Trace-Based Evaluation of Job Runtime and Queue Wait Time Predictions in Grids

Ozan Sonmez, Nezih Yigitbasi, Alexandru Iosup, Dick Epema

Parallel and Distributed Systems Group (PDS)
Department of Software Technology
Faculty EEMCS, Delft, the Netherlands
Introduction

• Grids
  • Multi-site and heterogeneous resource structure
  • Dynamic and heterogeneous workloads
  ➔ Highly variable job runtimes and queue wait times limit the efficient use of the resources by users
Introduction (cont.)

- **Remedy:** Prediction-based methods
  - Extensive body of research for space-shared Parallel Production Environments (PPEs)
  - **Grids** differ from traditional PPEs in both structure and typical use (e.g., heterogeneous resources, more bursty job arrivals)

- **Goal:**
  - A systematic evaluation of job runtime and queue wait time predictions in grids using real traces
What to predict?

- Job Runtime
- Queue Wait Time
- CPU Load
- Resource Availability
- Resource Failure Rates
What to predict?

• Job runtime predictions for
  • Improving the performance of backfilling in batch queueing systems*
  • Predicting queue wait times

• Queue wait time predictions for
  • Guiding the decisions of a user/grid scheduler

Prediction Methods

- **Time Series-based**
  - Analytical Benchmarking
  - Code Profiling
  - Genetic Algorithms
  - Instance-based Learning

Easy to implement
Fast delivery of predictions
Time Series Prediction

- Based on historical (classified) data
  - Time ordered set of past observations

- Example: Last2
## Grid Workload Traces*

<table>
<thead>
<tr>
<th>Traces</th>
<th>Type</th>
<th># CPUs</th>
<th>Duration (Months)</th>
<th># Tasks</th>
<th>Parallel Jobs</th>
</tr>
</thead>
<tbody>
<tr>
<td>DAS2</td>
<td>Research</td>
<td>400</td>
<td>18</td>
<td>1.1 M</td>
<td>66%</td>
</tr>
<tr>
<td>GRID5000</td>
<td>Research</td>
<td>2500</td>
<td>27</td>
<td>1.0 M</td>
<td>45%</td>
</tr>
<tr>
<td>DAS3</td>
<td>Research</td>
<td>544</td>
<td>18</td>
<td>2 M</td>
<td>15%</td>
</tr>
<tr>
<td>SHARCNET</td>
<td>Research</td>
<td>6828</td>
<td>12</td>
<td>1.2 M</td>
<td>10%</td>
</tr>
<tr>
<td>AUVER</td>
<td>Production</td>
<td>475</td>
<td>12</td>
<td>0.4 M</td>
<td>0%</td>
</tr>
<tr>
<td>NORDU</td>
<td>Production</td>
<td>2000</td>
<td>24</td>
<td>0.8 M</td>
<td>0%</td>
</tr>
<tr>
<td>LCG</td>
<td>Production</td>
<td>24515</td>
<td>4</td>
<td>0.2 M</td>
<td>0%</td>
</tr>
<tr>
<td>NGS</td>
<td>Production</td>
<td>-</td>
<td>6</td>
<td>0.6 M</td>
<td>0%</td>
</tr>
<tr>
<td>GRID3</td>
<td>Production</td>
<td>3500</td>
<td>18</td>
<td>1.3 M</td>
<td>0%</td>
</tr>
</tbody>
</table>


15.06.2009

TUDelft
Grid Workload Traces:
Bursty Job Arrivals (5 minute intervals)

Bursty arrivals reduce predictability!
Research Questions

1. What is the performance of job runtime predictors in grids?

2. What is the performance of queue wait time predictors in grids?

3. Can prediction-based grid scheduling policies perform better than traditional policies?
Job Runtime Predictions

• We have evaluated the accuracy of five time series methods under four job classifications

• Time series methods
  • Last
  • Last2
  • Running Mean (RM)
  • Sliding Median (SM)
  • Exponential Smoothing (ES)
Job Runtime Predictions

- **Job Classification Methods**
  - Create classes according to job attributes
  - Site, User, User on Site,
    - (User + Application Name + Job Size) on Site

- **Performance Metric**

\[
\text{accuracy} = \begin{cases} 
1 & \text{if } P = T_r, \\
\frac{T_r}{P} & \text{if } P > T_r, \\
\frac{P}{T_r} & \text{if } P < T_r,
\end{cases}
\]

\( P \): Predicted runtime

\( T_r \): Actual runtime
Job Runtime Predictions

Classification: (User + Application Name + Job Size) on Site

- **w/o Cl**: best results from the other three classifications
- **w Cl**: results with this classification

More specific classification improves the accuracy

No dominant prediction method
Job Runtime Predictions

Job runtimes are predicted more accurately in research grids.
Job Runtime Predictions: Summary of the results

• More specific classification improves job runtime prediction performance

• Job runtime prediction accuracy is low across all grids (except SHARCNET)
  • **Bursty Arrivals:** Same prediction error is made for all the jobs submitted together
  • Lack of **Stationarity**
    (no constant long-term mean and variance)
Queue Wait Time Predictions

- **Point-value predictions**
  - Simulate the local scheduling policy with predicted job runtimes to predict job queue wait times

- **Upper-bound predictions**
  - Predict upper bounds for queue wait times with a specified confidence level
  - Obviate the need to know the internal operation of local scheduling policies
Point-Value Predictions

- **Simulation Model**
  - **FCFS** as the local scheduling policy
  - Jobs assigned to their original execution sites
  - A point-value predictor runs on each site
    - Job runtimes are predicted with **Last2**

- **Prediction Correction Mechanism**
  - On departure, update the predicted runtimes of both the queued and the running jobs accordingly

- **Traces:** DAS2, DAS3, GRID5000, and AUVER
Point-Value Predictions

Accuracy of the point-value predictor is low. Correction mechanism improves the prediction accuracy (1% to 10%).
Upper-Bound Predictions

- Binomial Method Batch Predictor (BMBP)*
  - Predicts the specified quantile of the wait time distribution with a specified confidence level

- A predictor based on **Chebyshev’s Inequality**
  - No more than \(1/k^2\) of the values are more than \(k\) standard deviations away from the mean

- We consider a quantile (for BMBP) and a confidence level of 95%

- **Traces:** DAS2, DAS3, GRID5000, and AUVER

## Upper-Bound Predictions

<table>
<thead>
<tr>
<th>Grid-Site</th>
<th>Avg. Accuracy</th>
<th>Under-predictions</th>
<th>Perfect-predictions</th>
<th>Over-predictions</th>
</tr>
</thead>
<tbody>
<tr>
<td>DAS2-FS1</td>
<td>0.50</td>
<td>8%</td>
<td>9%</td>
<td>83%</td>
</tr>
<tr>
<td>DAS3-FS4</td>
<td>0.41</td>
<td>15%</td>
<td>4%</td>
<td>81%</td>
</tr>
<tr>
<td>Auver-clr01</td>
<td>0.20</td>
<td>12%</td>
<td>1%</td>
<td>87%</td>
</tr>
<tr>
<td>GRID5K-G1</td>
<td>0.72</td>
<td>20%</td>
<td>0%</td>
<td>80%</td>
</tr>
</tbody>
</table>

BMBP

<table>
<thead>
<tr>
<th>Grid-Site</th>
<th>Avg. Accuracy</th>
<th>Under-predictions</th>
<th>Perfect-predictions</th>
<th>Over-predictions</th>
</tr>
</thead>
<tbody>
<tr>
<td>DAS2-FS1</td>
<td>0.21</td>
<td>8%</td>
<td>0%</td>
<td>92%</td>
</tr>
<tr>
<td>DAS3-FS4</td>
<td>0.23</td>
<td>7%</td>
<td>1%</td>
<td>82%</td>
</tr>
<tr>
<td>Auver-clr01</td>
<td>0.10</td>
<td>7%</td>
<td>0%</td>
<td>93%</td>
</tr>
<tr>
<td>GRID5K-G1</td>
<td>0.24</td>
<td>16%</td>
<td>0%</td>
<td>84%</td>
</tr>
</tbody>
</table>

Chebyshev

**Trade-off between accuracy and tightness of the upper bounds**
Upper-Bound Predictions

- Both BMBP and Chebyshev fail when jobs arrive in bursts
- **User runtime estimates**, if available, can also be used in predicting upper bounds

A burst period of DAS3-FS4
Performance of Prediction-Based Grid Scheduling

- **Global Scheduling Policies**
  - Earliest Completion Time (ECT)-Perfect
  - ECT-Last2
  - Load Balancer
  - Fastest Processor First (FPF)

- **Simulation Model**
  - DAS3 and AUVER
  - Jobs arrive to a global scheduler
  - A point-value predictor runs on each cluster (Last2+Correction)

<table>
<thead>
<tr>
<th>Trace</th>
<th>Period</th>
<th>Number of Jobs</th>
<th>Avg. Util.</th>
</tr>
</thead>
<tbody>
<tr>
<td>DAS3</td>
<td>July-Oct. 2008</td>
<td>~220,000</td>
<td>~30%</td>
</tr>
<tr>
<td>AUVER</td>
<td>Aug.-Nov. 2006</td>
<td>~90,000</td>
<td>~70%</td>
</tr>
</tbody>
</table>

15.06.2009
Performance of Prediction-Based Grid Scheduling

Prediction-based policies perform better

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>DAS3</td>
<td>1320</td>
<td>105</td>
</tr>
<tr>
<td>ECT-Perfect</td>
<td>1400</td>
<td>186</td>
</tr>
<tr>
<td>ECT-Last2</td>
<td>4318</td>
<td>3061</td>
</tr>
<tr>
<td>LB</td>
<td>1911</td>
<td>681</td>
</tr>
<tr>
<td>FPF</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Performance of Prediction-Based Grid Scheduling

All policies have similar performance

<table>
<thead>
<tr>
<th>Policy</th>
<th>AUVER</th>
<th>ECT-Perfect</th>
<th>ECT-Last2</th>
<th>LB</th>
<th>FPF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Avg. Response Time [s]</td>
<td>40951</td>
<td>41003</td>
<td>40959</td>
<td>41334</td>
<td></td>
</tr>
<tr>
<td>Avg. Wait Time [s]</td>
<td>6515</td>
<td>6574</td>
<td>6534</td>
<td>6898</td>
<td></td>
</tr>
</tbody>
</table>
Conclusion

- We presented a systematic evaluation of job runtime and queue wait time predictions in grids using **real traces**
  - Simple time-series methods revealed low accuracy
  - Current predictors cannot handle bursty arrivals
  - More accurate predictions do not imply a better performance of grid scheduling

- **Future Work**
  - Simple vs. Complex (AI-based) prediction methods
Questions?

More Information:

• The Grid Workloads Archive: [http://gwa.ewi.tudelft.nl/pmwiki/](http://gwa.ewi.tudelft.nl/pmwiki/)
• DGSim: [www.pds.ewi.tudelft.nl/~iosup/dgsim.php](http://www.pds.ewi.tudelft.nl/~iosup/dgsim.php)
• see PDS publication database at: [www.pds.twi.tudelft.nl/](http://www.pds.twi.tudelft.nl/)

email: o.o.sonmez@tudelft.nl

This work was carried out in the context of the Virtual Laboratory for e-Science project ([www.vl-e.nl](http://www.vl-e.nl)). Part of this work is also carried out under the FP6 Network of Excellence CoreGRID funded by European Commision.