Leakage-Aware Thermal Management for Multi-Core Systems Using Piecewise Linear Model Based Predictive Control

Xingxing Guo, Hai Wang*, Chi Zhang, He Tang, and Yuan Yuan
University of Electronic Science and Technology of China, Chengdu, China
*Corresponding author: wanghai@uestc.edu.cn

ABSTRACT

Performing thermal management on new generation IC chips is challenging. This is because the leakage power, which is significant in today's chips, is nonlinearly related to temperature, resulting in a complex nonlinear control problem in thermal management. In this paper, a new dynamic thermal management (DTM) method with piecewise linear (PWL) thermal model based predictive control is proposed to solve the nonlinear control problem. First, a PWL thermal model is built by combining multiple local linear thermal models expanded at several Taylor expansion points. These Taylor expansion points are carefully selected by a systematic scheme which exploits the thermal behavior property of the IC chips. Based on the PWL thermal model, a new predictive control method is proposed to compute the future power recommendation for DTM. By approximating the nonlinearity accurately with the PWL thermal model and being equipped with predictive control technique, the new DTM can achieve an overall high quality temperature management with smooth and accurate temperature tracking. Experimental results show the new method outperforms the linear model predictive control based method in temperature management quality with negligible computing overhead.

KEYWORDS

Thermal management, leakage power, multi-core, model predictive control.

1 INTRODUCTION

Power density keeps increasing with technology scaling, causing severe thermal related problems in high performance multi-core systems, including system reliability and performance degradation issues [4]. In order to find economical and efficient methods to solve the high temperature issue and improve both system performance and reliability, researchers have proposed dynamic thermal management (DTM) methods, which control the thermal behavior of multi-core systems by management actions including task migration and dynamic voltage & frequency scaling (DVFS) [5, 8, 12, 13].

However, most DTM methods do not consider leakage power, resulting in less accurate thermal management. For current high performance systems manufactured using new technology, leakage power, which even accounts for over 50% of the total power consumption, cannot be neglected anymore [6]. To make matters worse, leakage power depends on temperature exponentially, forming a positive feedback between power and temperature, which can lead to thermal runaway in the worst case. Therefore, the leakage power induced thermal problem has already become one of the most important limiting factors of IC system performance today.

The major challenge of considering leakage power in DTM lies in building an accurate leakage-aware thermal model which works well with DTM methods. This is because most DTM methods require linear thermal model, but the accurate leakage-aware thermal model is inherently nonlinear as aforementioned. There are very few existing leakage-aware DTM methods by using approximation based thermal models to replace the original nonlinear thermal model. For example, the methods in [3, 5, 13] use linear approximation thermal model. However, they suffer from low accuracy issue due to the large linear approximation error. Also, DTM with quadratic polynomial based approximation thermal model was introduced in [11]. Although it has higher accuracy than the linear approximation based methods, this DTM can only be used for single-core systems instead of multi-core systems as stated in [11].

The discussions above reveal that it is difficult to design an accurate leakage-aware DTM method for multi-core systems. In this work, we resolve this problem by proposing a leakage-aware DTM using piecewise linear (PWL) model based predictive control. The major contributions of this work include:

- In order to solve the nonlinear control problem in leakage-aware thermal management, we propose to use the PWL thermal model to approximate the original nonlinear thermal model. With the PWL thermal model, predictive control is enabled for leakage-aware DTM.
- We derived the formulation of the PWL thermal model for leakage-aware DTM. The PWL thermal model formulation is concise and can be integrated into the predictive control framework elegantly.
- A systematic Taylor expansion point selection scheme is developed for the PWL thermal model by exploiting the thermal behavior property of the IC chips.
- We show how to integrate the new PWL thermal model into the model predictive control (MPC) framework. With the PWL thermal model based predictive control, accurate future power recommendations can be computed for the multi-core system.
- We have experimentally compared the new DTM method with traditional DTM using linear thermal model based MPC. Our numerical results show the new method outperforms

ASP-DAC 2019, Jan. 2019, Tokyo, Japan

ACM ISBN 123-4567-24-567/08/06…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

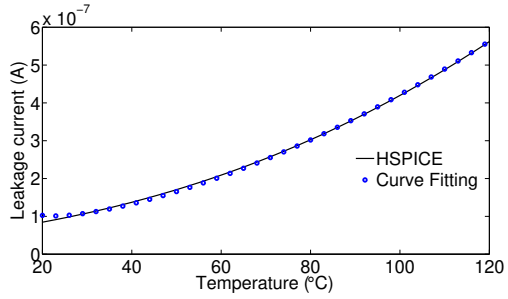

Figure 1: Comparison of leakage of a TSMC 65 nm process MOSFET from HSPICE simulation with its curve fitting result using (2).

traditional method in thermal management quality with negligible overhead introduced.

2 BACKGROUND

In this section, the leakage power model used in this work will be introduced first. After that, we briefly review thermal management using model predictive control (MPC) and reveal its problem for leakage-aware DTM.

2.1 Modeling of the leakage power

The total power of chip is composed of dynamic power p_d and leakage power p_s (which is also called static power). The dynamic power depends on the activity of the chip, and thus can be easily estimated by performance counter based methods [14]. Unlike dynamic power, leakage power p_s is independent of the chip's activity. Instead, it depends on the temperature of the chip, and can be modelled as [7, 10]

$$p_s = V_{dd} I_{leak} = V_{dd}(I_{sub} + I_{gate}), \tag{1}$$

$$\begin{aligned} I_{sub} &= K {v_T}^2 e^{\frac{V_{GS}-V_{th}}{\eta v_T}} \left(1 - e^{\frac{-V_{DS}}{v_T}}\right) \\ &\approx K {v_T}^2 e^{\frac{V_{GS}-V_{th}}{\eta v_T}}, \end{aligned} \tag{2}$$

where subthreshold current I_{sub} (which is highly related to temperature) and gate leakage current I_{gate} (which can be considered as a constant) are the main parts of leakage current I_{leak} [7, 10]. $v_T = \frac{kT_p}{q}$ is the thermal voltage and T_p is a scalar representing the temperature at one place,[1] K and η are process related parameters, and V_{th} is the threshold voltage. Apparently, the leakage power has a complex nonlinear relationship with temperature.

In order to see the accuracy of the leakage power model given in (1) and (2), Fig. 1 shows an HSPICE simulation result of leakage using TSMC 65 nm process model and its curve fitting result using approximate leakage model. From the figure, we can see that the leakage power model (1), (2) has high accuracy for all common temperatures of IC chips.

[1] T introduced latter in (3) is a vector representing temperatures at multiple positions.

2.2 Thermal management using model predictive control

In order to use model predictive control (MPC), a thermal model should be built first. For a l-core system with m total thermal nodes, we can get its thermal model as [5, 15]

$$\begin{aligned} GT(t) + C\frac{dT(t)}{dt} &= BP(T,t), \\ Y(t) &= LT(t), \end{aligned} \tag{3}$$

where $T(t) \in \mathbb{R}^m$ is the temperature vector (distinguished from scaler T_p), representing temperatures at m places of the chip and package; $G \in \mathbb{R}^{m\times m}$ and $C \in \mathbb{R}^{m\times m}$ contain equivalent thermal resistance and capacitance information respectively; $B \in \mathbb{R}^{m\times l}$ contains the power injection topology information; $P(T,t) \in \mathbb{R}^l$ is the power vector of l cores, including both dynamic power vector $P_d(t)$ and leakage power vector $P_s(T,t)$. $Y(t) \in \mathbb{R}^l$ is the output temperatures of l cores; $L \in \mathbb{R}^{l\times m}$ is the output selection matrix which selects the l core temperatures from $T(t)$.

In order to be used in computer, the thermal model (3) is discretized for a given time step h as [9]

$$\begin{aligned} T(k+1) &= AT(k) + DP_d(k) \\ &\quad + \int_0^h e^{-(h-\tau)C^{-1}G} C^{-1} B P_s(T,\tau)\,\mathrm{d}\tau, \end{aligned} \tag{4}$$

with

$$A = e^{-hC^{-1}G},\ D = \int_0^h e^{-(h-\tau)C^{-1}G} C^{-1} B\,\mathrm{d}\tau,$$

where k is the time in discrete form.[2] Note that $A \in \mathbb{R}^{m\times m}$ and $D \in \mathbb{R}^{m\times l}$ are *constant* matrices which are computed *offline* for a given time step h [9].

By using thermal model (4), MPC calculates the future power recommendation P_d in order to track a user defined temperature, with the following steps.

First, at current time k, we denote the future dynamic power trajectory (which is unknown and needs to be computed in the end) into the future N_c steps (where N_c is called the *control horizon* in MPC) as

$$\mathcal{P}_d = [P_d(k)^T, P_d(k+1)^T, \ldots, P_d(k+N_c-1)^T]^T. \tag{5}$$

Then, the prediction of core temperatures is written as

$$\mathcal{Y} = [Y(k+1)^T, Y(k+2)^T, \ldots, Y(k+N_p)^T]^T, \tag{6}$$

where N_p is called the *prediction horizon* (with $N_p > N_c$) in MPC and $Y(k+j)$ is the predicted temperatures at time $k+j$ using information of current time k.

Corresponding to (6), the target temperature vector $Y_g \in \mathbb{R}^l$ is written in a vector trajectory as

$$\mathcal{Y}_g = [Y_g^T, Y_g^T, \ldots, Y_g^T]^T. \tag{7}$$

The objective of the MPC is to bring the predicted output temperature $\mathcal{Y}$ as close as possible to the target temperature $\mathcal{Y}_g$ by adjusting the dynamic power $\mathcal{P}_d$, which is equivalent to minimizing the following cost function

$$\mathcal{J} = (\mathcal{Y}_g - \mathcal{Y})^T (\mathcal{Y}_g - \mathcal{Y}) + \mathcal{P}_d^T R \mathcal{P}_d. \tag{8}$$

[2] We use k to represent the discrete time, and t to represent the continuous time. $k+1$ is equivalent to $t+h$, with h as the discretization time step.

Here $R = rI_{N_c l \times N_c l}$ is tuning matrix with r as the tuning parameter. The regulation term $\mathcal{P}_d^T R \mathcal{P}_d$ is added to the cost function in (8) because we prefer power distribution not to change drastically for practical consideration [1, 16].

Next, optimization is performed to find the $\mathcal{P}_d$ which minimizes (8). However, because there is an integral with the nonlinear P_s in thermal model (4), we cannot express $\mathcal{Y}$ using P_d as the variable. Therefore, the optimization problem (8) cannot be solved to find the future power recommendation, meaning predictive control cannot be directly used for the leakage-aware thermal management.

3 LEAKAGE-AWARE DTM USING PIECEWISE LINEAR MODEL BASED PREDICTIVE CONTROL

In this section, we present the new leakage-aware DTM method using PWL model based predictive control.

3.1 Building local linear thermal model using Taylor expansion

Before presenting the PWL methods, we first show the formulation of the local linear thermal model (at a Taylor expansion point) which will be used in PWL approximation.

First, we can get a local linear leakage power model by performing Taylor expansion on the original nonlinear model (1), (2), expressed in matrix-vector form as

$$P_s = \widehat{P} + \widehat{H}T, \tag{9}$$

where $\widehat{P} \in \mathbb{R}^l$ is a constant vector not associated with temperature T. $\widehat{H} \in \mathbb{R}^{l \times m}$ is a constant rectangular diagonal matrix. Due to the page limitation, please refer to [15] for the detailed derivation.

Then, by integrating (9) into (3) and letting $\widehat{G} = G - B\widehat{H}$, we obtain a local linear thermal model as

$$\begin{aligned} \widehat{G}T(t) + C\frac{dT(t)}{dt} &= B(P_d(t) + \widehat{P}), \\ Y(t) &= LT(t). \end{aligned} \tag{10}$$

Similar to (4), the local linear thermal model (10) can be discretized into the following form but without the integral term in (4):

$$T(t+h) = \widehat{A}(h)T(t) + \widehat{D}(h)P_d + \widehat{D}(h)\widehat{P}, \tag{11}$$

where

$$\widehat{A}(h) = e^{-hC^{-1}\widehat{G}}, \ \widehat{D}(h) = \int_0^h e^{-(h-\tau)C^{-1}\widehat{G}} C^{-1} B \, \mathrm{d}\tau.$$

3.2 PWL thermal model formulation

In this part, we formulate the PWL thermal model using the local linear thermal model presented in Section 3.1. The PWL thermal model can then be integrated into the predictive control framework for leakage-aware DTM.

3.2.1 Taylor expansion thermal points selection scheme for leakage-aware DTM. Although there is PWL approximation based leakage-aware thermal estimation method [15], it is not straightforward to apply similar PWL approximation to DTM due to the difficulty in Taylor expansion thermal points selection. In thermal estimation problem, the Taylor expansion point can be easily chosen by using the self-estimated temperature or the on-chip thermal sensor temperature [15]. However, DTM will not know the proper Taylor expansion points directly, because its computing target is the future power recommendation, not the temperature. The only things that DTM knows are the current temperature, the target temperature, and also the fact that the temperature prediction trajectory (excited by the unknown future power recommendation to be computed) should be between the two temperatures. In this work, we propose a novel Taylor expansion points selection scheme as the following.

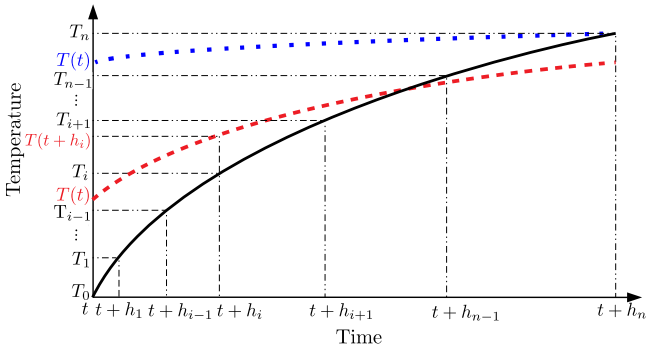

Figure 2: The sketch map of the PWL method for one control step. $T_1, T_2, \ldots, T_n$ are the potential Taylor expansion points. $t, t+h_1, \ldots, t+h_n$ are the potential local linear model switching time points. The black solid line is the extreme temperature trajectory. The red dashed line is a common temperature trajectory. The blue dot line represents the temperature trajectory which is already very close to the target at time t.

First, we define two thermal management cases called *rising case* and *falling case*, depending on the current temperature of the core. We have the falling case if the current temperature is higher than the target temperature. DTM should lower the core temperature to target temperature for reliability in this case. Otherwise, we have the rising case for performance. Here we use the rising case as illustration example. Please note that DTM for the falling case can be performed in the same way.

Let us denote T_0 as the lowest temperature and T_n as the target temperature of the chip.[3] The operating temperature of rising case lies between T_0 and T_n. We introduce n potential expansion points in the operating temperature range: $\{T_1, T_2, \ldots, T_n\}$.[4] For simplicity, assume all the potential expansion points are uniformly placed in the operating temperature range, i.e., $T_i - T_{i-1} = \frac{T_n - T_0}{n}$ for any integer $i \in [1, n]$, as shown in Fig. 2.

Next, corresponding to the Taylor expansion points, we also need to determine the potential local model switching time points $\{t, t+h_1, \ldots, t+h_n\}$ within one control step. The extreme temperature trajectory in the rising case, which starts from $T(t) = T_0$ and ends at $T(t+h_n) = T_n$ is used to determine these time points. As shown in Fig. 2, the extreme temperature trajectory is the solid

[3] Usually, the lowest temperature is set to be the same or slightly higher than the ambient temperature.

[4] Please note that T_0 is *not* a potential Taylor expansion point.

black line, and the switching time point $t + h_i$ is chosen as the one which satisfies $T(t + h_i) = T_i$ for this trajectory.

PWL approximation will be performed by using the linear local thermal models constructed using some of these potential Taylor expansion points $\{T_1, T_2, \ldots, T_n\}$ switched only at the corresponding switching time points $\{t, t + h_1, \ldots, t + h_n\}$ as shown in the next part.

3.2.2 PWL thermal model for temperature prediction. If the current temperature $T(t)$ lies between T_{i-1} and T_i, the DTM thermal prediction trajectory should look like the red dashed line in Fig. 2 exited by the future power recommendation (which is unknown and need to be computed).[5] For this trajectory, $T(t + h_n)$ can be represented[6] in the following way.

First, $T(t + h_i)$ (shown as red font in Fig. 2) is represented using the local linear model (11) expanded at T_i as

$$T(t + h_i) = \widehat{A}_i T(t) + \widehat{D}_i P_d + \widehat{D}_i \widehat{P}_i, \tag{12}$$

where $\widehat{A}_i = \widehat{A}(h_i)$, $\widehat{D}_i = \widehat{D}(h_i)$, and $\widehat{P}_i$ are the local linear thermal matrices in (11) with T_i as the expansion point.

Then, temperatures at the subsequent time points $t + h_{i+1}$, $t + h_{i+2}, \ldots, t + h_n$ are represented iteratively by using the local linear thermal models expanded at time points $T_{i+1}, T_{i+2}, \ldots, T_n$, respectively, as the following:

$$\begin{aligned} T(t + h_{i+1}) &= \widehat{A}_{i+1} T(t + h_i) + \widehat{D}_{i+1} P_d + \widehat{D}_{i+1} \widehat{P}_{i+1}, \\ T(t + h_{i+2}) &= \widehat{A}_{i+2} T(t + h_{i+1}) + \widehat{D}_{i+2} P_d + \widehat{D}_{i+2} \widehat{P}_{i+2}, \\ &\vdots \\ T(t + h_n) &= \widehat{A}_n T(t + h_{n-1}) + \widehat{D}_n P_d + \widehat{D}_n \widehat{P}_n, \end{aligned} \tag{13}$$

where $\widehat{A}_j = \widehat{A}(h_j - h_{j-1})$ and $\widehat{D}_j = \widehat{D}(h_j - h_{j-1})$ for $j = i + 1, i + 2, \ldots, n$.

Finally, the temperature at the end of the control step $(t + h_n)$ is expressed by combining the equations above as

$$T(t + h_n) = \widehat{\mathcal{A}} T(t) + \widehat{\mathcal{D}} P_d + \widehat{\mathcal{D}}_i \widehat{P}_i + \cdots + \widehat{\mathcal{D}}_n \widehat{P}_n, \tag{14}$$

where $\widehat{\mathcal{A}} = \widehat{A}_n \widehat{A}_{n-1} \cdots \widehat{A}_i$, $\widehat{\mathcal{D}} = \widehat{A}_n \widehat{A}_{n-1} \cdots \widehat{A}_{i+1} \widehat{D}_i + \widehat{A}_n \widehat{A}_{n-1} \cdots \widehat{A}_{i+2} \widehat{D}_{i+1} + \cdots + \widehat{D}_n$, and $\widehat{\mathcal{D}}_i = \widehat{A}_n \widehat{A}_{n-1} \cdots \widehat{A}_{i+1} \widehat{D}_i$.

In order to be compatible with MPC, we rewrite (14) into the discrete form as

$$\begin{aligned} T(k + 1) &= \widehat{\mathcal{A}} T(k) + \widehat{\mathcal{D}} P_d(k) + \widehat{\mathcal{D}}_i \widehat{P}_i + \cdots + \widehat{\mathcal{D}}_n \widehat{P}_n, \\ Y(k + 1) &= LT(k + 1). \end{aligned} \tag{15}$$

We call this newly formulated thermal model (15) as the *PWL thermal model*. The PWL thermal model matrices will be computed offline after Taylor expansion points selection.

Now, we have successfully approximated the original nonlinear temperature prediction using the PWL thermal model in (15). Next, we will demonstrate how to formulate the PWL thermal model based predictive control by replacing the original nonlinear thermal prediction (4) with the PWL model based thermal prediction (15).

[5] The end point of the red dashed line is slightly off the target temperature, because the regulation term $\mathcal{P}_d^T R \mathcal{P}_d$ is added to the cost function $\mathcal{J}$ in (8).

[6] $T(t + h_n)$ is represented, but not computed, because P_d (the future power recommendation) is the actual unknown to be calculated.

3.3 PWL model based predictive control

With the PWL thermal model (15), MPC should be able to calculate the power recommendation P_d to track a user defined output temperature as presented in this part.

By analyzing the MPC mechanism, we know the future temperature prediction trajectory can be described as the following. For the first control time step[7] into the future, the temperature prediction trajectory is similar to the red dashed line in Fig. 2, because the power recommendation will bring the temperature toward the target temperature. Assume the temperature prediction is close to the target temperature at time $k + 1$, then at time $k + j$, where $j = 2, 3, \ldots, N_p$, all temperature prediction trajectories should look like the blue dot line in Fig. 2.

With the observation above, for N_p steps temperature prediction into the future (from k to $k + N_p$), we only need to use the temperature prediction with multiple Taylor expansion points at the first control step (from time k to $k + 1$) expressed by the PWL thermal model (15).

For the rest of the control steps (from $k + 1$ to $k + N_p$), only one segment of the PWL thermal model (11) is needed with target temperature Y_g (which equals to T_n) as the expansion point. The temperature predictions for these steps are expressed by simply changing h_i to h_n in (12) and write it into discrete form as

$$\begin{aligned} T(k + j) &= \widehat{A}_n T(k + j - 1) + \widehat{D}_n P_d(k + j - 1) + \widehat{D}_n \widehat{P}_n, \\ Y(k + j) &= LT(k + j), \end{aligned} \tag{16}$$

where $j = 2, 3, \ldots, N_p$, and $P_d(k + j - 1) = \mathbf{0}$ for $j > N_c$.

Combining equations (15) and (16), we can get the predicted temperature trajectory $\mathcal{Y}$ as

$$\mathcal{Y} = FT(k) + V\mathcal{P}_d + \phi_1 \widehat{\mathcal{P}} + \phi_2 \widehat{\mathcal{P}}_n, \tag{17}$$

where $\widehat{\mathcal{P}} = [\widehat{P}_i^T, \widehat{P}_{i+1}^T, \ldots, \widehat{P}_n^T]^T$, $\widehat{\mathcal{P}}_n = [\mathbf{0}^T, \widehat{P}_n^T, \ldots, \widehat{P}_n^T]^T$,

$$F = \begin{bmatrix} L\widehat{\mathcal{A}} \\ L\widehat{A}_n \widehat{\mathcal{A}} \\ \vdots \\ L\widehat{A}_n^{N_p - 1} \widehat{\mathcal{A}} \end{bmatrix},$$

$$V = \begin{bmatrix} L\widehat{\mathcal{D}} & \mathbf{0} & \cdots & \mathbf{0} \\ L\widehat{A}_n \widehat{\mathcal{D}} & L\widehat{D}_n & \cdots & \mathbf{0} \\ L\widehat{A}_n^2 \widehat{\mathcal{D}} & L\widehat{A}_n \widehat{D}_n & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ L\widehat{A}_n^{N_p - 1} \widehat{\mathcal{D}} & L\widehat{A}_n^{N_p - 2} \widehat{D}_n & \cdots & L\widehat{A}_n^{N_p - N_c} \widehat{D}_n \end{bmatrix},$$

$$\phi_1 = \begin{bmatrix} L\widehat{\mathcal{D}}_i & L\widehat{\mathcal{D}}_{i+1} & \cdots & L\widehat{\mathcal{D}}_n \\ L\widehat{A}_n \widehat{\mathcal{D}}_i & L\widehat{A}_n \widehat{\mathcal{D}}_{i+1} & \cdots & L\widehat{A}_n \widehat{\mathcal{D}}_n \\ L\widehat{A}_n^2 \widehat{\mathcal{D}}_i & L\widehat{A}_n^2 \widehat{\mathcal{D}}_{i+1} & \cdots & L\widehat{A}_n^2 \widehat{\mathcal{D}}_n \\ \vdots & \vdots & \ddots & \vdots \\ L\widehat{A}_n^{N_p - 1} \widehat{\mathcal{D}}_i & L\widehat{A}_n^{N_p - 1} \widehat{\mathcal{D}}_{i+1} & \cdots & L\widehat{A}_n^{N_p - 1} \widehat{\mathcal{D}}_n \end{bmatrix},$$

[7] Please note that the duration from t to $t + h_n$ in Fig. 2 equals to only one control step in MPC (for example, from k to $k + 1$, or from $k + j - 1$ to $k + j$).

C11	C12	C13	C14
C21	C22	C23	C24
C31	C32	C33	C34
C41	C42	C43	C44

Figure 3: The configuration of the 16-core chip used for the experiment.

$$\phi_2 = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & L\widehat{D}_n & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & L\widehat{A}_n\widehat{D}_n & L\widehat{D}_n & \cdots & \mathbf{0} \\ \mathbf{0} & L\widehat{A}_n^2\widehat{D}_n & L\widehat{A}_n\widehat{D}_n & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & L\widehat{A}_n^{N_p-2}\widehat{D}_n & L\widehat{A}_n^{N_p-3}\widehat{D}_n & \cdots & L\widehat{A}_n^{N_p-N_c}\widehat{D}_n \end{bmatrix},$$

with $\mathbf{0}$ as the zero matrix with suitable size.

Plugging (17) into (8), standard MPC optimization is performed to minimize (8) by making the first derivative of (8) (with respect to $\mathcal{P}_d$) equal to zero. The solution of $\mathcal{P}_d$ is

$$\mathcal{P}_d = (V^T V + R)^{-1} V^T (\mathcal{Y}_g - FT(k) - \phi_1 \widehat{\mathcal{P}} - \phi_2 \widehat{\mathcal{P}}_n). \tag{18}$$

At each MPC time k, only $P_d(k)$ (the first element of $\mathcal{P}_d$) will be outputted as the power recommendation for thermal management. Frequencies and task loads of the multi-core system will be adjusted according to $P_d(k)$. How to perform the management actions based on future power recommendation is presented in many DTM works such as [13], which will not be given here due to page limitation.

4 EXPERIMENTAL RESULTS AND DISCUSSIONS

In this section, we evaluate the performance of the new leakage-aware DTM method.

4.1 Experiment setup

The experiment is performed on a 16-core system plant with its chip configuration shown in Fig. 3. We place one thermal sensor for each core, which provides on-chip temperature information for the DTM. The ambient temperature is 40 °C, and the target temperature in DTM is 80 °C. We set the operating temperature range for the rising case as from 40 °C to 80 °C, and the range for the falling case as from 120 °C to 80 °C. All the experiments are performed on a PC with an Intel Core i5-2400 CPU and 4 GB memory.

Power estimator Wattch [2] is used to generate the dynamic power by running the standard SPEC benchmarks. The different power traces from SPEC benchmarks are randomly assigned to different cores of the multi-core system. The golden leakage power of the multi-core system plant is obtained by using the iteration based leakage-aware thermal simulation method with simulation step 1 ms to ensure accuracy. The control step of DTM is set as 1 s.

In order to show the advantage of the new DTM method with PWL model based predictive control (we call it the *new DTM*), we compare it with the traditional DTM method with linear model based MPC [13] (called the *traditional DTM*). The traditional DTM shares the same settings as the new DTM except that its only Taylor expansion point is set at the target temperature (80 °C).

4.2 Performance evaluation of leakage-aware DTM with PWL model based predictive control

We apply both the new DTM and traditional DTM, and record the management performance results in Table 1. In order to see the accuracy of the new DTM with different configurations, we test it with different expansion point number, prediction horizon length N_p, and control horizon length N_c.

We mainly focus on two DTM performances in the comparison. The first is the temperature tracking difference between the actual plant temperature and the target temperature, which indicates the effectiveness and accuracy of the DTM. The second is the overhead (computing time and memory cost) of the DTM, with respect to different numbers of expansion points as well as different N_p and N_c.

For traditional DTM, the difference between the actual temperature and the target temperature is large for all cases as shown in Table 1. Even for the best case ($N_c = 1$, $N_p = 2$), the average difference is over 1.3 °C and the maximum difference is around 6 °C. This is because the linear model cannot approximate the nonlinearity in leakage power accurately.

On the contrary, for the new DTM, the temperature tracking difference is much smaller than the traditional DTM for all cases. The tracking accuracy improvement is achieved by approximating the nonlinearity accurately using the PWL thermal model. Especially, the average tracking difference is only 0.72 °C when the number of expansion points is 11.

On the runtime side, we observe that the computing time of the new DTM is only a little higher than that of the traditional method, because the new DTM has one more term (ϕ_1) than the traditional DTM. But still, the new DTM is extremely fast, with only less than 1.2 ms computing time for each 1 s control step. Since the computation is performed on one core out of the 16 cores, this overhead only leads to around 0.01% system throughput degradation.

Memory cost of the new DTM is higher than traditional DTM, which also increases linearly with the expansion point number as shown in Table 1. This is because more matrices computed offline need to be stored, such as the PWL thermal model matrices F and ϕ_1. It is the major trade-off between accuracy and overhead in the new DTM. For practical usage, engineers need to balance the accuracy and memory cost by choosing the proper Taylor expansion point number.

Finally, we plot the transient plant temperature comparison results in Fig. 4 by activating both DTMs at 1 s. We only plot the results of core C32 due to page limitation. It is observed that the temperature controlled by traditional DTM shows large tracking overshot especially when the current temperature is far from target (from 1 s to 2 s). On the other hand, the temperature controlled by the new DTM tracks the target accurately. This clearly demonstrates the advantage of the new DTM in thermal management quality.

Table 1: Computing time (time), storage memory (mem), and tracking difference (difference) comparison results of the traditional DTM and new DTM. Computing time is recorded as the average computing time for each thermal management action (every 1 s). The tracking difference is in °C.

Methods	$N_c = 1, N_p = 1$				$N_c = 1, N_p = 2$				$N_c = 1, N_p = 3$				$N_c = 2, N_p = 3$			
	time	mem	difference		time	mem	difference		time	mem	difference		time	mem	difference	
	(ms)	(KB)	max	avg	(ms)	(KB)	max	avg	(ms)	(KB)	max	avg	(ms)	(KB)	max	avg
Traditional	1.01	7	6.02	1.35	1.13	13	5.97	1.32	1.22	21	5.86	1.34	1.30	23	5.94	1.37
New (3 points)	1.12	14	1.25	0.84	1.28	23	1.20	0.81	1.39	42	1.21	0.82	1.46	44	1.23	0.85
New (5 points)	1.12	22	1.21	0.79	1.28	36	1.16	0.78	1.39	65	1.18	0.80	1.47	67	1.20	0.82
New (7 points)	1.13	32	1.15	0.77	1.29	51	1.12	0.75	1.40	91	1.09	0.76	1.47	94	1.13	0.79
New (11 points)	1.14	58	1.08	0.75	1.30	90	1.05	0.72	1.42	150	1.11	0.74	1.49	154	1.09	0.75

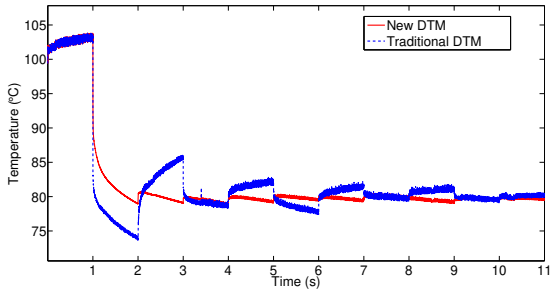

Figure 4: The transient temperatures of C32 using the new DTM (with 5 expansion points) and traditional DTM. Both methods have $N_P = 2$ and $N_c = 1$. The temperature target is set as 80 °C. Both DTM methods are activated at 1 s.

In summary, experimental results show that the new DTM outperforms the traditional DTM in thermal management quality with negligible computing overhead. Trade-off between accuracy and memory cost of the new DTM can be made by adjusting the Taylor expansion point number.

5 CONCLUSION

In this paper, we have proposed a new leakage-aware DTM method for multi-core systems using PWL model based predictive control. We built a PWL thermal model by combining multiple local linear thermal models expanded at several Taylor expansion points. These expansion points are selected by a systematic scheme which exploits the thermal behavior property of the IC chips. Based on the PWL thermal model, predictive control is used to find the optimal future power recommendations for thermal management. Experimental results show the new method outperforms the linear model based MPC method in temperature management quality with negligible computing overhead.

ACKNOWLEDGMENTS

This research is supported in part by National Natural Science Foundation of China under grant No. 61404024, in part by the Fundamental Research Funds for the Central Universities under grant No. ZYGX2016J043, in part by the Scientific Research Foundation for the Returned Overseas Chinese Scholars, State Education Ministry.

REFERENCES

[1] Stephen Boyd and Lieven Vandenberghe. 2006. *Convex Optimization.* Cambridge University Press.
[2] David Brooks, Vivek Tiwari, and Margaret Martonosi. 2000. Wattch: A Framework for Architectural-Level Power Analysis and Optimizations. In *Proc. Int. Symp. on Computer Architecture (ISCA).* 83–94.
[3] Vivek Chaturvedi, Huang Huang, and Gang Quan. 2010. Leakage Aware Scheduling on Maximum Temperature Minimization for Periodic Hard Real-Time Systems. In *IEEE International Conference on Computer and Information Technology.* 1802–1809.
[4] Hadi Esmaeilzadeh, Emily Blem, Renee St. Amant, Karthikeyan Sankaralingam, and Doug Burger. 2012. Dark silicon and the end of multicore scaling. *IEEE MICRO* 32, 3 (May 2012), 122–134.
[5] Vinay Hanumaiah, Sarma Vrudhula, and Karam Chatha. 2011. Performance optimal online DVFS and task migration techniques for thermally constrained multicore processors. *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems* 30, 11 (November 2011), 1677–1690.
[6] John L Hennessy and David A Patterson. 2012. *Computer Architecture: A Quantitative Approach, Fifth Edition.* Elsevier.
[7] Yongpan Liu, Robert Dick, Li Shang, and Huazhong Yang. 2007. Accurate Temperature-Dependent Integrated Circuit Leakage Power Estimation is Easy. In *Proc. European Design and Test Conf. (DATE).* 1–6.
[8] Zao Liu, Sheldon X.-D. Tan, Xin Huang, and Hai Wang. 2015. Task migrations for distributed thermal management considering transient effects. *IEEE Trans. on Very Large Scale Integration (VLSI) Systems* 23, 2 (Feb. 2015), 397–401.
[9] Efraim Rotem, Alon Naveh, Doron Rajwan, Avinash Ananthakrishnan, and Eliezer Weissmann. 2007. TILTS: A Fast Architectural-Level Transient Thermal Simulation Method. *Journal of Low Power Electronics* 3, 1 (April 2007), 13–21.
[10] Ruijing Shen, Sheldon X.-D. Tan, Hai Wang, and Jinjun Xiong. 2012. Fast Statistical Full-Chip Leakage Analysis for Nanometer VLSI Systems. *ACM Trans. on Design Automation of Electronics Systems* 17, 4 (October 2012), 51:1–51:19.
[11] Bing Shi and Ankur Srivastava. 2015. Dynamic Thermal Management Considering Accurate Temperature-Leakage Interdependency. In *ENCYCLOPEDIA OF THERMAL PACKAGING: Thermal Packaging Tools.* World Scientific, 39–60.
[12] Cheng Tan, Thannirmalai Muthukaruppan, Tulika Mitra, and Lei Ju. 2015. Approximation-Aware Scheduling on Heterogeneous Multi-core Architectures. In *Proc. Asia South Pacific Design Automation Conf. (ASP-DAC).* 618–623.
[13] Hai Wang, Jian Ma, Sheldon X.-D. Tan, Chi Zhang, He Tang, Keheng Huang, and Zhenghong Zhang. 2016. Hierarchical Dynamic Thermal Management Method for High-Performance Many-Core Microprocessors. *ACM Trans. on Design Automation of Electronics Systems* 22, 1 (July 2016), 1:1–1:21.
[14] Hai Wang, Sheldon X.-D. Tan, Xue-Xin Liu, and Ashish Gupta. 2012. Runtime power estimator calibration for high-performance microprocessors. In *Proc. European Design and Test Conf. (DATE).* 352–357.
[15] Hai Wang, Jiachun Wan, Sheldon X.-D. Tan, Chi Zhang, He Tang, Yuan Yuan, Keheng Huang, and Zhenghong Zhang. 2018. A Fast Leakage-Aware Full-Chip Transient Thermal Estimation Method. *IEEE Trans. on Computers* 67, 5 (May 2018), 617–630.
[16] Liuping Wang. 2009. *Model Predictive Control System Design and Implementation Using MATLAB.* Springer.