A Model Driven-Based Approach For Global Scheduling of Real-Time Embedded Systems

Amina Magdich, Yessine Hadj Kacem, Adel Mahfoudhi and Mohamed Abid
Computer and Embedded System (CES) laboratory
National Engineering School of Sfax (ENIS), Tunisia

Abstract—**The choice of the appropriate scheduling approach and algorithm for a Real-Time Embedded Systems (RTES) is a challenging step, which requires a vast knowledge and expertise about the scheduling field. Some approaches were proposed to support automatic choice of scheduling algorithms, but there are few studied using high-level scheduling approaches and supporting multiprocessor scheduling allowing task migration. The aim of this study is to use high-level techniques to guide designers while choosing the appropriate scheduling algorithm for a studied system while supporting task migration. In this context, this paper proposes a model-based approach for an automatic choice of scheduling algorithm while supporting the global scheduling approach, which supports task migrations.**

Keywords-Automatic scheduling; MDE; UML/MARTE; RTES; Global Scheduling.

I. INTRODUCTION

The ever growing complexity in Real-Time applications requires the utilization of more powerful resources to implement the various functions that meet users' requirements. Such increasing complexity needs to be managed properly while respecting the system requirements regarding performance, power, cost and time-to-market.

In this context, the main key issue in Real-Time development that must be addressed carefully is the scheduling step. In fact, various research works proposed approaches to support Real-Time Systems scheduling [2,5,16]. Nevertheless, no attention has been given to the use of high-level methodologies that overcome the complexity of the scheduling step mainly while considering multiprocessor architectures.

With regard to this issue, some research works used high-level methodologies to deal with RTES scheduling [9,20]. However, along with the variety of scheduling algorithms, the use of high-level methodologies is not sufficient to overcome the ever growing complexity of RTES scheduling and help designers to properly schedule their studied systems. In fact, there is still a need to assist designers during the scheduling algorithm choice. With this regard, much research work proposed methodologies to guide designers while choosing the appropriate scheduling algorithm [15,21].

These research studies addressed properly the automatic scheduling of multiprocessor systems to assist designers during the design process and avoid failure risks. However, only the partitioned scheduling approach that prohibits task migration was supported. Moreover, no attention was given to the use of high-level methods that facilitate the modeling step and support the whole life cycle of systems development. Hence, an attempt to use Model-Driven Engineering (MDE) [19] while supporting automatic scheduling was addressed in various research studies [4,6]. Authors of these papers used also design patterns to support the high-level modeling and automatic scheduling.

The proposed approaches [4,6] seem to be an adequate solution that uses high-level techniques to help designers overcoming the scheduling step complexity. However, only the partitioned scheduling was supported. Moreover, no attention was given to the optimization of tasks allocation on processors.

With regard to these issues, we propose a model-based approach for automatic choice of scheduling algorithm regarding the global scheduling approach that supports tasks migration. While allocating tasks on processors, an attempt to maximize Central Processing Units (CPU) occupation and minimize the energy consumption is considered while using Tabu search [17].

The remainder of this paper starts with section 2 in which we define the various concepts used in this proposal. Section 3 highlights the proposed approach with its major steps. Experimental results are given in section 4 to validate our proposal. We discuss in section 5 the advantages of our approach compared to existing ones. Finally, section 6 gives a summary of the paper.

II. TECHNICAL BACKGROUND

Before describing the proposed approach we used in conducting this research, we first present the scheduling theory and the used techniques and languages.

A. *Scheduling theory:*

The scheduling theory [10] represents a solution to deal with the allocation of tasks on the available computing resources while respecting temporal requirements. Two major types of scheduling strategies are documented in the literature; the monoprocessor and the multiprocessor scheduling.

In fact, RTES are subject of a lot of constraints that necessitate the use of a multiprocessor architecture, which offers powerful execution hosts. In this context, we provide a brief overview of the three commonly used scheduling approaches for RTES multiprocessor scheduling [3]; the partitioned, the semi-partitioned and the global approaches.

- **The partitioned scheduling approach:**

Given a set of tasks and execution hosts, the partitioned scheduling approach assigns each task to be allocated on only one processor during its execution periods.
Since this approach prevents task migrations, some execution hosts may be free for a while and delays may be caused. Consequently, the rate of CPU utilization may be not optimal.

- **The global scheduling approach:**

The goal of the global scheduling is to reach optimal CPU use by allowing full tasks migrations. What is worthwhile to note is that the full migration of tasks may be costly in terms of context switching.

- **The semi-partitioned scheduling approach:**

The semi-partitioned scheduling enables reaching CPU optimality by enabling restricted task migrations. It offers a trade between context switching and CPU occupation.

B. *Model-Driven Engineering for RTES co-design:*

The model-driven engineering [19] is a paradigm that enables several significant improvements in the development of complex systems by focusing on more abstract concern rather than a classical programming. Due to its importance, MDE is used in different fields such as early analysis of RTES [7], scheduling [18], Data Warehouse systems [8], etc.

MDE is based on three major concepts; meta-modeling, modeling and models transformations. Indeed, for modeling/meta-modeling systems, various norms have been documented in the literature such as the Meta-Object Facilities (MOF), Unified Modeling Language (UML), XML Metadata Interchange (XMI), Object Constraints Language (OCL), etc.

In our proposal, we adopt the UML and specifically the UML profile for Modeling and Analysis of Real-Time Embedded systems (MARTE) profile to model the studied systems.

While modeling systems, the designer should respect the conformity of a model to its meta-model. For models transformation, various languages may be used to ensure model to model or model to text transformations.

A model transformation is based on a set of rules that allow the transformation of a graphical or textual model to another graphical or textual model (figure 1).

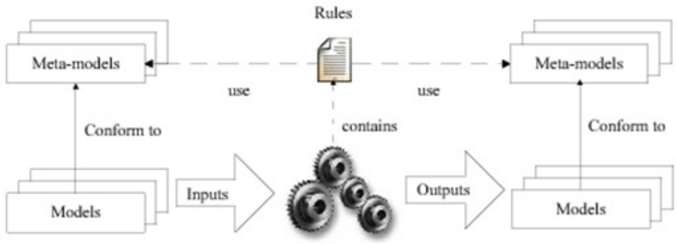


Figure 1. MDE concepts

III. The proposed approach for automatic scheduling:

The proposed model-based approach (Figure 2) for global scheduling is based on four steps; system abstraction, system properties extraction, automatic scheduling algorithm choice and system scheduling and finally schedulability analysis.

The System Abstraction step consists on modeling the studied system using UML/MARTE profile and specifically GRM sub-profile. The built model encloses the studied system properties which will be used to establish the automatic choice of scheduling algorithms.

To extract these properties (System Properties Extraction step), a model to text transformation must be done using an ACCELEO template. Consequently, every class in the GRM model will be automatically transformed to a JAVA class that contains the corresponding properties.

Based on the extracted system properties, an appropriate scheduling algorithm which may be applied on the studied system is selected by our tool.

Using the selected scheduling algorithm, tasks are allocated/scheduled on the available processors while maximizing CPU occupation and minimizing energy consumption. This step is performed using Tabu search.

After the scheduling of tasks, the designer has to establish an early schedulability analysis to check whether tasks meet their deadlines. Thus, a dynamic view annotated through MARTE/SAM (Schedulability Analysis Modeling) sub-profile is built for the studied system. This view is transformed through a model to model transformation to Cheddar tool.

In case of non schedulability of the application or the dissatisfaction of the designer about obtained results, a feed-back has to be done to change the selected scheduling algorithm. These steps will be accurately described in next sections.

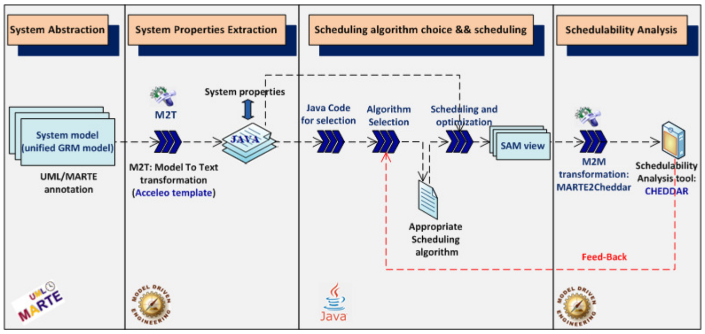

System Abstraction
System Properties Extraction
Scheduling algorithm choice && scheduling
Schedulability Analysis
System model (unified GRM model)
UML/MARTE annotation
M2T
System properties
JAVA
M2T: Model To Text transformation (Acceleo template)
Java Code for selection
Algorithm Selection
Scheduling and optimization
SAM view
Appropriate Scheduling algorithm
M2M transformation: MARTE2Cheddar
Schedulability Analysis tool: CHEDDAR
Feed-Back
UML MARTE
Java

Figure 2. An MDE-based approach for global scheduling

A. *System Abstraction:*

This step represents the modeling of the studied system by specifying its properties while using a modeling language. In our proposal, we used UML/MARTE to establish a high-level modeling.

Originally, MARTE supported only the modeling of the partitioned scheduling. Consequently, we proposed extensions [11,12,13] for MARTE to support both semi-partitioned and global scheduling. These extensions are used in our proposal to model systems for global scheduling.

The use of MARTE supports the model reuse concept. Hence, we proposed a generic model (Figure 3) that can be used to model different systems having similar architectures.

This model is composed of Software (SW) and Hardware (HW) parts. The SW part, modeled through the package APPLICATION, encloses the different tasks of the system.

This application is allocated on a target architecture modeled through the package ARCHITECTURE which contains the hardware components of the system.

Each package is composed of classes annotated through the appropriate MARTE stereotypes. A class modeling mutual exclusions is used to control the concurrent access to shared resources.

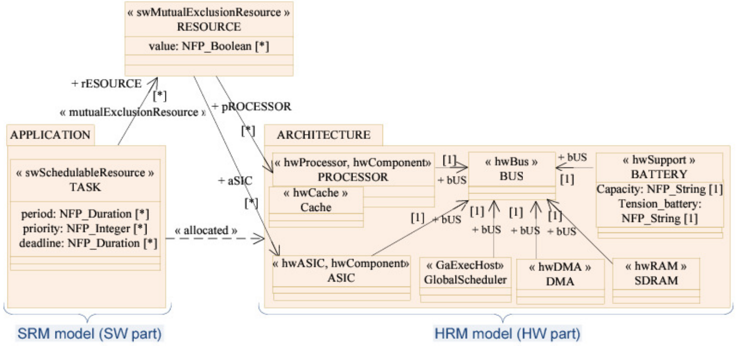


Figure 3. UML/MARTE model for global scheduling (GRM view)

For a studied system, a model instance that imports the generic model and encloses the system properties has to be built.

B. System Properties Extraction:

The studied system properties that are used for scheduling algorithm choice have to be automatically extracted using MDE concepts (transformation concepts). In this context, different transformation techniques are documented in the literature such as M2T (Model To Text) and M2M (Model To Model).

The M2T type represents a transformation from model to text (code). It is founded on existing parsers (such as XML/XSLT) which are based on programming languages (JAVA) or mapping templates (JET/ACCELEO).

Regarding the M2M technique, it uses mapping languages (ATL or Kermeta) to translate a model to another model while adopting a syntactic and semantic analysis.

To extract the studied system properties, we used a M2T transformation based on ACCELEO allowing the generation of java code from the given system model.

Each class in the MARTE model will be transformed to a JAVA class with the corresponding properties that will be used as input to the proposed tool to perform the automatic choice of the scheduling algorithm.

C. Scheduling Algorithm Choice and System Scheduling

Each scheduling algorithm supports specified task types. Considering this classification, we have prepared a list of scheduling algorithms.

Based on this list coupled with the studied system properties and mainly the software properties, our Interactive Tool for Automatic Global Scheduling (IT-AGS) proposes a scheduling algorithm that may be applied on the studied system.

Our tool shows a flexibility regarding the scheduling algorithm choice. In case of dissatisfaction about the scheduling results, an expert designer may intervene to select another scheduling algorithm.

While allocating n tasks on the available m processors using the selected scheduling algorithm, an optimization method (Tabu search [17]) and an objective function (equation1) are used for optimizing the tasks placement in terms of energy consumption and worst execution time.

With this regard, we implemented algorithm1 using JAVA.

$$MinZ = Cij \times Xij + Dij \times Xij \qquad (1)$$

$$\forall\, i,j \in \{i,\ldots,n\} \times \{j,\ldots,m\}$$

With:

- Cij is the allocation cost of Ti on $CPUj$
- Dij is a characteristic of task Ti if it is affected to $CPUj$: we choose the Worst Execution Time needed for a task to be executed on a processor $CPUj$
- $Xij = \begin{cases} 1 & \text{if the task will be allocated on } CPUj \\ 0 & \text{else} \end{cases}$

The allocation cost Cij of $task_i$ on a processor $CPUj$ is computed using equation 2.

$$cost = (PwCost - minPw)/minPw \qquad (2)$$

With:

- $PwCost$ is the consumption cost of the system.
- $minPw$ is the minimum cost of the system implantation (after tasks allocation on processors) in terms of consumption. It is computed through the consumption cost of tasks allocation.
- The ratio $(PwCost\text{-}minPw)/$ minPw specifies the variation of the consumption $cost$ after task scheduling.

Actually, it is not allowed to sum a variation ($cost\ Cij$) and an execution time (Dij). Consequently, there is a need to normalize the members of the proposed objective function in order to stay in the same range of values ([0; 1]).

Let x and y two strictly positive real such as $x{=}Cij$ and $y{=}Dij$. Thus, the objective function will be as specified in equation 3.

$$minZ = f(x) + g(y) \qquad (3)$$

f and g two strictly increasing functions given that the maximum values that can take x and y are respectively $Xmax$ and $Ymax$ such as :

$$\begin{cases} f(Xmax) = g(Ymax) = 1 \\ f(x) = x/Xmax \\ g(y) = y/Ymax \end{cases}$$

Consequently, the members of the objective function will be in the same range of values [0;1].

To implement the objective function using Tabu search (Algorithm1), we defined a neighborhood that represents the possible implantations of tasks on processors such as:
$N_T(s) = \{init_{solution} \epsilon\ NT(s) with\ init_{solution} \notin Tabu\_matrix\ \ or\ F(init\_solution)\ < F(opt\_solution)\}$.

It is to be noted that current_solution is the current solution, init_solution is the initial solution and opt_solution is the best contoured solution.

Algorithm1: Optimal Allocation Research

```
repeat
  Generate n samples without Tabu movement
  Choose an arbitrary initial solution that minimizes the
  neighborhood  init_solution $\epsilon$ $N_T(s)$
  opt_solution = init_solution
  if $F(current\_solution) < F(opt\_solution)$ then
      opt_solution = current_solution
  end if
  update current movement in Tabu_matrix
until the satisfaction of the termination criteria
```

D. *Schedulability Analysis:*

Once the scheduling algorithm was selected and tasks were scheduled, the designer has to check whether its application leads to temporal constraints respect.

To fulfill this step, a M2M transformation has to be performed by transforming the dynamic view of the studied system annotated through MARTE/SAM to the model of the chosen schedulability analysis tool.

What is worthwhile to note is that the mapping concept promotes the separation between the development flow steps as it fosters the independence of the development flow towards the used tools. With this regard, any schedulability analysis tool that supports global scheduling may be used in our proposal for schedulability analysis.

In our approach, the Cheddar tool was used to validate the temporal behavior of the studied system scheduling.

What is worthwhile to note is that originally Cheddar did not support the global scheduling. An extension of Cheddar was established [18] to support the global scheduling.

The transformation from a MARTE model, mainly a SAM model, to the Cheddar tool model is based on a set of concepts and rules.

Table I shows some concepts used to perform the M2M transformation from SAM view to Cheddar tool for schedulability analysis.

Specifically, these concepts are used to establish the required rules for M2M transformation.
An example of transformation rule is given through R1.

R1: Every element stereotyped by <<GRM:Scheduler>> is transformed to the <<scheduler>>

	MARTE concepts	**Cheddar concepts**
Classes annotations	GQAM:gaExecHost	Processor
	SRM:swSchedulableResource	Task
	GRM:Scheduler	Scheduler
Properties annotations	GRM:Scheduler:IsPreemptible	isPreemptible
	SRM:swConcurrentResource: Period	Period

TABLE I. MARTE2CHEDDAR TRANSFORMATION CONCEPTS

IV. Case study:

To evaluate the proposed tool for automatic global scheduling, we study a pedagogic case study [14]. The properties of the studied system are described in table II.

Task Name	**Ri**	**Ci**	**Di**	**Pi**
T1	0	6	20	20
T2	0	6	15	15
T3	0	13	40	40
T4	0	15	40	40
T5	0	6	30	30
T6	0	12	20	20
T7	0	8	20	20
T8	0	10	25	25
T9	0	6	10	10
T10	0	8	20	20

TABLE II. TASKS PARAMETERS

The modeling of the studied system is described in Figure 4 through a class diagram annotated using SRM and HRM sub-profiles.

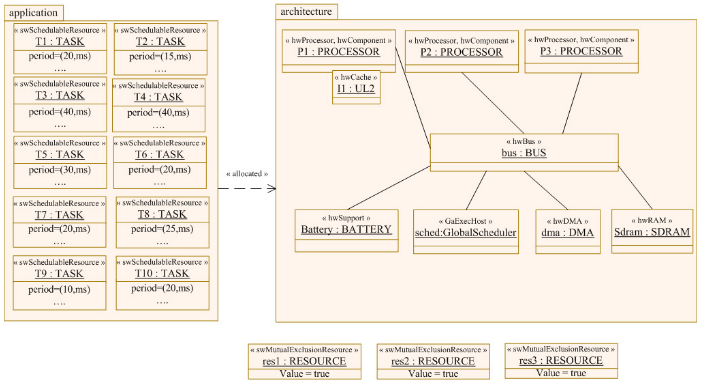
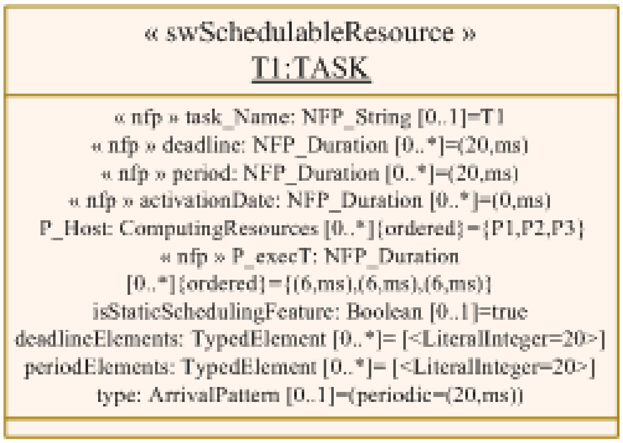


Figure 4. A static modeling for the studied system

This model encloses SW and HW parts. The SW part encloses ten classes annotated <<swSchedulableResource>> to model tasks. The HW part contains classes modeling processors and other hardware components. Mutual exclusion resources are used to manage the access to shared resources. They are modeled using classes annotated <<swMutualExclusionResources>>.

These classes are not connected with the processors using dependencies since they may be specified in the attributes of the corresponding stereotype. The filling of the classes' properties and attributes is hidden under MARTE. Thus we give an example (Figure 5) of a class filling.



« swSchedulableResource »
T1:TASK

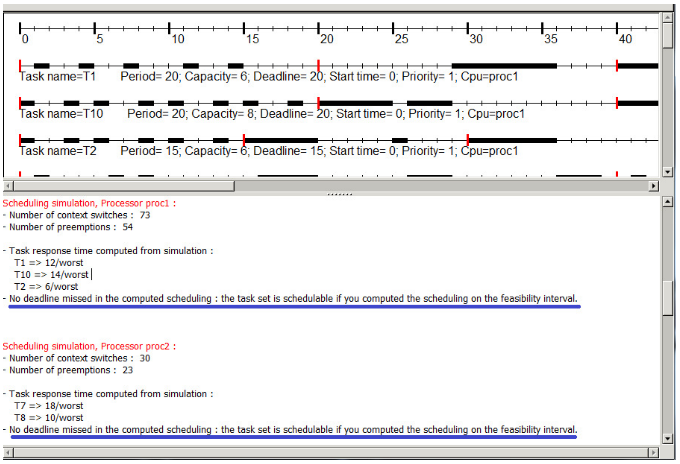
« nfp » task_Name: NFP_String [0..1]=T1
« nfp » deadline: NFP_Duration [0..*]=(20,ms)
« nfp » period: NFP_Duration [0..*]=(20,ms)
« nfp » activationDate: NFP_Duration [0..*]=(0,ms)
P_Host: ComputingResources [0..*]{ordered}={P1,P2,P3}
« nfp » P_execT: NFP_Duration [0..*]{ordered}={(6,ms),(6,ms),(6,ms)}
isStaticSchedulingFeature: Boolean [0..1]=true
deadlineElements: TypedElement [0..*]= [<LiteralInteger=20>]
periodElements: TypedElement [0..*]= [<LiteralInteger=20>]
type: ArrivalPattern [0..1]=(periodic=(20,ms))

Figure 5. A class filling under MARTE

The properties specified in this diagram were extracted using a M2T transformation based on an ACCELEO template.

Based on the system properties, the PFair1 scheduling algorithm was selected to schedule the studied system.

A simulation of the system scheduling was done using Cheddar tool. This simulation (Figure 6) shows that using the selected scheduling algorithm and the proposed allocations of tasks, the system leads to its temporal constraints.



```
Task name=T1      Period= 20; Capacity= 6; Deadline= 20; Start time= 0; Priority= 1; Cpu=proc1
Task name=T10     Period= 20; Capacity= 8; Deadline= 20; Start time= 0; Priority= 1; Cpu=proc1
Task name=T2      Period= 15; Capacity= 6; Deadline= 15; Start time= 0; Priority= 1; Cpu=proc1

Scheduling simulation, Processor proc1 :
- Number of context switches :  73
- Number of preemptions :  54

- Task response time computed from simulation :
   T1 => 12/worst
   T10 => 14/worst
   T2 => 6/worst
- No deadline missed in the computed scheduling : the task set is schedulable if you computed the scheduling on the feasibility interval.


Scheduling simulation, Processor proc2 :
- Number of context switches :  30
- Number of preemptions :  23

- Task response time computed from simulation :
   T7 => 18/worst
   T8 => 10/worst
- No deadline missed in the computed scheduling : the task set is schedulable if you computed the scheduling on the feasibility interval.
```

Figure 6. Schedulability analysis using Cheddar tool

V. Discussion

The use of high-level methodologies for systems development has a major impact on complexity management. In fact, taking benefit from MARTE profile and MDE concepts (M2M transformation) helped us to reduce the development time and separate between the concerns.

High-level methodologies [9,20] were proposed to overcome the complexity of the scheduling step mainly while considering multiprocessor architectures. Nevertheless, no attention was given to the automatic choice of scheduling algorithm to properly schedule systems and avoid failures.

This issue, was solved by the proposal of high-level methodologies [4,6] that support automatic choice of scheduling algorithm. Nevertheless, no attention was given to the optimization of tasks placement on processors. Also, only the partitioned scheduling that prohibits task migration was supported.

- It uses high-level techniques to support automatic choice of scheduling algorithms while considering global scheduling rather than only the partitioned scheduling.
- It uses Tabu search to offer an optimization of tasks placement regarding the global scheduling approach and in the context of automatic scheduling.

VI. Conclusion:

Throughout this paper, we proposed a model driven approach for automatic scheduling regarding the global scheduling approach. To fulfill our proposal, we took benefit from the use of MDE concepts and UML/MARTE profile.

The Tabu search was used to optimize the placement of tasks on processors. Simulation results based on Cheddar tool for schedulability analysis showed that the selected scheduling algorithm schedule the studied system without missing deadlines.

As future work, we seek to apply the proposed methodology in the context of semi-partitioned scheduling.

References

[1] Andreson, J., Holman, P. and Srinivasan, A. Fair Scheduling of Real-time Tasks on Multiprocessors, Handbook of Scheduling Algorithms, Models, and Performance Analysis, 2004.

[2] Mouli, C. and Smriti, A. An (M, K) Model Based Real-Time Scheduling Technique for Security Enhancement. International Journal of Computer Science and Information Secutrity (IJCSIS), vol 13, No 10, October 2015. P10-18.

[3] Dorin, F.; Yomsi, P. M.; Goossens, J. and Richard, P. Semi-Partitioned Hard Real-Time Scheduling with Restricted Migrations upon Identical Multiprocessor Platforms CoRR, 2010.

[4] Fritzsche, R.; Ristig, C. and Siemers, C. An Approach and Design Pattern for Intra-Application Scheduling, Clausthal University of Technology, 2010.

[5] G. Umarani Srikanth, V. Uma Maheswari, A. P. Shanthi, Arul Siromoney ,"Task Scheduling Model",Indian Journal of Science and Technology,2015 Apr, 8(S7), Doi no:10.17485/ijst/2015/v8iS7/63120.

[6] Gaudel, V.; Singhoff, F.; Plantec, A.; Rubini, S.; Dissaux, P. and Legrand, J. An Ada Design Pattern Recognition Tool for AADL Performance Analysis Proceedings of the 2011 Annual International Conference on Special Interest Group on the Ada Programming Language, ACM, 2011, 61-68.

[7] HadjKacem, Y.; Mahfoudhi, A.; Magdich, A.; Karamti, W. and Abid, M. Using MDE and Priority Time Petri Nets for the schedulability analysis of Embedded Systems modeled by UML activity diagrams. The 19th Annual IEEE International Conference and Workshops on the Engineering of Computer Based Systems (ECBS), 2012, pp: 316-323.

[8] Hilal, I., Afifi, N., Hilali,R. and Ouzzif, M. Toward a New Approach for Modeling Dependability of Data Warehouse System. International Journal of Computer Science and Information Secutrity (IJCSIS), vol. 11, no. 6, pp. 47-54, Nov. 2013.

[9] Li, S.; Singhoff, F.; Rubini, S. and Michel, B. Applicability of Real-time Schedulability Analysis on a Software Radio Protocol Proceedings of the 2012 ACM Conference on High Integrity Language Technology, ACM, 2012, 81-94.

[10] Liu, C. L. and Layland, J. W. Scheduling Algorithms for Multiprogramming in a Hard-Real-Time Environment. J. ACM, ACM, 1973, 20, 46-61.

[11] Magdich, A.; Hadj Kacem, Y.; Mahfoudhi, A. and Abid, M. Reducing UML/SAM modeling view size for schedulability analysis. The 1st IEEE international conference on Computer Applications Technology (ICCAT), IEEE, 2013.

[12] Magdich, A.; Kacem, Y. H. ; Mahfoudhi, A., Extending UML/MARTE-GRM for Integrating Tasks Migrations in Class Diagrams. 11th International Conference on Software Engineering Research, Management and Applications SERA (selected papers), Springer, 2013, 496, pp: 73-84.

[13] Magdich, A.; Kacem, Y. H.; Mahfoudhi, A. and Abid, M. A MARTE extension for global scheduling analysis of multiprocessor systems. The 23th IEEE International Symposium on Software Reliability Engineering (ISSRE), IEEE Computer Society, 2012, pp: 371-379.

[14] Muhammad Khurram, B.; Cécile, B. and Michel, A. Two-level Hierarchical Scheduling Algorithm for Real-time Multiprocessor Systems. Journal of Software JSW, 2011, 6, pp: 2308-2320.

[15] Olteanu, A.; Pop, F.; Dobre, C. and Cristea, V. An adaptive scheduling approach in distributed systems IEEE International Conference on Computational Photography (ICCP), IEEE Computer Society, 2010, 435-442.

[16] P. Sivakumar, B. Vinod, R. S. Sandhya Devi, E. R. Jayasakthi Rajkumar,“Real-Time Task Scheduling for Distributed Embedded System using MATLAB Toolboxes”,Indian Journal of Science and Technology,2015 July, 8(15), Doi no:10.17485/ijst/2015/v8i15/55680.

[17] Pérez, E.; Resano, J.; Mozos, D.; Mecha, H. and Septién, J. A Multi-objective Dynamic Cost Function for Codesign Hardware/Software Partitioning, XVIII Design of Circuits and Integrated Systems Conference, DCIS'03, Ciudad Real, 2003.

[18] Rubini, S.; Fotsing, C.; Singhoff, F.; Tran, H. N. and Dissaux, P. Scheduling Analysis from Architectural Models of Embedded Multi-Processor Systems. EWiLi Workshop, 2013.

[19] Schmidt, D. C., Model-Driven Engineering, IEEE Computer, 2006.

[20] Starke, R. and de Oliveira, R. A Heterogeneous Preemptive and Non-preemptive Scheduling Approach for Real-Time Systems on Multiprocessors Critical Embedded Systems (CBSEC), 2012 Second Brazilian Conference on, 2012, 70-75.

[21] Zamfirache, F. and Frincu, M. Automatic Selection of Scheduling Algorithms Based on Classification Models, international Conference on Knowledge Engineering: Principles and Techniques, 2011.