Feedback Thermal Control for Multicore Real-Time Systems

Yong Fu, Chenyang Lu
Washington University in St. Louis

Nicholas Kottenstette, Xenofon Koutsoukos
Vanderbilt University
Thermal Control for Real-Time Systems

- Prevent thermal failure of multicore processors

- Guarantee real-time performance
  - Avoid hardware throttling → unpredictable slowdown
  - Enforce schedulable utilization bound

- DVFS: Dynamic Voltage Frequency Scaling
  - Voltage/frequency → power consumption → temperature
  - Frequency → execution time → utilization
Challenges

- Meet both thermal and real-time requirements
- Uncertainties: power consumption, ambient temperature
- Multicore: interaction of thermal dynamics among cores
- DVFS limitations
  - Limited discrete frequencies
  - All cores share the same voltage/frequency
- Run-time efficiency
Example: Intel Core 2 Duo (T9400)

- Two cores with Intel SpeedStep technology

- A small number of discrete frequencies
  - 0.8GHz, 1.6GHz, 2.53GHz

- Two cores always run at the same frequency
  - Share only one DVFS control circuit

www.intel.com
Feedback Control Problem

- Objective: maximum core temperature tracks set point $y_s$
- Constraint: core utilization $\leq$ schedulable utilization bound $U_b$

$$U = \sum_{i=1}^{n} \frac{C_i}{P_i} \leq U_b$$

- Example: $U_b(n) = n(2^{1/n} - 1)$ for rate monotonic ($n$: number of tasks)
- Actuator: DVFS

- Solution: *Real-Time Multicore Thermal Control (RT-MTC)*
RT-MTC: Feedback Control Loop

- Pulse Width Modulation
- Proportional Control with Saturation
- Multicore Processor
Control Design

- **Proportional Control with Saturation**
  - Saturation corresponds to frequency limits

- **Pulse Width Modulation (PWM)**
  - Transform continuous control output to discrete frequencies
  - Choose two adjacent frequencies \( (f_{\text{high}}, f_{\text{low}}) \) and switch time \( t_{sw} \)
    - Schedulable utilization bound \( \rightarrow \) lowest frequency allowed
  - Switch from \( f_{\text{high}} \) to \( f_{\text{low}} \) at \( t_{sw} \)

- **Passivity based stability analysis**
  - Passivity: no more energy output than energy stored in the system
    - Passivity \( \rightarrow \) stability for our system
  - Accommodate nonlinearity induced by Max and saturation
  - Determine proportional control gain
Thermal RC Model

Model Identification

- Benchmarks from SPEC CPU 2006 & Mibench (Bzip2, CRC)

Models fitness level >80%
Experiments

- User-level implementation on Linux 2.6.32
- Mixed workload includes Bzip2 and CRC
- Lenovo W500, Intel T9400 (2.53, 1.6, 0.8 GHz)
- Temperature bound 60 °C

Track temperature set point while enforcing utilization bound
Simulations

- Thermal/power parameters [S. Martin 2002, D. Bild 2008]
- CPU frequencies: 0.8, 1.2, 1.6, 2.0 GHz
- Temperature bound: 60°C

Baselines

- **Reactive**: reset frequency when temperature reaches set point
  - Cannot deal with uncertainty in thermal parameters
- **Model Predictive Control (MPC)**
  - **PWM**: use Pulse-Width Modulation to handle limited frequency
  - **QUAN**: direct quantization
  - Computation intensive
Constant Power Variation (Power Ratio = 4)
Dynamic Power Variation

RT-MTC

MPC-QUAN

MPC-PWM

Reactive
Related Works

- Thermal and utilization control for single core
  - X. Fu [RTCSA 2009]
    - DVFS + task rate adaptation
    - Model Predictive Control (MPC): computation intensive
  - Y. Fu [RTAS 2010]
    - Task rate adaptation
    - Efficient nested control structure

- Thermal control of multicore processors [F. Zanini 2010]
  - Model Predictive Control
  - No utilization control => not real-time
  - Unrealistic assumptions about DVFS
    - Continuous frequency scaling
    - Independent frequency scaling for each core
Conclusion: RT-MTC

- Control temperature under schedulable utilization bound
- Handle practical DVFS limitations
  - Limited frequency levels
  - Shared frequency by all cores
- Efficient control algorithm
  - Proportional Control with Saturation
  - Pulse Width Modulation
- Passivity based control design
  - Ensures stability
  - Robust to uncertainties