SYNOPSYS®
Silicon to Software™

Resource Aware Scheduling for EDA Regression Jobs

PMACS - 2019
INTERNATIONAL EUROPEAN CONFERENCE ON PARALLEL AND DISTRIBUTED COMPUTING (EURO-PAR) 2019

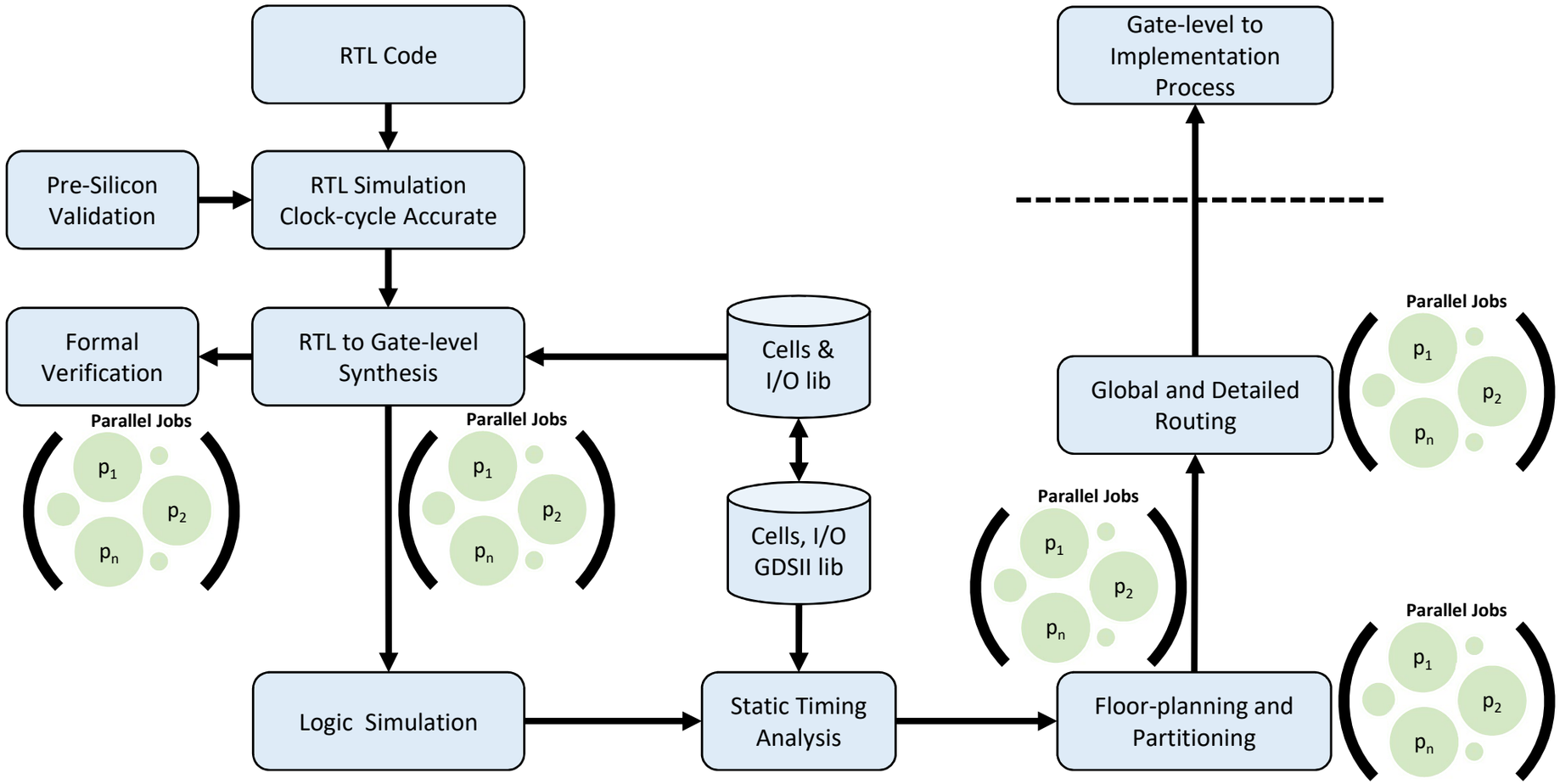
Saurav Nanda, ***Ganapathy Parthasarathy**,
Parivesh Choudhary, and Arun Venkatachar

Agenda

- Background
  - EDA Flow
  - Regression Jobs
- Motivation
- Problem Statement
- Proposed Solution
- Experiments and Results
- Conclusion

Modern Integrated Circuit design flow

Simplified view

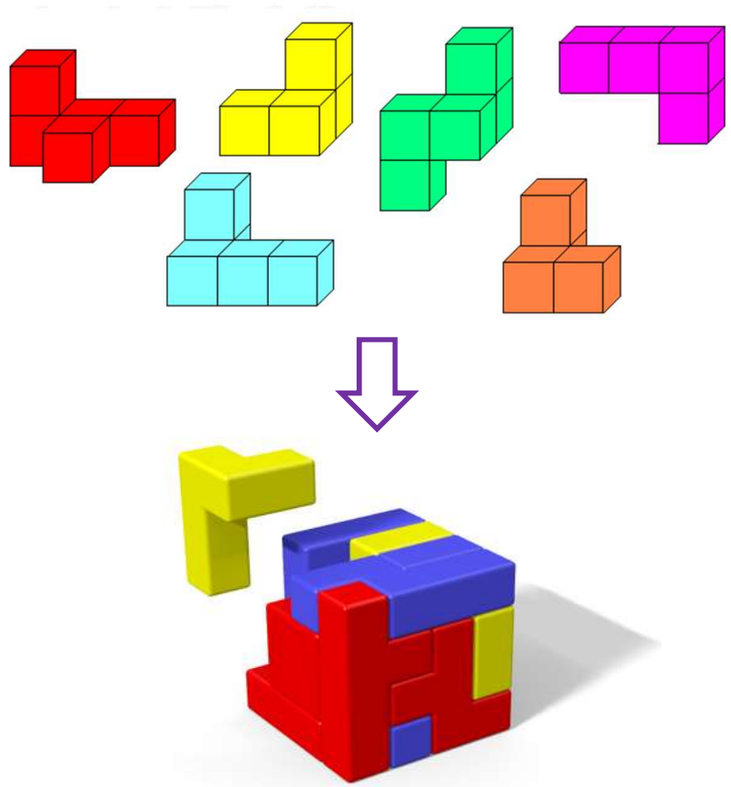


SYNOPSYS®

The Big Picture

Job scheduling for EDA regression jobs

- **Multi-resource grid scheduling algorithm**
  - pack jobs based upon individual resource requirements.



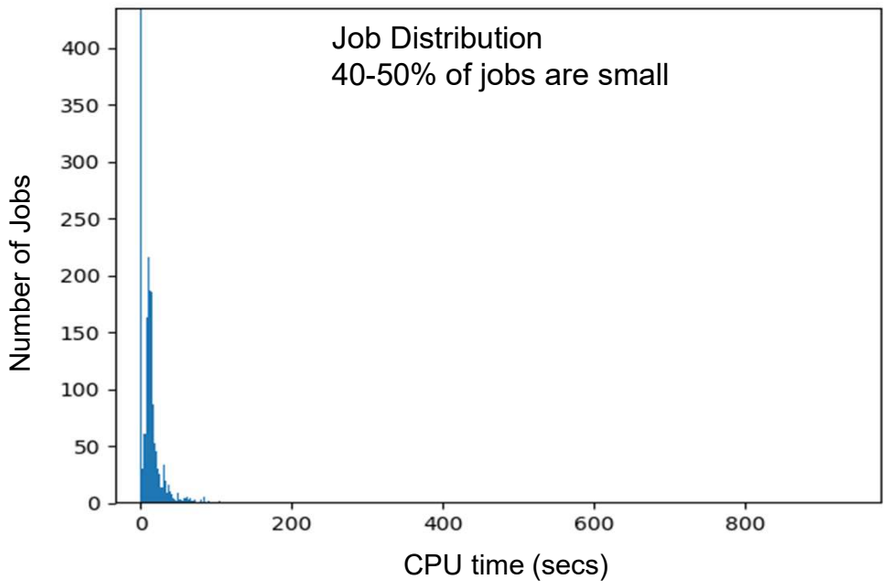
- **Tradeoff – performance vs fairness**
- **Goals**
  - Maximize the resource utilization in the grid.
  - Minimal execution time (TAT).
  - Minimize the operation cost of infrastructure.
- **Proof-of-concept**
  - Histogram-based binning for regression jobs.

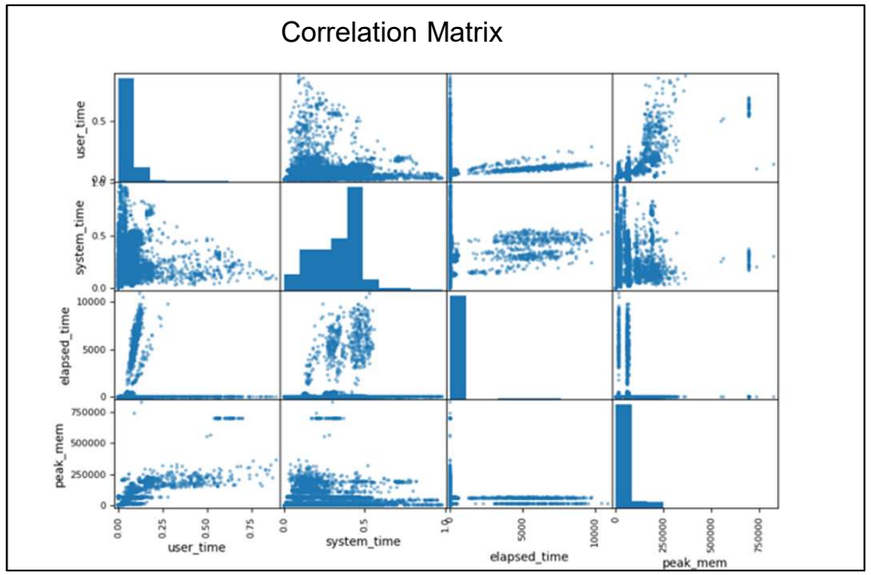
SYNOPSYS®

Motivation

EDA job Submission using traditional grid engines

- Variation in TAT of regression runs is very high.
- Longtail in regression jobs.
- We propose a resource-aware scheduling
  - Balance performance and fairness.
- Analysis of historical profile information from a set of regressions
- Highly variable resource requirements to demonstrate
  - regression jobs are well suited for efficient packing on grid machines.
- Our solution uses adaptive histogram-based binning techniques
- Evaluated the performance of proposed solution using real workload on the compute grid.

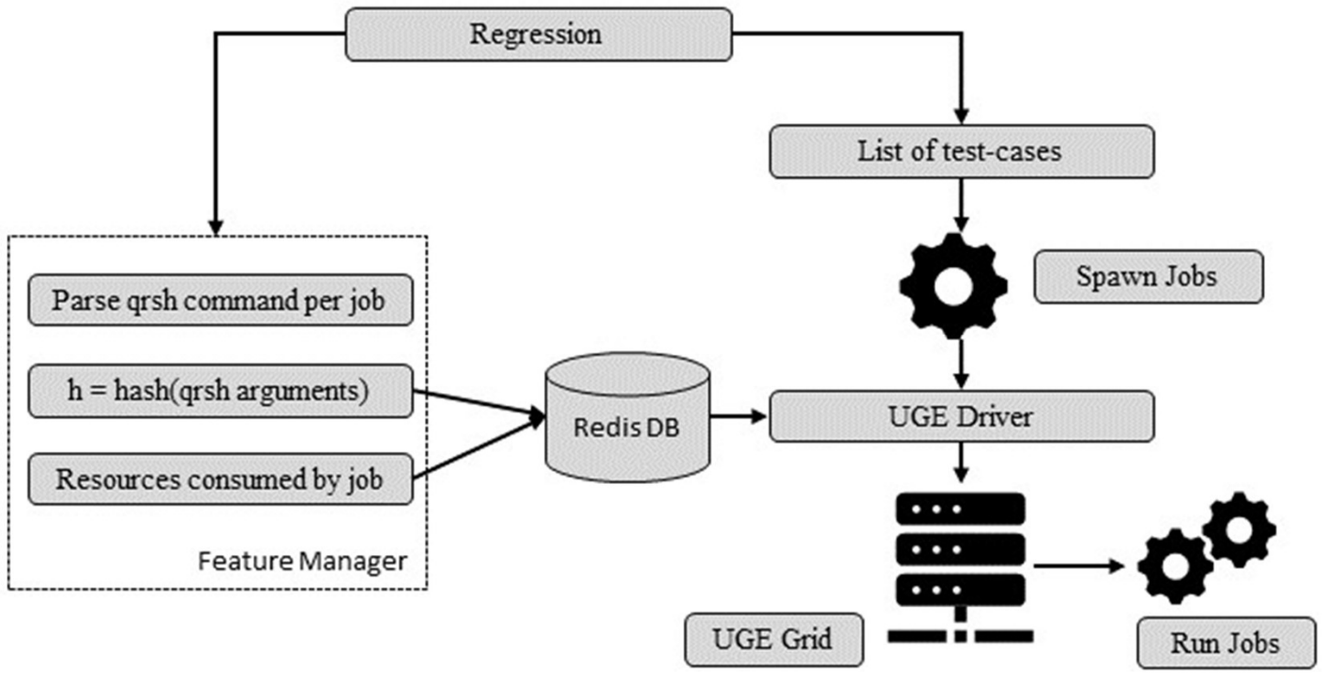




SYNOPSYS®

Proposed Approach

Job Scheduling for EDA regression jobs

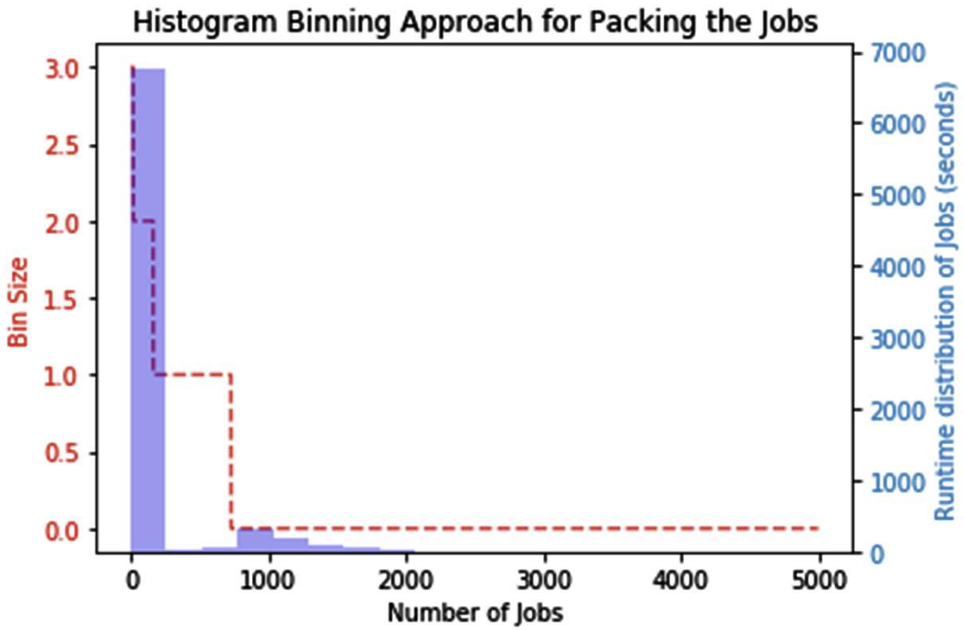


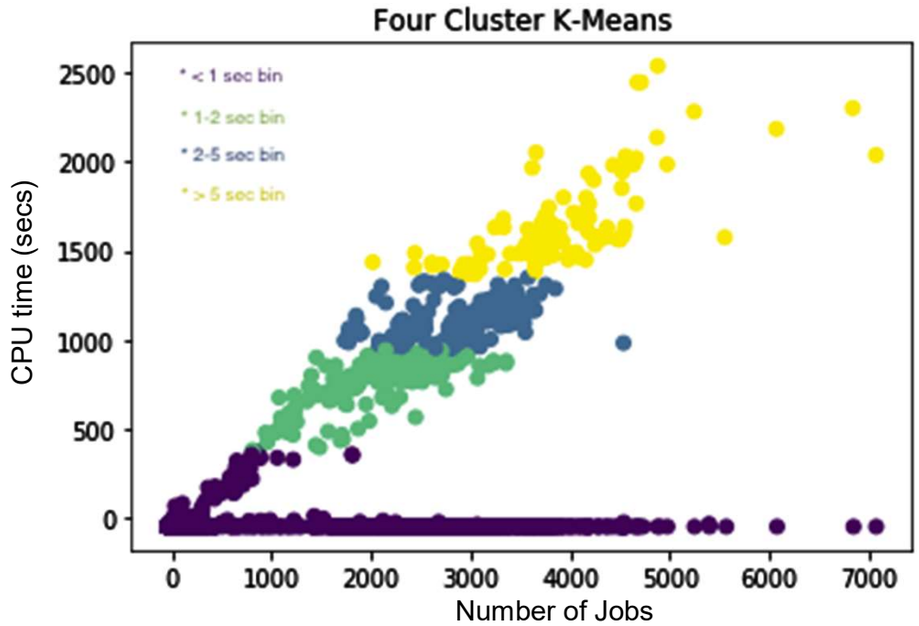
- *Collect and parse logs to capture job characteristics.*
- *Custom driver replaces actual "**qsub**"*
  - *gets all incoming jobs from any given testcase.*
- *In-memory db for historical information*
- *Small jobs placed locally*

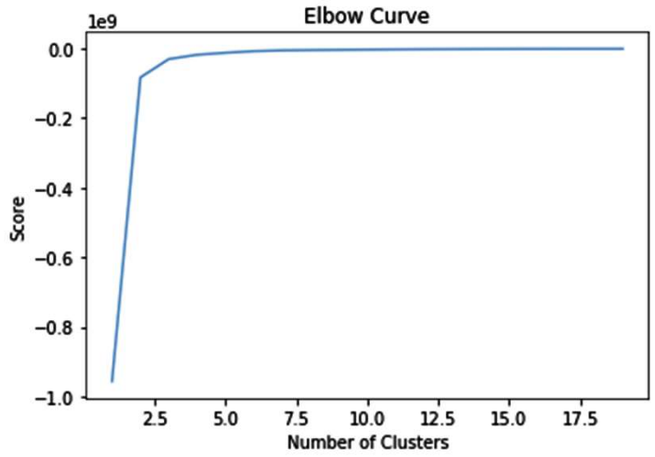
SYNOPSYS®

Job Packing and Scheduling Techniques

Histogram Binning and K-means Clustering





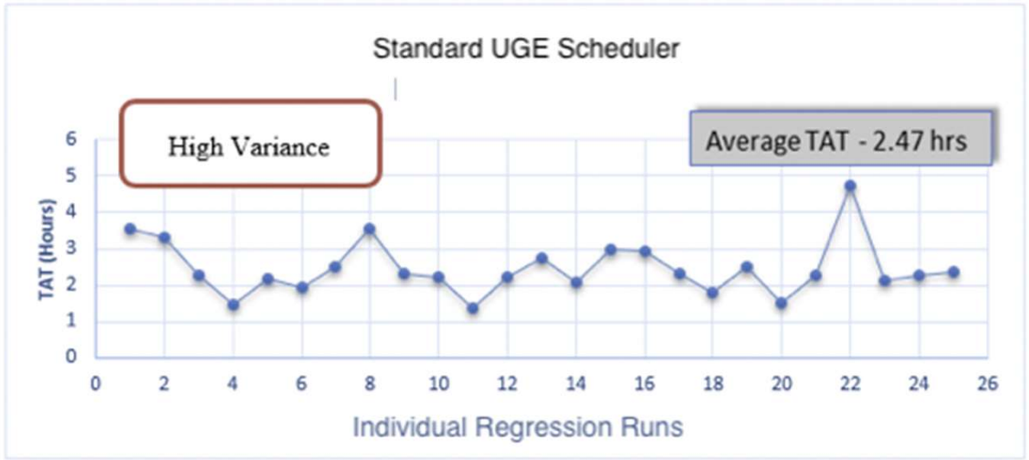


- k lies between 3 and 4,
  - close to number of bins from histogram binning approach.
- We generate the clusters using k = 4.

SYNOPSYS®

Experiments and Results

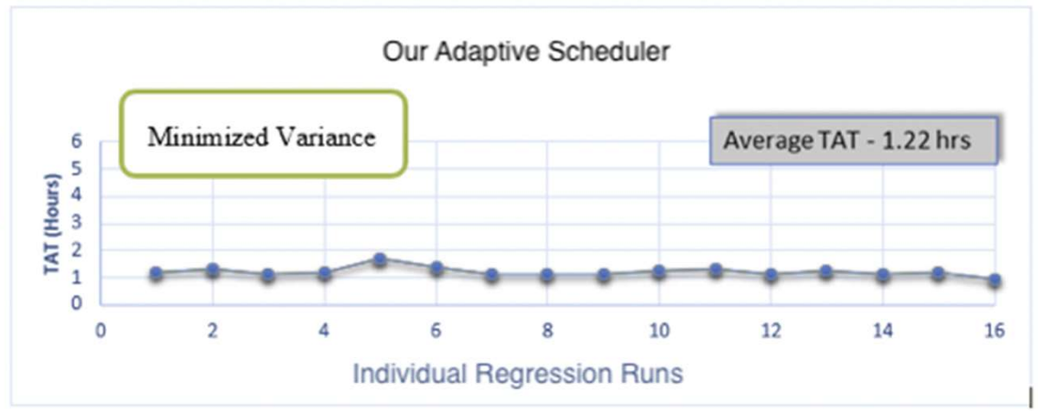
Dedicated Grid Environment - where no other jobs are running



Mean	2.46
Median	2.28
Sample Variance	0.55
Minimum	1.36
Maximum	4.75

Assumptions in Analysis

- Same number of test-cases
  - ~500
- All runs w/ outliers discarded
  - System issues
  - Testcase timeouts excluded



Mean	1.22
Median	1.19
Sample Variance	0.03
Minimum	0.93
Maximum	1.70

Current Results

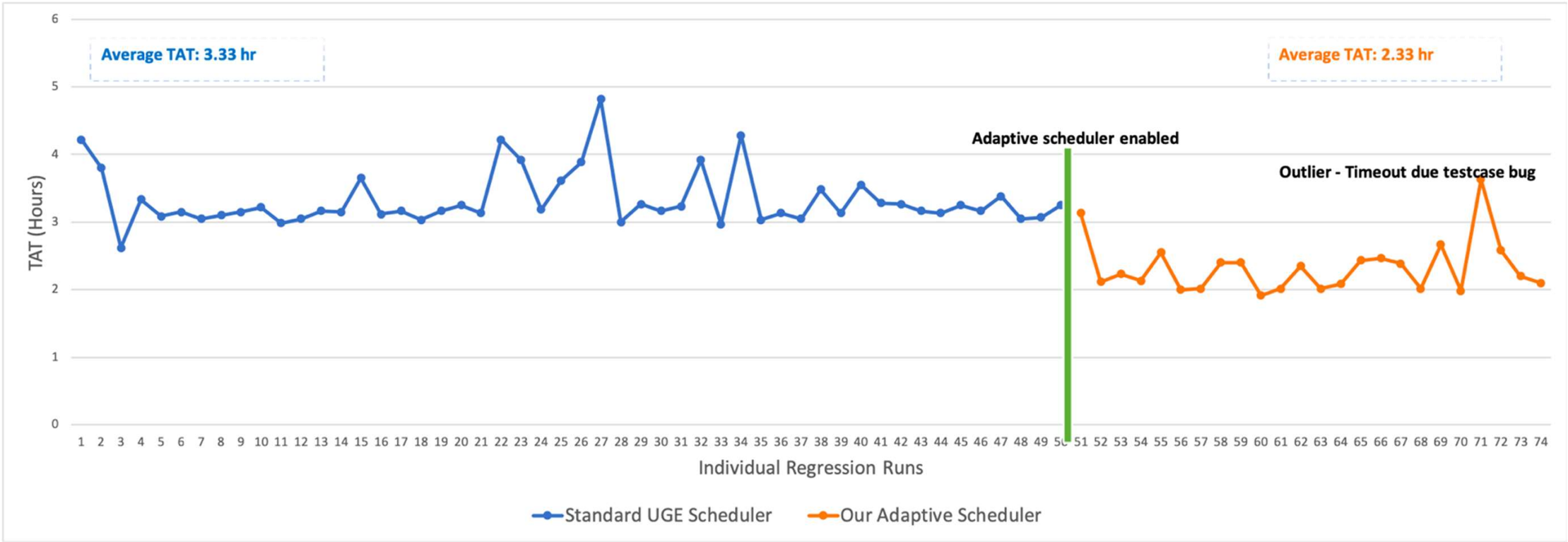
- ~25% reduction in mean TAT
- ~50% reduction in variance

Total elapsed time: adaptive scheduler versus standard UGE scheduler to run 550 test cases per iter. on dedicated grid

SYNOPSYS®

Experiments and Results

Real-world Grid Environment - where other HPC jobs are running in parallel

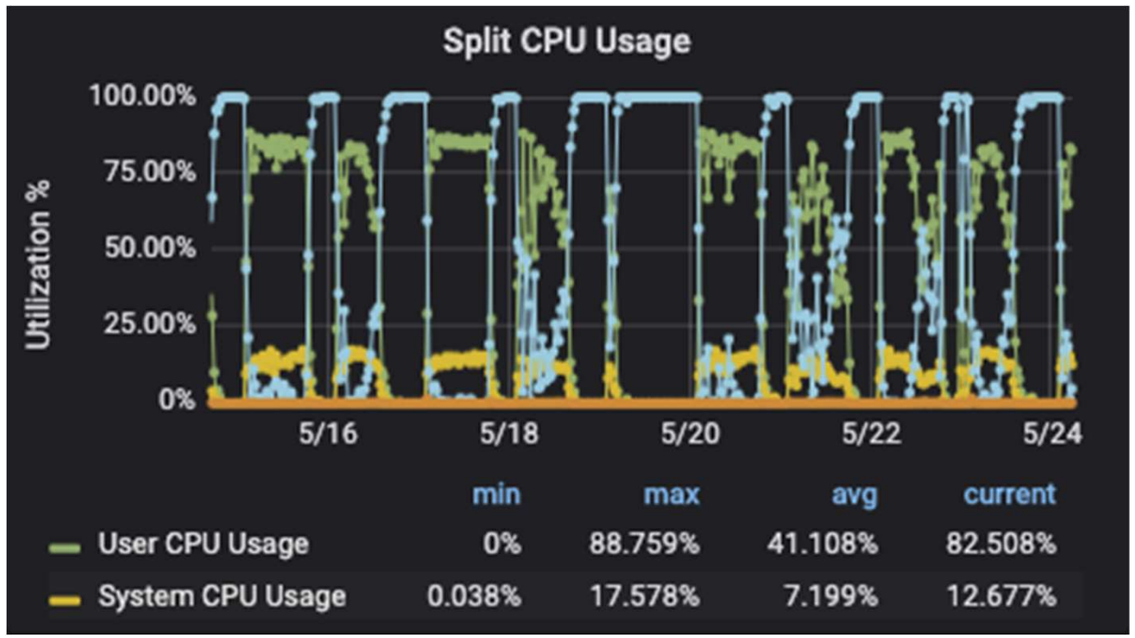


Total elapsed time: adaptive scheduler versus standard UGE scheduler to run 550 test cases per iteration.

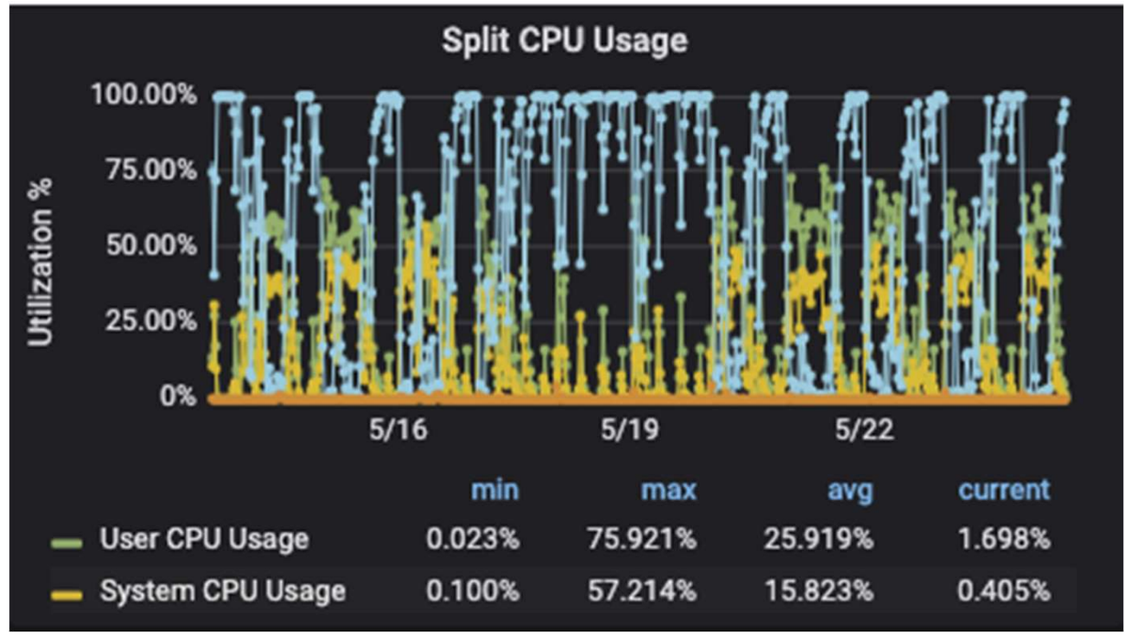
SYNOPSYS®

Experiments and Results

Improvement in Grid Utilization



Utilization with our optimization



Utilization without our optimization

Improvement in grid (cluster of servers) resource utilization with our resource-aware job scheduling technique.

SYNOPSYS®

Conclusion

Lessons Learned and Future Path

- An improvement of 34% compared to standard algorithm used by UGE job scheduler.
- This technology can be bundled as part of solutions to aid farm management at customer site
  - Minimal execution time (TAT).
  - Maximize the resource utilization in the grid.
  - Minimize the operation cost of infrastructure.
- Future work:
  - Our POC proves the feasibility of applying ML to perform more sophisticated job scheduling.
  - Dynamic partitioning and placement.
  - Building custom low-overhead measurement utilities.

SYNOPSYS®
Silicon to Software™

Thank You