ILLINOIS INSTITUTE OF TECHNOLOGY

Toward Smart HPC via Intelligent Scheduling

Zhiling Lan
lan@iit.edu
Department Computer Science, Illinois Institute of Technology

Keynote at the ICPP 2020 SRMDPS Workshop

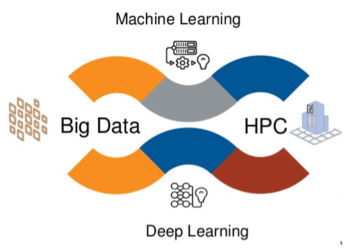
ILLINOIS INSTITUTE OF TECHNOLOGY

Outline

- What is smart HPC?
- What is intelligent (batch) scheduling?
- Two research projects
  - Power aware scheduling
  - Multi-resource scheduling
- Open issues

ILLINOIS INSTITUTE OF TECHNOLOGY

HPC Trends: Workload



Simulation Applications	Big Data Applications	Machine Learning Applications

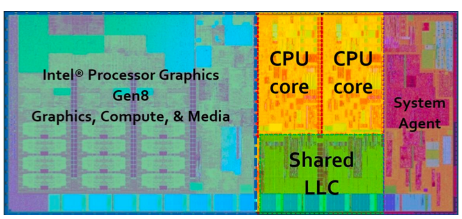
System Software

Large Scale System

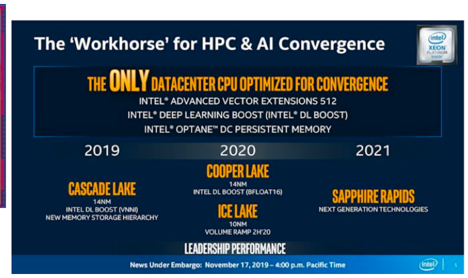
Z. Lan 3

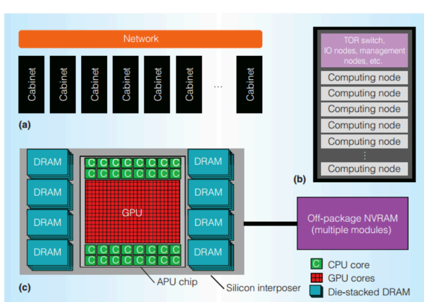
ILLINOIS INSTITUTE OF TECHNOLOGY

HPC Trends: Platform



Intel Heterogeneous Computing Pipelining



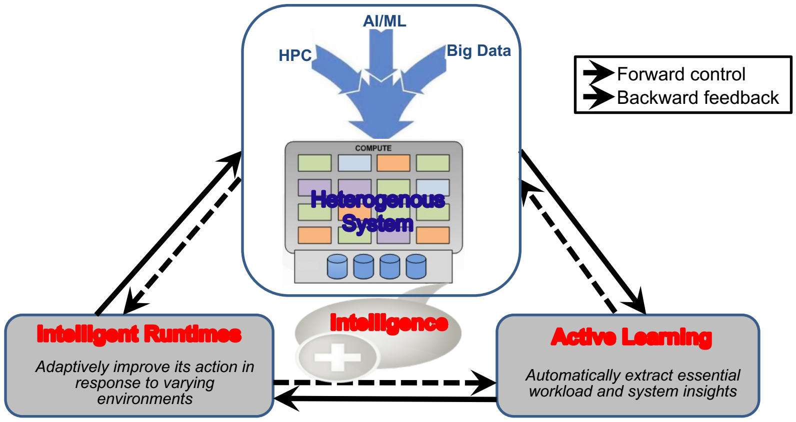


AMD Exascale Strategy Hinges on Heterogeneity

Z. Lan 4

ILLINOIS INSTITUTE OF TECHNOLOGY

Smart HPC



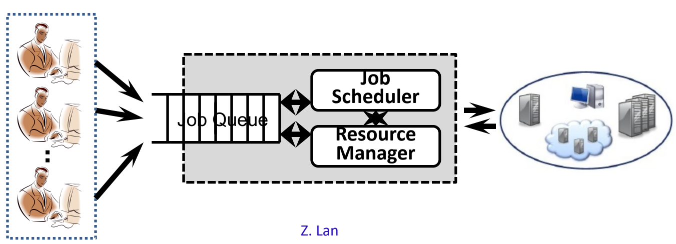
future HPC => smart HPC

Z. Lan 5

ILLINOIS INSTITUTE OF TECHNOLOGY

Batch Scheduler

- A **system software** sitting between workloads and computing platforms
  - To decide when and where to execute user jobs
- Aka job scheduler, cluster scheduler, workload management
  - Cobalt, Slurm, PBS, Moab/Torque, …

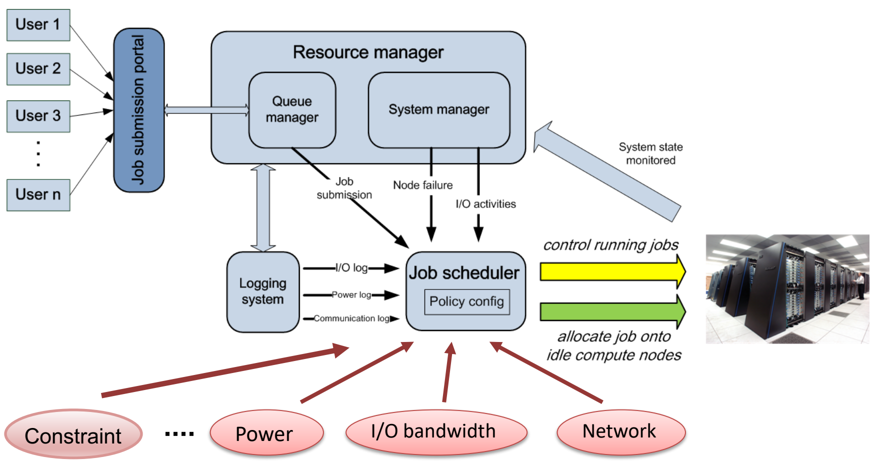


Z. Lan 6

ILLINOIS INSTITUTE OF TECHNOLOGY

Cobalt: Batch Scheduler at Argonne

https://trac.mcs.anl.gov/projects/cobalt/



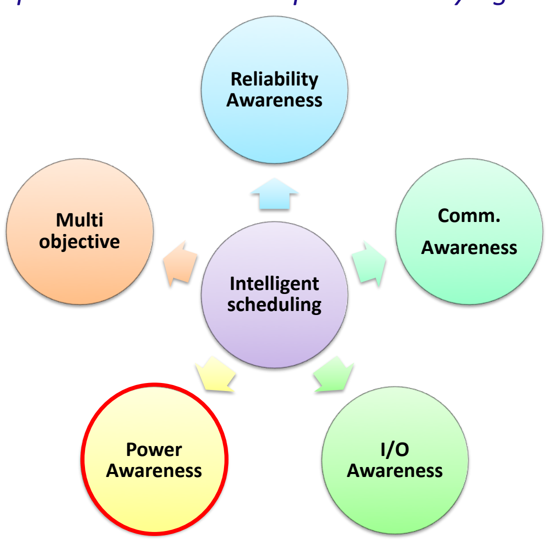
W. Allcock, P. Rich, Y. Fan, and Z. Lan, "Experience and Practice of Batch Scheduling on Leadership Supercomputers at Argonne", *JSSPP* 2017.

Z. Lan 7

ILLINOIS INSTITUTE OF TECHNOLOGY

Intelligent Scheduling

Adaptively improve its action in response to varying environments



Z. Lan, W. Allcock, P. Rich, and M. Papka, "Flexible, Intelligent, and Multi-Dimensional Resource Management System", *Position paper to DOE Extreme Heterogeneity Workshop, 2018.*

ILLINOIS INSTITUTE OF TECHNOLOGY

Power Awareness

- Actively observes, analyzes, and assesses power behaviors of the system and user jobs
  - To control system wide power consumption
  - To minimize impact on system utilization
  - To be applicable to general HPC applications
- Three components:
  1. Power analysis
  2. Dynamic power learning
  3. Power aware scheduling

S. Wallace, X. Yang, V. Vishwanath, W. Allcock, S. Coghlan, M. Papka, and Z. Lan, "A Data Driven Scheduling Approach for Power Management on HPC Systems", *SC'16*.

Z. Lan 9

ILLINOIS INSTITUTE OF TECHNOLOGY

Workload Power Analysis

- Study power data from Mira environmental database in 2014 -- statistics of job power profiles in kw per rack

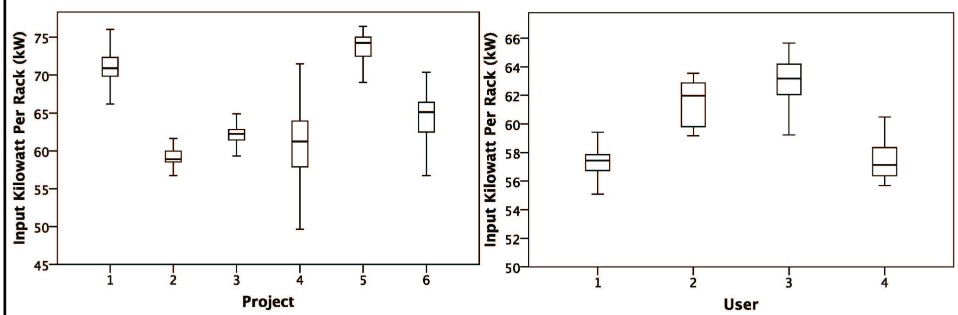
Minimum	36.48
Mean	65.67
Maximum	160.60
Percentile 05	56.60
Percentile 25	61.20
Percentile 75	71.03
Percentile 95	74.57
Percentile 99	77.99
Standard Deviation	6.18

HPC jobs have distinct power profiles with the difference being as high as 4.4 times!

Z. Lan 10

ILLINOIS INSTITUTE OF TECHNOLOGY

Workload Power Analysis

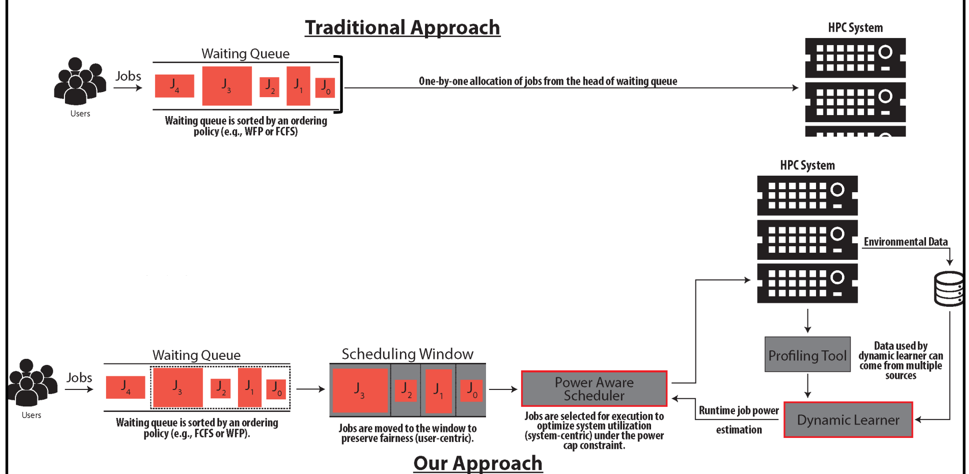


Not only do jobs have distinct power profiles, so do the projects they belong to!

Z. Lan 11

ILLINOIS INSTITUTE OF TECHNOLOGY

Power Aware Scheduling



Z. Lan 12

Illustrative Example

	Job	Job Size (Racks)	Power Profile (kW/rack)
Head of Wait queue	J_0	3	60
	J_1	1	50
	J_2	5	30
	J_3	4	40

- Allocated onto a 6-rack system with total power cap of 230 kW.
  - Conventional FCFS always selects <J_0, J_1> leaving 2 racks unused.
  - Our approach selects <J_1, J_2> as this maximizes utilization under power cap.

Dynamic Leaner

- To take power data from power monitoring facilities
- At each scheduling instance, the learner has two tasks
  - Estimate peak power requirement of **each job in the queue**
  - Calculate **the available power budget for incoming jobs** by measuring power usage of running jobs

Estimating Job Power Profile

- Two possibilities
  - No power information for the job
    - If the group's power profile is known, use group power profile
    - Otherwise, assume the maximum
  - Previous power data available
    - Use job's previous profile as its current power profile
- Our learner constantly updates power profiles of jobs and groups using **two-sample T-test**

ILLINOIS INSTITUTE OF TECHNOLOGY

Power Aware Scheduling

- To select jobs in the queue for execution
- Two major parts:
  - In contrast to one-by-one scheduling approach, we adopt **a window-based approach**
  - A **0-1 knapsack** problem is formulated to describe power aware scheduling

$$\max\Big(\sum_{1\le i\le k)} x_i \cdot n_i\Big) \le N - N_{used}, \qquad x_i = 0 \text{ or } 1$$

$$s.t. \sum_{1\le i\le k} x_i \cdot p_i \le PB - P_{used}$$

ILLINOIS INSTITUTE OF TECHNOLOGY

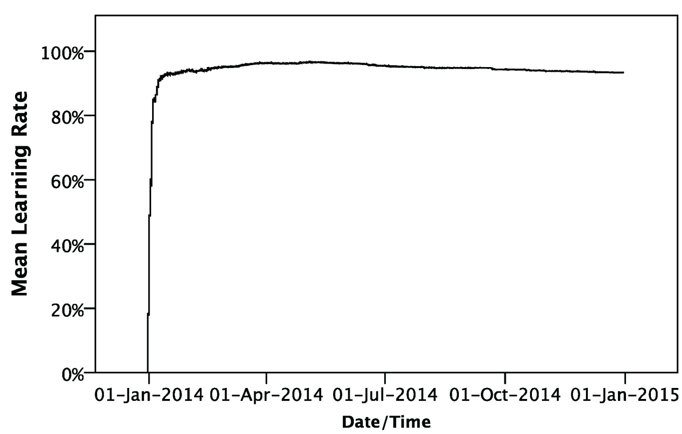
Evaluation

- Trace-based simulation
  - A stream version of CQSim (https://github.com/SPEAR-IIT/CQSim)
- Workload traces
  - Mira 2014 traces: workload log & CMCS power data
- Metrics:
  1. Learning accuracy
  2. Capping success rate
  3. Scheduling performance

Z. Lan 17

ILLINOIS INSTITUTE OF TECHNOLOGY

Learning Accuracy

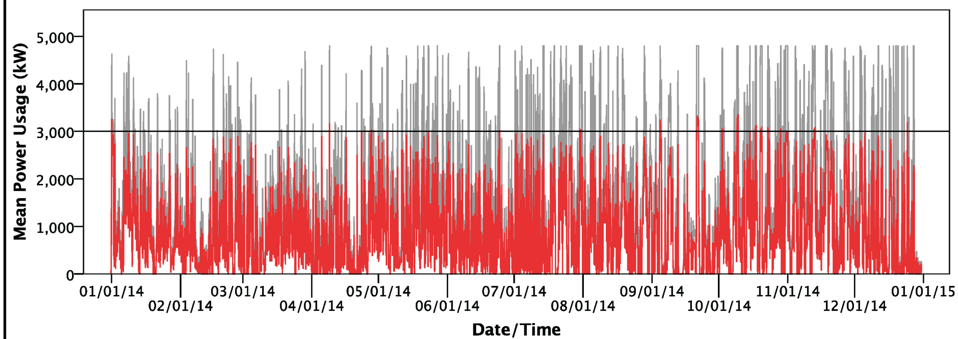


Observe 94% accurate after just 26 days of execution.

Z. Lan 18

ILLINOIS INSTITUTE OF TECHNOLOGY

Capping Success Rate

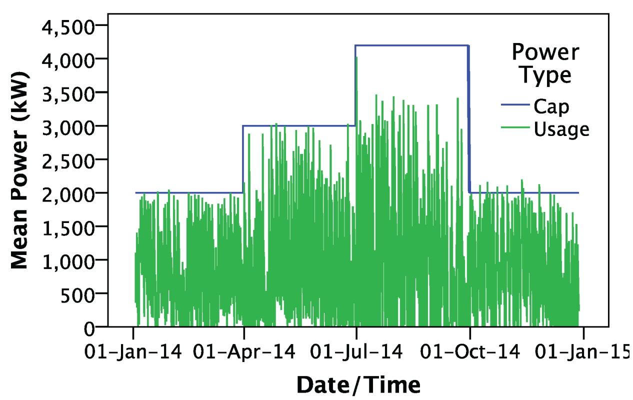


Observe 99% for the entire year.

Z. Lan 19

ILLINOIS INSTITUTE OF TECHNOLOGY

Capping Success Rate

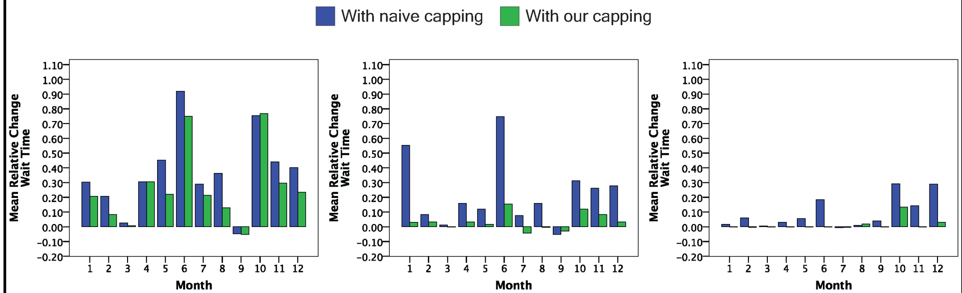


Observe 99.2% of all execution under power cap

Z. Lan 20

ILLINOIS INSTITUTE OF TECHNOLOGY

Scheduling Performance



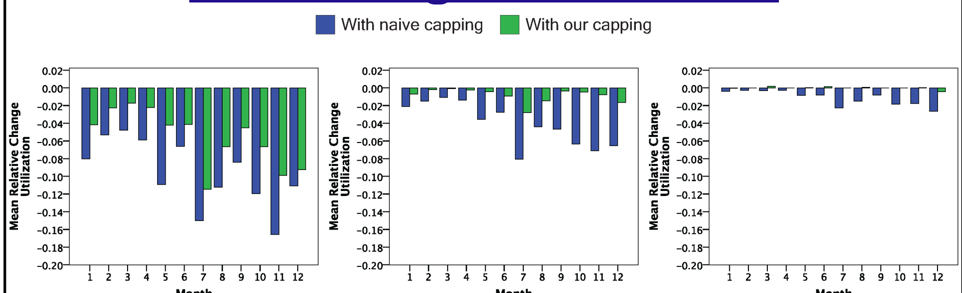
2,000 kW 3,000 kW 4,000 kW

Relative change on job wait time, normalized to un-capped FCFS

Z. Lan 21

ILLINOIS INSTITUTE OF TECHNOLOGY

Scheduling Performance



2,000 kW 3,000 kW 4,000 kW

Relative change on system utilization

Observe

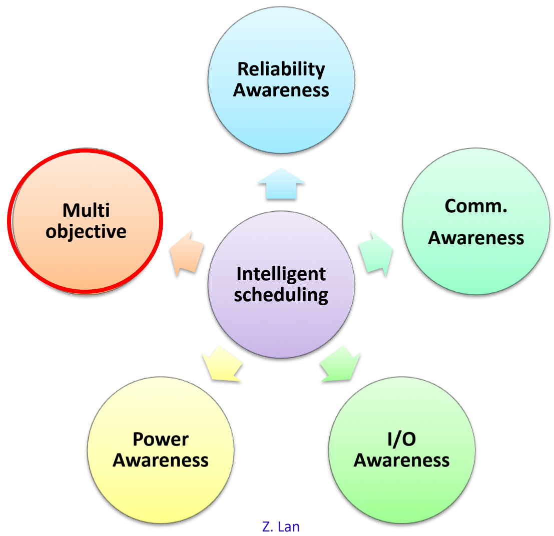
When power cap is about 83% of the maximum, minor or no degradation in scheduling performance!

Z. Lan 22

ILLINOIS INSTITUTE OF TECHNOLOGY

Intelligent Scheduling

dynamically improve its action in response to varying environments

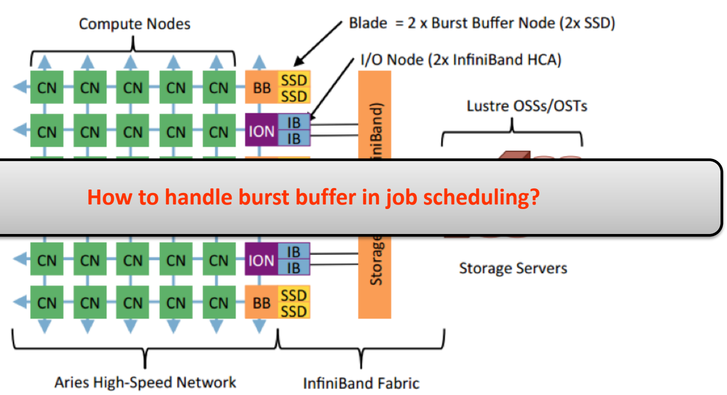


Z. Lan

23

ILLINOIS INSTITUTE OF TECHNOLOGY

Multi-Resource Scheduling



How to handle burst buffer in job scheduling?

Y. Fan, Z. Lan, P. Rich, W. Allcock, M. Papka, B. Austin, and D. Paul, "Scheduling Beyond CPUs for HPC", *HPDC'19*.

Z. Lan

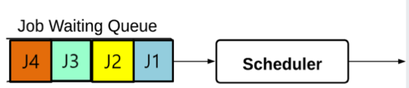
24

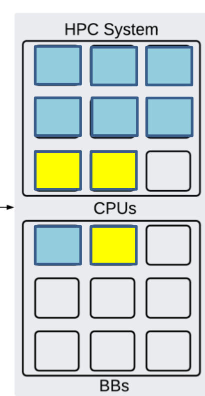
Existing Solutions

- **Naïve**: run jobs in sequence (Slurm)
- **Constrained**: maximize utilization of one resource (SC'16)
- **Weighted**: maximize the combination of utilizations of multiple resources (ICDCS'12)
- **Bin Packing**: select big jobs iteratively (SIGCOMM'14)

Illustrative Example

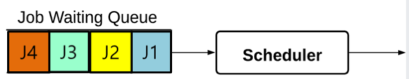
- J1: <6 CPUs, 1 BBs>
- J2: <2 CPUs, 1 BBs>
- J3: <3 CPUs, 5 BBs>
- J4: <3 CPUs, 3 BBs>

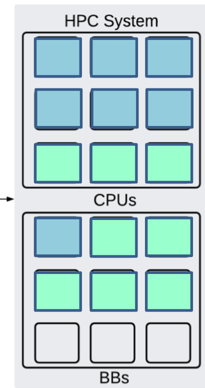




Solution	Methods	CPU Util.	BB Util.
1	Naive	89%	22%

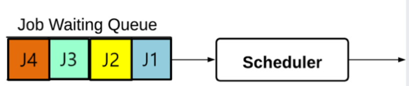
- **J1: <6 CPUs, 1 BBs>**
- **J2: <2 CPUs, 1 BBs>**
- **J3: <3 CPUs, 5 BBs>**
- **J4: <3 CPUs, 3 BBs>**

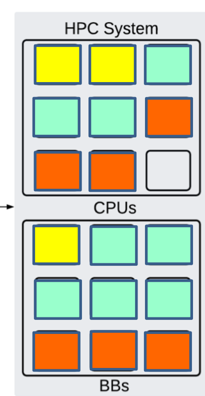




Solution	Methods	CPU Util.	BB Util.
1	Naive	89%	22%
2	Constrained, Weighted, Bin Packing	100%	67%

- **J1: <6 CPUs, 1 BBs>**
- **J2: <2 CPUs, 1 BBs>**
- **J3: <3 CPUs, 5 BBs>**
- **J4: <3 CPUs, 3 BBs>**





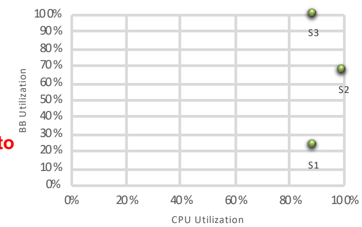
Solution	Methods	CPU Util.	BB Util.
1	Naive	89%	22%
2	Constrained, Weighted, Bin Packing	100%	67%
3	new solution	89%	100%

ILLINOIS INSTITUTE OF TECHNOLOGY

Pareto Set

Solution	CPU Util.	BB Util.
1	89%	22%
2	100%	67%
3	89%	100%

Pareto Set



- **S2 and S3 form Pareto set**
  - **S2: higher CPU; lower BB**
  - **S3: higher BB; lower CPU**

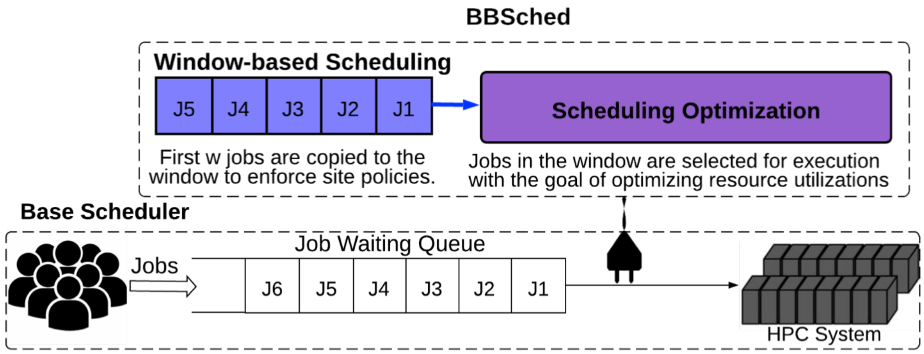
ILLINOIS INSTITUTE OF TECHNOLOGY

BBSched for Multi-Resource Scheduling

- Goal:
  - Being capable of identifying Pareto set
- Key components:
  - ***Window-based*** scheduling to preserve job ordering
  - Formulate as ***multi-objective optimization (MOO)*** problem
  - Rapidly solve the problem via ***genetic algorithm***

ILLINOIS INSTITUTE OF TECHNOLOGY

BBSched



ILLINOIS INSTITUTE OF TECHNOLOGY

Multi-Objective Optimization (MOO)

- Multiple objectives:

  Node utilization: $\sum_{i=1}^{w} \boldsymbol{n_i \times x_i}$

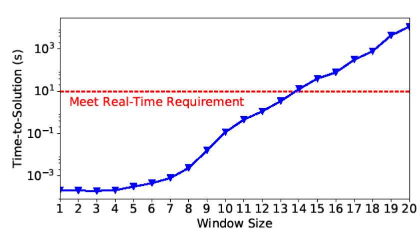
  BB utilization: $\sum_{i=1}^{w} \boldsymbol{b_i \times x_i}$

- Constraints:

  $\sum_{i=1}^{w} \boldsymbol{n_i \times x_i \leq N_{available}}$

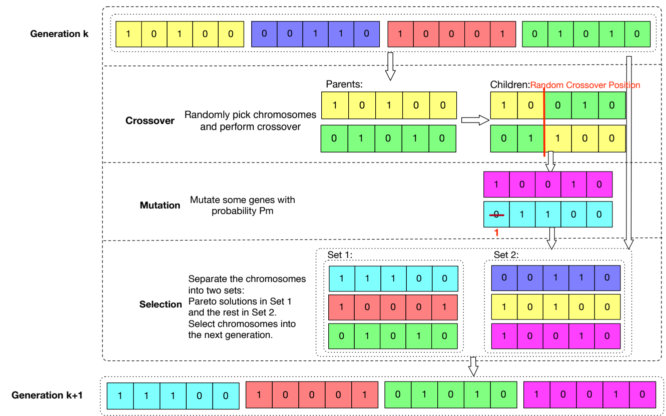
  $\sum_{i=1}^{w} \boldsymbol{b_i \times x_i \leq B_{available}}$

- MOO is NP-hard



Solving MOO

- Genetic algorithm
  - Approximate true Pareto set iteratively
  - Require much less time than exhaustive search



Scheduling Decision

- Select a solution out of the Pareto set

Solution	CPU Utilization	BB Utilization
S1	80% **10%↓**	70% **40%↑**
S2	50% **40%↓**	90% **60%↑**
S3	70% **20%↓**	80% **50%↑**
S4	90%	30%

- **S4: Maximum CPU utilization**
- **S1 and S3: BB gain 2x> CPU loss**
- **S3: Maximum BB improvement**

ILLINOIS INSTITUTE OF TECHNOLOGY

Evaluation

- Trace-based simulation via **CQSim** https://github.com/SPEAR-IIT/CQSim
- Workload traces (Cori@LBNL, Theta@ALCF)

	Cori	Theta
Location	NERSC	ALCF
Scheduler	Slurm	Cobalt
System Types	Capacity computing	Capability computing
Compute Nodes	12,076 (2,388 Haswell; 9,688 KNL)	4,392 (4,392 KNL)
Aggregated Memory	1,304.5TB	913.5TB
Shared Burst Buffer	1.8PB	1.26PB (projected)
Trace Period	Apr. 2018 - Jul. 2018	Jan. 2018 - May. 2018
Number of Jobs	2,607,054	70,507
BB Data Source	Slurm log	Darshan log
BB Range	[1GB, 165TB]	[1GB, 285TB]

Z. Lan 35

ILLINOIS INSTITUTE OF TECHNOLOGY

Workloads

- Insufficient BB requests, as BB is new
- No records on some requests
- S1 -> S4, weak to strong confliction

Workload	% of jobs requesting BB	BB Range
S1	50%	5TB+
S2	75%	5TB+
S3	50%	20TB+
S4	75%	20TB+

Z. Lan 36

ILLINOIS INSTITUTE OF TECHNOLOGY

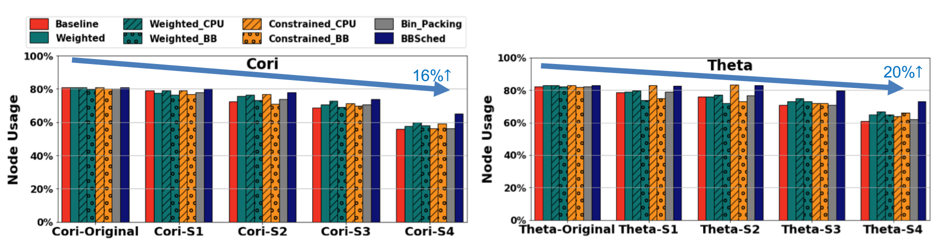
Comparison

- Methods:
  - ***Baseline***: Naïve method, no optimization
  - ***Weighted***: 50% node, 50% BB
  - ***Weighted_CPU***: 80% node, 20% BB
  - ***Weighted_BB***: 20% node, 80% BB
  - ***Constrained_CPU***: maximize node utilization
  - ***Constrained_BB***: maximize BB utilization
  - ***Bin_Packing***
  - ***BBSched***: w = 20, G = 500, P = 20
- Evaluation Metrics
  - ***Resource Utilization***: node, BB
  - ***Average Job Wait Time***

37

ILLINOIS INSTITUTE OF TECHNOLOGY

Node Utilization



BBSched improves node utilization by up to 20%

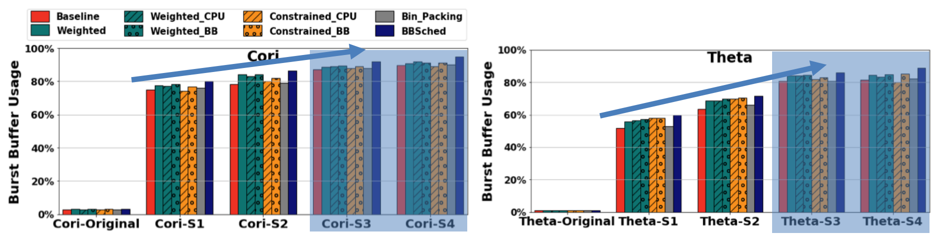
Constrained_CPU has good performance on Original, S1 and S2

S1→S4, **node utilization** ↓ due to intensive BB requests

Z. Lan

38

BB Utilization



Observe

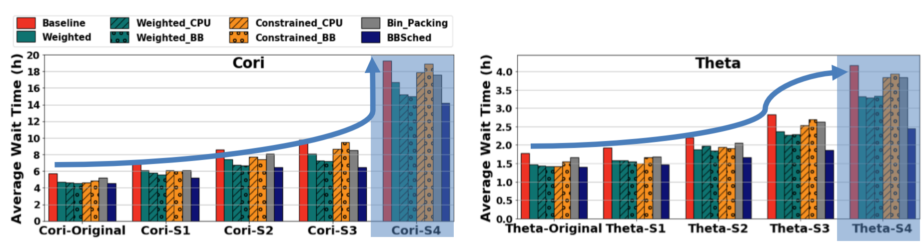
BBSched has the highest BB utilization (up to 15% increase)

BBSched > Weighted > Constrained > Bin Packing

S1→S3, BB utilization increases

S3 ≈ S4, because BBs are saturated

Avg Job Wait Time



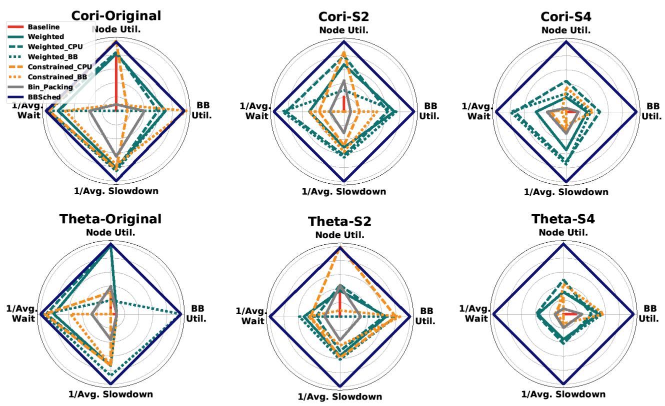
Observe

Optimization methods reduce average job wait ti**me**

BBSched achieves the greatest reduction, esp with S4 (41%)

S1→S4, average wait time increases because resources are saturated

Overall Comparison



larger area

means

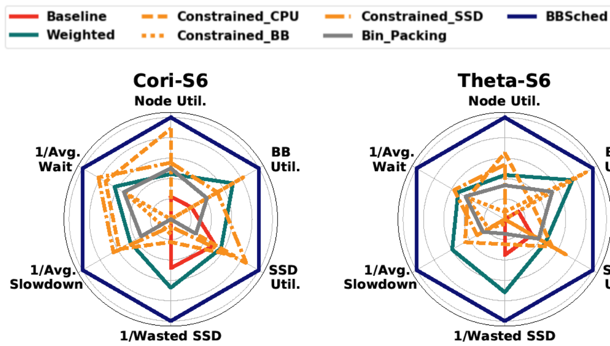
Better performance

Adding More Resources

- Three resources: CPUs, BBs, and **local SSDs**
  - Half of nodes having 128GB SSDs
  - Half of nodes having 256GB SSDs
- Add two more objectives:
  - Maximize local SSD utilization: $\sum_{i=1}^{w} s_i \times n_i \times x_i$
  - Minimize wasted local SSD: $\sum_{i=1}^{w} (\sum_{j=1}^{n_i} (l_{ij} - s_i)) \times x_i$

ILLINOIS INSTITUTE OF TECHNOLOGY

Evaluation



Z. Lan 43

ILLINOIS INSTITUTE OF TECHNOLOGY

Key Takeaway

Considering all resources in an explicit optimization is essential for **multi-resource scheduling**!

Z. Lan 44

ILLINOIS INSTITUTE OF TECHNOLOGY

Open Issues

- Emerging latency-sensitive, short-running, malleable, ensemble-based jobs
    - Borrow techniques from cloud scheduling
    - E.g., hierarchical approach (Mesos)
- Hybrid nodes (CPUs & accelerators)
    - Fine-grained job scheduling
- Existing approaches are heuristics or optimization base
    - Exploit advanced RL to improve scheduling efficiency
- Finally, data collection and real-time processing for driving intelligent scheduling

Z. Lan 45

ILLINOIS INSTITUTE OF TECHNOLOGY

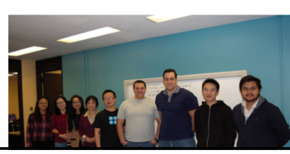
http://cs.iit.edu/~lan/SPEAR-Team.html

SPEAR Research Group

SPEAR

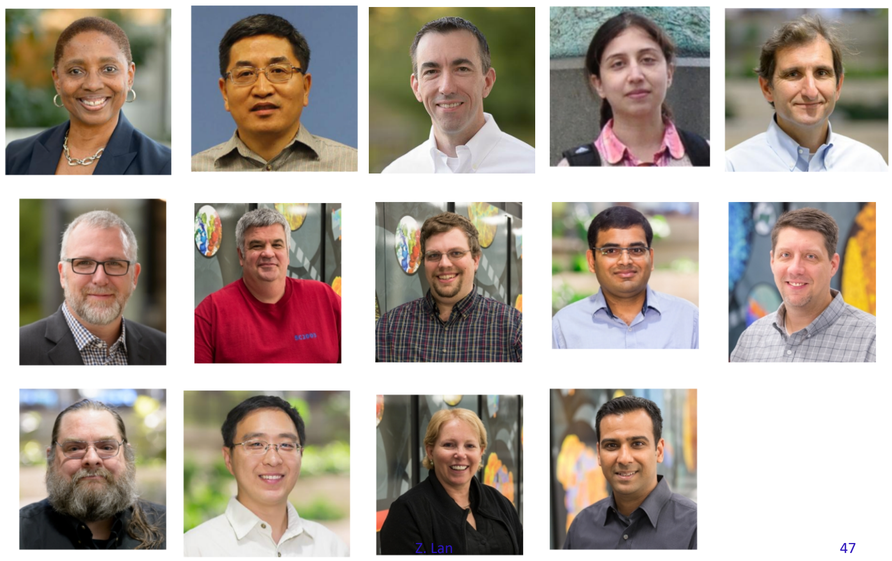
Welcome to the SPEAR (Systems for Performance, Energy, And Resiliency) team's web page!

The team conducts research spanning various areas of parallel and distributed systems including cluster management, interconnection networking, performance modeling and simulation, power and energy efficiency, and fault tolerance. Our mission is to design scalable methods and software for large-scale HPC, AI, and data analysis. The team has a strong collaboration with the ALCF and MCS divisions at Argonne National Lab.



Z. Lan 46

Collaborators at Argonne



Contact: Zhiling Lan

Email: lan@iit.edu

Website: http://www.cs.iit.edu/~lan