An Effective and Accurate Data-Driven Approach for Thermal Simulation of CPUs

Lin Jiang
Department of Electrical & Computer Engineering
Clarkson University
Potsdam, NY, USA
jiangl2@clarkson.edu

Yu Liu
Department of Electrical & Computer Engineering
Clarkson University
Potsdam, NY, USA
yuliu@clarkson.edu

Ming-C. Cheng
Department of Electrical & Computer Engineering
Clarkson University
Potsdam, NY, USA
mcheng@clarkson.edu

***Abstract*—A data-driven reduced-order approach is applied to develop a thermal simulation model for a quad-core CPU, AMD ATHLON II X4 610e. The model is based on proper orthogonal decomposition (POD) that projects the physical domain of the CPU onto a functional space represented by a small set of basis functions (or modes). To generate an optimal set of modes, these modes are trained by thermal solution data collected from numerical simulation tools, FEniCS and HotSpot Grid. If good-quality data are used in the training process, the process optimizes the POD modes that are then able to offer a very accurate simulation model with a very small numerical degree of freedom (DoF). Each of the developed POD models is verified against its simulation tool used in its training. A very accurate prediction is observed in the POD model derived from FEniCS with a reduction in the numerical DoF by nearly 5 orders of magnitude, which amounts to more than a 3-order reduction in computing time. The POD model derived from HotSpot Grid is however not able to offer accurate simulation due to its inadequate data quality.**

Keywords—Thermal simulation of CPUs, proper orthogonal decomposition, data-driven learning, FEniCS, HotSpot

I. INTRODUCTION

With the miniaturization and multi-functional design for CPUs, the numbers of transistor and power dissipation per area have been increasing dramatically in the last several decades. The continuing advance in CPU technology has generated much more joule heat and led to sever temperature escalation and hot-spot generation in CPUs. High temperature and hot spots not only impair the CPUs performance but also reduce its lifetime [1]-[4] because of thermal stress enhanced by the material property mismatch and resulting in electromigration in the interconnects [5]-[7]. Lower temperature operation not only maintains higher performance and longer lifetime, but also reduces the cooling costs of CPUs. To decrease the temperature and suppress the hot spots in CPUs, thermal-aware task scheduling is desired, which however relies on effective thermal management [8],[9]. To achieve effective thermal management of CPUs, an efficient and accurate thermal simulation is needed.

There exist several different approaches for thermal simulations of CPUs or semiconductor integrated circuits (ICs) with different levels of accuracy and efficiency. For situations that require both detailed and accurate thermal profiles, direct numerical simulations (DNSs), based on either the finite element or finite difference method, are usually needed. These approaches however demand extensive computational efforts, especially in a large domain structure like a CPU. It is therefore prohibitive to apply DNS to predict the detailed temperature distribution and hot spot locations in a CPU, and DNSs are only suitable for small structures where more detailed thermal profiles may be needed. There are many commercial or open-source tools that offer such DNSs. For example, FEniCS [10] is an open-source DNS platform based on the finite element method (FEM) that can be applied to predict detailed thermal distributions in semiconductor ICs.

Due to intensive computing time needed in DNSs, the lumped RC thermal circuit model has always been used for the thermal-profile prediction in large-scale semiconductor chips. For example, the Block model of HotSpot [11]-[13] (or HotSpot-Block hereafter) is one of the most popular thermal simulators at the architecture level for CPUs using the compact RC thermal model. The compact RC thermal circuit model is used to describe the heat flow as "current" passing through "thermal resistance" and "thermal capacitance" based on the analogy between the charge and heat flows. In the RC thermal circuit model, spatial details are lumped into thermal elements (capacitances and resistances), and spatial locations are represented by the thermal nodes. Using the efficient thermal circuit model, HotSpot Block does not offer fine-enough resolution to capture the hot spots in CPUs and cannot appropriately account for distributed heat transfer in CPUs, particularly for the lumped RC elements with large aspect ratios. In spite of the efficiency of HotSpot-Block , the accuracy of the HotSpot-Block thermal circuit model has been challenged in [14] which claimed that an error exceeding 200% could appear in HotSpot-Block with some floorplans compared to DNS. To improve the accuracy of HotSpot, the Block model was improved and the Grid model was developed [15], in which each functional unit can be divided into several smaller elements with each element as a thermal node. Therefore, to predict hot spots in a semiconductor IC accurately using HotSpot, the Grid model with fine-enough meshes needs to be used, which basically performs DNS, and HotSpot becomes time-consuming as well.

As the multi-core architecture becomes standard in modern CPU design, the thermal issues in CPUs have been enhanced significantly in recent years and the prediction of high temperature gradients and hot-spot locations in CPUs has become crucial. To be able to predict thermal profiles effectively in such complex and large structures, an approach beyond the DNSs or the RC thermal model is needed. The desired approach

must be as efficient as the RC thermal circuit model and as accurate as the DNS. In addition, it needs to be able to offer fine-enough resolution to capture critical hot spots in CPUs.

In this work, a thermal simulation technique developed recently [16]-[19] enabled by the data-driven proper orthogonal decomposition (POD) [20],[21] is adopted for CPU thermal simulation. This data-driven learning technique is applied to develop a POD thermal model for a quad-core CPU, AMD ATHLON II X4 610e CPU [22]. The temperature profile in this CPU is predicted using this POD approach compared against FEniCS and the HotSpot Grid model (hereafter as HotSpot-Grid). The POD is one of the reduced-order methods and is able to effectively reduce the numerical degree of freedom (DoF) by projecting the problem of interest from its physical domain onto a functional space described by a small number of basis functions. Unlike many other projection-based methods where the basis functions are usually assumed, the POD method extracts its orthogonal basis functions (or called POD modes) from processing the solution data of the physical structure obtained by DNSs [16]-[19],[23].

In this study, FEniCS (FEM) and HotSpot-Grid (a thermal circuit model with very small elements) are used to perform dynamic thermal simulations to collect two separate sets of temperature solution data of AMD ATHLON II X4 610e under its normal operating condition. By processing these data sets separately, two sets of POD modes are extracted from each set of the FEniCS and HotSpot-Grid data. Using this "training" process (including the data collection and mode generation), the POD modes are tailored to the normal operating condition in the AMD ATHLON II X4 610e CPU and are optimal in the least squares sense to predict its dynamic temperature distribution. This study shows that the developed POD model for this quad-core CPU could offer an accurate prediction of its dynamic thermal distribution with a reduction in the numerical DoF by nearly 5 orders of magnitude if good-quality data are used in the training.

II. Thermal Simulation Method based on POD

Temperature in space and time $T(\vec{r},t)$ can be represented by a linear combination of the selected basis functions φ_i as

$$T(\vec{r},t) = \sum_{i=1}^{M} a_i(t)\varphi_i(\vec{r}), \tag{1}$$

where φ_i is the POD mode and M is the number of basis functions (or modes) selected to represent the temperature, which determines the accuracy and efficiency of the POD approach. The time-dependent parameter $a_i(t)$ is the coefficient of its POD mode.

The POD process can be applied to optimize the POD modes by maximizing the mean square inner product of the thermal solution with the modes, which can be expressed as

$$\frac{\langle\left(\int_\Omega T(\vec{r},t)\,\varphi d\Omega\right)^2\rangle}{\int_\Omega \varphi^2\, d\Omega}, \tag{2}$$

where Ω is the physical domain of the selected structure and brackets $\langle\rangle$ denote the average over a data ensemble of temperature. In this study, the average indicates the temporal average over many samples in time, subjected to an operation setting specified by the boundary conditions (BCs) and power sources. The Fredholm equation shown below is obtained from the maximization problem in (2) for the POD modes,

$$\int_{\vec{r}'} \mathbf{R}\left(\vec{r},\vec{r}'\right)\cdot\vec{\varphi}\left(\vec{r}'\right)d\vec{r}' = \lambda\vec{\varphi}\,(\vec{r})\ , \tag{3}$$

where $\mathbf{R}\left(\vec{r},\vec{r}'\right)$ is a two-point correlation tensor expressed as

$$\mathbf{R}\left(\vec{r},\vec{r}'\right) = \langle T(\vec{r},t)T(\vec{r}',t)\rangle\,. \tag{4}$$

The maximization process in (2) thus leads to an eigenvalue problem given in (3) with λ as the eigenvalue and $\vec{\varphi}\,(\vec{r})$ as the eigenvector. In this work, two sets of temperature data $T(\vec{r},t)$ for the selected structure in (4) are obtained, one from FEniCS and the other from HotSpot-Grid. The snapshot method [24],[25] is applied to solve (3) for the eigenvalues and POD modes.

Once the POD modes are determined, the heat conduction equation can be projected onto a functional space represented by POD modes using the Galerkin projection,

$$\int_\Omega \left(\varphi_i(\vec{r})\frac{\partial\rho CT}{\partial t} + \nabla\varphi_i\cdot k\nabla T\right)d\Omega = \int_\Omega \varphi_i(\vec{r})\cdot P_d(\vec{r},t)d\Omega - \int_S \varphi_i(\vec{r})(-k\nabla T\cdot\vec{n})\,dS, \tag{5}$$

where k is thermal conductivity, ρ is the density, C is the specific heat, $P_d(\vec{r},t)$ is the power density, S is the boundary surface of the selected domain and $\vec{n}$ is the outward normal vector of boundary surface. With given POD modes, (5) can be reduced to an M-dimensional ordinary differential equation (ODE) for $a_i(t)$,

$$\sum_{j=1}^{M} c_{i,j}\frac{da_j}{dt} + \sum_{j=1}^{M} g_{i,j}\,a_j = P_i,\ \ i = 1\ \text{to}\ M, \tag{6}$$

where P_i representing the last 2 terms of (5) is the power density dissipated in the POD space and can be pre-evaluated since the shape of power density is predefined. $c_{i,j}$ and $g_{i,j}$ are the elements of thermal capacitance and thermal conductance matrices in the POD space and defined as

$$c_{i,j} = \int_\Omega \rho C\vec{\varphi}_i\vec{\varphi}_j\, d\Omega\,, \quad g_{i,j} = \int_\Omega k\nabla\vec{\varphi}_i\nabla\vec{\varphi}_j\, d\Omega. \tag{7}$$

Once a_j is determined from (6), the temperature solution can be evaluated from (1).

The POD has been known for offering an effective approach to simulate dynamic and steady-state behaviors of physical quantities with high accuracy [16]-[21], [23]-[25]. In this study,

we demonstrate that the POD concept can also be used to verify the quality of the collected DNS data and thus quality of the simulation tool used to generate the DNS data. The characteristics of the POD modes generated from (3) using the data collected from DNSs are fully determined by these data. To derive a POD thermal model given in (6), these modes are also applied to project the heat conduction equation (the same equation used in the DNS) onto the functional space represented by these modes. To result in an accurate POD model, the collected data and the heat conduction equation must be consistent. In cases when numerical settings are not rigorous enough (such as coarse meshes) or serious approximations are included in the DNSs, the POD model may not provide an accurate prediction of the thermal distribution even if a large number of POD modes are included. The demonstrations of the POD method illustrate not only the accuracy and efficiency of the POD thermal simulation of a CPU but also the quality of data generated by FEniCS and HotSpot-Grid.

III. Application to Thermal Simulation of a CPU

A quad-core CPU, AMD ATHLON II X4 610e [22], is selected in this study to demonstrate the POD thermal simulation model. As shown in Fig. 1 for the floorplan, this quad-core processor includes the following units: four 512KB L2 caches, a northbridge in the center and I/O and DDR3 placed around the periphery. In the DNSs for collecting thermal data from FEniCS and HotSpot-Grid, dynamic power density in the device layer of each unit is assumed uniform. The simulation domain of this processor covers a volume of 14mm × 12mm × 650μm in the *x, y* and *z* directions, respectively. All the surfaces of the chip, except for the bottom surface, are assumed adiabatic. The convection BC is implemented on the bottom with a constant heat transfer coefficient and an ambient temperature of 45°C, similarly to the settings implemented in HotSpot with a further assumption that the temperature in each layer of the heat spreader, interface material and heat sink is uniform. The dynamic power density in each unit applied in the DNS is averaged over 48k CPU cycles at 3.5 GHz with the percentage of the power consumption in each unit similar to [26]. To be more realistic, the power density over each average period in each unit is generated randomly.

It should be mentioned that, for such a large domain structure of the CPU, it is impractical to include detailed variations of materials properties even though other materials exist in CPUs. The vast majority of the materials in CPUs are however silicon. To be able to manage the numerical settings more reasonably, as a general practice for thermal simulation at the architecture level [11]-[13],[15], silicon with a thermal conductivity of 100 W/(m·K), a specific heat of 751.1 J/(kg·K) and a density 2330 (kg/m^3) is assigned throughout the CPU.

A. Direct Numerical Simulations of the Selected CPU

FEniCS and HotSpot-Grid are applied to perform thermal simulations of the selected CPU with the identical dynamic power density distribution and BCs described above. Due to the higher power density in Core 1, the maximum temperature appears in Core 1 at (5.8mm, 9.8mm) indicated by the intersect of 2 lines along the A and B plotting paths shown in Fig. 1. Dynamic evolution of the maximum temperatures obtained from FEniCS and HotSpot-Grid are compared in Fig. 2. Results show that the dynamic temperature difference between HotSpot-Grid and FEniCS is very small at low temperature and increases with time as temperature becomes higher. The temperature resulting from HotSpot-Grid is always higher than that from FEniCS and the deviation reaches 4.3% at t = 2.1ms.

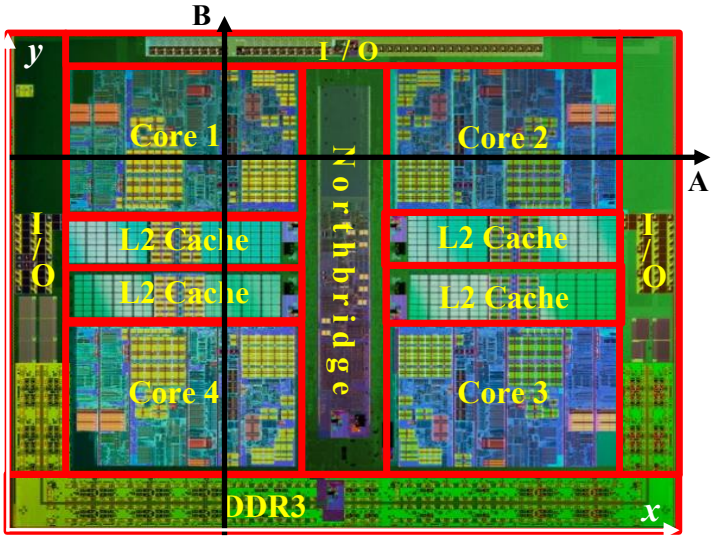

Fig. 1. Floorplan of the quad-core CPU, AMD ATHLON II X4 610e [26]. The lines with arrows indicate the paths of the temperature distributions shown in Figs. 3, 6 and 7. The intersection of these 2 lines is at (5.8mm, 9.8mm).

In addition, the temperature profiles at t = 2.1ms are given in Figs. 3(a) and 3(b) along the lines in the A and B plotting paths, respectively, indicated in Fig. 1. As shown in Fig. 3(a), the temperature distribution along the A plotting path (see Fig. 1) across I/O, Core 1, Northbridge, Core 2 and I/O from left to right reveals the highest temperature in Core 1. The temperature distribution along the B plotting path passing DDR3, Core 4, two L2 Caches, Core 1 and I/O is shown in Fig. 3(b). There are some differences between the temperature profiles predicted by FEniCS and HotSpot-Grid, as illustrated in Figs. 3(a) and 3(b). The difference is larger in the regions where the temperature is high and the maximum difference is as high as 4.3%, as observed in Fig. 2. However, the results from both DNS tools agree quite well with each other in the low temperature regions.

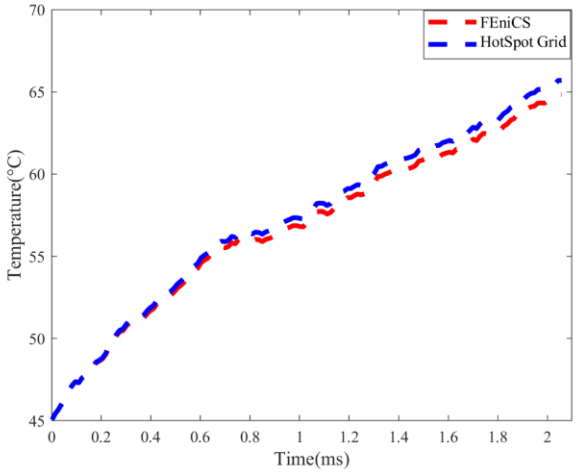

Fig. 2. Comparison of the maximum dynamic temperature in Core 1 obtained from HotSpot-Grid and FEniCS simulations

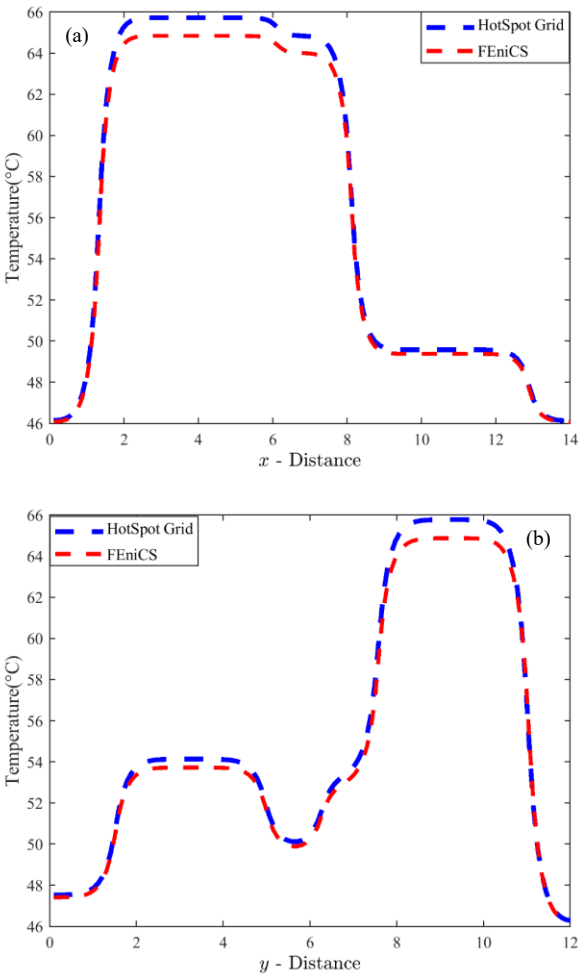

Fig. 3. Temperature distributions at $t = 2.1$ms along the paths indidated in Fig. 1 (a) in the A plotting path and (b) in the B plotting path.

B. *Eigenvalues and POD modes*

Two sets of eigenvalues and POD modes are generated from the sampled data of the dynamic temperature distributions in the 3D CPU domain collected independently from FEniCS and HotSpot-Grid simulations. The eigenvalue spectrums of the two-set thermal data are shown in Fig. 4, and they appear to be very close to each other. The eigenvalues represent the mean squared temperature variations captured by each POD mode and thus indicate the number of POD modes needed to predict the temperature solution with acceptable accuracy. As shown in Fig. 4, the eigenvalue drops more than three orders of magnitude from the first to the third mode, and a decrease by more than 4 orders of magnitude is observed from the first to the fourth mode. It is therefore expected that a POD model for the selected quad-core CPU is able to offer a reasonable prediction of the dynamic temperature distribution with just 3 modes and a very accurate prediction with just 4 or 5 modes. However, the expectation can be achieved only if the thermal solution data collected from the DNS tools are consistent with the heat conduction equation. Once the POD modes are generated, POD model parameters or coefficients given in (6) can then be determined.

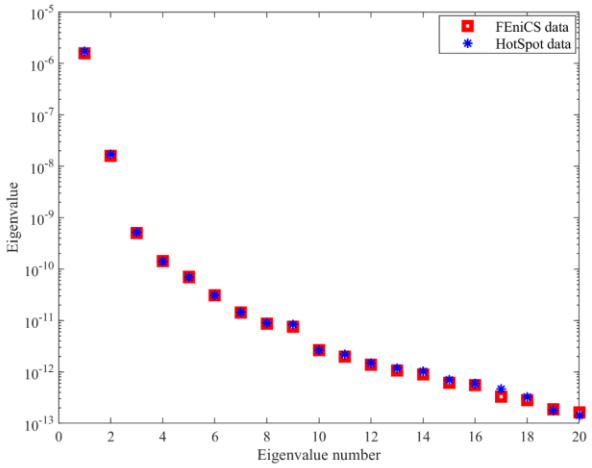

Fig. 4. The eigenvalue spectrum derived from the HotSpot-Grid data and FEniCS data.

C. *Demonstrations*

To verify the validity of the two developed POD models, they are applied to thermal simulations of the selected CPU with a dynamic power trace different from that used in the thermal data collection. More specifically, the power consumption in each unit is generated using a different random sequence from that used in the thermal data collection/training. However, the power density distribution over all the units remains similar. The dynamic temperature solution for the CPU determined by each of these 2 approaches with different numbers of modes are compared in Figs. 5-7 against the solution resulting from the simulation tool used to construct its POD model.

Fig. 5(a) shows the dynamic temperature at the location of (5.8mm, 9.8mm) indicated in Fig. 1 obtained from FEniCS and its POD model. As anticipated according to the eigenvalue spectrum shown in Fig. 4, the FEniCS-POD model is in good agreement with the FEniCS result using just 3 POD modes. The POD results with 5 or 7 modes nearly overlap with that from FEniCS. Only a 0.4% deviation from the FEniCS simulation is observed when using the 7-mode FEniCS-POD model. It is however interesting to notice a very different outcome from the HotSpot-POD model. It is shown in Fig. 5(b) that, except when temperature is low, the HotSpot-POD model with 5 and 7 modes approaches an erroneous solution that is approximately 20%-30% lower than the HotSpot-Grid results.

The temperature distributions along the paths shown in Fig. 1 at $t = 2.1$ ms are displayed in Figs. 6 and 7 for the FEniCS-POD and HotSpot-POD, respectively, compared to their DNS tools. As shown in Figs. 6(a) and 6(b), the FEniCS-POD model with 3 modes already offers a good prediction of the dynamic temperature distribution in the CPU. When using 5-7 modes for the POD model, an improved agreement with the FEniCS DNS is observed. Similarly to Fig. 5(b) for the dynamic temperature predicted by HotSpot-POD, the temperature profiles derived from the HotSpot-POD model shown in Figs. 7(a) and 7(b) approach to a solution 20%-30% lower than the HotSpot-Grid results in the higher temperature region. It is worthwhile to point out that the HotSpot-POD model predicts a solution that does not converge in the region of 6 mm $< x <$ 8mm when the number

of modes increases from 3 to 7 modes. In the same location, the solution provided by FEniCS-POD model with 3 or more modes however converges and agrees very well with the DNS solution from FEniCS.

D. Discussions

Based on the comparison between the temperature solutions predicted by FEniCS-POD and HotSpot-POD with each of their DNS tools (FEniCS and HotSpot-Grid, respectively) presented in Figs. 5-7, crucial findings are described below. It has been demonstrated that the developed FEniCS-POD model for AMD ATHLON II X4 610e, a quad-core CPU, is able to offer a very efficient prediction of dynamic thermal distribution with high accuracy in the selected CPU, compared to FEniCS DNS. The FEniCS-POD model offers a very accurate prediction of the thermal profile in a CPU with just 4 to 5 numerical DoF. This amounts to a reduction in the numerical DoF by nearly 5 orders of magnitude, compared to DNS, and a decrease in computing time by more than 1000 times. The results also strongly suggest that the thermal solution derived from FEniCS DNS is consistent with the heat conduction equation and offers good quality of POD modes to construct the FEniCS-POD model.

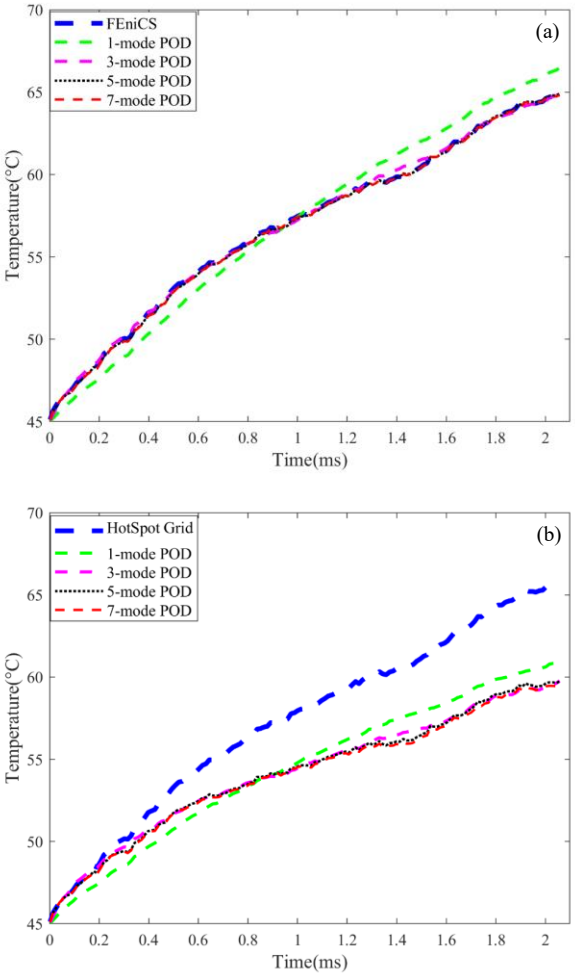

Fig. 5. Temperature evolution in time. (a) Comparison between the FEniCS-POD and FEniCS simulations and (b) comparison between the HotSpot-POD and HotSpot-Grid simualtions.

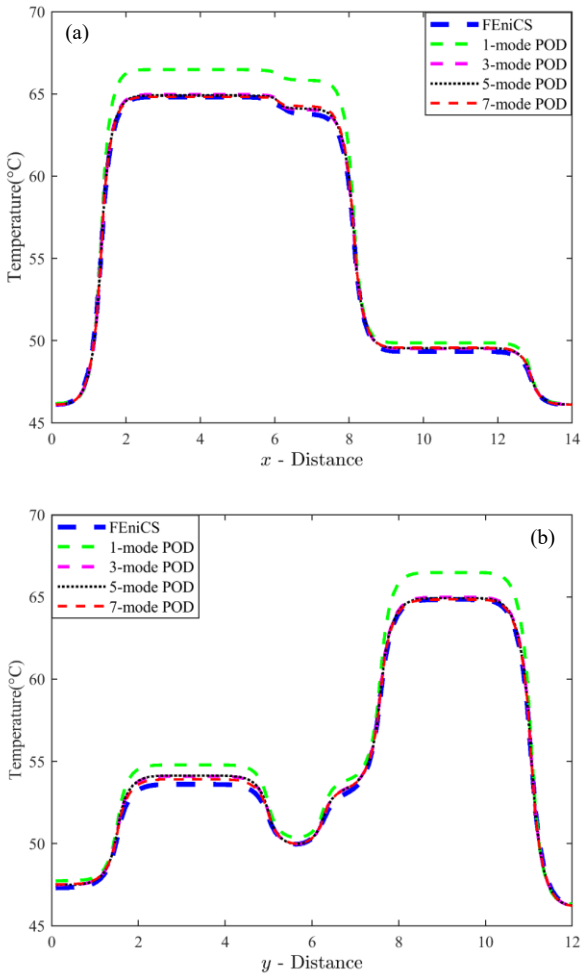

Fig. 6. Temperature distributions obtained from FEniCS and FEniCS-POD simulations at t = 2.1ms along the (a) A plotting path and (b) B plotting path.

On the contrary, the thermal solution generated from DNS using HotSpot-Grid is not quite consistent with the heat conduction equation, according to the comparisons with its POD model presented in Figs. 5(b) and 7, as well as the least square errors shown in Table I. This perhaps results from the approximation made in the Grid model to evaluate the thermal elements. It is interesting that the thermal solution from HotSPot-Grid with only approximately 4.3% deviation from the FEniCS DNS, as shown in Figs. 2 and 3, actually generates POD modes that are not able to represent the heat conduction equation transformed to the POD space. This indicates that the accuracy of POD modes strongly depends on the quality of the solution data used to generate them.

It should be mentioned that the maximization process given in (2) for the POD methodology optimizes the least square error over the complete simulation time and the whole spatial domain instead of the local error. As can be seen in Table I, the least square error for the HotSpot-POD actually increases from 1 to 3 modes and it fluctuates around 20.8% more than 3 modes. In contrast to HotSpot-POD, FEniCS-POD with more modes achieves a smaller least square error, and the error decreases to 1.63% with 7 modes.

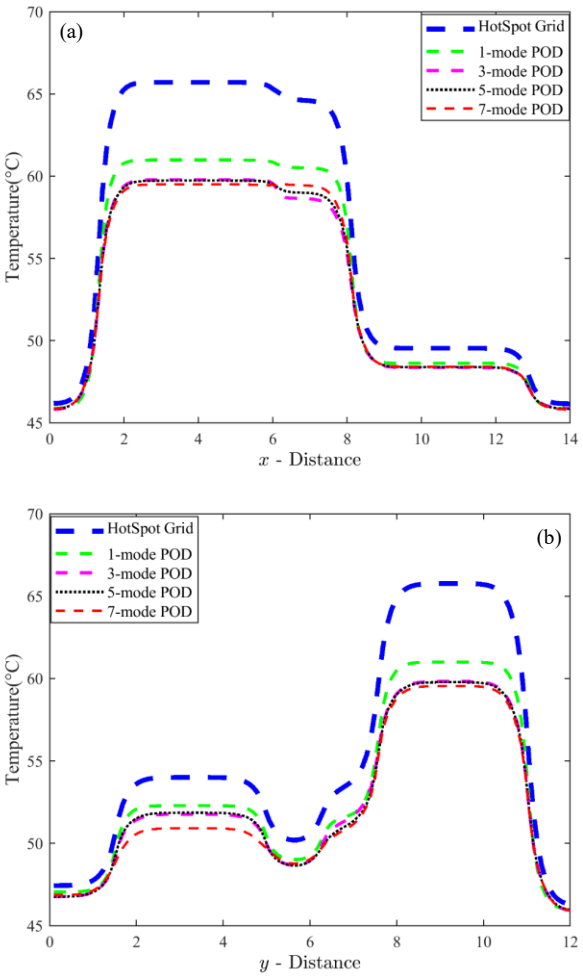

Fig. 7. Temperature distributions obtained from HotSpot-Grid and HotSpot-POD simulations at t = 2.1ms along the (a) A plotting path and (b) B plotting path.

TABLE I. THE LEAST SQUARE ERROR OVER TIME AND ENTIRE DOMAIN

Type of POD	**Number of POD modes**			
	1	***3***	***5***	***7***
HotSpot-POD (%)	17.31	20.80	20.76	20.81
FEniCS-POD (%)	7.17	1.89	1.73	1.63

IV. CONCLUSION

Thermal simulations for the AMD ATHLON II X4 610e CPU, a quad-core CPU, have been conducted using the DNS tools (FEniCS and HotSpot-Grid) and the data-driven POD models [16]-[19] developed from these DNS tools. Although simulation results from both DNS tools are consistent at low temperature, at higher temperature HotSpot-Grid always overestimates the temperature, compared with FEniCS.

In this study, a data-driven model reduction approach based on the POD has been applied to develop an efficient and accurate thermal simulation model for CPUs. To develop such a reduce-order model, the physical domain of the selected CPU is projected onto a functional space described by the POD modes generated from the thermal solution data that are collected from DNS of the CPU. It has been demonstrated that the model described by the POD modes trained by FEniCS DNS is able to predict the dynamic thermal distribution of the selected CPU with high accuracy and a reduction in the numerical DoF by nearly five orders of magnitude can be achieved, compared to the DNS. This reduces the computing time by a factor of several thousands. Contrarily to the accurate FEniCS-POD model, the data-driven POD model represented by the modes trained by HotSpot-Grid cannot predict the dynamic temperature profile accurately due to inadequate quality of the thermal solution data used in the training. This is probably caused by approximations used for evaluating thermal elements in HotSpot-Grid. As shown in Table I, the least square error of HotSpot-POD with respect to HotSpot-Grid simulations is as high as 20% with 3-7 modes. The large errors for HotSpot-POD in time and space are also observed in Figs. 5(b) and 7. The least square error of FEniCS-POD with respect to FEniCS simulations on the other hand decreases to 1.63% with 7 modes.

This work presents the first study of the POD learning approach for the thermal simulation of a CPU. This approach is able to offer a thermal distribution as accurate as the DNS and as efficient as the RC thermal circuit model. In addition, the resolution of the POD approach is determined by its modes whose resolution is as fine as the DNS used to train them. However, to train such a large domain structure with an enough resolution, the demand for computational resources may become prohibitive, especially when applying the approach to CPUs with a large number of cores or GPUs with hundreds or thousands of cores. To make the POD approach more applicable for such applications and to maintain enough resolution to capture critical hot spots, a multi-block approach is needed [16]. With the multi-block concept, the training of the POD modes for each smaller block by the thermal data collected from DNSs becomes more feasible. Application of multi-block POD methodology to CPUs and GPUs will be investigated in the near future.

Acknowledgment: This work is supported by National Science Foundation under Grant No. ECCS-2003307

REFERENCES

[1] X. Zhou, Y. Xu, Y. Du, Y. Zhang and J. Yang, "Thermal management for 3D Processors via task scheduling," 37th Int. Conf. Parallel Processing, 2008, pp. 115-122.

[2] H.F. Sheikh, I. Ahmad, Z. Wang, S. Ranka, "An overview and classification of thermal-aware scheduling techniques for multi-core processing systems," Sustainable Computing: Informatics & Systems, Vol. 2, 3, pp. 151-169, Sep. 2012.

[3] A. Heinig, R. Fischbach, and M. Dietrich, "Thermal analysis and optimization of 2.5D and 3D integrated systems with wide I/O memory," in Proc. Conf. Therm. Thermomech. Phenom. Electron. Syst. (ITherm), 2014, pp. 86–91.

[4] J. Zhou, J. Yan, K. Cao, Y. Tan, T. Wei, M. Chen, G. Zhang, X. Chen, S. Hu, "Thermal-aware correlated two-level scheduling of real-time tasks with reduced processor energy on heterogeneous MPSoCs", J. Systems Architecture, Vol. 82, pp. 1-11, Jan. 2018.

[5] Y.W. Chang, Y. Cheng, L. Helfen, F. Xu, T. Tian, M. Scheel, et al., "Electromigration mechanism of failure in flip-chip solder joints based on discrete void formation." Sci. Rep., vol. 7, pp. 17950, Dec. 2017.

[6] X. Huang, A. Kteyan, X. Tan and V. Sukharev, "Physics-based electromigration models and full-chip assessment for power grid networks", IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 35, no. 11, pp. 1848-1861, Feb. 2016.

[7] Y. Liu, M. Li, M. Jiang, D. W. Kim, S. Gu and K. N. Tu, "Joule heating enhanced electromigration failure in redistribution layer in 2.5D IC", Electronic Components and Technology Conference (ECTC), 2016, pp. 1359-1363.

[8] J. W. Sheaffer, K. Skadron and D. P. Luebke, "Studying thermal management for graphics-processor architectures", Proc. ISPASS, 2005, pp. 54-65

[9] R. Nath, R. Ayoub and T. S. Rosing, "Temperature aware thread block scheduling in GPGPUs", Proc. 50th ACM/EDAC/IEEE Design Autom. Conf. (DAC), 2013, pp. 1-6

[10] The FEniCS Project, M. S. Alnaes, J. Blechta, J. Hake, A. Johansson, B. Kehlet, A. Logg, C. Richardson, J. Ring, M. E. Rognes and G. N. Wells, Archive of Numerical Software, Vol. 3, 2015

[11] HotSpot 6.0 Temperature Modeling Tool. [online] Available: http://lava.cs.virginia.edu/HotSpot/.

[12] K. Sankaranarayanan, S. Velusamy, M. R. Stan, and K. Skadron, "A case for thermal-aware floorplanning at the microarchitectural level," J. Instruction-Level Parallelism, vol. 7, pp. 1–16, Oct. 2005.

[13] K. Skadrony, M. Stanz, M. Barcellaz, A. Dwarkaz, W. Huangz, Y. Liy, et al., "HotSpot: Techniques for modeling thermal effects at the processor-architecture level", Proc. Int. Workshop THERMal Investigations ICs Syst., 2002, pp. 1-4.

[14] D. Fetis and P. Michaud, "An Evaluation of HotSpot-3.0 Block-Based Temperature Model", Proc. Workshop Duplicating Deconstructing and Debunking in conjunction with Int'l Symp. Computer Architecture, 2006.

[15] W. Huang, K. Sankaranarayanan, R. J. Ribando, M. R. Stan and K. Skadron, "An improved block-based thermal model in hotspot 4.0 with granularity considerations", Proc. WDDD, 2007.

[16] W. Jia, B.T. Helenbrook, Ming-C. Cheng, "Fast Thermal Simulation of FinFET Circuits Based on a Multi-Block Reduced-Order Model", IEEE Trans. CAD ICs & Systems, vol. 35, no. 7, pp. 1114-1124, Jul. 2016.

[17] W. Jia, B.T. Helenbrook, Ming-C. Cheng, "Thermal Modeling of Multi-Fin Field Effect Transistor Structure Using Proper Orthogonal Decomposition", IEEE Trans. Electron Devices, Vol. 61, No. 8, pp. 2752-2759, Aug. 2014.

[18] M.C. Cheng and B.T. Helenbrook, "Method for Thermal Simulation", US Patent 8,539,408, Sep. 2013.

[19] Ravon Venters, B.T. Helenbrook Kun Zhang, Ming-C. Cheng, "Proper Orthogonal Decomposition Based Thermal Modeling of Semiconductor Structures," IEEE Trans. Electron Devices, Vol.59, No. 11, pp. 2924-2931, Nov. 2012.

[20] J. L. Lumley, "The structure of inhomogeneous turbulent flows," in Atmospheric Turbulence and Radio Wave Propagation. Moscow, Russia: Nauka, 1967, pp. 166–178.

[21] G. Berkooz, P. Holmes, and J. L. Lumley, "The proper orthogonal decomposition in the analysis of turbulent flows," Annu. Rev. Fluid Mech., vol. 25, pp. 539–575, Jan. 1993.

[22] CPU-World. [online] Available: https://www.cpu-world.com/CPUs/K10/AMD-Athlon%20II%20X4%20610e%20-%20AD610EHDK42GM%20(AD610EHDGMBOX).html

[23] A. Marquez, J. J. E. Oviedo and D. Odloak. "Model reduction using proper orthogonal decomposition and predictive control of distributed reactor system." J. Control Sci. Eng., Vol. 3, pp. 1-19, Jan. 2013.

[24] Z. Wang, B. McBee, T. Iliescu. "Approximate partitioned method of snapshots for POD," J. Comput. Appl. Math., Vol. 307, pp. 374-384, 2016.

[25] T. G. Ritto, F. S. Buezas and R. Sampaio, "Proper orthogonal decomposition for model reduction of a vibroimpact system", J. Brazil. Soc. Mech. Sci. Eng., vol. 34, no. 3, pp. 330-340, 2012.

[26] K. Dev, A. N. Nowroz, and S. Reda, "Power mapping and modeling of multi-core processors," in Proc. IEEE Int. Symp. Low-Power Electron. Design, Sep. 2013, pp. 39–44.