Visualization and clustering of computational cluster delays in 3 dimensional analytical space

Salnikov A.N., Begaev A.A., Maysuradze A.I.

Lomonosov Moscow State University

salnikov@cs.msu.ru,akagitsunesan@gmail.com,maysuradze@cs.msu.ru

https://github.com/clustbench

Introduction

Modern supercomputers are used for solving a huge spectre of computational and research tasks: solving sophisticated systems of equations, mathematical modeling and processing huge amounts of data. These tasks are solved by parallel algorithms and programs. Calculations are processed on a large number of computational cluster nodes. During their operations huge volumes of information are transmitted via internal network of supercomputer, containing switches, wires and network cards. These components are combined in regular manner (intercommunication topology), which could not be documented by the vendor of computational cluster.

Delays

The transfer of data between the nodes consumes some time, called delay. Often it is assumed that the delay depends only on the number of network components, which the data will pass during the transmission and it is constant. But in practice, this model doesn't represent the influence of delays on the performance of parallel programs. Physical state of communication environment (malfunctioning components) and hidden features of network topology should be considered.

Testing method

Authors propose to test a supercomputer for researching its communication environment in special way. Developed testing application initializes the transfers of data between the nodes of supercomputer. The transmitted data called message. Messages having varying lengths are sent between all nodes of supercomputer in different modes, potentially emulating the real appearing transfers of data during the work of parallel program. In this paper authors tested computational clusters Lomonosov and K60 (Keldysh Institute of Applied Mathematics) in "one to one" mode, which conclude in consistent transmission of data between the nodes.

Problem with manual analysis of delays

Values of delays are described by three parameters: number of "sender" node, number of "receiver' node, length of message, which was sent. The analytical space appears to be 3 dimensional. The large number of supercomputer nodes and large number of messages' length provide us a giant dataset of delays, which is very hard to analyze manually and take a significant amount of memory to store them.

Network components of supercomputer and settings of calculations distribution are maintained by the system administrators of supercomputers. It is necessary to provide them special tools, which could ease their routines, such as revealing the malfunction components or picking optimal settings, depending on undocumented features of supercomputer topology.

Structure of delay behavior and similarity

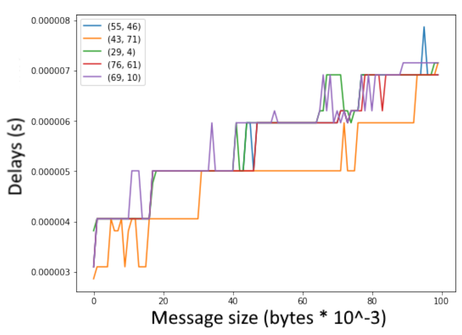


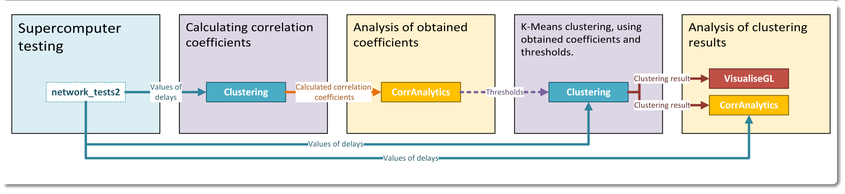
Fig. 1. Visualisation of delays for random pairs "sender"-"receiver" for K-60 supercomputer

Authors noticed that the behavior of delays between some nodes of supercomputer is identical. Also, there are features for behavior:

- Step – change of delay after which, the delay keeps the same as the message length increases.
- Peak – significant change of delay, but it goes back for the next size of message length.

"Steps" are more important for our research, because they most accurately show the behavior of delays. "Peaks" influent on calculating the "steps", so, their number should be minimal for more accurate results.

Application workflow



Proposed method

Authors proposed a method, which uses correlation analysis, for revealing "steps" and "peak" in structures of delays. The "peak" in k-th position is modeled by unit vector $y^{(k)} = (0, ..., 1, ..., 0)$, where $"1"$ stays in k-th position. So, for looking up for "peaks" the Pearson's correlation coefficients are calculated between the set of proposed model vectors (for gaining this set the k is varied from 0 to l – the maximum message length) and vector of delays m between all tested nodes. For gaining the information about "steps" locations the calculation of correlation coefficients stays the same, but the Δm vector, where $\Delta m_i = m_{i+1} - m_i$, is used instead the m vector in calculation of correlation coefficients.

$$corr^p{}_k = \frac{m_k - \overline{m}}{\sqrt{\sigma^2{}_{y^{(k)}}}\sqrt{\sigma^2{}_m}}, k \in [0, l-1] \quad (1)$$

$$\sqrt{\sigma^2{}_{y^{(k)}}} = \frac{n-1}{n^2}, \sqrt{\sigma^2{}_m} = \overline{m^2} - \overline{m}^2 \quad (2)$$

$$corr^j{}_k = \frac{\Delta m_k - \overline{\Delta m}}{\sqrt{\sigma^2{}_{y^{(k)}}}\sqrt{\sigma^2{}_{\Delta m}}}, k \in [0, l-1] \quad (3)$$

$$\sqrt{\sigma^2{}_{y^{(k)}}} = \frac{n-1}{n^2}, \sqrt{\sigma^2{}_{\Delta m}} = \overline{\Delta m^2} - \overline{\Delta m}^2 \quad (4)$$

For decreasing influence of "peaks" on calculation of "steps" correlation coefficients median filtration is performed. For detecting the "step" or "peak" the **threshold** should be chosen. This threshold is chosen imperative, looking on visualisation of delays between random pairs of nodes "sender"-"receiver". All values of correlation coefficient, which will be above the threshold will be treated as feature of behavior, depending on set of correlation coefficients. Finally, after calculating all correlation coefficients for all messages lengths and all nodes the clustering by **K-Means** algorithm is performed, using calculated data. The number K for K-Means is chosen according to SSE metric.

Found features of behavior for K-60 supercomputer

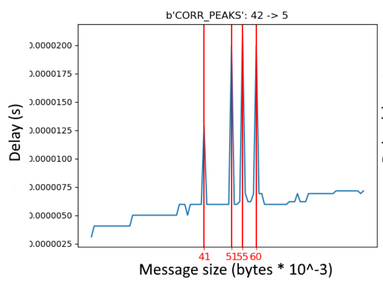


Fig 2. "Peaks"

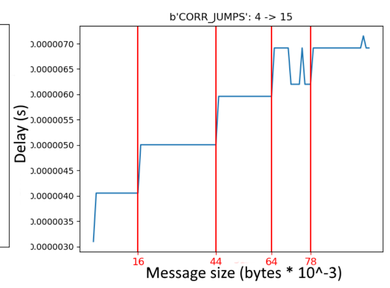


Fig 3. "Steps"

For K-60 supercomputer threshold was chosen equal 0.25, so, using this number we visualised "peaks" and "steps" for fixed pair of nodes for K-60 supercomputer.

Revealed structure of delays for K-60 supercomputer

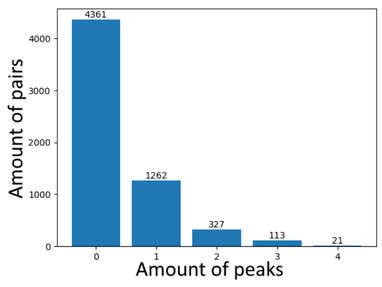


Fig 4. Pairs containing "peaks"

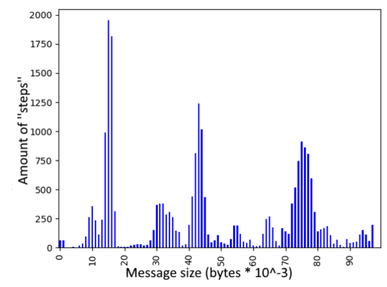


Fig 5. Distribution of "steps"

As it was mentioned above, "peaks" have strong influence on calculating "steps". So, we can see on fig. 4 the amount of "peaks" for most pairs equals 0 or 1, so the calculating the "steps" will be accurate for K-60. According to the fig. 5 the distribution of "steps" has several message lengths where their density is high, so, the clustering can be performed.

Clusters found for K-60 supercomputer

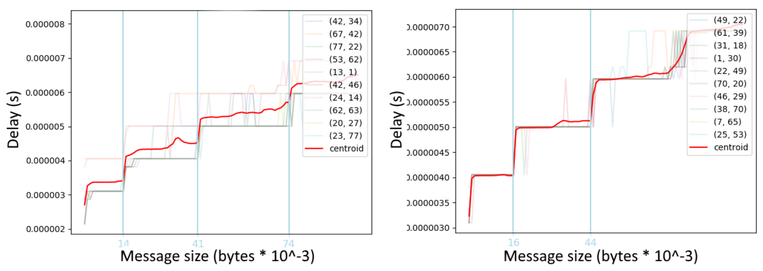


Fig. 6, 7. Plots of 2 clusters found

As it can be seen from plots above - the calculated clusters represent the same behaviors of delays in each cluster. The red line shows centroid of cluster, which can be used to represent the behavior of delay for each cluster. Storing only information about clusters: pairs "sender"-"receiver", which are included into it and centroid the amount of data can be decreased. So, the amount of storing data was decreased from 4754Kb to 99Kb in 48 times.

3D Visualisation of found clusters with their features

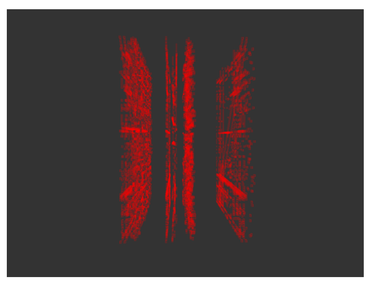


Fig. 8. 3D Visualisation of steps found in clusters for K60 supercomputer

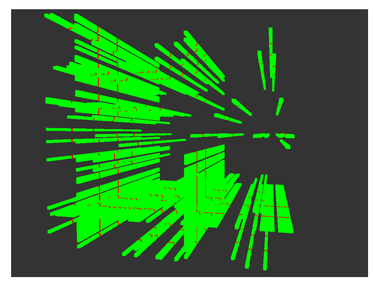


Fig. 9. 3D Visualisation of one cluster found for K60 supercomputer

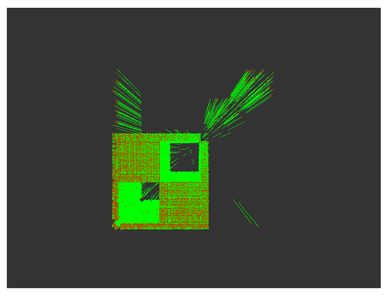


Fig. 10. One of clusters found for Lomonosov-1.

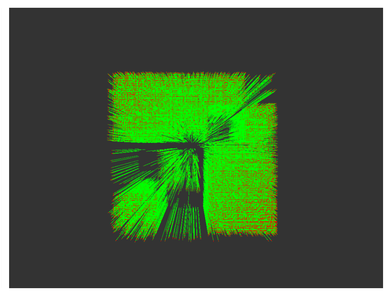


Fig. 11. One of clusters found for Lomonosov-1.

For K-60 supercomputer the most meaningful visualisation was plotting the "steps" for all clusters in 3D. All "steps" combine in several planes, which tells us about homogeneous topology of message delivery of K-60 supercomputer. Also, the same procedure was made for dataset collected for Lomonosov-1 supercomputer. Found clusters for Lomonosov-1 are combined in cells, which are connected to information about delivery topology of this supercomputer.

Conclusion

It was revealed, that the structure of delays contain small amount of "peaks" and obtained clusters have jumps in common messages sizes, which tells us, that proposed method works for current dataset. But there are another delays behavior structures exist. They appear if supercomputer will be tested in different modes. The authors want to propose new model for revealing and analysing these structures in future works. Also, the developed solution should be integrated in ***clustbench*** pack of tools for testing the supercomputer interconnect.

Bibliography

[1]Clustbench – HPC cluster benchmarking toolkit: URL: https://github.com/clustbench, last accesed: 05.09.2018.

[2]Salnikov A.N., Andreev D.Yu, Lebedev R.D., "Toolkit for analyzing the communication environment characteristics of a computational cluster based on MPI standard functions" in Moscow University Computational Mathematics and Cybernetics, vol. 36, no. 1, US: Allerton Press Inc., 2012, pp. 41–49.

[3]D. York, "Least squares fitting of a straight line with correlated errors" in Earth and Planetary Science Letters, vol. V, Elsevier, 1968, pp. 320–324.

[4]B. S. Everitt, S. Landau, M. Leese, and D. Stahl, Cluster Analysis, 5th ed., UK: Wiley, 2011.

[5]J. Cohen, P. Cohen, S. G. West, and L. S. Aiken, "Applied Multiple Regression/Correlation Analysis for the Behavioral Sciences," Mahwah, New Jersey: Lawrence Erlbaum Associates, 2003.