Energy Proportionality and Workload Consolidation for Latency-critical Applications

George Prekas, Mia Primorac, Adam Belay, Christos Kozyrakis, Edouard Bugnion

Current Approaches for Low Latency
• Dedicated servers in polling mode
• Low utilization due to diurnal patterns
• High energy drain at low load

Motivation
• Increase resource efficiency in data centers:
  1) Reduce CPU power consumption under low/medium loads → energy proportionality of latency-critical app
  2) Minimize number of servers → workload consolidation of background job and latency-critical app
• Maintain microsecond-scale tail latency

IX Design
• Dataplane OS for event-driven apps [OSDI ‘14]
• Protection through virtualization (Dune): app, dataplane and control plane isolation
• Efficient execution model
• NEW: control plane (IXCP) dynamically allocates resources
  • Add/remove cores/HT
  • Adjust DVFS

IX execution model

IX architecture

Pareto-Optimal Static Configurations Methodology
• Two dynamic control policies and Pareto baseline derived from the exhaustive analysis of 224 static configurations (configuring 16 HTs and 13+1 DVFS levels)

Mechanisms for Dynamic Configuration Management in IX dataplane OS
• Detection of load changes
  • Queue backlog
  • Detection in sub-second timescales
• Adjusting current configuration
  • Atomic RSS flow group migration w/o losing or reordering packets
    → Completes in less than 2 ms 95% of the time
  • DVFS (Dynamic Voltage and Frequency Scaling) via Linux host

IX Dynamic Resource Control Evaluation on Memcached
• Evaluation of IX control loop under multiple load patterns for memcached key-value store
  (1) Energy proportionality and (2) workload consolidation experiments
• Memcached energy proportionality policy:
  1. Add core by core
  2. Enable HTs on all cores
  3. Gradually increase the clock rate
1) Saving 44%-54% of CPU energy
   (85%-93% of Pareto bound)
2) Running bg app at 32%-46% of peak
   (82%-92% of Pareto bound)
• Adequate SLO compliance