

A Survey on Issues in Mobile Grid Computing

Abhishek Bichhawat and R. C. Joshi
 Department of Electronics and Computer Engineering,
 Indian Institute of Technology Roorkee, India
 Email: {abhibpec, rcjosfec}@iitr.ernet.in

Abstract— Mobile grid computing broadens the scope of grid computing by including a vast resource pool available in the form of mobile devices. As the number of mobile devices in use is bound to increase in coming years, and with enhanced features, mobile grid computing will have a lot to offer in the area of high performance computing. This paper surveys the paradigm of mobile grid computing and the features it offers in the current computing environment. We discuss the role of mobile devices in the existing grid technology. We also discuss the various challenges the paradigm faces as a result of inclusion of the feature of mobility in the grid computing environment.

Index Terms— Mobile grid, issues, survey

I. INTRODUCTION

Grid computing can be defined as a set of parallel and distributed systems that enable the sharing of geographically distributed, independent and heterogeneous resources like computers, software, etc., for effectively solving computationally intensive problems and to enable cooperative working. As put by Ian Foster in [1], grid computing is concerned with coordinated resource sharing and problem solving in dynamic, multi-institutional virtual organizations. In [2], grid is defined as a system that coordinates resources that are not subject to centralized control using standard, open, general-purpose protocols and interfaces to deliver nontrivial qualities of service. Since its inception, grid has been promising better and efficient utilization of resources along with delivering quality of service.

Mobile computing is another computing paradigm of distributed systems, considering mobility, portability and wireless communications [3]. It has been an area of continuous development and has defined a new approach to communication. It includes the mobility of both physical and logical computing devices. The number of mobile devices in use is enormous and newer and better featured devices continue to be modeled. They promote mobile communication and ubiquity of computers. However, they bring along problems of intermittent network, failures and faults, and limited resources.

While today's grid computing environment consists of resources that are wired and static, the future envisages the need for a mobile grid scenario as the mobility of users is increasing day by day and so is their access to the resources. The extension of the grid to mobile computing by making it available to the users, even when they are mobile forms the basis of mobile grid. Mobile grid is a full inheritor of grid with the additional feature of supporting mobile users and resources in a seamless, transparent, secure and efficient way [5]. This extension of the grid dawn the new era of pervasive high performance

computing. As the computer gets ubiquitous, it would require the extension of the various facilities available to it today. Mobile grid computing provides a new technology for solving complex and compute intensive problems in mobile or nomadic environments.

The main factors that hinder the growth of this paradigm as compared to the grid computing paradigm are the issues related to mobility and the constraints of mobile computing [3]: mobile devices are poor in available resources as compared to wired systems; mobile devices are more prone to security breaches; mobile connectivity is highly variable in performance and reliability; mobile devices rely on a finite energy source.

In order to enable the concept of mobile grid take over the existing grid, it is necessary that the effect of these constraints be minimized. Mobility needs to be integrated with the features of the existing grid technology. Section II presents some related works. Section III discusses the paradigm of mobile grid computing briefly. The various issues and their impacts are studied in section IV. The paper concludes with scope for future work discussed.

II. RELATED WORKS

Various authors have surveyed and reviewed the work done in the field of wireless grid computing including mobile grid computing [30-32]. But, none of them discuss the issues faced by mobile grid computing paradigm. In this paper, we provide a brief review of the various issues faced by mobile grid computing.

III. OVERVIEW OF MOBILE GRID

A. Definition of Mobile Grid

Mobile grid is a class of distributed systems, both wired and mobile, that autonomously engage in sharing of resources and exchange of services to provide higher performance and better utilization of the otherwise unutilized resources on the mobile devices. As the resources on the host are limited with power and memory constraints, it seems unnecessary for extending certain features of grid to the mobile host. But in future, when the systems become all-pervading, the constraints of power and memory would seem trivial when compared to the advantages offered by mobile grid. Besides, with continuous development in device technology, the constraints of power and memory are diminishing and mobile devices are having better specifications when compared to their predecessors. Mobile devices can make use of the available resources in the grid to perform any task or use them when on the move.

B. Existing Mobile Grid Architectures

The architectures proposed have generally added up certain features to the existing grids so as to allow it to handle mobile devices. Many mobile grid infrastructures have been modeled in order to support mobile devices in the grid. One of the works in this direction was initiated by the European Union with the Akogrimo project. The base of the model was that of a service grid, as the mobile grid is based on its principles [6]. Bruneo et al. presented a mobile agent based middleware for the mobile grid paradigm [7]. Mobile OGSINET, an extension of the grid implemented in OGSINET to the mobile devices [8]. In [4], clusters of certain mobile devices are created. Each cluster is centered on a proxy which can be either another mobile device, a non-mobile node within the Grid, or a dedicated middleware server. A variation to the proxy architecture is the Signal architecture that integrates mobile devices with existing grid platforms to allow operations using proxy-based system [9]. Architecture in [10], describes modules such as discovery service to identify and locate a middleware by a mobile device, communication interface to communicate between the client and the middleware etc. An energy efficient middleware for supporting multimedia services in mobile grid environments was suggested in [11]. A similar architecture was proposed in [12] which consist of a Broker Module that manages the resources, schedules jobs and monitors their execution, storing the data and the job results in task databases.

C. Need of Mobile Grid

Grid computing paradigm has become one of the most important techniques in the area of high performance computing. As the number of prospective users increase rapidly, the available resources can be supplemented with resources available from the mobile devices. As the mobile devices in use are huge in number, even a small percentage of their resource utilization in a grid environment shall be highly advantageous and useful [4]. Although significant work is being done in the field of increasing the processing and power capabilities of mobile devices, its value is lesser than that of a wired device.

The main difference between the grid and the mobile grid is mobility. Mobility of devices is not supported in most of the grids that are in operation today. This problem might seem trivial, but with millions of mobile devices running today and their processing power remaining unutilized to the maximum, it seems a good idea to include these in the existing family of grids. Apart from utilizing resources on the mobile devices, the mobile grid can provide mobile devices with an opportunity to use the resources on the grid, thereby saving their own resources considerably and overcoming the physical shortcoming of the device. Mobile devices having access to the grid as users would thus be able to perform certain tasks on the run which otherwise would have been done only when the user could access a wired device. Certain situations could allow the mobile devices to contribute to the grid resource warehouse. Though the idea of using resources of the grid on a mobile device seems unwanted and vague, certain situations may support it [12]. And these situations are sure

to increase in the near future as computers get ubiquitous.

IV. ISSUES IN MOBILE GRID

A. Overview

As a combination of both mobile and grid computing, mobile grid raises a combined set of issues involving factors from both the computing areas. Mobility of devices raises much more issues or aggravates the existing issues. Higher mobility causes the execution time of a task to increase whereas higher availability of resources decreases execution time [13]. The mobility of a device also increases the monitoring overhead. Individually, work has been done in the areas described below in either grid or mobile for providing better features in the two areas of computing. After integration, certain parameters need to be concerned before applying those methodologies. The issues with mobile grid include a combination of those from grid computing like scheduling, security, resource management, fault-tolerance etc. and those from mobile computing like security, network connectivity, variable bandwidth, quality of service, energy-aware computing etc. as mentioned earlier.

The issues which concern the development of the paradigm are those related to scheduling, fault-tolerance, resource management, security, energy-awareness etc. The issues of network connectivity and bandwidth relate to fault-tolerance issue. Energy consumption can be optimized by having a trade-off between energy consumed and the quality of service provided in the computing environment [14]. The power available on the device is generally affected by operations performed by it. The solutions to the issues of scheduling, fault-tolerance etc. must consider the issue of energy because it is an important aspect of mobile devices. A successful integration of mobile devices into the grid would require that the functionalities of the grid are energy-aware for mobile devices.

B. Quality of Service

Quality of Service (QoS) in terms of mobile grid may be defined as the quality prerequisite of the type of service offered by the grid to the users. It has various parameters which govern the type of service being offered and the quality of results presented to the user. In context to grid computing, the various parameters defining QoS include the response time, throughput, availability, and security, and co-allocation of multiple resource types to meet complex user demands [2]. In context to mobile computing many QoS parameters have been defined in [15], including network connectivity, available bandwidth, disconnected operation, energy aware computing and secure computing.

In a mobile grid, these QoS parameters need to be taken care of to provide a good and reliable service. For instance, the communication between the user and service provider could break while executing a job that requires data communication, requiring the job to wait until the connection is re-established. This results in degradation of performance and results in failure to extract the full advantages of the grid. A certain QoS is guaranteed by

most implementations to ensure maximum advantage to the user. In case of a mobile grid, the QoS is generally compromised to hoard power available on the device. But on the whole, a minimum QoS is always guaranteed.

C. Scheduling and Resource Management

Users make use of resources by submitting tasks to the grid. Using some scheduling algorithms, the grid allocates those tasks to appropriate resources. Effective scheduling of jobs in a grid is complicated by a number of factors. The challenges in a mobile grid environment for job scheduling are more as compared to the grid due to the unreliable communication [20]. The scheduling is also affected by mobility and availability of resources.

The large number and the heterogeneous potential of the grid computing resources cause great challenges in resource management. Resource management includes resource discovery, provisioning and monitoring. The most important aspect of the resource management area is the selection of the apt resource from the pool and provisioning it to facilitate the user's needs. In addition to these, the resource manager must also provide efficient monitoring of resources to collect utilization, availability and other resource-related information.

The simplest method of scheduling is to schedule the task to the available mobile node as presented in [16]. In this method, the node returned the output to the user at the completion of the task. If the node was out of the network or disconnected, the user waited until the node reconnected and returned the output. The node executed the task irrespective of connection. This method was limited to executing the task in a state of disconnection and did not deal with other factors such as limited power. It reduced the overhead incurred to the scheduler for determining the scheduling plan for allocating a job to a resource. The transfer had to wait until reconnection and hence might be delayed for a very long period.

An important aspect of scheduling is to allocate the job optimally minimizing the total cost of execution for the user. A method proposed in [18] used game theoretic pricing model for job allocation and optimizing the total system cost. The model was enhanced to handle mobility of nodes in [19]. It took into account the priority of jobs and offered pre-emptive scheduling.

Alongside mobility, the energy available on a mobile device is another crucial factor affecting scheduling. The method suggested in [22] considered various factors such as mobility, battery power to assign the resources to the jobs. Based on the dynamic characteristics of the mobile devices, mobile devices were ranked and grouped. The method dealt with the issue of fault tolerance by allocating the job concurrently to various nodes in a group. In case a higher ranking node failed to deliver the result, a lower ranking node provided the output, thus overcoming the issue of faults and failures. The method in [23] made use of Weibull distribution to estimate the failure rate of a node. Based on the estimated rate, it replicated and allocated the task to various nodes in the network. These resulted in redundancy of the result and unwanted utilization of resources in certain cases.

Due to the availability of redundant outputs the efficiency of the grid reduces and the main purpose of the grid is lost. Katsaros et al. presented an Installments scheduling policy [28]. The Installments policy involved partitioning the task into fragments and executing them on different nodes. Thus, in case of a node failure only the fragment executing at that node needed to be rescheduled and executed. This saved the task of performing the task again from beginning. It also reduced the wastage of resources, which, otherwise, happened in case of replication. The method proved to be inefficient when the tasks could not be fragmented.

Using the mobility pattern of the resources, an approximate period for which the resource is available can be estimated. The nodes in an area tend to follow a pattern of movement and based on that pattern, the availability of a node on the network can be determined. Such an algorithm was presented by Lee et al. [20]. The resource availability is measured and classified as: full availability, partial availability, unavailability by using the user's mobility pattern. The jobs are allocated based on the communication interval and the number of times communication is needed amongst the devices. This model was a heuristic approach of scheduling jobs learning from the past behavior of the user.

D. Security

Mobile grid attracts many attacks due to its wide and open nature. With mobility, large distance inter-domain interactions between the users increase and hence the security mechanisms must scale to a global level. The mobility also adds the issue of physical security. Devices are less secure physically when mobile. There is always a threat of damage, theft etc. The wireless nature of the network poses an additional threat. The wireless network is more prone to eavesdropping, tracing and tampering of data [27]. Thus, means for secure handoff, managing disconnection etc. are needed to secure the computing environment to prevent intruders and theft of resources. Addressing the vulnerabilities of both the mobile and the grid is necessary to combat attacks in a mobile grid. Security in a mobile grid environment requires secure authentication and key management services along with authorization, integrity and confidentiality [24]. Secure logon does not ensure the integrity and confidentiality of messages, which remain vulnerable to attacks, and hence, require to be addressed.

Security features can be borrowed from the actual grid and mobile computing scenarios. The security features implemented in grid and mobile computing environment can be extended to the mobile grid environment by merging those features with each other. In [25], Rosado et al. presented security use cases to extend the features of grid security in a mobile grid environment. An efficient security approach is to separate the implementations of grid and mobile part of mobile grid. The security features of the wireless environment take care of transfer and receipt of jobs and other details. The grid security features take care of authentication, single sign-on and auditing. Wong et al. in [26] presented a mobile grid services framework which offered security support using the security features

available in the grid middleware and the mobile agent platform, which was used for imparting mobility support to the grid.

E. Fault-tolerance

The problems faced by mobile devices result in unpredictable network behavior and unreliable communication. Due to this, mobile grids also suffer the problem of frequent communication failure, which has an adverse effect on the grid, as there might be a data transfer in progress or some message passing amongst processes which gets interrupted and result in degradation of performance. In case of a failure, the same job has to be re-executed to obtain the desired results. This results in wasteful utilization of the resources. In order to effectively utilize the resources, certain fault tolerant implementations are required.

Two common ways for ensuring fault tolerance include checkpointing and task replication. Checkpointing is the process of saving the current state of the process. In case of a failure, the task can be restarted from the last saved state. Task replication provides fault tolerant performance by executing the task on several devices by creating replicas of the task. When a fault occurs, results from the other device can be used. This requires proper determination of the number of replicas to be run on different devices because if the number is too large, it would result in worthless redundant information, unless all other devices fail, and if too small, there might be a possibility that the results are not obtained and the resources were wasted on the devices. The main issue with task replication is the fact of the redundant information. The resources on certain devices might have done a job which is not useful in actual computation if all the devices have produced the result without any faults. This results in improper utilization of resources.

A simple method for implementation of fault tolerance suggested in [16] is to allow disconnected operation. When a mobile node joined a network, a job is extracted from the queue and assigned to it. The mobile node performed the job and transferred the result back to the mobile queuing server. In case of a disconnection, the job was still executed at the node. The results were transferred back when the node reconnected to the server. The problem with this scheme was that the time between disconnection and reconnection varies largely and depends on the network parameters other than the device's mobility. Hence, it was undeterminable when the results shall be available to the server.

Another method to ensure fault tolerance is to provide mobile host proxies at the servers as in [17]. The mobile host proxy is the only interface to communicate between the mobile device and the base station. The base station coordinated amongst the various proxies to take checkpoints as initiated by the mobile host. The checkpoint was taken by coordinating amongst the various processes related to the one running on the mobile host. The context information of the mobile node such as the location information and the network status information could be used to optimize the process of checkpointing [21]. The context information was sent to the proxy and analyzed.

The proxy then informed the mobile device of the time to take a checkpoint using the analyzed information. The checkpointing depended on the calculation of available bandwidth. If the available bandwidth was above a certain threshold, local checkpointing was done; else checkpoint was taken on storage of the proxy. When a fault occurs, it discovered a resource that was functionally similar to the device with the fault, and enabled the continuation of the interrupted operation on the resource.

A decentralized approach to checkpointing was suggested by Darby and Tzeng [29]. A mobile host chose to take a checkpoint and saved it on its neighboring hosts rather than saving it on a base station. The choice of the host to save the checkpoint was based on a Reliability Driven methodology. The method assumed that all mobile hosts were aware of the quality of its own connection and that of its neighbors. The algorithm allowed a provider to pair with only one consumer. Hence, a weaker reliability link to a consumer was broken if another consumer requested and had a stronger reliability link. This resulted in an overhead on the mobile host as requests and breaks occur continuously. Message traffic also increased in the network which was undesirable. Another factor was the memory, which is limited in mobile devices where the checkpoints were stored. The method did not provide any information about the process of recovery in case of a fault.

In [22], an approach to address the problem of fault occurrence by concurrent execution of the required job at two or more devices was suggested. In case of a failure, the result from the other devices was used. Thus by redundant execution of the job at multiple nodes, the issue of fault-tolerance was dealt with. A similar model was proposed by Litke et al. in [23] which used Weibull distribution to determine the failure probability of a resource. The scheme replicated tasks based on the reliability function of the resources with which the job is associated. The replication techniques reduce the effective utilization of the resources because multiple resources execute the same job and in scenarios with fewer failures, redundant data is available. This is not desirable in a grid computing environment where we look to maximize the resource utilization for better computation. But in cases where fault tolerance is a priority, redundancy of data is of less consideration.

V. CONCLUSION

As advances continue in the field of mobile computing and handheld devices continue to be designed with better features, the mobile version of grid shall become the basis for ubiquitous and pervasive high performance computing in the forthcoming years. It shall provide more computational power to applications and services from the mobile resources which otherwise remain underutilized. It can be identified that mobility is the contributing factor to the problems faced by mobile grid apart from the issues of the normal grid. The issue of low power is another area of concern and must be taken care of along with other issues. QoS is an important consideration in certain real time scenarios which also needs to be considered in conjunction with power and other issues. Research is going on in the discussed areas to allow successful integration of the

mobile devices in the grid and to provide better utilization of the resources. With continuous work in afore-mentioned areas, mobile grid is sure to be an important computing paradigm in the near future.

REFERENCES

- [1] I. Foster, C. Kesselman and S. Tueke. "The anatomy of the grid: Enabling scalable virtual organizations," *Int. J. Supercomp. Appl.*, vol.15 [3], pp. 200-222, 2001.
- [2] I. Foster. "What is the grid? A three point checklist," Argonne National Laboratory, fp.mcs.anl.gov/~foster/Articles/WhatIsTheGrid.pdf, 2002.
- [3] M. Satyanarayanan, "Fundamental challenges in mobile computing," *Proc. of the 15th annual ACM Symposium on Principles of Distributed Computing*, Philadelphia, Pennsylvania, pp. 1-7, 1996.
- [4] T. Phan, L. Huang and C. Dulan, "Challenge: Integrating mobile wireless devices into the computational grid," *Proc. of the 8th ACM International Conference on Mobile Computing and Networking (MobiCom '02)*, Atlanta, GA, pp. 271-278, September 25-27, 2002.
- [5] A. Litke, D. Skoutas, and T. Varvarigou, "Mobile grid computing: Changes and challenges of resource management in a mobile grid environment," *5th Int. Conf. on Practical Aspects of Knowledge Management*, 2004.
- [6] M. Waldburger and B. Stiller, "Toward the mobile grid: Service provisioning in a mobile dynamic virtual organization," *4th ACS/IEEE International Conference on Computer Systems and Applications (AICCSA-06)*, pp. 579-583, Dubai/Sharjah, UAE, March 2006.
- [7] D. Bruneo, M. Scarpa, A. Zaia, and A. Puliafito. "Communication paradigms for mobile grid users," *Proc. of the 3rd IEEE/ACM International Symposium on Cluster Computing and the Grid*, pp.669-676, 2003.
- [8] D. C. Chu and M. Humphrey, "Mobile OGSINET: Grid computing on mobile devices," *Proc. of the 5th IEEE/ACM International Workshop on Grid Computing*, pp. 182-191, November 8, 2004.
- [9] J. Hwang and P. Aravamudham, "Middleware services for P2P computing in wireless grid networks," *IEEE Internet Comput.*, vol. 8 [4], pp. 40-46, July 2004.
- [10] U. Kalim, H. Jameel, A. Sajjad and S. Lee, "Mobile-to-grid middleware: An approach for breaching the divide between mobile and grid environments," *ICN2005, LNCS 3420*, pp. 1-8, 2005.
- [11] Y. Huang, S. Mohapatra and N. Venkatasubramian, "An energy-efficient middleware for supporting multimedia services in mobile grid environments", *Proc. of the International Conference on Information Technology: Coding and Computing - Volume II*, pp. 220-225, 2005.
- [12] F. Navarro, A. Schulter, F. Koch, M. Assunção, C. B. Westphall, "Towards a middleware for mobile grids," *10th IEEE/IFIP Network Operations and Management Symposium (NOMS 2006)*, Vancouver, Apr. 2006.
- [13] A. S. Nandeppanavar, M. N. Birje, S. S. Manvi and Shridhar. "Mobility impact on performance of mobile grids," *International Journal of Computer Science and Information Security (IJCSIS)*, vol. 7 [1], 2010.
- [14] C. Li and L. Li, "Tradeoffs between energy consumption and QoS in mobile grid," *J. Supercomput.*, Springer, 2009.
- [15] Suresh Singh, "Qualities of service guarantees in mobile computing," *Journal of Computer Communications*, vol. 19 [4], pp. 359-371, April 1996.
- [16] S. Park, Y. Ko and J. Kim, "Disconnected operation service in mobile grid computing," *International Conference on Service Oriented Computing (ICSOC'2003)*, Trento, Italy, pp. 499-513, Dec 2003.
- [17] I. Rao and T. Chung, "A proxy based efficient checkpointing scheme for fault recovery in mobile grid system," *High Performance Computing - HiPC 2006*, pp. 448-459, 2006.
- [18] P. Ghosh, N. Roy, K. Basu and S.K. Das, "A game theory based pricing strategy for job allocation in mobile grids," *Proc. of the 18th IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, New Mexico, pp. 82a, 2004.
- [19] P. Ghosh, N. Roy, K. Basu and S.K. Das, "Mobility-aware efficient job scheduling in mobile grids," *Proc. of the 7th IEEE International Symposium on Cluster Computing and the Grid*, pp.701-706, 2007.
- [20] J. Lee, S. Song, J. Gil, K. Chung, T. Suh and H. Yu, "Balanced scheduling algorithm considering availability in mobile grid," *Proc. of the 4th Intl. Conf. on Advances in Grid and Pervasive Computing*, pp.211-222, 2009.
- [21] J. Park, H. Yu, "Context information based fault tolerant technique in mobile grid," *Whitepapers*, Mar. 5, 2008.
- [22] S. Choi, I. Cho, K. Chung, B. Song, H. Yu, "Group-based resource selection algorithm supporting fault-tolerance in Mobile Grid," *Proc. of the 3rd International Conference on Semantics, Knowledge and Grid*, pp.426-429, 2007.
- [23] A. Litke, D. Skoutas, K. Tserpes and T. Varvarigou, "Efficient task replication and management for adaptive fault tolerance in mobile grid environments," *Future Gener. Comp. Sy.*, Elsevier Science Publishers, vol. 23 [2], pp.163-178, 2007.
- [24] K. Moon, N. Park, J. Jang, S. Sohn and J. Ryou, "An integrated XML security mechanism for mobile grid application," *Computational Science and Its Applications - ICCSA 2004*, pp.776-783, 2004.
- [25] D. G. Rosado, E. F. Medina and J. López, "Reusable security use cases for mobile grid environments," *Proc. of the 2009 ICSE Workshop on Software Engineering for Secure Systems*, pp. 1-8, 2009.
- [26] Sze-Wing Wong and Kam-Wing Ng, "Security support for mobile grid services framework," *Proc. of International Conference on Next Generation Web Services Practices (NWeSP'06)*, pp. 75-82, 2006.
- [27] A. Talukder and R. Yavagal, "*Mobile Computing*," McGraw Hill Pub., Oct. 26, 2006.
- [28] K. Katsaros and G.C. Polyzos, "Evaluation of scheduling policies in a mobile grid architecture," *Proc. International Symposium on Performance Evaluation of Computer and Telecommunication Systems*, Edinburgh, UK, June 2008.
- [29] P. J. Darby III and N. Tzeng, "Decentralized QoS-aware checkpointing arrangement in mobile grid computing," *IEEE Trans. Mobile Comput.*, vol. 9 [8], pp. 1173-1186, Aug. 2010.
- [30] S. P. Ahuja and J. R. Myers, "A survey on wireless grid computing," *J. Supercomput.*, Springer, 37, pp. 3-21, 2006.
- [31] S. S. Manvi and M. N. Birje, "A review on wireless grid computing," *International Journal of Computer and Electrical Engineering*, vol. 2[3], pp. 469-474, Jun. 2010.
- [32] G. Li, H. Sun, H. Gao, H. Yu and Y. Cai, "A survey on wireless grids and clouds," *Proc. 8th Intl. Conf. on Grid and Cooperative Computing*, pp. 261-267, 2009.