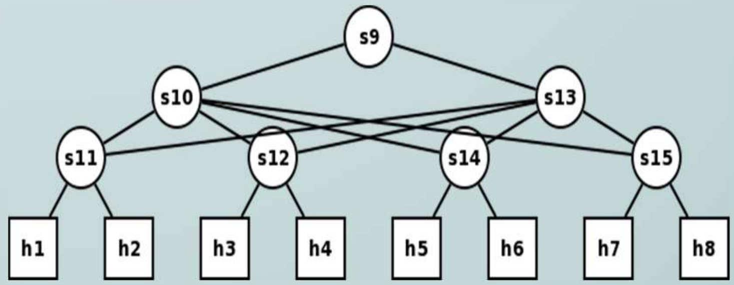
Communication-Aware Performance Prediction of Parallel Jobs

Motivation

- Network contention between concurrently running jobs on HPC systems is a primary cause of performance variability.
- Optimized job allocation that may minimize network sharing is crucial to alleviate the potential performance degradation.
- In order to do so effectively, an understanding of the interference among parallel jobs, their communication patterns, and contention in the network is required.

Problem Statement

- Analyze the interferences occurring in the parallel running jobs.
- Train a Long-Short Term Memory (LSTM) model so that it may learn about communication interference between common nodes.
- How these interference affect the actual runtime of jobs as a result.



Visualization of Compute Nodes and Switches in a Tree Topology

Approach and LSTM Model

- We propose a Long-Short Term Memory (LSTM) model that uses parallel profiling data to learn about interference between jobs.
- Input: 12 features based on job characteristics
- Used device-level communication data
- Analyzed time-series data to identify concurrent jobs
- Batching policy based on timestamp, so that model learns the status of congestion at each device at that instant.
- Prediction of performance (output): actual runtime
- Jobs with a demand of <= 2 nodes were disregarded because they create negligible interference.
- Evaluation Metrics: RMSE, MAPE

Challenges

- What parameters to choose from the available data?
- How to organize parameters to make it useful for training?
- How to select training and test data?
- What batch sampling will be useful for training ?
- Which jobs do not affect performance of other jobs?
- How to accelerate training of large datasets?

Input Dataset for Training LSTM

Timestamp	JobID	UserName	Jobname	TotalNodes	TotalCores	RequiredTime	Device	IN_PORT	Rx	EX_PORT	Tx
25 days 06:56:00	1003686	[redacted]	test20	4	80	5760	hpc554	1.0	1.566247e+09	1.0	1.573291e+09
25 days 06:56:00	1003686	[redacted]	test20	4	80	5760	IB_SW_32	8.0	1.573270e+09	22.0	2.432102e+09

- Data gathered from IIT Kanpur's PARAM Sanganak supercomputer from 5 June'23 to 5 July'23 for our experiments.
- We parse these files to create a dataset (shown above). This includes job specifications and required parameters at each timestamp.

- ***Timestamp*:** relative to first timestamp || ***JobID:*** unique job id || ***Username*:** username, useful to detect similar jobs by similar user
- ***Jobname**: user-specified name* || ***TotalNodes, TotalCores**: required resources* || ***Tx, Rx***: Number of transmitted and received packets
- ***Device**: Node/Switch involved in the communication* || ***IN_PORT, EX_PORT***: receive and transmit port of the devices

Experiments and Results

Train : Test	RMSE	MAPE
70:30	0.2266	0.3320
80:20	0.2737	0.2460
90:10	0.1778	0.4549

- The results of our testing of the model are shown above for various train:test ratios (first column).
- We trained it on data of 1/3rd jobs, including actual runtime data and, a timestamp-wise incremental-training batching policy was used.
- Above table is on same timestamp policy and whole dataset shows that RMSE decreases with increased training data. 80:20 gave the best MAPE.
- Future work will focus on fine-tuning of model and the incorporation of other batching criteria based on job characteristics such as node hours and job profiles.

References & Acknowledgement

[1] A. Pal and P. Malakar, "An Integrated Job Monitor, Analyzer and Predictor", IEEE International Conference on Cluster Computing (CLUSTER), 2021.

[2] A. Pal and P. Malakar, "MAP: A Visual Analytics System for Job Monitoring and Analysis", IEEE International Conference on Cluster Computing (CLUSTER), 2020.

*** We truly acknowledge Mr. Akshay Sharma and Computer Centre for providing us data and support.

Application No. 2330088
Name- Havi Bohra
Mentor: Prof. Preeti Malakar
Dept. of Computer Science and Engineering