Improving Schedulability and Energy Efficiency for Real-Time Systems with (m,k)-Guarantee

Linwei Niu
Department of Computer and Electrical Engineering and Computer Science
California State University Bakersfield
Bakersfield, CA
lniu@csub.edu

Kuai Xu
School of Mathematical and Natural Sciences
Arizona State University
Glendale, AZ
kuai.xu@asu.edu

ABSTRACT

In this paper, we explore improving the schedulability and energy performance for real-time systems with (m,k)-constraints, which require that at least m out of any k consecutive jobs of a task meet their deadlines. The preliminary results demonstrated that our proposed approach is very promising in achieving these dual goals.

1. INTRODUCTION

Many real-time applications, such as multimedia processing and real-time communication systems, can tolerate occasional deadline misses, which can be modeled using (m,k)-model [1]. In this paper, we explore improving the schedulability and energy efficiency for such kind of systems based on EDF scheme.

Two widely used mandatory/optional partitioning techniques for (m,k)-model are R-pattern and E-pattern [3]. It is known that E-pattern has much better schedulability than R-pattern due to its even distribution of mandatory jobs [3]. In [4], Quan *et al.* showed that the schedulability of E-pattern could be improved by shifting the E-pattern. However, there still exists a large number of task sets that can not be scheduled with any shifted patterns. For example, for the task set in Figure 1(a), however we shift the E-pattern such as in Figure 1(b), the task set can not be schedulable. Also in [4], a genetic algorithm was introduced to find the general (m,k)-pattern for task sets based on fixed-priority scheme. However, their approach is not applicable to EDF case. In this paper, we developed a Simulated Annealing (SA) algorithm to solve the problem for systems based on EDF scheme, which is based on the following concept:

DEFINITION 1. ***(Intensity)*** *For a real-time task set whose mandatory jobs are determined by the given (m,k)-pattern, the intensity in the time interval $[t_s,t_f]$ is defined to be*

$$I(t_s,t_f) = \frac{\sum_i M_i(t_s,t_f) \times C_i}{t_f - t_s} \quad (1)$$

where C_i is the worst case execution time of task τ_i, and $M_i(t_s,t_f)$ is the number of mandatory jobs of τ_i which are released at or after t_s, and have deadlines less than or equal to t_f.

With the definition of intensity, our SA algorithm works by iteratively adjust the mandatory/optoinal job patterns such that the highest intensity of the system could be minimized. For example, if we apply our SA algorithm to the task set in Figure 1(a), after some iterations, the job patterns will eventually converge to the configurations similar to that in Figure 1(c) or Figure 1(d). And both (with equal highest intensities of 0.92) are schedulable. Moreover, since energy consumption is a convex function of the processor speed, which is closely related to the highest intensity of the system, the energy efficiency of the resulting task set could also be improved.

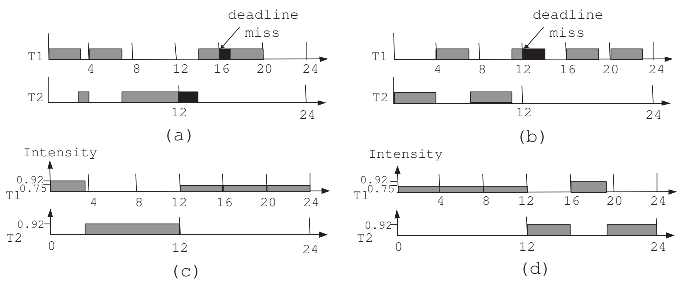

Figure 1: The given task set $\{(D_1,C_1,m_1,k_1)$=$(4,3,4,6)$; (D_2,C_2,m_2,k_2)=$(12,8,1,2)\}$: (a) NOT schedulable with E-pattern; (b) NOT schedulable with shifted E-pattern; (c) and (d) schedulable with Non-evenly distributed patterns achieved by our SA algorithm.

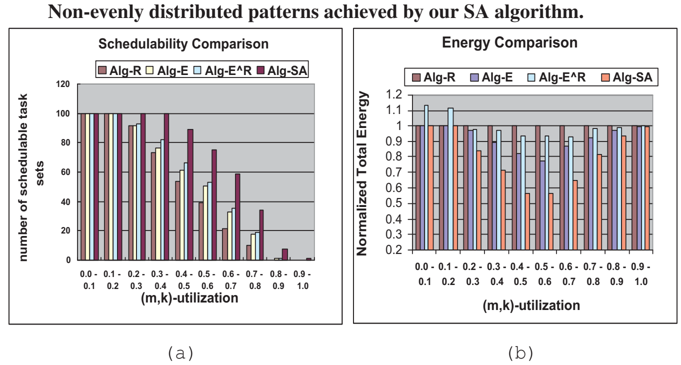

Figure 2: (a) shedulability comparison; (b) energy comparison.

2. PERFORMANCE AND CONCLUSIONS

Our experiments are based on the same task model in [3] and processor model in [2]. From Figure 2(a), for all (m,k)-utilization intervals, our SA algorithm, *i.e.*, **Alg-SA** has much better schedulablity than the approaches based on existing patterns, *i.e.*, **Alg-R** (based on R-pattern), **Alg-E** (based on E-pattern), and **Alg-E^R** [3]. Moreover, from Figure 2(b), compared to the previous approaches, significant energy reduction could be achieved by our approach.

It is also noted that SA is inherently a sequential approach and hence can be slow for problems with large search spaces. As part of our current and future work, we are exploring the parallel simulated annealing techniques on multi-core platforms to further improve the timing and energy performance of our proposed SA approach.

3. REFERENCES

[1] M. Hamdaoui and P. Ramanathan. A dynamic priority assignment technique for streams with (m,k)-firm deadlines. *IEEE Transactions on Computers*, 44:1443–1451, Dec 1995.

[2] INTEL-XSCALE, 2003.

[3] L. Niu and G. Quan. Energy minimization for real-time systems with (m,k)-guarantee. *IEEE Trans. on VLSI*, pages 717–729, July 2006.

[4] G. Quan and X. Hu. Enhanced fixed-priority scheduling with (m,k)-firm guarantee. In *RTSS*, pages 79–88, 2000.