Optimizing User Oriented Job Scheduling within TORQUE

Dalibor Klusáček[1,2], Václav Chlumský[1,2] and Hana Rudová[2]
[1]CESNET, Czech Republic — [2]Faculty of Informatics, Masaryk University, Czech Republic
klusacek@cesnet.cz, vchlumsky@cesnet.cz, hanka@fi.muni.cz

1. Motivation

Many works have proposed the use of planning and metaheuristics when scheduling jobs in HPC or Grid-like environments. In simulations, these approaches show significant improvement over classical queue-based solutions like (prioritized) FCFS or backfilling [1].

However, actual complexity of real-life resource managers and schedulers so far prevented any implementation of such a solution in a production and mainstream (open source) resource manager.

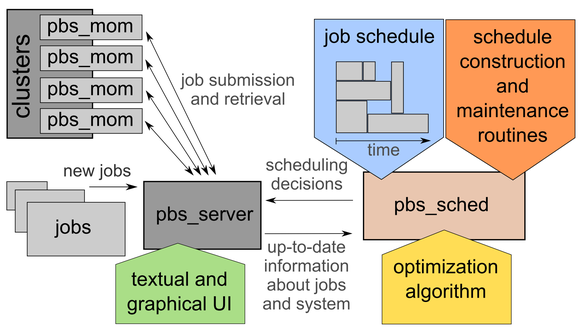
2. Contribution and Overview

Our main contribution is a **new scheduler with an advanced planning functionality** within a freely available TORQUE resource manager. Notably, it offers the following important features:

- **Predictability** (job execution plan)
- **Schedule optimization** using metaheuristic
- Use of **multi resource fairness** policy reflecting heterogeneous jobs' demands
- **Textual and graphical user interfaces**

It is currently **experimentally deployed in the Czech National Grid Infrastructure** *MetaCentrum* [3] and tested by selected users. Soon it will be freely available as an alternative to the current queue-based solution.

3. New TORQUE Scheduler



The developed solution is based on a newly developed TORQUE scheduler (`pbs_sched` module) featuring:

- **Complete job schedule** data structures
- **Schedule construction** (backfill-like) algorithm
- **Schedule optimization** via metaheuristic
- **Maintenance routines**
  - adjust the schedule in time upon dynamic events such as (early) job completions
  - using so called **schedule compression**

4. Periodic Schedule Optimization

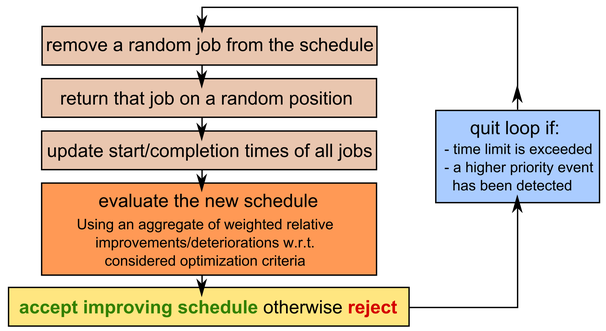
The job schedule is **optimized using a metaheuristic**. Following **performance-related** criteria are minimized:

- **Mean wait time (WT)**
- **Mean response time (RT)**
- **Mean bounded slowdown (SD)**

User-to-user fairness is also optimized by minimizing the mean and the standard deviation of **Normalized User Wait Time (NUWT)**.

- For a given user, NUWT is the total user wait time divided by the amount of previously consumed system resources by that user
- Multiple consumed resources are reflected when calculating NUWT (see the box bellow)

The following figure describes the applied metaheuristic:

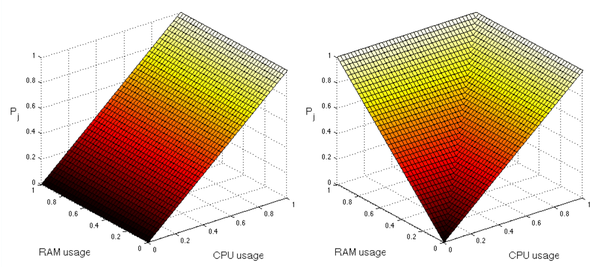


5. Multi Resource Fairness

We use recent methods to **guarantee user-to-user fairness** [2]. Notably, the solution is aware of **heterogeneous jobs' demands** concerning different resources.

- **The schedule is optimized using a fairshare principle**
- A user with **lower resource usage and/or higher total wait time gets higher priority**
- Both **CPU and RAM consumption are used when measuring users' resource consumption**

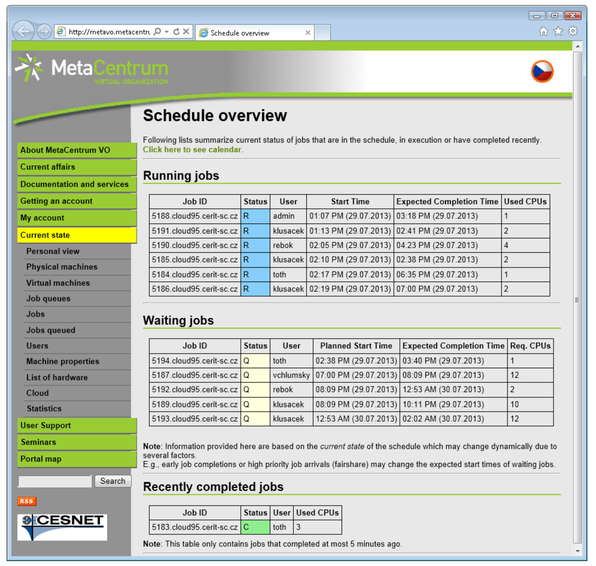
The following figure demonstrates the penalty distribution (Z-axis) according to the relative consumption of available CPU and RAM resources (X and Y-axes) when either single resource-based (left) or multi resource-based (right) fairness is applied.



6. User Interfaces

Two interfaces provide users information about planned start times and expected completion times of their jobs:

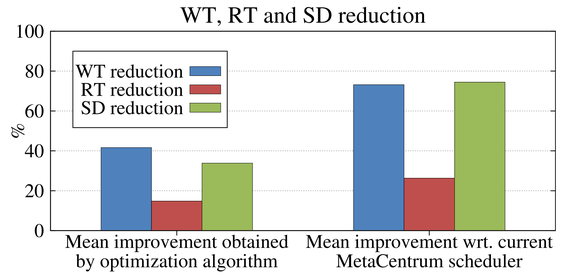
- An **extended** `qstat` **command**
- A **web application** that gets up-to-date job schedule from the `pbs_server` and displays **planned start/completion times of running/waiting jobs**

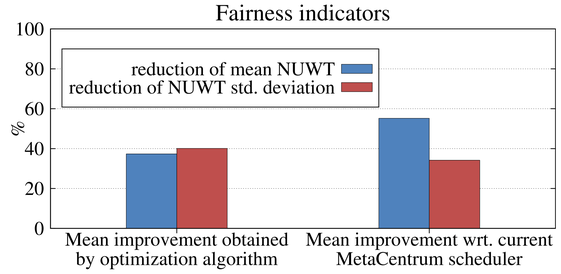


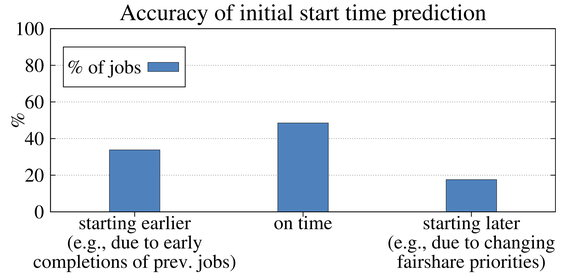
7. Evaluation

Prior to its deployment, our solution was **tested using MetaCentrum workloads**. The following charts show:

- Mean improvement in wait time, response time (WT, RT) and slowdown (SD)
- Mean improvement in fairness indicators (mean NUWT and std. deviation of all NUWT values)
- The accuracy of initial predictions







8. Conclusion and Future Work

We present a unique extension to the widely used TORQUE resource manager that supports:

- **planning** (schedule construction)
- periodic schedule **optimization**
- **multi resource-aware user-to-user fairness**

The system will be further improved as it is currently in the process of **experimental deployment in the MetaCentrum Grid** infrastructure. Last but not least, the system can be freely obtained at: `http://www.metacentrum.cz/docs/pbts`.

Acknowledgments

This work was supported by the grant No. P202/12/0306 of the Grant Agency of the Czech Republic and by the project LM2010005 funded by the Ministry of Education, Youth, and Sports of the Czech Republic.

References

[1] D. Klusáček and H. Rudová. Efficient Grid scheduling through the incremental schedule-based approach. *Computational Intelligence*, 27(1):4–22, 2011.

[2] D. Klusáček, H. Rudová and M. Jaroš. Multi Resource Fairness: Problems and Challenges. In *JSSPP*, 2013.

[3] MetaCentrum, October 2013. `http://www.metacentrum.cz/`.