Two Fast Approaches for 3D Thermal Simulation of Integrated Circuits

Wenjian Yu
Dept. Computer Science and Technology
Tsinghua Univ., Beijing, China
2014.10.29

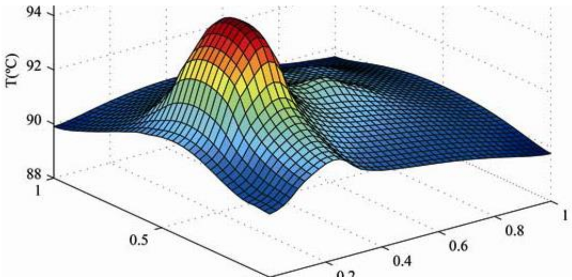
Thanks to Tao Zhang, Yuan Liang of Tsinghua, and Dr. Haifeng Qian of IBM Watson

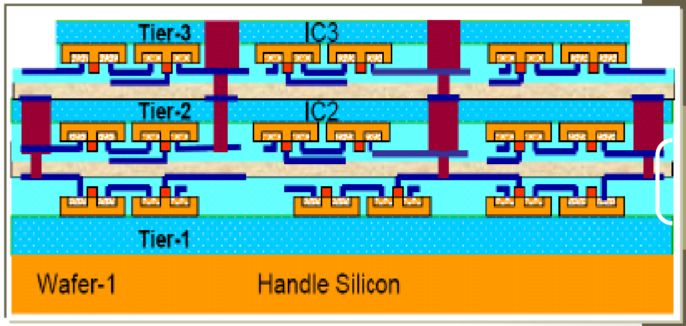
Outline

- Background
- 3D FVM for IC Thermal Simulation
- Domain Decomposition Techniques
- A Hybrid Random Walk Method
- Conclusions

Background

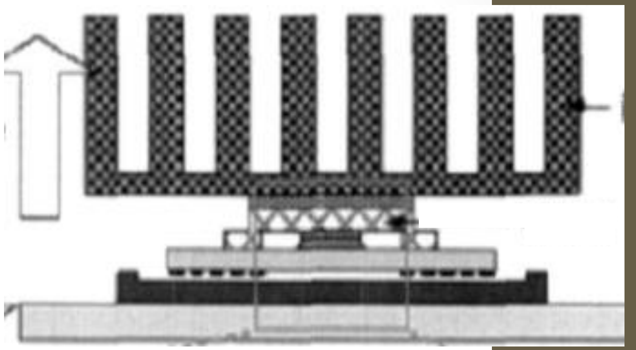
- Motivation of IC thermal analysis
  - The devices in an IC continuously increase
  - Heat dissipation and thermal management become problems threatening circuit reliability and performance
  - Chip-level thermal analysis (simulation): sign-off stage, also for design-time circuit optimizations
  - 3D IC is a trend: reduce delay, enable heterogeneous integration
  - Severe heat dissipation problem
  - Importance of accurate thermal simulation during design



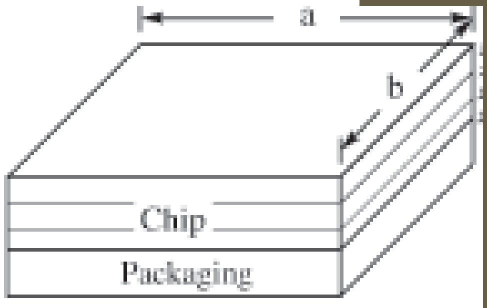


Background

- Chip-level thermal simulation
  - Should consider heat sink components
  - Simulating the whole IC thermal model (w/ irregular geometry) brings *computational challenges*



- Existing works
  - 1.Consider simplified rectangular domain
  - [Li, ICCAD’04]: Geometric multigrid iterative
  - [Zhan, TCAD’07]: Green’s function based
  - 2.Consider realistic pyramid-geometry domain
  - [Heriz, Thermal’07]: Convolution based, only for <u>low-resolution</u>
  - [Qian, ICCAD’10-TODAES’12]: Fast Poisson solver (FPS)+PCG, with <u>increased unknowns</u> and geometry-dependent <u>convergence</u>



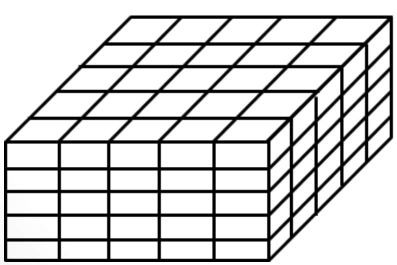
cause > 10°C error !

3D FVM for Thermal Simulation

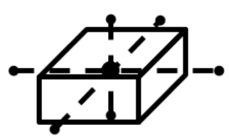
- Problem formulation
  - 3D steady-state heat equation

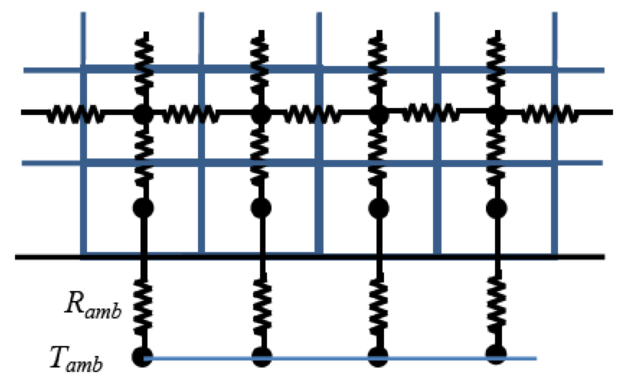
$$k \cdot \left(\frac{\partial^2 T(x,y,z)}{\partial x^2} + \frac{\partial^2 T(x,y,z)}{\partial y^2} + \frac{\partial^2 T(x,y,z)}{\partial z^2} \right) = -p(x,y,z)$$

  - Boundary conditions: Neumann (adiabatic) condition, convective condition $k \dfrac{\partial T}{\partial \vec{n}} + h(T - T_{amb}) = 0$
  - Finite difference (volume) discretization



Thermal resistor!

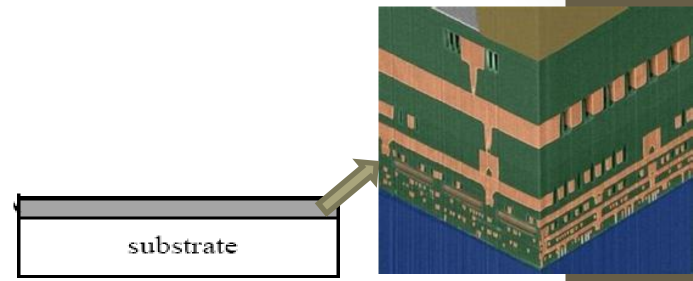




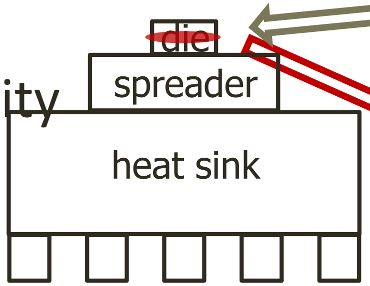
3D FVM for Thermal Simulation

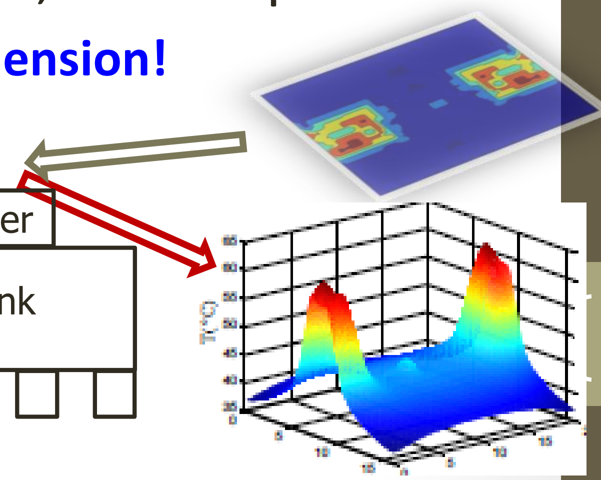
- Practical considerations
  - Inhomogeneous material in IC region
  - Thermal resistors across various material interfaces
  - To simplify, approximate the interconnect layer with a homogeneous layer

$$k_{eff} = r_{metal} \cdot k_{cooper} + (1 - r_{metal}) \cdot k_{oxide}$$



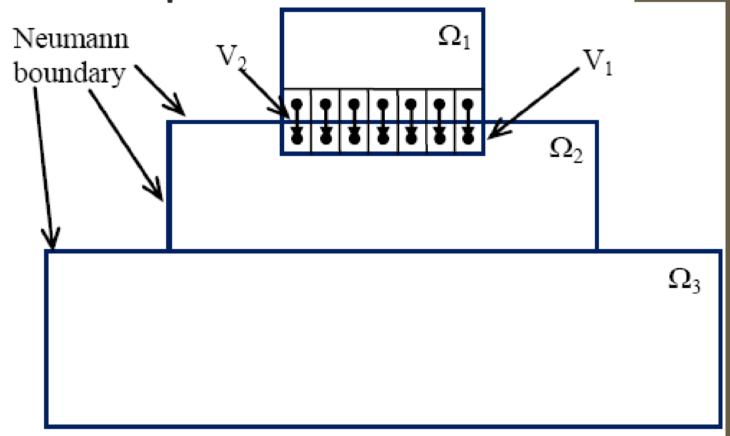
  - Power source resembles current source; solve equivalent circuit equation: $AT = f$ **Huge dimension!**
  - Two observations
    - Subdomain geometry regularity
    - Concern only the die region





Domain Decomposition Technique

- A general method for simulating complex domain
  - Nonoverlapping DDM from FVM-circuit viewpoint
    - The solution of Ω_2 provides Dirichlet boundary condition for Ω_1
    - The heat flow at the bottom of Ω_1 provides Neumann condition for Ω_2
  - Different iteration schemes
    - Top-to-bottom order
    - Bottom-to-top order
    - Middle-to-end order
    - End-to-middle order
    - Nested two-subdomain order (more reliable but costly)

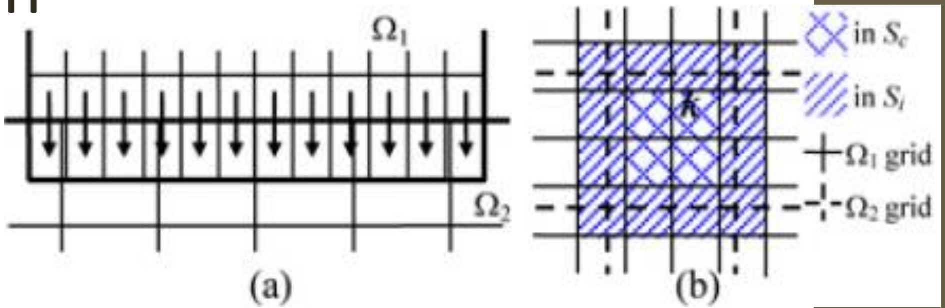


- Check convergence w/ the interfacial quantities
- Relaxed iterative scheme:

$$T_{V1}^{(i+1)} = T_{V1}^{(i)} + \omega(\tilde{T}_{V1}^{(i+1)} - T_{V1}^{(i)})$$

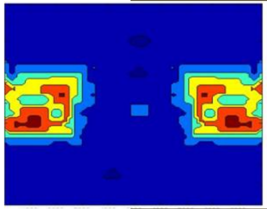
Domain Decomposition Technique

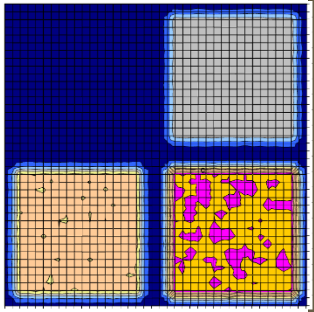
- Nonconformal discretization
  - Solve subdomains separately
  - Much coarser discretization used for heat spreader/sink



  - Linear interpolation converting quantities across interface; less affects the temperature in IC subdomain
- Exploit the regularity of subdomain
  - With conductivity homogenization, most subdomains are rectangular ones with simple configurations and conditions
  - FPS [Qian, ICCAD’10] with O(nlogn) time-complexity and O(n) space-complexity used for solving subdomains

Domain Decomposition Technique



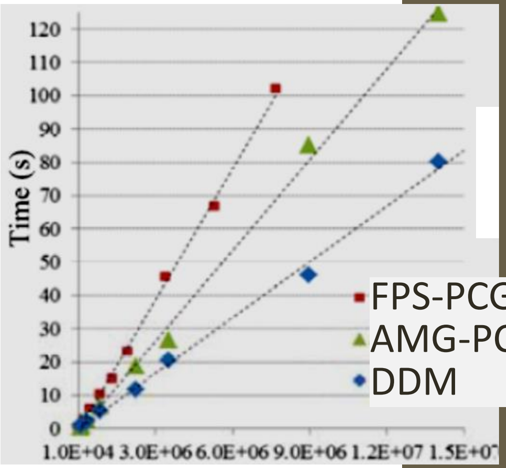


- Test cases
  - Case 1: A 2D chip imitating the Power6 (175W in 1.6x2 die)
  - Case 2: A 4-core 2D chip (176W in 1x1 die)
  - Case 3: A 3D chip with Case 1+ 2 SRAM dies
- Experiments
  - Apply conformal discretization, and compare with Matlab "\", ICT-PCG, AMG-PCG (the fastest iterative PG solver), and FPS-PCG
  - With the result of Matlab "\", check accuracy
  - Apply nonconformal discretization, to show efficiency improvement
  - Use the simulator in heat sink component design

Domain Decomposition Technique

Matrix	n	"\"	ICT-PCG		AMG-PCG			DDM	
		Time(s)	Iter.	Time(s)	Iter.	Time(s)	Mem.	Time(s)	Mem.
M1-1	1.45e5	**35.4**	75	**16.7**	16	**1.15**	14MB	**1.20**	< 8MB
M1-2	1.51e5	**38.9**	75	**16.8**	13	**1.01**	33MB	**1.28**	< 8MB
M1-3	3.42e5	**331.5**	110	**43.1**	14	**2.63**	72MB	**2.48**	< 8MB
M1-4	5.47e6	--	--	--	15	**51.1**	1.1GB	**32.3**	80MB
M1-5	1.51e7	--	--	--	14	**142.6**	3.0GB	**90.7**	218MB
M1-6	3.42e7	--	--	--	--	--	--	**207.2**	494MB
M2-1	8.97e4	**18.4**	66	**6.88**	12	**0.52**	14MB	**0.82**	< 10MB
M2-2	1.40e5	**43.4**	78	**16.7**	12	**0.89**	31MB	**1.13**	< 10MB
M3-3	3.59e5	**359.3**	103	**204.9**	12	**2.68**	78MB	**2.21**	< 10MB
M2-4	3.51e6	--	--	--	12	**26.63**	718MB	**20.5**	55MB
M2-5	1.40e7	--	--	--	13	**124.6**	2.8GB	**80.1**	217MB
M2-6	3.31e7	--	--	--	--	--	--	**185.2**	493MB
M3-1	1.50e5	**39.6**	77	**16.8**	13	**1.08**	18MB	**1.28**	< 8MB
M3-2	1.55e5	**40.8**	77	**16.9**	13	**1.11**	37MB	**1.32**	< 8MB
M3-3	3.46e5	**493.4**	112	**33.3**	14	**2.81**	78MB	**2.61**	< 8MB
M3-4	5.54e6	--	--	--	14	**49.1**	1.1GB	**33.5**	80MB
M3-5	1.55e7	--	--	--	15	**160.8**	3.1GB	**92.4**	218MB
M3-6	3.46e7	--	--	--	--	--	--	**209.1**	494MB

- >10X memory save!
- Faster than iterative solvers for large case
- Converge in 8, 9 steps



- The temperature error in IC region is less than 0.01 °C

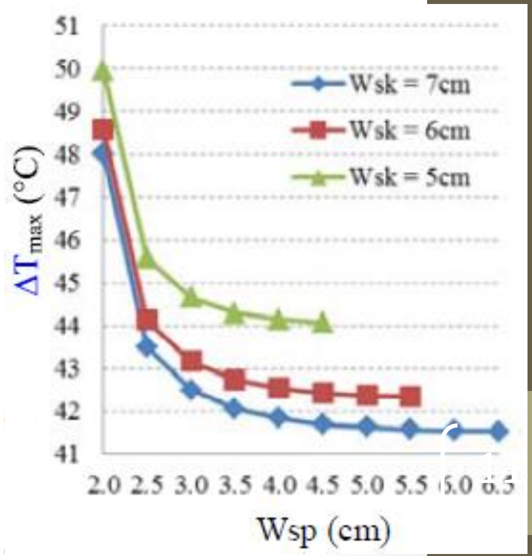
Domain Decomposition Technique

- Nonconformal discretization

Matrix	Conformal grid			Non-conformal grid				
	n	Time(s)	T_{max}(°C)	n	Time(s)	T_{max}(°C)	Err_{max}(°C)	Speedup
M1-4	5.47e6	32.3	61.89	5.73e5	3.86	61.90	0.11	**8.4**
M1-5	1.51e7	90.7	61.97	1.65e6	8.98	61.95	0.03	**10.1**
M1-6	3.42e7	203.2	61.88	1.99e6	11.2	61.85	0.04	**18**
M1-10	5.41e7	310.1	61.85	3.22e6	18.5	61.84	0.02	**17**

- Apply to larger case
- < 0.05 °C Error on hot spot
- >10X memory save

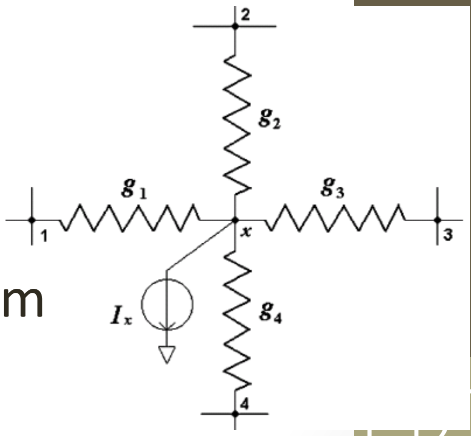
- High-resolution simulation and design
  - 1.05×10^7 unknown (50μm discretization-step) in IC region, needs 72s simulation time
  - Study the temperature variations with the widths of heat spreader/sink changed
  - 24-configuration simulation costs 7 mins.



A Hybrid Random Walk Method

- Thermal simulation for hot-spots at device layer
  - Entire temperature profile is often not required
- Random walk method for thermal analysis
  - Equivalent to the P/G analysis problem, w/ thermal resistors
  - P/G Random walk methods [Qian, TCAD'05][Miyakawa, GLSVLSI'11]
  - [Wong, DATE'06]: used for thermal via planning in 3D IC

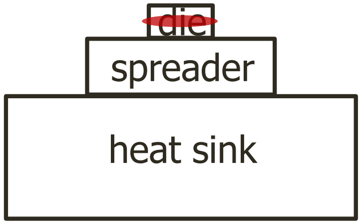
$$V_x = \sum \frac{g_i}{\sum g_i} V_i - \frac{I_x}{\sum g_i}$$

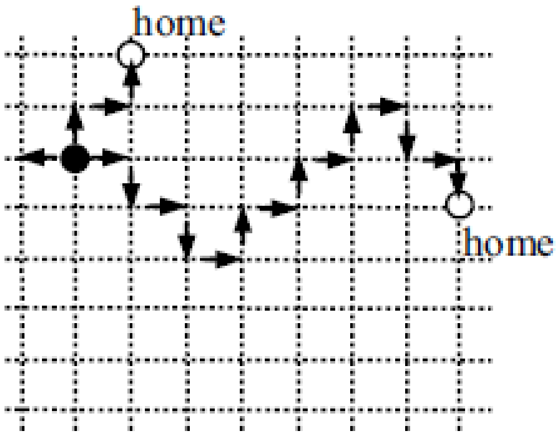


  - Drawbacks of existing works
    - Not consider characteristics of IC thermal problem (boundary condition, geometry features)
    - The computational speed is very slow

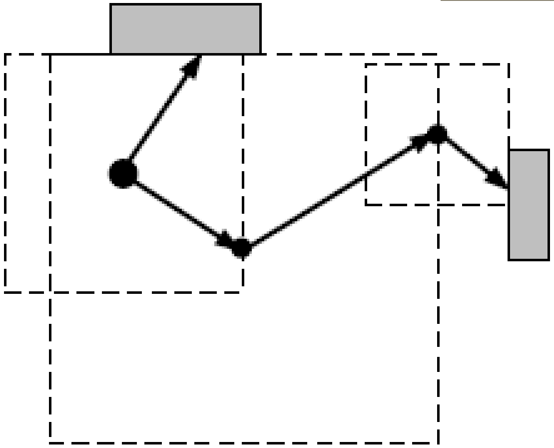
A Hybrid Random Walk Method

- Main ideas
  - Heat sink components (e.g. Pyramid-shape)
  - Slow speed of RW (called GRW) is due to the long length of a walk path
  - Another RW (FRW) is able to reduce the length of a walk path
  - FRW encounters difficulty if there are the *source item* and *Neumann, convective boundary condition*





Generic random walk

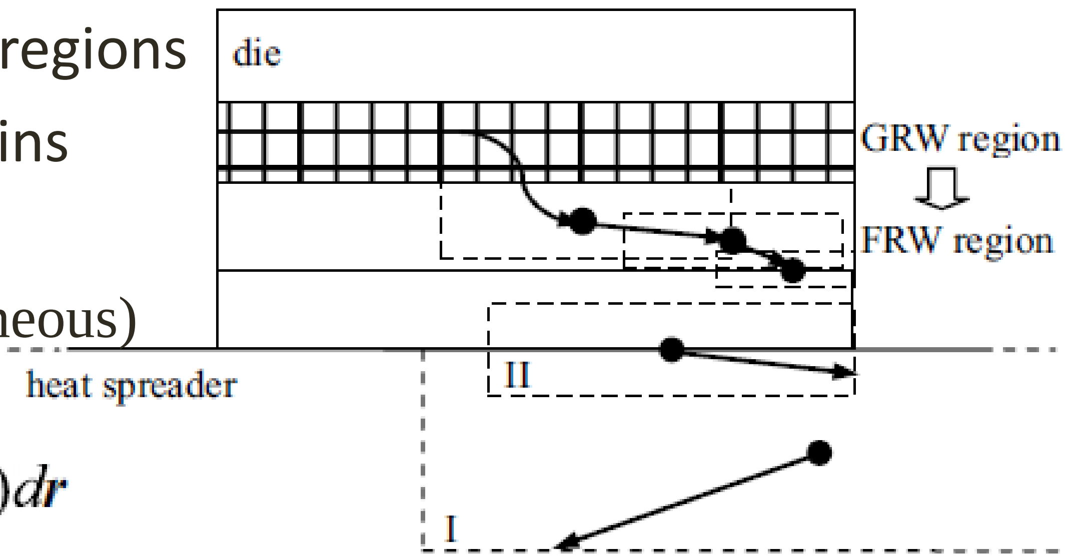


Floating random walk

Can We Combine Them?

A Hybrid Random Walk Method

- GRW+FRW
  - Perform FRW in simple regions
  - *Cuboid* transition domains
    - I (homogeneous)
    - II (half &half homogeneous)
  - FRW transition



$$T_c = \oint_S G_s(\boldsymbol{r}) T(\boldsymbol{r}) d\boldsymbol{r}$$

  - Pre-characterize the transition domains with a hop-target table
  - Neumann boundary: path reflection / special transition domain
  - Convective boundary: Large $\boldsymbol{R_{amb}}$ barriers GRW hop; convective-specific transition domain

A Hybrid Random Walk Method

- Test cases
  - Case 1: A 4-core 2D chip (176W in 1x1 die)
  - Case 2: A 2D chip imitating the Power6 (175W in 1.6x2 die)
- Experimental results
  - Hybrid0: GRW+FRW; Hybrid1: Neumann boundary treatment; Hybrid2: convective boundary treatment (1% 1-σ error)

Average runtime for calculating the temperature of a node (s)

Test case	#node	GRW			Hybrid random walk				
		time	#walk	#hop	hybrid0	hybrid1	hybrid2	#hop	Sp*
1-1	5.24e5	49.8	5471	2.34e5	25.6	23.5	2.76	8.77e3	18
1-2	4.19e6	199	5522	9.28e5	44.8	33.3	4.12	8.13e3	48
1-3	6.55e7	949	6409	5.64e6	54.1	38.8	3.64	7.52e3	261
2-1	5.33e5	35.5	3576	2.52e5	30.1	25.0	1.26	1.05e4	28
2-2	4.26e6	143	3709	9.90e5	41.5	37.6	2.85	1.79e4	50
2-3	6.66e7	762	3281	5.98e6	72.7	48.1	5.17	3.12e4	147

A Hybrid Random Walk Method

- Experimental results
  - Memory overhead: ~81MB for transition-domain characterization
  - The pre-characterization runs only once for same chip structure or chips manufactured by same materials
  - Accuracy validated by Matlab “\”
  - How the aspect ratio of the cuboid transition domain in FRW affects the efficiency of the proposed hybrid method ?

Average Hybrid1’s runtime for calculating a node’s temperature (s)

Aspect ratio	1	5	8	10	12	15	20
Time/node	170	43.3	41.9	41.3	40.9	41.8	42.5

  - Choosing cuboid transition domain (a.r.=10) brings 4.1X speedup

Conclusions

- Domain decomposition method
  - Make the fast thermal solvers workable while appreciating the effect of heat sink components
  - Can beat iterative equation solver for large case
  - Nonconformal discretization grid reduces the computation with negligible loss of accuracy
- Hybrid random walk method
  - Combining GRW and FRW brings one or two orders of magnitude speedup, with some memory overhead
  - Suitable for the scenarios where only the temperature of some local hot-spots is needed

Thanks !