ITherm VIRTUAL 2021

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

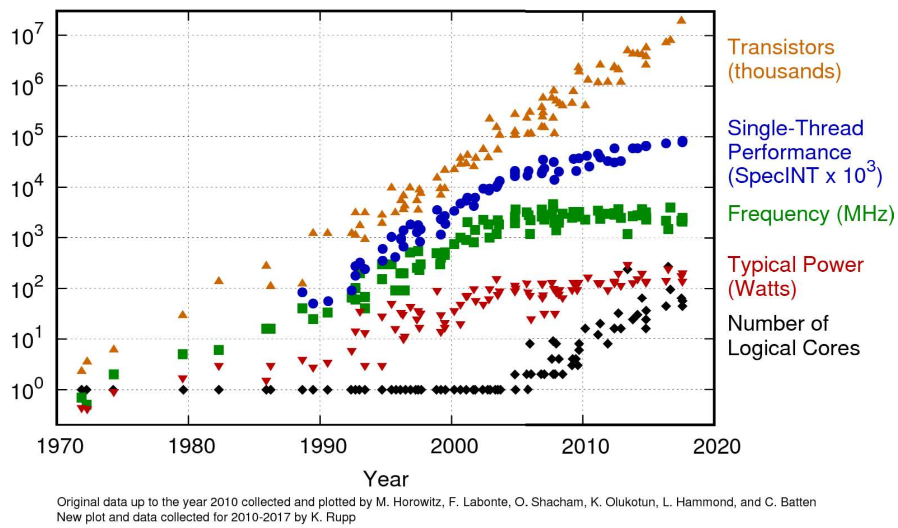
Lin Jiang, Yu Liu, Ming-C Cheng

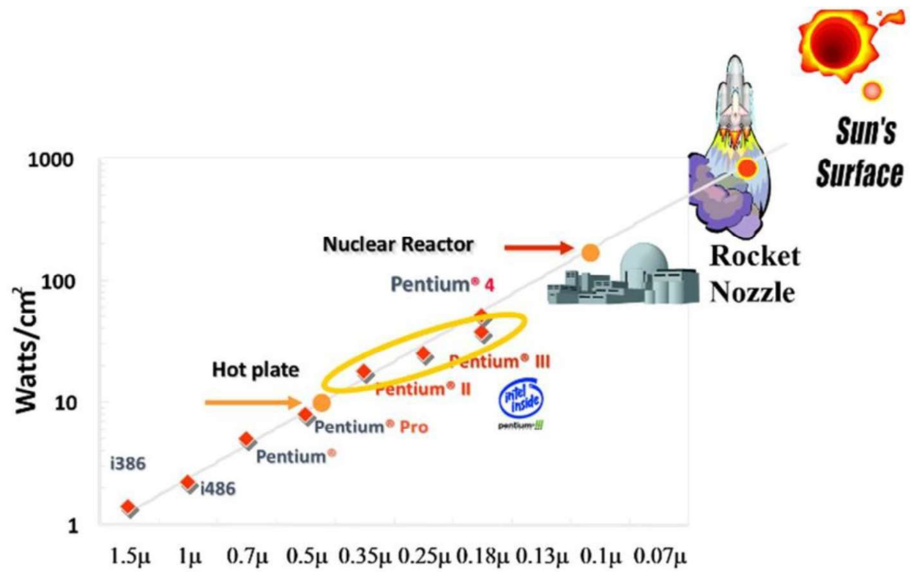
Department of Electrical and Computer Engineering
Clarkson University
April 2021

Supported under Grant No. ECCS-2003307

Clarkson

Background





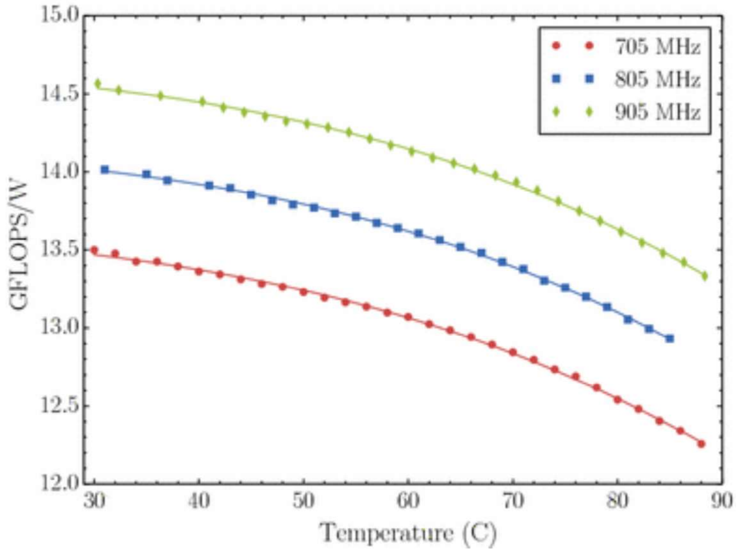
Power density for successive nodes[2]

[1] K. Rupp, "42 years of microprocessor trend data," in Proc. High-Perform. Comput. Symp. San Francisco, CA, USA: GitHub, Feb. 2018.
[2] D. Etiemble, "45-year CPU evolution: one law and two equations", *Second Workshop on Pioneering Processor Paradigms*, Feb. 2018, [online] Available: https://hal.archives-ouvertes.fr/hal-01719766.

Background

Impact of high temperature

- Degrade GPU and CPU performance
- Degrade GPU and CPU reliability
- Increasing the cooling cost



GPU performance VS temperature[3]

Reduction of temperature:

- Thermal management (thermal-aware scheduling)
- Thermal aware exploration for GPU and CPU floorplanning

Effective thermal simulation method is needed.

[3] Price, D.C., Clark, M.A., Barsdell, B.R. et al. Optimizing performance-per-watt on GPUs in high performance computing. Comput Sci Res Dev 31, 185–193 (2016).

Thermal Simulation of GPUs and CPUs

Difficulties for thermal simulation of GPUs and multi-core CPUs:

- Fine enough mesh is required to capture hot spots in a chip.
- Due to large degrees of freedom (DoF), the thermal simulation is very time-consuming.

Solution:

- Reduce the numerical DoF via appropriate projection from the physical domain onto a functional space, for example using proper orthogonal decomposition(POD).

An effective projection:

Extraction of POD modes: $\int_{\vec{r}'} \langle T(\vec{r},t)T(\vec{r}',t)\rangle \vec{\varphi}(\vec{r}')d\vec{r}' = \lambda\vec{\varphi}(\vec{r}).$

Temperature solution via coefficients $a_i(t)$ and POD modes $\varphi(\vec{r})$: $T(\vec{r},t) = \sum_{i=1}^{M} a_i(t)\varphi(\vec{r}).$

Thermal Simulation of GPUs and CPUs

ITherm VIRTUAL 2021

Proper Orthogonal Decomposition

- The POD modes are optimized by maximizing the mean square inner product of temperature and POD modes

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

- Project the heat conduction equation to POD space

$$\int_{\Omega}\left(\vec{\varphi}_i \frac{\partial \rho C_s \vec{T}}{\partial t} + \nabla\vec{\varphi}_i \cdot k\nabla\vec{T}\right) d\Omega = \int_{\Omega} \vec{\varphi}_i \cdot P_d(\vec{r},t) d\Omega - \int_{S} \vec{\varphi}_i(-k\nabla\vec{T} \cdot \vec{n})\, dS.$$

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

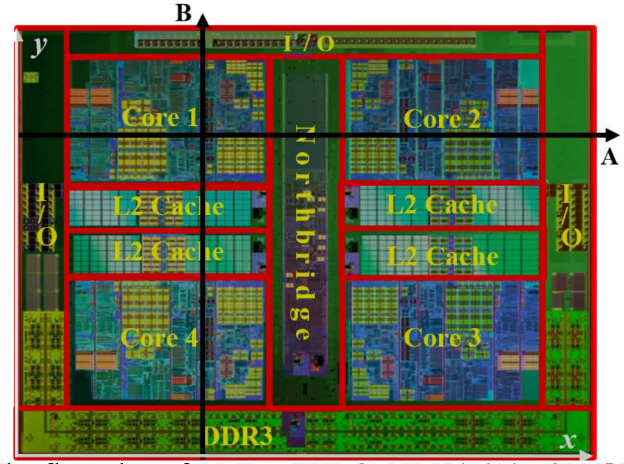
- The data accounting for variations of boundary conditions (BCs) and power trace are needed to train the POD modes $\varphi_i(\vec{r})$.

Clarkson

Thermal Simulation of GPUs and CPUs

AMD ATHLON II X4 610e CPU

- *hmmer* and *soplex* benchmarks running in Core 1 and Core 3 [4].
- The dynamic power is averaged over 48k CPU cycles at 3.5 GHz.



The floorplan of AMD ATHLON II X4 610e CPU[5].

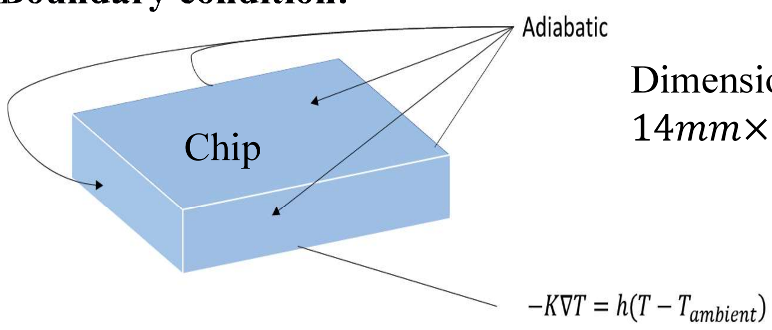
	Core 1	L2-1	Core 2	L2-2	Core 3	L2-3	Core 4	L2-4	I/O	N. B.	DDR3
Percentage of power (%)	24.1	2.2	5.2	1.4	17.5	2.3	10.6	3.7	3.2	26.2	3.6

[4] 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
[5] CPU-World. [online] Available: https://www.cpu-world.com/CPUs/K10/AMD-Athlon%20II%20X4%20610e%20-%20AD610EHDK42GM%20(AD610EHDGMBOX).html

Thermal Simulation of GPUs and CPUs

AMD ATHLON II X4 610 e CPU

Boundary condition:



Dimension of chip:
$14mm \times 12mm \times 650um$

Material properties:

Property	Density	Thermal conductivity	Specific heat
Chip	$2330\text{kg}/m^3$	$100\text{W}/(\text{m}\cdot K)$	$751.1\text{J}/(\text{kg}\cdot K)$

[6] https://fenicsproject.org/
[7] https://github.com/uvahotspot/hotspot

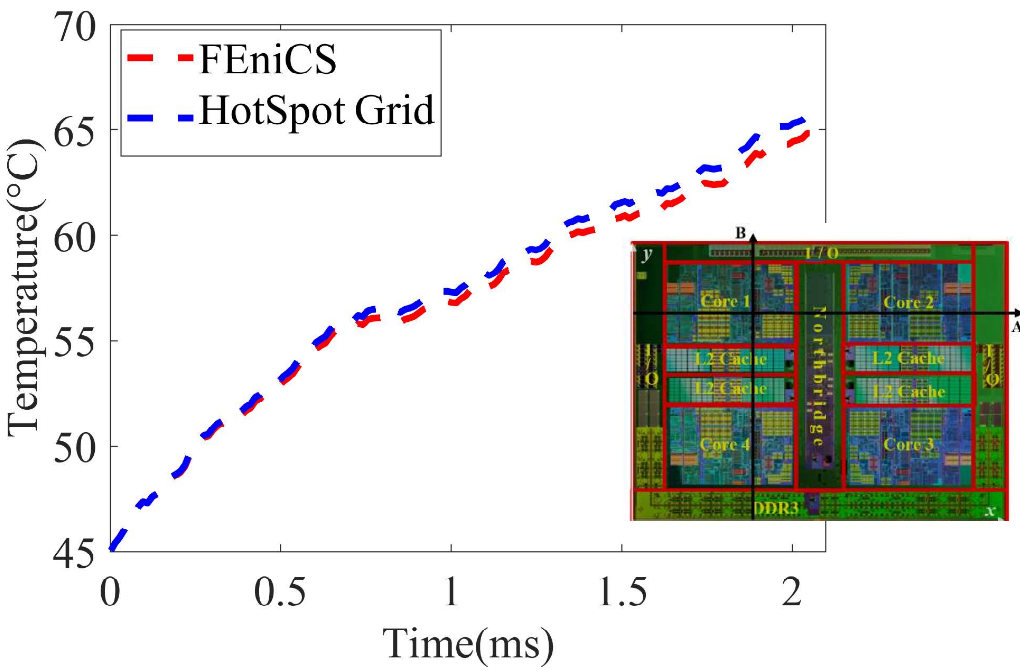
Thermal simulators:

- FEniCS[6]: A popular open-source computing platform for solving partial differential equations using finite element method (FEM).
- HotSpot-Grid model[7]: A thermal simulator based on lumped thermal circuit element model. In this work, very small elements are used and HotSpot-Grid is similar to finite difference method (FDM).

Data Collection

FEniCS VS HotSpot-Grid

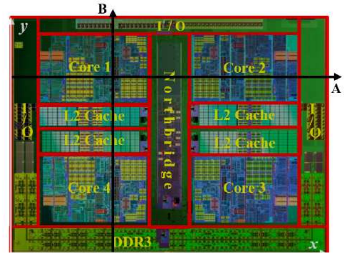
The two sets of thermal data are collected via FEniCS and HotSpot-Grid for the multi-core CPU.



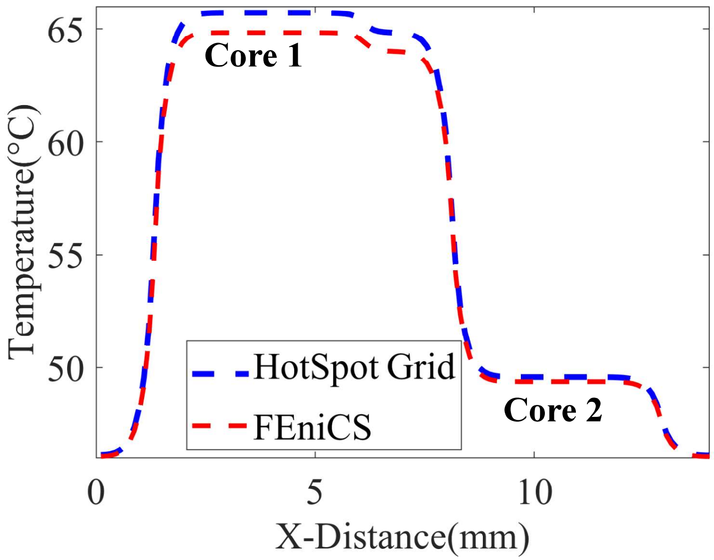
The temperature evolution at (5.8mm, 9.8mm) :

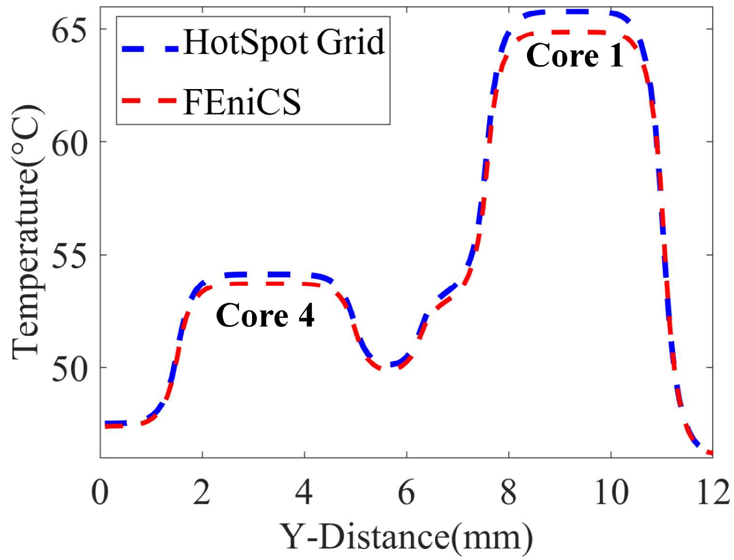
- HotSpot-Grid results agree with FEniCS result at low temperature.
- With the temperature increasing, difference becomes bigger (4.3% at 2.1ms).

Data Collection



FEniCS VS HotSpot-Grid (The temperature profile at 2.1ms**)**





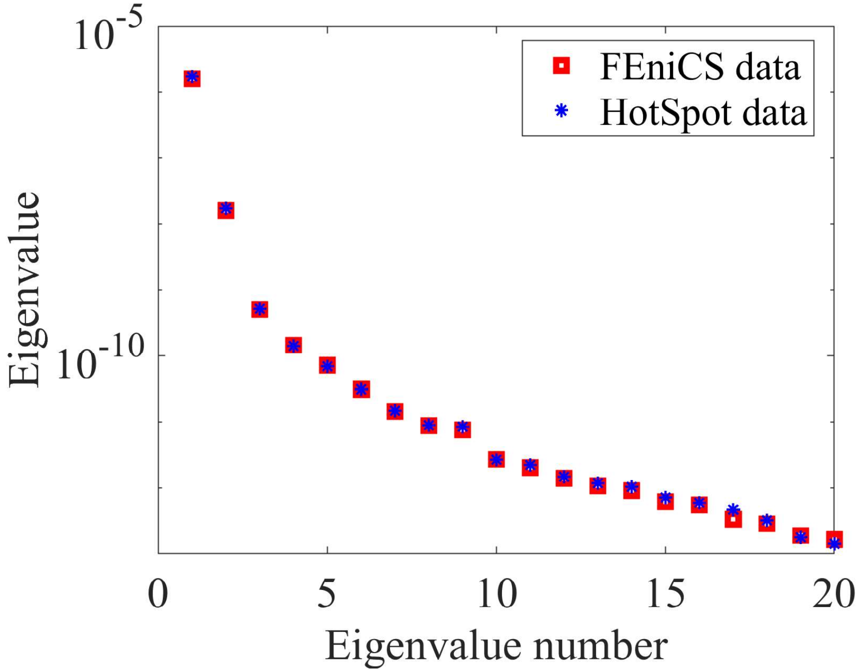
- The deviation between two simulators is greater at high temperature region for example Core 1.
- The maximum temperature appears in Core 1.

Clarkson™

Data-Driven POD Approach

The data collected by FEniCS and HotSpot-Grid is used to train two sets of POD modes separately.

Eigenvalue Spectrum



- 3rd mode: reduced by more than 3 orders from the first mode eigenvalue.
- Expect that the POD model with 3 modes will offer a good prediction for POD models trained by data from both FEniCS and HotSpot-Grid

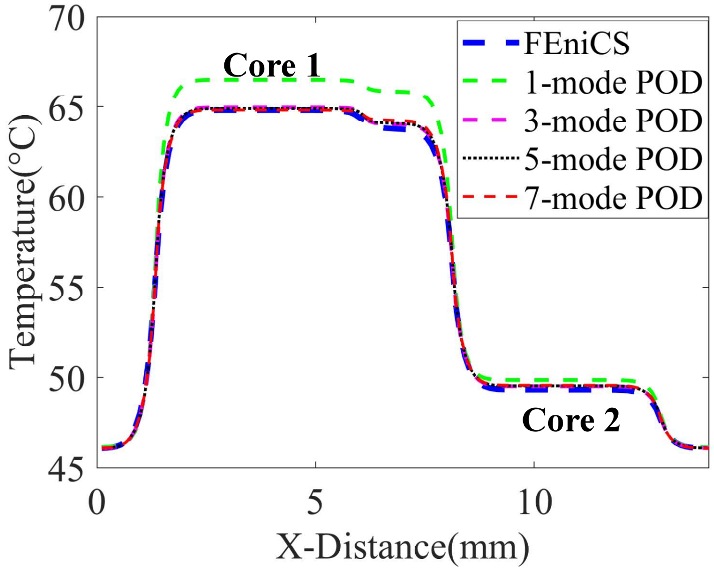
(if data quality is sifficient)

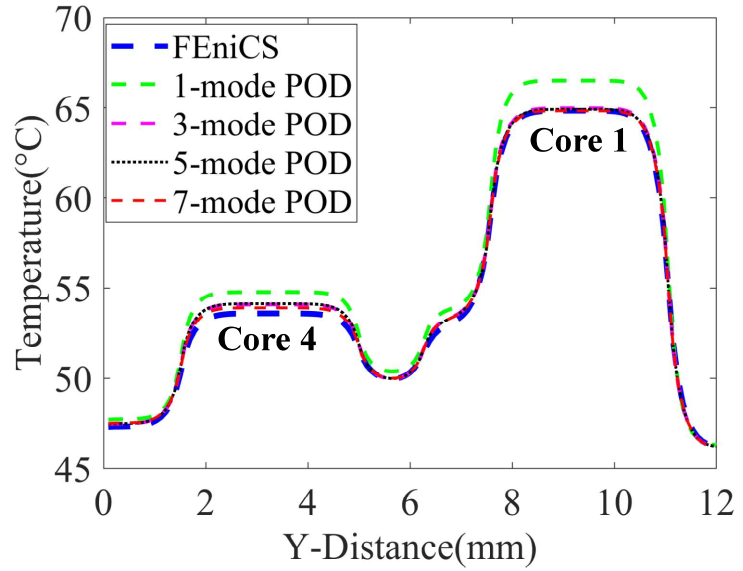
Clarkson™

Data-Driven POD Approach

FEniCS-POD

The temperature distribution at 2.1ms:

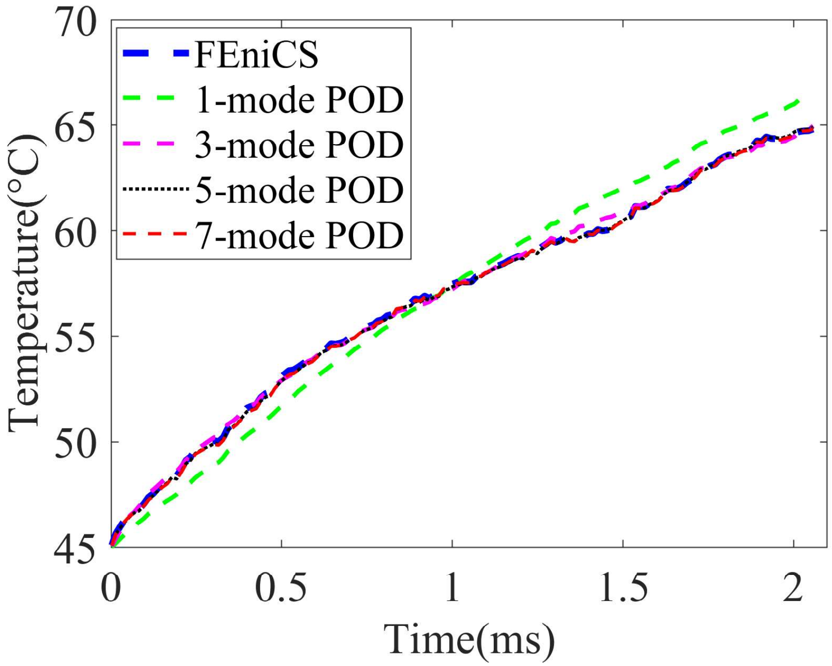




3 FEniCS-POD modes already provide a good prediction. 5 and 7 modes offer an improved agreement with FEniCS result.

Data-Driven POD Approach

FEniCS-POD



Dynamic temperature given by FEniCS-POD model:

- 3 modes offer a good prediction compared with DNS result. 5 and 7 modes results almost overlap with DNS results.
- Only 0.4% deviation with 7 modes at 2.1ms.

Reduction of DoF:

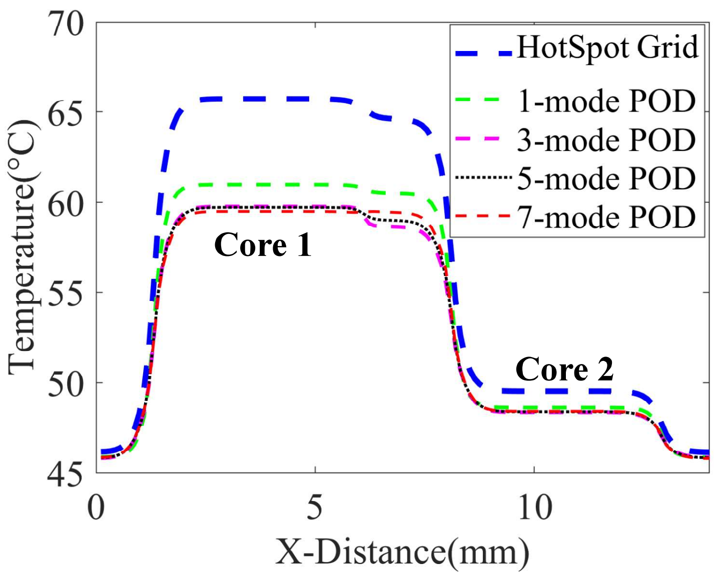
- FEniCS-POD model with 3-7 modes offers an accurate prediction for the CPU.
- Reduction in DoF by nearly 5 orders of magnitude.
- Decrease in computing time by more than 1000 times. (for predicting temperature in the entire chip)

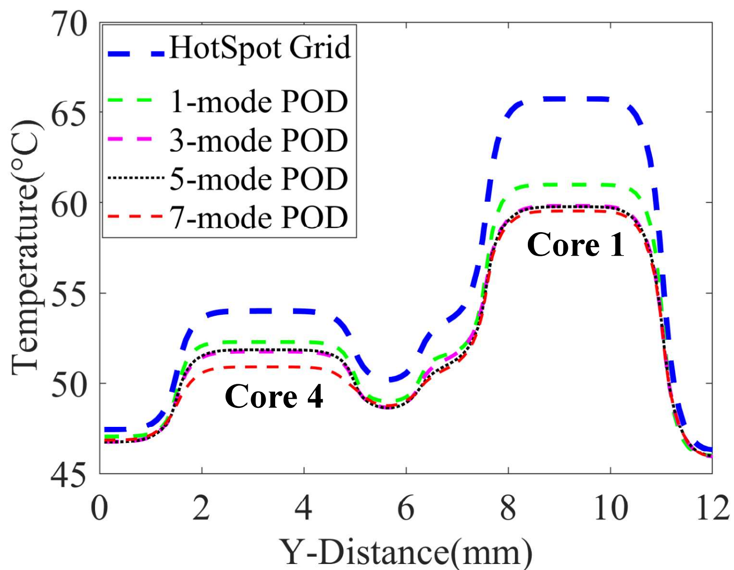
Data-Driven POD Approach

HotSpot-POD

The temperature distribution at 2.1ms:

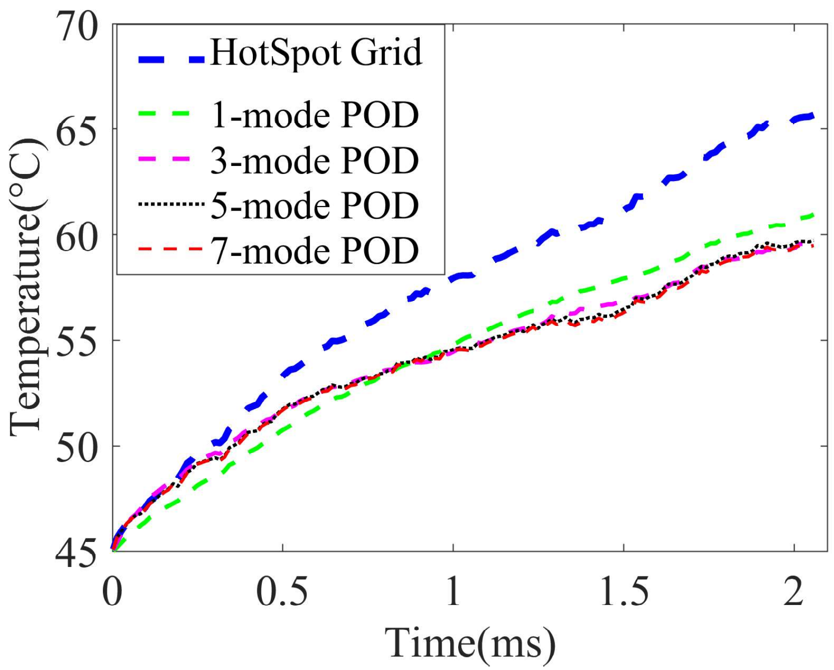
- ➢ The solution from HotSpot-POD model does not converge with 7 modes.
- ➢ With 7 modes, 20%-30% lower than the DNS results is observed in the high temperature region.





Data-Driven POD Approach

HotSpot-POD



Dynamic temperature:

- Solution offered by HotSpot-POD model are consistent with DNS result at low temperature period ($t < 0.25ms$).
- The deviation becomes considerably bigger with temperature increasing.

Discussion:

- The failure of the prediction via HotSpot-POD indicates the poor quality of data.
- Perhaps, this results from approximation made in HotSpot-Grid RC thermal elements.

Data-Driven POD Approach

POD model optimizes the least square error over the entire simulation time and domain rather than local error.

Least square error (LS error)

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

FEniCS-POD:

- Decrease to 1.6% with 7 modes.

HotSpot-POD:

- The LS error increases from 1 to 3 modes.
- It fluctuates around 20.8% more than 3 modes.

The results strongly suggest:

- The thermal solution derived from FEniCS DNS is consistent with the heat conduction equation (good quality);
- High quality data offers robust POD modes to construct the data-driven POD approach.