

SEARCH FOR ANOMALIES IN THE COMPUTATIONAL JOBS OF THE ATLAS EXPERIMENT WITH THE APPLICATION OF VISUAL ANALYTICS

M.A. Grigorieva^{1,a}, **A.A. Alekseev**^{3,b}, **T.P. Galkin**^{2,c},
T.A. Korchuganova^{3,d}, **I.E. Milman**^e, **V.V. Pilyugin**^{2,f}, **M.A. Titov**^{1,g}
on behalf of ATLAS Collaboration

¹ National Research Centre «Kurchatov Institute», Moscow, Russian Federation

² National Research Nuclear University "MEPhI", Moscow, Russian Federation

³ National Research Tomsk Polytechnic University, Tomsk, Russian Federation

E-mail: ^a maria.grigorieva@cern.ch, ^b frt@tpu.ru, ^c z@wqc.me, ^d tatiana.korchuganova@cern.ch,
^e igal.milman@gmail.com, ^f VVPilyugin@mephi.ru, ^g mikhail.titov@cern.ch

ATLAS is the largest experiment at the LHC. It generates vast volumes of scientific data accompanied with auxiliary metadata, representing all stages of data processing, Monte-Carlo simulation, properties of detector and computing environment. Terabytes of metadata was accumulated by the workflow and data management, and metadata archiving systems. These metadata can help physicists carrying out studies to evaluate in advance the duration of their analysis jobs. As these jobs are executed in a heterogeneous distributed and dynamically changing infrastructure, their duration varies across computing centers and depends on many factors. Ensuring the uniformity in job execution requires searching for anomalies and analyzing the reasons of non-trivial job execution behavior to predict and avoid the recurrence in future. Detailed analysis of large volume of job execution benefits from application of machine learning and visual analysis methods. The approach of visual analytics technique was demonstrated on the analysis of jobs archive. The proposed method allowed to identify computing sites having non-trivial job execution process, and the visual cluster analysis showed parameters affecting or indicating possible time delays. Further work will concentrate on increasing of the amount of analyzed jobs and the development of interactive visual models, facilitating the interpretation of analysis results.

Keywords: visual analytics, machine learning, data analysis, anomalies, non-trivial

© 2018 Maria Grigorieva, Aleksandr Alekseev, Timofei Galkin, Tatiana Korchuganova, Igal Milman, Victor Pilyugin, Mikhail Titov

1. Introduction

ATLAS is the largest experiment at the LHC, it generates a great amount of data and metadata and utilizes a great variety of computing resources, like WLCG, HPC, Academic and university clusters and volunteer computers [1]. The main entities in ATLAS computing are tasks and jobs. Task contains execution code, input and output files, corresponding to underlying physics process and initial conditions. However, so many events are being produced within a task that, for practical reasons, each task is fragmented in jobs, which correspond to a fixed number of events. For the last decade ATLAS has processed over 10 millions of physics analysis tasks and 3 billions of jobs. The amount of data keeps growing and soon will reach the exascale level. At the same time, there is a constant increase in size and complexity of the distributed computing infrastructure.

Large-scale distributed systems, like in ATLAS computing, face the following challenges: big diversity and complexity, highly dynamic computing environments, ongoing competition for computing resources among different threads of computing jobs, complex workflows and workloads, uncountable possible reasons of failures and time delays. All these challenges increase the complexity of the data management architecture and make it difficult to predict periods of system's maximum load, and the probability of system failure.

An ultimate goal is to increase the stability and efficiency of the distributed data processing and analysis systems. The first step is to analyze the job execution processes to figure out trivial and non-trivial behavior and their possible reasons. We are solving this, applying analysis methods from statistics and machine learning, for detection of disruptions of jobs execution process. In this work we propose to extend these methods, and get the benefits from the interactive visual analytics, providing the use of dynamic and static spatial interpretations of analyzed data, with the help of human strong cognitive possibilities.

2. Traditional Data Analysis Workflow

Multidimensional data analysis usually implies the usage of machine learning methods, which help to categorize, cluster, associate or correlate the data. But typically, domain experts (the end-users of the data analysis), have limited involvement in the process of data analysis. In the traditional machine-learning workflow the domain-experts involvement is limited to providing data, answering domain-related questions, or giving some feedback about the model. This kind of iterative interaction, instead of a cooperative one, may not be effective. So, the data analysis process itself becomes long and complex, with a lot of asynchronous iterations. An implementation of visual platforms as an integration of machine learning algorithms with interactive visualization gives the experts the ability to interact directly with the data and models [2]. In case of ATLAS metadata the domain-experts involvement in the data analysis is crucial, because of the exceptional multidimensionality and complexity of the data as well as the presence of peculiar qualities, known only by experts.

3. ATLAS Data Sources and Job Execution Metrics

ATLAS data sources that may be useful in the analysis of job execution are listed below to show the complexity and level of dimensionality [3,4].

- Rucio (Distributed Data Management System)¹ provides information about the storage usage (total size, used space, free space and expired space) of each endpoint.
- NWS (Network Weather System)² provides information about network state between nodes.
- AGIS (ATLAS Grid Information System)³ stores the characteristics of sites and queues.
- MemoryMonitor⁴ service - I/O metrics.

¹ <https://rucio.cern.ch/>

² <http://atlas-adc-netmetrics-lb.cern.ch/>

³ <http://atlas-agis.cern.ch/agis/>

⁴ <https://twiki.cern.ch/twiki/bin/viewauth/AtlasComputing/IOMonitoring>

- ProdSys2/PanDA (Workload Management System) [5] contains DEFT and JEDI components, stores the information about tasks, jobs, and other components.
- DKB (Data Knowledge Base) [6] provides the metadata integration from multiple sources.

Data from these sources was partly transferred to the ElasticSearch storage and currently it is used for the purpose of data analysis.

According to these data sources, job execution metrics can be divided into 4 groups: application, middleware, resource and network-level metrics. Investigation of such complex data, having over 200 features, is not a trivial task, which requires the tight connection between data analysis methods and expert opinion.

4. The method of visual analysis of multidimensional data

To analyze job execution process we propose to use geometric representation of data. Initial data is presented in a tabular form. Rows of the table correspond to multidimensional points in the space, and the values of metrics are the coordinates of these points. The distances between points in multidimensional space are calculated as Euclidean or Mahalanobis distances. Then the points are projected to 3-dimensional space and drawn as spheres. If the distance between the points is less than the threshold, given by the analyst using the interactive interface, then a cylinder is constructed to connect the spheres. The color of the cylinder simulates the distance between the points from red (small distance) to blue (long distance). The resulting set of spheres and cylinders forms a spatial scene with a given geometry and optical (color) characteristics.

4.1. IVAMD (Interactive Visual Analysis of Multidimensional Data) Prototype

In this project we used the software prototype of multidimensional visual analysis – IVAMD. It's based on Autodesk 3DSMax with a combination of maxscript scripts and C# modules. Depending on the amount of memory, the software can handle up to a couple of hundreds of objects. Spheres in clusters are coded with different colors. And the prototype allows interactive work with the spatial scene. We can rotate, change the scale of image, click at the spheres to get their names and coordinates. And the results can be exported to excel (xlsx) files [7]. Current prototype uses the standard 3DSMax color scheme, which will be changed in future.

5. The analysis of job execution

5.1. Trivial and not-trivial job execution process

At first, we must understand what is the trivial and non-trivial job execution behavior and make a hypothesis about it. We analyzed all finished jobs of one computing task and observed the matching of the distributions of execution time (timeExe) and CPU time for most of computing sites. We suggest that this could be a sign of trivial behavior. Then non-trivial behavior may be determined by the difference of the CPU and execution time distributions (example of non-trivial job execution on site 2 is shown on Figure 1). For the CPU time the distribution is between 2 to 6 minutes. But the execution time fluctuates a lot from several minutes to 7 hours. We decided to analyze the possible reasons of such behavior.

5.2. Analysis of non-trivial job execution on computing site

Jobs executed on one computing site were analyzed. We took only jobs, belonging to one task, to ensure that all of them have the same execution code and input data. The number of jobs in data sample is ~1900. At the beginning we chose only numerical metrics from jobs_archive-* index from the ElasticSearch instance at Chicago university. It had over 50 parameters. To reduce the set of relevant metrics to a humanly manageable one without losing much information, all features with a high percentage of missing values, collinear (highly correlated) features and features with a single unique value have been removed.

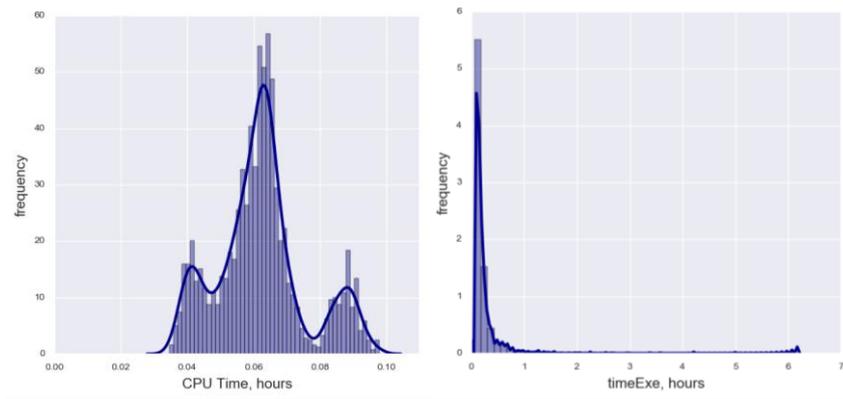


Figure 1. Illustration of non-trivial jobs execution by the difference between the distributions of CPU Time and execution time on computing site (timeExe)

The next step is the construction of interactive visual representation of multidimensional data. To avoid overplotting on the resulting spatial scene with spheres and cylinders, the number of rows in the initial data sample should be reduced to several hundreds (in our case we've chosen 200). To archive this, K-means clustering was applied to split dataset into 200 data clusters. Then initial data was grouped by clusters with mean values of all features. IVAMD prototype was used to build 3-dimensional spatial scene (current projection WallTime – WorkDirSize– IObytesRead is shown on Figure 2), interactive interface allowed to tune the distance threshold iteratively, so we could watch changing of cluster structure and the appearance of anomalous points.

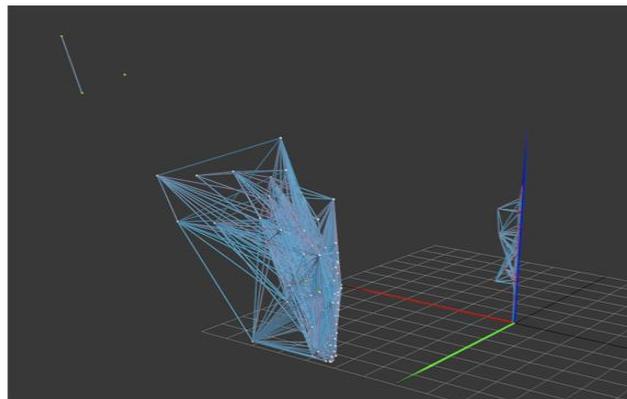


Figure 2. 3-dimensional spatial scene built using IVAMD prototype

Two clusters can be located on the resulting spatial scene: large cluster with the average wall time of 25 minutes (we suggest that this cluster illustrates the trivial behavior), small cluster with the average wall time of 10 minutes, and irregular points with a very high wall time (227 minutes ~ 4 hours).

5.3. The results of job execution analysis

All available metrics of 2 clusters and irregular points were analyzed on the initial data sample. Results are presented in Table 1. All values are calculated as mean of all metrics for clusters and irregular points. The WallTime values vary greatly from 10 minutes to 4 hours, but the CPU time is in the expected range for all jobs. Staging timings metrics are widely spread, but negligible for the wall time. The amounts of RAM and virtual memory are almost the same for all jobs. Input and output file sizes are 300 and 600 Mb respectively for all clusters and points. Written data (IObytesWritten) is close to the output file size. But we observed that input data (IObytesRead) are much larger than input file sizes (6 times larger for normal cluster and 10 times larger for irregular points). Possible reasons could be that jobs executed on the same site at the same time lead to an overload of the data streams. The read/write rates of irregular points are 5 times slower in comparison with the large cluster. And the small cluster has the highest rates of data read/write (twice larger than in normal cluster) and the

shortest wall time. Probably it can be connected with the workDirSize (size of the directory on the endpoint), which is only 8 MB for this cluster, unlike large cluster where its size is 600 MB.

Table 1 – The results of cluster analysis

Feature Name	Large Cluster	Small Cluster	Irregular Points
WallTime	25 min	10 min	227 min
CPUTime	3,8 min	3,2 min	3,5 min
TimeStageIn	370 sec	110 sec	356 sec
TimeStageOut	59 sec	33 sec	680 sec
MaxRSS	825 MB	817 MB	814 MB
MaxVmem	3 041 MB	2 767 MB	3 056 MB
IObytesWritten	531 MB	488 MB	568 MB
IObytesRead	1 957 MB	1 704 MB	3 029 MB
WorkDirSize	600 MB	8 MB	600 MB
IObytesReadRate	3,868 MB/sec	6,716 MB/sec	0,690 MB/sec
IObytesWriteRate	1,068 MB/sec	1,945 MB/sec	0,180 MB/sec

6. Conclusion

As a result of current research the methodology of data analysis with the combined usage of machine learning and interactive visual analytics was proposed. This methodology was demonstrated using the IVAMD prototype for the analysis of job execution data in the ATLAS experiment. Our work showed that the method of visual analytics can be successfully applied to the analysis of ATLAS metadata. In the near future we are going to increase the size of investigated meta-information to obtain more representative data samples. Currently we are using only numerical metrics, but there are a lot of categorical values, which also have to be analyzed. At the first stage of the work only one data source was used. Currently we are working on adding other data sources, like AGIS or NWS, that provide the information about sites and network status during job execution. The development of the visual analytics tools includes the implementation of a web-compatible prototype and its integration in the ATLAS Monitoring System.

7. Acknowledgements

This work has been supported by the RSCF grant No. 18-71-10003.

References

- [1] Aad G. et al. [ATLAS Collaboration]. The ATLAS Experiment at the CERN Large Hadron Collider // JINST 2008, vol.3, p.S08003
- [2] Aggarwal C., Reddy C. Data Clustering: Algorithms and Applications // CRC Press 2014
- [3] Grigorieva M. et al. Evaluating non-relational storage technology for HEP metadata and meta-data catalog // Journal of Physics: Conference Series 2016, vol.762, no.1, p.012017
- [4] Grigorieva M. et al. Knowledge base for Scientific Experiment // Open Systems. DBMS 2016, vol.24, no.4, pp.42-44 (in Russian)
- [5] Barreiro F. et al. PanDA for ATLAS distributed computing in the next decade // Journal of Physics: Conference Series 2017, vol.898, no.5, p.052002
- [6] Kaida A. et al. Development of DKB ETL module in case of data conversion // Journal of Physics: Conference Series 2018, vol.1015, no.3, p.032055
- [7] Milman I. et al. Interactive Visual Analysis of Multidimensional Geometric Data // Vaclav Skala-UNION Agency 2016, pp.233-238. ISBN:978-80-86943-58-9