Dynamic Task Set Partitioning Based on Balancing Memory Requirements to Reduce Power Consumption

Diana Bautista, Julio Sahuquillo, Houcine Hassan, Salvador Petit, José Duato
Universidad Politécnica de Valencia
Department of Computer Engineering
diabaura@upvnet.upv.es, {jsahuqui,husein,spetit,jduato}@disca.upv.es

EXTENDED ABSTRACT

Because of technology advances power consumption has emerged up as an important design issue in modern high-performance microprocessors. As a consequence, research on reducing power consumption has become a hot research topic. Different ways to reduce power consumption consist on using processors that do not implement the most power-hungry microarchitectural mechanisms, attacking hot spots, or reducing consumption in the larger microprocessor components like the cache. Unlike these works which focus on specific parts of the microprocessor, Dynamic Voltage Scaling (DVS) is a technique which applies on the whole microprocessor die. This technique allows the system to work at different frequency/voltage levels. DVS costs in a multicore system can be reduced by sharing the same DVS regulator among the cores (global DVS). In this context, to handle energy efficiently, the workload must be properly balanced among the cores.

In this paper we propose a new partitioning heuristic aimed at increasing the execution overlapping to reduce energy consumption in coarse-grain multithreaded multicore processors working on a global DVS regulator. The proposed algorithm works by distributing hard real-time tasks attending to two criteria while guaranteeing real-time constraints. Tasks are firstly distributed among cores by balancing memory requirements. This criterion is followed until the utilization of any core reaches a given threshold. Then, as all cores must work at the same speed (global DVS), the less loaded core must be selected each time a task is launched.

Energy savings depend on the range of frequency/voltage levels that DVS implements. Experimental results show that the proposed heuristic reduces the energy consumption in almost 3 times with respect to a system with no DVS regulator and applying no heuristic.

Categories and Subject Descriptors

J.7 [**Computer Applications**]: Computer in other systems – *Real-Time.*

ICS'09, June 8–12, 2009, York Town Heights, New York, USA.
ACM 978-1-60558-498-0/09/06.

General Terms

Performance, Design, Algorithms, Experimentation.

Keywords

Power-Aware, Scheduling, Coarse-Grain Multithreaded, Real-Time, Multicore.

1. PROPOSAL

This paper presents a hard real-time power-aware partitioner and scheduler for a coarse-grain multicore processor. The modeled system is composed of the source and sink of real-time tasks, a workload partitioner, a power-aware scheduler, and a multicore processor. Each core is a coarse-grain multithreaded processor and shares a global power-aware scheduler. The schedulers share a global DVS [1] regulator that adjusts the speed and consumption of all the cores in the system to the requirements of the EDF (Earliest Deadline First) [2] schedulers.

Each scheduler selects the minimum speed at which the task set running on its core is schedulable following the EDF algorithm and informs the global DVS control hardware about this requirement. The scheduler also launches to execution the most priority ready tasks in its task set. Regarding EDF, these tasks are those whose deadlines come first earlier. If some thread context is occupied by less priority tasks, preemption is applied.

The speed requirements of these schedulers depend on the workload that they manage, which is partitioned among cores by the workload partitioner as real-time tasks arrive from the source. The partitioner applies an heuristic referred to as *Load-bounded Memory Balancing* (LMB), which distributes complementary tasks among the cores in order to balance memory requirements among them.

The algorithm assigns the most memory-consuming task to the core with least accumulated consumption of memory. Then, it updates the task set to be distributed, as well as the accumulated memory consumption and utilization of the target core. This behavior is maintained whenever the accumulated utilization of the target core after the assignment does not exceed a given theoretical threshold, which is defined as the average utilization of the cores if the workload were completely balanced. If this threshold is exceeded, the task will be assigned instead to the least loaded core. In this way, the algorithm pursues to balance the workload, enabling global speed and consumption reductions.

To evaluate how the proposed heuristic performs it has been compared to the Worst Fit (WF) heuristic, which only addresses workload balancing, and to date, it is the heuristic that provides the best performance [3].

2. EXPERIMENTAL RESULTS

To run the experiments, a wide set of benchmarks from the Real-Time Systems Benchmark repository [4] have been used. Using these benchmarks, different mixes have been designed to experimentally explore the benefits on energy savings. Mixes were designed in order to be composed of a heterogeneous set of benchmarks, and their execution was planned to achieve a system utilization falling in between 30% and 90%.

The power-aware scheduler has been modeled working with different (2, 3, and 5) numbers of frequency/voltage levels. These models will be referred to as 2L, 3L, and 5L respectively. In order to evaluate the performance of the proposed system, two different baseline schedulers have been also considered: a one-level scheduler (1L), which always works at the maximum processor speed, and a naive two-level one (2LN). The 2LN scheduler assumes that the processor is always working at the maximum speed except when there is no task running in any core. In that case, the system frequency is dropped down to the minimum one.

To obtain the total energy consumption for a given mix, it is executed until its hyperperiod. Then, the time that the processor spends at each frequency/voltage level is obtained. Finally, the accumulated energy is calculated by multiplying the previous time by its corresponding energy consumed per cycle.

According to the obtained results for a two-core system shown in Figure 1, two main conclusions can be drawn. First, regarding the partitioning heuristics, the proposed LMB heuristic provides by about 4% less energy than the WF heuristic for the model with five frequencies, which represents about 10% relative energy savings. Second, using a power-aware scheduler with a high number of frequency levels (5 levels) and a fair heuristic strategy, normalized energy can be 65% less than a system working all the time at the maximum speed.

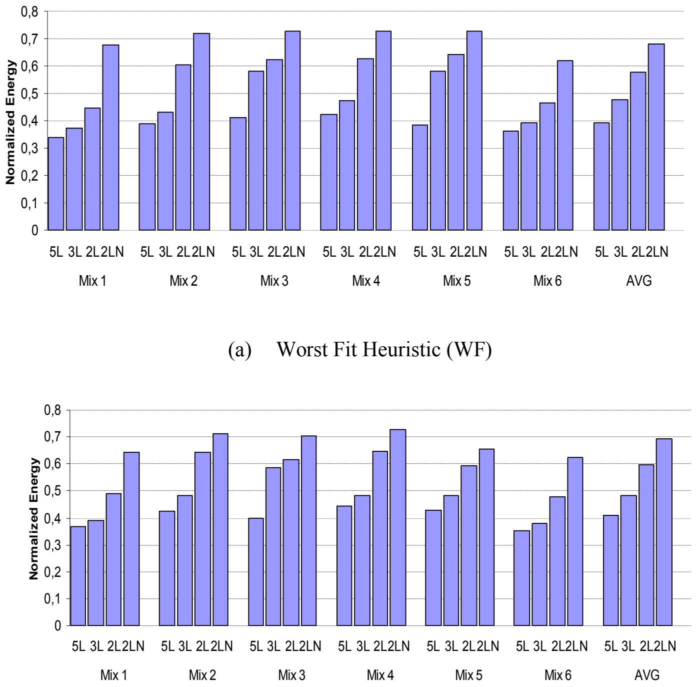
It has been also explored the energy benefits for a higher number (four) of cores (Not shown due to space restrictions). Experimental results show that when the task set fulfills certain conditions (i.e., average resource requirements and average utilization per core) the algorithm is able to sustain the energy benefits (expressed in percentage) showed for two cores.

3. CONCLUSIONS

This paper has introduced a hard real-time power-aware partitioner and scheduler for a coarse-grain multicore processor. The scheduler guarantees the tasks deadlines while applying DVS techniques to save energy.

Regarding the partitioner, a new heuristic to balance the workload has been introduced. The proposed heuristic distributes the task set according to their CPU or memory requirements among the system cores in order to increase the overlapping time to get extra slack time, which can be devoted to reduce power consumption.

Results show that the proposed heuristic (LMB) reduces the energy consumption in almost 3 times with respect to a system with no DVS regulator and applying no heuristic. Comparing the proposal with existing (i.e., WF) heuristics with a power-aware scheduler and a five-level DVS, the energy consumption is about 10% lower. On the other hand, if no algorithm is applied, the LMB provides about 14% more energy benefits than WF.



(a) Worst Fit Heuristic (WF)

(b) Limited Memory Resource.

Figure 1. Normalized Energy.

4. ACKNOWLEDGMENTS

This work was supported by Spanish CICYT under Grant TIN2006-15516-C04-01, by CONSOLIDER-INGENIO 2010 under Grant CSD2006-00046, and by Universidad Politécnica de Valencia under Grant PAID-06-07-20080029.

5. REFERENCES

[1] C. Hung, J. Chen, and T. Kuo. Energy-efficient real-time task scheduling for a DVS system with a Non-DVS processing element. In Proc. of the 27th IEEE International Real-Time Systems Symposium, pp. 303–312, 2006.

[2] D H. Chetto and M. Chetto. Some results of the earliest deadline scheduling algorithm. IEEE Transactions on Software Engineering, 15(10):1261–1269, Oct 1989.

[3] T. AlEnawy and H. Aydin. Energy-aware task allocation for Rate Monotonic scheduling. In Proc. of the 11th IEEE Real Time on Embedded Technology and Applications Symposium, pp. 213–223, 2005.

[4] Maarladen Real time Research Center. WCET analysis project. WCET Benchmark programs. Technical report, [Online].Available: http://www.mrtc.mdh.se/projects/wcet/benchmarks.html, 2006.