ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 4, April 2014

An Efficient Fast Admission Control Based Optimal Task Scheduling in Heterogeneous Multicore Systems

S.Rajeswari[1], R.Sajitha[2]

Assistant professor, Dept. of ECE, PSN College of Engineering and Technology,Thirunelveli, Tamilnadu, India [1]

PG Student [EST] , Dept. of ECE, PSN College of Engineering and Technology,Thirunelveli, Tamilnadu, India [2]

ABSTRACT: online task scheduling of heterogeneous multicore system-on-a-chip is a complex problem due to priority constraints and non-pre-emptive task execution in the synergistic processor core. To achieve high performance in the midst of increased complexity, multi-core system- on-a-chip is used. This study first proposes an online heterogeneous dual-core scheduling framework for dynamic workloads with real-time constraints. Both the general purpose and synergistic processor core are dedicated to separate schedulers with different allocation properties, and priority constraints among different tasks are deal with the interaction between the two schedulers. This framework is configurable for low priority inversion and high system utilization. This framework is also extended to heterogeneous multicore systems with well-known dispatcher schemas. This paper presents the practicability of the proposed methodology, and also a series of extensive simulations to obtain comparison studies using different workloads and scheduling algorithms.

KEYWORDS: Real-time system, task scheduling, heterogeneous multicore system.

I.INTRODUCTION

In modern real-time system, applications using embedded system are becoming more complex. To maintain the evaluate in the level of increased complexity, the industry has widely used the scheduling algorithm based on heterogeneous multicore system-on-a-chip built from one general purpose processor core and one or more synergistic processor cores (coprocessors)[1].The timing constraints are used for improving the performance evaluation and efficient power supply. Therefore, an important system specification for an Online scheduling algorithm must be fast and predictable. However, the scheduling problems are complicated by priority constraints of tasks and without pre-emptive executions of tasks in the coprocessor.

To overcome the task scheduling problems in heterogeneous multicore systems many illustrations in the existing system using the synchronization protocol mechanism. By managing the coprocessor without preemptive task as a resource yields the consumption of poor system utilization [2].

When the processor is in idle state the management priority inversions leads to execution of task with less Priority. By utilizing the signal processing coprocessors [3] the system consumption is improved due to the extension of existing protocol related to the system and timing constraints. The algorithm specified for the management of resource with the constraints based on priority is used in computer graphic coprocessor [4].

Research based on previous studies has dealt with scheduling algorithm based on homogeneous multi-core system-on-a-chip [6] is used to reduce the problem caused by the management of priority inversion. The rationale between the processor and coprocessor are distinct due to the significant overhead of preemptive task of the coprocessor is not adapted for executing the preemptive task. The cause of the overhead arises from the available registers, various stages in pipeline and cache memory [7].

Motivation for this online scheduler for different multicore-system, implement the ratio between management of priority inversion and improving the system consumption. Initially, this study focuses a scheduling module with high speed controlling admissions leads to the management of online changing tasks in different dual core systems. Hence, it proposes from the configurability between greater system consumption and lesser inversion of priority. Based on the existing system, two different scheduling policies are adapted for the processor and coprocessor according to their varied functions. This proposed framework extends the different multicore systems using suitable scheduling algorithms.

Organization: The rest of this paper is organized as follows. Section 2 narrates the terms our system framework under evaluation. Section 3 provides the rules for online task dispatching under heterogeneous dual-core scheduling algorithm. Section 4 describes schedulability test and configuration of the proposed scheduling module. Section 5 describes the design and impelmentation of heterogeneous multi- core scheduling algorithm. In Section 6, the performance of algorithm is evaluated, and its capabilities are demonstrated. Section 7, conclusion work in this paper.

II.SYSTEM MODEL

For heterogeneous multicore system-on-a-chip processor is activated by the following functions such as signal processing. Assumption of the processor is assigned by the general purpose processor and coprocessor is assigned by the synergistic processor core. We have further assuming the processor and the coprocessor communication is a achieved by using dedicated shared memory and mailbox.Analyis of communication time in worst case scenario can be analysed a part of the execution time in worst case scenario [6].

All heterogeneous multicore systems, comprises the processor is adapted to control the signal management and coprocessor is adapted to computing the data. Hence a task is executed on that system is a group of subtasks with a ordering of tasks in partial manner. Every task in each subtasks ti, j analysed and executed on a signal control management and the worst case scenario computation time of t i,j is known in precedence as c i,j. The execution of subtask a task in the signal control management and computing the data are known as managing control signal subtasks and data computation subtasks

Various issues addressed in this proposed model are as follows.1) Based on priority constraints, how the derivation and bounding of response time for each task should be evaluated,

2) Minimizing the blocking times of emergency tasks in a coprocessor without preemptive tasks should be determined.

3) Performing the schedulability test for a changing workload with less timing and space costs.

III. HETEROGENEOUS DUAL-CORE SCHEDULING (HDS) ALGORITHM

A master-slave relationship is implemented in a different multicore system for the processor and coprocessor. Initially purpose a module used for different dual-core utilized a processor and coprocessor.

The online task scheduling, every new task is verified by admission control. It enhances the dispatch ability of new task Admission control ensures the Schedulability of the new task and other existing tasks. Task will be rejected if a new task does not attain the pass result. The arrival time of subtask is no predictable and it depends on the completion time of the previous subtask. Calculating the response time of subtask in each coprocessor yields the partition between subtasks of any two processors and the time of partition in each task. The Coprocessor Scheduling the pre-emption point to avoid unacceptable blocking times of tasks in the nonpreemptible coprocessor.

Algorithm:

I Assigning Deadline:

-Assignment of a processor density Si, is achieved by each task ti and a bandwidth server size Ui.

-Assignment of deadlines in processor and co-processor subtasks is achieved by the density and the server size of relative task.

-A concept based on bandwidth server is assigned the deadline of each task.

II Rule for scheduling:

-When each task is passed the schedulability test, then each task is accepted and its relative subtask is assigning the deadline alignment.

-Subtasks are dispatched by their local deadlines is achieved by deadline-driven scheduler based on pre-emption.

.Subtasks are dispatched by the allocation of coprocessor in correspondence with the bandwidth related to its server with the points of pre-emption.

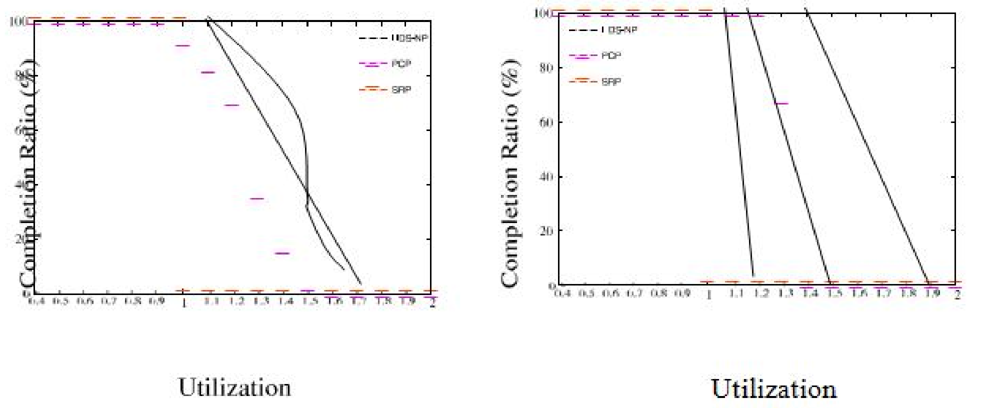
IV.ADMISSION CONTROL

We narrate the performance of the schedulability test of each task set by the admission control based on the task, which is arrived.Assigning the density of a processor and server size of each task is an NP-complete problem and it should be resolved to a bin- packing problem. Prior to the consumption of each core as mentioned, initially define the ratio of the number of available subtask consumption in the processor to the number of subtask consumption in the coprocessor.

The heterogeneous dual-core scheduling algorithm normally used to timing constraints and the worst case computation time. We calculate computation time, there are a lot of results in previous research, and the server size of the coprocessor can be differently adapted by feedback control schemes and probability queuing mechanisms. With the available adapted server size, the heterogeneous dual-core scheduling algorithms and admission control can also be extended for uncertain computation time. dual-core scheduling algorithm the schedulability test is categorized into three major divisions. They are processor consumption test, coprocessor consumption test and deadline based on task test.The scheduling problem on homogeneous multicore algorithm is dispatched as completed no preemption problem. Using separation module, all jobs of a specific task executed on the homogeneous coprocessor.

Global schema illustrates the proposed module of subtask in the coprocessor and is received by its related server. Each subtask is ready with that of instances, the related server sets the fixed value of deadline and push it ready to queue in a global manner.Partition schema illustrates that the server is represented to the coprocessor. The instance related to this coprocessor is inserted into the queue as ready with that of a sporadic activity.Hybrid schema is assumed as various server and are separated to that particular coprocessor among the processor. The details of the separated schematic rule are not illustrated in this figure. The corresponding coprocessor and the remaining servers between coprocessors, subtasks might be split.

V. PERFORMANCE EVALUATION

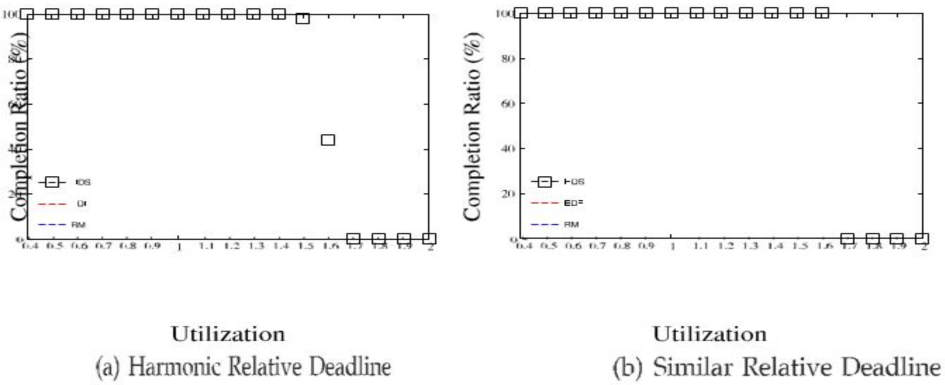


(a) Harmonic Relative Deadline (b) Similar Relative Deadline

Fig. 1Completion ratios of protocols under varied task utilizations.

In the fig 1, Utilization of processor is used for controlling the event input task and output task and the coprocessor is used for decoding calculations. The Communication of processor and the coprocessor is achieved by the quality of service and mailbox. The buffer size in input and output is 16 KB. Each task set has three subtasks. The first and second task performs the input and output control, respectively, and both are executed on the processor. The second task in the subtask performs the decoding computation and is execute on the coprocessor. The computation time of each and every subtask is estimated by exhaustive profiling. In this experiment, the input and output control functions executed on the processor takes 9.9 and 1.5 ms, respectively. The decoding function is executed on the coprocessor in 18 ms. There are two pre-emption points inserted in the decoding function. Decoding process based on JPEG is not an application based on hard real time, but it is the constraint based on the real time process due to quality of service. Utilizing various quality-of-service, arranging the decoding processes through Motion JPEG, for evaluating algorithmic performance. More frames per second(FPS) indicates a good quality of service, and the assumption of dropped frames are not critical. Given a task set with frame rates of 35, 25, 15, and 4 FPS, the task set is not schedulable using SRP and PCP, and the performances of the schedulers (HDS, EDF, and RM) are similar. With frame rates 11, 8, 5, and 5

FPS, using the task set is executed heterogeneous dual-core scheduling algorithm. The experiment results reveal that when the coprocessor is assumed to be fully pre-emptive, HDS improves the frame rate of this task set from 4 to 8 FPS as compared to the scheduler implementations (RM and EDF). Compared to the protocol implementations,HDS improves the frame rate from 4 to 25 FPS.3 Assumption of our frames which are all dropped are not critical. Given a task set with frame rates of 25, 10, 5, and 4 FPS, the task set is not schedulable using SRP and PCP, and the performances of the schedulers (HDS, EDF, and RM) are similar.



(a) Harmonic Relative Deadline

(b) Similar Relative Deadline

Fig. 2 Completion ratios of schedulers under varied task utilizations.

In the fig 2, With frame rates 25, 10, 8, and 5 FPS, the task set is only schedulable using HDS. The experiment results reveal that when the coprocessor is assumed to be fully pre-emptive, HDS improves the enhancing frame rate of subtask set from 4 to 8 FPS as compared to process is not a hard real-time application, the real-time constraint of the process is due to the quality-of-service requirement in the system. Assumption of our frames which are all dropped are not critical, based on the performance evaluation. Given a task set with frame rates of 25, 10, 5, and 4 FPS, the task set is not schedulable using SRP and PCP, and the performances of the schedulers.System one processor and multiple coprocessors, because most heterogeneous multi- core systems are equipped with multiple coprocessors to accelerate task executions. We implemented Heterogeneous dual-core scheduling algorithm based on first fit, worst fit, Global, and Hybrid dispatchers on a system with one processor and three coprocessors and then first fit and worst fit dispatchers partition tasks by the maximum needed server. The global dispatcher always dispatches the job with the shortest deadline assigned by Heterogeneous dual-core scheduling algorithm into the coprocessor. The hybrid dispatcher first assigns tasks to the particular processors according to the first fit.

VI.CONCLUSION

The contributed work of this paper explains the real-time task, dispatching errors in different coprocessors and deals with the timing and priority constraints without pre-emptive execution of tasks. Initially propose the schedulability test by the acceptance of changing the value of each jobs. An inversion pre-emption point based solution is depicted as the configuration of overhead mechanisms and bounding capability in the coprocessor the extended mechanism of different coprocessor is achieved by using adaptable schedulers.

Results obtained by various experiments are observed by the evaluation of scheduling rules under varying jobs and tasks. Future work related to the online efficiency mechanisms of different coprocessor
and consumption bounding and blocking time different coprocessor. Enhancing various pre-emptive mechanisms will tend to the development of design in mobile system utilization.

REFERENCES

[1] Qualcomm,Inc.,”Snapdragon,”technical report,Qualcomm.http://www.qualcomm/media/documets/snapdragon-s4-processors-system-chip-solutions-new-mobile-age,2011 report.

[2] Paolo Gai, Luca Abeni, Giorgio Buttazzo. Multiprocessor DSP Scheduling in System-on-a-chip Architectures. Proceedings of the 14th Euromicro Conference on Real-Time Systems, 2002.

[3] L. Sha, [3] R. Rajkumar, and J. Lehoczky, "Priority Inheritance Protocols: An Approach to Real-Time Synchronization," IEEE Trans. Computers, vol. 39, no. 9, pp. 1175-1185, Sept. 1990

[4] Bautin, [2] M., Dwarakinath, A., And Chiueh, T. Graphics Engine Resource Management. In Proc. MMCN (2008).

[5] L. Benini, D. Bertozzi, A. Guerri, and M. Milano,"Allocation, Scheduling and Voltage Scaling on Energy Aware MPSoCs," Proc. Conf. Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems, 2006.

[6] K.-Y. Hsieh, Y.-C. Lin, C.-C. Huang and J.-K. Lee, "Enhancing Microkernel Performance on VLIM DSP Processors via Multiset Context Switch," 2006.

[7] L. Steffens, M. Agarwal, and P. Vander Wolf, "Real-Time Analysis for Memory Access in Media Processing Socs: A Practical Approach," Proc. Euromicro Conf. Real-Time Systems, 2008.