Power-Aware Resource Allocation with Guaranteeing Fair QoS

Fumiko Harada, Toshimitsu Ushio,
Graduate School of Engineering Science, Osaka University
{harada@hopf.,ushio@}sys.es.osaka-u.ac.jp

Yukikazu Nakamoto
Graduate School of Applied Informatics, University of Hyogo
nakamoto@ai.u-hyogo.ac.jp

Abstract

In real-time systems, the execution of tasks with higher Quality of Service (QoS) requires larger resources and higher frequency in CPUs. Dynamic power-aware resource management using dynamic voltage scaling technique is needed to resolve a trade-off between the QoS maximization and the energy consumption minimization. By taking into the consideration a fairness of QoS levels of all tasks, we formulate the trade-off as a mathematical programming problem called FQ-PARA (Fair QoS-based Power-Aware Resource Allocation problem). The problem is described by two variables: the QoS level of tasks and the voltage of CPUs. Moreover, under a certain condition, it is reduced to the minimization of a single variable function.

1. Introduction

The spread of mobile and portable devices leads to demands in low power techniques. For such devices with displays and non-volatile storages, various methods are applied to reduce power consumption. Moreover, advances of circuits and implementation technologies reduce the power consumption of CPUs. However, a CPU still occupies large amount of the power consumption so that a fine grain power management is required[1].

The dynamic voltage scaling (DVS) technique is a method to reduce power consumption by controlling the CPU frequency or varying the voltage. Since the clock rate is reduced linearly as the voltage level can be reduced, it has an effect on task execution time. In [2], the computational time is divided to fixed periods and clock speeds are adjusted to eliminate idle time in each period. Predicting future execution load based on the previous execution or average execution time is used to adjust the clock speed [3, 4, 5, 6].

For low energy consumption, Phillai and Shin proposed a scheduling algorithm which defers as much work as possible by setting the frequency to meet the minimum work while ensuring the deadlines are met[7]. Liu *et al.* formalized DVS scheduling problems for hard real-time systems as a nonlinear optimization problem and proposed an optimal off-line scheduling algorithm for solving it[8]. Nakamoto *et al.* proposed optimal real-time scheduling algorithms for aperiodic task arrivals to minimize power consumption of secondary batteries[9].

Decreasing the clock speed to reduce energy consumption, however, leads to system performance degradation. Thus, there is a trade-off between the energy consumption and the system performance. To solve the energy-performance trade-off problem, Pisharath *et al.* present an application-driven scheme that aims to improve the performance and reduce the energy consumption of a heterogeneous embedded system[10]. A new power-performance modeling toolkit for a high-end microprocessor is described in [11]. Yuan *et al.* have developed an energy-efficient soft real-time CPU scheduler, GRACE-OS, for mobile multimedia systems[12]. GRACE-OS performs soft real-time scheduling and DVS based on the probability distribution of cycle demands of each multimedia task. The dynamic power management is a methodology for dynamically reconfiguring systems to provide the requested services and performance with the minimum energy consumption[13].

On the other hand, fairness of QoS level is also important in real-time systems. When CPU utilization factors are decreased with the same ratio for all tasks to avoid overload situations or to arbitrate the QoS levels of competitive tasks, the deviation of the QoS levels could occur. To prevent unfairness of QoS levels, we have proposed a QoS control method to achieve the fair QoS[14].

In this paper, we formulate the optimal power-aware resource allocation with fair QoS as an optimization problem called FQ-PARA (Fair QoS-based Power-Aware Resource Allocation problem). FQ-PARA is described by two variables: the fair QoS level and the CPU supply voltage. Moreover, under a certain condition, it is reduced to the minimization of a single variable function.

2. System Model and Assumption

We consider a uniprocessor real-time system consisting of a independent task set $\mathbf{T} = \{\tau_1, \tau_2, \ldots, \tau_n\}$ and a CPU with DVS.

Let v be the supply voltage of the CPU, which can be changed between the minimum voltage $v^{\min}$ and the maximum voltage $v^{\max}$. We assume that the CPU frequency f (CPU cycles per unit time) and the power consumption p depend on v as follows:

$$f = k_f v, \; p = k_p v^3. \tag{1}$$

Each task τ_i has a deadline D_i. τ_i comprises a mandatory part and an optional part, where the mandatory part runs before the optional one before its deadline D_i. Execution of the mandatory part produces the acceptable minimal QoS level of task τ_i, which is subsequently enhanced by the optional part.

The numbers of CPU cycles needed for execution of the mandatory and the optional parts are denoted by m_i and o_i, respectively. Since the total CPU cycles to execute τ_i is $(m_i + o_i)$, the CPU utilization factor u_i required by τ_i is given by

$$u_i = \frac{m_i + o_i}{D_i f} = \frac{m_i + o_i}{k_f D_i v} \text{ (From Eq. (1))}. \tag{2}$$

The energy consumed for completion of τ_i Thus the energy consumption e_i to complete τ_i depends on both the CPU supply voltage v and the number of CPU cycles $(m_i + o_i)$ as follows:

$$e_i = \frac{m_i + o_i}{f} \cdot p = \frac{k_p}{k_f}(m_i + o_i)v^2. \tag{3}$$

Allocation of m_i CPU cycles to execute the mandatory part guarantees the minimally acceptable QoS level $QoS_i^{\min}$ while execution of the optional one enhances the QoS level. The highest QoS level $QoS_i^{\max}$ is achieved by additional allocation of $o_i^{\max}$ CPU cycles to the optional part (the maximum CPU cycle requirement). Since achievement of each QoS level requires a certain number of CPU cycles, the QoS level of τ_i is determined uniquely for allocated CPU cycles. To evaluate fairness, we introduced a normalized QoS level Q_i as follows[14]:

$$Q_i = \frac{QoS_i^{\max} - QoS_i}{QoS_i^{\max} - QoS_i^{\min}}, \tag{4}$$

If QoS_i achieves the maximum QoS requirement $QoS_i^{\max}$, Q_i is equal to 1, while $Q_i = 0$ if $QoS_i = QoS_i^{\min}$. In the following, for simplicity, the normalized QoS level Q_i will be called a QoS level.

To quantify the CPU cycle requirement o_i, a resource consumption function ϕ_i is associated with the QoS level Q_i, namely

$$o_i = \phi_i(Q_i). \tag{5}$$

The resource consumption function represents the number of CPU cycles required to achieve a QoS level. We make some assumptions on o_i, Q_i, and ϕ_i. First, for simplicity, o_i and Q_i are real-valued within $[0, o_i^{\max}]$ and $[0, 1]$, respectively. This assumption implies that ϕ_i is a mapping from $[0, 1]$ to $[0, o_i^{\max}]$. Second, ϕ_i is a monotonically increasing convex and differentiable function (See Fig. 1).

Each task is scheduled by a scheduling algorithm such as EDF. Let U be the maximum total CPU utilization factor which guarantees schedulability of $\mathbf{T}$. To avoid an overload situation, the following CPU constraint must be satisfied:

$$\sum_{i=1}^{n} u_i \le U. \tag{6}$$

3. Problem Formulation

A CPU cycle allocation to $\mathbf{T}$ is said to be fair when all tasks have the same QoS level, namely

$$Q_1 = Q_2 = \cdots = Q_n. \tag{7}$$

Each task needs extra CPU cycles and power consumption to achieve a higher QoS level, which causes a trade-off between the minimization of power consumption and the maximization of the QoS level. To evaluate the trade-off, we introduce the following cost function as the weighted sum of these objectives:

$$w\left(1 - \frac{1}{n}\sum_{i=1}^{n} Q_i\right) + (1 - w)\sum_{i=1}^{n} e_i, \tag{8}$$

where $w \in [0, 1]$ is a weight. Intuitively, w is larger when the improvement of the QoS level is more important than the saving of the power consumption.

Our main purpose is to find a CPU cycle allocation and a CPU supply voltage so as to minimize the cost function subject to Eq. (7), which is described by the

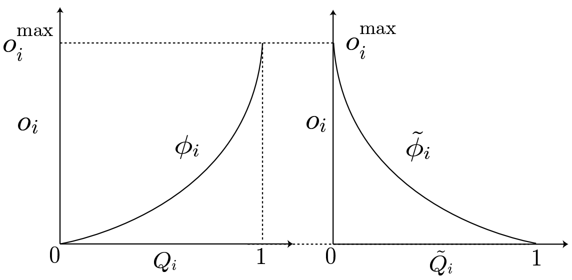

Figure 1. Illustrations of ϕ_i and $\tilde{\phi}_i$.

following nonlinear optimization problem:

$$\text{minimize} \quad w\left(1-\frac{1}{n}\sum_{i=1}^{n}Q_i\right)+(1-w)\sum_{i=1}^{n}e_i \tag{9}$$

$$\text{subject to} \quad Q_1=Q_2=\ldots=Q_n \tag{10}$$

$$\sum_{i=1}^{n}u_i\leq U \tag{11}$$

$$v^{\min}\leq v\leq v^{\max} \tag{12}$$

$$0\leq o_i\leq o_i^{\max},\ i\in\{1,2,\ldots,n\} \tag{13}$$

Obviously, a fair CPU allocation is given by $o_i = \phi_i(Q)$ for all $i \in \{1, 2, \ldots, n\}$, where Q is a QoS level for the fair allocation. Let $\tilde{Q} = 1 - Q$, and $\tilde{\phi}_i(\tilde{Q})$ is defined by $\tilde{\phi}_i(\tilde{Q}) = \phi_i(1-\tilde{Q})$ (See Fig. 1). Intuitively, $\tilde{Q}$ represents degradation of the QoS level. Note that $\tilde{\phi}_i$ is a monotonically decreasing convex function since ϕ_i is monotonically increasing and convex.

From Eqs. (2) and (3), the above optimization problem can be rewritten as follows:

$$\text{minimize} \quad W(\tilde{Q},v):=w\tilde{Q}+(1-w)\frac{k_p}{k_f}v^2\times\sum_{i=1}^{n}(\tilde{\phi}_i(\tilde{Q})+m_i) \tag{14}$$

$$\text{subject to} \quad \frac{1}{v}\sum_{i=1}^{n}\frac{m_i+\tilde{\phi}_i(\tilde{Q})}{D_i}\leq k_fU \tag{15}$$

$$v^{\min}\leq v\leq v^{\max} \tag{16}$$

$$0\leq\tilde{Q}\leq 1 \tag{17}$$

The above optimization problem will be called FQ-PARA (Fair QoS-based Power-Aware Resource Allocation problem). It is noted that the above problem has two variables $\tilde{Q}$ and v. The second term of the cost function depends on the number of tasks as a summation of required CPU cycles, and the calculation of the cost function has linear complexity with respect to the number of tasks, which is an advantage of FQ-PARA.

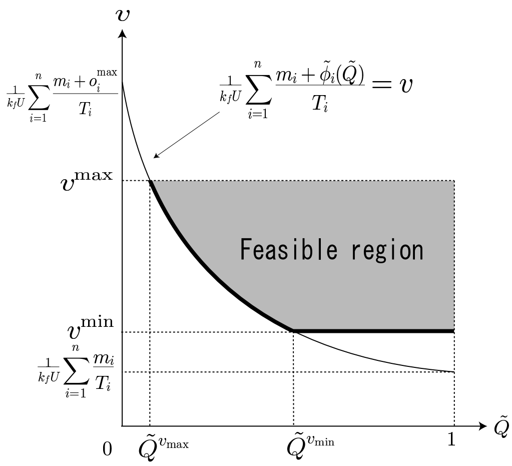

Figure 2. Illustration of the feasible region: the optimal solution exists in its border shown by a thick solid line.

4. Existence of Optimal Solution

This section derives a basic property of FQ-PARA. Since

$$\frac{\partial W(\tilde{Q},v)}{\partial v}=2(1-w)\frac{k_p}{k_f}v\sum_{i=1}^{n}\frac{m_i+\tilde{\phi}_i(\tilde{Q})}{D_i} > 0, \tag{18}$$

the cost function $W(\tilde{Q}, v)$ is strictly increasing with respect to v, which means that more energy is needed as the voltage (hence CPU frequency) is increased. Therefore, the optimal solution $(\tilde{Q}^*, v^*)$ satisfies

$$v^*=\frac{1}{k_fU}\sum_{i=1}^{n}\frac{m_i+\tilde{\phi}_i(\tilde{Q}^*)}{D_i}\ \text{or} \tag{19}$$

$$v^*=v^{\min}. \tag{20}$$

Figure 2 shows a feasible region of FQ-PARA when there exist $\tilde{Q}^{v^{\max}}$ and $\tilde{Q}^{v^{\min}}$ such that

$$\frac{1}{k_fU}\sum_{i=1}^{n}\frac{m_i+\tilde{\phi}_i(\tilde{Q})}{D_i}\geq v^{\max},\ \tilde{Q}\leq\tilde{Q}^{v^{\max}}\ \text{and} \tag{21}$$

$$\frac{1}{k_fU}\sum_{i=1}^{n}\frac{m_i+\tilde{\phi}_i(\tilde{Q})}{D_i}\leq v^{\min},\ \tilde{Q}\geq\tilde{Q}^{v^{\min}}. \tag{22}$$

Equation (21) implies that no QoS level higher than $\tilde{Q}^{v^{\max}}$ can be achieved even if the voltage is set to its maximal value $v^{\max}$.

We define the following functions W_1 and W_2 of $\tilde{Q}$:

$$W_1(\tilde{Q}) := W\left(Q, \frac{1}{k_f U}\sum_{i=1}^{n}\frac{m_i + \tilde{\phi}_i(\tilde{Q})}{D_i}\right) \tag{23}$$

$$= w\tilde{Q} + (1-w)\sum_{i=1}^{n}\big(m_i + \tilde{\phi}_i(\tilde{Q})\big) \times \frac{k_p}{{k_f}^3 U^2}\left\{\sum_{i=1}^{n}\frac{m_i + \tilde{\phi}_i(\tilde{Q})}{D_i}\right\}^2, \tag{24}$$

$$W_2(\tilde{Q}) := W(\tilde{Q}, v^{\min}) \tag{25}$$

$$= w\tilde{Q} + (1-w)\frac{k_p}{k_f}{v^{\min}}^2\sum_{i=1}^{n}\left(m_i + \tilde{\phi}_i(\tilde{Q})\right). \tag{26}$$

Note that they are convex functions since $\frac{d^2 W_i}{d\tilde{Q}^2} > 0 (i = 1, 2)$.

As shown in Fig. 2, the optimal solution $(\tilde{Q}^*, v^*)$ and the optimal value W^* of FQ-PARA under Eqs. (21) and (22) satisfy the following conditions:

A: If $\tilde{Q}^{v^{\max}} \le \tilde{Q}^* \le \tilde{Q}^{v^{\min}}$, then

$$v^* = \frac{1}{k_f U}\sum_{i=1}^{n}\frac{m_i + \tilde{\phi}_i(\tilde{Q}^*)}{D_i} \text{ and} \tag{27}$$

$$W^* = W_1(\tilde{Q}^*). \tag{28}$$

B: If $\tilde{Q}^{v^{\min}} \le \tilde{Q}^* \le 1$, then

$$v^* = v^{\min} \text{ and } W^* = W_2(\tilde{Q}^*). \tag{29}$$

If $\frac{dW_1(Q^{v^{\min}})}{d\tilde{Q}} > 0$, then we have $\frac{dW_2(Q^{v^{\min}})}{d\tilde{Q}} > 0$ from Eq. (18), which implies that the optimal solution satisfies the case **A**, otherwise the case **B**.

Since both W_1 and W_2 are one-dimensional convex functions, the optimal solution $(\tilde{Q}^*, v^*)$ can be obtained by minimization of W_1 in the interval $[\tilde{Q}^{v^{\max}}, \tilde{Q}^{v^{\min}}]$(*resp.* W_2 in $[\tilde{Q}^{v^{\min}}, 1]$) for the case **A**(*resp.* **B**). The minimization is performed by a numerical method such as the Newton method. Thus, FQ-PARA can be solved by minimizing a single variable function.

5. Conclusion and Future Work

This paper formulated an optimization problem called FQ-PARA to achieve low power consumption and fair CPU allocation. Though FQ-PARA is a nonlinear optimization programming problem, the optimal solution can be found by solving the minimal point of one-dimensional convex functions.

Our future work is to extend FQ-PARA to real-time systems with discrete-valued QoS levels, since many applications have discrete QoS levels (e.g. picture resolution, color depth, and so on) and DVFS CPUs have discrete voltage levels. Moreover, it is also future work to propose an efficient algorithm for solving FQ-PARA on-line.

References

[1] T. Truman, T. Pering, R. Doering, and R. Brodersen, "the InfoPad Multimedia Terminal: A Portable Device for Wireless Information Access," IEEE Trans. on Computers, Vol.47, No.10, pp.1073–1087, October 1998.

[2] M. Weiser, B. Welch, A. Demers, and S. Shenker, "Scheduling for Reduced CPU Energy", In Proc. of the 1st Symposium on Operating Systems Design and Implementation, pp.13–23, November 1994.

[3] H. Aydi, R. Melhem, D. Mossé, and P. Mejía-Alvarez, "Dynamic and Aggressive Scheduling Techniques for Power-Aware Real-Time Systems," In Proc. of the 22nd IEEE Real-Time Systems Symposium, pp.95–105, December 2001.

[4] Y. Lee and C. Krishna, "Voltage-Clock Scaling for Low Energy Consumption in Real-Time Embedded Systems", In Proc. of the 6th International Conference on Real-Time Computing Systems and Applications, pp.272–279, December 1999.

[5] V. Raghunathan, P. Spanos, and M. Srivastava, "Adaptive Power-Fidelity in Energy-Aware Wireless Embedded Systems," In Proc. of the 22nd IEEE Real-Time Systems Symposium, pp.106–115, December 2001.

[6] D. Zhu, R. Melhem, and B. Childers, "Scheduling with Dynamic Voltage/Speed Adjustment Using Slack Reclamation in Multi-Processor Real-Time Systems," In Proc. of the 22nd IEEE Real-Time Systems Symposium, pp.84–94, December 2001.

[7] P. Pillai and K. Shin, "Real-Time Dynamic Voltage Scaling for Low-Power Embedded Operating Systems," In Proc. of the 18th ACM Symposium on Operating System Principles, pp.89–102, October 2001.

[8] Y. Liu and A. Mok, "An Integrated Approach for Applying Dynamic Voltage Scaling to Hard Real-Time Systems," In Proc. of the 9th IEEE Real-Time and Embedded Technology and Applications Symposium, pp.116–123, May 2003.

[9] Y. Nakamoto, Y. Tsujino, and N. Tokura, "Real-Time Task Scheduling Algorithms for Maximum Utilization of Secondary Batteries in Portable Devices," In Proc. of the 7th International Conference on Real-Time Systems and Applications, pp.347–354, December 2000.

[10] J. Pisharath, N. Jiang, and A. Choudhary, "Evaluation of Application-Aware Heterogeneous Embedded Systems for Performance and Energy Consumption," In Proc. of the 9th IEEE Real-Time and Embedded Technology and Applications Symposium, pp.124–132, May 2003

[11] D. Brooks, J. Wellman, P. Bose, and M. Martonosi, "Power-Performance Modeling and Tradeoff Analysis for a High-End Microprocessor," In Proc. of the 1st International Workshop on Power-Aware Computer Systems, pp.126–136, November 2000.

[12] W. Yuan and K. Nahrstedt, "Energy-Efficient Soft Real-Time CPU Scheduling for Mobile Multimedia Systems," In Proc. of the 19th ACM Symposium on Operating Systems Principles, pp.149–163, October 2003.

[13] L. Benini, A. Bogliolo, and G. Micheli, "A Survey of Design Techniques for System-Level Dynamic Power Management," IEEE Transaction on Very Large Scale Integration System, Vol.8, No.3, pp.299–316, June 2000.

[14] F. Harada, T. Ushio, and Y. Nakamoto, "Adaptive Resource Allocation Control with On-Line Search for Fair QoS Level," In Proc. of the 10th IEEE Real-Time and Embedded Technology and Applications Symposium, pp.352–359, May 2004.