Future Generation Computer Systems 25 (2009) 460–465

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

Model-based simulation and performance evaluation of grid scheduling strategies

Hui Li [a,*,1], Rajkumar Buyya [b]

[a] SAP Research, Vincenz-Priessnitz-Strasse 1, 76131 Karlsruhe, Germany
[b] The GRIDS Lab, CSSE Department, University of Melbourne, VIC 3010, Australia

ARTICLE INFO

Article history:
Received 15 June 2008
Received in revised form
23 September 2008
Accepted 23 September 2008
Available online 10 October 2008

Keywords:
Workload modeling
Performance evaluation
Simulation
Grid computing

ABSTRACT

Simulation studies of Grid scheduling strategies require representative workloads to produce dependable results. Real production Grid workloads have shown diverse correlation structures and scaling behavior, which are different to the characteristics of the available supercomputer workloads and cannot be captured by Poisson or simple distribution-based models. We present statistical models that are able to reproduce various autocorrelation structures, including pseudo-periodicity and long range dependence. By conducting model-based simulation, we quantitatively evaluate the performance impacts of workload autocorrelations in Grid scheduling. The results indicate that autocorrelations result in system performance degradation, both at the local and the Grid level. It is shown that realistic workload modeling is not only possible, but also necessary to enable dependable Grid scheduling studies.

1. Introduction

Grid computing is rapidly evolving as the next-generation platform for system-level sciences and beyond. In such a dynamic and heterogeneous environment, good scheduling mechanisms are needed to deliver nontrivial quality-of-service. Understanding the workload characteristics is crucial because not only workload is an indispensable part in scheduling evaluation but also a deep understanding will give us hints on how to improve the scheduling heuristics.

A study of workload dynamics on clusters and Grids has been conducted in [13]. It is shown that workload characteristics on clusters and Grids, particularly in data-intensive environments, are significantly different to those on conventional supercomputers. Job arrivals show a variety of correlation structures, including short range dependence, pseudo-periodicity, and long range dependence. "Bag-of-tasks" behavior with a strong degree of temporal locality is observed, which leads to the long autocorrelation lags in workload attributes such as run time. Simple models such as Poisson or distribution-based methods are not able to capture the second-order properties such as autocorrelation.

In this paper, we present an overview of workload models developed for Grid environments that are able to reproduce the correlation structures as in the real traces. To show that the models are not only possible but also practical, we conduct model-driven simulations of Grid scheduling strategies. Experiments are designed to quantify the performance impacts of workload correlations in Grid scheduling. The impacts, as we will show later, are very large. Long range dependence results in big performance degradation, which effects should be taken into consideration in the scheduling evaluation studies.

The rest of the paper is organized as follows. Section 2 provides an overview of some of the representative research in Grid scheduling. The focuses are on how workloads are treated and what is the simulation environment. Section 3 discusses the workload models developed for capturing the statistical properties of real Grid traces, including short range dependence, pseudo-periodicity and long range dependence. A comprehensive model is obtained by combining job arrival process and series of job attributes such as run time. Section 4 describes the simulation setup. We build the simulation environment based on GridSim and develop two cases for performance evaluation studies, namely Grid resource case and Grid broker case. Section 5 presents the experimental results for the two cases, namely, the performance impacts of autocorrelations on one FCFS queue with multiple servers, and on a Grid broker and multiple clusters with background workload. Section 6 comes to the conclusion that autocorrelations cause performance degradation in both cases and future work on how to improve scheduling are discussed.

* Corresponding author. Tel.: +49 6227 752674; fax: +49 6227 78 51912.
E-mail address: hui.li@computer.org (H. Li).

[1] This work of H. L. was carried out while he was affiliated with Leiden University, The Netherlands.

0167-739X/$ – see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.future.2008.09.012

2. Evaluation of scheduling algorithms

Efficient and effective scheduling at a meta-level is very important in a Grid computing environment. In order to develop and evaluate new Grid scheduling algorithms, two fundamental issues have to be addressed for performance evaluation studies. On one hand, representative workload traces are needed to produce dependable results. On the other hand, a good testing environment should be set up, most commonly through simulations. In this section, we review some of the current research in Grid scheduling, with a special emphasis on the mentioned two issues. A Grid scheduling architecture typically consists of two levels, namely, the Grid scheduler(s) and the local resource management systems. Since the clusters/resources participating in a Grid have their own local activities, the workloads are further categorized into Grid-level jobs (Grid workload) and locally generated jobs (background workload). Due to the lack of traces at the Grid level, simplified assumptions on workloads are commonly made in scheduling studies. In [5,22] bulk sizes of 200 to 1000 jobs are used to evaluate the proposed "off-line" scheduling algorithms. For "on-line" mode of scheduling, jobs either arrive in fixed intervals [7], or strictly in sequence [17]. More realistic treatments include the use of real workload traces. In [6] traces obtained from Network Weather Service (NWS) are used to study a set of heuristics for parameter sweep applications, including max-min, min-min, Sufferage, and XSufferage. In [21] there are two specific traces under study: one is obtained from iPSC/860 parallel workload at NAS, the other consists of parameter sweep applications (PSA). In [2] traces from a multi-cluster environment (DAS) are utilized in the study of processor co-allocation strategies. In [18] workloads on parallel supercomputers available from the Parallel Workload Archive are used in evaluating a SLA-based cooperative superscheduling algorithm. Work in [8,16] focus on workflow scheduling, in which workflows are randomly generated or based on real traces. Trace-based simulations have the advantages of being easy-to-use, and the results obtained are reproducible and comparable. However, it is not as flexible as models in case that many traces have to be generated to enable a Grid scheduling study. The traces available from parallel workloads can also have significantly different characteristics compared with Grid workloads, which has been empirically observed in [13]. Such differences, in return, may lead to considerably different performance evaluation results.

Background workload is another important issue to be addressed in a heterogeneous and non-dedicated Grid environment. Much previous work does not include background load information because traces or characterization are not widely available concerning the background workloads on clusters. Some research employs models to generate local jobs as background. In [22] the local system load is modeled as a Gaussian distribution with predefined mean and variance. In [21,8] background job arrivals are modeled as a Poisson process and run times are drawn from an exponential distribution in [8]. Although such models are simple to use and analytically tractable, it might not reflect real job characteristics at the cluster level.

The third problem is how to set up a simulation environment for performance evaluation. GridSim is a popular choice to build Grid simulations [5,22,3,18,16,20]. Other simulators developed specially for Grids include Simgrid [6], GangSim [7] and ChicSim [17]. Some researchers build their own version of simulators to meet their research goals [8,21]. Commercially available products are also employed in conducting simulations [2]. Although many simplifications and assumptions are made in the simulations compared to real Grid environments, simulations are commonly considered a flexible and tractable way of evaluating different Grid scheduling algorithms as well as other design issues.

The main focus of this paper is on realistic workloads. Although far from an exhaustive list of Grid scheduling literature, we can see that a large amount of research work either use traces not typically from real production Grids, or use simple workload models (Poisson, fixed-interval arrivals, or Gaussian system load). These traces or models, however, exhibit significantly different characteristics than the traces on production Grids. As has been studied and reported in [13], *pseudo-periodicity*, *long range dependence (LRD)*, and *"bag-of-tasks" behavior* with strong temporal locality are the main properties that characterize production Grid workloads. Therefore, it is important that representative models be developed to capture the salient properties of Grid workloads. In the following sections, we present an overview of the recent work on workload modeling for clusters and Grids. Moreover, by using the developed models we conduct model-based simulation of Grid scheduling strategies and quantify the performance impacts of various autocorrelation structures in workloads.

3. Workload modeling in grids

Based on workload traces from a large production Grid and several participating clusters (Table 1), we developed models that are able to reproduce the statistical properties of traces at different levels. The following presentations are based on research in [9–12] and discuss job arrivals and job attributes, respectively.

3.1. Job arrivals

Job arrivals can be described as a *point process* and two representations are commonly used, namely, *interarrival time process* and *count/rate process*. The *count process* is formed by dividing the time axis into equally spaced contiguous intervals and counts the number of events within each interval. Forming the sequence of counts generally loses information but it allows the correlation in the counts to be readily associated with that in the point process [14]. The *rate process* is basically the sequence of counts normalized by the count interval.

In the following discussions, doubly stochastic models are the so-called "full" models because they fit the interarrivals. Models for pseudo-periodicity and long range dependence operate on the count processes, by which the correlation structures can be reliably revealed. Algorithms are also proposed to convert rates back to interarrivals. Another advantage of modeling the count process lies on its additive nature: models for different VOs can be added together to generate an aggregated trace in which the VO labels are preserved. This is useful for evaluating scheduling strategies in which policies are based largely on VOs.

3.1.1. Doubly stochastic models

The homogeneous Poisson processes are well-known "zero-memory" models, whose interarrivals and counts are independently and identically distributed (I.I.D.) random variables. A generalization of the Poisson process is the so-called doubly stochastic Poisson process (DSPP). Its rate $\mu(t)$ is modulated by a positive-valued continuous-time stochastic process rather than a fixed constant. The resulting process is thus doubly random: one source of randomness arises from the stochastic rate $\mu(t)$ while another comes from the intrinsic Poisson events. A Markov modulated Poisson process (MMPP) is a DSPP whose rate is controlled by a finite state continuous-time Markov chain. MMPP models have several attractive properties, such as being able to capture correlations between interarrival times while still remaining analytically tractable. MMPPs are suitable to generate processes that are short or middle range dependent [11].

Table 1
Summary of workload traces used in the experimental study of this paper.

Trace	Location	Arch.	Scheduler	CPUs	Period	#Jobs
LCG1	Grid wide	data Grid	Grid Broker	∼30K	Nov 20–30, '05	188,041
LCG2	Grid wide	data Grid	Grid Broker	∼30K	Dec 19–30, '05	239,034
NIK05	NIKHEF, NL	PC cluster	PBS/Maui	288	Sep – Dec, '05	63,449
RAL05	RAL, UK	PC cluster	PBS/Maui	1,000	Oct – Nov, '05	332,662
LPC05	LPC, FR	PC cluster	PBS/Maui	140	Feb – Apr, '05	71,271

3.1.2. Pseudo-periodicity

Pseudo-Periodicity is considered as one basic pattern that originates from automated submission schemes, which is present in large-scale data-intensive environments. Our approach for modeling the pseudo-periodic pattern is inspired and adapted from a signal decomposition methodology called *matching pursuit*. It is a greedy, iterative algorithm which searches a family of candidate functions (also called "atoms") for the element that best matches the signal and subtracts this function to form a residual signal to be approximated in the next iteration. Sinusoidal and harmonic models are used for fitting the job arrival count processes, whose parameters are estimated via matching pursuit. Matching pursuit is also shown to be able to extract patterns from signals and makes it possible to model patterns individually. For example, some long range dependent processes could be mixed with certain high-frequency periodic components. Matching pursuit is able to separate these two patterns so that suitable models can be applied individually. We refer to [10] for details about the matching pursuit approach in modeling pseudo-periodic job arrivals.

3.1.3. Long range dependence

A process $X(t)$ is said to be long range dependent (LRD) if either its autocorrelation function (ACF) or power spectrum satisfies the following conditions:

$$R(k) \sim c_r k^{\alpha-1}, \quad k \to \infty, \quad \text{or} \quad S(f) \sim c_f f^{-\alpha}, \quad f \to 0. \tag{1}$$

The autocorrelation function $R(k)$ decays so slowly that $\sum_{k=-\infty}^{\infty} R(k) = \infty$ and $S(0) = \infty$. LRD is one class of the general scaling process [1]. Job arrival processes exhibit long range dependence at many levels, including VO, cluster, and the Grid [13]. LRD is closely related to temporal burstiness, in which jobs tend to arrive within bursty periods. This is in accordance with the "bag-of-tasks" arrival behavior in data-intensive Grid environments. We apply the multifractal wavelet model (MWM) [19] to fit the count/rate processes because it provides a coherent wavelet framework for analysis and synthesis of the scaling behavior. It is shown that second order properties such as the autocorrelation function (ACF) and the scaling behavior can be well reconstructed by MWM [9].

3.2. Job attributes

For data-intensive workloads running on production clusters and Grids, it has been pointed out that strong temporal locality and "bag-of-tasks" behavior lead to long correlation lags in job attributes such as run time and memory consumption [13]. We have proposed a model for workload attributes that can capture not only the marginal distribution but also the second order statistics such as the autocorrelation function (ACF) [12]. This is fulfilled by a two-stage approach: first, a *mixture of Gaussians* model is used to fit the probability density function (PDF), whose parameters are estimated via a framework called *model based clustering* (MBC). The MBC framework can further cluster the data according to the Gaussian components, which plays an important role in creating correlations in the next stage. Second, a novel *localized sampling* algorithm is proposed to generate correlations in the synthetic data series. It is discovered that the number of repetitions of cluster labels obtained via MBC empirically follow a Zipf-like (power law) distribution. Sampling repeatedly from a certain cluster according to the Zipf law is able to create correlations in the series. Furthermore, a *cluster permutation* procedure is introduced so that the autocorrelations in the synthetic data can be controlled to match those in the real trace via a single parameter. Experimental results have shown that the proposed model can fit the marginal distribution well at the same time match the autocorrelation function of the original trace [12]. This model is referred as *MBC-LSP* in the context of this paper.

3.3. A comprehensive model

Although correlations and the scaling behavior can be reliably revealed using the count/rate process, it is necessary to generate a point process in the form of interarrival times so that a full description can be obtained. A so-called *controlled-variability integrate-and-fire* (CV-InF) algorithm can be used for such conversion [9]. Since the rates are additive, it is possible to add up several rate processes with different patterns to form an aggregated process, and convert it into interarrivals. By combining job arrival process and series of job attributes such run time, we obtain a comprehensive model for independent tasks in data-intensive Grids. Parallelism is not taken into account here because there are not enough parallel jobs available for study in the production Grid traces, which mostly consist of sequential jobs such as those from high energy physics and biomedical sciences.

Our goal is to demonstrate the feasibility and advantages of using workload models to drive simulations. The example is to investigate the performance impacts of workload correlations in Grid scheduling. For this purpose we generate synthetic traces with different correlation structures. Job arrival processes can be not dependent (NoD), short range dependent (SRD), and long range dependent (LRD), which can be modeled by a Poisson process, a 2-state Markov modulated Poisson process (MMPP2), and a multifractal wavelet model with CV-InF conversion (MWM). Job run times have the same three correlation structures and they can be modeled by MBC-LSP with different permutation window sizes. Experimental results of using these models to generate Grid-level and background workloads are presented in Section 5.

4. Grid simulation

We build the simulation environment based on GridSim [4]. GridSim provides a discrete-event framework for simulating core Grid entities such as jobs, resources, and information services. For the performance evaluation of Grid scheduling under correlated workloads we implement two case studies, which are elaborated in the following sections.

4.1. Grid resource case

The first case is a computing cluster with one FCFS queue. The simulated cluster is space-shared and has 100 CPUs. In order to understand what are the workload characteristics we analyze the

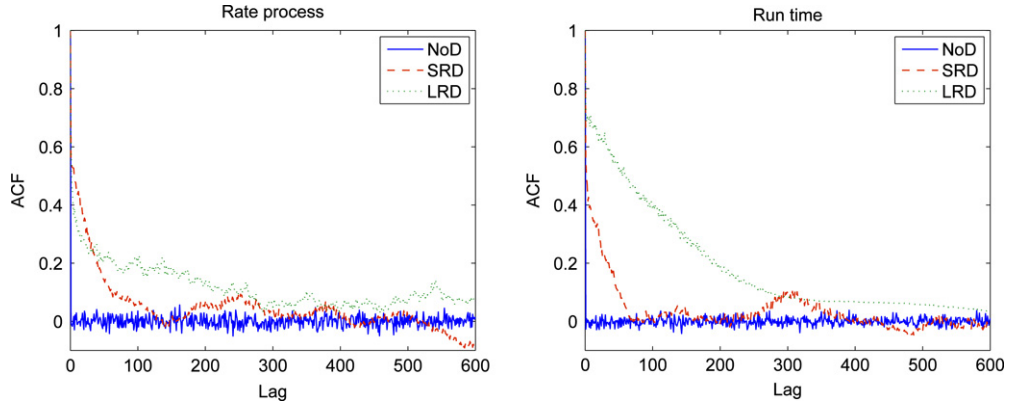

Fig. 1. Synthetic workload traces with different correlation structures. For job arrival rate processes, NoD — a Poisson process, SRD — a MMPP2 process, LRD — a MWM process with CV-InF conversion. For job run times, NoD — MBC with random sampling, SRD — MBC with localized sampling ($W = 1$), LRD — MBC with localized sampling ($W = 500$).

Table 2
Statistics for job rate processes on clusters (s — seconds, P.P. — Pseudo periodic).

	View	Mean	CV	Distribution	ACF
	Local	0.04/s	9.9	Long tail	SRD
RAL05	Grid	0.02/s	2.1	Short tail	MRD
	All	0.06/s	6.3	Long tail	SRD
	Local	0.002/s	8.7	Long tail	P.P.
NIK05	Grid	0.005/s	4.1	Long tail	MRD
	All	0.006/s	4.4	Long tail	MRD
LPC05	All	0.01/s	2.2	Short tail	LRD

Table 3
Statistics for job run times on clusters (the unit of run time is seconds).

	View	Mean	CV	Distribution	ACF
	Local	10,401	1.9	Long tail	LRD
RAL05	Grid	13,973	1.7	Long tail	LRD
	All	11,727	1.9	Long tail	LRD
	Local	14,584	1.9	Long tail	MRD
NIK05	Grid	16,934	1.9	Long tail	LRD
	All	16,336	1.9	Long tail	LRD
LPC05	All	4,585	3.7	Long tail	LRD

traces on three representative data-intensive clusters (Table 1). For *RAL05* and *NIK05* we are able to roughly distinguish the Grid jobs and the locally generated jobs. By examining the "user name" field in the traces, jobs from "pool account" (usually a VO name plus a unique number) are considered Grid jobs while jobs from a "real" user name are seen as local jobs. As is shown in Table 2, different clusters have different job arrival rates and autocorrelation structures. The arrival ratio and patterns of local jobs versus Grid jobs are also highly diversified. The job run times, on the other hand, have relatively smaller variances and are almost all long range dependent. These statistics give us a good reference on how to adjust the model parameters for synthetic workload generation (Table 3).

4.2. Grid broker case

The second case naturally extends to the Grid level. In our environment we simulate 8 space-shared clusters whose properties resemble those of the eight largest clusters in the LHC production Grid (LCG).[2] These properties are shown in Table 4. Each cluster has its own local background workload, in which the job arrival rate scale with the capacity of the resource. The chosen algorithm for the Grid broker case is called MCT (Minimum Completion Time) [15]. MCT assigns each incoming job to the cluster with the minimum expected completion time for that job. Clusters are assumed to be FCFS-based so the minimum completion time can be estimated by simulating FCFS scheduling for the local queue. The estimated minimum completion times are published to the Information Service and can be used by the broker for making a scheduling decision. The job flow at the Grid level is sent to the broker and has an average arrival rate of 0.1/s. The workload models generate synthetic traces with different structures and are stored in text files. GridSim reads the workloads from the files and carries out the simulation.

5. Experimental studies

In previous sections, we discussed the workload models and the simulation environment setup. In this section, we present the evaluation results that quantify the performance impacts of workload correlations in Grid scheduling. Table 5 shows the model parameters used to generate synthetic workload traces. For the interpretation of these parameters we refer to the corresponding papers. In terms of parameter space, the tradeoff is that we need more complex models to generate processes with longer range dependence. Different correlation structures and associated models are shown in Fig. 1. For all generated processes the means and standard deviations remain unchanged, only the dependencies in the series are different. This is the basis of the comparison studies presented as follows.

1. *What is the performance impact of autocorrelations on one FCFS queue with multiple servers?*

We study the Grid resource case first. Performance is measured by the average job slowdown[3] as a function of system utilization,[4] which is shown in Fig. 2. We can see that the impact of autocorrelations is very large: the bigger the ACF, the worse the performance. Similar results have been reported in a clustered web server environment [23]. The cause of such performance degradation is the high degree of temporal burstiness in a LRD process. Bursty arrivals, which is the opposite of smoothness (e.g. Poisson), result in a long queue of waiting jobs. Consequently it leads to much longer queueing delays (bigger slowdown for jobs) and overall lower system utilization.

2. *What is the performance impact of autocorrelations on a Grid broker and multiple clusters with background workload?*

[2] LCG is a data storage and computing infrastructure for the high energy physics community that will use the Large Hadron Collider (LHC) at CERN. http://lcg.web.cern.ch/LCG/.

[3] *Slowdown* is defined as the average job response time (run time plus queue wait time) divided by the average job run time.

[4] *Utilization* means the average system utilization and it is calculated as the proportion of system's resources which are busy.

Table 4
Characteristics of the largest eight clusters in the LCG Grid (data obtained in April, 2007) and corresponding parameters used in the simulation. BG workload shows the local job arrival rate on the cluster. Run times of local jobs are scaled for different utilizations.

Site	Location	#CPUs	Downscale	SpecINT2k	BG workload
CERN-PROD	CH	3534	354	970	0.05/s
FZK-LCG2	DE	2662	266	1289	0.04/s
USCMS-FNAL	US	1925	193	1600	0.033/s
UKI-QMUL	UK	1644	164	1381	0.033/s
IN2P3-CC	FR	1454	145	892	0.025/s
SARA-LISA	NL	1352	135	1636	0.025/s
RAL-LCG2	UK	1266	127	1000	0.02/s
INFN-T1	IT	1238	124	747	0.02/s

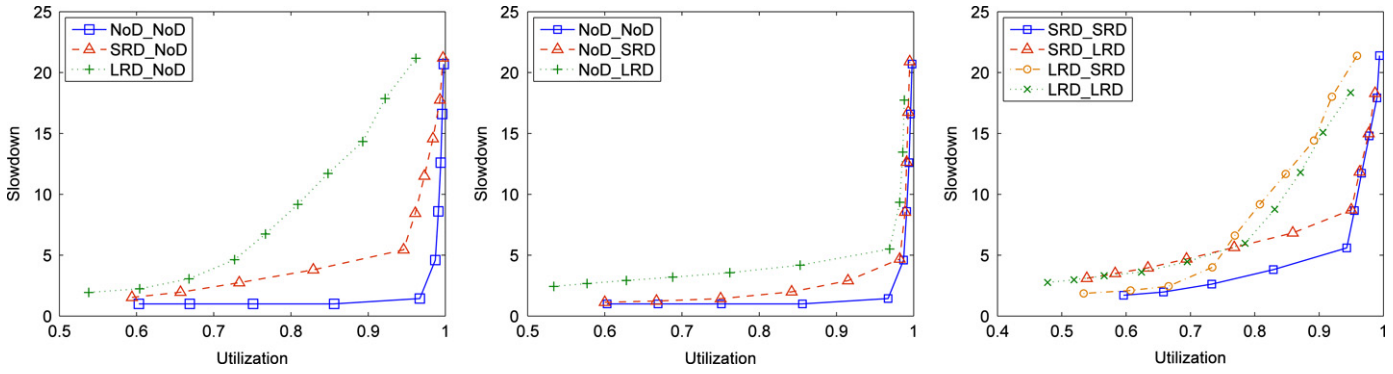

Fig. 2. Performance impacts of autocorrelations on a cluster with one FCFS queue. Workload structure is denoted as "arrival"_"run time".

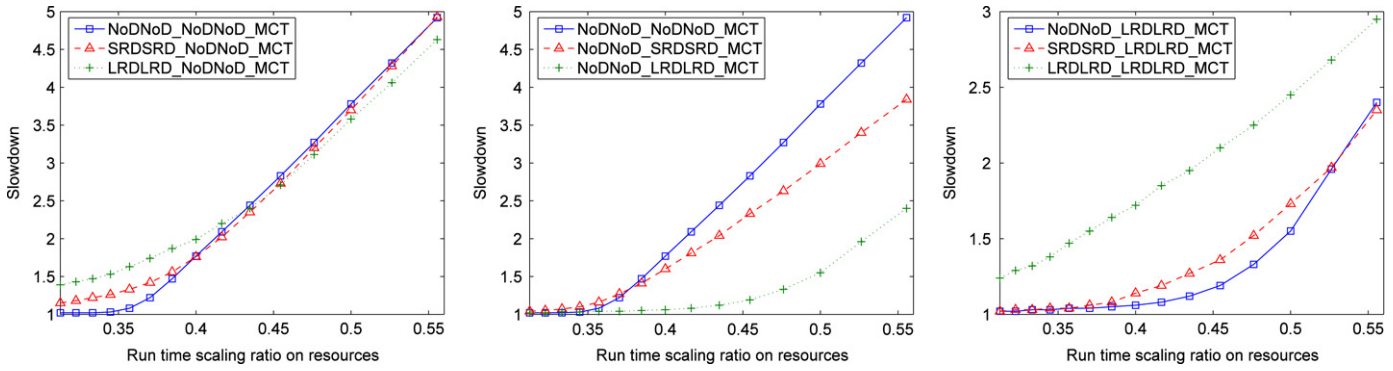

Fig. 3. Performance impact of autocorrelations in Grid scheduling. Workload structure is denoted as "Grid arrival""Grid run time"_"local arrival""local run time"_"scheduling algorithm". Run time scaling ratio is defined as the job MIPS rating versus resource MIPS rating.

Table 5
Model parameters used in the experimental study. MWM parameters are fitted using *biomed, LPC05*. MBC-LSP parameters are fitted for *hep1, RAL05* (parameters for Gaussian mixtures are not shown).

Model	Parameters
Poisson	$\mu = 10$
MMPP2	$\sigma_1 = 0.04, \sigma_2 = 0.01, \lambda_1 = 8.0, \lambda_2 = 1.0$
MWM	$p = [3.3, 5.3, 6.6, 7.5, 6.7, 7.1, 4.8, 3.0, 2.2, 1.4]$, $\mu_c = 0.28, \sigma_c = 0.33$
MBC-LSP	$\alpha = 1.79, N = 1262, W = 1,500$

In the Grid broker case, at the cluster level the resource generates its own local background workload. At the Grid level, one stream of jobs flow into the broker. Therefore there are two levels of freedom in combining the autocorrelation structures, with each level having two attributes – job arrival and job run time. In this case, the performance is measured by the average job slowdown for Grid-level jobs as a function of the run time scaling ratio on resources. The run time scaling ratio is the job MIPS rating versus resource MIPS rating and a higher ratio indicates a larger average run time. By varying the run time scaling ratio we get the results as shown in Fig. 3. First, we investigate the impacts of Grid-level autocorrelations by setting the local background workloads to be not dependent (Fig. 3 left). Although not as large as in the Grid resource case, performance degradation is observed for larger autocorrelations in the lower range of the scaling ratios. Second, we study the implications of different autocorrelation structures in the local background workloads (Fig. 3 middle). Interestingly, we can see that Grid-level jobs actually perform better when the background workloads have stronger autocorrelations. This is explained by the lower system utilization resulted by the stronger temporal locality in more autocorrelated processes at the cluster level. If we set the local background workloads to be long range dependent and vary the correlation structures at the Grid level, we can see a large performance degradation by long autocorrelations. By combining these effects we conclude that autocorrelations in the workloads result in performance degradation both at the local and the Grid level.

6. Conclusions

In this paper, we propose the use of workload models to drive simulations of Grid scheduling strategies. Real production Grid workloads have shown rich correlation and scaling behavior,

which are different to conventional parallel workloads and cannot be captured using simple models such as Poisson or distribution-based methods. The introduced models are able to reproduce a variety of correlation structures, including pseudo-periodicity, short range dependence (SRD), and long range dependence (LRD), for job arrivals and job attributes such as run time. The practical use of these models is also demonstrated by simulation studies. By using the synthetic traces we are able to quantify the performance impacts of workload correlations in Grid scheduling. The results indicate that autocorrelations in workload attributes can cause performance degradation, in many situations this effect is huge. We can see that the development of good workload models are not only possible, but also necessary for dependable performance evaluation of scheduling strategies.

Further research includes how to improve scheduling under autocorrelations. In a two-level Grid scheduling scenario, long range dependence is not necessarily a bad situation. For instance, Fig. 3 (middle) shows that better performance is obtained for Grid-level jobs under LRD background workloads on clusters. Temporal burstiness, the opposite of smoothness, implies that the system has more idle periods or "holes" in the time line. This provides opportunities for the broker to do better load balancing at the Grid level.

Acknowledgments

The LCG Grid traces are provided by the HEP e-Science group at Imperial College London. *NIK05* and *RAL05* traces are provided by colleagues at NIKHEF (NL) and RAL (UK), respectively. *LPC05* trace is obtained from Parallel Workload Archive. The authors want to express the gratitude to all who graciously provide the data. The authors also thank M. Muskulus, L. Wolters, S. Venugopal, R. Ranjan, J. Broberg, A. Sulistio, and the Gridbus team for their suggestions and support.

Part of this research of H. Li is carried out as a visiting researcher in the GRIDS lab, University of Melbourne, which is funded by Leiden University Fonds under grant no. 6145/13-11-06/MvT, LIACS, and the University of Melbourne via ARC/DEST grants.

References

[1] P. Abry, M.S. Taqqu, P. Flandrin, D. Veitch, Wavelets for the analysis, estimation, and synthesis of scaling data, in: K. Park, W. Willinger (Eds.), Self-Similar Network Traffic and Performance Evaluation, Wiley, 2000.
[2] A. Bucur, D. Epema, Trace-based simulations of processor co-allocation policies in multiclusters, in: Proceedings of the 12th IEEE Symposium on High Performance Distributed Computing, HPDC, 2003.
[3] David Abramson, Rajkumar Buyya, Jonathan Giddy, A computational economy for grid computing and its implementation in the Nimrod-G resource broker, Future Generation Computer Systems 18 (8) (2002) 1061–1074.
[4] R. Buyya, M. Murshed, Gridsim: A toolkit for the modeling and simulation of distributed resource management and scheduling for grid computing, Concurrency and Computation: Practice and Experience (CCPE) 14 (2002) 1175–1220.
[5] R. Buyya, M. Murshed, D. Abramson, S. Venugopal, Scheduling parameter sweep applications on global grids: A deadline and budget constrained cost time optimization algorithm, Software Practice and Experience 35 (2005) 491–512.
[6] H. Casanova, A. Legrand, D. Zagorodnov, F. Berman, Heuristics for scheduling parameter sweep applications in grid environments, in: proceedings of the 9th Heterogeneous Computing Workshop, HCW'2000, 2000.
[7] C. Dumitrescu, I. Raicu, I. Foster, Di-gruber: A distributed approach to grid resource brokering, in: Proceedings of Supercomputing, SC, 2005.
[8] L. He, S. Jarvis, D. Spooner, D. Bacigalupo, G. Tan, G. Nudd, Mapping dag-based applications to multiclusters with background workload, in: proceedings of the 5th IEEE Symposium on Cluster Computing and The Grid, CCGrid, 2005.
[9] H. Li, Long range dependent job arrival process and its implications in grid environments, in: MetroGrid Workshop, Proceedings of First Intl. Conf. on Networks for Grid Applications, GridNets, ACM Press, 2007.
[10] H. Li, R. Heusdens, M. Muskulus, L. Wolters, Analysis and synthesis of pseudo-periodic job arrivals in grids: A matching pursuit approach, in: Proceedings of IEEE Symposium on Cluster Computing and The Grid, CCGrid, 2007.
[11] H. Li, M. Muskulus, Analysis and modeling of job arrivals in a production grid, ACM SIGMETRICS Performance Evaluation Review 34 (2007) 59–70.
[12] H. Li, M. Muskulus, L. Wolters, Modeling correlated workloads by combining model based clustering and a localized sampling algorithm, in: proceedings of 21st ACM International Conference on Supercomputing, ICS, 2007.
[13] H. Li, L. Wolters, Towards a better understanding of workload dynamics on data-intensive clusters and grids, in: Proceedings of the 21st IEEE International Parallel and Distributed Processing Symposium, IPDPS, 2007.
[14] S.B. Lowen, M.C. Teich, Fractal-Based Point Processes, John Wiley and Sons, Inc, 2005.
[15] M. Maheswaran, S. Ali, H.J. Siegel, D. Hensgen, R.F. Freund, Dynamic mapping of a class of independent tasks onto heterogeneous computing systems, Journal of Parallel and Distributed Computing 59 (2) (1999) 107–131. URL: citeseer.ist.psu.edu/article/maheswaran99dynamic.html.
[16] A. Ramakrishnan, G. Singh, H. Zhao, E. Deelman, R. Sakellariou, K. Vahi, K. Blackburn, D. Meyers, M. Samidi, Scheduling data intensive workflows onto storage-constrained distributed resources, in: Proceedings of the 7th IEEE Symposium on Cluster Computing and The Grid, CCGrid, 2007.
[17] K. Ranganathan, I. Foster, Decoupling computation and data scheduling in distributed data-intensive applications, in: Proceedings of the 11th IEEE Symposium on High Performance Distributed Computing, HPDC, 2002.
[18] R. Ranjan, A. Harwood, R. Buyya, Sla-based cooperative superscheduling algorithms for computational grids, ACM Transactions on Autonomous and Adaptive Systems (2008).
[19] R.H. Riedi, M.S. Crouse, V.J. Ribeiro, R.G. Baraniuk, A multifractal wavelet model with application to network traffic, IEEE Transactions on Information Theory 45 (3) (1999) 992–1019.
[20] G. Singh, C. Kesselman, E. Deelman, A provisioning model and its comparison with best effort for performance-cost optimization in grids, in: Proceedings of the Sixteenth IEEE International Symposium on High-Performance Distributed Computing, HPDC07, 2007.
[21] S. Song, K. Hwang, Y.-K. Kwok, Trusted grid computing with security binding and trust integration, Journal of Grid Computing 3 (2005) 53–73.
[22] S. Venugopal, R. Buyya, A set coverage-based mapping heuristic for scheduling distributed data-intensive applications on global grids, in: Proceedings of the 7th IEEE/ACM International Conference on Grid Computing, Grid06, 2006.
[23] Q. Zhang, N. Mi, A. Riska, E. Smirni, Load unbalancing to improve performance under autocorrelated traffic, in: Proceedings of IEEE Intl. Conf. Distributed Computing Systems, ICDCS, 2006.

Hui Li received his Ph.D. degree in Computer Science from Leiden University in 2008. Currently he is a researcher at SAP Research Karlsruhe, SAP AG, Germany. He was a visiting researcher in University of Melbourne (03–06/2007) and worked as a research scientist at NIHHEF in the EU DataGrid project (2003). Hui Li has co-authored over 20 papers in peer-reviewed conferences and journals, and received the Best Paper Award in IEEE eScience 2007. His research interests include workload modeling and prediction, scheduling and performance evaluation, system management, machine learning, and service computing.

Rajkumar Buyya is an Associate Professor and Reader of Computer Science and Software Engineering; and Director of the Grid Computing and Distributed Systems (GRIDS) Laboratory at the University of Melbourne, Australia. He received B.E and M.E in Computer Science and Engineering from Mysore and Bangalore Universities in 1992 and 1995 respectively; and Doctor of Philosophy (Ph.D.) in Computer Science and Software Engineering from Monash University, Melbourne, Australia in 2002. He was awarded Dharma Ratnakara Memorial Trust Gold Medal in 1992 for his academic excellence at the University of Mysore, India. He received Richard Merwin Award from the IEEE Computer Society (USA) for excellence in academic achievement and professional efforts in 1999. He received "Research Excellence Award" from the University of Melbourne for productive and quality research in computer science and software engineering in 2005. The Journal of Information and Software Technology in Jan 2007 issue, based on an analysis of ISI citations, ranked Dr. Buyya's work (published in Software: Practice and Experience Journal in 2002) as one among the "Top 20 cited Software Engineering Articles in 1986–2005". He received the Chris Wallace Award for Outstanding Research Contribution 2008 from the Computing Research and Education Association of Australasia, CORE, which is an association of university departments of computer science in Australia and New Zealand.