An Approach for Energy Efficient Execution of Hybrid Parallel Programs

Lavanya Ramapantulu, Dumitrel Loghin, Yong Meng Teo
Department of Computer Science
National University of Singapore

4 June 2015

Outline

- Motivation
- Objective
- Related Work
- Approach
- Analysis
- Conclusions

Motivation

- Processors are over-provisioned for floating point operations
- Ratio of floating point speed to memory bandwidth increasing at 15-33 % per year*
  - Need for optimizing data movement
- **Hybrid programming** model has dual advantage of
  - Low-latency intra-node data movement
  - Scaling with inter-node communication
    - Overcomes Dennard scaling limitation

* ICPP 2014 Keynote by Jack Dongarra

Research Questions

1. What is an energy efficient system configuration to execute a hybrid program ?
2. How much time does a hybrid program spend on computation versus communication and other overheads ?
3. …

Objective

- To develop an approach that predicts the execution time and energy usage for a given hybrid parallel program
  - used to determine energy-efficient system configuration
  - energy-efficient configuration executes the program with minimum energy for a given execution time deadline

Related Work

Performance Analysis Approaches				
	Profiling	**Simulation**	**Statistical prediction**	**Analytical Modeling**
Accuracy	✓	✓	✓	✗
Non-intrusiveness	✗	✓	✗	✓
Generalization	✗	✗	✗	✓
Related work	Mao et al. EuroMPI ’14, Morris et al. ICPP’10, Chung et al. IPDPS’09	Janssen et al. SIGMETRICS’11 , Binkert et al. MICRO’06, Austin et al. TC’02	Vetter et al. PPoPP’01, Lee et al. ASPLOS’06, Barnes et al. ICS’08	Hill & Marty TC’08, Woo et al TC’08, Cho et al. TPDS’10

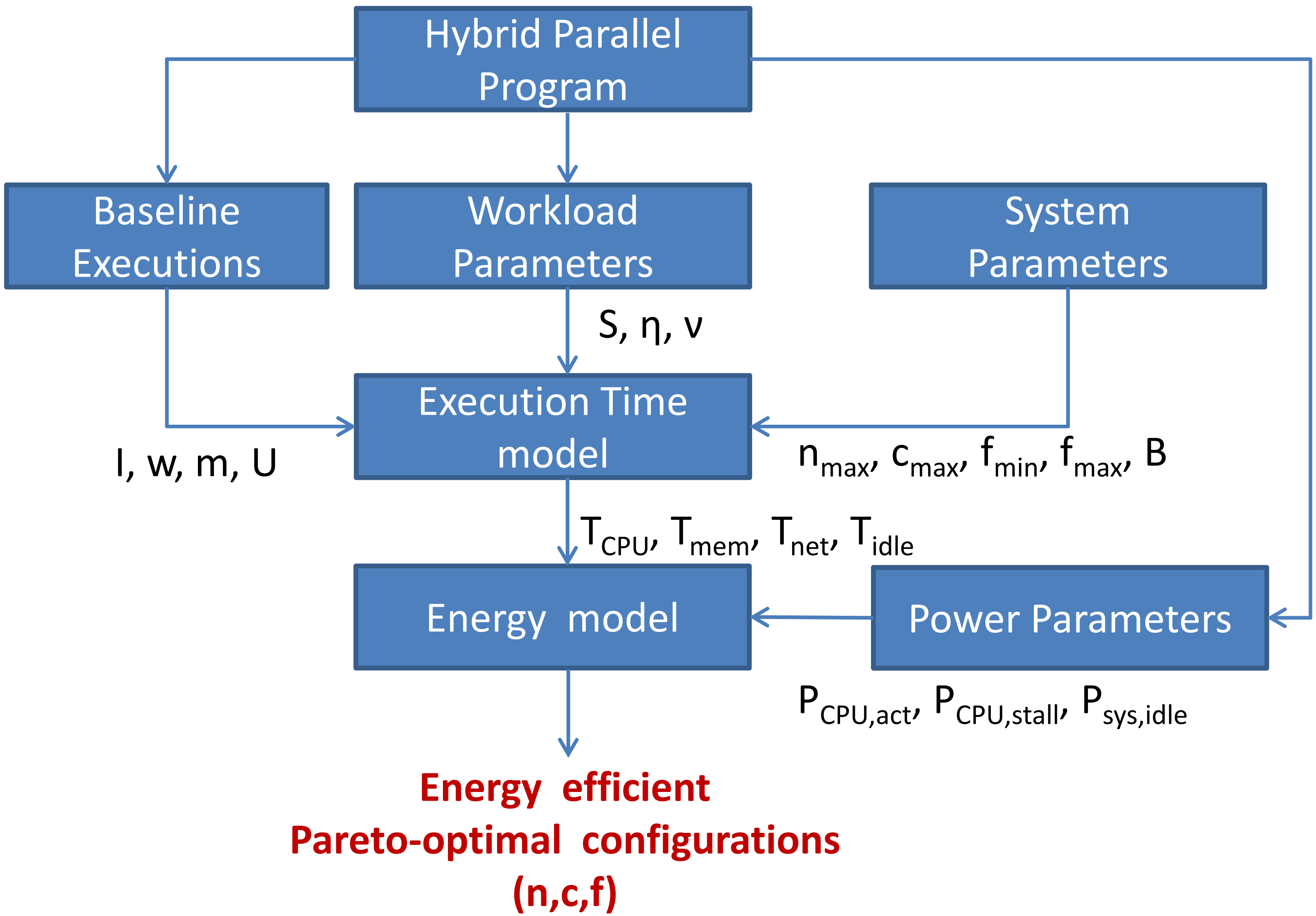
Related Work

Performance Analysis Approaches					
	Profiling	Simulation	Statistical prediction	Analytical Modeling	Measurement based-analytical model
Accuracy	✓	✓	✗	✗	✓
Non-intrusiveness	✗	✓	✗	✓	✓
Generalization	✗	✗	✗	✓	✓
Related work	Mao et al. EuroMPI '14, Morris et al. ICPP'10, Chung et al. IPDPS'09	Janssen et al. SIGMETRICS'11, Binkert et al. MICRO'06, Austin et al. TC'02	Vetter et al. PPoPP'01, Lee et al. ASPLOS'06, Barnes et al. ICS'08	Hill & Marty TC'08, Woo et al. TC'08, Cho et al. TPDS'10	Tudor et al. SIGMETRICS'13 Ramapantulu et al. ICPP'14 **This work**

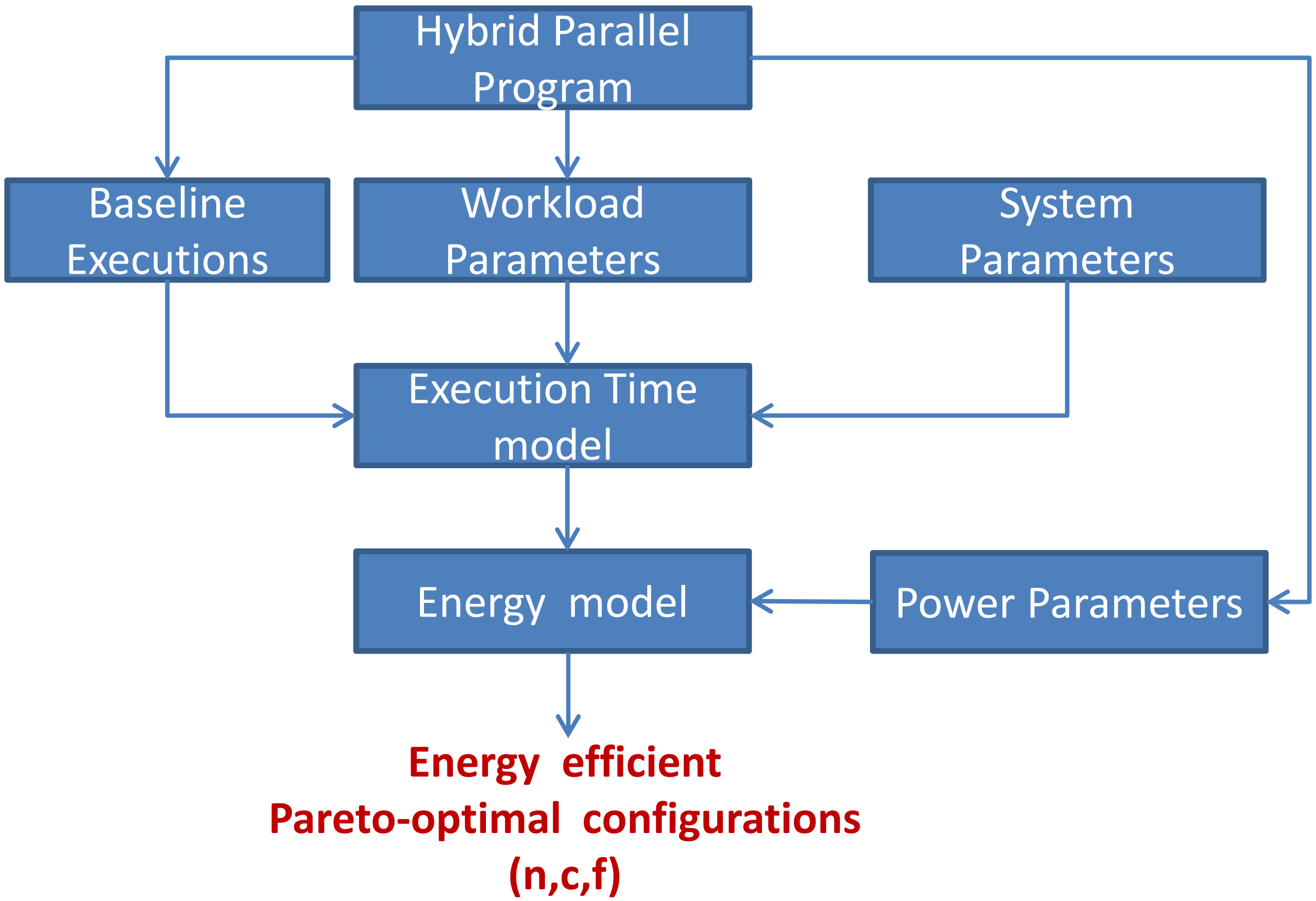
Outline

- Motivation
- Objective
- Related Work
- Approach
- Analysis
- Conclusions

Approach Overview

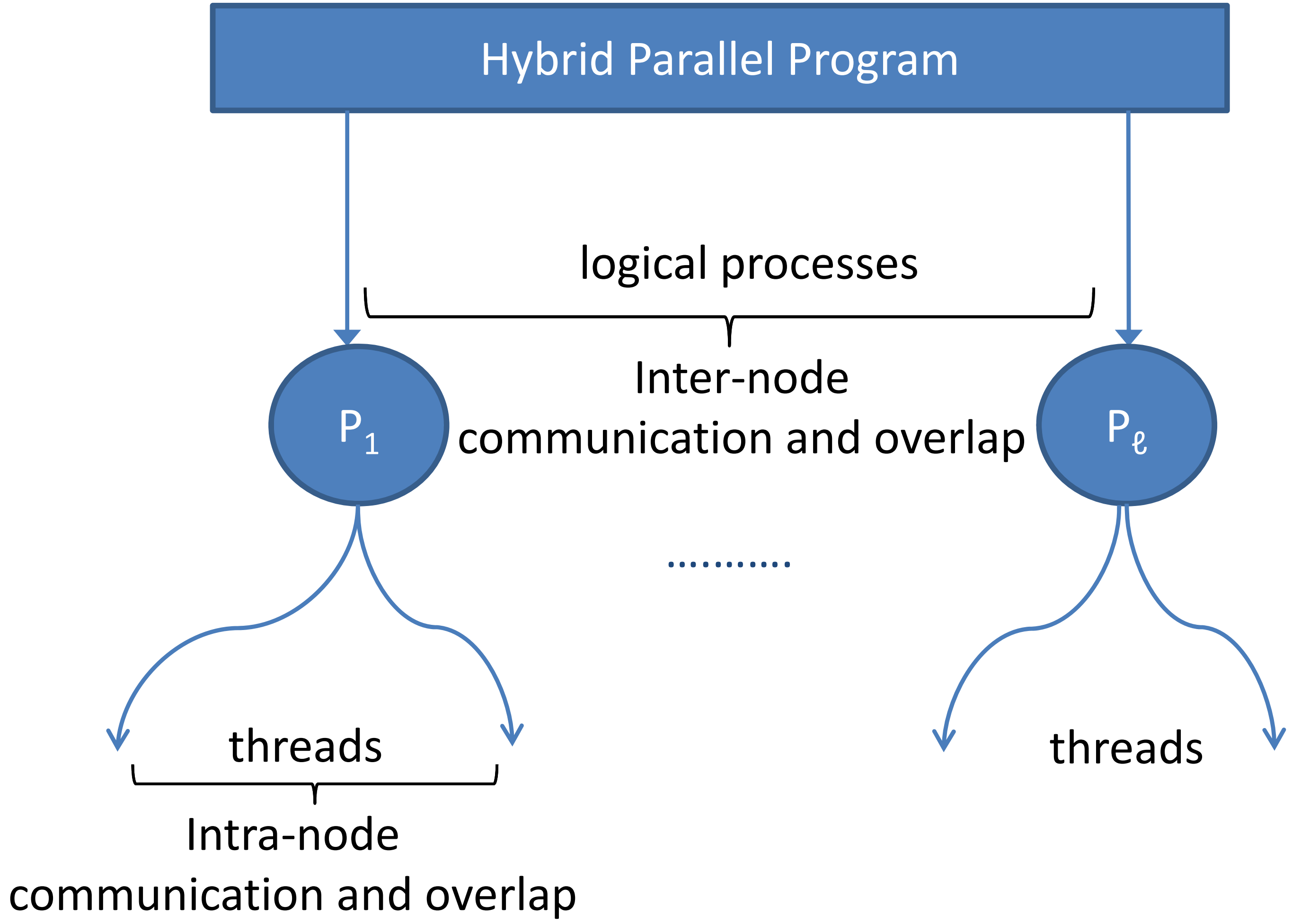


Approach Overview

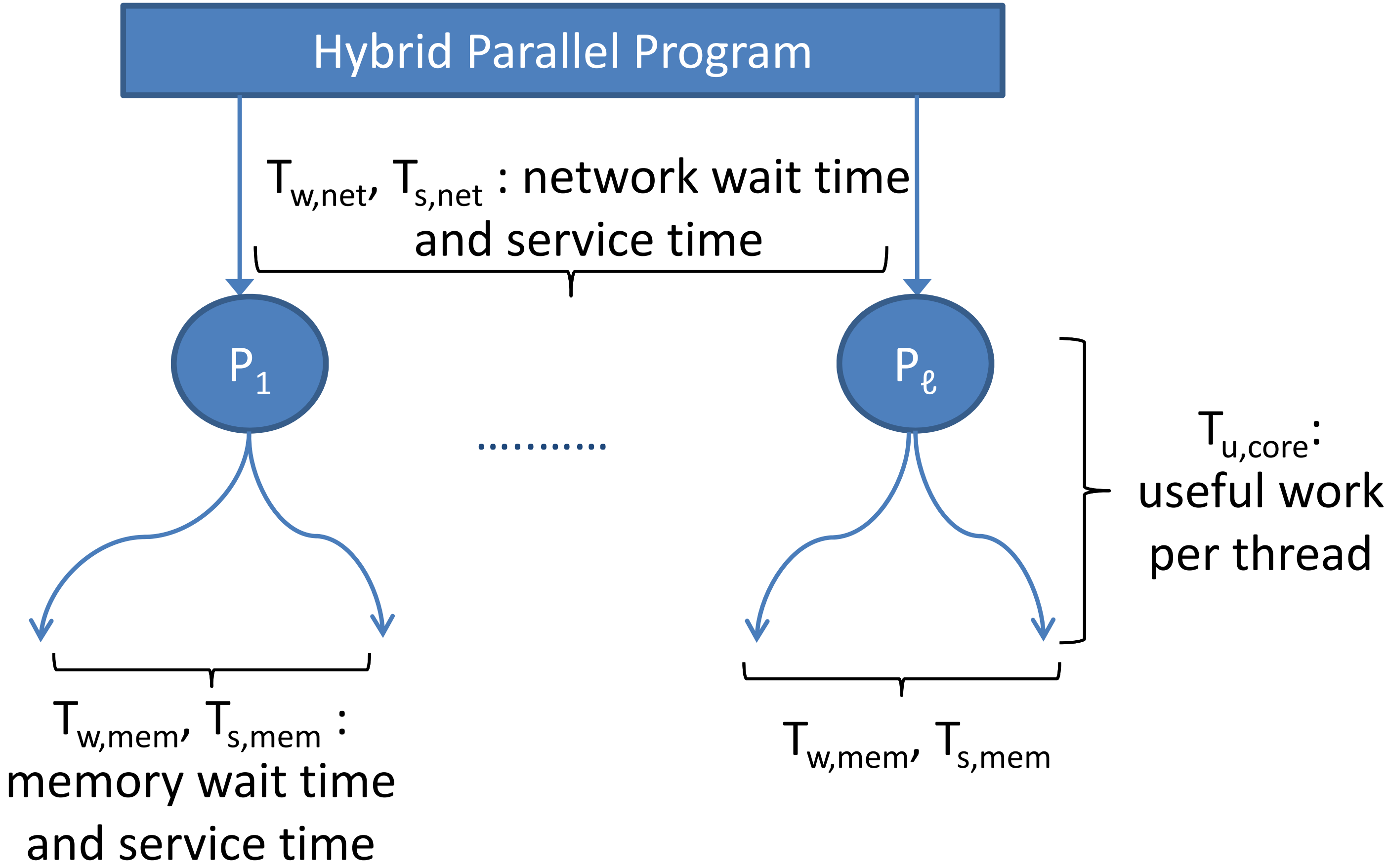


- Inter- and intra-node resource overlaps
- Inter- and intra node resource contention

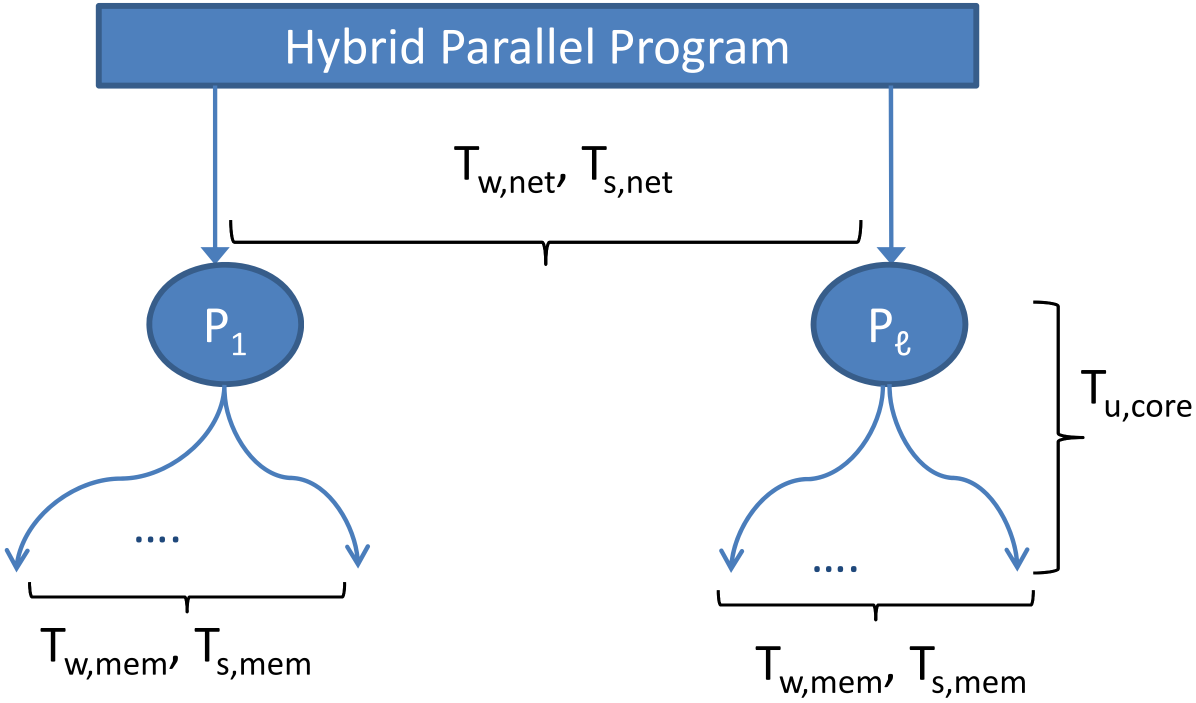
Hybrid Program



Execution Time Model



Execution Time Model



$$T_{total} = \underbrace{T_{u,core}}_{\text{overlapped computation and communication}} + \underbrace{T_{w,mem} + T_{s,mem} + T_{w,net} + T_{s,net}}_{\text{non-overlapped communication}}$$

Workloads

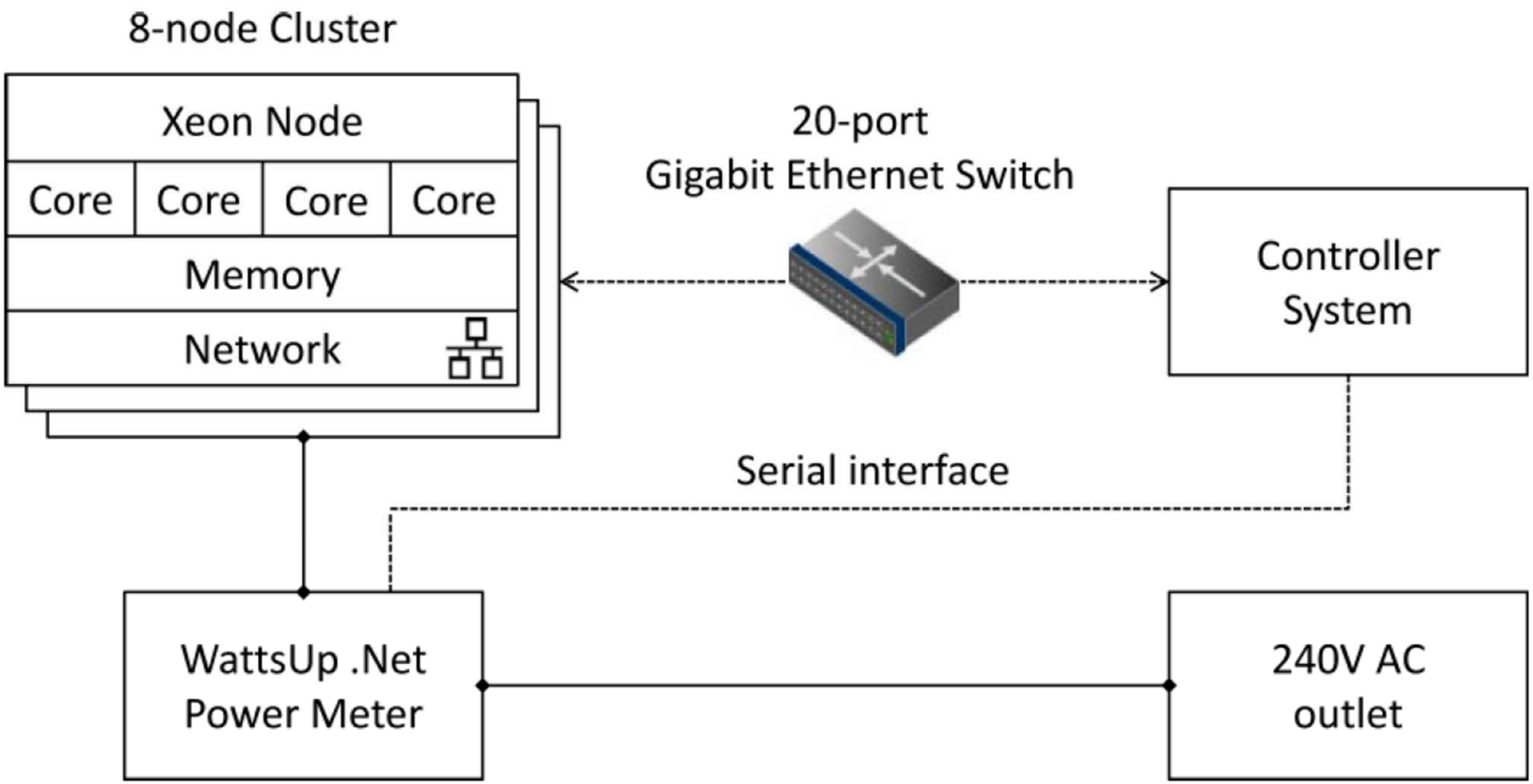
Domain	Benchmark Suite	Program	Problem Size
Non-linear PDE solvers	NAS multi-zone parallel Benchmark (NPB3.3-MZ)	LU, BT, SP	Class B, C
electronic-structure & nano-scale material modeling	Quantum Espresso (v5.1)	CP	32 and 64 molecules of H_2O
Computational fluid dynamics	OpenLB (olb-0.8r0)	cavity3d cuboid	20 and 40 time-step simulations

NAS: NASA Advanced Supercomputing

Systems

System	Intel Xeon E5-2603	ARM Cortex-A9
ISA	x86_64	ARMv7-A
Nodes	8	8
Cores/node	8	4
Clock Frequency	1.2–1.8 GHz	0.2–1.4 GHz
L1 data cache	32kB / core	32kB / core
L2 cache	2MB / node	1MB / node
L3 cache	20MB / node	NA
Memory	8GB DDR3	1GB LP-DDR2
I/O bandwidth	1Gbps	100Mbps

Validation Setup



Cluster Validation Results

Program	Execution Time error[%]				Energy error[%]			
	Xeon		Cortex-A9		Xeon		Cortex-A9	
	Mean	Std. Dev.	Mean	Std. Dev.	Mean	Std. Dev.	Mean	Std. Dev.
LU	4	5	3	2	5	8	6	6
SP	6	9	4	3	2	10	4	5
BT	8	7	4	6	8	7	5	6
CP	1	10	5	12	1	14	7	12
LB	6	8	4	8	15	12	7	9

Outline

- Motivation
- Objective
- Related Work
- Approach
- Analysis
- Conclusions

Analysis

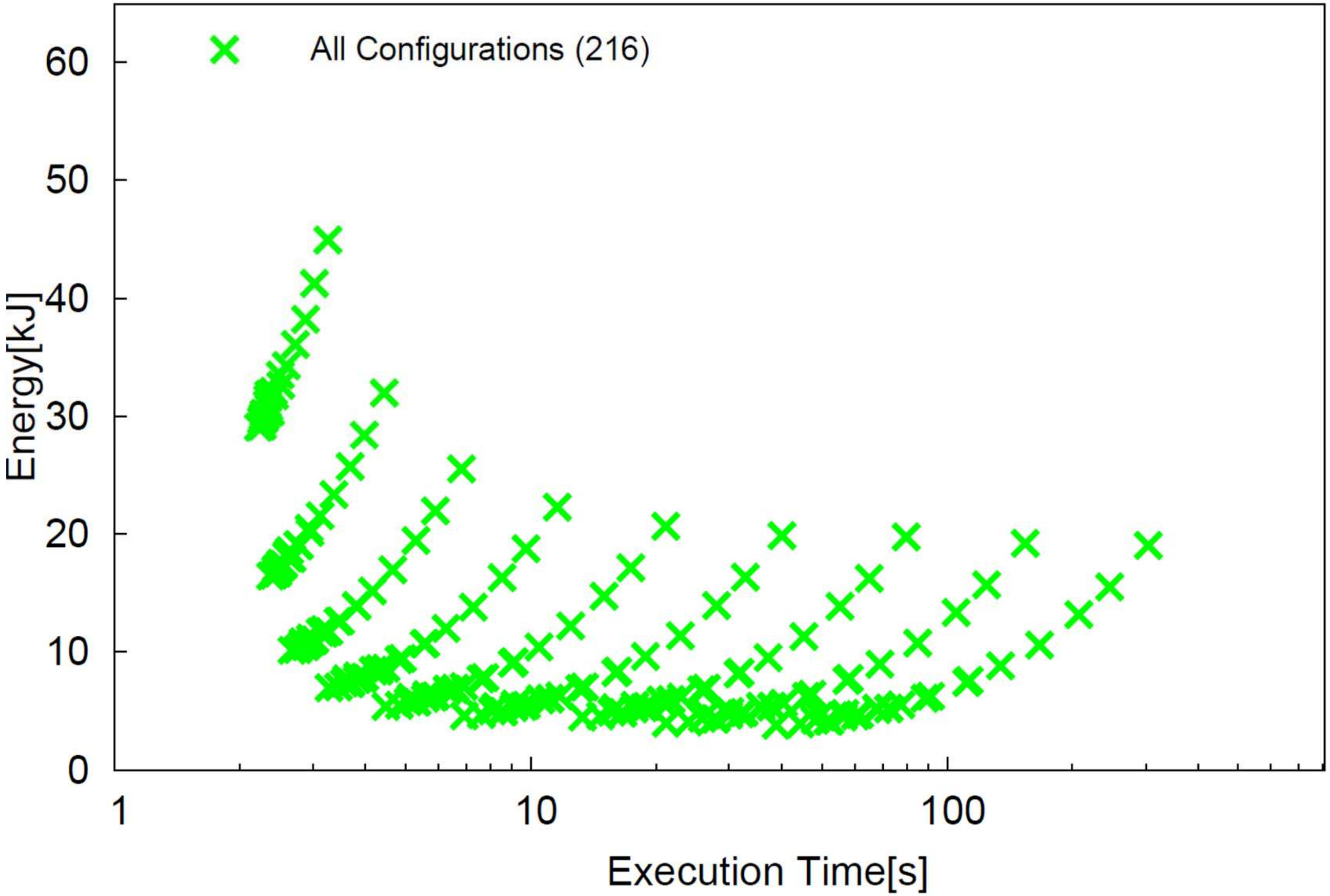
- Pareto-optimal configurations

- Multi-node versus multi-core

- Improving Pareto-optimal configurations

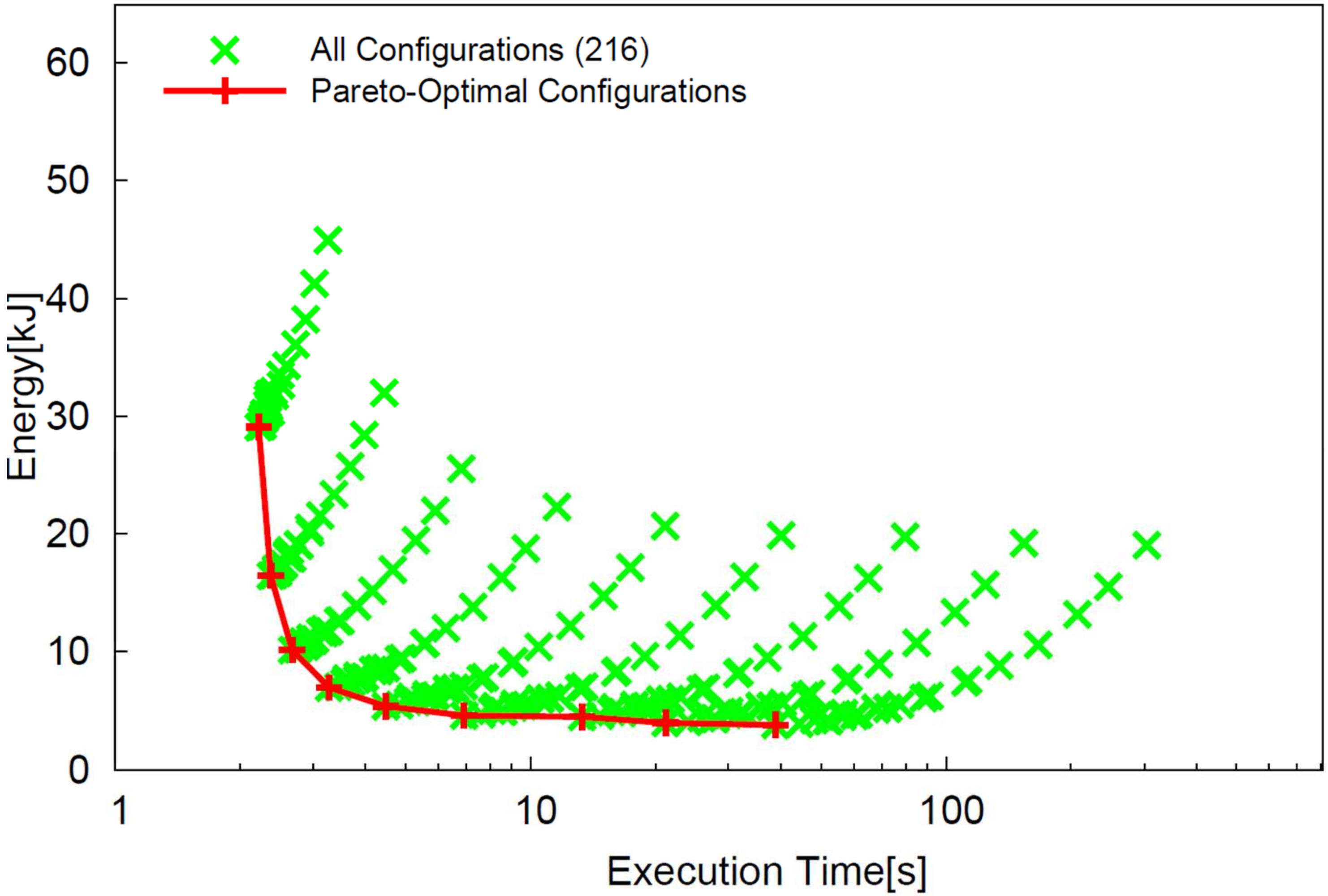
Energy-efficient Configurations

Xeon cluster executing SP program



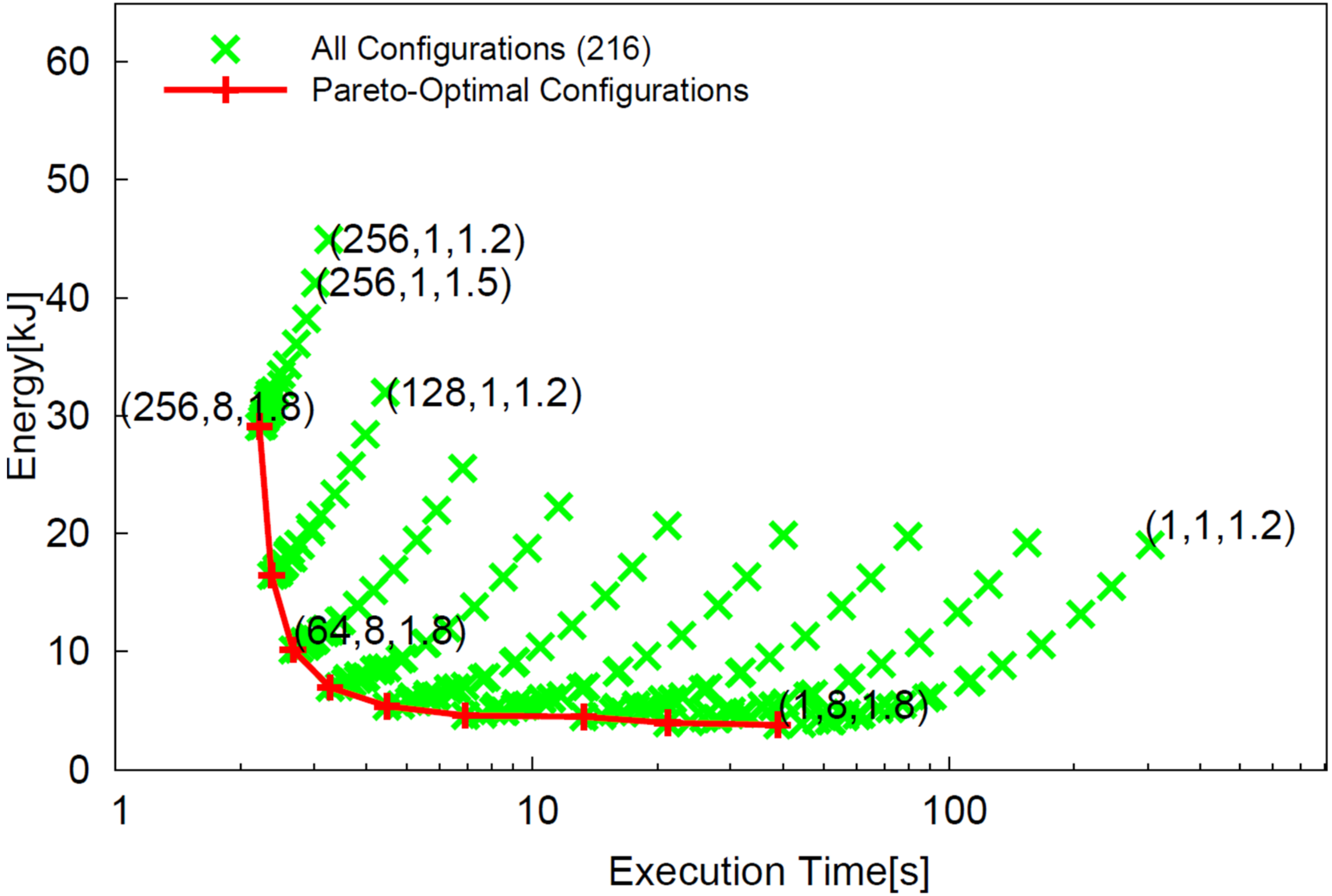
Energy-efficient Configurations

Xeon cluster executing SP program



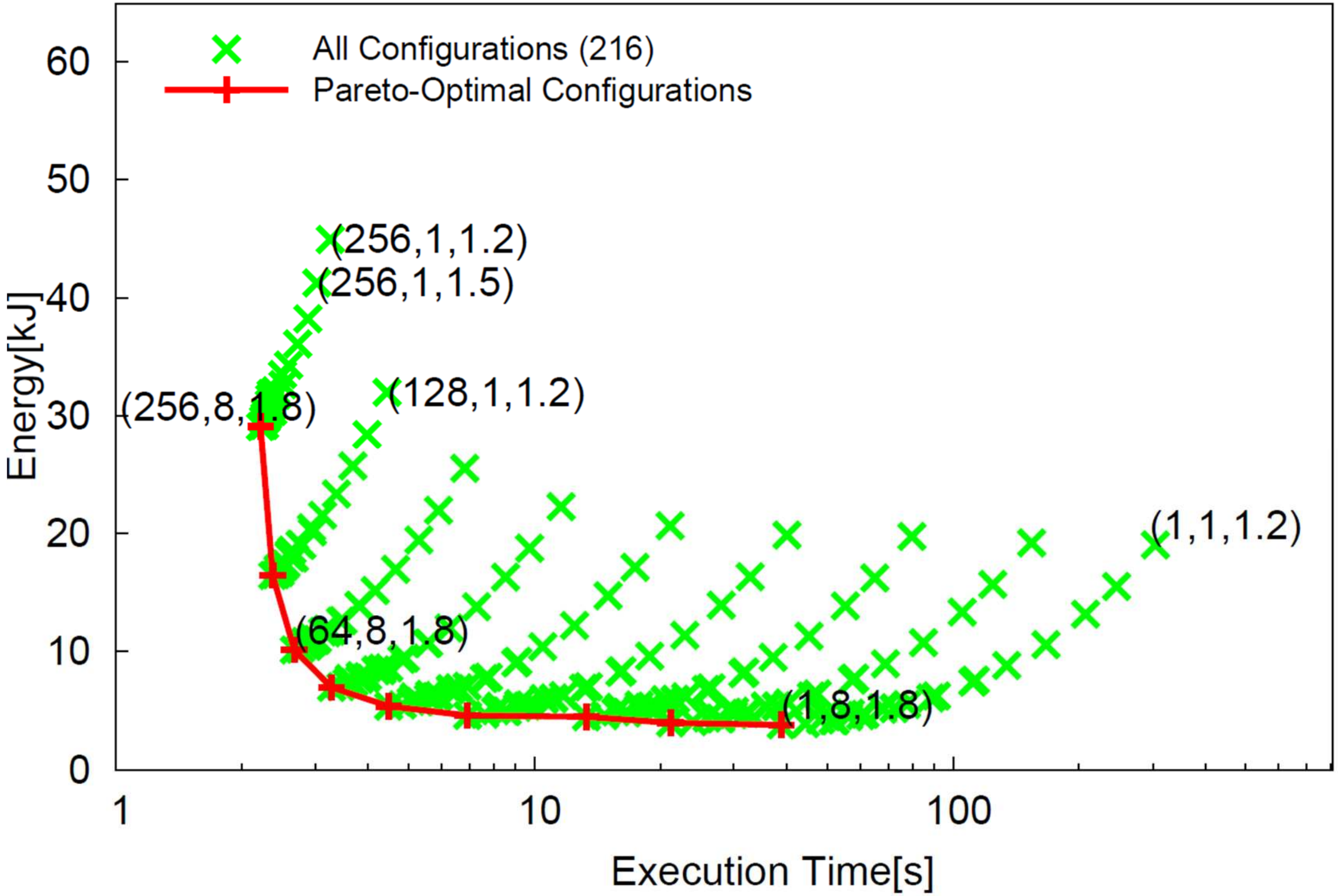
Energy-efficient Configurations

Xeon cluster executing SP program



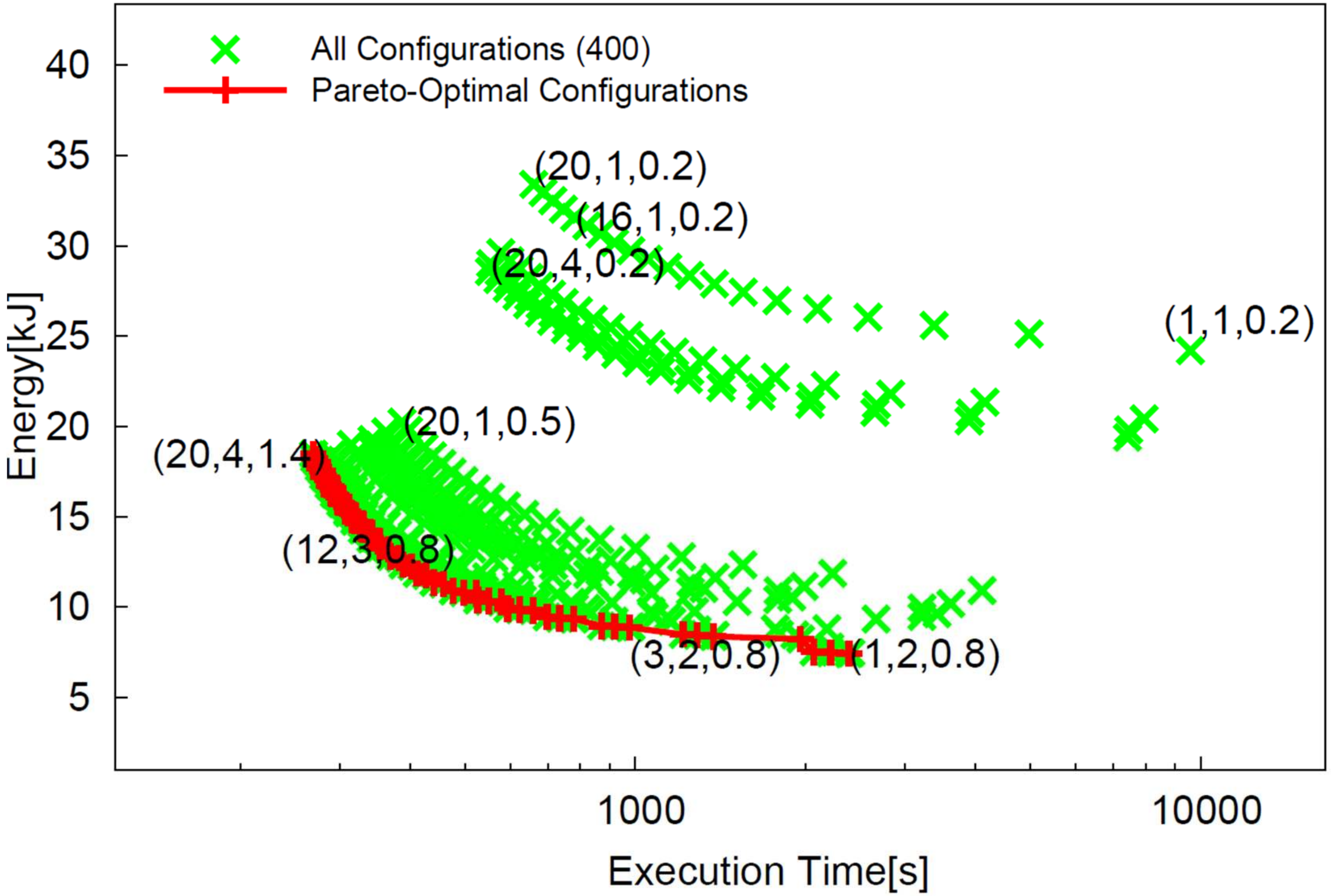
Is multi-node more energy-efficient than multi-core ?

Xeon cluster executing SP program



Is multi-node more energy-efficient than multi-core ?

ARM cluster executing CP program



Is multi-node more energy-efficient than multi-core ?

- One size does not fit all
- depends on workload characteristics
  - computation demands versus communication
- depends on system characteristics
  - power characteristics of the node across cores and clock frequencies

Useful Computation Ratio (UCR)

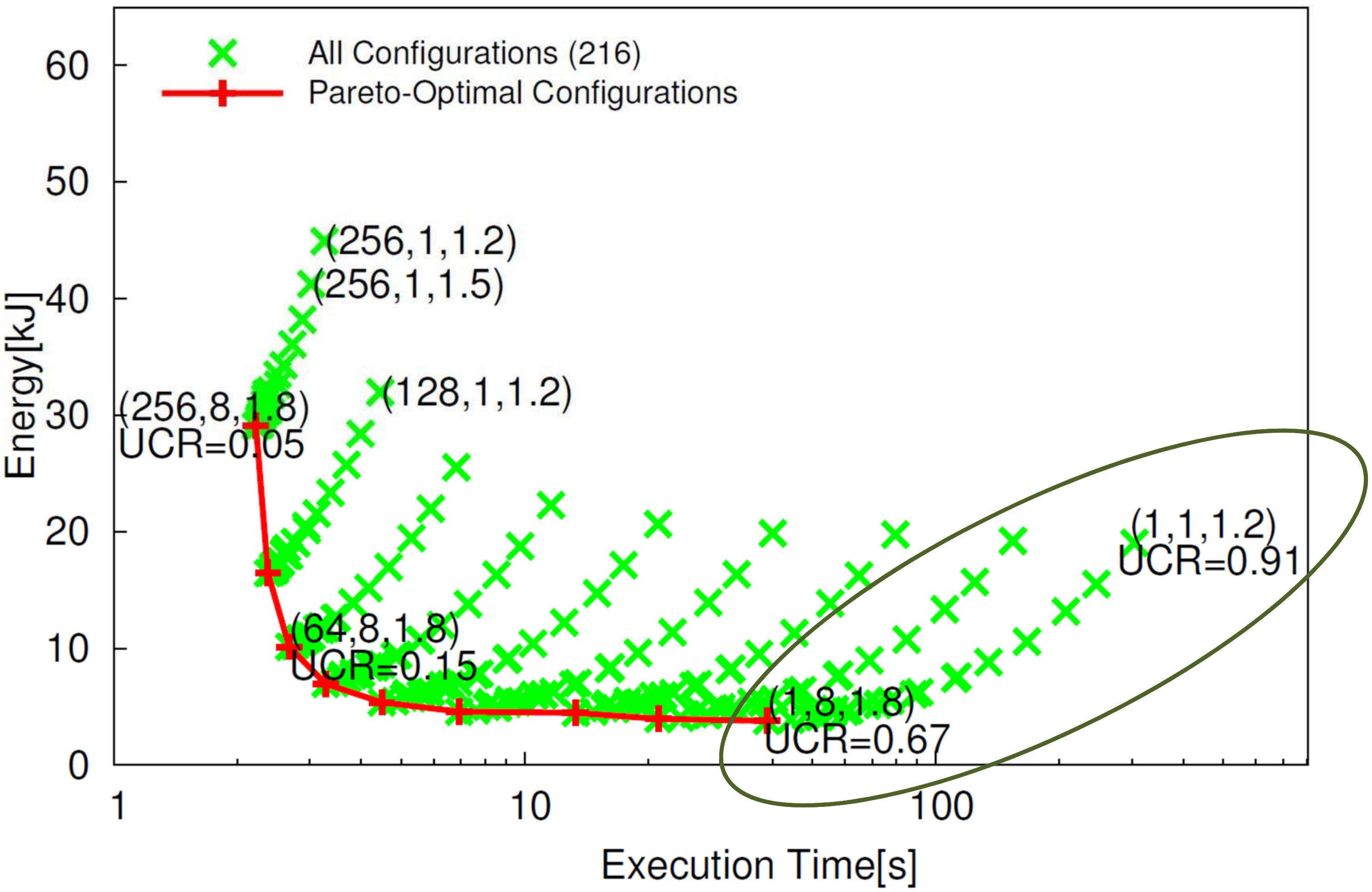
- Quantifies the performance of a program with respect to the amount of useful computations performed
  - normalized metric for ease of comparison among configurations

$$UCR = \frac{T_{useful}(= T_{CPU})}{T}$$

$$T = T_{CPU} + T_{data\_dep} + T_{mem\_contention} + T_{net\_contention}$$

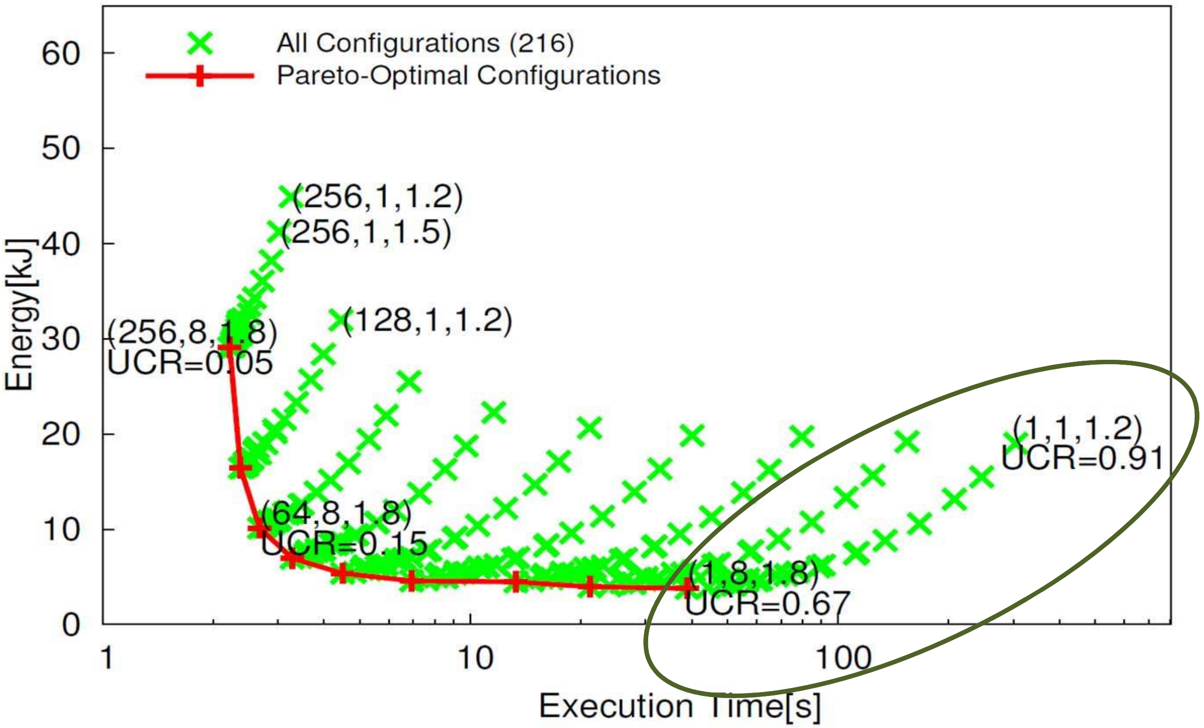
Useful Computation Ratio (UCR)

Xeon cluster executing SP program



Improving Pareto-Optimal Configurations

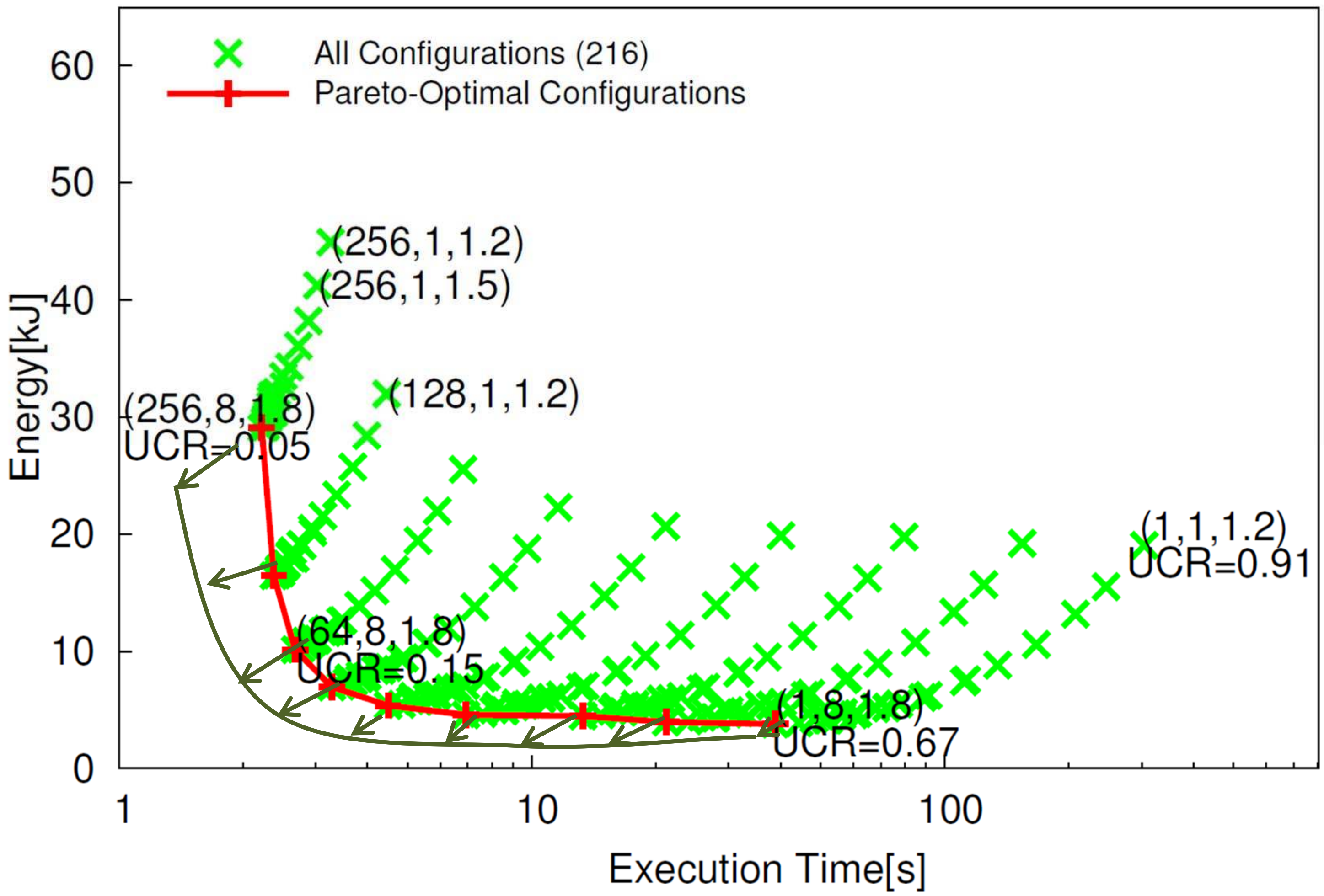
Xeon cluster executing SP program



- Doubling the memory-bandwidth Increases UCR from 0.67 to 0.81
  - reduces execution time by 7 seconds
  - reduces energy by 590 Joules

Improving Pareto-Optimal Configurations

Xeon cluster executing SP program



Conclusions

- Measurement-driven analytical model to determine time-energy performance of hybrid parallel programs
- Pareto-frontier determines time-energy optimal system configurations for executing hybrid programs
- Pareto-frontier can be further improved by identifying resource imbalances using the Useful Computation Ratio (UCR) metric

Questions ?

Thank you

[lavanya, teoym]@comp.nus.edu.sg

Department of Computer Science

National University of Singapore

L. Ramapantulu, D.Loghin and Y.M. Teo, **An Approach for Energy Efficient Execution of Hybrid Parallel Programs**, Proceedings of 28th International Parallel and Distributed Processing Symposium, Hyderabad, India, May 25-29, 2015.

Thank you

Backup slides

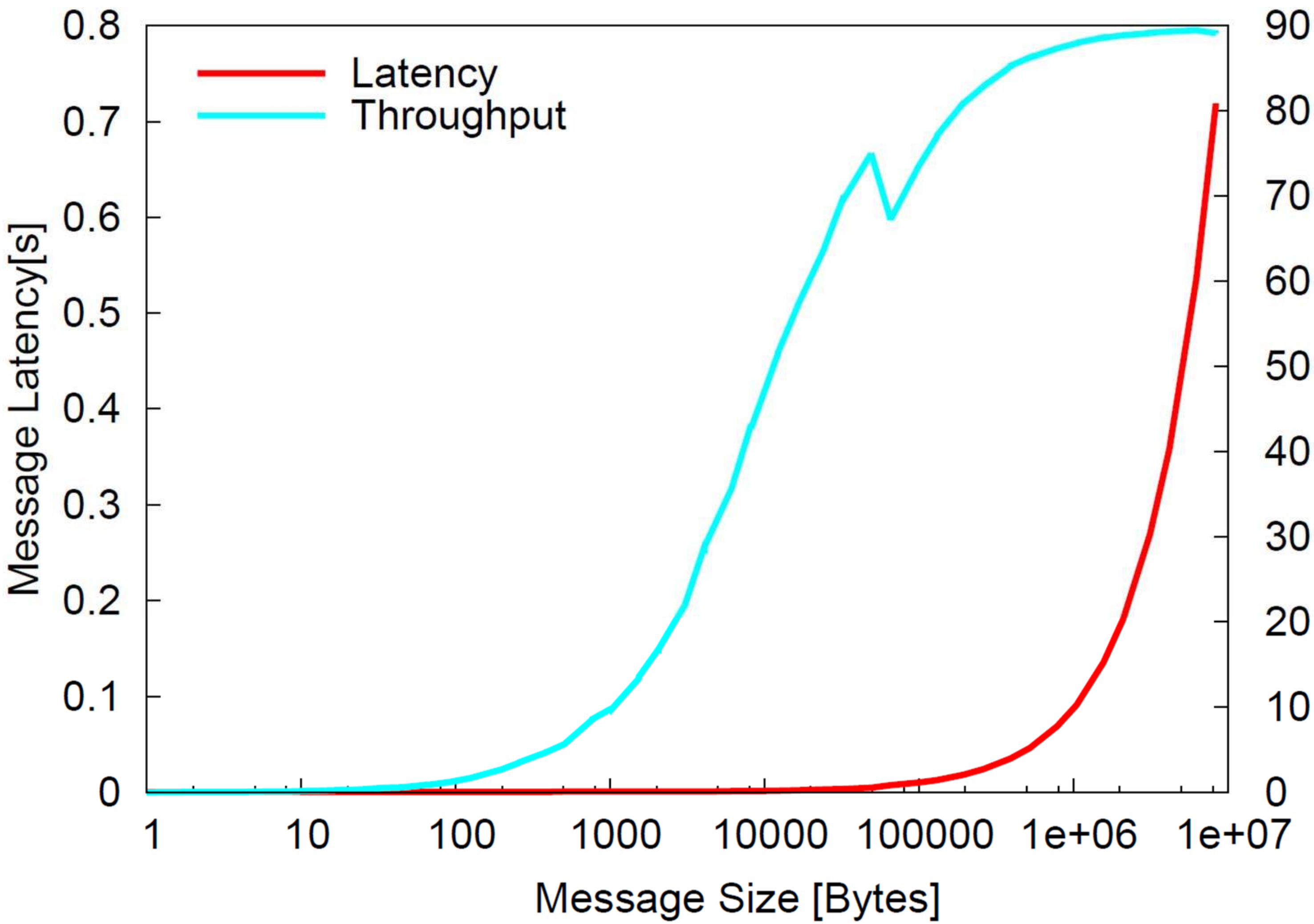
Notation

Workload Parameters			
S	number of iterations in a program	**η**	no. of messages sent/received
		v	Volume (in bytes) per message
System Parameters			
n	number of computing nodes	**B**	network bandwidth
c	number of cores per node	**f**	core clock frequency
Baseline Execution Parameters			
w	useful work cycles	**m**	stall cycles because of memory contention
b	non-memory stall cycles	**U**	core utilization
Execution Time Model Output			
$T_{u,core}$	time to perform useful work	$T_{w,mem}$	waiting time due to memory contention
$T_{s,mem}$	service time due to non-overlapped memory requests	$T_{w,net}$	waiting time because of network contention
$T_{s,net}$	service time due to non-overlapped n/w communication	T_{total}	total execution time of program

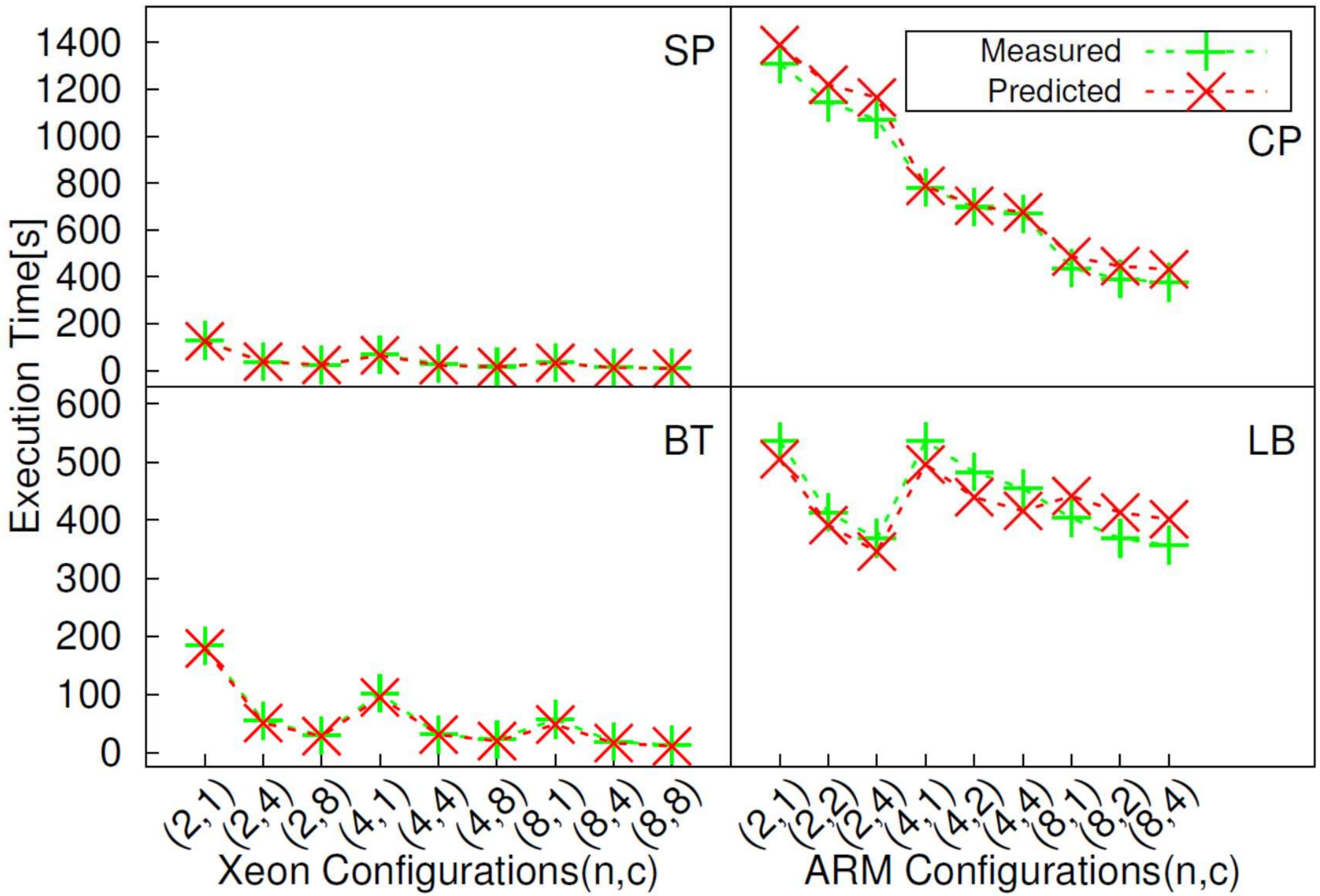
Workload Characterization

- Execute baselines for a single node
  - computational resource demands
  - communication service demands
- Micro-benchmarks
  - Power characterization: core active and core stall
  - MPI characterization: Bandwidth and latency

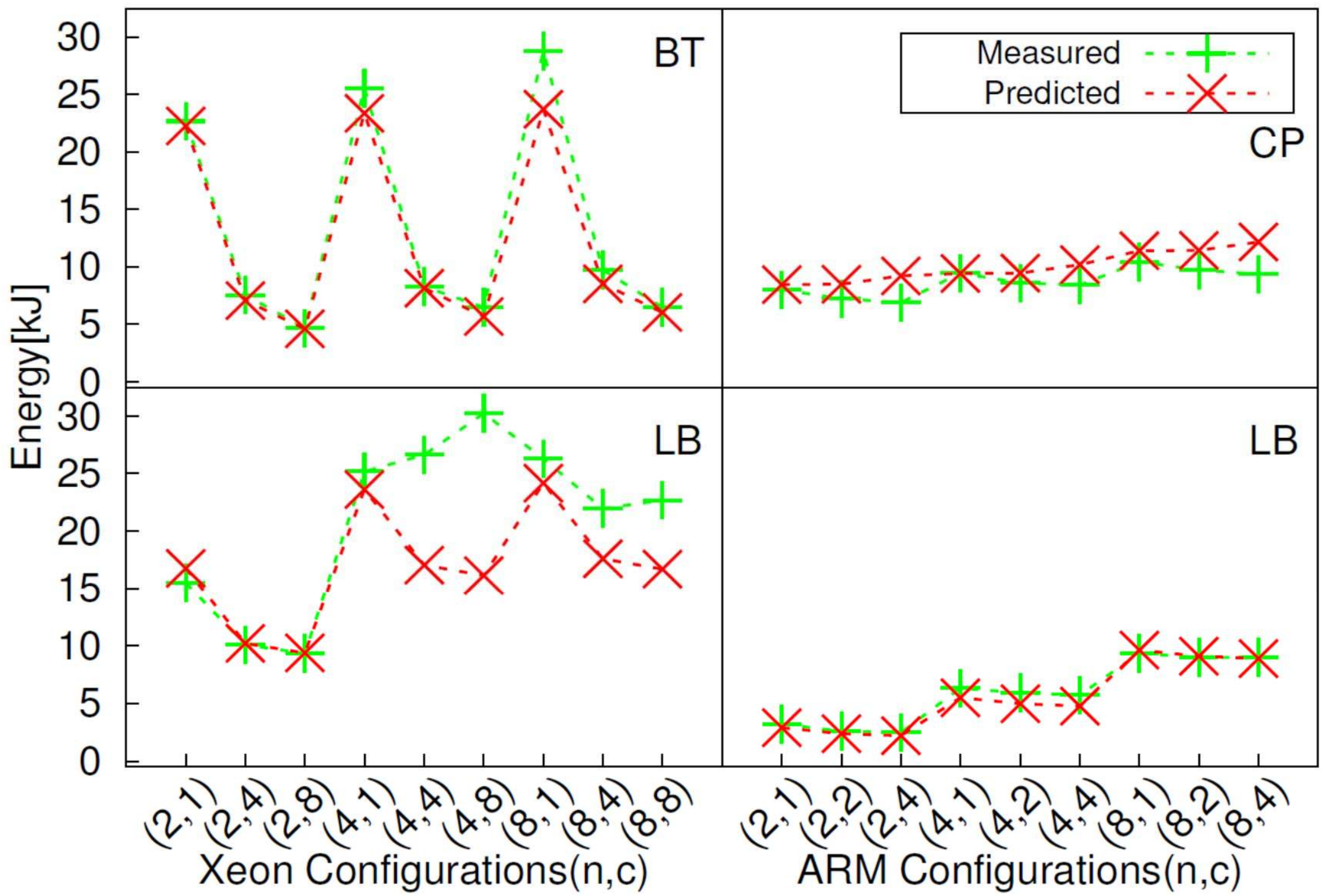
Network Characterization



Execution Time Validation

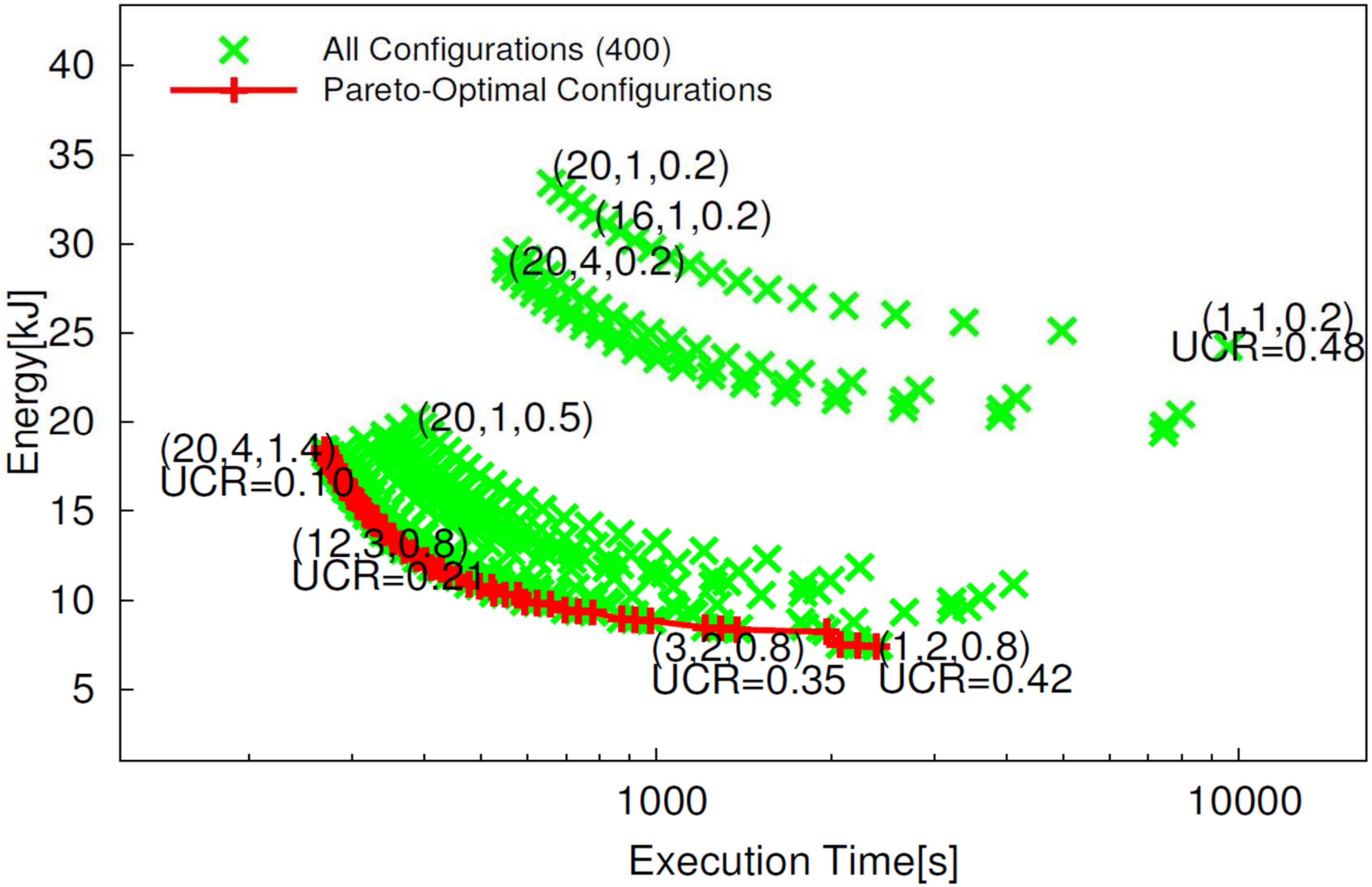


Energy Validation



Useful Computation Ratio (UCR)

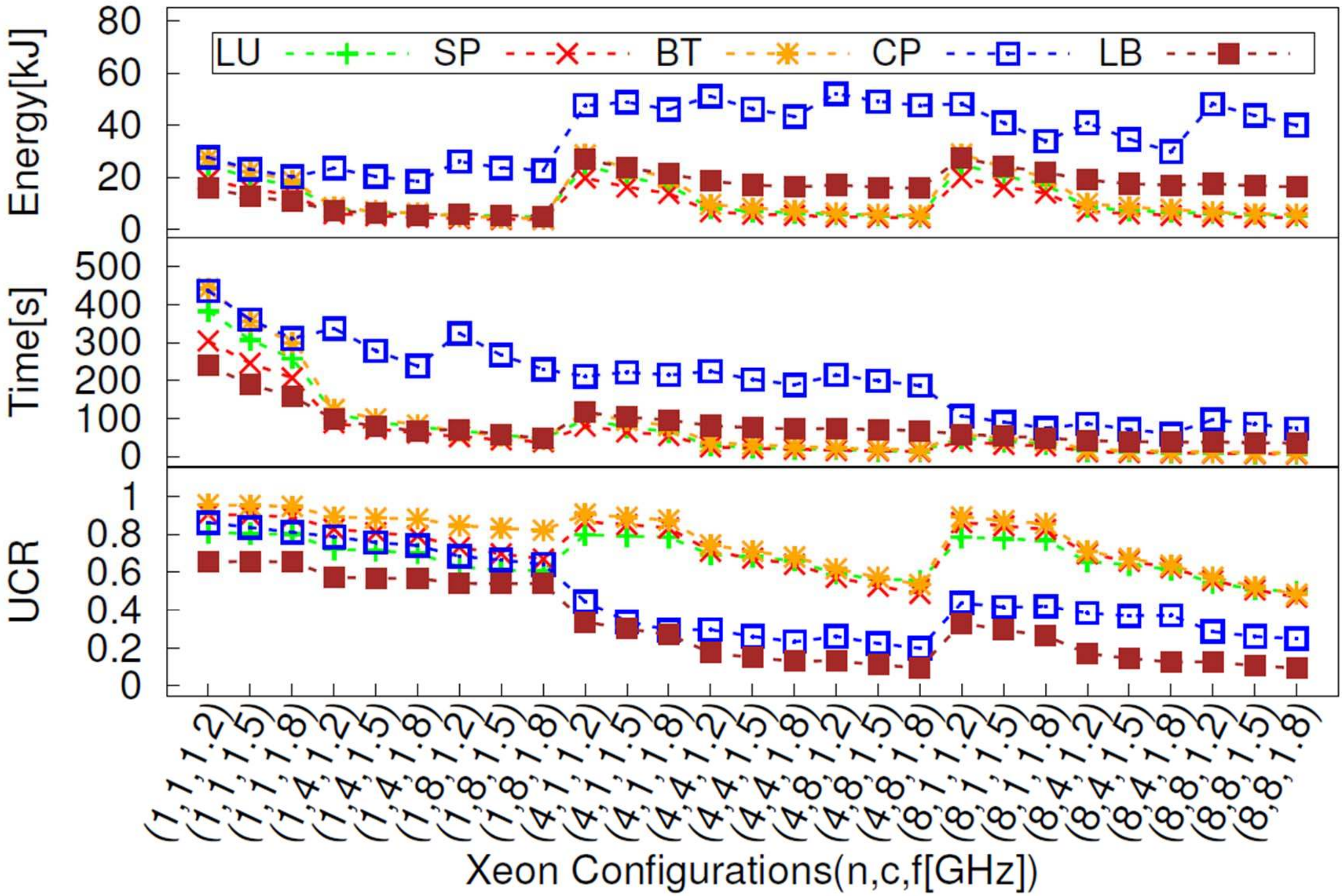
ARM cluster executing CP program



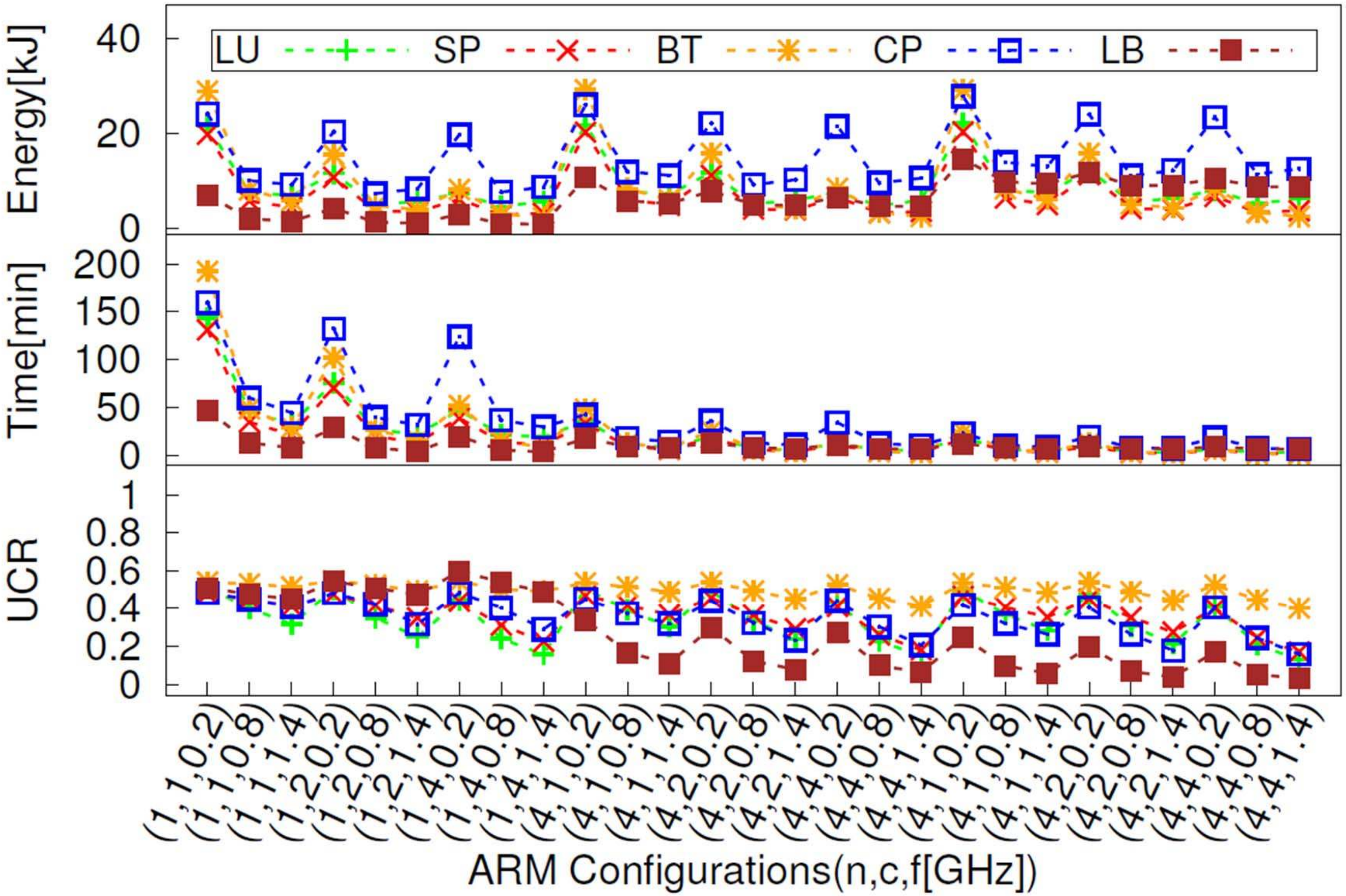
UCR Insights

- UCR enables determining program level imbalances
  - logical processes versus parallel threads
- UCR enables determining system resource mismatches
  - computation versus communication
- However, UCR cannot determine energy efficiency

Time-energy-UCR (Xeon)



Time-energy-UCR (ARM)



References

- G. Mao, D. B¨ohme, M.-A. Hermanns, M. Geimer, D. Lorenz, F. Wolf, Catching Idlers with Ease: A Lightweight Wait-State Profiler for MPI Programs, *Proc. of the 21th European MPI Users' Group Meeting*, 2014
- A. Morris, A. Malony, S. Shende, K. Huck, Design and Implementation of a Hybrid Parallel Performance Measurement System, *Proc. of 39th ICPP*, pages 492–501, Sept 2010.
- I.-H. Chung, S. R. Seelam, B. Mohr, J. Labarta, Tools for Scalable Performance Analysis on Petascale Systems, *Proc. of IPDPS*, 2009
- C. L. Janssen, H. Adalsteinsson, J. P. Kenny, Using Simulation to Design Extremescale Applications and Architectures: Programming Model Exploration, *SIGMETRICS Perform. Eval. Rev.*, 38(4):4–8, Mar. 2011
- N. Binkert, R. Dreslinski, L. Hsu, K. Lim, A. Saidi, S. Reinhardt, The M5 Simulator: Modeling Networked Systems, *MICRO*, 26(4):52–60,2006
- T. Austin, E. Larson, D. Ernst, SimpleScalar: an Infrastructure for Computer System Modeling, *Computer*, 35(2):59–67, Feb 2002

References

- J. S. Vetter, M. O. McCracken, Statistical Scalability Analysis of Communication Operations in Distributed Applications, *Proc. of 8th PPoPP*, pages 123–132, 2001.
- B. C. Lee, D. M. Brooks, Accurate and Efficient Regression Modeling for Microarchitectural Performance and Power Prediction, *In Proc. Of 12th ASPLOS*, volume 41, pages 185–194, 2006
- B. J. Barnes, B. Rountree, D. K. Lowenthal, J. Reeves, B. de Supinski, M. Schulz, A Regression-based Approach to Scalability Prediction, *Proc. of 22nd ICS*, pages 368–377, 2008.
- M. Hill, M. Marty, Amdahl's Law in the Multicore Era, *Computer*, 41(7):33–38, 2008.
- D. H. Woo, H.-H. S. Lee, Extending Amdahl's Law for Energy-Efficient Computing in the Many-Core Era., *IEEE computer*, 41(12):24–31, 2008.
- S. Cho, R. G. Melhem, On the Interplay of Parallelization, Program Performance, and Energy Consumption, *Trans. on Parallel and Distributed Systems*, 21(3):342–353, 2010.