ILLINOIS INSTITUTE OF TECHNOLOGY

ILLINOIS INSTITUTE OF TECHNOLOGY

Argonne NATIONAL LABORATORY

The Effect of System Utilization on Application Performance Variability

Boyang Li*, Sudheer Chunduri+ , Kevin Harms+ , Yuping Fan* , Zhiling Lan*

Illinois Institute of Technology*
Argonne National Laboratory+

Outline

- Motivation
- Related Work
- Project Contributions
- Summary

ILLINOIS INSTITUTE OF TECHNOLOGY

Motivation

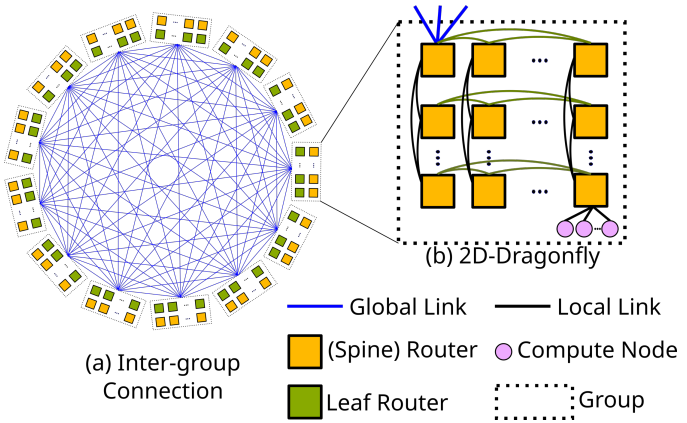
Dragonfly topology becomes popular

- High-radix
- Low-diameter

Theta at Argonne

- 4,392 nodes
- Peak performance of 11.69 petaflops
- 2D-Dragonfly topology

Performance variability due to network sharing!



Dragonfly topology

Related Work

- Communication interference due to network contention is a dominant cause of performance variability.

- Existing studies of exploiting job scheduling to mitigate communication interference:
  - Job placement
  - Routing policy
  - Task mapping

[1] Nikhil Jain, Abhinav Bhatele, Xiang Ni, Nicholas J Wright, and Laxmikant V Kale. 2014. Maximizing throughput on a dragonfly network SC14'

[2] Xu Yang, John Jenkins, Misbah Mubarak, Robert B Ross, and Zhiling Lan. 2016. Watch out for the bully! job interference study on dragonfly network. SC16'

[3]Xin Wang, Misbah Mubarak, Xu Yang, Robert B Ross, and Zhiling Lan. 2018. Trade-Off Study of Localizing Communication and Balancing Network Traffic on a Dragonfly System. IPDPS18'

ILLINOIS INSTITUTE OF TECHNOLOGY

Overview

Distinct from previous studies, we investigate how system utilization influences application runtime variability.

- Empirical analysis:

- Log analysis
- Application experiments (over 4000 tests)

- New scheduling design:

-CEIL (***C***ut-off ***E***xtreme h***I***gh uti***L***ization) design

ILLINOIS INSTITUTE OF TECHNOLOGY

Empirical Study - Log Analysis

Table: Logs of Theta at ALCF

Log name	Number of record items	Time period
Aprun log	307303	Jan/2018-March/2018
Cobalt log	44870	Jan/2018-March/2018

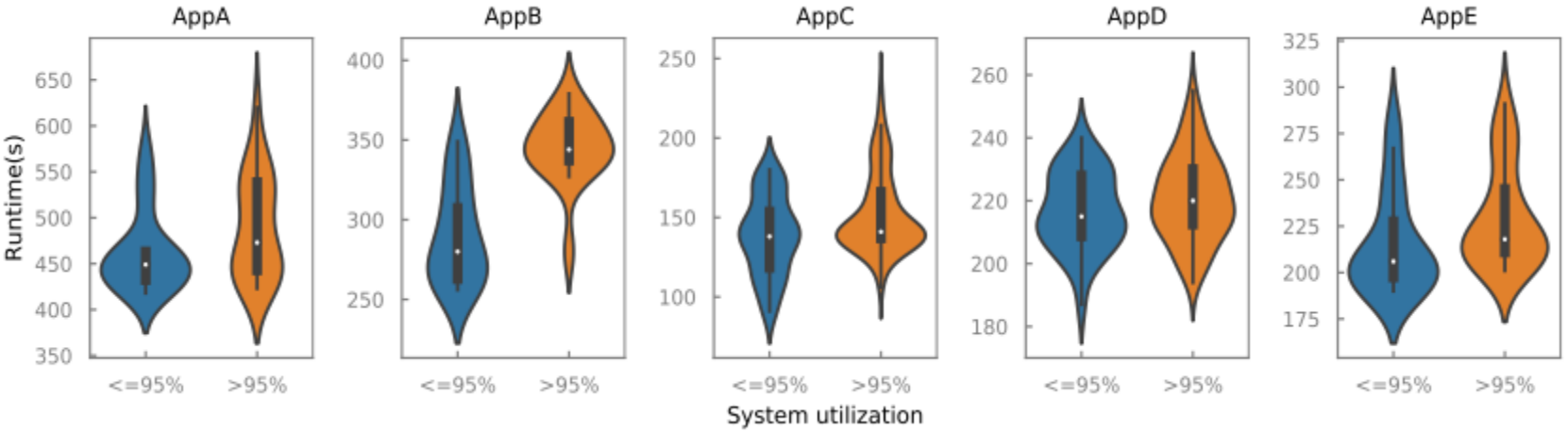
Table: Theta Aprun log field names and description

Symbol	Description
USERNAME	user name
NUM_NODES	number of nodes
EXECUTABLE	name of executable file
PROJECT_NAME	project name
CMD_LINE	aprun command to run the job
JOB_COMMAND	script name and location
CWD	current working directory
EXIT_CODE	0 means exit normally

- Records belong to the same application: all of the above Aprun log information is matched
- Fifteen applications that have multiple executions are identified.
- Top five applications with high repetition frequency for various job sizes are presented.

ILLINOIS INSTITUTE OF TECHNOLOGY

Empirical Study - Log Analysis



Application runtimes (Jan-March of 2018 on Theta) under different system utilization rates.

Positive correlation between high system utilization and application performance degradation (up to 21%)

Maximum runtime always occurred during high utilization periods.

ILLINOIS INSTITUTE OF TECHNOLOGY

Empirical Study - Application Experiments

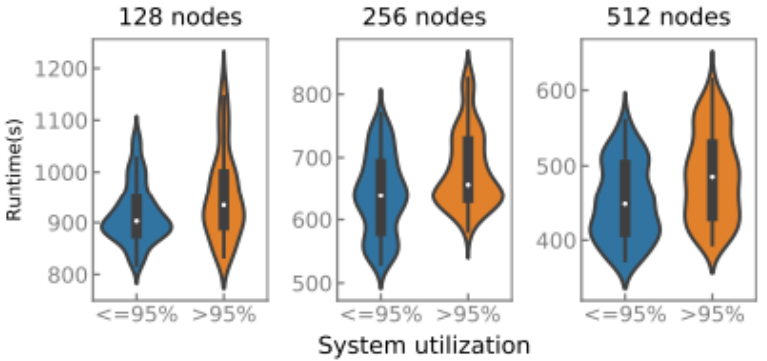
Table: Experiment description

Application name	Number of nodes	Number of runs
MILC	128	502
	256	520
	512	440
Reordered MILC	128	241
	256	509
	512	560
Nek5000	128	156
	256	205
	512	120
NEKBONE	128	365
	256	319
	512	259

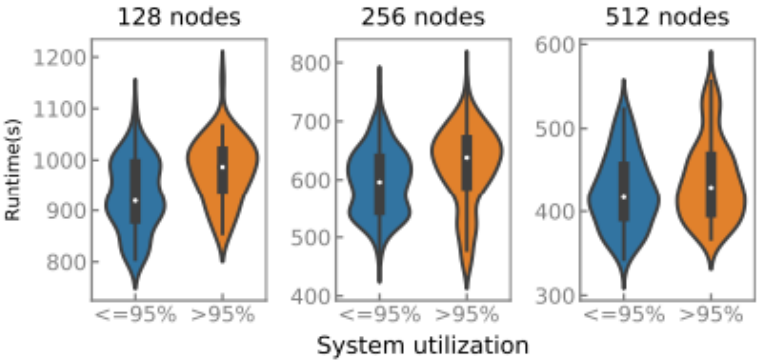
- Four applications: MILC, Reordered MILC, Nek5000, NEKBONE
- Over 4000 application tests in total on different days and times
- Cobalt log => average system utilization during these application runs.

ILLINOIS INSTITUTE OF TECHNOLOGY

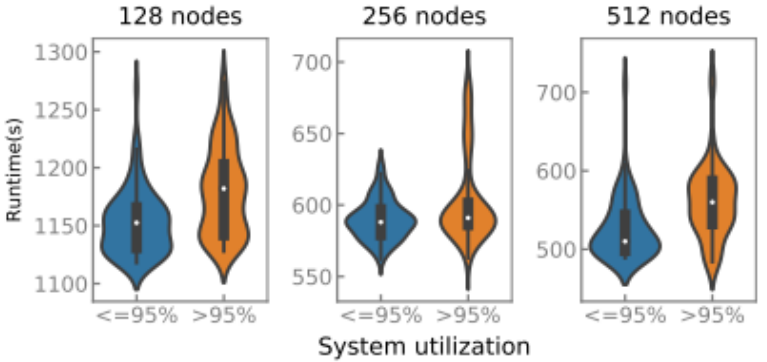
Empirical Study - Application Experiments



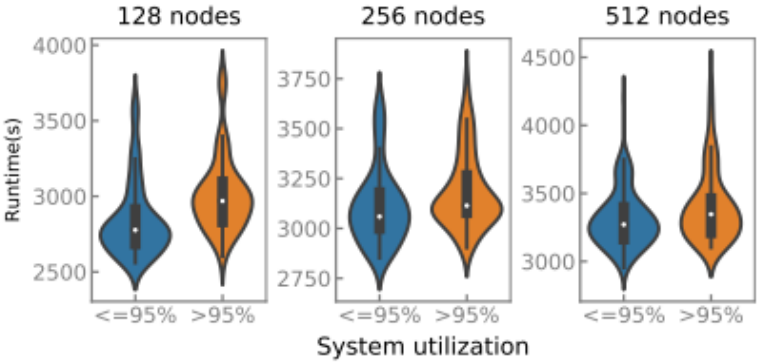
(a) MILC



(b) Reordered MILC



(c) Nek5000



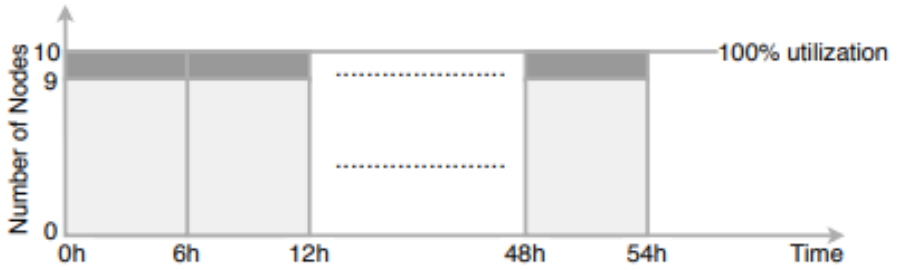
(d) Nekbone

Same observation as from log analysis!

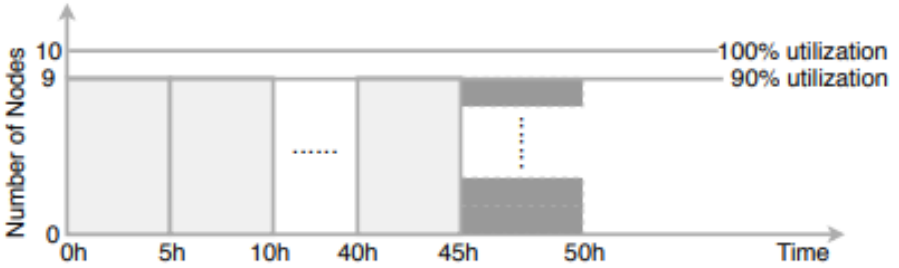
Rethink HPC Scheduling Design

Q: Shall we solely target high system utilization on Dragonfly system for scheduling?

Illustrative Example



(a) A scheduling design targeting at high system utilization



(b) A scheduling design targeting at high system productivity (i.e., makespan)

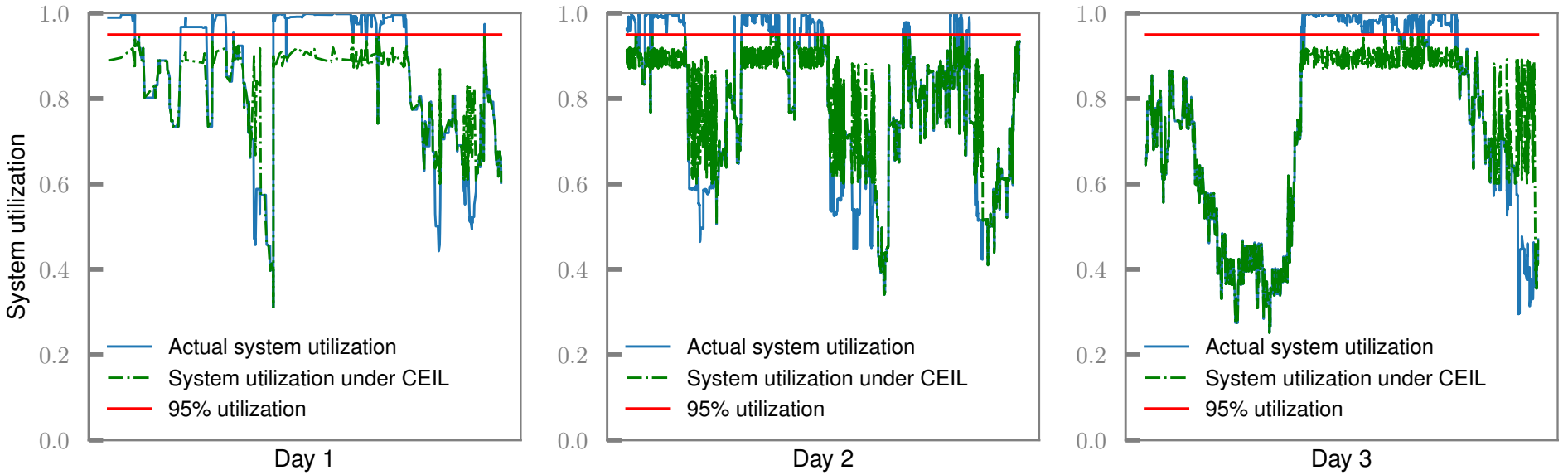
Scheduling for utilization vs for productivity

- Nine 9-node jobs and nine 1-node jobs, each having a runtime estimate of 5 hours
- Assume each application's runtime will be increased by 20% (thus becoming 6 hours) due to network sharing when system utilization is greater than a threshold (e.g., 95%).

High system utilization does not necessarily mean high system productivity

ILLINOIS INSTITUTE OF TECHNOLOGY

CEIL: Scheduling Design



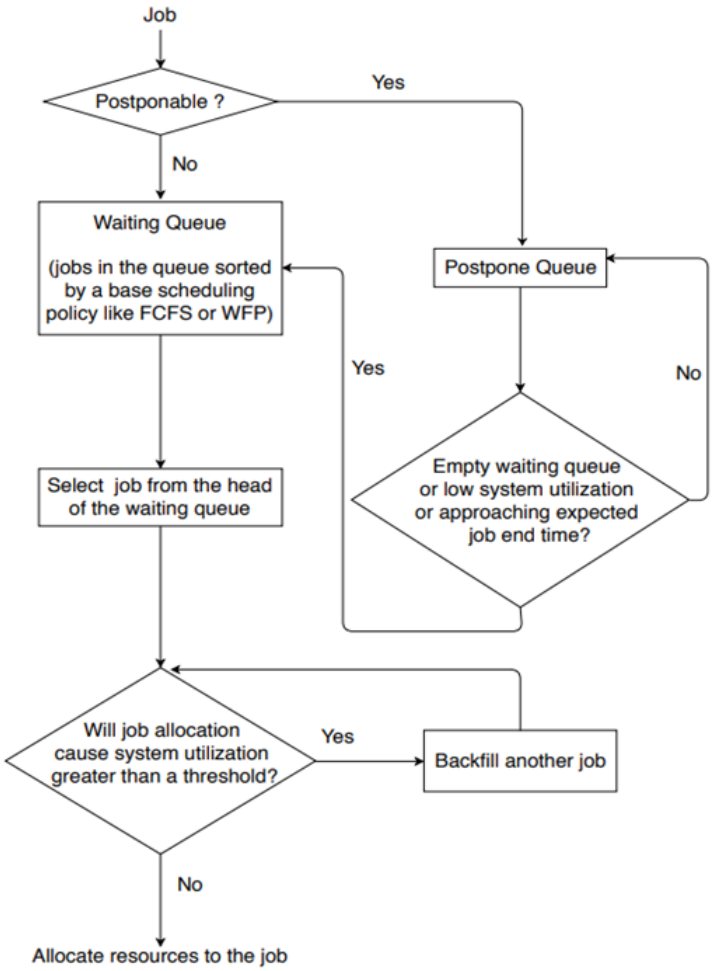
Two assumptions:

- Resource utilization exhibits a fluctuating pattern throughout a day.
- Not all the users are in a hurry for the job completion.

CEIL: Scheduling Design

CEIL (***C***ut-off ***E***xtreme h***I***gh uti***L***ization) scheduling design:

- There is an additional Postpone Queue besides traditional Waiting Queue
- Only the jobs in the Waiting Queue can be scheduled for execution.
- One of the following conditions is satisfied, jobs move from Postpone Queue to Waiting Queue
  - Empty Waiting Queue
  - Low utilization
  - Approaching user's expected job completion time



Flowchart of CEIL design

Scheduling Evaluation

- Theta workload logs

Table: Workload traces from Theta at ALCF

Time period	Nodes	Users	Projects	Jobs
07/01/2017-/07/31/2017	3624	148	41	7665
01/01/2018-/01/31/2018	4392	132	75	16204

- Synthetic logs

Table: Workloads with various postponed rates

Trace	Postponable jobs%	Workload name
Theta in 07/2017	30%	Workload 1
	50%	Workload 2
	70%	Workload 3
Theta in 01/2018	30%	Workload 4
	50%	Workload 5
	70%	Workload 6

- Trace-based scheduling simulator: CQSim

  CQSim github link: https://github.com/SPEAR-IIT/CQSim

ILLINOIS INSTITUTE OF TECHNOLOGY

Evaluation Metrics

System centric metrics:

- Makespan (e.g., to evaluate scheduling throughput)
  -Total length of the schedule to complete all the jobs.

- Percentage of high utilization periods
  -Proportion of the time when the system utilization is higher than 95% in this study

User centric metrics:

- User wait time
  -Time period between a job's expected end time and its actual end time.

- Job bounded slowdown
  -Ratio of job response time (user wait time plus job runtime) to the job runtime

ILLINOIS INSTITUTE OF TECHNOLOGY

System Centric Results

- We compare CEIL with WFP (original scheduling policy deployed on Theta).
- EASY Backfilling is used to mitigate resource fragmentation.

Table: Comparison of system-level scheduling metrics

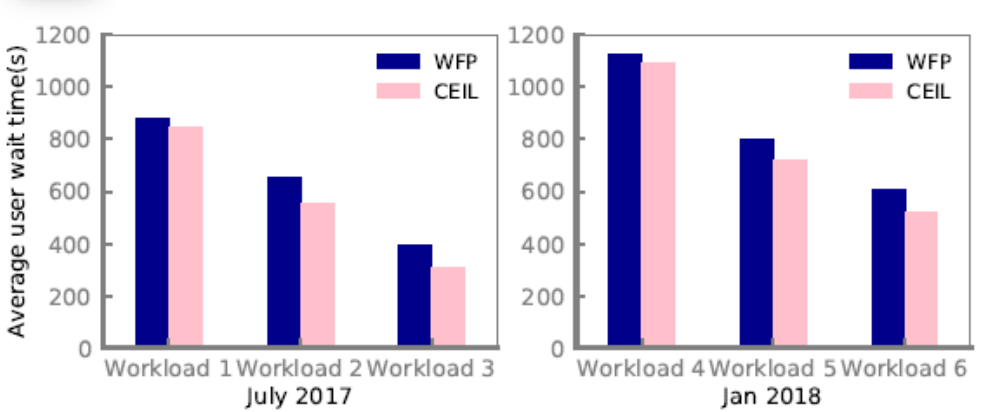
Workload	Scheduling policy	Makespan(s)	Percentage of high utilization periods
Workload 1,2,3	WFP	2608532	21.81%
	CEIL	2608497	0.06%
Workload 4,5,6	WFP	2684287	45.20%
	CEIL	2684202	0.09%

CEIL can significantly reduce the percentage of high utilization periods.

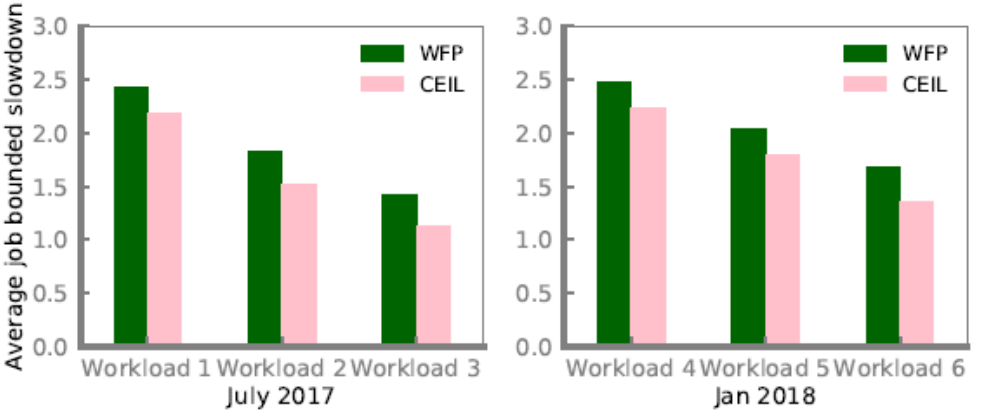
CEIL does not does not impact system throughput.

ILLINOIS INSTITUTE OF TECHNOLOGY

User Centric Results



(a) Average user wait time



(b) Average job bounded slow down

Comparison of CEIL and WFP

CEIL can effectively reduce average user wait time by 12.5%-35.3%.

Job bounded slowdown is reduced by 7.4%−20.2%.

ILLINOIS INSTITUTE OF TECHNOLOGY

Summary

In this work, our contributions are summarized as below:

- There is a ***strong*** correlation between application runtime and system utilization.
- We have investigated a scheduling strategy CEIL to ***proactively*** avoid job allocation under high system utilization.

This is a proof of concept study. Limitations:

- Selection of 95% as the high utilization is specific to the Theta workload.
- Not suitable for the systems which are always heavily utilized.

ILLINOIS INSTITUTE OF TECHNOLOGY

Acknowledgement

ILLINOIS INSTITUTE OF TECHNOLOGY

ILLINOIS INSTITUTE OF TECHNOLOGY

ILLINOIS INSTITUTE OF TECHNOLOGY

Argonne NATIONAL LABORATORY

Questions?

Thank you!