s \leq \frac{D_i}{F + 1} \left(1 - \sum_{Tk \in T} \frac{c_k}{p_k} \right) \quad (5)$$

ما زالت هذه الخوارزمية ضمن نطاق الدراسة والعمل البحثي حيث لم يتم تطبيقها وإعطاء نتائجها إلا على المهام شبه الدورية باستخدام 16 معالجا فقط، كما لم تدرس عدد الهجرات والمقاطعات وتبديلات السياق وزمن الانتظار الوسطي للمهام الذي تسببه هذه الخوارزمية حتى للمهام شبه الدورية التي تم العمل عليها في الدراسة المرجعية. يعنى هذا البحث بإتمام العمل لتحليل أداء هذه الخوارزمية وتطبيقها على المهام الدورية لأول مرة ومحاولة تحسين أدائها على كل من المهام الدورية وشبه الدورية.

4-4 خوارزميات الجدولة الأخرى المستخدمة في البحث:

1 - خوارزمية الزمن الأقل خمولا أولاً (LLF) Least Laxity First Scheduling [11]:

تعمل هذه الخوارزمية على إعطاء أعلى أولوية للمهمة التي تحوي أقل زمن متبقي حيث يعرف الزمن المتبقي بالفرق بين الزمن الباقي من تنفيذ المهمة والزمن الحدي للمهمة. في أثناء تنفيذ مهمة ما باستعمال هذه الخوارزمية، قد تخفف أولويتها إذا وردت مهمة أخرى لها زمن متبقي أقل من زمن هذه المهمة لهذا تعتبر خوارزمية جدولة ديناميكية. نصادف مشكلة في حال وصول مهمتين لهما نفس الزمن المتبقي حيث سيتم تنفيذ قسم من المهمة الأولى ثم تقاطعها الثانية لينفذ قسم منها وبالعكس. لذلك سيحصل العديد من تبديلات السياق أثناء زمن حياة العملية. وتعطى العلاقة الرياضية لهذه الخوارزمية بالشكل التالي:

$$L_i = D_i - e_i \quad (6)$$

حيث e_i هو الزمن الباقي من تنفيذ المهمة ويعطى بالفرق بين الزمن الكلي للمهمة E_i والزمن الذي تم تنفيذه e_i وله العلاقة الرياضية التالية:

$$e_i = E_i - e_i \quad (7)$$

2 - خوارزمية معدل التواتر بالجدولة العامة (RM) Rate Monitoring [12]:

تعمل على جدولة المهام وفقاً لأطوال أدوارها P_i ، إذ تعطى المهمة ذات الدور الأقل أولوية عليا، وتتصف بأن أولويات مهماتها ستاتيكية (ثابتة) لا تتغير خلال عمليات التنفيذ وقد تم تطبيقها بطريقة الجدولة العامة.

3 - خوارزمية معدل التواتر بالجدولة الجزئية (P_RM) Partitioning_Rate Monitoring:

هي خوارزمية معدل التواتر نفسها ولكن تم تطبيقها بطريقة الجدولة الجزئية.

4 - خوارزمية الزمن الحرج أولاً عند الخمول الصفري Earliest Deadline First until Zero Laxity

(EDZL) Scheduling [13]:

تعمل على إعطاء الأولوية الأعلى للمهمة التي تملك الزمن الحدي الأقل ولكن إذا وردت مهمة أخرى لها زمن متبقي صفري عندها تعطى الأولوية الأعلى، ولهذا أيضاً تعتبر خوارزمية جدولة ديناميكية.

5 - خوارزمية الأولويات الثابتة (FP) Fixed Priority:

تعطي هذه الخوارزمية لكل مهمة أولوية خاصة وفريدة priority وينفذ المعالج المهمة ذات الأولوية العليا.

3-4 برنامج المحاكاة المستخدم في البحث:

تمت عملية المحاكاة ورصد النتائج باستخدام برنامج حديث في هذا المجال تم العمل به أول مرة عام

2013 يدعى (SimSo) وهو اختصار Simulation of Real-Time Multiprocessor Scheduling with

[14] Overheads.

النتائج والمناقشة:

تمت الدراسة بجدولة عدد متزايد من المهام الدورية وشبه الدورية، المستقلة، القابلة للمقاطعة (n مهمة) على عدد متزايد أيضاً من المعالجات المتجانسة (m معالج) بحيث:

n = 8, 16, 32, 64, 128 مهمة على m = 4, 8, 16, 32, 64 معالج، أي (n=2m) عدد المهام ضعف

عدد المعالجات وذلك لدراسة أثر ازدياد المهام والمعالجات على أداء خوارزمية الجدولة المدروسة. إن اختيار عدد

المهام ضعف عدد المعالجات لا يعني بالضرورة أن يكون لكل معالج مهمتين، لأنه من الممكن أن تكون مهمة ذات

تواتر دوري كبير فتشغل معالج واحداً وتوزع باقي المهام على باقي المعالجات. وقد تم اختيار هذه النسبة في عدة

دراسات مرجعية حديثة في هذا المجال مثل الدراسة [5].

يعمل برنامج المحاكاة على توليد مجموعات المهام الدورية وشبه الدورية المطلوبة بحيث يكون $U_{max}(T) \leq$

1، ويختار الباحث قيمة U_{sum} بحيث تكون $U_{sum}(T) \leq m$. تم اختيار $U_{sum}(T) = m$ لكافة عمليات الجدولة أي

مجموع إنجازيات كافة المهام يساوي الحد الأعلى المسموح به وهو عدد المعالجات. وتم تحديد قيم دور المهام ضمن

المجال [1 100]ms وتسجيل النتائج الظاهرة عبر نافذة مراقبة observation window مجالها بين

[0 1000]ms.

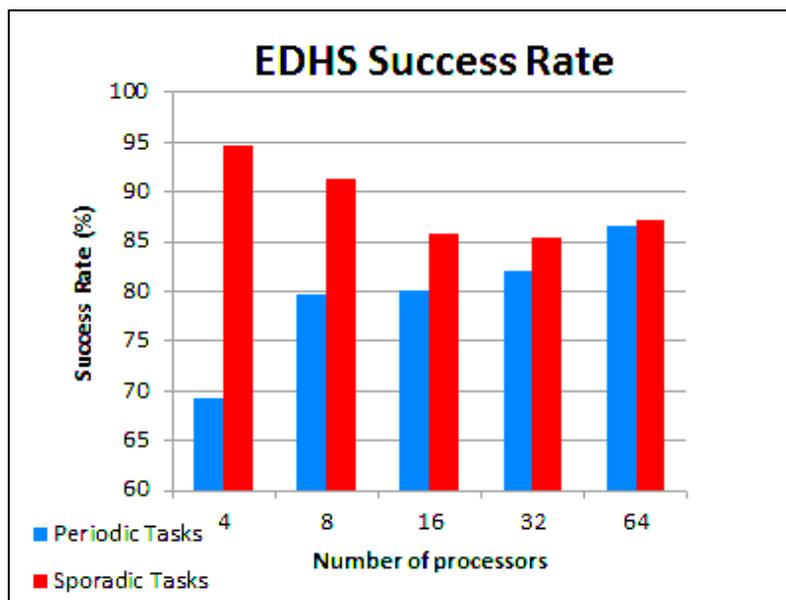
تم تكرار عملية توليد مجموعات المهام الدورية وشبه الدورية (تكرار عملية الجدولة) 100 مرة ثم حساب

المتوسط النهائي في كل مرة نختار فيها عدد المعالجات. (مثلاً: عند توليد 8 مهام على 4 معالجات تم التكرار

100 مرة في كل مرة يتم توليد ثمان مهام جديدة وبنهاية العملية يتم حساب المتوسط وهكذا).

4-1 معدل نجاح الجدولة لخوارزمية EDHS على المهام الدورية وشبه الدورية:

يظهر الشكل (5) نتائج معدل نجاح خوارزمية EDHS عند تطبيقها على المهام الدورية وشبه الدورية.



الشكل (5) معدل نجاح الجدولة لخوارزمية EDHS على المهام الدورية وشبه الدورية

نجد من الشكل (5) أن معدل نجاح خوارزمية EDHS في جدولتها للمهام الدورية قليل نسبياً. ولتحسين معدل نجاح الجدولة يقترح البحث إجراء دراسة تحليلية يقارن فيها أداء خوارزمية EDF التي تستخدمها خوارزمية EDHS في جدولتها للمهام المخصصة على المعالجات مع مجموعة أخرى من الخوارزميات لمعرفة معدل النجاح الذي تحققه هذه الخوارزمية وفيما إذا يوجد خوارزمية أخرى يكون معدل نجاح الجدولة فيها أعلى.

2-4 مقارنة معدل نجاح الجدولة لخوارزمية EDF مع عدة خوارزميات:

يظهر الجدول (1) قائمة بأسماء الخوارزميات المدروسة والمختصرات التابعة لها باللغتين العربية والإنكليزية.

الجدول (1) مختصرات أسماء الخوارزميات

المعنى العربي	المعنى الإنكليزي	رمز اسم الخوارزمية
الزمن الحرج الأقصر و التقسيم للأولوية الأعلى	Earliest Deadline and Highest Priority Split	EDHS
الزمن الحرج الأقصر أولاً	Earliest Deadline First	EDF
الزمن الأقل خملاً أولاً	Least Laxity First	LLF
معدل التواتر	Rate Monitoring	RM
معدل التواتر بالجدولة الجزئية	Partitioning_ Rate Monitoring	P-RM
الزمن الحرج أولاً عند الخمول الصفري	Earliest Deadline First until Zero Laxity	EDZL
الأولويات الثابتة	Fixed Priority	FP

يظهر الجدول (2) مقارنة لنتائج معدل نجاح خوارزمية EDF عند تطبيقها على المهام الدورية مع عدة خوارزميات.

تمت عملية الجدولة باستخدام معالجين ثم 4 ثم 8 ثم 16 وبعدهم مهام ضعف عدد المعالجات.

الجدول (2) معدل نجاح الجدولة لعدة خوارزميات على المهام الدورية

	2	4	8	16
LLF	92.24	98.3	99.82	99.39
EDF	92	95.27	96.67	97.8
RM	84.47	91.56	94.51	96.12
EDZL	87.77	90.28	92.35	94.56
P_RM	85.5	88.58	90.12	90.65
FP	78.07	84.81	89.5	92.32

يظهر الجدول (3) مقارنة لنتائج معدل نجاح خوارزمية EDF عند تطبيقها على المهام شبه الدورية مع عدة خوارزميات.

الجدول (3) معدل نجاح الجدولة لعدة خوارزميات على المهام شبه الدورية

	2	4	8	16
LLF	98.16	99.61	99.94	99.85
EDF	99.78	99.89	99.91	100
RM	93.93	98.09	99.58	99.98
EDZL	99.78	99.89	99.91	100
P_RM	82.18	76.23	74.98	67.67
FP	93.93	98.09	99.58	99.98

نجد من الجدول (2) بأن خوارزمية LLF حققت أعلى معدل نجاح في عملية جدولة على المهام الدورية بين الخوارزميات المدروسة ويفرق أعلى وأفضل من خوارزمية EDF ، أما بالنسبة للمهام شبه الدورية في الجدول (3) نجد بأن جميع الخوارزميات قد ارتفع معدل نجاح الجدولة فيها بشكل كبير وملحوظ، ويعود السبب في ذلك إلى أن توارد المهام شبه الدورية يبقى أقل من المهام الدورية مما يخفف الضغط على المعالجات وبالتالي ارتفاع نسبة نجاح الجدولة. نستنتج من الجدولين السابقين بأنه في حال استبدلنا خوارزمية EDF التي تستخدمها خوارزمية EDHS في جدولة المهام المخصصة بخوارزمية LLF قد تؤدي إلى نتائج أفضل في جدولتها للمهام الدورية بدون أن نقلل من أداء خوارزمية EDHS ونسبة نجاح الجدولة فيها على المهام شبه الدورية.

يقترح البحث طريقة جديدة للتخلص من المشكلة التي تعاني منها خوارزمية LLF عند وصول مهمتين لهما نفس الزمن المتبقي، وهي أن تعطى الأولوية الأعلى للمهمة ذات معدل الاستخدام الأكبر ($U_i = e_i / p_i$) أي المهمة ذات الدور الأقل لأنه كلما كان دور المهمة صغير كلما كان تكرار تنفيذها أكبر على المعالجات فمثلاً: المهمة ذات الدور المساوي 2 ميلي ثانية تتكرر أكثر من المهمة ذات الدور 10ميلي ثانية وكل تكرار للمهمة يرافقه زمن حدي. وبالتالي إذا أعطينا الأولوية العليا للمهمة ذات معدل الاستخدام الأكبر نكون قد ساهمنا في حل مشكلتي مهمتين لهما نفس الزمن المتبقي و بحيث تم التقليل من عدد تجاوزات المهام للزمن الحدي الخاص بها.

4-2-1 الاستنتاج الرياضي لمعادلة الخوارزمية الجديدة المقترحة:

نجد من العلاقة الرياضية رقم (5) للخوارزمية EDHS أنها متعلقة بالزمن الحدي الحرج D_i وذلك بسبب استخدام خوارزمية EDF في جدولة المهام المخصصة على المعالج.

$$c'_s \leq \frac{D_i}{F+1} \left(1 - \sum_{Tk \in T} \frac{c_k}{p_k} \right)$$

وبالتالي عند استخدام خوارزمية LLF يجب استبدال D_i بالعلاقة رقم (6) للخوارزمية LLF فتصبح العلاقة الجديدة بالشكل التالي:

$$c'_s \leq \frac{D_i - e_i}{F+1} \left(\sum_{Tk \in T} \frac{c_k}{p_k} \right) \quad (8)$$

وباستبدال e_i بالعلاقة الرياضية رقم (7) $e_i = E_i - e_i$

مع ملاحظة أن الزمن الكلي لتنفيذ المهمة E_i متعلق بزمن الانتظار الوسطي للمهام بالعلاقة التالية:

$$(Waiting\ Time)W_i = R_i - E_i \quad (9)$$

حيث R_i هو زمن الاستجابة للمهمة (Response Time) وهو الفرق بين لحظة بداية تنفيذ المهمة ولحظة نهاية تنفيذها.

وبالتالي نستنتج أن :

$$E_i = R_i - W_i$$

وباستبدال E_i بالعلاقة رقم (7) نحصل على العلاقة التالية :

$$e_i = (R_i - W_i) - e_i \quad (10)$$

وباستبدال العلاقة رقم (10) ضمن العلاقة رقم (8) نحصل على العلاقة التالية :

$$c'_s \leq \frac{D_i - ((R_i - W_i) - e_i)}{F+1} \left(\sum_{Tk \in T} \frac{c_k}{p_k} \right) \quad (11)$$

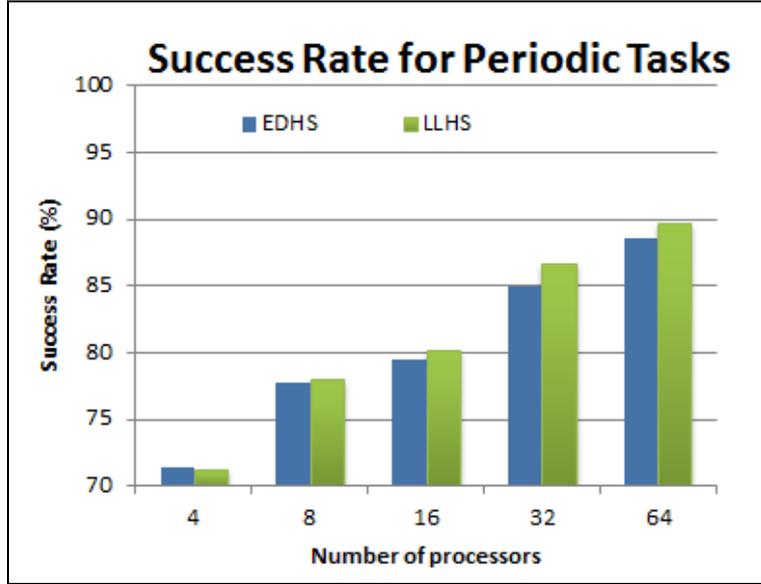
وهي العلاقة الرياضية للخوارزمية الجديدة ونلاحظ أنها متعلقة بشكل واضح بزمن الانتظار الوسطي للمهام.

3-4 نتائج استخدام خوارزمية LLF لجدولة المهام المخصصة في خوارزمية EDHS:

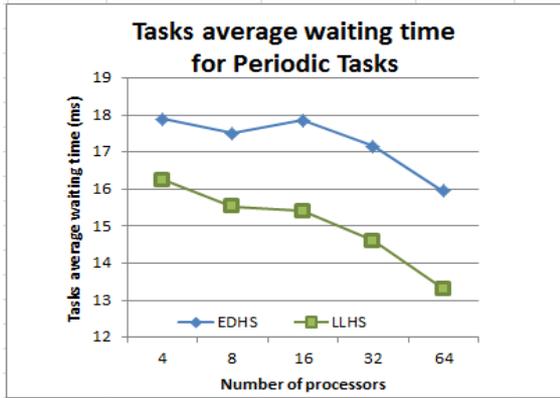
تمت تسمية الخوارزمية الجديدة المعدلة بخوارزمية LLHS وذلك لتمييزها عن الخوارزمية الأصلية.

1-3-4 مقارنة لأداء عمل خوارزميتي EDHS و LLHS على المهام الدورية:

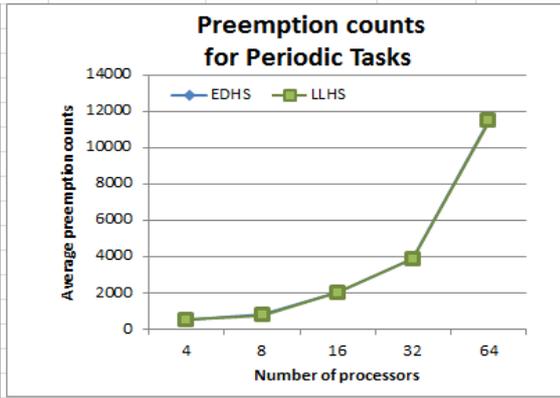
يبين الشكل (6) نتائج معدل نجاح الجدولة عند تطبيق خوارزمية EDHS وخوارزمية LLHS على المهام الدورية.



الشكل (6) مقارنة لمعدل نجاح الجدولة لخوارزميتي EDHS و LLHS على المهام الدورية

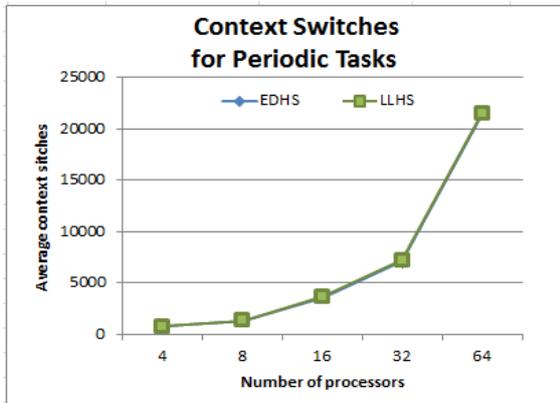


الشكل (7 - ب) متوسط زمن الانتظار للمهام

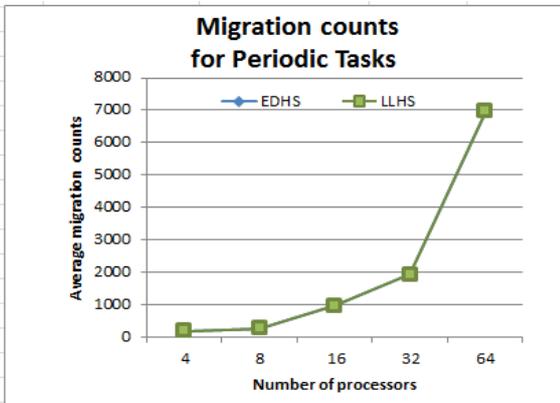


الشكل (7 - أ) عدد المقاطعات

الشكل (7) مقارنة لعدد المقاطعات ومتوسط زمن الانتظار للمهام بين خوارزميتي EDHS و LLHS على المهام الدورية



الشكل (8 - ب) تبديلات السياق

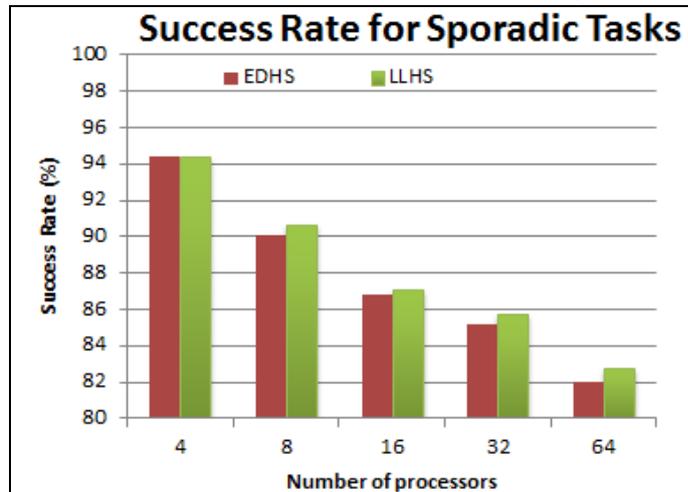


الشكل (8 - أ) عدد الهجرات

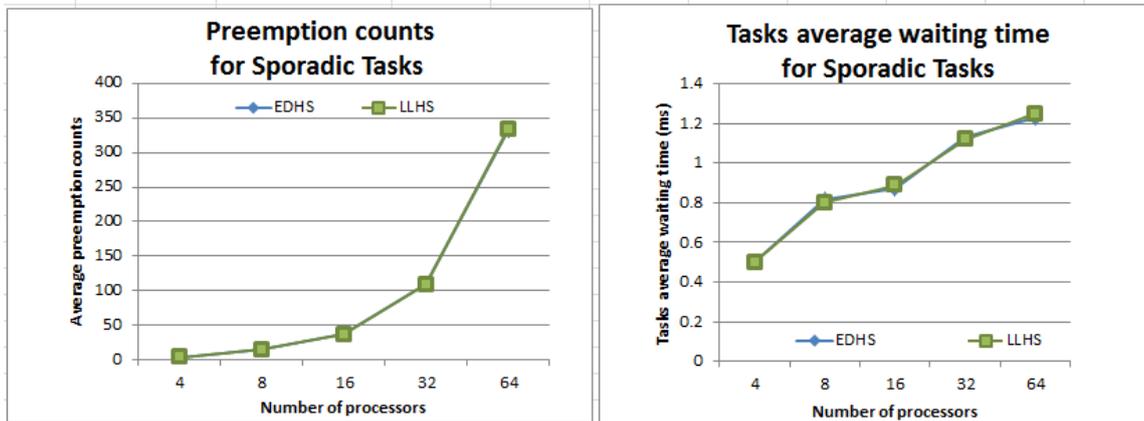
الشكل (8) مقارنة لعدد الهجرات وتبديلات السياق بين خوارزميتي EDHS و LLHS على المهام الدورية

2-3-4 مقارنة لأداء عمل خوارزميتي EDHS و LLHS على المهام شبه الدورية:

يبين الشكل (9) نتائج معدل نجاح الجدولة لخوارزمية EDHS وخوارزمية LLHS على المهام شبه الدورية.



الشكل (9) مقارنة لمعدل نجاح الجدولة لخوارزميتي EDHS و LLHS على المهام شبه الدورية

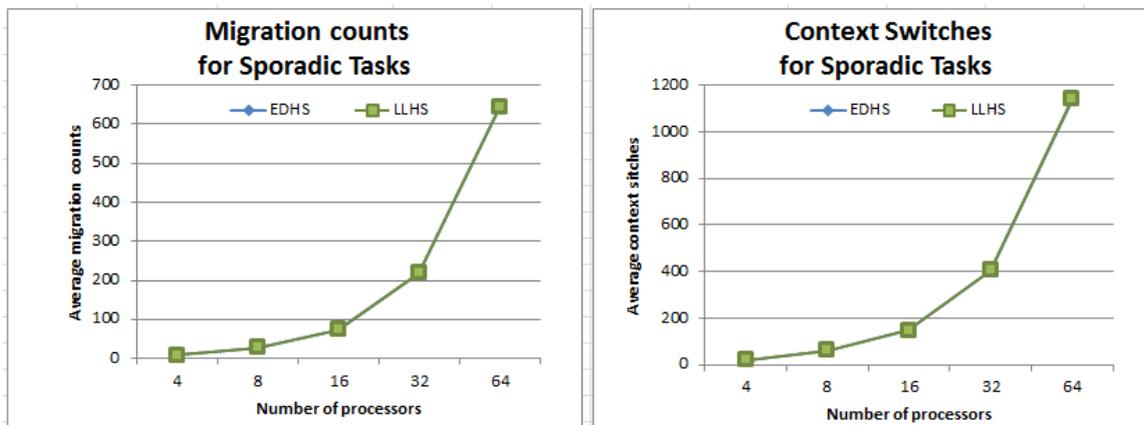


الشكل (10-ب) عدد المقاطعات

الشكل (10-أ) متوسط زمن الانتظار للمهام

الشكل (10) مقارنة لعدد المقاطعات ومتوسط زمن الانتظار للمهام لخوارزميتي EDHS و LLHS على المهام شبه

الدورية



الشكل (11-ب) عدد الهجرات

الشكل (11-أ) تبديلات السياق

الشكل (11) مقارنة لعدد الهجرات وتبديلات السياق لخوارزميتي EDHS و LLHS على المهام شبه الدورية

الاستنتاجات والتوصيات :

تظهر نتائج الدراسة التحليلية لهذا البحث بأن التعديل المقترح على خوارزمية EDHS (أي خوارزمية LLHS) قد أعطى نتائج أفضل من ناحية معدل نجاح الجدولة على المهام الدورية (الشكل (6)) وعلى المهام شبه الدورية (الشكل (9)) وقد استطاع أيضاً التقليل بشكل واضح وكبير في الزمن الوسطي لانتظار المهام الدورية في الأرتال (الشكل (7-ب)) ولم يظهر هذا التحسين في زمن الانتظار للمهام شبه الدورية. يعود السبب في ذلك أن مقدار الزمن المنتظر في المهام الدورية أعلى بكثير منه في المهام شبه الدورية، حيث تظهر النتائج أن مجال زمن الانتظار المهام الدورية [16-18]ms لخوارزمية EDHS أما بالنسبة لخوارزمية LLHS يصبح المجال أقل بشكل كبير [13-16]ms.

أما بالنسبة للمهام شبه الدورية تظهر النتائج أن مجال زمن الانتظار للمهام [0.5 1.2]ms لكلا الخوارزميتين. أما بالنسبة لباقي البارامترات المدروسة مثل عدد المقاطعات والهجرات وعدد تبديلات السياق وكانت النتائج متطابقة لكلا الخوارزميتين أي أن التحسين الذي طرأ على معدل نجاح الجدولة لنوعي المهام الدورية وشبه الدورية وعلى زمن الانتظار الوسطي للمهام الدورية لم يقلل من أي من باقي البارامترات المدروسة. إن استبدال عملية جدولة المهام بحسب الزمن الحرج الأقرب أولاً (EDF) باستخدام الجدولة حسب الزمن المتبقي (LLF) بإضافة التعديل في حال تساوي الزمن المتبقي لمهمتين، قد أعطى مرونة أكبر للمجدول باختيار المهام لتنفيذها وبالتالي قلل من زمن انتظار المهام في الأرتال وهذا التقليل الذي ساهم برفع معدل نجاح الجدولة للخوارزمية المقترحة.

التوصيات:

إن عملية الجدولة للمهام الدورية وشبه الدورية على عدة معالجات في نظم الزمن الحقيقي تعتبر من أصعب الدراسات في الزمن الحقيقي، حيث أن مقدار العمل البحثي المخصص لتحليل وتطوير الخوارزميات المتوفرة يشهد على صحة ذلك.

يقترح للدراسات المستقبلية:

- 1 تحليل أداء خوارزميات أخرى ومقارنة النتائج مع الخوارزميات المدروسة.
- 2 تطبيق الخوارزمية في نظام زمن حقيقي يحتوي مهام مختلطة بين دورية وشبه دورية ولادورية.
- 3 تغيير قيمة الإنجازية الكلية لكافة المهام $(U_{sum}(T) \leq m)$ لتصبح أقل من الحد الأقصى المسموح به الذي اعتمده البحث وهو $U_{sum}(T) = m$.
- 4 دراسة نتائج تغيير نسبة عدد المهام n إلى عدد المعالجات m لتصبح أكبر من $(n=2m)$ ومقارنتها مع نتائج هذا البحث.

المراجع:

- [1] S. Baruah and J. Goossens, "Scheduling Real-time Tasks: Algorithms and Complexity," *Computer and Information Science Series*, vol. 28, p. 38, 2004.
- [2] J. A. Stankovic and K. Ramamritham, "Tutorial Hard Real-Time Systems", IEEE Computer Society Press, 1998.
- [3] M. Arezou and A. G. Selim, "Scheduling Algorithms for Real-Time Systems", School of Computing, Canada, 2005.
- [4] MULLER D, "Schedulability Tests for Real-Time Uni and Multiprocessor Systems". Habilitation, Germany, 2013
- [5] Alhussian.H, Zakari.N, Hussin.F.A, "An Efficient Real-Time Multiprocessor Scheduling Algorithm", *Journal of Convergence Information Technology (JCIT)* Volume9, Number2, March 2014.
- [6] Bertogna.M, Cirinei.M, Lipari.G, "Schedulability analysis of global scheduling algorithms on multiprocessor platforms", *IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS*, VOL. X, NO. X, JUNE 2008.
- [7] SANDHU A., "Performance Comparison of RTS Scheduling Algorithms", *International Journal of Computer Science and Technology*, vol(2), 391-396, 2011.
- [8] ALIYU S.; ABDULRAHIM A.; MUSTAPHA A.; ABDULLAHI S., "An Additional Improvement in Round Robin (AAIRR) CPU Scheduling Algorithm", *International Journal of Advanced Research in Computer Science and Software Engineering*, vol(4), 601-610, 2014.
- [9] S. Kato and N. Yamasaki, "Semi-partitioning technique for multiprocessor real-time scheduling," in Proc. of the WIP Session of the 29th Real-Time Systems Symposium (RTSS), IEEE Computer Society, 2008.
- [10] J. A. Stankovic, M. Spuri, K. Ramamritham, and G. C. Buttazzo, "Deadline Scheduling for Real-Time Systems, EDF and related algorithms", Kluwer Academia Publishers, 1998.
- [11] Lee.J, Easwaran.A, Shin.I, "Laxity dynamics and LLF schedulability analysis on multiprocessor platforms", Springer Science Business Media, LLC 2012.
- [12] NAIK R.; MANTHALKAR R., "Instantaneous Utilization Based Scheduling Algorithms for Real Time Systems". *International Journal of Computer Science and Information Technologies*, vol(2),2011.
- [13] PARK.M, HAN.S, KIM.H, "Comparison of Deadline-Based Scheduling Algorithms for Periodic Real-Time Tasks on Multiprocessor", *IEICE TRANS. INF. & SYST.*, VOL.E88-D, NO.3 MARCH 2005.
- [14] Cheramy.M, Deplanche.A and Hladik.P, "Simulation of Real-Time Multiprocessor Scheduling with Overheads", published in SIMULTECH, Reykjavik: Islande, 2013.