Comparison of Approaches to the Analysis of Supercomputers Usage Efficiency by the Example of Lomonosov and Lomonosov-2 Supercomputers*

Sergei Leonenkov[1,2] and Sergey Zhumatiy[1]

[1] Research Computing Center of Lomonosov Moscow State University, Moscow, Russia
[2] Lomonosov Moscow State University, Moscow, Russia
{leonenkov,serg}@parallel.ru

Abstract. “Resource planning efficiency” of HPC-systems is usually defined as the utilization of its resources. The number of queued jobs in most modern supercomputer complexes is much bigger than the number of jobs executed at the same moment of time. That high demand and the evolution of widely-used planning algorithms, which can boost utilization up to 0,95 - 1, allow system administrators to more properly manage computational resources and not only meet the needs of cluster owners in maximizing utilization, but also improve customer experience. We conducted a research of the two largest CIS supercomputer systems' (Lomonosov and Lomonosov-2) usage history and proposed a new multi-metrics definition of “resource planning efficiency” concept. In this article, our goal was to compare both approaches and explain why the increased demand for computational resources poses new challenges to the creators of resource planning algorithms and how the proposed approach will improve customer service. Discussed multi-metrics efficiency estimation approach is a part of a bigger project, which aims to provide full jobs scheduling eco-system. We examined general architecture of this environment , which will allow to qualitatively change the system settings of the supercomputer job scheduler on the fly and adapt to the changing flow of jobs.

Keywords: Lomonosov supercomputer · Lomonosov-2 supercomputer · Resource Management · Supercomputer Job Scheduling Efficiency

1 Introduction

In general, supercomputers are expensive and consume a considerable amount of energy. This poses a major problem of the efficiency of their usage. But what is efficiency? In most cases, ”efficiency” means utilization of supercomputer resources, however, our experience suggests that this indicator does not always

* Supported by RFBR (project No. 17-07-00719).

prove to be accurate. On practice, there are nuances that must be taken into account, such as the priorities of individual users or groups, the average size of the job queue, the waiting time for jobs in the queue, and others. The work of a supercomputer depends on the settings of a job scheduler, which are flexible and can be changed by an administrator. The question that arises is how to assess if a supercomputer works effectively at chosen settings. The utilization is no longer a univocal indicator, as there are other factors that have to be considered.

We proposed an efficiency (performance) metric that allows to combine several metrics and carry out a comprehensive assessment of the work of a supercomputer (and job scheduler). In short, the proposed efficiency metric includes several minor metrics that are important from our perspective. It is possible to change the weight of each of them or supplement them by other metrics.

In the article we provided a number of case studies using both the traditional and the new metric, as well as interpretations of the proposed metric values in some specific cases. In Section 2, we considered the features of Lomonosov-1 and Lomonosov-2 supercomputers, which had inspired us to develop the proposed metric. Section 3 discusses traditional and proposed versions of the efficiency metric. In Section 4, we compared these metrics and analyzed the differences. Section 5 describes our plans for further development of the proposed approach.

2 Background

2.1 Lomonosov and Lomonosov-2 Supercomputers

The two core high-performance computing systems of Moscow State University are Lomonosov and Lomonosov-2 supercomputers.

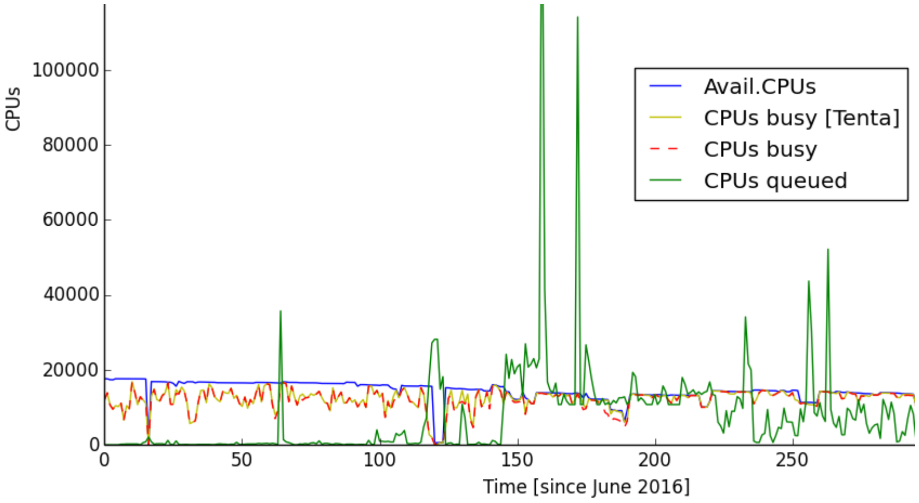

Fig. 1. Lomonosov-2 supercomputer utilization

More than 900 scientific research groups (3,000 active accounts) were provided access to both supercomputers. More than 1,000 jobs are processed every day. SLURM (Simple Linux Utility for Resource Management) and our self-created external scheduler are used to manage all these jobs. To highlight the big workload of the system, let's review the overall Lomonosov and Lomonosov-2 supercomputers utilization performance in the period of 4 years (article [2]). Users submitted more than 820,000 jobs on Lomonosov supercomputer from March 2014 to March 2017; the SLURM native backfill scheduler (article [4]) provided over 0,88 utilization.

Same situation can be found on supercomputer Lomonosov-2, where our external scheduler provided over 0,92 utilization (Fig. 1). Figure 1 shows approximately 300 days of Lomonosov-2 supercomputer usage. Red and yellow lines represent busy CPUs on each day of that period. As it can be seen from Figure 1 overall utilization was less than 0,85 before January of 2017, but after allowing much more users to run their jobs on Lomonosov-2 we have reached our current utilization efficiency.

The other important metric of the job flow for any supercomputer complex is the average time of waiting for queued job to be started. For instance, this parameter is more than 22 hours on Lomonosov supercomputer. Such a busy queue provides a good opportunity to work on improving not only the resource utilization, but also other metrics that will increase the quality of service for supercomputer users.

2.2 Supercomputers Scheduling: Main Terms

Here we introduce several terms that allow us to simplify the description and comparison of both approaches to the evaluation of the supercomputer's scheduling efficiency. Let's set a strip with fixed width H, which shows resources utilization of a computing system in time (H - number of supercomputer's nodes). The strip has an XY coordinate system (X corresponds to time, Y - number of nodes).

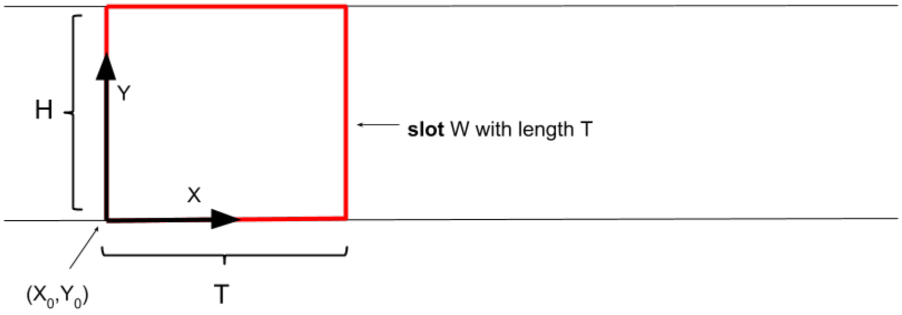

Fig. 2. A strip and W slot

In the strip we set a slot W with length T, which represents a time interval. Slot start coordinate is the coordinate of its bottom left angle (X_0, Y_0) (see Fig. 2). Job is a user's program that has two states: it is either in a queue or is being executed in computing resources.

Definition 1. *Job is a set of elements* $J_i = \{X_i, T_i, H_i, R_i, U_i, Q_i\}$*, where:*

- X_i *- execution start time of a job in computing resources;*
- T_i *- time length of job execution in computing resources;*
- H_i *- number of computing nodes required to execute a job;*
- R_i *- non-empty setup of* j *pairs* (y_{ij}, h_{ij})*, which describes job allocation on nodes as a rectangle with bottom left angle coordinate* (Xi, y_{ij})*,* T_i *execution time and* h_{ij} *number of nodes such that* $\sum_j h_{ij} = H_i$ *(Fig. 3, 4);*
- U_i *- identifier of a user associated with a job;*
- Q_i *- job queuing time.*

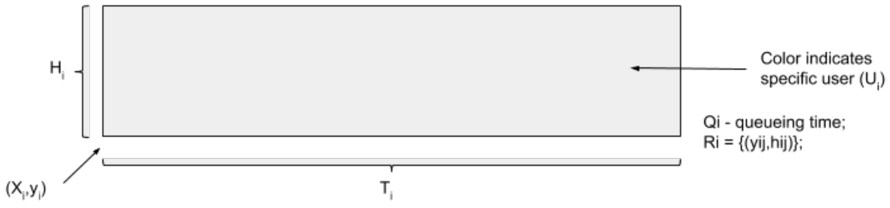

Fig. 3. Job

Notations interpretation: a job is represented as a rectangle with set coordinates of a bottom left angle, defined size (H_i and T_i are rectangle's sizes on Y and X axes respectively) and color (corresponds to user identifier), decomposition of R_i among nodes (Fig. 4).

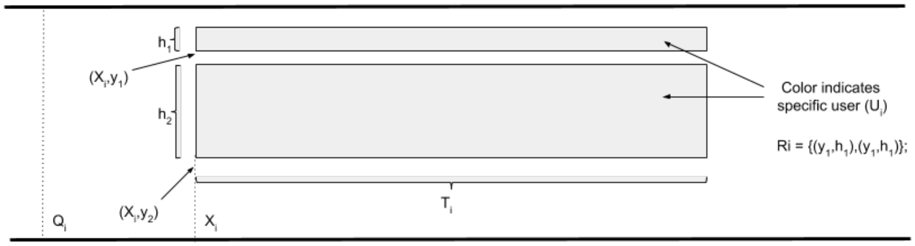
h_1
(X_i,y_1)
h_2
Color indicates specific user (U_i)
Ri = {(y_1,h_1),(y_1,h_1)};
(X_i,y_2)
T_i
Q_i
X_i

Fig. 4. Job J_i in a strip and R_i decomposition example. Both parts of that decomposition are gray, this coloring indicates that these parts corresponds to the same user

More detailed information about used notations (i.e what is jobs packing, packing quality loss function and etc.) can be found in article: "Supercomputer Efficiency: Complex Approach Inspired by Lomonosov-2 History Evaluation" by Sergei Leonenkov and Sergey Zhumatiy in Springer CCIS, article [1] is still in publishing.

3 Evaluation of Supercomputer's Efficiency

3.1 Utilization Approach

The most widely used resource management quality characteristic is the utilization of computing nodes. The main goal of a supercomputer complex is to minimize the idle resources. Until recently, we also used this resources planning efficiency indicator for Lomonosov and Lomonosov-2 supercomputers (as a definition of ”usage efficiency”).

$$Utilization(Z, W) = 1 - \sum_{i=1}^{|Z|}(H_i * (min(T, X_i + T_i) - X_i)/(H * T) \tag{1}$$

In Formula 1 Z is a setup of jobs that was executed on start of slot W or queued during slot W. Let's suppose that sets Z_{start} and Z_{queue} represent executed jobs on slot W start time and queued during whole slot W respectfully.

3.2 Advanced Approach

Basing on Lomonosov and Lomonosov-2 supercomputers usage history, we offered a set of metrics, which allows to consider the task of CPU hours scheduling efficiency more comprehensively, and a formula, which provides a means of comparing different settings of any scheduling algorithms. In addition to the already described Utilization, we also want to use the following metrics: average start time of the first job of users (Formula 2), average start time of jobs belonging to a specific class(Formula 3), number of running jobs (Formula 4) and number of users (Formula 5), whose jobs from Z_{queue} were started in chosen slot W.

$$FUJST(Z, W) = \sum_{u=1}^{UNum} min_{j \subset UJobs(u)}(X_j - Q_j)/UNum(Z) \tag{2}$$

$$AVGST(Z, W, Class) = \sum_{i \subset Class}(X_j - Q_j)/|Class|; \tag{3}$$

$$StartedJobs(Z, W) = (|Z_{start}| + |Z_{queue}| - |Z|)/(|Z_{queue}|) \tag{4}$$

$$StartedUsers(Z, W) = UNum(Z_{start}) + UNum(Z_{queue}) - UNum(Z) \tag{5}$$

Finally, our proposed efficiency formula is representing a weighted sum of 5 chosen metrics (Formula 6). Additional limitation for weights (Formula 7) is created to normalize efficiency value on [0,1].

$$Efficiency = \sum_{i=1}^{5} PriorityCoefficient_i * MetricsValue_i \tag{6}$$

$$\sum_{i=1}^{5} PriorityCoefficient_i = 1 \tag{7}$$

4 Comparison of Two Supercomputer's Efficiency Evaluation Approaches

This section compares two considered efficiency evaluation approaches: utilization-based and multi-metrics.

An important question that will directly influence the multi-metrics efficiency function is the right choice of weights (priority coefficient). To use this approach each supercomputer complex has to configure it independently. It is impossible to find a universal set of weights for all complexes, as each owner of such system has a unique flow of jobs launched by the clients, and sets his own narrow goals when using the system. For example, now the utilization of the processor time is the cornerstone in the management of HPC-systems, so it is not correct to set the same coefficients for this metric and the other, as other metrics are more prone to volatility. Setting a pair of new jobs for execution can significantly shift the "efficiency" in one direction while the utilization remains unchanged.

All the cases that we examined in this article are considered using model examples of sets of weights. In view of the complexity of interpreting the values of the multi-metrics efficiency function, we will use sets of weights, where three of the five weights are equal to zero.

4.1 Utilization Efficiency Metric on Today's Supercomputers Workloads Does not Show the Quality of Customer Experience

Modern supercomputer resource planning techniques and algorithms have already achieved significant results in maximizing utilization. The usage history of the two MSU facilities shows that the maximum possible utilization was not achieved because of the factors that have no connection with the quality of the algorithms, such as reservations for allocated accounts, system failures, etc. On the other hand, there are other examples. Let's suppose that the computing field of the supercomputer is 100 percent occupied. When a certain amount of resources is released, the scheduler needs to decide which job from the queue can be added to this location. Let's say there are two identical jobs launched by two different users at different times, but one already has jobs on the account, and the other does not. Usually, the scheduler, tuned to maximize utilization, will launch the one that was queued before. The planner, which tries to maximize our multi-metrics function with given weights, will launch the job, the author of which does not yet have jobs on the account. An example of both scenarios is given in Fig. 5.

But what is the difference? The utilization for both launch scenarios is the same, but in the case of the multi-metrics efficiency function, we get more users whose jobs are on the account, which means that on average users will receive the first results of their calculations faster. Thus, not only the owners of supercomputer systems, who spend a large amount of resources to support their work, are satisfied, but also individual users, who can quickly move from waiting for the first results to their analysis.

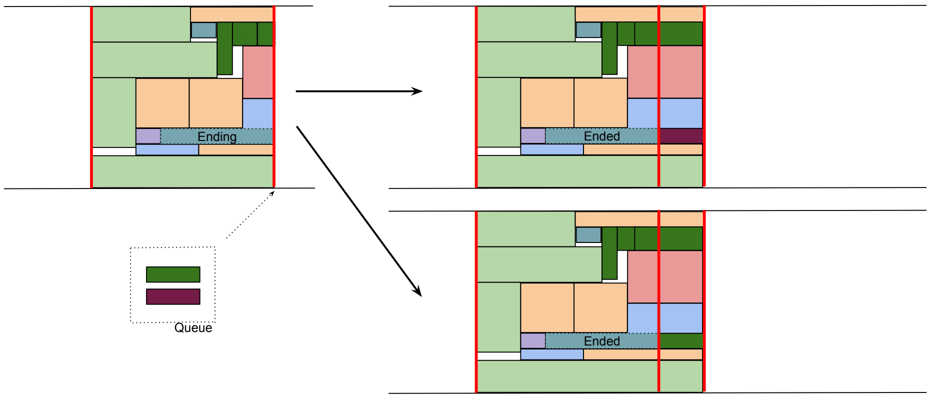

Fig. 5. Utilization efficiency metric on todays supercomputers workloads does not show quality of customer experience

4.2 The First User's Job Start Time Metric Should be Used for Managing Faster Access to First Calculations Results

We have already reviewed multi-metrics resource planning efficiency based on utilization and number of users, whose jobs are being executed at a given moment of time, (with weight equal to 1/2 each). Let's now discuss this efficiency function but based on utilization and average first user's job start time metric (with weight equal to 1/2 each). This choice of metrics follows a similar scenario, like the previous one, but the start time of the first user's jobs is an additional parameter, which the scheduler have to take into account in order to achieve optimal job planning.

This additional metric controls the location of the user's first jobs in the queue, shifting them all to the very beginning of the current queue regardless of their queue time. All this cannot be tracked using only utilization.

4.3 Large Jobs Start Time Problem

Another problem that was noticed by the system administrators at the RCC MSU is that, when achieving the highest system utilization rates, the scheduler sometimes abuses large size jobs (these jobs move in the queue much more slowly than jobs of smaller sizes). This effect arises due to the fact that the scheduler is trying to fill all available empty nodes of the system with small jobs. To cope with this significant problem Chebyshev supercomputer managing policies contained special day (each Thursday), when all accumulated large jobs were given highest priority to start execution and all smaller jobs - lowest priority. We strongly believe that there is no need in such unclear for users optimizations and our proposed set of metrics can help scheduler to cope with described problem. To solve this type of efficiency planning problem, we have added to the general list the average start time of jobs of a specific class metric, where class can be defined as a class of jobs with a size from a certain interval. Additionally, this metric can be used to boost a specific class, for example: jobs from specific group of users, jobs with specific programming package or even jobs from specific user.

4.4 Multi-metrics Approach with Utilization Weight Equal to 0

As we have already mentioned, the selection of weights in multi-metrics efficiency evaluation approach is a challenging task. We have reviewed three different convolutions and sets of weights (Sections 4.1-4.3). Each of the efficiency calculation formulas included utilization metrics. But what goes wrong if the utilization weight is set to 0? Let's discuss the efficiency function based on the number of users whose jobs are executing and average first user's job start time metrics (with weight equal to 1/2 each). The optimal algorithm for the scheduler will be to set only the first job of each user and no longer place any jobs for execution in order, so that if a new user appears in the queue, he could immediately get to the execution, thus retaining the optimal value of the effectiveness. In this regard, the availability of utilization metric in determining the efficiency of supercomputer resource planning is vital.

5 Future Work

At the heart of the future work is the desire to create a recommendation system that will advise the system administrator on changing the current scheduler settings [3] in order to maximize proposed multi-metrics efficiency function. The general architecture of such system is presented on Fig. 6.

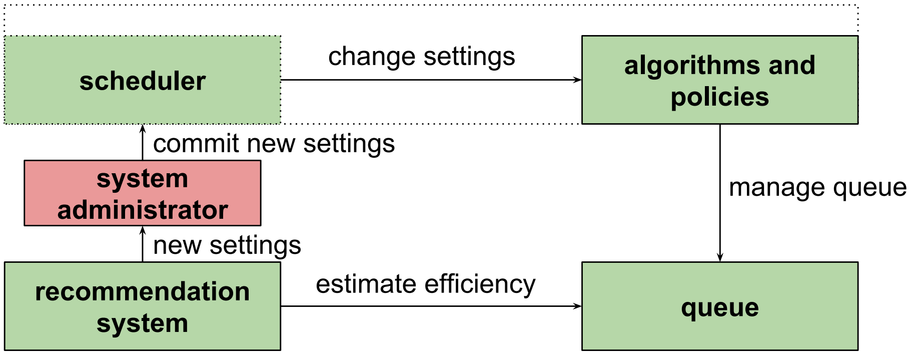

Fig. 6. Recommendation system architecture

Subsequently, this system should evolve into an autonomous cluster management system.

Acknowledgements

This material is based upon the work supported by Russian Foundation for Basic Research (project No. 17-07-00719). The research is carried out using the equipment of the shared research facilities of HPC computing resources at Lomonosov Moscow State University [5, 6].

References

1. Leonenkov S., Zhumatiy S.: Supercomputer Efficiency: Complex Approach Inspired by Lomonosov-2 History Evaluation. Springer CCIS (2018)
2. V. Sadovnichy, A. Tikhonravov, Vl. Voevodin, and V. Opanasenko: "Lomonosov": Supercomputing at Moscow State University. In Contemporary High Performance Computing: From Petascale toward Exascale (Chapman and Hall/CRC Computational Science), pp.283-307, Boca Raton, USA, CRC Press, 2013.
3. Leonenkov S., Zhumatiy S.: Introducing New Backfill-based Scheduler for SLURM Resource Manager, Procedia Computer Science. Volume 66, 2015, (pp 661-669).
4. SLURM Homepage, https://slurm.schedmd.com. Last accessed 14 September 2018
5. Lomonosov-2 supercomputer on TOP50 list, http://top50.supercomputers.ru/. Last accessed 14 September 2018.
6. Lomonosov — T-Platforms, http://www.top500.org/system/177421. Last accessed 14 September 2018.