A Stochastic Local Hot Spot Alerting Technique

Hwisung Jung and Massoud Pedram

University of Southern California
Dept. of Electrical Engineering

Asia and South Pacific - Design Automation Conference 2008

Overview

- Motivation and Background
- Uncertainty-Aware Estimation Framework
- The Proposed Hot Spot Alerting Algorithm
- Experimental Results
- Conclusion

Introduction

- As IC process geometries shrink below 65nm
  - ❑ Higher power density
  - ❑ Higher operating temperature
  - ❑ Lower circuit reliability
- Thermal management is a major design requirement
  - ❑ Elevated temperature increases leakage and lowers performance
  - ❑ Gate oxide lifetime is highly dependent on the T_J of IC
- Local hot spots are becoming more prevalent
  - ❑ Thermal runaway; CMOS latch-up
  - ❑ Functional and/or timing errors; accelerated aging

➡ *Timely identification and elimination of hot spots is a key goal*

Some Relevant Prior Work

- K. Skadron, et al. (ISCA 2003)
  - An architectural-level thermal model, *HotSpot*
- W. Huang, et al. (DAC 2004)
  - A compact thermal model for temperature-aware design
- D. Brook, et al. (HPCA 2001)
  - A thermal control mechanism, *Wattch*
- J. Srinivasan, et al. (ICS 2003)
  - Predictive dynamic thermal management
- R. Mukherjee, et al. (DAC 2006)
  - Thermal sensor allocation and placement

High Level Explanation of the Problem

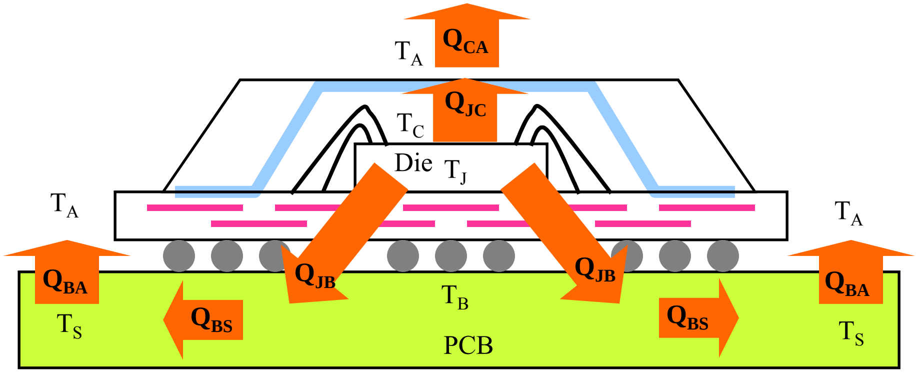
- Many researchers have examined techniques for:
  - Thermal modeling considering the chip-package-heat sink interface characteristics
  - Dynamic thermal management to control rapid temperature rise
- These techniques suffer from the following:
  - Thermal modeling based on equivalent circuit models cannot accurately account for real structures with complex shapes and boundary conditions
  - Thermal management utilizing external thermal sensors tend to provide readings which are often late and/or are inaccurate
  - On-chip sensors are expensive to employ and suffer from noise, nonlinearity, and low response speed

⬇

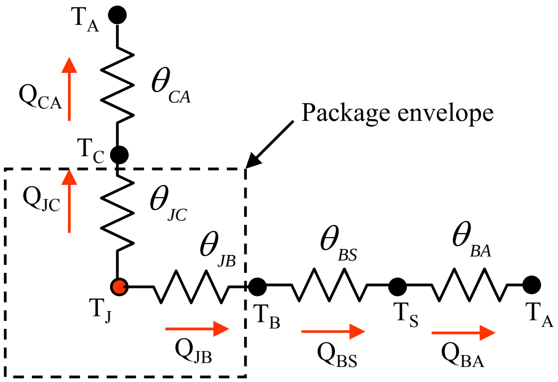
- There is thus uncertainty in identifying local hot spots
  - Stochastic modeling and early prediction of local (zone-based) temperatures is a necessity

Background (1/3)

- IC package can be characterized by its thermal resistance
  - Heat is transferred from the die into the ambient air
  - Value of the thermal resistance along with the on-chip power dissipation determines the temperature rise of the junction above a reference point



Heat flow in the PBGA + HS package



One of the IC package heat transfer paths and the corresponding thermal resistive model

Background (2/3)

- Thermal resistance is defined as

$$\theta_{JX} = (T_J - T_X) / P$$

  - ☐ θ_{JX} is the thermal resistance from device junction to specific point
  - ☐ T_J is the device junction temperature
  - ☐ T_X is the temperature of some specific reference point
  - ☐ P is the device power dissipation

- When the reference points are selected as the ambient air, PCB, and case top, we have

$$\theta_{JA} = \frac{(T_J - T_A)}{P} \qquad \theta_{JB} = \frac{(T_J - T_B)}{P} \qquad \theta_{JC} = \frac{(T_J - T_C)}{P}$$

Junction-to-air Junction-to-board Junction-to-case

  - ☐ T_A, T_B and T_C denote the temperatures of ambient air, PCB, and case top

Background (3/3)

- Junction-to-air thermal resistance may be calculated as

$$\theta_{JA} = \left(\frac{1}{\theta_{JB} + \theta_{BS} + \theta_{BA}} + \frac{1}{\theta_{JC} + \theta_{CA}} \right)^{-1}$$

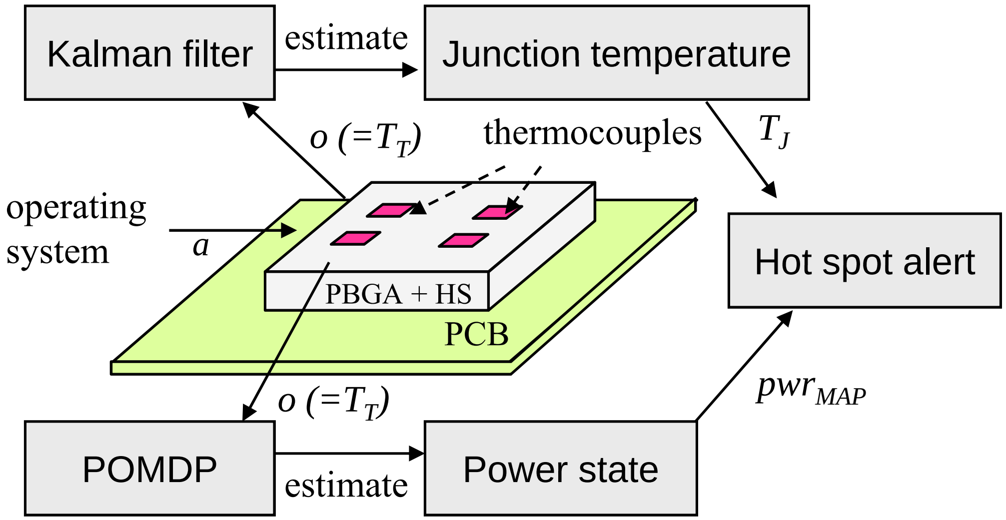
  - ❑ Junction temperature is estimated as: $T_J = T_A + P \cdot \theta_{JA}$
  - ❑ The goal of thermal design is to make the θ_{JA} value small so that the junction temperature T_J does not exceed some specified maximum value
- θ_{JA} cannot be modeled directly due to the complexity of thermal models for the substrate, package, and board stack and the cooling system which is in-place
  - ❑ In addition, θ_{JA} is typically taken to be a single parameter under the assumption that P is distributed uniformly across the die, which is unrealistic

Overview of the Proposed Solution

- We develop a hot spot alerting technique by estimating the *junction temperature* and the *system power state*
  - The thermal time constant of the die is larger than the circuit clock speed
  - Recognizing a temperature rise by relying on thermal sensors and subsequently employing thermal control mechanisms can result in too late a response (i.e., a corrective action to avoid thermal problems)
- Our proposed hot spot alerting technique combines:
  - State estimation for the junction temperature using Kalman Filter (KF)
  - State estimation for the system power dissipation using Partially Observable Markov Decision Process (POMDP) model

Architecture of the Estimation Framework

- Use (external temperature) observations to estimate the *Junction Temperature* (T_J) and *the Power state* (pwr_{MAP})



Uncertainty-aware hot spot warning framework

Temperature Estimation

- Kalman Filter
  - Estimate the state of a system based on the previous state, previous action, and the current observation
- Kalman Filter-based Temperature Estimation (KFTE) Framework
  - *s* is a state representing the junction temperature T_J
  - *a* is an action (voltage-frequency assignment) issued by the DP/TM
  - *o* is a temperature observation T_T
  - **X** denotes a state transition matrix
  - **Y** denotes an action-input matrix
  - **Z** denotes an observation matrix

$$s^{t+1} = \mathbf{X}s^{t} + \mathbf{Y}a^{t} + u^{t}, \quad u^{t} \sim N(0, Q^{t})$$

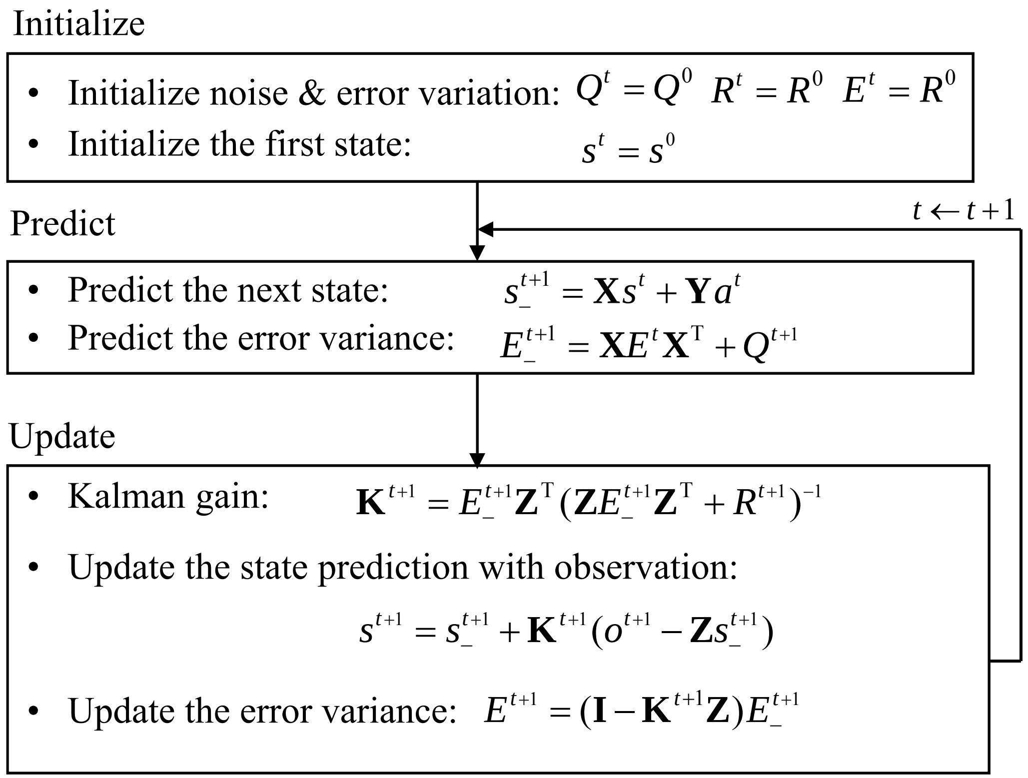
u: the temperature variation

$$o^{t+1} = \mathbf{Z}s^{t+1} + v^{t+1}, \quad v^{t+1} \sim N(0, R^{t})$$

v: the observation noise

Temperature Estimation (Cont'd)

- Estimation of the junction temperature of the chip
  - ❑ Based on the Kalman Filter



Initialize

- Initialize noise & error variation: $Q^t = Q^0 \quad R^t = R^0 \quad E^t = R^0$
- Initialize the first state: $s^t = s^0$

Predict

$t \leftarrow t+1$

- Predict the next state: $s_-^{t+1} = \mathbf{X}s^t + \mathbf{Y}a^t$
- Predict the error variance: $E_-^{t+1} = \mathbf{X}E^t\mathbf{X}^{\mathrm{T}} + Q^{t+1}$

Update

- Kalman gain: $\mathbf{K}^{t+1} = E_-^{t+1}\mathbf{Z}^{\mathrm{T}}(\mathbf{Z}E_-^{t+1}\mathbf{Z}^{\mathrm{T}} + R^{t+1})^{-1}$
- Update the state prediction with observation:

$$s^{t+1} = s_-^{t+1} + \mathbf{K}^{t+1}(o^{t+1} - \mathbf{Z}s_-^{t+1})$$

- Update the error variance: $E^{t+1} = (\mathbf{I} - \mathbf{K}^{t+1}\mathbf{Z})E_-^{t+1}$

Junction temperature estimation

Power State Estimation

- POMDP (Partially Observable Markov Decision Process)
    - It is a model for deciding how to act in an accessible, stochastic environment with a known transition model
    - It can model uncertainty and non-determinism in observations
- POMDP is a tuple <*S, A, O, T, Z*> such that
    - *S* is a finite set of discrete states
    - *A* is a finite set of actions
    - *O* is a finite set of observations
    - *T* is a state transition probability function
    - *Z* is an observation function
- POMDP may be transformed to a regular (continuous state) MDP by defining and maintaining a *belief state,* b^t
    - Vector *b* gives the probability distribution over the possible states of the system, $\sum_{s \in S} b^t(s) = 1$

Power State Estimation (Cont'd)

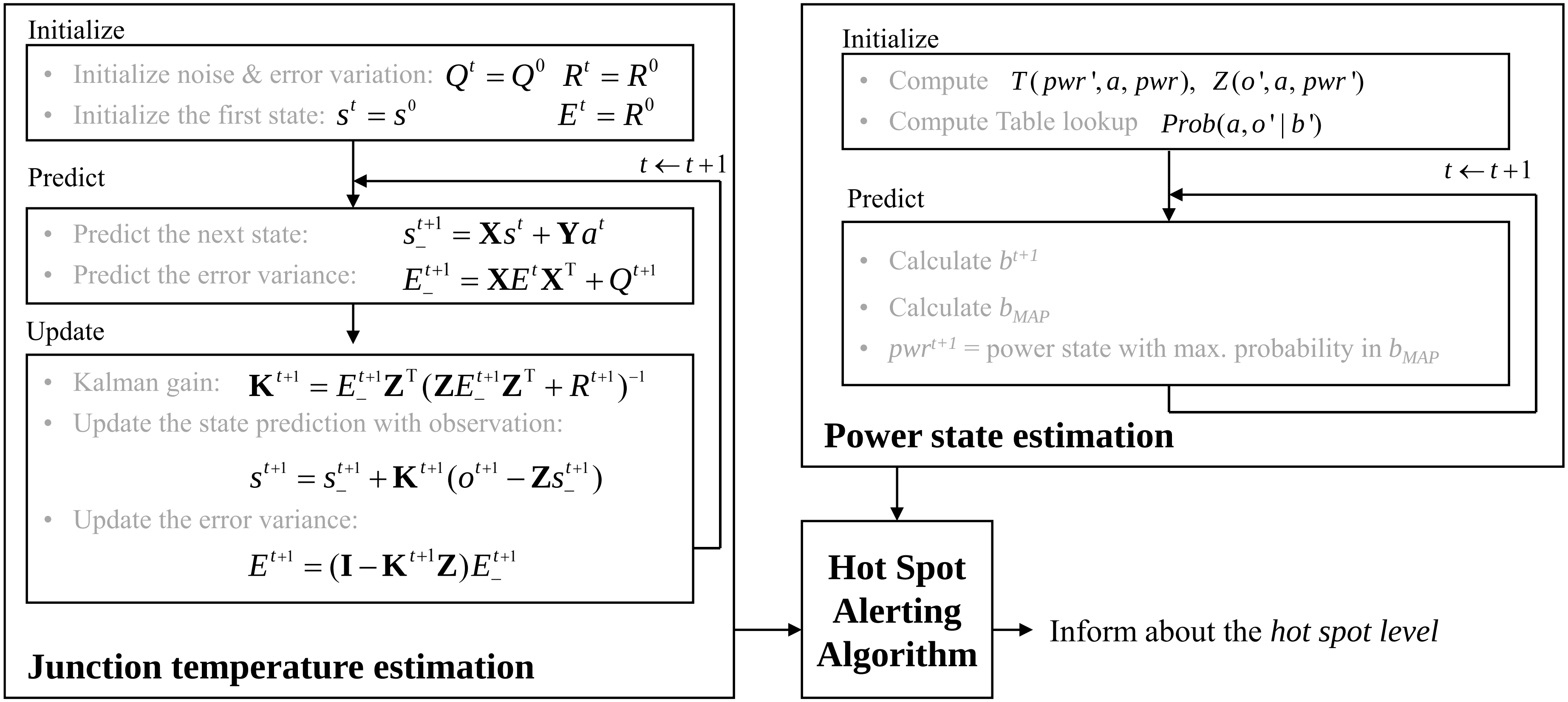
- Estimation of the power state of the system
  - Based on the Bayesian formula:

$$Prob(b^t \mid h^t) = \frac{Prob(h^t \mid b^t) \cdot Prob(b^t)}{Prob(h^t)}$$

  - *h* is a stream of action-observation pairs
  - *Prob*(b^t | *h*) is the posterior probability density function
  - *Prob*(h^t | b^t) is the likelihood function
  - *Prob*(h^t) is the prior distribution
  - *Prob*(b^t) is the probability of belief state
- The most probable power state can be computed as the maximum a posterior (MAP) estimate under Markovian assumption as:

$$b_{MAP} = \arg\max_{b \in B} Prob(b^t \mid h^t) = \arg\max_{b \in B} Prob(h^t \mid b^t) \cdot Prob(b^t)$$
$$= \arg\max_{b \in B} Prob(a^{t-1}, o^t \mid b^t) \cdot Prob(b^t)$$

Flow of the Estimation Framework



Junction temperature estimation

Initialize

- Initialize noise & error variation: $Q^t = Q^0$ $R^t = R^0$
- Initialize the first state: $s^t = s^0$ $E^t = R^0$

$t \leftarrow t+1$

Predict

- Predict the next state: $s_-^{t+1} = \mathbf{X}s^t + \mathbf{Y}a^t$
- Predict the error variance: $E_-^{t+1} = \mathbf{X}E^t\mathbf{X}^{\mathrm{T}} + Q^{t+1}$

Update

- Kalman gain: $\mathbf{K}^{t+1} = E_-^{t+1}\mathbf{Z}^{\mathrm{T}}(\mathbf{Z}E_-^{t+1}\mathbf{Z}^{\mathrm{T}} + R^{t+1})^{-1}$
- Update the state prediction with observation:

$$s^{t+1} = s_-^{t+1} + \mathbf{K}^{t+1}(o^{t+1} - \mathbf{Z}s_-^{t+1})$$

- Update the error variance:

$$E^{t+1} = (\mathbf{I} - \mathbf{K}^{t+1}\mathbf{Z})E_-^{t+1}$$

Power state estimation

Initialize

- Compute $T(pwr', a, pwr)$, $Z(o', a, pwr')$
- Compute Table lookup $Prob(a, o' | b')$

$t \leftarrow t+1$

Predict

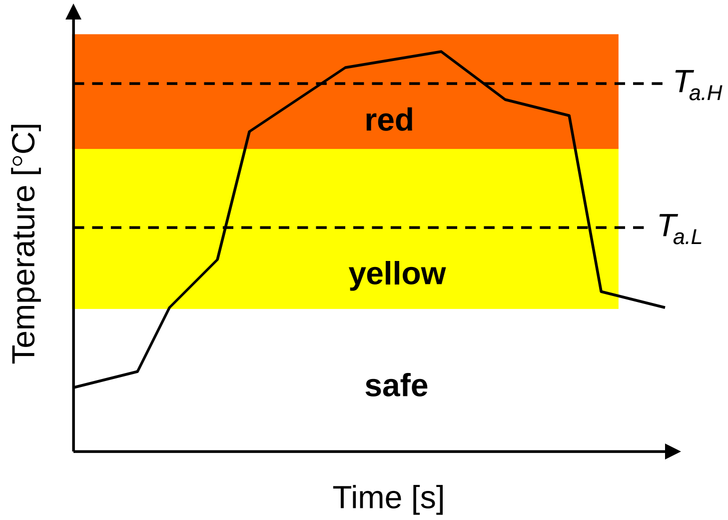
- Calculate b^{t+1}
- Calculate b_{MAP}
- pwr^{t+1} = power state with max. probability in b_{MAP}

Hot Spot Alerting Algorithm

Inform about the *hot spot level*

Hot Spot Alerting Algorithm

- The proposed hot spot alerting algorithm
  - Define *red* and *yellow* hot spot levels in terms of degree of thermal threat
  - $T_{a.H}$ and $T_{a.L}$ are pre-defined temperature thresholds ($T_{a.L} < T_{a.H}$)
  - P_a is a power dissipation threshold
  - $G_{j.a}$ is a temp. gradient threshold



```
do forever
    predict the junction temperature, $T_j^{t+1}$
    predict the power state of the processor, $pwr^{t+1}$
        if $T_j^{t+1} \ge T_{a.H}$
            alert red hot spot
        else if $T_{a.L} \le T_j^{t+1} < T_{a.H}$
            if $pwr^{t+1} \ge P_a$
                alert red hot spot
            else
                alert yellow hot spot
        else
            if $\partial T_j / \partial t \ge G_{j,a}$
                alert yellow hot spot
return hot spot level
```

Experimental Setup

- The technique is applied to a 32-bit RISC processor
- Set the parameter values for estimation framework

	power [W] state			observation [°C] state		
	pow_1	pow_2	pow_3	o_1	o_2	o_3
range	[0.6 1.4]	(1.4 2.2]	(2.2 3.0]	[86 93]	(93 100]	(100 107]

- PBGA package thermal performance data (T_A=70°C)

Air velocity		T_{J_max}[°C]	T_{T_max}[°C]	ψ_{JT} [°C/W]	θ_{JA} [°C/W]
m/s	ft/min				
0.51	100	107.9	106.7	0.51	16.12
1.02	200	105.3	104.1	0.53	15.62
2.03	300	102.7	101.2	0.65	14.21

[ψ_{JT} : Junction-to-top of package thermal characterization parameter]

Experimental Results (1/3)

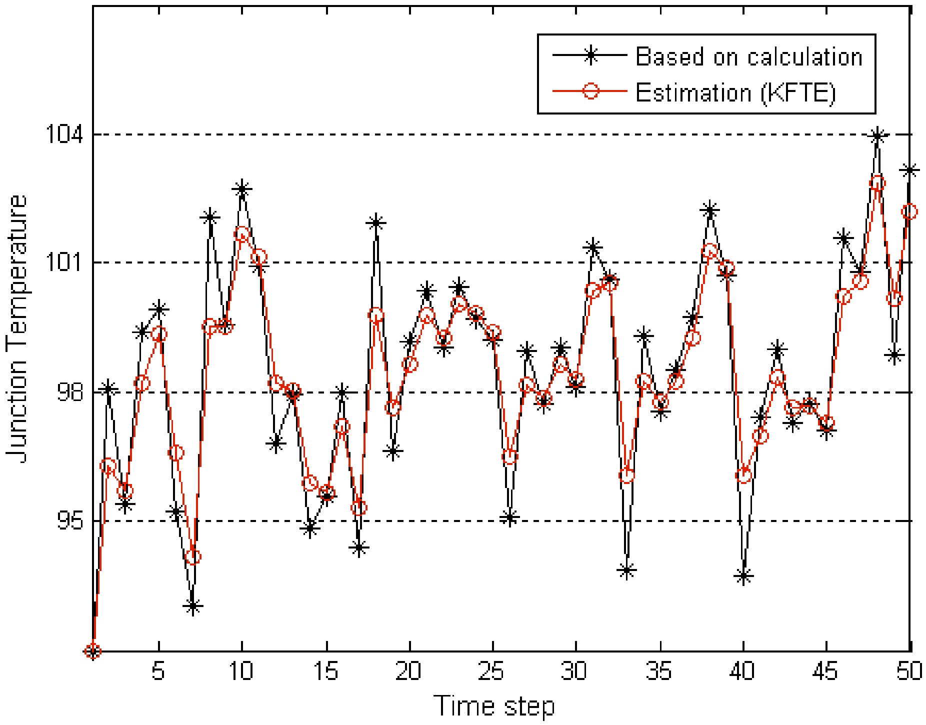
- Choose a sequence of 50 application programs
  - E.g., gap_1 - $gzip_2$ - gap_3 -gcc_4 -…- gap_{50}.
- Trace of the junction temperature
  - Golden result:

    Calculate T_T *from*

    $T_T = T_A + P\cdot(\theta_{JA} - \psi_{JT})$, where

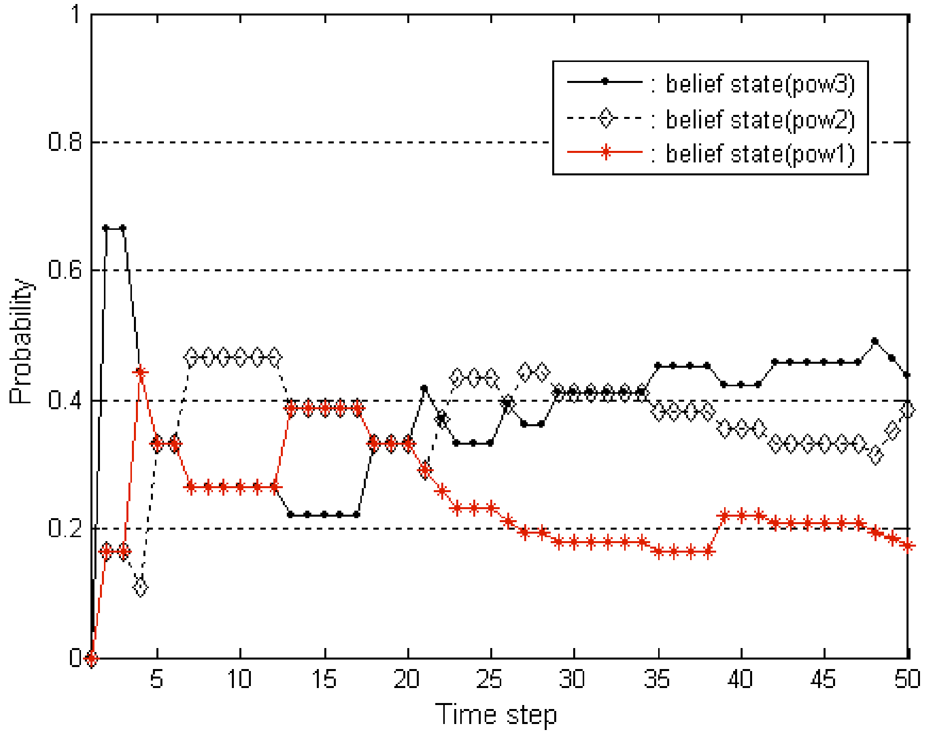
    $P \sim N(P_{sim}, (\Delta P)^2)$
  - Estimated result:

    Based on KFTE



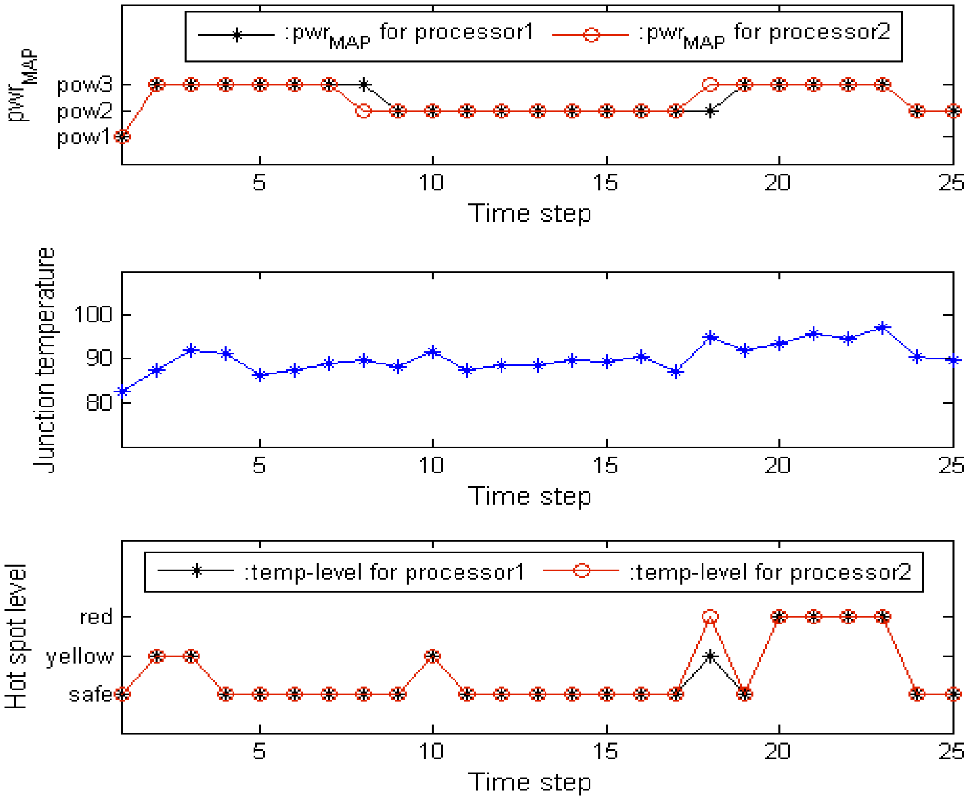
Experimental Results (2/3)

- Trace of the power belief state
    - e.g., belief state(pow_1): probability for power state pow_1
    - Evaluated by the POMDP-based Power Profile Estimation (P3E) method



Experimental Results (3/3)

- Evaluation of the proposed hot spot alerting algorithm
  - Hot spot levels defined: *red*, *yellow*, and *safe*



Conclusion

- The stochastic hot spot alerting technique based on
  - ❑ estimation of the junction temperature of the device and
  - ❑ prediction of the power state of the system
- The proposed uncertainty-aware estimation framework efficiently captures
  - ❑ stochastic behavior of the system
  - ❑ PVT variations in system performance parameters, and
  - ❑ inaccuracies in temperature measurements
- Experimental results demonstrate that the proposed technique provides early warning about thermal threats under large variations