Statistical Properties of Task Running Times in a Global-Scale Grid Environment

Menno Dobber
Vrije Universiteit,
De Boelelaan 1081a,
1081 HV Amsterdam,
The Netherlands
E-mail: amdobber@few.vu.nl

Rob van der Mei
CWI and Vrije Universiteit,
c/o Kruislaan 413,
1098 SJ Amsterdam,
The Netherlands
E-mail: mei@few.vu.nl

Ger Koole
Vrije Universiteit,
De Boelelaan 1081a,
1081 HV Amsterdam,
The Netherlands
E-mail: koole@few.vu.nl

Abstract*— Grid computing technology connects globally distributed processors to develop an immense source of computing power, which enables us to run applications in parallel that would take orders of magnitude more time on a single processor. Key characteristics of a global-scale grid are the strong burstiness in the amount of load on the resources and on the network capacities, and the fact that processors may be appended to or removed from the grid at any time. To cope with these characteristics, it is essential to develop techniques that make applications robust against the dynamics of the grid environment. For these techniques to be effective, it is important to have an understanding of the statistical properties of the dynamics of a grid environment. Today, however, the statistical properties of the dynamic behavior of real global-scale grid environments are not well understood. Our main focus is on highly CPU-intensive grid applications that require huge amounts of processor power for running tasks. Motivated by this, we have performed extensive measurements in a real, global-scale grid environment to study the statistical properties of the running times of tasks on processors. We observe (1) a strong burstiness of the running times over different time scales, (2) a strong heterogeneity of the running-time characteristics among the different hosts, (3) a strong heterogeneity of the running-time characteristics for the same host over different time intervals, and (4) the occurrence of sudden level-switches in the running times, amongst others. These observations are used to develop effective techniques for the prediction of running times. They can be used to develop effective control schemes for robust grid applications.

Acknowledgment

The authors would like to thank Mathijs den Burger, Thilo Kielmann, and Henri Bal, for their useful comments.

I. Introduction

Over the years, clusters of processor units have evolved into grids that globally connect processors via the Internet. Generally, the processors of grids are shared by the different grid applications that run on them. Grid environments are fundamentally different from clusters and are highly unpredictable for several reasons: the processor load on fluctuates all the time, the processor capacities and link rates are often unknown, users can connect and disconnect processors at any time. As a consequence, applications that perform well in a cluster environment may perform badly when executed in a grid environment. This raises the need for techniques that make applications robust against the dynamics of grid environments. For example, an effective means to do so is to implement dynamic load balancing (DLB) schemes that can dynamically update the load offered to different nodes in a grid in response to changing circumstances. The efficiency of such control schemes strongly depends on the effectiveness of prediction schemes, which in turn require an understanding of the statistical properties of the dynamics of a grid environment. In addition, understanding the characteristics of a grid is also extremely useful for performing simulations or computations to assess the effectiveness of control strategies *prior* to their creation. This enables us to select the most promising control schemes to be subjected to extensive experimentation, thereby saving a tremendous amount of time and cost involved in redundant and time-consuming experimentation with ineffective control strategies. Our main focus is on CPU-intensive applications that require large amounts of processing power for running tasks. Therefore, in this paper, we focus on the characteristics and statistical properties of the running times of tasks.

In the literature, a significant number of papers have been devoted to data analysis of different properties of grids or networks. Three types of grid-property investigations can be distinguished: (1) a complete focus on the investigation of one grid property: for example, the statistical characteristics of network arrivals [1], of availability [2], and of load [3]; (2) an exploration on the statistical properties of a grid property and followed by simulation studies to address different types of questions, varying from grid design questions (e.g., is a global grid feasible?) to basic questions (e.g., what scheduling strategies are needed?). Examples are the investigations on the statistical properties of life times of unix processes to develop load balancing strategies [4], [5], and investigations on a characterization of the availability of desktop grids to explore the affection on its utility [6], and (3) research on the statistical characteristics of a grid property to develop a prediction method: those research steps have been done for load to predict total run times of applications [7], for load to predict the load [8], and for the throughput to predict the throughput [9], [10]. Consequently, the final step is to use these predictions to develop a dynamic load-balancing or scheduling algorithm (see [11] with its previous papers [3], [8] and [12], and [13] in combination with its previous paper [14]). Despite the fact that many papers focus on the statistical charateristics of grid properties, no papers concentrate on the running times of consecutive tasks on shared processors of a grid.

Extensive research has been done on the relation between processor load and the running times by Dinda et al. [12], [15]. Although these two factors in theory are closely related, they find that in practice it is hard to relate the running times and the load, because many other factors (e.g., memory space) also have an influence on the running times. For that reason, load and running times of tasks may have strongly different characteristics, and it is necessary to investigate the running time characteristics.

In this paper, we extensively investigate the statistical properties of consecutive tasks on nodes in a grid environment. To this end, we gather real datasets of running times of tasks on distributed processors in a global-scale testbed environment. How we collected the data is described in Section II. In Section III we perform extensive data analysis of the observed task running times, generated from different globally dispersed nodes. Statistical representations as the Boxplots, histograms, Auto-Correlation Functions, and other statistical properties will be investigated. The results show that the characteristics of the running times vary strongly among the different nodes, and moreover, that the characteristics (e.g., mean, burstiness) for each given node may differ strongly over different time periods. In Section IV we address a number of topics for further research.

II. Data Collection

In this section we describe the data collection procedure.

To perform experiments on shared processors in a real grid environment, we used Planetlab [16], a commonly used grid test bed environment shared by many users. At the time of our experiments, Planetlab version 2.0 was installed on the nodes. We ran in total 40 runs on 18 different Planetlab nodes. Each single run generated a dataset and consists of the running times (i.e., wallclock times) of 2000 consecutive and identical tasks. In order to correlate the statistical properties of the running times at the different nodes, at each day all runs were kicked off *simultaneously* at 9:00 CET.

III. Statistical analyses

In this section we analyze the statistical properties of 40 real datasets of running times of tasks on shared processors. We successively discuss the Box-and-Whisker plots, histograms, Auto Correlation Functions (ACFs), and other statistical properties of the datasets.

A. Box-and-Whisker plots

Constucting Box-and-Whiskerplots, commonly known as Boxplots [17], is a common way to highlight the differences between the medians, the quartiles, the minimum, the maximum and the outliers of different datasets. Figure 1 shows the Boxplots of the running times (in milliseconds) of 6 representative datasets. Datasets 9 and 10, and also 31 and 32 are generated on the same node on consecutive days. The plot gives a macroscopic view of the data. More precisely, we consider the three quartiles: the 25%-percentile (denoted as $Q1$), the median (denoted as $Q2$) and the 75%-percentile, $Q3$. These quartiles are plotted as a long horizontal line. In addition, we consider the statistical measures $A_{down} := Q1 - 1.5(Q3 - Q1)$, $A_{up} := Q3 + 1.5(Q3 - Q1)$, which are indications of the data points that should not be considered as outliers, and indicated by short horizontal lines. Finally, the outliers are plotted by small circles.

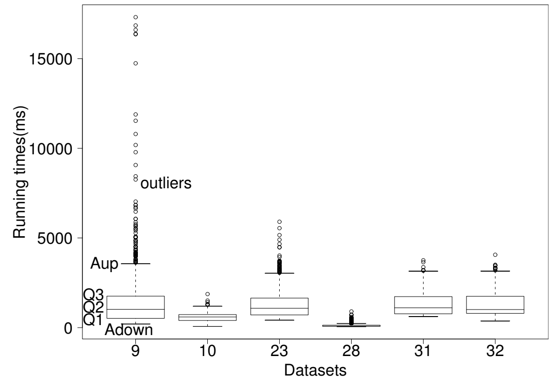


Fig. 1. Boxplot of 6 datasets of running times

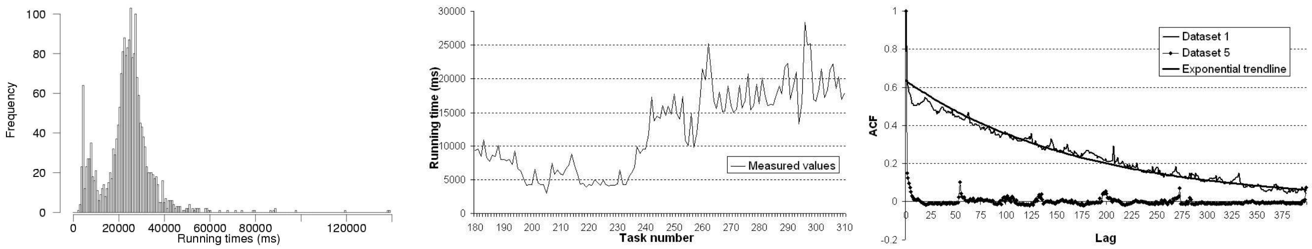
Figure 1 leads to the following observations: (1) the characteristics of the running times at a given node on different days in some cases differ strongly (see for example the results for datasets 9 and 10), but can be quite similar in other cases (see for example datasets 31 and 32), (2) the running-time characteristics of different nodes are strongly different, even when experiments are done at the same time, (3) the running times at a given node within a given run are highly bursty, and have a large number of strong outliers, and (4) in most cases the outliers correspond to very large values of the running times, but we have also seen that in some cases outliers correspond to very small running times.

We reemphasize that the observed heterogeneity of characteristics of the running times in the grid environment differs *fundamentally* from the running-time characteristics in dedicated clusters of processors, which are homogeneous and well predictable.

B. Histogram

Many grid properties can be fitted in standard distributions [1], [2], [4]–[6], and therefore, it is easy to make simulations or computations with those properties. To analyze the frequency distribution of the running times in more detail, Figure 2(a) shows the histogram of the marginal runnning-time distribution of a representative dataset.

Figure 2(a) shows that the running-time distribution is *multi-modal*. This is caused by a level switch in the running times over sustained time periods (ranging from minutes to hours). To illustrate this, Figure 2(b) gives a graphical representation of parts of the dataset. Figure 2(b) shows that from datapoints 235 to 260 a level switch occurs. From task number 180 to



(a) **Histogram of dataset 1** (b) **Level switch in dataset 1** (c) **ACF plots of 2 different datasets**

Fig. 2. Histogram, level switch, and ACF plots

230, the running times are between 3500 and 6000, which explains the first peak in Figure 2(a). From task number 260 to 310 the measured running times are between 15000 and 30000, which explains the second peak in the histogram. Level switches are presumably caused by changes in the processor load due to the launching or termination of other tasks on the same node.

C. Auto Correlation Function

The Auto Correlation Function (ACF) is an effective way to investigate whether the datapoints show correlations over different time scales (for more details, see [18]). The ACFs of datasets 1, 5, and the exponential trend of dataset 1 are shown in Figure 2(c). The results show that the ACFs do not follow predictable patterns, and show strong differences over the small (ranging from 1 up to 20) and the large (higher than 20) lag values. For example, the ACF of dataset 1 has significant autocorrelations over the small lag values. However, the ACF of dataset 5 decreases very quickly to 0, even for small lag numbers. For the datasets with significant autocorrelations for the large lag values, we consistently observe exponentially decaying autocorrelations which suggests that the successive running times are short-range dependent.

D. Other statistical properties

In this subsection we provide an datasets analysis by another set of statistical properties. We analyze the 40 datasets by 5 different statistics. The formulas of the statistics are omitted for brevity. Figure 3 gives a graphical representation of the results for a representative selection of datasets. Next we discuss the results in detail.

The first statistic is the Coefficient of Variation (CoV). The CoV equals the standard deviation divided by the average, and is a scale-invariant indicator for the variability of the datapoints. We conclude that the average and the standard deviations of the running times of the tasks may differ strongly between the dataset. Figure 3(a) shows that the CoV of the running times are fairly low, ranging between 0.22 and 1.81.

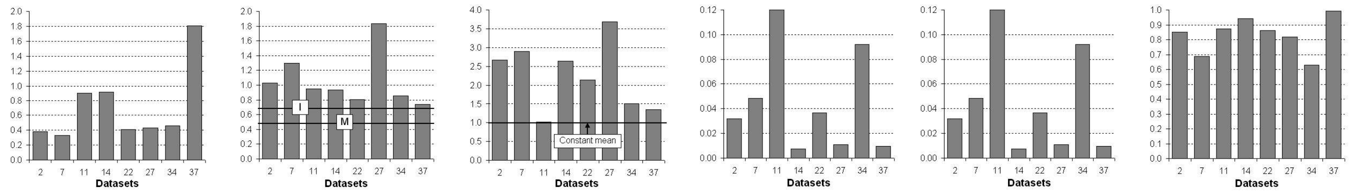
The next statistic, which is represented in Figure 3(b), is the standard deviation over the Root of the Mean Squared Deviation (RMSD) of two successive datapoints. This statistic indicates to what extent the standard deviation is caused by short- or long-term fluctuations. This statistic theoretically varies between 0.5 and infinity. When the statistic has the theoretical minimum value of 0.5 (see line M in Figure 3(b)), it indicates that the values alternatingly switch between two values: any two successive values show a correlation of -1. A value of $0.7 (= \frac{1}{2}\sqrt{2})$ (see line I in Figure 3(b)) indicates that the values are independent and identically distributed. The higher the value the more long-term fluctuations influence the standard deviation. The average of 1.12 for this statistic, which is significantly higher than 0.7, indicates a significant amount of long-term fluctuations. Moreover, we conclude from the high variety of the height of this statistic, which is shown by Figure 3, that the proportion of long- and short-term fluctuations differs per dataset.

The third statistic is the CoV of the averages and indicates to what extent the average of the running times of tasks changes during the run. A value of 1.0 for this statistic indicates that the average stays constant during the run. Figure 3(c) shows that for all datasets the averages fluctuate significantly: the statistic shows values that are signifcantly higher than 1.0. Nevertheless, the datasets show a high diversity in to what extent the averages fluctuate. We conclude from the results of this statistic that all the datasets have many level switches.

The fourth statistic is the standard deviation of standard deviations of, in this paper, 20 successive values divided by the average of those standard deviations. This statistic indicates how much the standard deviation (stdev) of the running times changes during the run. The closer this statistic is to 0, the more constant the standard deviation is during the run. Figure 3(d) shows that all the datasets have substantial fluctuations in the standard deviations. Nevertheless, there is diversity in the fluctuations of the standard deviations between the different sets.

The next statistic is the fraction of peaks in each dataset. A peak is defined as a value that differs (up- or downwards) from its previous value more than 2 times the standard deviation. The results show that the fraction of the peaks show some difference between the datasets, ranging between 1% and 22%. The average fraction of peaks (6%) is relatively low compared to the normal distribution (16%) and to the exponential distribution (13%). Figure 3(e) shows significant variation among the datasets in the fraction of peaks.

The last statistic indicates the fraction of the total amount of fluctuations (i.e., RMSD of two successive datapoints) that is



(a) **Variability** (b) **Short- or longterm** (c) **Changes in averages** (d) **Changes in stdevs** (e) **Fraction of peaks** (f) **Impact peaks**

Fig. 3. Statistical properties of the datasets

caused by the peaks. Figure 3(f), the graphical representation of this statistic shows that on average 76% of the total amount of fluctuations is caused by the peaks. This value does not differ significantly from the fraction of the normal distribution (76%) and the exponential distribution (81%).

To summarize, the statistical data analysis of the datasets shows that: (1) the characteristics of the datasets are mostly completely different, (2) the datasets show on average more long-term than short-term fluctuations and the proportion differs per dataset, (3) the averages fluctuate significantly during the run, with differences in the amount of fluctuations between the different nodes, (4) the standard deviations fluctuate significantly during the run, with differences in the amount of fluctuations between the different nodes, and (5) the datasets contain a small amount of peaks that have a huge influence on the standard deviation, the total amount of fluctuations, and the variance.

IV. Conclusions and challenges

In this paper we have presented the results of an extensive statistical analysis of the running times in a global-scale grid environment. We have found a number of characteristics that have not been observed before in the context of global-scale grids. The results of this paper are very useful for a number of purposes, and address a number of challenging topics for further research. First, the observations provide valuable input for the development of effective techniques for the *prediction* of running times, which can be explored to develop efficient control schemes for robust grid applications; example of such techniques are dynamic load balancing techniques. Second, the results presented here are also very useful for performing simulations or computations to assess the effectiveness of control strategies prior to their creation. In this way we can do an initial assessment of the effectiveness of control strategies in a simulation environment, and selecting the most promising ones for further exploration. In this way, tremendous time and cost savings can be realized, performing extensive and time-consuming experimentation only for a limited number selected control strategies.

References

[1] V. Paxson and S. Floyd, "Wide area traffic: the failure of poisson modeling," *IEEE/ACM Trans. Netw.*, vol. 3, no. 3, pp. 226–244, 1995.

[2] M. Mutka and M. Livny, "The available capacity of a privately owned workstation environment," *Performance Evaluation*, vol. 12, no. 4, pp. 269–284, 1991.

[3] P. A. Dinda, "The statistical properties of host load," *Scientific Programming*, vol. 7, no. 3-4, pp. 211–229, 1999.

[4] W. Leland and T. J. Ott, "Load-balancing heuristics and process behavior," in *SIGMETRICS '86/PERFORMANCE '86: Proceedings of the 1986 ACM SIGMETRICS joint international conference on Computer performance modelling, measurement and evaluation*. New York, NY, USA: ACM Press, 1986, pp. 54–69.

[5] M. Harchol-Balter and A. B. Downey, "Exploiting process lifetime distributions for dynamic load balancing," in *SIGMETRICS '96: Proceedings of the 1996 ACM SIGMETRICS international conference on Measurement and modeling of computer systems*. New York, NY, USA: ACM Press, 1996, pp. 13–24.

[6] D. Kondo, M. Taufer, C. Brooks, H. Casanova, and A. Chien, "Characterizing and evaluating desktop grids: An empirical study," in *Proceedings of the International Parallel and Distributed Processing Symposium (IPDPS'04), April 2004.*, 2004.

[7] J. Schopf and F. Berman, "Performance prediction in production environments," in *Proceedings of IPPS/SPDP*, vol. 00, 1998, pp. 647–653.

[8] Y. Qiao and P. Dinda, "Network traffic analysis, classification, and prediction," *Tech. Rep. NWU-CS-02-11*, January 2003.

[9] L. Cheng and I. Marsic, "Modeling and prediction of session throughput of constant bit rate streams in wireless data networks," in *Proceedings of the 2003 IEEE Wireless Communications and Networking Conference (WCNC'03)*, 2003, pp. 1733–1741.

[10] D. Lu, Y. Qiao, P. A. Dinda, and F. E. Bustamante, "Characterizing and predicting tcp throughput on the wide area network," in *ICDCS '05: Proceedings of the 25th IEEE International Conference on Distributed Computing Systems (ICDCS'05)*. Washington, DC, USA: IEEE Computer Society, 2005, pp. 414–424.

[11] P. A. Dinda, "A prediction-based real-time scheduling advisor," in *IPDPS '02: Proceedings of the 16th International Parallel and Distributed Processing Symposium*. Washington, DC, USA: IEEE Computer Society, 2002, pp. 35–42.

[12] ——, "Online prediction of the running time of tasks," *Cluster Computing*, vol. 5, no. 3, pp. 225–236, July 2002.

[13] H. Dail, G. Obertelli, F. Berman, R. Wolski, and A. Grimshaw, "Application-aware scheduling of a magnetohydrodynamics application in the legion metasystem," in *HCW '00: Proceedings of the 9th Heterogeneous Computing Workshop*. Washington, DC, USA: IEEE Computer Society, 2000, pp. 216–228.

[14] R. Wolski, "Experiences with predicting resource performance on-line in computational grid settings," *SIGMETRICS Performance Evaluation Review*, vol. 30, no. 4, pp. 41–49, 2003.

[15] P. A. Dinda and D. R. O'Hallaron, "Host load prediction using linear models," *Cluster Computing*, vol. 3, no. 4, pp. 265–280, 2000.

[16] "http://www.planet-lab.org."

[17] J. W. Tukey, *Exploratory Data Analysis*. Reading, MA: Addison-Wesley, 1977, pp. 39–43.

[18] G. E. P. Box and G. Jenkins, *Time Series Analysis: Forecasting and Control*. Holden-Day, 1976, pp. 28–32.