Ecosystem Model Based Grid Resource Optimization Management

Yunlan Wang, Tao Wang, Lei Tang, Dong Zhong

Center for High Performance Computing, Northwestern Polytechnical University, Xi'an 710072, China

Summary

In this paper, the grid computing system is seemed as an ecosystem. The object of the optimization resource management is to promote the balance and evolution of the computing ecosystem. The architecture of the ecosystem model based grid resource management system is presented, which has the self-aware and self-optimization mechanism. The knowledge discovery based self-aware mechanism has the ability to reveal the behavior patterns knowledge hide in the history grid information system. The discovered knowledge can be used to predict the resource requirement and to optimize the resource allocation. The antigen identification mechanism is studied which can identify the factors related to the computing ecosystem unbalance state. With the self-optimization mechanism of the computing ecosystem, the resource allocation problem can be abstracted as a multi-objects optimization problem. The computing expectation, ecosystem environment, and the application characteristic are considered to design policy based adaptive resource allocation and job scheduling algorithm.

Key words:

Grid Computing, Computing Ecosystem, Resource allocation, Job scheduling, Knowledge Discovery

1. Introduction

In the network-based transparent and virtualization computing environment, there are huge demand for integration and sharing the distributed and heterogeneous high performance computing resource. Grid system is the information infrastructure for sharing and collaborating computing resource. Resource management is a key task of grid system management. However, since grid system has the intrinsic characteristics such as heterogeneous, distributed, dynamic, and evolutive; and the applications, computing resource and management policy of the grid system are in the self-organized mode; so the gird system appears complicated behaviors and uncertain performance. It determines that the management of grid system is very complicated and challengeable.

An ecosystem [1] is a self-sustaining system whose members benefit from each other's participation via energy exchange, substance metabolism, and information transfer. Ecosystem balance is a state of highly adaptive, harmonious and uniform. Every partition of the ecosystem closely relates, depends on each other, and co-evolves.

Grid computing system [2] can be seemed as an ecosystem, which is composed of hardware and software resources, management policies, diversity applications, and grid users. The balance and harmony of the computing ecosystem means that through the recycle of computing resources be occupied and released, the effect of management policy and intrinsic grid rules, the fluid of computing expenses, the transfer of data, information, and knowledge, to achieve the favorable system performance, to meet the grid applications' QoS requirement, and to realize the integration and sharing of grid resources. The competition in the computing ecosystem is the competing for resource between different grid users, and the competing for applications and users in different resources; the evolving of computing ecosystem means that the self-optimization of resource allocation policy.

In this paper, the view point of computing ecosystem is used to study the resource scheduling, the self-aware mechanism and the self- optimization mechanism. Specially, due to the computing flow dependence, data dependence and the dynamic, heterogeneous and autonomy grid environment, the grid jobs scheduling is very challengeable. The optimization grid resource management should concerns the resource utilization, the performance of application, the computing economy, etc. The study in this paper can provide support for resource sharing and cooperate computing; it can also provide the possibility of self-optimized and self-evolving mechanism to grid resource management system.

There are several aspects in the study of computing ecosystem resource management.

1) It should have the self-optimization ability to maintain the balance and harmony state of computing ecosystem. It means that the resource sharing and collaboration is realized, so the users can share the resource on demand. The core of the resource management is resource allocation and job scheduling. Since resource allocation should meet the dependence of

flow and data, meet the application's QoS requirement, minimize the expenditure of the users, and meet the requirement of load balance, etc, these make the scheduling problem very complicated.

2) With the increasing of the grid system scale, the grid is more and more complicated; so it needs intelligent management techniques. The computing system should have the ability of self-aware, self-optimization and self-evolution. The intrinsic rules should be achieved to support the intelligent management. The knowledge that can reveal the intrinsic rules about the complexity computing system should be extracted and used to improve the intelligent of the computing system. There are plenty of knowledge hide in the grid history information system; knowledge discovery is the non-trivial procedure of acquire the knowledge that is novel, effective, and potential useful. The knowledge discovered from the history database is significant for the intelligent, self-optimization and self-evolution resource management.

This paper is organized as follows: the state of the art about grid resource management is reported in part 2; in part 3, we present the architecture of the computing ecosystem model based resource management system. The implementation method is proposed in part 4; and part 5 is the conclusion.

2. State of the Art

There are many researchers focus in grid resource management; and there are plenty of achievements. However, they are many aspects are not concerned or have not been resolved.

According to the scheduling level, there are single-point decision centralized scheduling, level-wise scheduling and P2P scheduling. Condor [3] and Nimrod [4] are the examples of centralized scheduling; the centralized algorithm incurs to single-point fault and system bottleneck. Once the scheduler performance decreases, it may incur the system inefficient; and with the increasing of the number of computing resources, it is more and more difficult for the scheduling decision, thus it is baffle to the availability and scalability of the grid system. There are global scheduling and local scheduling in the level-wise scheduling. This mode is suitable for the resource allocation of large scale resource environment. The computing grid system such as Globus [5] and Legion [6] adopts this mode. In addition, reference [7] presents a job scheduling algorithm based on P2P mode.

According to the scheduling objects, it can be divided as user oriented and application oriented scheduling. Reference [8] presents a multi-objects scheduling algorithm driven by user preference; it can support multi-dimension QoS requirement. AppLeS[9] is devoted to improve the application performance through data collaboration. Reference [10] proposes a heuristic benefit function for resource allocation in the virtual computing environment. Condor-G [11] aims at high system throughputs. As for the optimized algorithm, there are algorithms such as MCT (Minimum Completion Time) [12], Min-min [13], Max-min [14], Sufferage, genetic algorithm [15], simulated annealing [16], Tabu algorithm [17], etc.

However, the foresaid algorithms only consider the hardware requirement and rarely concern the software requirement. They do not consider the users' expectation for resource price, security policy, etc. Reference [18][19] propose the resource allocation techniques based on computing economy model. However, it only considers the resource price and performance. It cannot meet the overall expectations of the resource consumers and owners. For example, considering the security and the relationship, the owner may be wish provide the resource to some users with high priority and forbid some users to use the resource.

At the same time, the flow dependence and data dependence makes the scheduling of computing flow challengeable. Above mentioned scheduling algorithm is not concern the dependence of tasks, thus it cannot be used to computing workflow scheduling. Reference [20] adopts genetic algorithm to minimize the workflow execution time, it is an application oriented algorithm. Reference [21] adopts economy model for grid workflow scheduling. Reference [22][23] select resource based on trust value. In the same way, all the above algorithms do not consider fine granularity access control policy and the computing expectation of the resource users and owners. Furthermore, they do not consider how to optimize the allocation and dynamic scheduling policy.

There are intrinsic rules about the computing ecosystem behind the stochastic. These rules can be used to predict the state of the computing ecosystem. Reference [24] adopts the statistical rules and the application similarity to predict the jobs runtime. Reference [25] executes part of the jobs to predict jobs runtime. NWS [26][27] predicts the future network and computing resource status. These are the initial study of the system behavior and prediction of the system performance.

The complexity of the computing ecosystem determines it is a non-trivial procedure to reveal the behavior knowledge of the computing ecosystem. There are some researchers adopts the KDD techniques to discovering knowledge in network alarm [28][29]. But there is rare study for complexity grid system. Reference [30] proposes

to use the fuzzy association rules to optimize the grid jobs scheduling. However, compare to the complexity of grid system and the plenty of the knowledge can be discovered, the above mentioned study is only very initial.

From the state of the art, it can be concluded as followings: there is a good start for grid resource management. However, from the point view of computing ecosystem, it still needs to optimize the grid resource management, study of behavior mode and utilize the intrinsic knowledge. Therefore, the object of this paper is the self-study, self-optimize and self-evolve of computing ecosystem. This study is significance for improve the sharing and cooperating of grid computing. It also can reveal the intrinsic rule of grid computing ecosystem.

3. The System Architecture

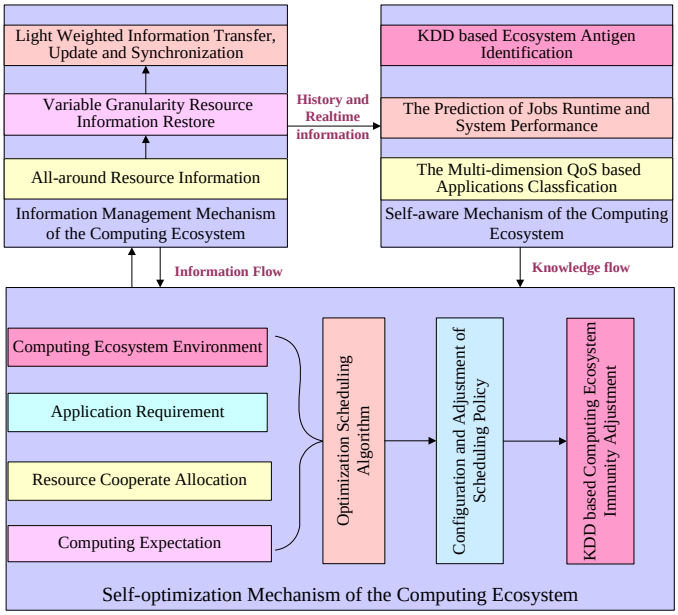

Fig.1 The architecture of computing ecosystem model based resource management system

This paper study the mechanism of self-aware and self-optimization of computing ecosystem. The architecture of computing ecosystem model based resource management system is shown in Fig.1. The KDD techniques are adopted to discover the behavior patterns knowledge hide in the history information system. The discovered knowledge can be used in the performance predicting and resource management optimizing. This is the self-aware mechanism of the computing ecosystem; based on the self-aware mechanism, the self-optimization mechanism can be designed and implemented, which can balance and optimize the computing ecosystem. The self-optimization mechanism can meet the computing expectation and application requirement, which is policy-based and have the ability of self-control, dynamic policy configuration and immunity adjustment.

3.1 The KDD based Self-aware Mechanism of Computing Ecosystem

1) Application Classification Algorithm based on Multi-dimension QoS

Because there are a lot of application types, the diversity application has various requirements for hardware resource, software resource, network resource, etc. Even if the same type application, resource requirement is varied with the different QoS requirement. Coarse granularity resource allocation and scheduling are impossible to achieve good performance; so the application oriented resource allocation algorithm must be studied. This means that we must study the application classification algorithm. Classification [31] is one of the main filed of knowledge discovery.

2) KDD based Prediction of Job Runtime and Computing Ecosystem Performance

Since the dynamic and evolving characteristic of computing ecosystem, the variety of the resource can affect the application performance, so the predicting of system performance and resource status and application status is very important and changeable. We must study the algorithm for predicting the job runtime under various computing ecosystem environment, various resource condition, and various workload; furthermore, study and present the method to predict the total performance of the computing ecosystem.

3) KDD based Antigen Identification Mechanism

Computing ecosystem, with the ability of self-aware and self-optimization, should self-study the knowledge that under what situation the computing ecosystem has good performance and under what situation the performance is inferior and cannot meet the application and user's QoS requirement. At the same time, when the factors that incur the unbalance of computing system appear, the system should recognize it; in other words, it should have the antigen identification ability.

3.2 The Self-optimization Mechanism of the Computing Ecosystem

Resource allocation algorithm is the core of the resource management. It should have the following abilities: meet the expectation of the resource supplier and consumer, allocate the resource intelligently and automatically, and improve the performance of the computing system.

1) The Grid Workflow Scheduling Algorithm which Concern the Computing Expectation, Resource Co-

allocation, Application Characteristic, and Computing Ecosystem Environment

The resource allocation policy may be different for different users. For example, from the viewpoint of the owner, the resource may be free and has high priority for some special user; however, the resource may be not allowed for some users, the reason can be security, competition, etc; for the others users, the resource usage policy may be determined by the QoS requirement or service level agreement (SLA). The grid users also have their preference for the resource performance, resource price, the credit and the security of the resource supplier. In this paper, all of these factors are named as computing expectation.

Under some instance, it is possible that the resource in the single management domain cannot meet the application requirement; so the different management domains must cooperate to allocate the computing resource. At the same time, there are diversity application fields in the complexity computing ecosystem and the entities such as resource, workload, and access and control policy are under continuously dynamic changing and evolving; so the resource allocation and job scheduling polices must be selected according to the computing environment, and it should meet the dependence relations of grid workflow. Resource allocation also should aware of the effect of the new submission jobs to the computing system performance. And it should minimize the resource fragment so that the free resource can be allocated to the large jobs. Furthermore, the overhead of communication and synchronization should be considered.

2) The Configuration and Adjustment of Computing Workflow Scheduling Policy

Policy based resource management can reduce the difficult of resource management. It is the way to realize the QoS management of grid applications. The difficult of policy based management is that how to select the scheduling policy according to application requirement and the computing ecosystem environment and how to generate and configure new policy.

3) The Immunity Adjustment of Computing Ecosystem

Immunity means that the computing ecosystem has the stable balance point; this point is corresponding to the immunity memory. When the antigen that incurs ecosystem unbalance intrudes, it disturbs the ecosystem balance state; the computing ecosystem can reach to new balance state through the reactions among the entities of the computing ecosystem. This is the immunity adjustment mechanism.

4. The Implementation Methods

4.1 Implementation Methods for KDD based Self-aware Mechanism

1) KDD Techniques for Analysis the Performance Data of Computing Ecosystem

The performance data such as the resource utilization, the response time, and the satisfaction degree are related to the allocation and deploy policy of hardware resource, software resource and date resource. The knowledge hide in the performance information can be discovered through frequent episodes [32], association rules [33], and sequential patterns [34]. At the same time, the application attributes can be viewed as the set of condition attributes, and the performance attributes can be viewed as the set of decision attributes; so we can construct the fuzzy object information system [35] and according to the rough set theory of fuzzy information system, the fuzzy rules can be discovered.

2) The Multi-dimension QoS based Application Classification Algorithm

The application is clustered based on the service level agreement, the resource requirement, and the runtime characteristic. The resource requirement is come from user's demand, statistic information, and the KDD results of the performance data. Based on the achieved resource requirement, the decision tree algorithm [36], the information entropy algorithm [37], the fuzzy set, or the rough set theory [38] can be adopted to classify the application.

3) Predicting Algorithm for Job Runtime and Computing Ecosystem Performance

Due to the stochastic and the fuzzy property of the grid system, the grid application performance has stochastic and fuzzy property to some degree. The lattice valued information system theory can be used to mine the knowledge in the computing ecosystem. We can also adopt the association rules or the sequential patterns algorithm to discover the intrinsic knowledge about the grid system. The discovered knowledge is useful for predicting the jobs' runtime and the performance of the computing system.

4) KDD based Antigen Identification Mechanism of the Computing Ecosystem

We can recognize the intrinsic difference between balanced state and unbalance state of the computing ecosystem; the method can be association rules, sequential patterns, frequent episodes, or classification algorithm. Especially, the characteristic of the critical state should be discovered and related to the resource management and

scheduling policy. The intrinsic rules and relations can be discovered, so the antigen of the computing ecosystem can be identified.

4.2 Implementation Methods for self-optimization mechanism

1) Abstract the Resource Allocation Problem as Optimization Problem

The resource allocation problem, which consider the computing expectation, resource cooperate, application characteristic and ecosystem environment, can be abstracted as a multi-objects and multi-constraints optimization problem. The constraints which must be met are hard condition; the constraints which are expected to meet are soft restrictions. This optimization problem can be solved by the punishment function method. The solutions which can satisfy the hard constraints are the feasible solutions.

2) The Configuration and Adjustment Mechanism of Scheduling Policy

The thinking and concepts in IP service policy based network management [39] presented by IETF can be borrowed in the design of self-adjustment mechanism. Then we can study the application-oriented or system-oriented computing workflow adaptive scheduling; the scheduling algorithm can be solidified as the scheduling policy. In other words, the fine granularity scheduling policy is designed for every type of applications. Thus, the application requirement can be satisfied, the QoS can be improved, and the management overhead can be controlled in the acceptable range. If the configured policy cannot meet the QoS or SLA, it triggers the policy dynamic deployment module to adjust and configure new policy.

3) The Immunity Mechanism of the Computing Ecosystem

Based on the antigen identification mechanism the system can trigger the immunity mechanism and self-adjust, when the system is in the critical state. So the new dynamic balance state can be achieved by the means of stop some jobs, migrate some jobs, or adjust the priority of the jobs.

5. Conclusion

In this paper, the complicated grid computing system is seemed as an ecosystem and the architecture of the computing ecosystem model based resource management system is presented. This system has the ability of self-aware and self-optimization. Its self-ability mechanism is based on the knowledge discovery techniques. The discovered knowledge can be used to predict the resource requirement and to optimize the resource allocation. In the self-optimization mechanism, the resource allocation problem is abstracted as a multi-objects and multi-constraints optimization problem; and the computing expectation, resource cooperation, application characteristic, and ecosystem environment are all considered to design and implement policy based adaptive resource allocation and job scheduling algorithm and implement the immunity adjusting mechanism.

References

[1] Guy Battle, Christopher McCarthy, Sustainable ecosystems and the built environment, Chichester: Wiley-Academy, 2001.

[2] Fran Berman, Geoffrey Fox and Tony Hey, Grid Computing: Making the Global Infrastructure a Reality, a special issue of Concurrency and Computation: Practice and Experience, published March 2003 by Wiley.

[3] Michael Litzkow, Miron Livny, and Matt Mutka, Condor – A Hunter of Idle Workstations, Proceedings of the 8th International Conference of Distributed Computing Systems, June, 1988: 104-111.

[4] Rajkumar Buyya, David Abramson, and Jonathan Giddy, Nimrod/G: An rchitecture for a Resource Management and Scheduling System in a Global Computational Grid. In Proceedings of the 4th International Conference on High Performance Computing in Asia-Pacific Region, Beijing, China, 2000:1-6

[5] I. Foster. “Globus Toolkit Version 4: Software for Service-Oriented Systems”. IFIP International Conference on Network and Parallel Computing, Springer-Verlag LNCS 3779, 2006: 2-13.

[6] S. Chapin, J. Karpovich, A. Grimshaw, The Legion Resource Management System, Proceedings of the 5th Workshop on Job Scheduling Strategies for Parallel Processing, April 1999.

[7] SungJin Choi, MaengSoon Baik, etc, Dynamic Scheduling Mechanism for Result Certification in Peer to Peer Grid Computing, Grid and Cooperative Computing - GCC 2005: 4th International Conference, 2005.

[8] Krzysztof Kurowski, Jarek Nabrzyski, Juliusz Pukacki. User Preference Driven Multi-objective Resource Management in Grid Environments. Proceedings of the 1st International Symposium on Cluster Computing and the Grid, 2001.

[9] Henri Casanova, Graziano Obertelli, Francine Berman and Rich wolski, The AppLeS Parameter Sweep Template: User-Level Middleware for the Grid, Proceedings of the Super Computing Conference (SC'2000).

[10] Qing Ding, Guoliang Chen, A Benefit Function Mapping Heuristic for a Class of Meta-Tasks in Grid Environments, 1st International Symposium on Cluster Computing and the Grid. 2001: 654.

[11] Frey J, Tannenbaum T, Livny M, Foster I, Tuecke S. Condor-G: A computation management Agent for multi-institutional grids. In: Proc. of the 10th IEEE Symp. on High Performance Distributed Computing (HPDC10). San Francisco: IEEE Computer Society Press, 2001: 7-9.

[12] R. F. Freund, M. Gherrity, S. Ambrosius, M. Camp-bell, M. Halderman, D. Hensgen, E. Keith, T. Kidd, M. Kussow, J. D. Lima, F. Mirabile, L. Moore, B. Rust, and H. J. Siegel, Scheduling resources in multi-user, heterogeneous, computing environments with SmartNet, 7th IEEE Heterogeneous Computing Workshop (HCW '98), Mar. 1998: 184–199.

[13] X He, X Sun,G V Laszewski. QoS guided min-min heuristic for grid task scheduling. Journal of Computer Science and Technology, 2003, 18(4):442-451.

[14] Maheswaran M, Ali S, Siegel H J, et al, Dynamic mapping of a class of independent tasks onto heterogeneous computing systems. In the 8th IEEE Heterogeneous Computing Workshop (HCW '99), San Juan, Puerto Rico, Apr. 1999: 30-44.

[15] M. Aggarwal, R.D. Kent and A. Ngom, Genetic Algorithm Based Scheduler for Computational Grids, in Proc. of the 19th Annual International Symposium on High Performance Computing Systems and Applications (HPCS'05), Guelph, Ontario Canada, May 2005: 209-215,.

[16] Fidanova, S. Simulated Annealing for Grid Scheduling Problem, IEEE John Vincent Atanasoff 2006 International Symposium on Modern Computing, Oct. 2006: 41-45.

[17] M Gendreau, G Laporte, F Semet. A dynamic model and parallel tabu search heuristic for real-time ambulance relocation. Parallel Computing, 2001, 27(2): 1641-1653.

[18] Rajkumar Buyya, David Abramson, Jonathan Giddy, and Heinz Stockinger, Economic Models for Resource Management and Scheduling in Grid Computing. The Journal of Concurrency and Computation: Practice and Experience 14, 13-15 (2002): 1507-1542

[19] Subramoniam K, Maheswaran M, Toulouse M. Towards a micro-economic model for resource allocation in grid computing systems. In:Proc.of the 2002 IEEE Canadian Conf.on Electrical and Computer Engineering.Manitoba: IEEE Press, 2002: 782-785.

[20] R. Prodan and T. Fahringer. Dynamic Scheduling of Scientific Workflow Applications on the Grid: A Case Study. In 20th Annual ACM Symposium on Applied Computing (SAC 2005), New Mexico USA, ACM Press, New York, NY, USA, March 2005.

[21] A. Geppert, M. Kradolfer, and D. Tombros. Market-based Workflow Management. International Journal of Cooperative Information Systems, World Scientific Publishing Co., NJ, USA, 1998.

[22] Song and K. Hwang. Security Binding for Trusted Job Outsourcing in Open Computational Grids. IEEE Transactions on Parallel and Distributed Systems (TPDS), submitted May 2004, revised Dec. 2004.

[23] S. Y. Zhao and V. Lo. Result Verification and Trust-based Scheduling in Open Peer-to-Peer Cycle Sharing Systems. Technical Report, University of Oregon, USA, 2005.

[24] Warren Smith, Ian Foster, Valerie Taylor, Predicting Application Run Times Using Historical Information. Proceedings of the Workshop on Job Scheduling Strategies for Parallel Processing, March 30, 1998: 122-142.

[25] Tao Yang, Xiaosong Ma, Frank Mueller, Predicting Parallel Applications' Performance Across Platforms Using Partial Execution, in Proceedings of ACM/IEEE Conference on Supercomputing, Nov 2005.

[26] Experiences with Predicting Resource Performance On-line in Computational Grid Settings (pdf), Rich Wolski, ACM SIGMETRICS Performance Evaluation Review, Volume 30, Number 4, March, 2003: 41-49.

[27] The Network Weather Service: A Distributed Resource Performance Forecasting Service for Metacomputing (compressed postscript) Rich Wolski, Neil Spring, and Jim Hayes, Journal of Future Generation Computing Systems,Volume 15, Numbers 5-6, October, 1999: 757-768.

[28] M. Klemettinen, H. Mannila and H. Toivonen, Interactive exploration of interesting findings in the Telecommunication Network Alarm Sequence Analyzer TASA. Information and Software Technology. 1999, 41: 557 - 567.

[29] Sasisekharan and Raguram. Data mining and forecasting in large-scale telecommunication networks. IEEE Expert. 1996, 11 (1): 37-43.

[30] Jin Huang, Hai Jin, Xia Xie, Qin Zhang, An Approach to Grid Optimization Based on Fuzzy Association Rule Mining. Proceedings of the First International Conference on e-Science and Grid Computing, 2005.

[31] Jiawei Han and Micheline Kamber, Data Mining: Concepts And Techniques, San Francisco: Morgan Kaufmann Publishers, 2001.

[32] H. Mannila, H. Toivonen and A. I. Verkamo, Discovery of frequent episodes in event sequences, Data Mining and Knowledge Discovery. 1997, 1(3): 259- 289.

[33] Cheung D W, Lee S D, Xiao V. Effect of Data Skewness and Workload Balance in Parallel Data Mining. IEEE Transactions on Knowledge and Data Engineering, May 2002, 14(3): 498-514.

[34] Pauray S.M. Tsai, Chien-Ming Chen, discovering knowledge from large databases using prestored information, information systems, Vol.26, No.1, 2001:1-14.

[35] Ziarko, W.: Variable Precision Rough Sets Model. Journal of Computer and System Sciences, 46, (1993) 39-59.

[36] Quinlan J R. Induction of decision trees. Machine Learning. 1986, 1: 81- 106.

[37] R. M. Gray, Entropy and Information Theory, Springer-Verlag, 1990.

[38] Pawlak Z., Rough sets, decision algorithms and Bayes' theorem. European Journal of Operational Research, 136(2002): 181-189.

[39] Dinesh C Verma, Simplifying Network Administration using Policy based Management, IEEE Network Interactive, 2002.

Yunlan Wang received her M.S. degree in mathmetics and Ph.D. degree in computer science from Xi'an Jiaotong university, China, in 2001 and 2004 respectively. She is curretly a researcher of Center for High Performance Computing, Northwestern Polytechnical University, China. Her research interests include grid computing, knowledge discovery.