SPECTR: Formal Supervisory Control and Coordination for Many-core Systems Resource Management

Amir M. Rahmani†‡, Bryan Donyanavard†, Tiago Mück†, Kasra Moazzemi†, Axel Jantsch‡, Onur Mutlu§, Nikil Dutt†

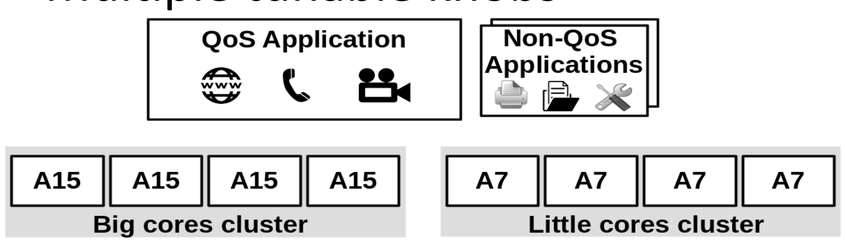
†University of California, Irvine, USA ‡TU Wien, Austria §ETH Zurich, Switzerland

{amirr1,bdonyana,tmuck,moazzemi,dutt}@uci.edu, axel.jantsch@tuwien.ac.at, onur.mutlu@inf.ethz.ch

Problem

Challenges in resource management of many-cores systems

- Several conflicting goals/constraints
- Heterogeneous resources
- Multiple tunable knobs



Current resource management solutions use *Ad hoc heuristics*

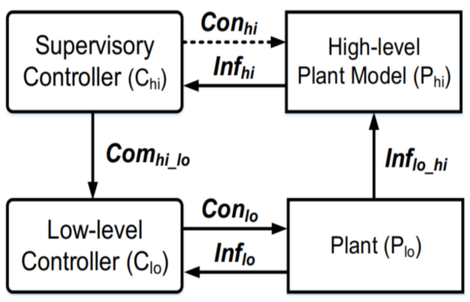
- Can be sub-optimal
- Hard to add learning
- No guarantees
- No formal methodology
- Lack of flexibility

MIMO Control Theory

- ✓ Formal methodology
- ✓ Provides guarantees
- ✗ Fixed policies/goals
- ✗ Not scalable
- ✗ Supports only continuous systems dynamics

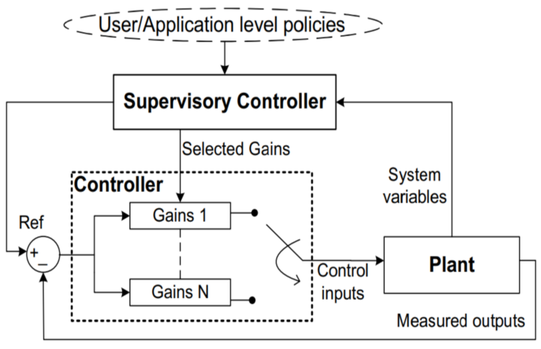
Solution

Scalability via Supervisory control



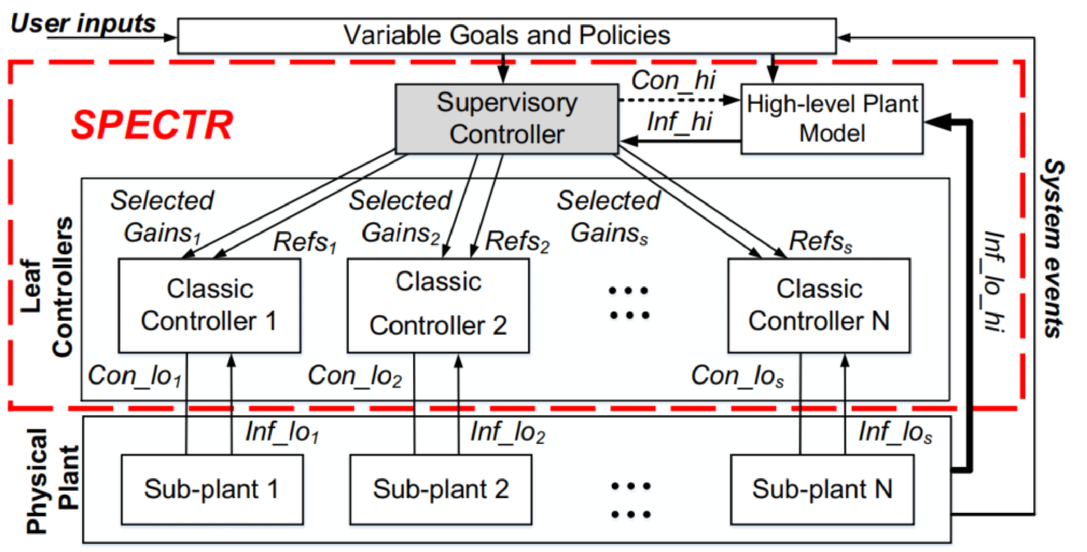
- Models discrete system dynamics
- Hierarchical system decomposition
- Automatic synthesis and verification
- High-level policy management

Autonomy via Supervisory control



- Allows gain scheduling and dynamic references
- low computational complexity
- Integrate logic with continuous dynamics

SPECTR



- ✓ Formal methodology
- ✓ Provides guarantees
- ✓ Adds Scalability
- ✓ Dynamically adapts policies
- ✓ Combines discrete and continuous logic

Results

System goals

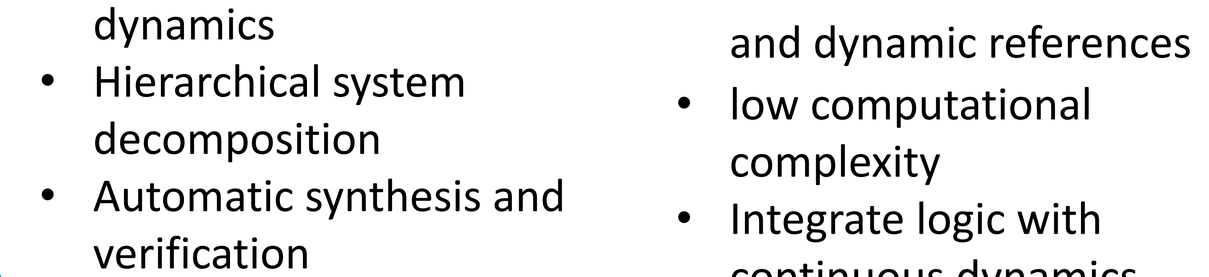
- Meet the QoS requirement of foreground app
- Stay below Thermal Design Power (TDP) and save energy
- QoS task: x264

3-phase execution scenario:

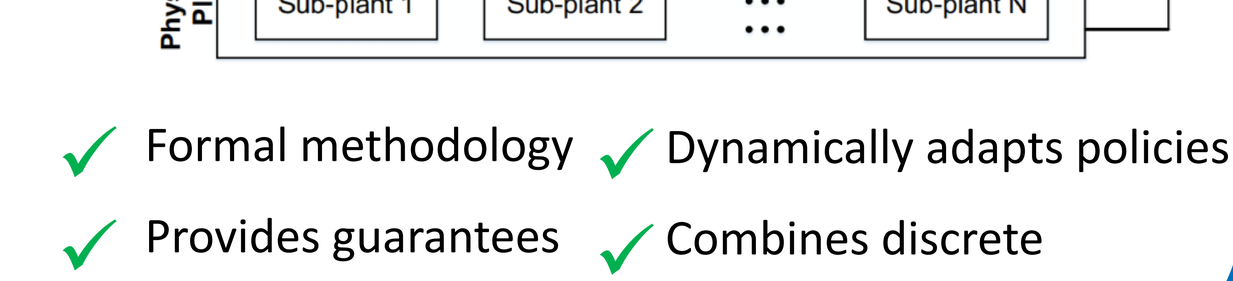
1) Only foreground app runs
2) Thermal emergency (max power decreases)
3) Background apps triggered

MIMO Power

Full System Controller



MIMO Performance



SPECTR

UCIrvine · TU Wien · Systems@ETH Zürich · ETH Zürich · acm · ASPLOS2018