OPEN ISSUES IN GRID PERFORMABILITY

Stephen Jarvis† Nigel Thomas‡ Aad van Moorsel‡

† Department of Computer Science
University of Warwick
Warwick, United Kingdom
saj@dcs.warwick.ac.uk

‡ School of Computing Science
University of Newcastle upon Tyne
Newcastle upon Tyne, United Kingdom
{nigel.thomas, aad.vanmoorsel}@ncl.ac.uk

Abstract:
This paper seeks to identify some of the technology challenges and research opportunities in assessing, optimising and assuring performance and dependability on the grid. Grid computing offers many scientific and commercial benefits, but also many technical and organisational challenges. Amongst the challenges for grid computing is the need to provide reliable quality of service to end users, while enabling increasingly dynamic and federated usage scenarios. Understanding and quantifying the performance capabilities of a grid system and its applications is a crucial issue, without which performance improvements will be at best ad hoc. Nevertheless, this is particularly difficult for complex and dynamic systems like the grid, where both the computing environment (in terms of the contributing platforms and networks) and the application base are in continual change. There is an internationally recognised need for research in this area and success will lay the foundation for next generation e-Science.

1. Introduction

Grid computing is a large and rapidly developing area of active research. It promises much for high performance computing but can also suffer from overheads introduced by cross-domain connectivity and extensive middleware, as well as more familiar problems generally associated with distributed systems, such as communication latency and unreliable infrastructure. In essence grids are collections of heterogeneous computation and storage resources spread across different administrative domains, with tools for the user to find, reserve and use desired resources. Using grid technologies, computing facilities of various kind can be offered as on-line services [1].

Grid computing research initially grew from distributed systems support for large, data and computationally intensive tasks. Originally, applications were academically driven [2, 3], but more recently computationally intensive financial, medical and media applications have benefited from grid technologies [4, 5]. These compute and data intensive jobs have in common that they can be handled as batch jobs and therefore can relatively easily be parallelised and distributed. More recently, grid systems have been proposed and investigated to support transactional and interactive applications as well, such as business applications and virtual environments [6, 7]. Many technological challenges still need to be resolved, but this trend illustrates a desire to provide computing, storage and applications as services.

An important driver behind the grid is business and commerce, since the services model opens up the possibility that there is profit to be made from providing computing instead of computers. One such model is an organisation that provides infrastructure for large scale (possibly distributed) computation; other organisations may then access that infrastructure to provide further services to customers. Obviously, such partnerships pose great challenges with respect to trade-off between cost-effective high-quality services. Since all parties in this process depend on the services they use, providers must ensure that their users are receiving a service quality which is acceptable; otherwise they will fail to meet the scientific or commercial aspirations of their clients and will lose business.

At the same time that the grid was developing as the architecture for service-oriented computing, major industry players started pushing it as the answer to another increasingly worrisome computing problem, namely IT management. The vision of companies like IBM and HP is best articulated in IBM's autonomic computing manifesto [8], see also [9]. The fear expressed in this document is that computing systems are becoming so complex that without extensively automated management they can not be maintained. The software platform that is supposed to provide the solutions to automated IT management is the Open Grid Services Architecture [10]. Whether the grid architecture can fulfil all demands put on it are up for discussion [9, 11], but it is clear that integrating autonomous IT management approaches in the grid pose additional challenges to quantitative assessment, optimisation and assurance.

Given all these demands for provable and trusted service quality, we discuss in this paper the assessment, optimisation and assurance of *Grid Performability*, that is, the joint consideration of performance and dependability. The term 'performability' was introduced by John F. Meyer 25 years ago [12] and was initially focussed on the need for a unified measure. Since that time it has become apparent that by considering the combined performance and dependability of systems we achieve greater impact in terms of system effectiveness, efficiency, and adaptability. As Tai *et al* [13] write:

> 'If separate evaluations of performance and dependability are to suffice in determining the overall quality of the delivered service, one must place certain constraints on how properties affecting performance interact with those affecting dependability.'

To illustrate this, it is clear that failure to meet demand functionally, for instance through a period of service unavailability, may have a profound effect on performance metrics such as response times. Vice versa, the techniques used to provide dependable systems, such as fault tolerance, themselves introduce additional overheads. Furthermore, many dependability impacts are themselves due to performance effects, for example the late delivery of an acknowledgement may cause a message to be resent or a transaction to fail. The integration of performance and dependability concerns is therefore a necessary part of a reliable computing solution.

This paper introduces seven performability challenges grid computing poses. These challenges are: (1) virtualisation and service-oriented computing, (2) scale, (3) on-line and run-time algorithms, (4) realistic model parameters, (5) business metrics, (6) unified performance, dependability and security, and (7) standards and tools. These challenges share one encompassing theme, that of *integration*: integration of concerns such as performance, reliability and security, integration of grid systems with on-line and run-time mathematical optimisation techniques, integration of business considerations and IT decisions, and integration (and separation) of views from providers and consumers.

2. Performability Challenges in the Grid

2.1. Virtualisation and Service-Oriented Computing

Virtualisation is an important property of grid systems. It enables a service to be offered seamlessly without a user or service being aware of what resources or underlying services are being used, where they are (either physically or logically), who is providing them or what is using them. This allows services to be provided on top of other services such that the service can be treated and managed as an atomic entity and yet be flexible to a changing provision of constituent services and resources. Thus it provides users with consistency and any application need only be aware of the services it requires and not the services that lie beneath those. From a specification and management perspective this is an ideal situation, however from a measurement and modelling perspective this can provide significant problems [14, 11, 15].

Because of virtualisation it is impossible to always determine what resources are being used, hence two uses of a given service may result in two entirely distinct sets of resources being employed. Furthermore, within the same operation a resource may become overused and a task migrated to an alternative provider, which may have quite different non-functional characteristics. This kind of situation does arise in other areas of distributed systems, for instance in RAID disk arrays, with the distinction that in grid systems the resources are generally heterogeneous and the virtualisation may cross organisational and geographical boundaries. Conversely different services may themselves employ common services to deliver their functionality. Thus two apparently distinct services might actually be strongly correlated and may share common modes of failure. This is not only a problem in analysis, but can also pose a problem when using replication to provide fault tolerance.

An alternative perspective on virtualisation is that it is extremely difficult to predict the workload a particular resource is subject to. This is because, even if it were possible to identify the resource from a service specification, there is no way to identify what else that resource is or could be used for. Seemingly the only viable approach here would be to monitor resource usage directly and build a model of workload from the bottom up, ignoring the complex interplay between compound services.

Thus, at best virtualisation will lead to extremely complex models of system behaviour, at worst, it may become intractable.

2.2. Evaluation of large scale systems

One of the defining features of grid systems is the volume of computation, network and data involved. The very size of these systems therefore creates problems for performability analysis. Grid systems are fundamentally heterogeneous collections of homogeneous systems, i.e. networks of clusters or tightly coupled parallel machines. This heterogeneity of grid systems creates additional, but not insurmountable, problems for measurement and modelling. However;

> "Whatever we think is large and complex now, won't be in the future", Robert Berry, IBM [16].

Berry [16] exemplifies this using a multi-player game example, which involves 10,000 simultaneous users and over 500,000 user communities. Multi-player games are real commercial activities, not only with significant profit to be made by the gaming companies, but also with commercial activity within the user communities trading virtual entities within the game world. IBM demonstrate this with a grid based *Massively Multi-player Online Game* (mmog) demonstration based on Quake2 [16]. Of course, games are not the only such market, but it is one field where a huge user demand is present and will therefore be one driver in the increasing size of grid systems. HP further support the involvement of grids in the entertainment market through their *Utility Data Centers* (UDCs), by providing facilities for small companies to make animated features without the need to invest in costly additional infrastructure [14].

Given the size and complexity of grid systems it is difficult to know what to measure and where to measure it. Furthermore, any realistic model of a grid system is going to be correspondingly complex and involve a very large state space. The model based evaluation of large complex systems requires large simulation programs to be written or complex models to be specified. Even with good tool support at an appropriate level of abstraction this can be a problematic and error prone process. However, once the model has been generated the problems are far from over.

Simulation can be a powerful tool, but its uses are limited. Long run times preclude its use in directing adaptive systems. Analytical modelling is far more potent a tool in this regard, but large models are still costly to solve. It is therefore important to develop techniques which allow large models to be solved efficiently and for simple approximations to be found. Hey et al [17] state that models of grid systems must be "usable, not accurate/sophisticate". Of course, we want both, but in most situations it is consistency and relative accuracy that is important, so an approximation is often just as valuable as an entirely accurate model. A crucial research activity therefore is in producing trustworthy approximations and verifiable techniques for model simplification.

Related to this is the issue of Grid benchmarking; that is, when we have a large system, how do we compare this with another large system (when the hardware is different, the software is different etc.) As we know from [18], there is work in comparing large (single site) systems, such as ASCI Q and EarthSim; but to our knowledge there is very little work on comparing grid implementations. For further information refer to the Grid benchmarking group at the Global Grid Forum [19]

2.3. Online algorithms

Effective management of resources is a crucial part of providing quality of service to customers, just as optimising algorithms is an essential part of getting quality of service from applications. Managing performability requires up to date knowledge of the state of the system operation. This is generally well understood in communication networks, for instance Internet routers exchange information which is used to determine a historical perspective of delay. In grid terms this can be achieved through an *Information Service* (IS), such as MDS [20]. The regularity of updates is crucial, and in volatile systems the approach may work poorly. Of course, being entirely up to date is unreasonable, it is impossible to constantly exchange status information across the system without seriously harming the overall performance. Instead the choice of where to direct a particular request must be based on the best information available. One approach is to poll potential resources at the request instant, however if there are many requests then this may be a significant bottleneck, and there will still be a delay on the status being reported and it being acted on. Some predictive mechanism may therefore be required (see [21, 22]).

In order to predict future availability from past trends requires the evaluation of some algorithm. As has already been observed, in the case of grid systems the models used may be highly complex. Evaluating such a model directly is unlikely to feasible in real time. Therefore it is important that efficient decomposition techniques, accurate approximations and scenario specific heuristics, are developed to facilitate fast evaluation of scheduling and related problems. For example, Palmer and Mitrani [23] have analysed the problem of dynamic server reallocation based on current demand. They determine an optimal policy for this problem, but this is computationally expensive, so a simpler approach is developed which can be verified against the exact model.

Where heuristics have been developed it is important to know how close to an optimal policy has been achieved. Therefore it becomes important to enable the analysis of grid systems by developers and not just modelling experts. This implies the provision of applied modelling tools that rely on application specific rather than modelling specific knowledge. Because of the component like nature of services it is likely that modelling formalisms that support a compositional approach are likely to be best suited to this task. Although compositional formalisms exist and are widely used, much work remains to provide application oriented mechanisms that allow these tools to be used easily by developers. In addition the development of case studies provides a basis, not only for future application oriented approaches, but also for further case studies by association.

2.4. Population of systems with realistic parameterisation

Performability models are only as good as the data that is used to populate them. If performance or availability is predicted on a conservative estimate for user demand then the system may have too little capacity and a far poorer expected performability. Thus it is important to have accurate information *on demand* and for proposed models to be accurately verified against real (or at least realistic) data. One way to do this is to use logging or bookkeeping data. This allows the system to be fed with a trace of actual grid usage, promoting decision making on the basis of experience.

One area where this issue has been taken up is in grid scheduling, see for example [21, 24, 25]. Using resource management systems, such as Condor [3], it is possible to get an up to date view of resource usage. For a wider view the *Network Weather Service* (NWS) [26, 27] allows the response from remote systems to be tracked. Various systems, such as [28] and [29], have used NWS to populate predictive models for future application deployment. The ICENI project has taken a broader approach [30, 31], providing a *performance repository* where relevant performance data is collected in any viable manner. Thus this system is easily extensible to incorporate any performance service available.

Filippopoulos and Karatza [32] propose a model where alternative clusters are available to service jobs and the load must be balanced across these to optimise performance. Some jobs must be co-allocated to both clusters simultaneously and these are treated with a different priority. The authors propose a local scheduling policy which is simple to implement but is shown to perform well below optimal. Thomas et al [33] have analysed a model of services which fail. There are several mechanisms by which a failure can be detected. However, all these mechanisms carry an overhead which, by the regularity of the required messaging, may be significant. Such messaging can be minimised by extending the intervals between messages; this however causes a delay between when a failure occurs and when the next communication takes place (or is expected). The length of the delay will be variable depending on the precise moment of failure relative to the expected time of the next message. Thus, there is a *staleness* to availability information which can have a significant effect on both response time and reliability.

An alternative approach is to use reservation to give greater confidence that resources are available when required. Djemame et al [34, 35], have developed a SNAP based three phase commit reservation protocol which they show, through analytical modelling, simulation and experimentation, can significantly improve both response time and utilisation. McGough et al [31] also employ a reservation mechanism, which they demonstrate as part of the ICENI system. This mechanism has been shown through experimentation to improve the predictability of service requests, hence increasing the applicability of service level agreements. This is the only work we are aware of that currently links resource monitoring with adaptive behaviour of resource usage in a fully automated way.

Despite the amount of active work in this area there is still much to be done to provide the right level of information across a wide range of systems (both functionally and geographically) in an accurate and timely manner. Furthermore, new applications require accurate historical data from similar applications in order to make accurate performability predictions. Thus far these activities, whilst significant, have been limited in scope, however this is one area at least where progress is continuing to be made. Open issues include:

1. What performance information do we need to drive these systems?
2. How accurate does that information need to be?
3. How 'live' (old) does it need to be?
4. How does the usefulness (of this data) degrade over time?

2.5. Evaluation of business metrics and business models

As stated earlier, one of the key drivers for grid system development is the commercial sector, particularly in the fields of finance and media. Given that this is the case it is clear that the real metrics of interest are in fact monetary. Increasing performability inevitably introduces cost. Therefore there is a clear need to incorporate cost models alongside performability estimation in order to investigate this trade off.

From a business perspective grid systems are not simply a technical solution, but rather a different way of organising business. Berry [16] identifies horizontal business integration as a key driver in the IBM *On Demand* initiative to provide greater business flexibility through grid computing. Systems such as HP's Utility Data Centers [14] are business oriented systems. Therefore the core model is going to be a business process model and the technical models, such as performability models, are likely to be add-ons to this. This implies a much greater emphasis on business process and technical modelling than has hitherto been applied.

Given a potential for profit from grid computing services there is a need for greater understanding of charging models and their impact on user behaviour and system performance. Such models are not new; indeed charging has been a part of computer communications and mainframe access for 40 years [36] and some efforts have been made to apply charging to limited grid scenarios [37]. Harder et al [38] discuss various potential market mechanisms for the grid and define a dynamic model of a grid market. Within this model service providers and *middlemen* (e.g. service brokers) attempt to set prices in order to maximise profit, whereas end users will adapt their behaviour when the service quality drops too low or the service price becomes too high. The *Grid Economic Services Architecture Working Group* have worked on defining protocols necessary for setting the cost of such services. These efforts are described by Newhouse et al [39], who also describe experimentation within the UK e-Science Programme with the Chargeable Grid service, the Resource Usage Service, and the Grid Banking Service.

The choice of charging model depends greatly on the application area and the expected user demand, but it can also have a profound impact on performance. To take a road traffic analogy; congestion charging was introduced in London not only to generate revenue, but principally to reduce peak user demand and hence provide a better quality of service to those prepared to pay, without the need for large scale investment in extra capacity. A good, well understood pricing structure can provide a means for regulating user access so that quality of service is maintained and revenue generated in computational systems. Clearly users are only going to be prepared to pay for a service that offers a sufficiently high performability. The relationship between charging and performability is therefore clearly highly complex.

Federated service provider and business-to-business computing implies that various forms of agreements will be in place, from service level agreements about infrastructure QoS to financial contracts about business deals [7, 40, 41]. Such agreements directly impact the decisions a service provider makes if resources are scarce or changes are requested. If such decision-making is automated, agreements should be encoded and maintained in the software platform, and should be input to the decision-making algorithms [42]. As a consequence, continuing research is needed in architectural such as in the GGF working group working on WS Agreement [43], modelling [44, 45] and algorithmic issues around agreements [7].

2.6. Performability: integrating performance, dependability and security

Initial scientific development of grid systems involved sharing large valuable data sets. The owners of these data sets are often unwilling to allow open access to the data because its capture has been costly and its existence may represent a significant potential scientific and commercial advantage. Protecting this advantage thus became a key issue in the development of grid systems in order to facilitate scientific exchange. The more recent development of commercial applications has further emphasised the need for system security.

Making open systems secure is clearly not a simple matter and it can involve a considerable communication and computational overhead. Developers encountering a performance problem may often observe that security measures form a significant bottleneck. They may even be tempted to turn off some security measures to alleviate the performance pressure.

It would appear that security and performability are orthogonal, however, this need not necessarily be the case. Indeed it would be disingenuous to suggest that security developers do not consider performance, however performance is understandably a secondary consideration to them. One problem is that the relationship between security and performance is not well understood. Very little work has been done in this area, but it is vital for systems (such as grid systems) where security and performance are crucial discriminating factors. Recently work has been done on applying stochastic extensions of classical process algebra to problems in security analysis concerning scalability and timing [46, 47], however this has not yet been applied to grid security.

Security is just one example of an additional service that must be employed in the running of an application. However as more services are added we encounter *the software stack problem* - that is of performance degrading as the size of the software stack increases, and the problem of monitoring (and assigning costs) when the stack is so large.

2.7. Standards and tools

The definition of standards has played an important role in the development of distributed systems, and thus also in the Grid. Standards are required for products from multiple vendors to interoperate, and successful standards have the potential to increase both vendor markets and purchaser flexibility.

With respect to performability assessment and optimisation, a good case can be made for the need of standards and tools with open interfaces. As an example, intrusion detection can not be done without good and pervasive logging mechanisms [48], let alone be benchmarked [49]. Optimising resource allocation can not be done without data about resource usage, and automation of SLA management is not possible without agreed upon monitoring interfaces and data collection [40].

In the area of Grid technologies, the Global Grid Forum (GGF) [50] brings together researchers and industry to initiate the standard process, leading to *de facto* standards for many areas of grid computing. Various efforts with a relation to performability are underway, such as the definition of benchmarks [19], common information models for resources [51] and common monitor and management interfaces. The latter has been moved to the OASIS web services standardisation working group on web service distributed management [52], the first group in which Grid and web services came together.

Important for the advancement of the Grid is the open source platform Globus [53], and recently the *Open Middleware Initiative* (OMI) [54] has been formed in the UK to develop and distribute stable and open tools for developing grid services. Several services based on performability measurement and prediction [55, 56] are currently being deployed. Other network monitoring tools, such as the Network Weather Service [27] and GridMon [57] are already available. Although various more or less tangential standards and tools efforts are already underway, a number of issues remain:

- Ontologies for specific performability assessment and benchmarking problems (such as intrusion detection, resource allocation).
- Choice between passive ('always-on') or active ('only when activated') monitoring infrastructures and standards.
- Agreement on protocols to exchange and query the monitored data across virtual organisations.
- Liability and responsibility of monitoring and data maintenance.

3. Conclusions

Grid computing offers a new dimension to many existing distributed systems questions. Some problems that have been well known for years have re-emerged and once again become important. A smaller number of entirely new ones have also come to light. Grid computing is something new in terms of complexity and scale of distributed systems, but it is also an area of great opportunity. Grid systems will rely on high performance and high dependability of services, but there is also the chance to build performability at the heart of the system through middleware and performability services.

Grid computing is not so much the methods that are being employed, but the highly complex interaction between applications, resources and middleware. The fact is that each of these areas has its own open performance issues and there are likely to be a number of solutions that can be employed in each environment. It is difficult to fix any one of these variables and there is no *Green Room* in which parts of this performability analysis can be rehearsed makes this a hugely complex problem. The lesson from performability is that there is no one solution.

In this paper seven key areas of research have been identified within and around performability in grid computing. These areas lead to many challenges which will need to be met by the performability community if grid systems are to be developed which meet expectation. Many of these challenges are already being addressed in some way and some of that work has been highlighted here. Many other challenges are not yet being addressed. This may be in part because there is as yet little idea as to how to solve the underlying problems, but in many respects these issues are only now becoming recognised in the mainstream of grid development. Until recently the emphasis of grid development has been towards a proof of concept. Now the focus has moved more towards production systems, in which performability has become a much more significant constraint. It is the integration and understanding of all these viewpoints that remains the real challenge.

Acknowledgements

This paper is broadly based on an earlier document [58] that arose from discussions during and following the Workshop on Grid Performability Modelling and Measurement held at the National eScience Centre, Edinburgh, UK, on 23rd March 2004. The authors would like to thank all the delegates at that workshop and particularly those who made presentations.

References

[1] A. Reinefeld and F. Schintke, Concepts and Technologies for a Worldwide Grid Infrastructure, in *LNCS 2400: Euro-Par 2002 Parallel Processing,* pp. 62–71, Springer Verlag, Berlin, Germany, 2002.

[2] W. E. Johnston, D. Gannon, B. Nitzberg, Grids as Production Computing Environments: The Engineering Aspects of NASA's Information Power Grid, *Eighth IEEE International Symposium on High Performance Distributed Computing*, p. 34, 1999.

[3] D. Thain, T. Tannenbaum and M. Livny, Condor and the Grid, in: F. Berman, G. Fox, and A. Hey (eds.), *Grid Computing: Making the Global Infrastructure a Reality*, Wiley and Sons, March 2003.

[4] Tabb Group, *Grid Computing in Financial Markets: Moving Beyond Compute-Intensive Applications*, November 2003.

[5] A. Cowen, 422 Fires Up HP Rendering Service: Rapid, Resource-Savvy, UDC-Based Animation, *HP Mpulse*, August 2003. *http://www.cooltown.com/cooltown/mpulse/0803-framefactory.asp*

[6] P. Goldsack, J. Guijarro, A. Lain, G. Mecheneau, P. Murray and P. Toft, SmartFrog: Configuration and Automatic Ignition of Distributed Applications, *10th OpenView University Association Workshop,* Geneva, June 2003.

[7] K. Appleby, S. Fakhouri, L. Fong, G. Goldszmidt, M. Kalantar, S. Krishnakumar, D. P. Pazel, J. Pershing and B. Rochwerger, Oceano–SLA Based Management of a Computing Utility, *Proceedings of the 7th IFIP/IEEE International Symposium on Integrated Network Management*, IEEE, 2001.

[8] P. Horn, *Autonomic Computing: IBM's Perspective on the State of Information Technology,* IBM, USA, 2002. *http://www.research.ibm.com*

[9] Bloor Research, The Grid Report: The Commercial Implications of the Convergence of Grid Computing, Web Services, and Self-Managing Systems, *Bloor Research North America*, August 2002.

[10] I. Foster, C. Kesselman, J. Nick and S. Tuecke, The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration, *Globus Project,* 2002.

[11] A. van Moorsel, Grid, Management and Self-Management, *The Computer Journal*, to appear, 2005.

[12] J.F. Meyer, On evaluating the performability of degradable computing systems, *IEEE Transactions on Computers*, vol. C-90, pp. 720-731, 1980.

[13] A. Tai, J.F. Meyer and A. Avizienis, *Software Performability: From Concepts to Applications*, Kluwer Academic Publishers, 1996.

[14] A. van Moorsel, Grid and Performability, Workshop on Grid Performability Modelling and Measurement, National eScience Centre, Edinburgh, March 2004. *http://www.nesc.ac.uk/action/esi/contribution.cfm?Title=379*

[15] A. Sahai, S. Graupner, V. Machiraju and A. van Moorsel, Specifying and Monitoring Guarantees in Commercial Grids through SLAs, *International Symposium on Cluster Computing and the Grid, CCGrid 2003,* Tokyo, Japan, 2003.

[16] R. Berry, Performability and Grid in an On Demand World, Workshop on Grid Performability Modelling and Measurement, National eScience Centre, Edinburgh, March 2004. *http://www.nesc.ac.uk/action/esi/contribution.cfm?Title=379*

[17] A. Hey, J. Papay and M. Surridge, Performance Engineering the Grid, Performance Engineering Workshop, National eScience Centre, Edinburgh, December 2002. *http://www.nesc.ac.uk/action/esi/contribution.cfm?Title=127*

[18] D. Kerbyson, A. Hoise and H.J. Wasserman, Modelling the performance of large-scale systems, *IEE Proceedings - Software*, **150**(4), 2003.

[19] Grid Benchmarking Research Group, *http://www.nas.nasa.gov/GGF/Benchmarks/*

[20] K. Czajkowski, S. Fitzgerald, I. Foster and C. Kesselman, Grid information services for distributed resource sharing, in: *10th International Symposium on High-Performance Distributed Computing*, IEEE Press, 2001.

[21] J.M. Schopf, A General Architecture for Scheduling on the Grid, *Argonne National Laboratory preprint*, ANL/MCS-P1000-1002, 2002.

[22] H. Lim Choi Keung, J. Dyson, S.A. Jarvis and G.R. Nudd, Performance Evaluation of a Grid Resource Monitoring and Discovery Service, *IEE Proc.-Software*, **150**(4), pp. 243–251, August 2003.

[23] J. Palmer and I. Mitrani, Optimal Server Allocation in Reconfigurable Clusters with Multiple Job Types, *School of Computing Science Technical Report CS-TR: 823, University of Newcastle*, 2003.

[24] B. Nitzberg and J.M. Schopf, Current Activities in the Scheduling and Resource Management Area of the Global Grid Forum, in: D.G. Feitelson, L. Rudolph and U. Schwiegelshohn, *Job Scheduling Strategies for Parallel Processing*, Lecture Notes in Computer Science 2537, Springer Verlag, 2002.

[25] D.P. Spooner, S.A. Jarvis, J. Cao, S. Saini and G.R. Nudd, Local grid scheduling techniques using performance prediction, *IEE Proceedings Computers and Digital Techniques*, **150**(2), pp. 87-96, 2003.

[26] R. Wolski, L. Miller, G. Obertelli and M. Swany, Performance Information Services for Computational Grids, in: J. Nabrzyski, J. Schopf and J. Weglarz (eds.), *Grid Resource Management: State of the Art and Future Trends*, Kluwer Academic Publishers, 2003.

[27] R. Wolski, Experiences with Predicting Resource Performance On-line in Computational Grid Settings, *Performance Evaluation Review*, **30**(4), pp. 41–49, 2003.

[28] A. Benoit, M. Cole, S. Gilmore and J. Hillston, Evaluating the performance of skeleton based gird programs, Workshop on Grid Performability Modelling and Measurement, National eScience Centre, Edinburgh, March 2004. *http://www.nesc.ac.uk/action/esi/contribution.cfm?Title=379*

[29] S. Akioka and Y. Muraoka, The Markov Model Based Algorithm to Predict Networking Load on the Computational Grid, *Journal of Mathematical Modelling and Algorithms*, **2**, pp. 251-261, 2003.

[30] M.Y. Gulamali, A.S. McGough, S.J. Newhouse and J. Darlington, Using ICENI to run parameter sweep applications across multiple Grid resources, in: *Global Grid Forum 10, Case Stud-*

ies on Grid Applications Workshop, Berlin, Germany, March 2004.

[31] A.S. McGough, Gathering and using Performance information within ICENI, Workshop on Grid Performability Modelling and Measurement, National eScience Centre, Edinburgh, March 2004. *http://www.nesc.ac.uk/action/esi/contribution.cfm?Title=379*

[32] D. Filippopoulos and H. Karatza,On the Performance Benefits of Dynamically Co-allocating Parallel Jobs in Heterogeneous Clusters, in: I. Awan (ed.) *Proceedings of the 20th Annual UK Performance Engineering Workshop*, University of Bradford, 2004.

[33] N. Thomas, J. Bradley and W. Knottenbelt, Stochastic Analysis of Scheduling Strategies in a Grid-based Resource Model, *IEE Proceedings - Software*, **151**(5), 2004.

[34] M. Haji, I. Gourlay, K. Djemame and P. Dew, A SNAP-based Community Resource Broker using a Three-Phase Commit Protocol: a Performance Study, *The Computer Journal*, to appear, 2005.

[35] I. Gourlay, M. Haji and K. Djemame, Performance Evaluation of a SNAP-based Grid Resource Broker, in: *Proceedings of the First European Performance Engineering Workshop*, LNCS 3236, Springer-Verlag, 2004.

[36] I.E. Sutherland, A futures market in computer time, *Communications of the ACM*, **11**(6), pp. 449–451, 1968.

[37] R. Wolski, J. Brevik, J. Plank and T. Bryan, Grid Resource Allocation and Control Using Computational Economies , in: F. Berman, G. Fox, and A. Hey (eds.), *Grid Computing: Making the Global Infrastructure a Reality*, Wiley and Sons, pp. 747-772, March 2003.

[38] U. Harder, P.G. Harrison, Maya Paczuski and Tejas Shah, A dynamical model of a GRID market, The Computing Research Repository (CoRR), October 2004. *http://aesop.doc.ic.ac.uk/pubs/cs-oct04-grid/*

[39] S. Newhouse, J. MacLaren and K. Keahey, Trading Grid Services Within the UK e-Science Grid, in: J. Nabrzyski, J.M. Schopf and J. Weglarz (eds.) *Grid Resource Management: State of the Art and Future Trends*, Kluwer Academic Publishers, 2003.

[40] V. Machiraju, A. Sahai and A. van Moorsel, Web Services Management Network: An Overlay Network for Federated Service Management, *Proceedings of International Symposium on Integrated Network Management, IM2003*, Kluwer, March 2003.

[41] C. Bartolini, A. Boulmakou, A. Christodoulou, A. Farrell, M. Salle and D. Trastour, Management by Contract: IT Management driven by Business Objectives, *HP Labs Technical Report, HPL-2004-184*, 2004. *http://www.hpl.hp.com/techreports/2004/HPL-2004-184.html*

[42] C. Molina-Jimenez, J. Pruyne and A. van Moorsel, Software Architecture Approaches for Agreements and Contracts, *Architecting Dependable Systems,* R. de Lemos, C. Gacek and A. Romanoski (Eds.), Springer LNCS, 2005, to appear.

[43] Grid Resource Allocation Agreement Protocol Working Group, *http://forge.gridforum.org/projects/graap-wg*

[44] A. Farrell, M. Sergot, M. Salle, C. Bartolini, D. Trastour and A. Christodoulou, Performance Monitoring of Service-Level Agreements for Utility Computing Using the Event Calculus, *HP Labs Technical Report, HPL-2004-20R1*, 2004. *http://www.hpl.hp.com/techreports/2004/HPL-2004-20R1.html*

[45] C. Molina-Jimenez, S. Shrivastava, E. Solaiman, and J. Warne, Run-time Monitoring and Enforcement of Electronic Contracts, *Electronic Commerce Research and Applications,* Elsevier, **3**(2), pp 108-125, 2004.

[46] N. Thomas, Performabilty of a secure voting algorithm, in: Proceedings of the Workshop on Practical Applications of Stochastic Modelling, *Electronic Notes in Theoretical Computer Science*, Elsevier, 2004.

[47] M.Buchholz, S.Gilmore, J.Hillston and F.Nielson, Securing statically-verified communications protocols against timing attacks, in: Proceedings of the Workshop on Practical Applications of Stochastic Modelling, *Electronic Notes in Theoretical Computer Science*, Elsevier, 2004.

[48] F. Salfner, S. Tschirpke and M. Malek, Comprehensive Logfiles for Autonomic Systems, in *18th International Parallel and Distributed Processing Symposium - Workshop 11*, p. 211b, April 2004.

[49] Maxion, Roy A. and Tan, Kymie M. C. Benchmarking Anomaly-Based Detection Systems, in *International Conference on Dependable Systems and Networks*, IEEE, pp. 623–630, June 2000.

[50] The Global Grid Forum, *http://www.gridforum.org/*

[51] GGF CIM-based Grid Schema Working Group (CGS), *http://www.daasi.de/wgs/CGS*

[52] OASIS Web Services Distributed Management Technical Committee (WSDM), *http://www.oasis-open.org/committees/tc_home .php?wg_abbrev=wsdm*

[53] Globus, *http://www.globus.org*

[54] The Open Middleware Infrastructure Initiative, *http://www.omii.ac.uk/*

[55] D. Spooner, J. Cao, S. Jarvis, L. He and G. Nudd, Performance-aware Workflow Management for Grid Computing, *The Computer Journal*, to appear, 2005.

[56] N. Looker, M. Munro and J. Xu, Simulating Errors in Web Services, *International Journal of Simulation: Systems, Science and Technology*, this issue.

[57] GridMon - Grid Network Performance Monitoring for UK e-Science, *http://www.gridmon.dl.ac.uk/*

[58] N. Thomas, Challenges and Opportunities in Grid Performability, *Technical Report, CS-TR: 842, School of Computing Science, University of Newcastle upon Tyne*, 2004.