Leakage Aware Energy Minimization for Real-Time Systems under the Maximum Temperature Constraint

Huang Huang, Gang Quan
Department of Electrical and Computer Engineering, Florida International University, Miami, FL 33174
E-mail: hhuan001@fiu.edu, gaquan@fiu.edu

Abstract

In this paper, we study the problem on how to reduce the overall energy consumption while at the same time ensuring the timing and maximum temperature constraints for a real-time system. We incorporate the interdependence of leakage, temperature and supply voltage into analysis and develop a novel method to quickly estimate the overall energy consumption. Based on this method, we then propose a scheduling technique to minimize the overall energy consumption under the maximum temperature constraint. Our experimental results show that the proposed energy estimation method can achieve up to four-order-of-magnitude speedup compared with existing approaches while keeping the maximum estimation error within 4.8%. In addition, simulation results also demonstrate that our proposed energy minimization method consistently outperforms previous related approaches significantly.

1. Introduction

The human's appetite for high performance of computing systems has driven the semiconductor technology into the deep submicron era. The continued shrinking of the transistor size, together with increasingly complicated circuit architecture, has resulted in a significant increase of the power density. It is predicted that in a matter of a few years, the number of transistors that can be integrated into one $300mm^2$ die will reach 100 billion and its power consumption can be as high as 300W [4]. The soaring power consumption has posed tremendous challenge not only on how to provide enough power sources for the computing systems, but also on how to remove the heat generated by these systems.

The escaladed power consumption has directly led to high temperature, which not only increases the packaging/cooling costs, degrades the reliability/performance of the system, but also significantly increases the leakage power consumption. From [12], the leakage power will increase by 38% when chip temperature raising from 65^0C to 110^0C. As the leakage power becomes more and more prominent in the deep submicron domain, i.e. up to 70% of the total energy consumption [10], a power/thermal aware design technique will simply become ineffective if the interdependence of leakage and temperature is not properly addressed.

In this paper, we study the problem of minimizing the overall energy for a real-time system under the maximum operating temperature constraint. As related works, there are approaches proposed to minimize the maximum operational temperature (e.g. [2, 6, 8, 18, 22]) when scheduling a real-time task. They show that, while temperature and power consumption are closely related, the thermal aware computing and the power-aware computing problems are distinctly different. There also have been extensive researches published on reducing both the dynamic and leakage energy consumption (e.g. [7, 11, 17]). However, these work ignore the interdependent relationship between the leakage and temperature.

There are a few researches ([3, 20, 21]) that are closely related to our research which have taken the leakage/temperature dependency into considerations when minimizing the energy consumption. Specifically, Yuan et al. [21] introduced a simple heuristic to turn on or off a processor based on its workload and temperature. This approach directly employs the circuit level leakage/temperature model ([12, 1]) to capture the interdependence of leakage and temperature. However, as a result of the complexity from the nonlinear and high order terms of this model, the approach in [21] can only be applied for soft real-time systems. Bao et al. [3] proposed to distribute the idle interval wisely when scheduling a task graph such that the temperature of the processor can be effectively "cooled down" and thus reduce the leakage power consumption. In this approach, the leakage power is modeled using a piece-wise linear function of temperature. However, leakage power changes not only with temperature but also supply voltage as well [12]. As evidenced in [9], the leakage model ignoring the effect of supply voltage can lead to results deviated far away from the actual values. Yang et al. [20] proposed a quadratic leakage model to simplify the leakage/temperature dependency. Based on this model, they proposed a "pattern-based" scheduling approach to periodically switch the processor between the active and dormant modes and reduce the energy. However, their energy estimation method can be only used when the processor reaches the temperature steady state. In addition, their scheduling method cannot guarantee the maximal temperature constraint.

978-3-9810801-7-9/DATE11/©2011 EDAA

In this paper, we study the problem on how to schedule a periodic hard real-time task and minimize the energy consumption under the timing and peak temperature constraints. To solve this problem, we first incorporate the interdependency of leakage, temperature and supply voltage into analysis and derive a closed-form energy consumption formula that can efficiently and accurately find the energy consumption of a given schedule. We further develop a scheduling policy based on the "M-Oscillating" technique [6] to minimize the total energy. The original "M-Oscillating" technique aims at controlling the maximum temperature under the ideal scenarios. We extend this approach for energy minimization and take both the timing and energy transition overheads into considerations. Our experimental results show that the proposed energy estimation method can achieve up to 4 orders of magnitude of speedup compared with existing approach [14] while keeping the estimation error within 4.8%. In addition, under different workload densities, the proposed method consistently outperforms the previous researches and has an average of 15.5% and 25.7% improvement for energy saving compared with the pattern-based approach [20] and the naive approach, respectively.

The rest of the paper is organized as follows. Section 2 introduces the system models and presents a motivational example. Section 3 presents our energy estimation method and scheduling approach. Experimental results are discussed in Section 4. Section 5 concludes this paper.

2. Preliminary

System Models The real time system considered in this paper consists of a set of periodic tasks with equal periods and deadlines. We can equivalently assume such a system consists of only one periodic task and further assume that its deadline equals to its period.

The thermal model used in this paper is similar to the one that has been used in the similar researches (e.g. [6, 20, 5, 8, 15]). Specifically, let $T(t)$ and T_{amb} be the chip temperature and ambient temperature respectively. Let $P(t)$ denotes the power consumption (in $Watt$) at time t, and R, C are the thermal resistance (in $^oC/Watt$) and thermal capacitance (in $J/^oC$) respectively. Then

$$RC\frac{dT(t)}{dt} = RP(t) + (T(t) - T_{amb}), \tag{1}$$

We can scale T such that T_{amb} is zero and then we have

$$\frac{dT(t)}{dt} = aP(t) - bT(t), \tag{2}$$

where $a = 1/C$ and $b = 1/RC$.

We assume that the processor can provide n different supply voltage levels with each voltage level associated with a different clock frequency. We further assume that for each speed transition, an extra energy overhead of E_{sw} will be consumed and a timing penalty of τ will be incurred during which the clock is halted that no computation can be taken place.

The overall power consumption of the processor is composed of two parts: the dynamic power P_{dyn} and leakage power P_{leak}. The dynamic power consumption is independent to the temperature and can be formulated as $P_{dyn} = C_2 v_k^3$ [19], where v_k is the kth supply voltage level and C_2 is a constant. The leakage power is temperature dependent and can be calculated as $P_{leak} = N_{gate} \cdot I_{leak} \cdot v_{dd}$, where N_{gate} represents the number of gate, v_{dd} is the supply voltage, and I_{leak} is the leakage current, which can be formulated by a non-linear exponential equation as [12]

$$I_{leak} = I_s \cdot (\mathcal{A} \cdot T^2 \cdot e^{((\alpha \cdot V_{dd} + \beta)/T)} + \mathcal{B} \cdot e^{(\gamma \cdot V_{dd} + \delta)}) \tag{3}$$

where I_s is the leakage current at certain reference temperature and supply voltage, T is the operating temperature, V_{dd} is the supply voltage, $\mathcal{A}, \mathcal{B}, \alpha, \beta, \gamma, \delta$ are empirically determined technology constants.

As reported in [13] that the leakage current changes super linearly with temperature and by using linear approximation method to model the leakage/temperature dependence can maintain reasonable accuracy, i.e. less than 5.5%. Thus, the leakage power for the processor running in mode k can be effectively estimated as [6]

$$P_{leak}(k) = C_0(k)v_k + C_1(k)Tv_k, \tag{4}$$

where $C_0(k)$ and $C_1(k)$ are constants. As we can see from equation (4), the leakage power depends on both the supply voltage and temperature. The total power consumption at the kth processor mode is thus

$$P(k) = C_0(k)v_k + C_1(k) \cdot Tv_k + C_2 v_k^3. \tag{5}$$

Accordingly, the temperature dynamics at the speed level k can be formulated as

$$\frac{dT(t)}{dt} = A(k) - B(k)T(t) \tag{6}$$

where $A(k) = a(C_0(k)v_k + C_2 v_k^3)$ and $B(k) = b - aC_1(k)v_k$. Hence, for a given time interval $[t_0, t_e]$, if the initial temperature is T_0, by solving equation (6), the ending temperature can be formulated as below:

$$\begin{aligned} T_e &= \frac{A(k)}{B(k)} + (T_0 - \frac{A(k)}{B(k)})e^{-B(k)(t_e - t_0)} \\ &= G(k) + (T_0 - G(k))e^{-B(k)(t_e - t_0)}. \end{aligned} \tag{7}$$

Formally, our problem can be formulated as follows.

PROBLEM 1. *Given a hard real-time task set with period p and worst case execution time c, develop a feasible schedule such that the overall energy consumption can be minimized under the peak temperature of T_{max}.*

Motivational Examples The major challenge of Problem 1 lies in the interactions between the power (leakage power) and the temperature. Let us consider a simple periodic task with period (p) of 1000s and the worst case execution time (c) of 700s. A simple naive approach, as shown in Figure 1(a) is to executes the task with a single speed and turns to sleep mode upon the finish of this task. Another approach, i.e. the pattern-based approach [20], as illustrated in Figure 1(b), also uses a single speed to run the task. This approach lets the processor's active state to be broken down to a number of intervals so that the processor can have a chance to cool down. The third approach, so-called the M-Oscillating approach [6], as illustrated in Figure 1(c), alternates two none-zero speeds (S_1 and S_2) to execute a task. This approach is shown to be very effective to control the peak temperature.

Figure 2 shows the normalized overall energy consumptions by all three different approaches based on a processor model built based

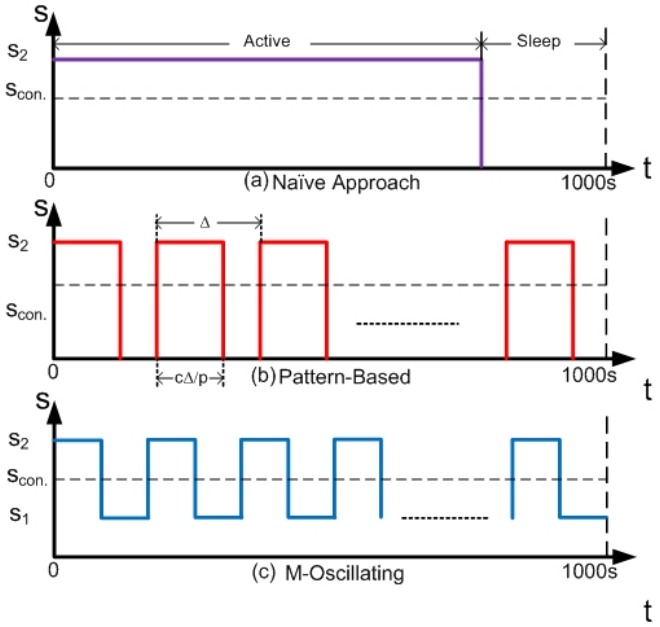


Figure 1. Three different scheduling approaches. S_{con} refers to the minimum constant speed needed to complete the workload.

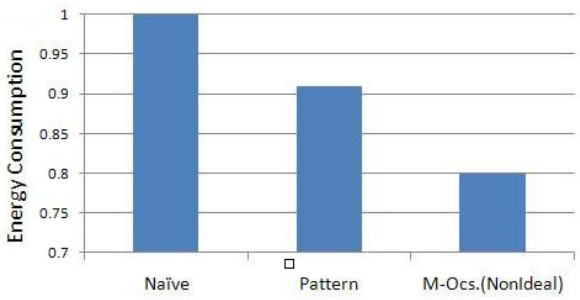


Figure 2. Normalized Energy consumptions by three different scheduling approaches

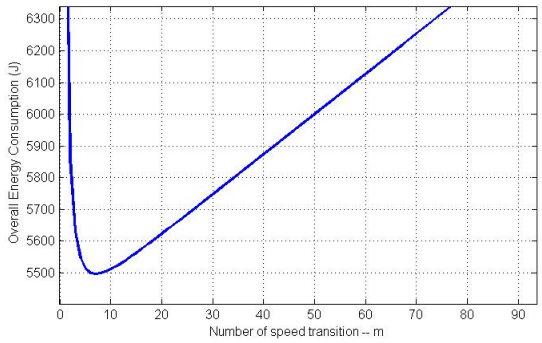


Figure 3. Energy consumption of M-Oscillating with transition overhead ($E_{sw} = 0.01J$, $\tau = 0.1s$, $S_1 = 0.8V$ and $S_2 = 1.0V$)

on the 65nm technology, similar to the one in [18]. As shown in the figure, while both the naive approach and the pattern-based approach consume the same amount of dynamic energy, the overall energy consumptions by different policies are quite different. The pattern-based approach can save approximately 10% of the total energy by the naive approach. The M-Oscillating method can save further, 10% of the energy by the pattern-based approach. This example highlights the importance to incorporate the temperature/leakage dependency into power aware analysis in the deep sub-micron domain.

According to the original M-Oscillating method, the more frequently the processor speed is alternated, the lower the peak temperature of the processor can be. However, in practical scenarios, changing processor modes takes both timing and energy overhead. If we account for these overheads, Figure 3 shows the relationship between the energy consumption and the number of speed transitions, i.e. m, of the M-Oscillating scheme. As we can see from Figure 3, as we increase the number of transitions, the timing and energy overheads can eventually outweigh the benefit and thus increase the overall energy consumption. Therefore, the problem is how to judiciously choose the appropriate number of transitions with timing and energy transition overheads in mind to achieve the best energy efficiency.

3. Our approach

To solve Problem 1, we first propose an efficient energy estimation method. Based on this method, we develop an algorithm to minimize the overall energy under timing and maximum temperature constraints.

3.1 Energy Consumption Calculation

The main challenge for accurate energy estimation lies in the fact that the leakage energy depends not only on supply voltage but also on temperature. One intuitive method, similar to that in [14], is to calculate the leakage power by dividing the interval into small sub-intervals, within which one can assume the temperature (and thus the leakage power) will remain constant. With the circuit level model (e.g. equation (3)), this method can estimate total energy consumption accurately as long as the sub-interval is sufficiently small. However, the computational cost can be very sensitive to the accuracy of this approach. An accurate energy estimation method with low computation cost is highly desirable.

Consider running a processor in mode k for interval $[t_0, t_e]$. The temperature varies following equation (2). By integrating both sides of equation (2), we have

$$\begin{aligned} \int_{t_0}^{t_e} \frac{dT(t)}{dt} \cdot dt &= a \cdot \int_{t_0}^{t_e} P(t)dt - b \cdot \int_{t_0}^{t_e} T(t)dt \\ T_e - T_0 &= a \cdot E(t_0, t_e) - b \cdot \int_{t_0}^{t_e} T(t)dt \\ E(t_0, t_e) &= 1/a \cdot (T_e - T_0 + b \cdot \int_{t_0}^{t_e} T(t)dt) \end{aligned} \quad (8)$$

where T_0 and T_e denote the starting and ending temperature of interval $[t_0,t_e]$ and $E(t_0,t_e)$ represents the energy consumed during this interval. Note that T_e can be analytically calculated using equation (7). Furthermore, since

$$T(t) = G(k) + (T_0 - G(k))e^{-B(k)(t-t_0)}. \quad (9)$$

we have

$$\int_{t_0}^{t_e} T(t)dt = \int_{t_0}^{t_e} G(k)dt + \int_{t_0}^{t_e} (T_0 - G(k)) \cdot e^{-B(k)(t-t_0)}dt$$
$$= G(k) \cdot (t_e - t_0) + 1/B(k) \cdot (G(k) - T_0)(e^{-B(k)(t_e-t_0)} - 1) \quad (10)$$

The overall energy consumed within interval $[t_0,t_e]$ can thus be expressed analytically as

$$\begin{aligned} E(t_0,t_e) &= 1/a \cdot (G(k) + (T_0 - G(k))e^{-B(k)(t_e-t_0)} \\ &- T_0 + b \cdot (G(k) \cdot (t_e - t_0) \\ &+ 1/B(k) \cdot (G(k) - T_0)(e^{-B(k)(t_e-t_0)} - 1))). \end{aligned} \quad (11)$$

It is not difficult to see that equation (11) has significantly lower computational cost than with the energy estimation method in [14]. It is worthy of mentioning that the accuracy to employ equation (11) to estimate energy consumption is contingent upon the accuracy using linear approximation method to estimate the leakage. In Section 4.1, we use experiments to quantitatively evaluate the accuracy and efficiency of our proposed energy estimation method.

3.2 Leakage/Temperature Aware Energy Minimization Scheduling

In this subsection, we introduce our scheduling technique to reduce the overall energy. Our scheduling approach is based on the principle of M-Oscillating approach [6], which has shown to be effective in controlling peak temperature. Given a task with its period of p and the worst case execution time of c, it is not always possible to use the constant speed $S_{con} = c/p$ to execute the task and two neighboring speeds of S_{con} (S_i and S_j, ($S_i < S_j$)) are used as shown in Figure 4(a). Let processor runs at S_i for t_i time units followed by t_j time units of S_j to complete the workload. The M-Oscillating approach calls for alternating S_i and S_j for every t_i/m and t_j/m, respectively.

Note that, due to the transition overhead, the clock will halt for a short interval τ at each speed transition and a workload loss is imposed for every transition. One solution to compensate this workload loss is to increase the processor speed so that more workloads can be finished with the same time. However, this option is not always viable, e.g. when the current speed is already the maximum speed. Alternatively, one could extend the high speed interval and reduce the low speed interval to make this compensation. Therefore, when considering the transition timing overhead and let t_{im} (and t_{jm}) be the adjusted length of the low (and high) speed interval in an M-Oscillating schedule with m divisions, then based on Figure 4(b), we have

$$m(t_{im} + t_{jm} + 2\tau) = p \quad (12)$$
$$S_i t_{im} + S_j t_{jm} = c/m \quad (13)$$

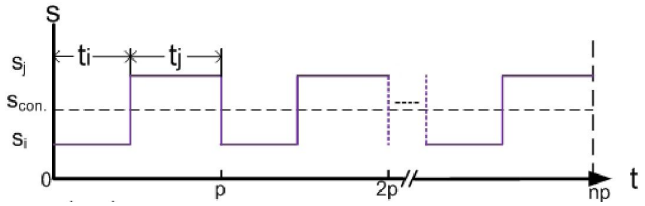
Hence, with any given m, we can immediately obtain values of t_{im} and t_{jm} accordingly. Furthermore, based on the above two equations, with several mathematic transformations, it is not difficult to bound the value of m as

$$m \le m_{Max} = \lfloor \frac{S_j p - c}{2S_j \tau} \rfloor. \quad (14)$$

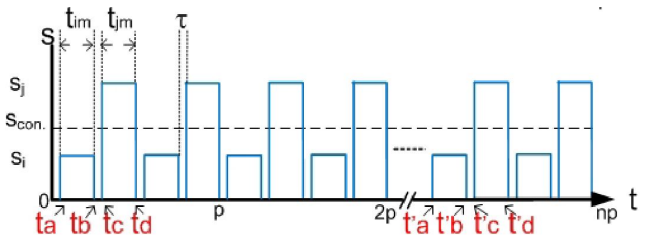
Besides the timing constraints, we also need to ensure the peak temperature constraint as well. Based on [18], the peak temperature of $T_{peak}(m)$ can be formulated as

$$T_{peak}(m) = T_0 + \frac{T_d(m) - T_0}{1 - K(m)} \quad (15)$$

where T_0 is the initial temperature, $K(m) = exp(-B(i)t_{im} - B(j)t_{jm} - 2B(0)\tau$, and $T_d(m)$ is the temperature at $t = t_d$ (the highest temperature in the first division of the M-Oscillating schedule), which is readily available based on equation (7).



(a) A periodic schedule using two processor speeds



(b) The corresponding M-oscillating schedule with non-negligible transition overhead

Figure 4. Proposed scheduling approach

From Figure 4(b), the overall energy consumption in one period at the stable status, i.e. $\tilde{E}(m)$, can thus be formulated as

$$\tilde{E}(m) = m \cdot (E(t'_a,t'_b) + E(t'_c,t'_d)) + 2m \cdot E_{sw}. \quad (16)$$

where $E(t'_a,t'_b)$ and $E(t'_c,t'_d)$ denote the energy consumption with respect to the low and high speed intervals correspondingly, and E_{sw} is the switching energy overhead and there are totally $2m$ transitions taken place. Our problem is therefore to minimize $\tilde{E}(m)$ subject to $m \le m_{Max}$ and $T_{peak}(m) \le T_{max}$. To search for the optimal value of m, we can use simple exhaustive search or some more elaborative ones such as those target at the unimodal functions[16]. Because of the simplicity of both our energy estimation technique as well as the peak/equilibrium temperature calculation method, we are afford to search for the optimal m exhaustively. The detailed algorithm is omitted due to page limitation. In the next section, we use experiments to evaluate the performance of our approach.

4. Experimental Results

In this section, we use experiments to evaluate our methods. First, the accuracy and efficiency of the proposed energy estimation method

Table 1. Test Energy Calculation Formula

PERIOD(s)	V_{dd} (v)	REAL (J)	PROPOSED (J)	ERROR	Ave.TIME-REAL (s)	Ave.TIME-PROP(s)	Ave.SPEEDUP (X)
10	0.6	0.0569×10^3	0.0542×10^3	4.8%	0.0092	0.0025	3.5X
	0.8	0.127×10^3	0.122×10^3	3.3%			
	1.0	0.267×10^3	0.261×10^3	2.2%			
	1.2	0.595×10^3	0.587×10^3	1.3%			
100	0.6	0.572×10^3	0.546×10^3	4.5%	0.288	0.0026	110X
	0.8	1.28×10^3	1.25×10^3	2.8%			
	1.0	2.72×10^3	2.67×10^3	1.6%			
	1.2	6.09×10^3	6.06×10^3	0.5%			
200	0.6	1.15×10^3	1.10×10^3	4.2%	1.1	0.0028	390X
	0.8	2.58×10^3	2.52×10^3	2.5%			
	1.0	5.51×10^3	5.45×10^3	1.1%			
	1.2	1.246×10^4	1.235×10^4	0.4%			
500	0.6	2.90×10^3	2.79×10^3	3.7%	6.5	0.0027	2400X
	0.8	6.57×10^3	6.46×10^3	1.7%			
	1.0	1.42×10^4	1.41×10^4	0.6%			
	1.2	3.28×10^4	3.29×10^4	0.2%			
1000	0.6	5.85×10^3	5.65×10^3	3.3%	101.3	0.0026	38846X
	0.8	1.33×10^4	1.31×10^4	1.2%			
	1.0	2.91×10^4	2.90×10^4	0.4%			
	1.2	6.867×10^4	6.861×10^4	0.1%			

will be examined. We then study the performance of the proposed energy minimization scheduling technique.

4.1 Accuracy and Efficiency of Energy Estimation Technique

To study the accuracy and efficiency of our energy estimation method, we adopted the processor model similar to the one in [18]. The timing penalty of speed transition is set to be 0.1s [5] and the transition energy overhead E_{sw} is 0.01J [20]. We let the processor run in intervals with different lengths and using different modes. We then compare the overall energy consumptions and computational costs achieved by using our method as detailed in section 3.1 with the one in [14]. For accurate energy estimation, we use a very small time interval, e.g. 0.01s, as the step size of the method in [14]. All experiments were running on a Window XP/SP3 platform powered by Intel(R) Core(TM)2 Duo CPU @ 2.93GHz with 3.21 GB of RAM. The energy consumption under each test cases and the corresponding CPU times are summarized in Table 1 where the results by [3] is labeled as "REAL" and ours as "PROPOSED". The relative errors and speedups of our approach under different test cases are also listed in the table.

Our experimental results clearly showed the accuracy and efficiency of our proposed energy estimation method. As can be seen in Table 1, the energy consumption estimated by using our proposed method closely matches the one by using the approach in [14], with the maximum relative error no more than 4.8% among all our test cases and 2.2% in average. The computational cost, on the other hand, is significantly reduced, i.e. up to a maximum of 38846 times of speedup or over 8000 times of speedup in average for our test cases. We can also see from Table 1 that the longer the interval and the higher the supply voltage level are, more accurate and efficient our method can be.

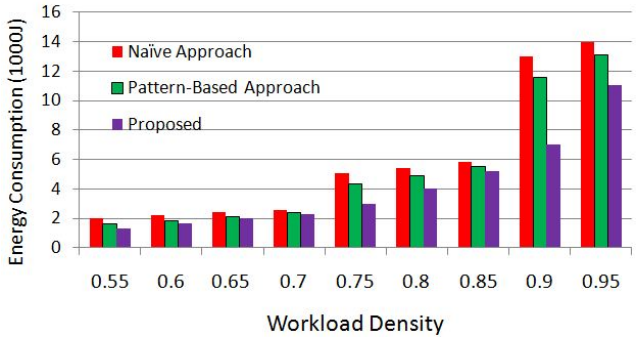
4.2 Leakage/Temperature Aware Energy Minimization under Peak Temperature Constraint

To study the effective of our scheduling approach, we compared our approach with the naive approach and the pattern-based approach (illustrated in Figure 1(a) and Figure 1(b) respectively). The periods of the tasks were set to 2000s. The worst case execution times were randomly generated with workload density (c/p) varying from 55% to 95% with 5% increment. The reason we do not further reduce the workload density is that when $c/p < 55\%$, the low speed mode used by the pattern-based approach and the proposed technique become the same, i.e. the lowest available supply voltage of $v = 0.6V$. The temperature constraint was set to be $T_{max} = 85^oC$. The average energy consumption of the feasible schedules in each group were calculated and plot in Figure 5(a).

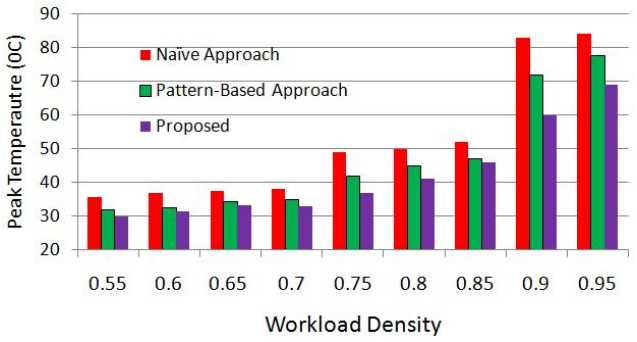
From Figure 5(a), both the pattern-based approach and the proposed technique can achieve significant energy reduction, and the proposed method consistently outperforms the pattern-based approach under different test scenarios. Compared with the naive approach, on average, 25.7% energy can be saved by using our method versus 11.2% by the pattern-based approach. We also compared the average peak temperature of the feasible schedules in each group as shown in Figure 5(b). The peak temperature achieved by the proposed approach can be 23^oC and 12^oC (or 9.5^oC and 4.5^oC in average) lower than the naive approach and the pattern-based approach respectively. Therefore, our approach can be even more effective when the peak temperature constraint becomes tighter.

5. Conclusions

In this paper, we proposed a novel real-time scheduling method to minimize the overall energy consumption under the timing and



(a) Energy Vs. Workload



(b) Temperature Vs. Workload

Figure 5. Performance comparison of three different scheduling approaches

maximum temperature constraints. We incorporate the interdependence of leakage, temperature, and supply voltage into analysis and develop a method that can accurately and quickly estimate the overall energy consumption. Based on this method, we then develop an energy minimization scheduling technique with practical factors such as the timing and energy overhead during the processor transition taken into considerations. Our experimental results show that the proposed energy estimation method can achieve up to four orders of magnitude of speedup compared with existing approach while keeping the maximum estimation error within 4.8%. In addition, simulation results also demonstrate that our proposed energy minimization method consistently outperforms the previous approaches significantly.

Acknowledgment

This work is supported in part by NSF under projects CNS-0969013, CNS-0917021, and CNS-1018108.

6. References

[1] Hotspot 4.2 temperature modeling tool. *University of Virgina*, page http://lava.cs.virginia.edu/HotSpot, 2009.

[2] N. Bansal, T. Kimbrel, and K. Pruhs. Speed scaling to manage energy and temperature. *Journal of the ACM*, 54(1):1–39, 2007.

[3] M. Bao, A. Andrei, P. Eles, and Z. Peng. Temperature-aware idle time distribution for energy optimization with dynamic voltage scaling. In *DATE*, pages 21 – 27, 2010.

[4] S. Borkar. Thousand core chips: a technology perspective. In *DAC*, pages 746–749, 2007.

[5] T. Chantem, X. S. Hu, and R. Dick. Online work maximization under a peak temperature constraint. In *ISLPED*, pages 105–110, 2009.

[6] V. Chaturvedi, H. Huang, and G. Quan. Leakage aware scheduling on maximal temperature minimization for periodic hard real-time systems. In *ICESS*, 2010.

[7] J.-J. Chen, H.-R. Hsu, and T.-W. Kuo. Leakage-aware energy-efficient scheduling of real-time tasks in multiprocessor systems. In *RTAS*, pages 408–417, 2006.

[8] J.-J. Chen, S. Wang, and L. Thiele. Proactive speed scheduling for real-time tasks under thermal constraints. *RTAS*, 0:141–150, 2009.

[9] H. Huang, G. Quan, and J. Fan. Leakage temperature dependency modeling in system level analysis. In *ISQED*, pages 447–452, 2010.

[10] ITRS. *International Technology Roadmap for Semiconductors (2008 Edition)*. International SEMATECH, Austin, TX., http://public.itrs.net/.

[11] R. Jejurikar, C. Pereira, and R. Gupta. Dynamic slack reclamation with procrastination scheduling in real-time embedded systems. *DAC*, 2005.

[12] W. Liao, L. He, and K. Lepak. Temperature and supply voltage aware performance and power modeling at microarchitecture level. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 24(7):1042 – 1053, 2005.

[13] Y. Liu, R. P. Dick, L. Shang, and H. Yang. Accurate temperature-dependent integrated circuit leakage power estimation is easy. In *DATE*, pages 1526–1531, 2007.

[14] Y. Liu, H. Yang, R. P. Dick, H. Wang, and L. Shang. Thermal vs energy optimization for dvfs-enabled processors in embedded systems. In *ISQED*, pages 204–209, 2007.

[15] S. Murali, A. Mutapcic, D. Atienza, R. Gupta, S. Boyd, and G. D. Micheli. Temperature-aware processor frequency assignment for mpsocs using convex optimization. In *CODES+ISSS*, pages 111–116, 2007.

[16] R. W. Pike. *Optimization for Engineering Systems*. Van Nostrand Reinhold Company, 2001.

[17] G. Quan, L. Niu, B. Mochocki, and X. S. Hu. Fixed-priority scheduling to reduce both the dynamic and leakage energy on variable voltage processors. *IJES*, 4:127 – 140, 2009.

[18] G. Quan and Y. Zhang. Leakage aware feasibility analysis for temperature-constrained hard real-time periodic tasks. *ECRTS*, pages 207–216, 2009.

[19] J. Rabaey, A. Chandrakasan, and B. Nikolic. *Digital Integrated Circuits: A Design Perspective*. Prentice Hall, 2003.

[20] C.-Y. Yang, J.-J. Chen, L. Thiele, and T.-W. Kuo. Energy-efficient real-time task scheduling with temperature-dependent leakage. In *DATE*, pages 9–14, 2010.

[21] L. Yuan, S. Leventhal, and G. Qu. Temperature-aware leakage minimization technique for real-time systems. In *ICCAD*, pages 761–764, 2006.

[22] S. Zhang and K. S. Chatha. Approximation algorithm for the temperature-aware scheduling problem. In *ICCAD*, pages 281–288, 2007.