Fast Transient Thermal Simulation Based on Linear System Theory

Yongkui Han, Israel Koren, C. Mani Krishna
ARTS Lab, ECE Dept
University of Massachusetts at Amherst

Thermal Simulation is Important

- Thermal simulation has become a necessity for contemporary microprocessors
- Transient Thermal Simulation Models
  - 1. Finite Elements Method (FEM)
    - Very accurate but computation-intensive
  - 2. Compact thermal RC models
    - Less accurate but faster simulation
- The HotSpot simulator
  - based on the compact thermal RC model - at architecture level
  - developed by LAVA group at Univ. of Virginia

HotSpot is Inefficient

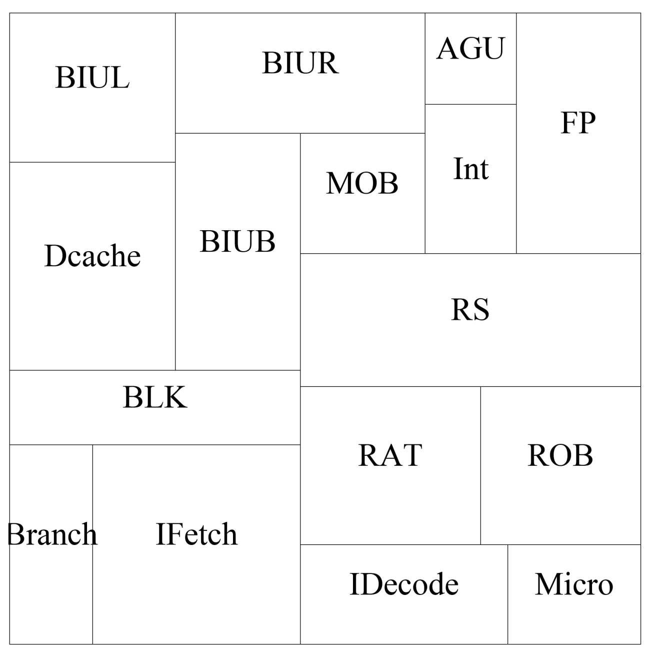
- HotSpot uses the fourth order Runge-Kutta method (rk4) for integration
  - To reduce the truncation error, the step size in rk4 must be small
  - Thousands of rk4 iterations are required when simulating for a 1ms time interval
- HotSpot becomes inefficient when attempting to obtain the temperature profile for a given benchmark

- Our new method improves the simulation speed of HotSpot

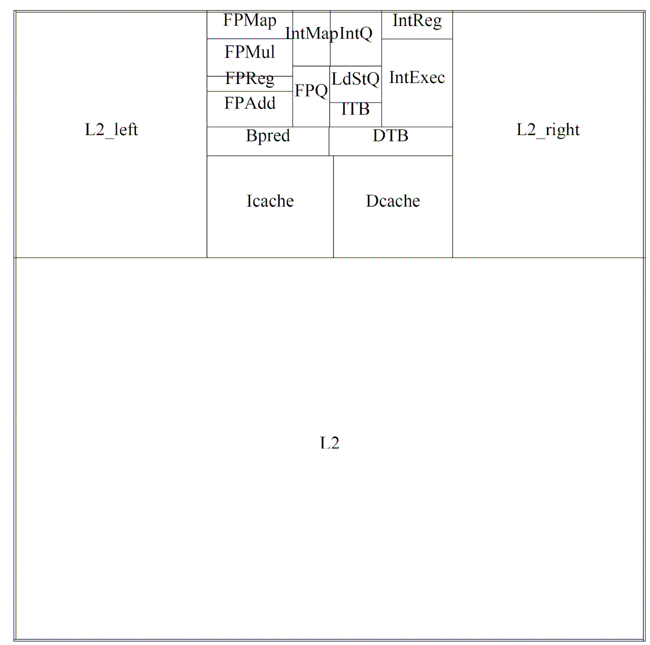
Experiments

- Two microprocessors
- SPEC2000 benchmarks
- Most recent HotSpot v3.0.2 is used

Pentium Pro processor:
16 power inputs, 58 nodes.

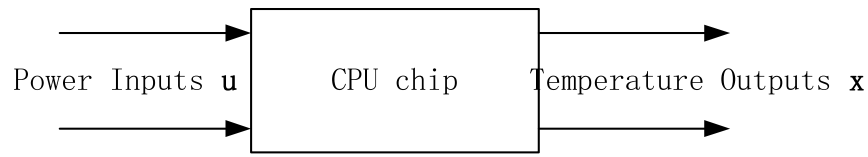


Alpha 21364 processor:
18 power inputs, 97 nodes.



CPU Chip as a Linear System

$$\dot{\mathbf{x}}(t) = \mathbf{F}\mathbf{x}(t) + \mathbf{G}\mathbf{u}(t) \qquad (1)$$



The response of the linear system is:

$$\mathbf{x}(t) = e^{\mathbf{F}t}x(0) + \int_0^t e^{\mathbf{F}(t-\tau)}\mathbf{G}\mathbf{u}(\tau)d\tau \qquad (2)$$

During a time interval [(i-1)Δt, iΔt], the input power u is fixed:

Let $\mathbf{A} = e^{\mathbf{F}\Delta t}, \quad \mathbf{B} = \int_0^{\Delta t} e^{\mathbf{F}(\Delta t-\tau)}\mathbf{G}d\tau$

u(i) is the input power during the interval [(i-1)Δt, iΔt]

$$\mathbf{x}(n) = \mathbf{A}\mathbf{x}(n-1) + \mathbf{B}\mathbf{u}(n-1) \qquad (3)$$

The dimensions of x, u depend on specific processor analyzed.

Time Invariant Linear Thermal System (TILTS) method

- TILTS
  - Precompute matrices **A** and **B** using rk4 method
  - Compute temperatures using (3)

- Many rk4 iterations are replaced by simple matrix multiplications

- TILTS does not incur any accuracy loss compared to the HotSpot simulator

Computation Reductions in TILTS

Comparing the number of Floating-Point Multiplications (FPM) in HotSpot and TILTS

processor	Δt	#FPM in HotSpot	#FPM in TILTS	ratio
Pentium Pro	5 μs	267670	4292	62
Alpha 21364	3.33 μs	514197	11155	46

Convolutional TILTS

- Convolutional TILTS (CTILTS) algorithm
  - Precompute matrices $\mathbf{A}^j$ and A^jB, j=1,2,…,p
  - Perform the convolution between the input power trace and the step response of the system
- CTILTS reduces the number of floating-point multiplications (FPM) by 5 or 6 times
- Small memory overhead

processor	interval	memory size	#FPM in TILTS	#FPM in CTILTS	ratio
Pentium Pro	5.1 ms	7.25 MB	4395008	953636	4.61
Alpha	3.4 ms	13.64 MB	11422720	1797313	6.36

Experimental Results

processor	prog	HotSpot	TILTS	speedup	CTILTS	speedup
Pentium Pro	*gcc*	15800s	237s	67x	59s	268x
Pentium Pro	*gzip*	15790s	238s	66x	59s	268x
Pentium Pro	*bzip2*	15792s	237s	67x	59s	268x
Pentium Pro	*art*	15800s	238s	66x	59s	268x
Pentium Pro	*mgrid*	15790s	238s	66x	59s	268x
Alpha	*gcc*	28010s	592s	47x	129s	217x
Alpha	*gzip*	28030s	591s	47x	129s	217x
Alpha	*bzip2*	28020s	592s	47x	129s	217x
Alpha	*art*	28010s	591s	47x	129s	217x
Alpha	*mgrid*	28020s	592s	47x	129s	217x

- **TILTS**
  - 67x for Pentium Pro, 47x for Alpha
- **CTILTS**
  - 268x for Pentium Pro, 217x for Alpha

Conclusions

- **Our Time Invariant Linear Thermal System (TILTS) method can greatly improve transient thermal simulation performance:**
  - 268 times faster than HotSpot for Pentium Pro processor
  - 217 times faster than HotSpot for Alpha 21364 processor
- **TILTS does not incur any accuracy loss compared to the HotSpot simulator**