An Autonomic Workflow Management System for Global Grids

Mustafizur Rahman and Rajkumar Buyya
Grid Computing and Distributed Systems (GRIDS) Laboratory
Department of Computer Science and Software Engineering
The University of Melbourne, Australia
{mmrahman,raj}@csse.unimelb.edu.au

Abstract

Workflow Management System is generally utilized to define, manage and execute workflow applications on Grid resources. However, the increasing scale complexity, heterogeneity and dynamism of Grid environment that includes networks, resources and applications have made such workflow management systems brittle, unmanageable and insecure. Autonomic computing provides a holistic approach for the design and development of systems/applications that can adapt themselves to meet requirements of performance, fault tolerance, reliability, security, etc., without manual intervention. Therefore, this research aims to design and develop mechanisms for building an autonomic workflow management system that will incorporate the properties of autonomic computing and exhibit the ability to reconfigure itself to the changes in the Grid environment, discover, diagnose and react to the disruptions of workflow execution, and monitor and tune Grid resources automatically.

1. Introduction

Since many of the large-scale scientific applications executed on present-day Grids are expressed as complex scientific workflows [1], workflow management has emerged as one of the most important Grid services in past few years. Scientific workflows that are also known as Grid workflows, can be defined as the aggregation of grid application services which are executed on distributed heterogeneous resources in a well defined order to satisfy the specific requirements of users. A Workflow Management System (WMS) [2] is generally employed to define, manage and execute these workflow applications on Grid resources.

However, the increasing scale complexity, heterogeneity and dynamism of Grid environment that includes networks, resources and applications have made such workflow management systems brittle, unmanageable and insecure. Autonomic computing provides a holistic approach for the design and development of systems/applications that can adapt themselves to meet requirements of performance, fault tolerance, reliability, security, etc., without manual intervention. Therefore, this research aims to design and develop mechanisms for autonomic workflow management that will enable a workflow management system to incorporate the properties of autonomic computing and exhibit the ability to protect itself, recover from faults, reconfigure as required by changes in the environment, and always maintain its operations at a near optimal performance.

1.1. Motivation and Problem Statement

In the current approaches to workflow scheduling, there is no cooperation between the distributed workflow brokers [3][4]. As a result, the problem of conflicting schedules can occur. For example, consider a Grid environment (as shown in Figure 1) that consists of n number of resources and m number of workflow brokers. Users submit their scientific applications to the workflow brokers. These brokers generate the schedules based on the resource information obtained from the Grid Information Services (GIS). A schedule is effectively defined as the mapping of a set of tasks in the workflow to a set of available resources. However, if Workflow Broker 1 and Workflow Broker 2 query the GIS at the same time, they will get the similar information about the resource availability pattern. Based on this information, Workflow Broker 1 and Workflow Broker 2 will generate the same mapping for the tasks in their locally submitted workflows, which will lead to conflicting schedules. Therefore, both the system and the application will suffer from degraded performance if the resources are heavily loaded.

Other major drawback involved with current workflow scheduling is that the existing workflow brokers rely on the centralized (refer to Figure 1) or semi-centralized hierarchical resource information services such as MDS-2,3,4 [5]. Current studies have shown that [6] the existing centralized model for information services do not scale well as the number of users, brokers and providers increase in the system. Hence in case, the centralized links leading to these services fail, then no broker in the system can undertake scheduling related activities due to the lack of up-to-date resource information. In addition, considering the sheer dynamism of Grid computing environment, any scheduling decision that is based on static resource information, would certainly not be an efficient one.

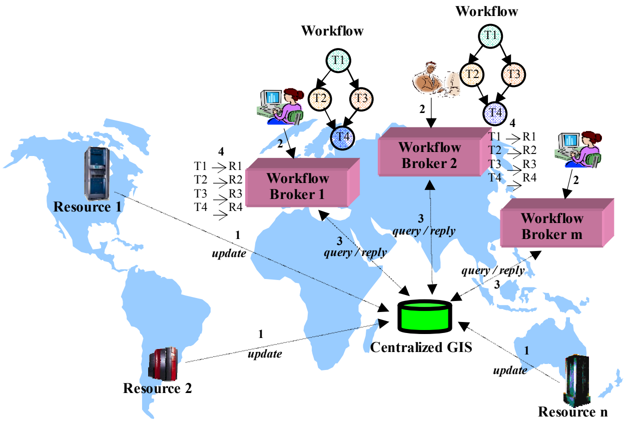

Figure 1. Existing workflow scheduling approach

Further, Grids [7] are heterogeneous and dynamic environments consisting of computing, storage and network resources with different capability and availability. In our previous work [8], it is shown that the dynamic scheduling algorithms based on heuristics adapt to the changing resource conditions of Grids by performing just-in-time scheduling (generating schedules that maps the tasks dynamically). But the workflow brokers running these dynamic algorithms, still need to be coordinated in order to avoid any conflict and generate efficient schedule globally.

To overcome the limitation of existing approaches, fully decentralized and cooperative workflow scheduling and resource discovery service are needed. However, decentralization of a system leads to an increased complexity in managing the system. The challenge is that many decentralized systems make central or global control impossible because the information needed to make decisions cannot be gathered centrally. Autonomic Computing (AC) [9] aims to create computer systems capable of self-management, to overcome the rapidly growing complexity of computing systems management. Thus AC is essential to keep the aforementioned decentralized systems manageable. However, in this context, AC is only possible when the decentralized entities autonomously coordinate with each other to maintain the self-* properties [10].

Apart from the limitations of the centralized workflow scheduling approach discussed above, there are some other challenges regarding the management of the workflow applications in a dynamic heterogeneous Grid environment. Addressing these challenges in a workflow management system with respect to the dynamism in Grids consequently raises the following questions:

- How can changes to the workflows at run time be accommodated?
- How can flexibility and adaptability of the running workflows be increased?
- How can workflow management system diagnose and react to any disruptions occurred during the execution of the workflows?

This research aims to address the aforementioned issues by proposing an autonomic workflow management system based on a decentralized and cooperative scheduling and resource discovery architecture. Our objective is to develop mechanisms for autonomic workflow management that will enable a workflow management system capable of adapting automatically to the dynamically changing Grid environment (self-configuring), discovering, diagnosing and reacting to disruptions of workflow execution (self-healing), monitoring and tuning Grid resources automatically (self-optimizing), and anticipating, detecting, identifying and protecting itself from malicious attacks or cascading failures to maintain overall system security and integrity (self-protecting).

The rest of the paper is organized as follows. In the next section, we present the research issues regarding autonomic workflow management. In Section 3, the existing work related to workflow management system as well as autonomic systems are described in brief. The proposed methodology is discussed in Section 4. Experiment details and some preliminary simulation results are presented in Section 5. Section 6 presents the research plan in regards to the proposed work. Finally, we conclude the paper by stating the timeline for completion of the future work in Section 7.

2. Research Issues

The key challenges that are required to be addressed in various aspects of autonomic workflow management are as follows.

Coordination of Workflow Brokers:

- What kind of coordination mechanisms need to be adopted which will ensure effectiveness and allow the scalability and reliability of cooperative workflow brokers?
- How can the workflow brokers reconfigure themselves when policies and components are added or removed from the system due to the changes in the environment?

***System monitoring*:**

- How can the autonomic element in the workflow brokers identify itself, discover and verify the identities of other entities of interest dynamically establish relationships with these entities and interact in a secure manner?

Failure management:

- How can workflow management system diagnose and react to any disruptions such as resource failure occurred during the execution of workflows?
- How can the system automatically anticipates, detects, identifies and protects against malicious attacks or cascading failures to maintain overall system security and integrity?

Runtime optimization:

- How can changes to the workflows at run time be identifies, accommodated and the execution to be optimized accordingly?

3. Related Work

Over the last few years, a number of Grid Workflow Management systems such as Pegasus [11], Triana [12], Taverna [13], Condor DAGMan [14], Kepler [15], Gridbus [3] and Askalon [4] have been developed by projects around the globe. These systems focus on various aspects of workflow management including workflow expression language, graphical environment for workflow composition and execution monitoring, workflow scheduling heuristics, data management, legacy applications and fault-tolerant mechanisms.

Among these systems, Triana supports decentralized Peer-to-Peer (P2P) based workflow management. However, the P2P communication in Triana is implemented by JXTA protocol which uses broadcast technique. In our previous work [23], we use a DHT (such as Chord, Pastry, CAN) based P2P system for handling resource discovery and scheduling coordination. The employment of DHT gives the system the ability to perform deterministic discovery of resources and produce controllable number of messages in comparison to using JXTA.

With regards to Autonomic Computing paradigm, several research efforts have focused on enabling the autonomic properties into the system by addressing four main areas: self-healing, self-protection, self-configuration, and self-optimization. Projects in both industry and academia such as OceanStore [16], Storage Tank [17], Oceano [18], AutoAdmin [19], Q-Fabric [20], have addressed autonomic behaviors at all levels such as hardware, software systems and applications. At the hardware level, systems may be dynamically upgradeable, while at the operating system level, active operating system code may be replaced dynamically. Moreover, at the application level, self-optimizing databases and web servers may be dynamically reconfigured to adapt service performance. In contrast, our work proposes to address these autonomic behaviors into the workflow management system.

4. Proposed Work

In order to develop an autonomic workflow management system based on a decentralized and cooperative scheduling and resource discovery architecture, we propose a fully decentralized and cooperative workflow scheduling algorithm based on a P2P coordination space. The P2P coordination space provides a global virtual shared space that can be concurrently and associatively accessed by all the participants in the system and this access is independent of the actual physical or topological proximity of the objects or hosts. New generation routing algorithms, which are more commonly known as the Distributed Hash Tables (DHTs) [21] form the basis for organizing the P2P coordination space.

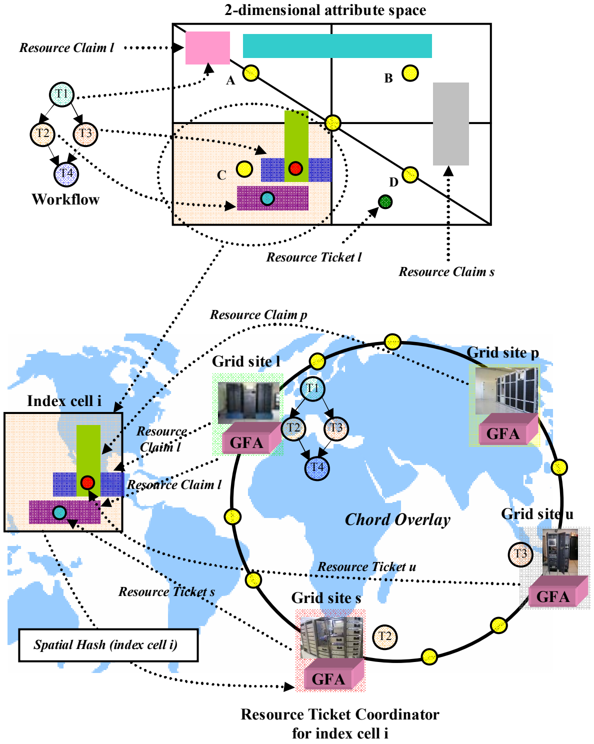

Figure 2. Resource allocation and workflow scheduling coordination across the Grid sites

In the proposed approach, workflow brokers post their resource demands by injecting a ‘Resource Claim’ object into the DHT-based decentralized coordination space, while resource providers update the resource information by injecting a ‘Resource Ticket’ object. These objects are mapped to the DHT-based P2P coordination space using a spatial hashing technique [22]. Once a resource ticket matches with one or more resource claims, the coordination space sends notification messages to the resource claimers such that it does not lead to the overloading of the concerned resource ticket issuer. Thus, this mechanism prevents the workflow brokers from overloading the same resource. More details of this decentralized and cooperative workflow scheduling strategy can be found in [23].

Further, we propose to build the autonomic workflow management system by leveraging the scalable and self-organizing nature of the above mentioned decentralized workflow scheduling approach. This proposed workflow scheduling approach utilizes the Grid-Federation [24] model in regards to resource organization and Grid networking. Grid-Federation aggregates distributed resource brokering and allocation services as part of a cooperative resource sharing environment. At every site in the federation, there is a Grid Federation Agent (GFA) that is responsible for managing the execution of the workflow applications submitted by the users (see Figure 2). These GFAs can be extended to have the properties of an autonomic system by adding the autonomic component [9] (refer to Figure 3). Therefore, the Grid sites managing the execution of workflows will be able to monitor and adapt with the system, which will enable them to protect themselves from malicious attacks, recover from faults, reconfigure as required by changes in the environment, and maintain optimized operations.

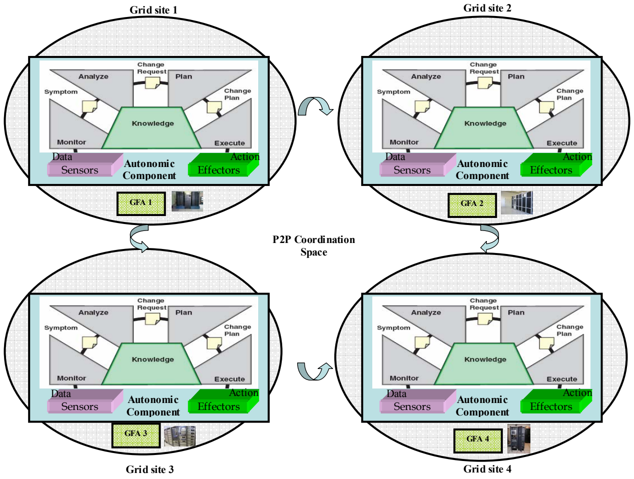

Figure 3. Decentralized autonomic workflow management

5. Preliminary Results

To study the proposed scheduling approach, we use simulation where the simulation infrastructure is created by combining two discrete event simulators namely GridSim [25], and PlanetSim [26]. The details of the simulation experiments and results are stated in [23]. In the simulation, we consider that the users submit e-Science workflow applications (fork-join workflow) that are executed in the Grid resources. An example of fork-join workflow is WIEN2K [1], which is a quantum chemistry application developed at Vienna University of Technology. We vary the number of tasks in a workflow from 20 to 100 and the size of each task is randomly generated from a uniform distribution between 50000 MI (Million Instruction) and 500000 MI.

The resources in the Grid environment are modeled by utilizing the configuration of resources that are deployed in various Grids including NorduGrid, AuverGrid, Grid5000, NaregiGrid, and SHARCNET [27]. Number of resources (GFA/broker) in the simulation is fixed to 100. The GFAs inject the ticket objects based on the exponential inter-arrival time distribution. The injection rate for the resource tickets is distributed over the interval [100, 300] in step of 100 seconds.

In order to evaluate the performance of the proposed scheduling approach, we measure the following metrics: (i) coordination delay which is the difference between the submission and execution start time of a task (ii) makespan which equals to the difference between the submission time of the entry task in the workflow and the output arrival time of the exit task in that workflow (iii) total number of messages generated in the system for successfully mapping the coordination objects and receiving notifications.

The results (refer to [23]) show that at higher inter-arrival delay of tickets, the tasks in a workflow experience increased coordination delay. The average makespan of the workflows also shows the similar growth over number of tasks and ticket inter-arrival delay as reflected in coordination delay. Thus in our proposed scheduling environment, if the resources update their availability information frequently, then the workflows submitted by the users will be completed early. However, if ticket inter-arrival delay and size of the workflow increase, the number of messages generated during simulation period increases. Therefore, the ticket inter-arrival delay should be chosen in such a way that a balance between coordination delay and message overhead can exist in the system.

According to the preliminary results obtained, it is evident that our approach is scalable in terms of scheduling message complexity and makespan. As we leverage the DHT-based coordination space for our scheduling, it can also avoid the limitation of single point of failure with respect to resource information service in centralized scheduling techniques.

6. Research Plan

6.1 Decentralized Workflow Scheduling Algorithm

In order to develop an autonomic workflow management system based on the decentralized and cooperative scheduling and resource discovery architecture, we propose a decentralized and cooperative workflow scheduling algorithm. The proposed approach utilizes a P2P coordination space in order to coordinate the application schedules among the Grid wide distributed workflow brokers. The proposed algorithm is completely decentralized in the sense that there is no central point of contact in the system. The functions of the main components (i.e. resource discovery and scheduling coordination) of the system are delegated to the P2P coordination space. With the implementation of this approach, efficient scheduling with enhanced scalability and better autonomy for the users are likely to be achieved.

6.2 Incorporation of Self-management Policies

In order to incorporate the autonomic behavior into the proposed workflow management system, we propose to explore the scalable and self-organizing nature of the decentralized workflow scheduling architecture. We also propose to develop a workflow task failure handling strategy to enable the system recovering from the failures by automatically discovering, diagnosing and circumventing from the issues that might cause service disruptions. We also propose to develop an optimization technique for the execution of workflow so that the system can continuously tune itself and adapt to the changes in the workload as well as dynamic nature of the interconnections among the entities in the environment.

6.3 Simulation Model

The work, mentioned above will be carried out through simulations using the decentralized workflow simulator based on GridSim [25]. Simulation can demonstrate the behavior of proposed scheduling approach and the effectiveness of self-management policies through the modeled Grid environment.

Simulation will be conducted using both real and synthetic workload data. The real workload data can be collected from high performance computing centers for a specific period of time and stored as trace files. By using real workload trace data, we can demonstrate the effectiveness of our proposed scheduling approach and self-management policies in meeting the actual demands in existing Grid environments. Additionally, we will create synthetic data based on distribution to conduct more detail analysis of the simulation. There will also be attempts to explain and justify the behavior of the simulation results.

6.4 Prototype Development for Autonomic Workflow Management System

This research will also implement a decentralized autonomic workflow management system prototype. The first step towards this is to develop the decentralized and cooperative workflow scheduling and resource discovery service. This can be done by leveraging the existing Grid frameworks such as Alchemi-Federation [24]. The Alchemi-Federation framework is developed with the aim of making distributed Grid resource integration and application programming efficient, flexible and scalable. It supports a P2P publish/subscribe based resource indexing service that makes the system scalable and a P2P coordination space for coordinating the distributed resource management in Grid.

The next step is to extend this decentralized workflow management architecture to adopt the self-management properties of autonomic computing system. We also aim to evaluate the performance of the resulting autonomic system with respect to different metrics available in the literature. The results accumulated from the prototype system will be compared against simulation results to verify the effectiveness of the system.

7. Conclusion and Future Work

In this paper, we present a research proposal for an autonomic workflow management system based on a decentralized and cooperative scheduling and resource discovery architecture. In order to leverage the scalable and self-organizing nature of the decentralize approach, we propose to coordinate the workflow brokers using a DHT-based decentralized coordination space. Further, these distributed workflow brokers are to be extended to comprise the properties of an autonomic system by adding the autonomic component into them.

Table 1: Timeline for the completion of future research activities

Period (Months)	Work Description
12 / 2007 – 03 / 2008	Evaluate performance of the proposed decentralized and cooperative workflow scheduling algorithm against the centralized and non-cooperative approaches through simulation. Incorporate possible optimization of the workflow by introducing intelligent task selection strategy in the coordination space.
04 / 2008 – 06 / 2008	Investigate the self-optimization and self- configuration property of an autonomic system and find out the requirements of the proposed workflow management system to exhibit such properties. Propose a methodology that can handle the dynamism in the Grid (changes in the resource and other supporting entities) and reschedule the tasks and reorganize the system accordingly.
07 / 2008 – 09 / 2008	Evaluate the performance of the proposed self-optimization and self- configuration mechanism through simulation.
10 / 2008 – 12 / 2008	Investigate the self-healing property of an autonomic system and find out the requirements of the proposed workflow management system to exhibit such property. Propose an effective approach to detect the failure of a task or resource and adapt to the changes accordingly.
01 / 2009 – 03 / 2009	Evaluate the performance of the proposed self healing mechanism through simulation.
04 / 2009 – 06 / 2009	Develop a prototype autonomic workflow management system by leveraging existing Grid technologies and deploy it in a real-world test bed to evaluate performance.
07 / 2009 – 09 / 2009	Write and submit thesis.

According to the preliminary results, our approach is scalable in terms of both scheduling message complexity and makespan and it can also avoid the limitation of single point of failure with respect to resource information service in centralized scheduling techniques. The time to be allocated for the future work regarding the proposed research is outlined in Table 1.

References

[1] P. Blaha, K. Schwarz, G.K.H. Madsen, D. Kvasnicka and J. Luitz, "WIEN2k, An Augmented Plane Wave + Local Orbitals Program for Calculating Crystal Properties", Vienna University of Technology, Austria, 2001, ISBN 3-9501031-1-2.

[2] J. Yu and R. Buyya, "Taxonomy of Workflow Management Systems for Grid Computing", Journal of Grid Computing, 3(3-4): 171-200, Springer, USA, September 2005.

[3] J. Yu and R. Buyya, "A novel architecture for realizing grid workflow using tuple spaces", In Proceedings of 5th IEEE/ACM Workshop on Grid Computing, IEEE CS Press, USA, 2004.

[4] T. Fahringer et al., "ASKALON: a tool set for cluster and Grid computing", Concurrency and Computation: Practice and Experience, 17:143-169, Wiley InterScience, 2005.

[5] S. Fitzgerald, I. Foster, C. Kesselman, G. von Laszewski, W. Smith, and S. Tuecke, "A directory service for configuring high-performance distributed computations", In Proceedings of 6th IEEE Symposium on High Performance Distributed Computing, IEEE CS Press, 1997.

[6] X. Zhang, J. L. Freschl, and J. M. Schopf, "A performance study of monitoring and information services for distributed systems", In Proceedings of 12th IEEE International Symposium on High Performance Distributed Computing, IEEE CS Press, June 2003.

[7] I. Foster and C. Kesselman, "The Grid: Blueprint for a New Computing Infrastructure", Morgan Kauffmann Publishers, Inc., 1999.

[8] M. Rahman, S. Venugopal, and R. Buyya, "A dynamic critical path algorithm for scheduling scientific workflow applications on global grids", In Proceedings of 3rd IEEE International Conference on e-Science and Grid Computing, India, December 2007.

[9] "An architectural blueprint for autonomic computing", White paper, IBM, June 2005.

[10] M. Parashar and S. Hariri, "Autonomic Computing: An Overview", UPP 2004, Mont Saint-Michel, France, Editors: J.-P. Banâtre et al. LNCS, Springer Verlag, Vol. 3566, pp. 247 – 259, 2005.

[11] E. Deelman, J. Blythe, Y. Gil, C. Kesselman, G. Mehta, S. Patil, M. H. Su, K. Vahi, M. Livny, "Pegasus: Mapping Scientific Workflow onto the Grid", Across Grids Conference, Cyprus, 2004.

[12] I. Taylor, M. Shields, and I. Wang, "Resource Management of Triana P2P Services", Grid Resource Management, Netherlands, June 2003.

[13] T. Oinn, M. Addis, J. Ferris, D. Marvin, M. Senger, M. Greenwood, T. Carver and K. Glover, M.R. Pocock, A. Wipat, and P. Li, "Taverna: a tool for the composition and enactment of bioinformatics workflows", Bioinformatics, 20(17):3045-3054, Oxford University Press, UK, 2004.

[14] M. Litzkow, M. Livny, and M. Mutka, "Condor-A Hunter of Idle Workstations", In Proceedings of 8th International Conference of Distributed Computing Systems, IEEE CS Press, USA, June 1988.

[15] B. Ludäscher, I. Altintas, C. Berkley, D. Higgins, E. Jaeger, M. Jones, E. A. Lee, J. Tao, and Y. Zhao, "Scientific Workflow Management and the KEPLER System", Concurrency and Computation: Practice & Experience, Special Issue on Scientific Workflows, 2005.

[16] J. Kubiatowicz, "OceanStore: Global-Scale Persistent Storage", Stanford Seminar Series, Stanford University, Spring 2001.

[17] J. Menon, D. A. Pease, R. Rees, L. Duyanovich, and B. Hillsberg, "IBM Storage Tank–A Heterogeneous Scalable SAN file system", IBM Systems Journal, 42(2):250–267, 2003.

[18] IBM Research, The Oceano Project, IBM.

[19] V. Narasayya, "AutoAdmin: Towards Self-Tuning Databases", November 2002, Guest Lecture at Stanford University.

[20] C. Poellabauer, "Q-Fabric-System Support for Continuous Online Quality Management", 2002.

[21] A. Rowstron and P. Druschel, "Pastry: Scalable, decentralized object location, and routing for large-scale peer-to-peer systems", In Proceedings of IFIP/ACM International Conference on Distributed Systems Platforms, pages 329–359. SpringerLink, Heidelberg, Germany, 2001.

[22] E. Tanin, A. Harwood, and H. Samet, "Using a distributed quadtree index in peer-to-peer networks", VLDB Journal, Vol 16, Issue 2, pages 165–178, 2007.

[23] R. Ranjan, M. Rahman, and R. Buyya, "A Decentralized and Cooperative Workflow Scheduling Algorithm", In Proceedings of 8th IEEE International Symposium on Cluster Computing and the Grid , France, May 2008.

[24] R. Ranjan, A. Harwood, and R. Buyya, "A case for cooperative and incentive based coupling of distributed clusters", Future Generation Computer Systems, Elsevier Science, The Netherlands, June 2007.

[25] R. Buyya, and 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, 14(13-15): 1175-1220, Wiley Press, USA, 2002.

[26] P. Garca, C. Pairot, R. Mondjar, J. Pujol, H. Tejedor, and R. Rallo, "Planetsim: A new overlay network simulation framework", In Proceedings of Software Engineering and Middleware, Austria, pages 123–137, Springer, Germany, 2005.

[27] The Grid Workloads Archive.
http://gwa.ewi.tudelft.nl/