Profitability-Based Power Allocation for Speculative Multithreaded Systems

Polychronis Xekalakis, Nikolas Ioannou
Salman Khan and Marcelo Cintra
University of Edinburgh

Introduction

- CMPs are here to stay
- Power and temperature limit performance
- No speedup for single thread applications
  – Use Thread Level Speculation to extract TLP
  – Energy Inefficient

Introduction

- CMPs are here to stay
- Power and temperature limit performance
- No speedup for single thread applications
  – Use Thread Level Speculation to extract TLP
  – Energy Inefficient
- Our proposal:
  – Steal power from non-profitable threads
  – Allocate it where it is useful

Contributions

- Propose power allocation based on thread profitability
- Propose a set of novel predictors to classify threads in profitable and non-profitable ones
- Our approach outperforms state-of-the-art TLS systems:
  - – ED by 21.2% (up to 39.6%)
  - –… while also reducing the temperature

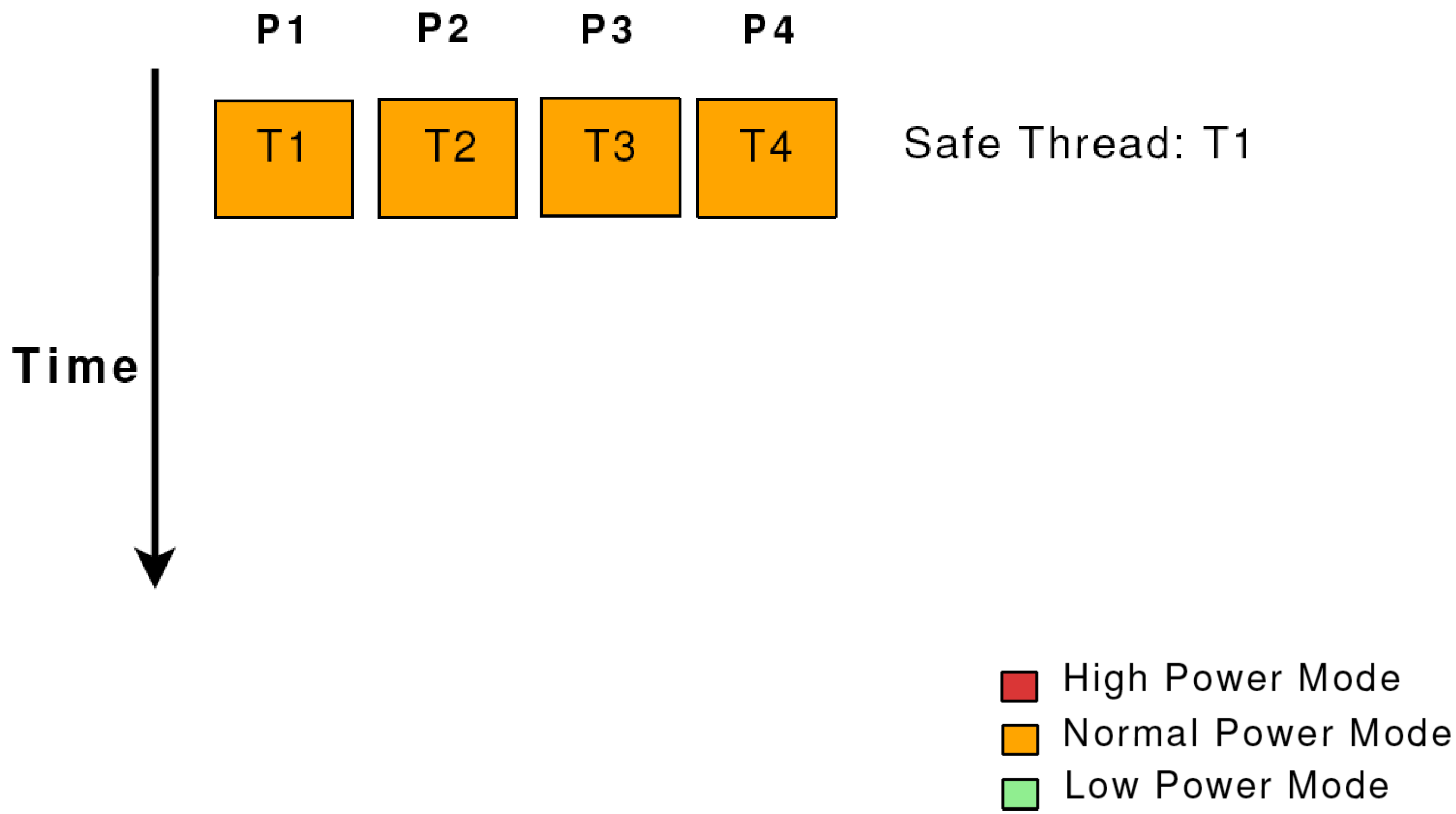
Speculative Multithreading

- Basic Idea: Use idle cores/contexts to speculate on future application needs
  - –TLS: speculatively execute parallel threads
  - –HT/RA: speculatively perform future memory operations
  - –MP: speculatively execute along multiple branch targets
- When speculation fails, power inefficiency results

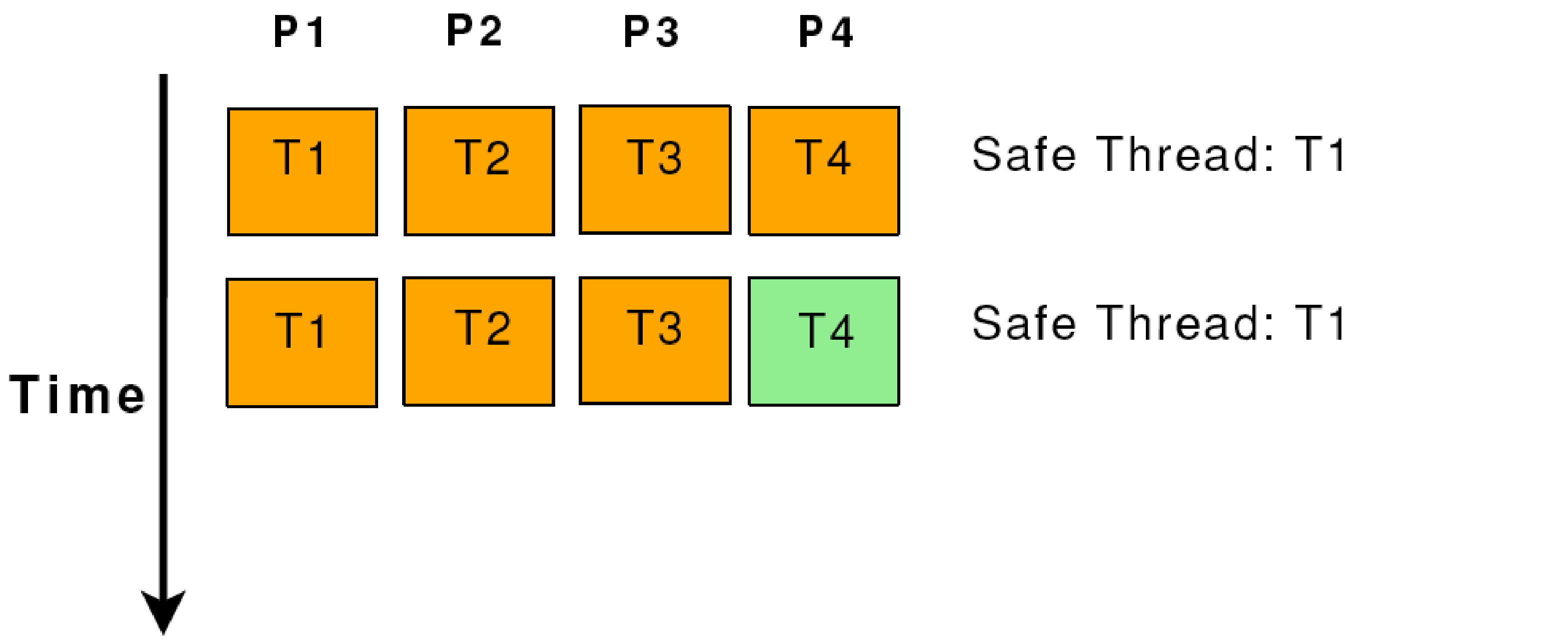
Outline

- Introduction
- Profitability Based Power Allocation
- Estimating Profitability
- Experimental Setup and Results
- Conclusions

Profitability Based Power Allocation



Profitability Based Power Allocation



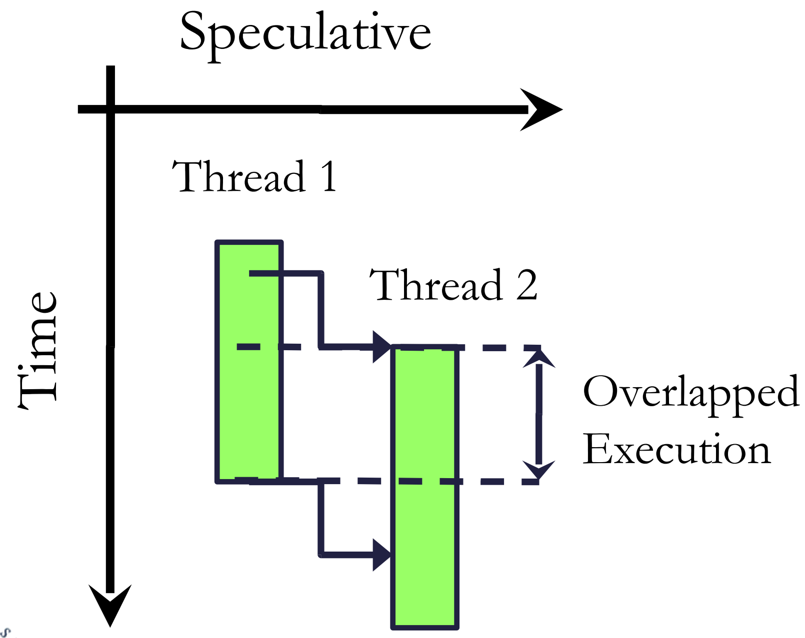
Profitability Based Power Allocation

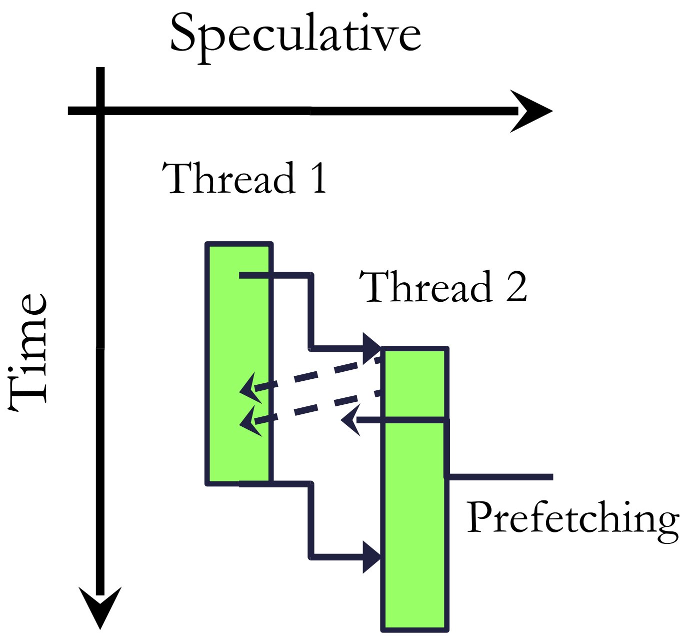
Outline

- Introduction
- Profitability Based Power Allocation
- Estimating Profitability
- Experimental Setup and Results
- Conclusions

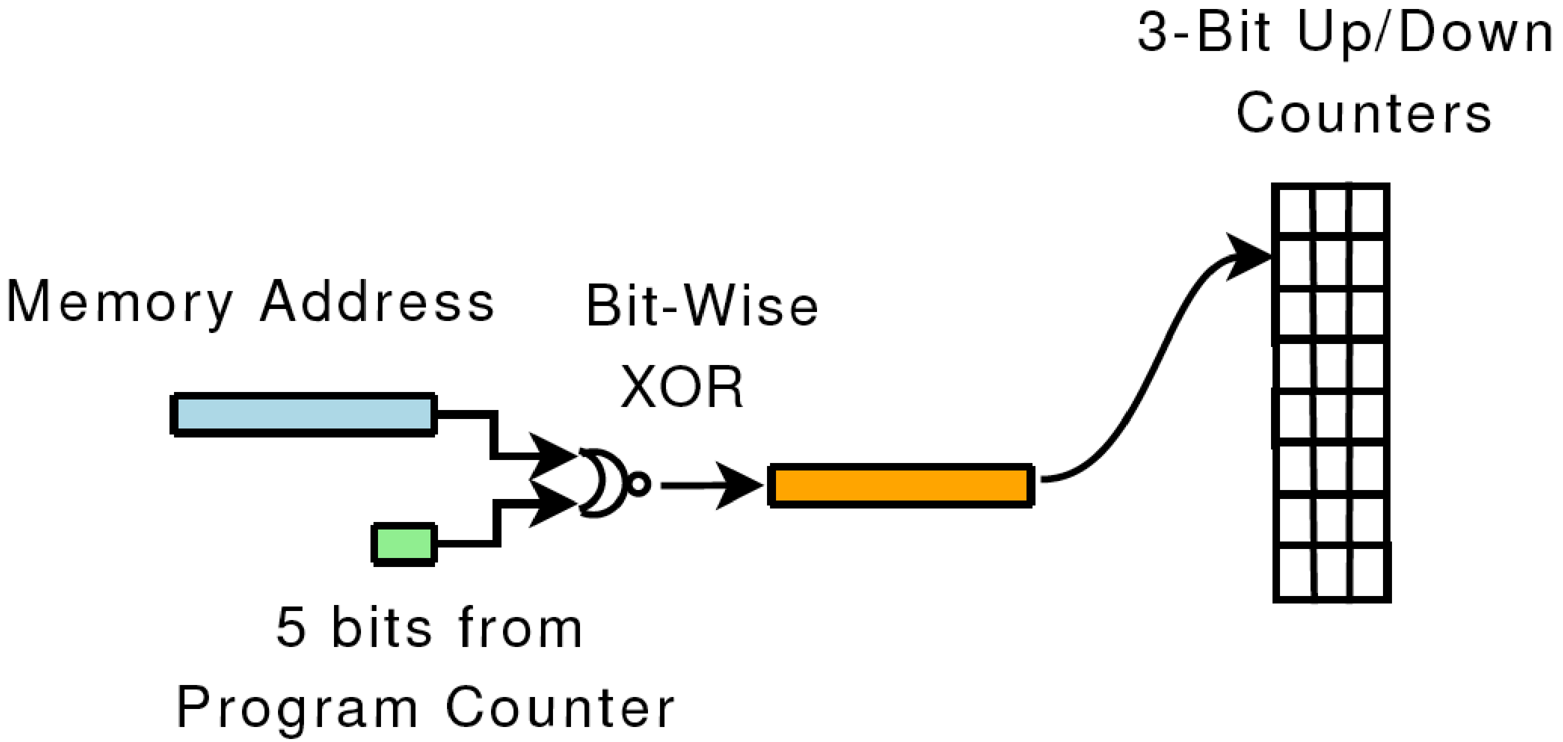
Estimating Profitability

- Benefits for TLS: TLP/ILP
  - TLP (Overlapped Execution)
  - ILP (Prefetching)

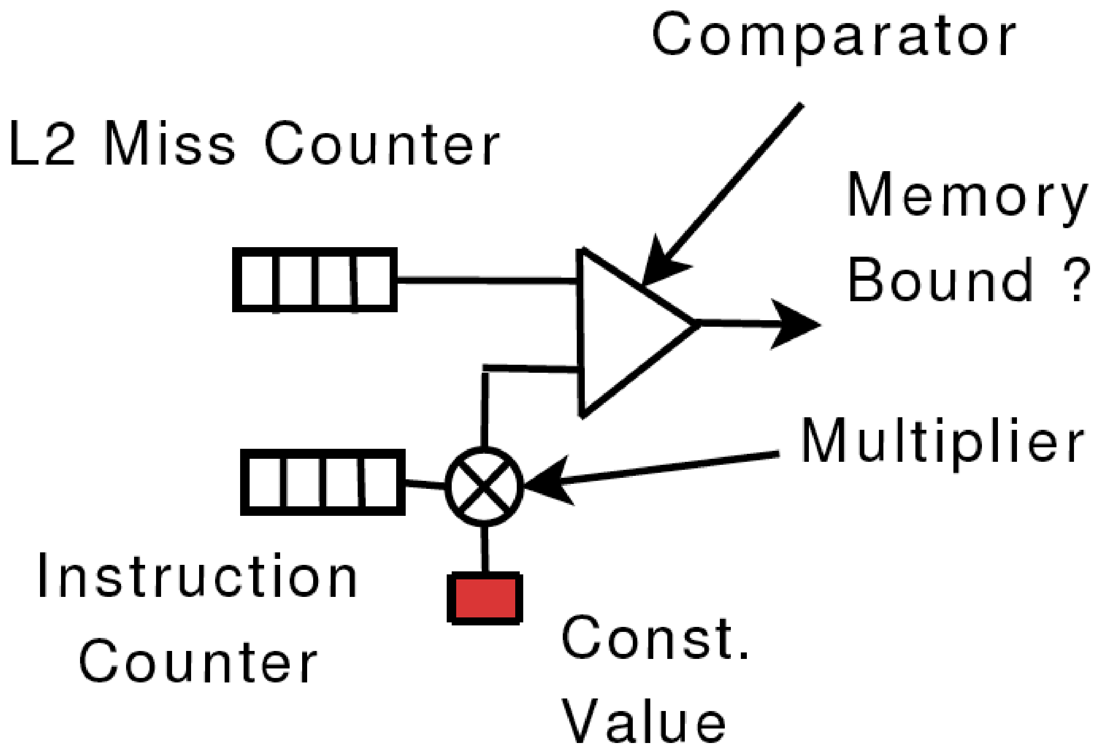




Estimating TLP



Estimating ILP



Power Mode Policy

- For threads that are predicted to squash:
  - – Place in low power mode on first prediction
  - – Place in very low power mode on third prediction
- For threads that are memory bound:
  - – Place in low power mode
- If power budget allows, place safe thread in high power mode

Outline

- Introduction
- Profitability Based Power Allocation
- Estimating Profitability
- Experimental Setup and Results
- Conclusions

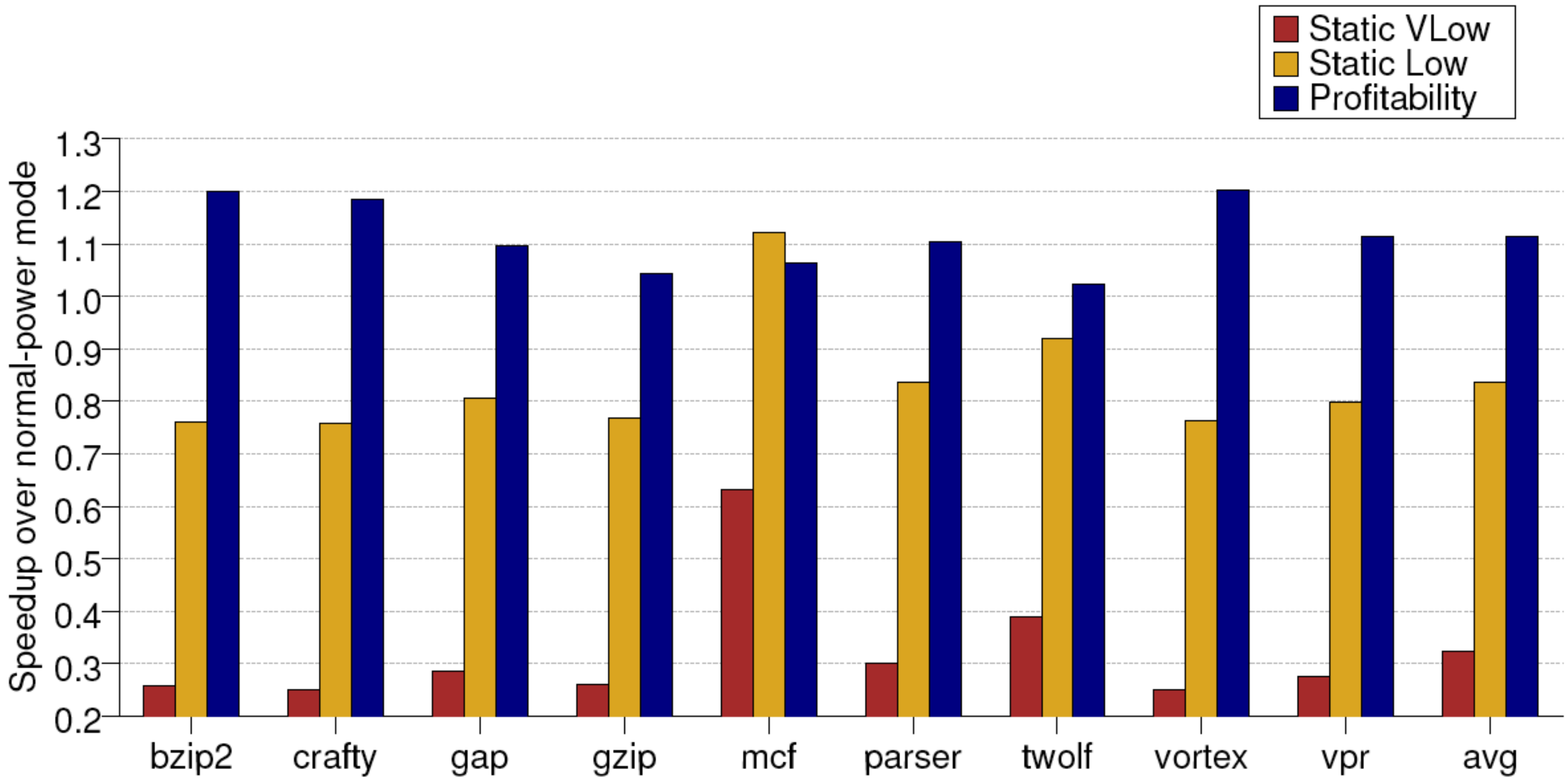
Evaluation Environment

- Simulator, Compiler and Benchmarks:
  - SESC (http://sesc.sourceforge.net/)
  - POSH (Liu et al. PPoPP '06)
  - Spec 2000 Int.
- Architecture:
  - Four way CMP, 4-Issue cores
  - 16KB L1 Data (multi-versioned) and Instruction Caches
  - 1MB unified L2 Caches
  - Inst. window/ROB – 80/104 entries

Power Modes Used

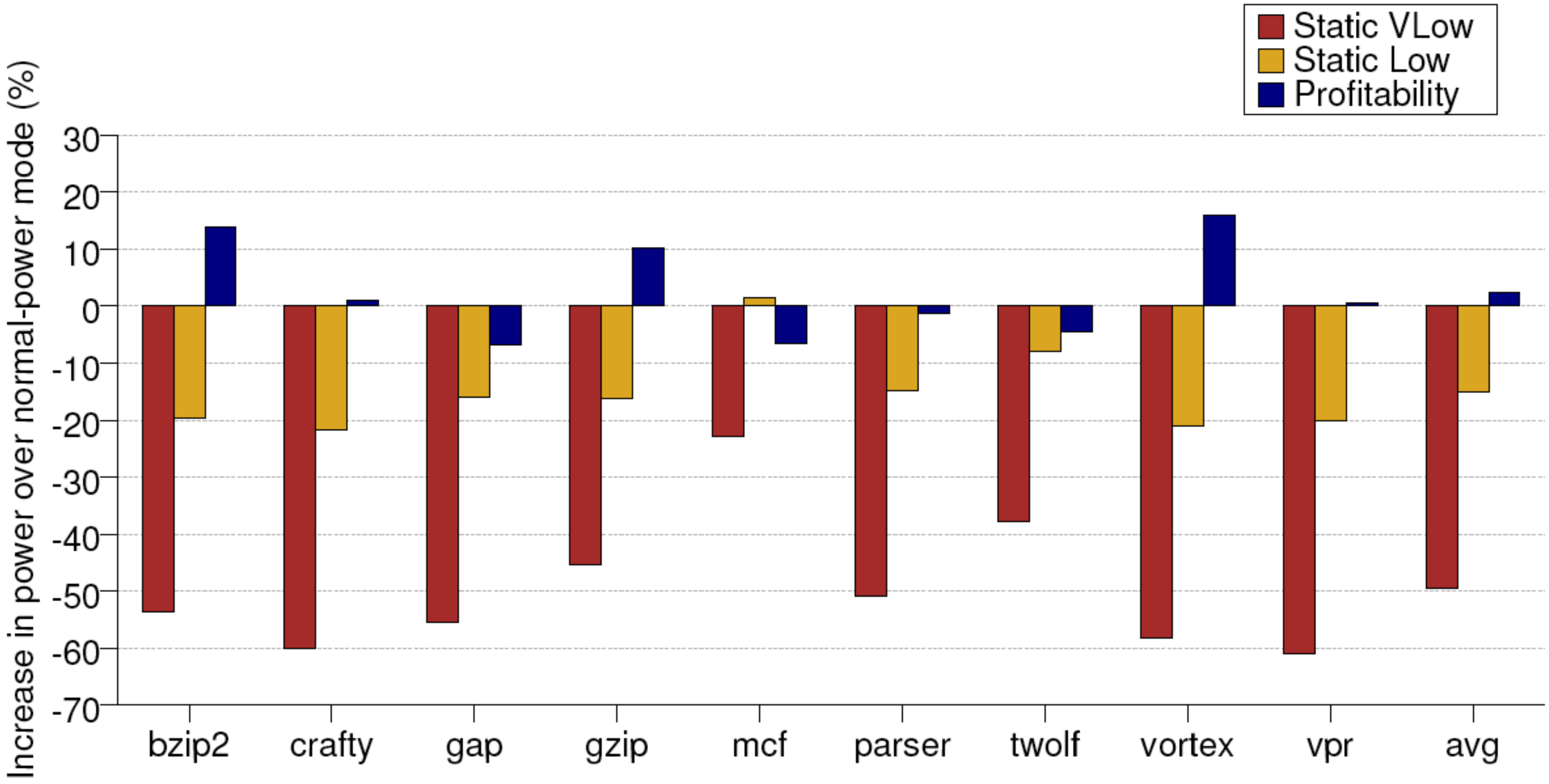
Mode	Voltage	Freq
High Power	1000 mV	5.0 GHz
Normal Power	950 mV	4.0 GHz
Low Power	900 mV	3.0 GHz
Very Low Power	700 mV	1.0 GHz

Performance-Power Analysis



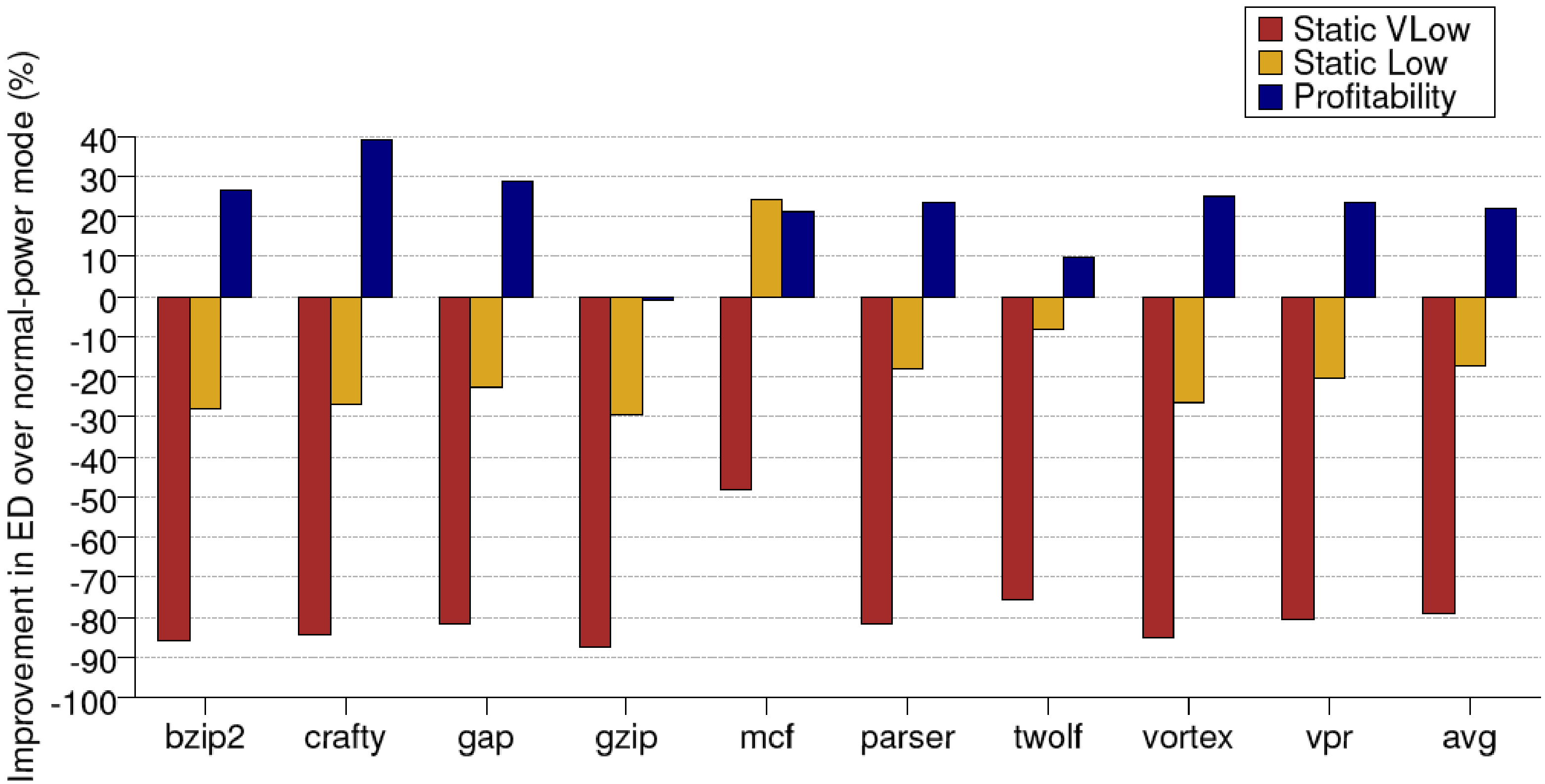
Speedup

Performance-Power Analysis



Power

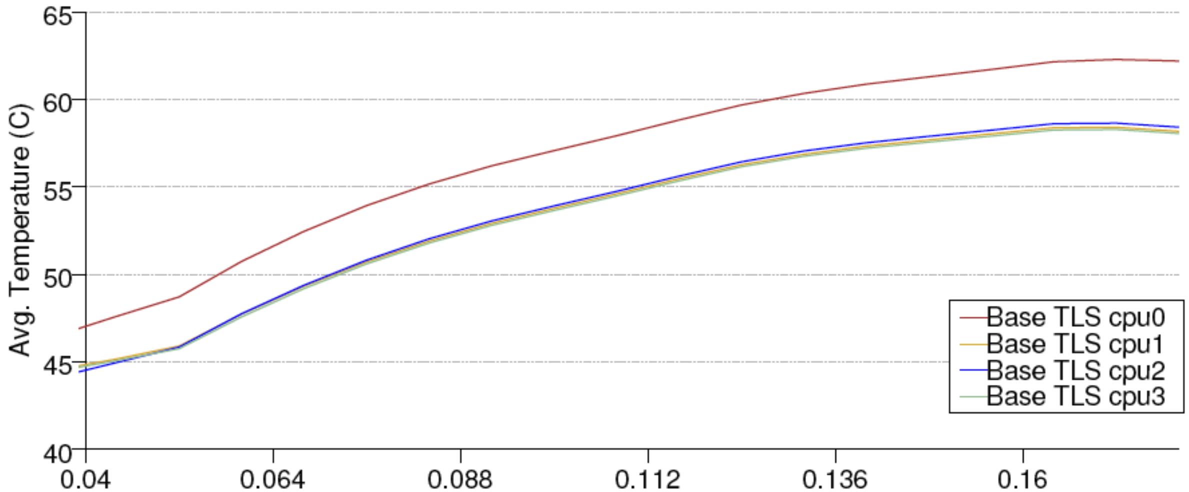
Performance-Power Analysis



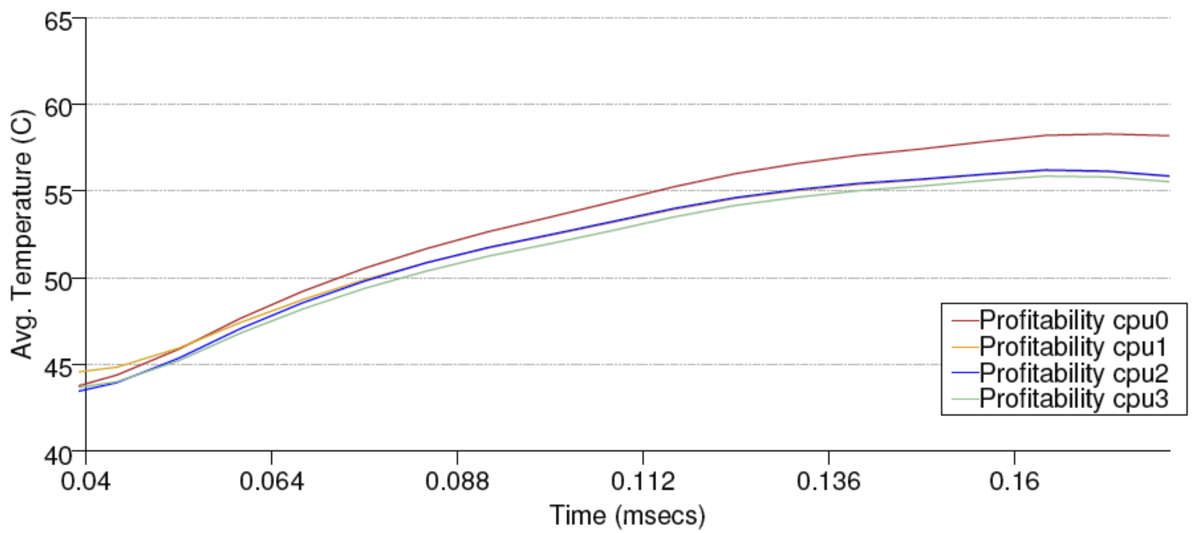
Energy Delay

Thermal Analysis

Base TLS



Profitability-based Scheme



Outline

- Introduction
- Profitability Based Power Allocation
- Estimating Profitability
- Experimental Setup and Results
- Conclusions

Conclusions

- CMPs are here to stay
- Power on chip needs to be effectively utilized
- Allocating power by profitability leads to improvements
- Squash and memory boundedness predictors can estimate thread profitability
- Our approach outperforms state-of-the-art TLS systems:
  – ED by 21.2% (up to 39.6%)
  – … while also reducing the temperature

Profitability-Based Power Allocation for Speculative Multithreaded Systems

Polychronis Xekalakis, Nikolas Ioannou
Salman Khan and Marcelo Cintra
University of Edinburgh