Providing Service Grid Capabilities on Mobile Phones

Meisam HEJAZINIA, Mohammad Reza RAZZAZI
Amirkabir University of Technology,
Dept. of Computer and IT Engineering, Tehran, Iran
hejazinia@aut.ac.ir, razzazi@aut.ac.ir

Abstract: *Mobile phones are ubiquitous today; this provides us a huge opportunity to provide services through them. Mobile phones have constraints in their memory capacity and processor, but still they have the same capability of PCs in early 90s. Many investigations have been made on grid services, but we have a lack of investigation on providing service grid functionality on mobile phones. Mobile phone grid concept, uses current infrastructure of grids, and only adds mobile phones, as terminals of processing, or using this current services, nevertheless having all other modules of grid computing on servers. We investigated providing service grid functionalities on mobile phones. In this paper, we will define scenarios that mobile phones could be used as service providers. We see mobile phones as servers not simple terminals. By using the underlying society network, we will show how services could be provided. Our contribution is providing the required mechanism to have mobile phone service grid. Many mechanisms developed previously for service grids are not suitable for unique features of mobile phones. Thus, we developed the required functionality, which is a new break through to the new world of having ubiquitous services on mobile phones.*

Keywords: *mobile phone, grid computing, service.*

1. Introduction

Grid computing has been developed rapidly in the last decade. It provides high performance computing, for scientific applications [1]. Interpretability is provided by adding the concept of service to grids. Grid infrastructure was developed to use the processing power of most servers [2]. The architecture uses a seamless network that is distributed. It uses many previously developed concepts of search engines, metadata, which are structured and in hierarchical form to support distribution [1].

Mobile phones have the same characteristics of grid components, which make them different from distributed systems [13]. Distributed systems are under the same domain policy. On the other hand, grids have different domains, with different policies [1]. The same is true for mobile phones. They are personalized and full of personalized data. Management of this environment could not be conducted based on a centralized manager. Each mobile phone could have different policies, so it needs the same concept of virtual organizations like what we have in grids.

The number of mobile phones is three times more than the number of PCs. Thus, mobile phones together could provide a huge processing power, and memory capacity. In order to leverage these capacities, distributed mechanisms should be developed. These mechanisms could not be the same as what we have in a grid environment. Grid computing has a high processing power, and a high memory capacity, but here mobile phones do not have these facilities. Moreover, input and screen capabilities are heterogeneous [10].

While grids consist of heterogeneous devices, this challenge becomes more severe on mobile phones. We not only have device heterogeneity, but we also have network heterogeneity, that should be handled. Mobile phones mostly support Bluetooth, and some support WLAN. So we do not have a seamless network like grids. Since mobile network is heterogeneous they are imposed by high network failure [12].

GRID computing has emerged as the dominant paradigm for wide-area distributed computing. The goal of the original Grid concept is to combine resources spanning many organizations into virtual organizations that can more effectively solve important scientific, engineering, business and government problems [10]. Grid computing has recently migrated from traditional high performance and distributed computing to pervasive and utility computing based on the advanced capabilities of wireless networks and lightweight, thin devices. As a result, this has seen the emergence of a new computing paradigm, which is the Mobile Grid. Mobile Grid is a full inheritor of the Grid, with the additional feature of supporting mobile users and resources in a seamless, transparent, secure and efficient way [12]. There are three schemes for the integration of mobile devices into the Grid. First, mobile devices are merely interfaces to resources available in the Grid system, and do not contribute any services. Secondly, raw resources such as CPU, memory, and storage in mobile devices are used to complete the tasks in the grid environment. The second method just considers mobile devices as conventional resources to achieve goals. The third scheme is to exploit services in mobile devices to supporting mobile services in a mobile grid; this method is to enable mobile devices to contribute services [9]. Many research has been conducted on mobile grid until now [4][5][6][7][8]. The main difference between our proposed model and previous research is that, all previous research focused on adding mobile devices to current grid environment. This means that only processing, or request is done by mobile devices, and other modules of grid computing should still been conducted on current infrastructure, i.e. severs.

To support service grid capability on mobile phones we should provide a workflow on them. In general definition workflow is an automation of a process consisting of many participants called sub processes, where information or data may be passed between participants based on a predefined set of rules [1]. In this sense, we have new services, formed from other services. Workflow defines order of execution of these services or activities, and this combination is to solve specific problems automatically. Through workflow coordination data dependencies should be considered, and orchestration of resources should be provided.

The advantage of our proposed system is that it provides the capability to build dynamic applications over heterogeneous distributed systems; each mobile phone could have its own policy, or autonomy, but all mobile phones together could provide thousands of services for consumers. Any software system, on mobile phones, for their unique characteristics, should have different nonfunctional requirements. This system should be lightweight and optimized for a lower memory capacity and a lower processing power, and battery power that is the most critical attribute and it must be preserved. Since mobile phones are heterogeneous, adaptability is a very important nonfunctional requirement. Flexibility is also desired for systems on mobile phones. These entire nonfunctional requirements are considered in our design. A decentralised approach is used in the design of model. In a centralised approach, the host controls all communication and data exchange within the wireless network. Although this provides better control, a single point of failure is evident.

The remaining parts of this paper are organized as follows: Section 2 discusses related work. Section 3, discusses scenarios for requirement of service grid capability on mobile phones. Section 4 provides a definition for society network, and explains why it is suitable for mobile platforms. Section 5 discusses our solution for data service functionality on mobile phones, and provides details on how differently the challenges should be overcome. Section 6 discusses the evaluation issue related to our proposed solution, and section 7 concludes the paper with highlighting our contribution.

2. Related work

As we discussed in introduction section, all previous work, has been conducted on mobile grid, focused on adding mobile phones to current infrastructure, they use seamless network, and all modules of grid computing still conducted on the servers. This is not appropriate on mobile phones, since data transfer cost through GPRS, is so much, and people couldn't afford it. This makes their models with low usability. In our proposed framework, we tried to omit previous infrastructure, and we provided mechanism, and new software system, to have service grid capability, only on mobile phones. We used mobile ad hoc network (MANET) as our infrastructure, and so our cost would be much lower. But this new concept needs all previous mechanisms to be redesigned. Research has been conducted on various models that support collaboration. However most of these models are not scalable and depend on specific technologies. Additionally these models are not extensible, meaning that they only support designated services. The models depend on an existing wireless infrastructure and operating within the range of a wireless access points. Ad-hoc or infrastructure wireless environments, allow users to make use of these services and not rely on a wireless access point. Reference [13] also discusses the design and development of an extensible model to support collaboration services between mobile devices in a wireless environment. All this work, are limited to only communication layer, and none of them focused on modules of higher layer, to provide service grid capability on mobile phones. In this section we will discuss previous work has been conducted on mobile grid.

Reference [4], present a prototype of a grid-based problem-solving environment for wireless mobile devices with limited processing power Its primary purpose is to allow mobile devices with limited resources to solve problems that they would not be able to solve individually. This goal is achieved by redistributing the computational load among many computing devices. Reference [5], presents a middleware infrastructure able to integrate mobile devices in the grid. As a matter of fact, classic grids do not provide mobile users with support to access resources and services at all (i.e. classic grids consists of wired, pre-configured, powerful stations). Moreover, whenever a user wants to execute her own application, she typically has to: (i) ask the grid for resources; (ii) allocate tasks; (iii) launch and control executions; (iv) get results; and (v) release resources. Reference [6], proposes a MABS platform on the computational Grid for mobile distributed computing. The findings of the experiment show that a Grid based MABS platform can provide a scalable simulation environment for mobile distributed applications. Mobile grid combines mobile computing and grid computing and develops rapidly. Reference [7], provides design of a general architecture of mobile grid, illustration of mobile grid architecture design principle, presentation of a novel architecture, analyzing of the mobile grid constitutes and logical structure. It also discusses mobile grid resource and service management and allocation mechanism from description, discovery, security, QoS and selection and assignment factors, etc.

3. Scenario for service grid on mobile phones

Grids are used for scientific, financial, commercial application which needs high capacity, and large processing power. But such applications could not be used on mobile phones. Mobile phones are portable and personalized, which are the key points for providing applications on mobile phones.

In economy the demand curve of market is derived from the demand curve of each consumer. In marketing, we do market research, and aggregate the data we collected from our questionnaires. We do the same aggregation from questionnaires to evaluate the quality of service. We aggregate data to understand the signal of market in finance. Aggregation is done also in production, when we want to have the master actual plan. We do aggregation to understand trends. When the data come from the society, collecting them all in one place is somehow hard and impossible. People are reluctant to open their laptops, or fill forms just for the person who polls to understand the society signal. But these data could simply be available on mobile phones. People are convenient with their mobile phones, and if we have aggregation services on mobile phones, aggregation of people's information is simply by being among them, and run an automatic poll collector of mobile phones. In this way, you should not pay even one cent to the network operator, while you are using Bluetooth of your mobile phone.

Most of the polls are conducted in one place. When people are in a buffet may fill the poll forms, but when they go home, and sit in front of their laptop, they are not really willing to fill your poll form, when you want to know whether they are satisfied with your service or not. This is because service is done in the place, and so if you do not poll in the same place, same time, the information validity may decrease. The other scenario is when a person walks in a street they lives. By collecting the requirements of households, and aggregating them, they could do business. They could buy the aggregated requirements, and deliver them at the door, of each neighbor. This system on mobile phone could solve under stock information availability which is not available now. Big store chains try to propose free of charge products, for whom encounters finished items. This is all done to know under stock and optimize inventory management system.

If people have their shopping list on their mobile phones, and check the item they were willing to buy, which is already finished, the store manager can simply collect the data from their mobile phones, when they wandering in the store, and this system is win-win for both the store manager, and the customers, because one can provide better services, and one can receive them. The same scenario is applied when the government wants to increase the quality of service and uses the polling system. The main quality of mobile phones that makes them suitable for these scenarios is personalization, which you could never find in any other platforms, except mobile phones. The main service that could be provided using data is aggregation, which is the main requirement of any businesses, and research. Over this aggregation services other statistical and mathematical calculation could be provided. This aggregation could be provided more than one level. In each level we could have many different services. In the following section, we describe the concept of society network, which is the underlying infrastructure to provide services. After that, we will go to the detail mechanisms, needed to support these scenarios on mobile phones.

4. Society network

There are four options available for providing a middleware, over mobile phones: Bluetooth, SMS, WLAN, and GPRS. GPRS is based on GSM. In the middle of these two technologies is the improved version of GSM, which cellular operators enhance their systems, to improve to GPRS. SMS is built over GSM; it is simple, fast, highly flexible, scalable, wide spread and user friendly. The specification of each of these infrastructures is shown in figure 1.

	Bluetooth	WLAN	GPRS
Bandwidth	1 Mbps	11Mbps	115-117 kbps
Power	1-10mw	50-70mw	200-800mw
Range	10-100m	100-200m	1KM
Cost	None	Low	High
Frequency	2.4GHz	2.4GHz	900/1800MHz

Fig. 1. Network Medium Comparison

To make decisions over these infrastructures, Bluetooth shows up, for it is free but has some shortcomings upon its limited range. Bluetooth and WLAN support the concepts of same time and same place. At first sight, we may think that this would not be a suitable infrastructure to provide service grid capability. But when we look thoroughly, we find out that there is another thing, which is society network. Society network means people we meet every day, in buses or bank queues, waiting for our flights. We are not staying in a place day and night, and we are moving, and this is a great opportunity for providing exchange with many people. Although we meet and talk with limited people, our device is not limited like us. It could automatically interact with many people we may not see, but they are in our proximity, and it could match profiles, use their processing power, or memory. Our device could even trigger and guide us to the right person. With this mentality, Bluetooth is not limited anymore, while we are not Robinson Crusoe who don't have any people around us in an isolated island, even when we are in an island there are many people around us, and this means a huge opportunity for computing and commerce. The main advantage of Bluetooth is that, we are not controlled by central agents anymore, there is no operator, and we could conduct commerce without the middleman (operators).

After that we have SMS, which is accessible from anywhere, but has a shortcoming in terms of cost and middleman controls and regulations. The third suitable choice is WLAN which is also free and suitable, with the ability to connect to the internet, and the fourth one is GPRS, due to having a high cost of transfer; it makes it difficult for a normal application to communicate. To overcome the shortcomings of each of these options there is a solution, which is a software layer over them for roaming transparently, so that the application over them wouldn't be notified that the underlying layer has changed the network medium. Thus, the first underlying layer searches for the mobile phones in the proximity to check whether they have the special service. If they do not, the software layer under it does the roaming and goes to WLAN or SMS according to the user preference between cost and time. In addition, if it is not found, then automatically it connects via GPRS to another mobile phone, or sends an SMS to a default mobile phone that he knows it contains the required service.

5. The proposed solution for service grid on mobile phones

The workflow in grid has four components. The first one is the service provider, which has a unique name. The second one is the activity model, which has an identified end point. Moreover, it contains all the activities of the provider. The third one is the composition model, which is a data flow between services that could be a direct flow, or a direct communication among them. The fourth one is the data model which describes data flow that contains the source/sink notification, and provides publish/subscribe capabilities. In grid service publication, search is very easy, and it is implemented using the previous methods developed in computer science, but since we have constraints in mobile phones, it is not possible to use them on this platform.

The first point we should discuss in order to provide the service grid capability on mobile phones is the data flow and how they should be represented. We define DID for each data item that is a unique number. Each data item has attributes that are shown as XML files. The consideration for mobile phone is that, data should be lightweight and we should avoid the nested XML. This provides scalability but with this consideration using data item could be possible. For DIDs, distributed services exist that provide the name of DID on mobile phones by getting DID and their schemas. Each user has also a UID, which is the phone number of that person that is unique in the entire network. Each service has also SID, which is unique across the network; it is derived by mapping signature of service, and metadata, to a number by hash functions. There are also services, which by getting SID, will give a short description of service. We will talk about the hash function, and uniqueness, and complete description of each service later in this section.

In the workflow, we have four types of relations between services. The first one is sequence, which means services are executed sequentially. The second one is parallelism, which may be possible during execution. The third one is choice, which uses dynamic conditions to execute. Finally, the fourth one is iteration, so a counter exists in this case. Workflows on mobile phones should be shown by XML files. The point is that this definition helps to have sub workflows that could be used to distribute workflow parts to be executed on different mobile phones. The elements of a workflow are DID, SIDs, and the relations that could be defined (This XML file could be generated by a simple adapted graphical design). In addition, workflows should not be nested due to the parsing overhead. Thus, it is better to break workflows; so that they would be lightweight and define different workflows, and use them sequentially, on local systems (also we should consider that our scenarios do not need a nested workflow). Input and output of each service is data item, so in our workflow, we use DID, which has a specific scheme, and condition, that is used to filter it according to the requirements. Each request for the service could be wandered on different mobile phones. If your mobile phone requests for the service that is not in the approximate mobile phones, they will put your request according to their policy on the list, and when they become in approximate of new mobile phones, they will do the same thing as your mobile did to them. Thus, after some time your response either directly by SMS, or indirectly by that person, next time he visits you again will return to your mobile phone. We will discuss the publishing later in this section.

For workflow execution we also have two options, first we could have workflow wandering like service request wandering. The advantage of this solution is that for data intensive services, it does not require to transfer so much data to you again, when you want to use it again as an input of another service. In this case, the service is executed near the data, so it will be optimal. But this approach will lead to wastage of battery power for transfer when workflow is not as lightweight as data in non-intensive data services. In this case, the second solution is appropriate, the workflow is persistent on the system, and just request for services wander over different mobile phones.

Our approach which contains granular services wandering that is lightweight, not only leads to a higher throughput, while different parts of the workflow could be executed on different mobile phones, but also it does not lead to high waiting time for large service executions. Consequently, it will result in a balanced load. We will define other mechanisms in this section which helps load balancing over mobile phones. These two approaches we proposed could be used automatically by specifying whether the service is data intensive or not. The specification is done by user. The main requirement to provide service grids on mobile phones is the publishing service. How could we publish data on mobile phones especially on society network? Grids provide hierarchical system, while we have a seamless network on grid, this is possible. But our proposed society network is not seamless, so such solution does not work. The broader challenge is that, we don't know even topology of society network.

What we propose is an adaptive request/response method. Our adaptive request/response method keeps track of data items that are done by the provider of services. In this case, if the provider of service finds out that this mobile phone asks for a specific service more than the threshold, then the metadata defining the service could be created on that mobile phone according to the local policy of the requester. After a few requests the optimal place of service advertisement could be found that could be used by the requester. Thus, our approach consists of two main parts, first queue, which preserves the request until it could be responded, or the deadline expires. The second part is a self aware service, which puts address of the requester in its cache, and if the requester asks for data, more than a threshold, it will provide service advertisement on that node.

This queuing also helps us to have routing information on mobile phones; it means that, each intermediate node that keeps the information request for some time could keep track of who answers this request, and instead of broadcasting the request in next stages, it will send the request to a specific address. This address could be either intermediate or final.

You may think that this approach only solves the problem of not having a seamless network, what about lightweight requirements of applications. What we propose is using a mechanism like cache; we could implement the Least Recently Used mechanism. Each service could only have a limited cache that could be configured according to a specific mobile phone's requirement. We will explain this issue, later in this section. In our adaptive approach we could also do omissions, which mean if more than some time we send someone a request, and they do not answer it, the probability that we send this person our request the next time will decrease. Our approach uses probability that is a number between 0 and 1. Our adaptation is done by simply calculating the number of times we did calculations over the number of times we requested for each mobile phone.

To complement our approach, for lightweight services, that means the content data is only a small number of SMS volume, the requester, himself, could have a list of popular services. Then he could ask for the service by SMS. This is something like having a phone book for data services. The request for specific data roams over the society network. We should have a mechanism to trust the service and data that we acquire. For this purpose, we propose the rating mechanism that different people rate one person in a specific domain of service and data. This could be done first according to the previous knowledge or after that when people use services or data. This rating mechanism could also have negative effects on the rate of each data provider. Since according to economical theories, a large number of people could not coalite so this rating could provide solution for the challenge, since we have a high distribution. Nodes could also provide invalidate poller services. So if someone haphazardly answers the poll, it could be identified by the statistical aggregator service, and after some time, credit of this person's answer will decrease. The search service is not a big issue for grids. But when it comes to mobile phone platforms, it is really a challenge, because mobile phones do not have enough processing power, and do not have access to huge metadata related to data items. Even if we want to have a simple search over mobile phones, the input capability is very limited, and it makes it difficult to search over them. But what is the solution for mobile phones? If we cannot have search, then there would not be any data grids over mobile phones.

We think we should use a PC platform to provide search, but on mobile phones, each service has a specific SID and each data item has a specific DID, as we explained before. We leverage a PC platform to find SIDs and DIDs. Since services and data we are searching for could be repeated over time, a simple mapping of domain, especially ontology to services and data items could solve our problem. This mapping is between metadata to SID and DID by using a hash function. But if a person does not have time to go to a PC, what should he/she do? We could have two solutions, first a centralized service, which you can contribute by SMS, then that center, there could be automatic machine, which according to your history of preferences helps you, or it could be a person behind SMS system, or you can use the call center to find out DIDs. The point is that using web applications, even configured for mobile phones, is not desirable on mobile phones, since it has a limited input ability, and a small screen, so we recommend using our two solutions.

Each request for service has a deadline. Additionally, it has a priority, which is identified by a credit of a person. The requests also have levels, which identifies how many people this request could roam to. If deadline exceeds and the number of person this request roamed exceeds level of the request, request will be killed. This is necessary for avoiding congestion in this process and memory intensive environments. Requests for services are put in a queue in the service providers' mobile phone. When request wants to in the queue for service, it first asks the provider how much time he thinks it should wait. The service provider based on a simple calculation of estimated time of service, deadline, priority, input data, and history data, will provide the estimated time. Then if the estimated time for the service is lower than the deadline of the request, it will stand in the queue. When it is the turn for the request to be served, the service provider asks the main requester (the address is in the request), whether they get the response yet or not. If the answer is no, the service provider will provide the service. The main point we should consider hear is that the estimated time for the multi service provider is calculated based on the requests in the queue of all services. This mechanism we provide could also lead to a balanced load. While a request will not stand in the queue, it will not answer the request in the deadline.

Each service provider has also its own queue for the services they do not provide. Our workflow system is stateful, in both solutions we proposed. To fulfill fault tolerance, we could use ordinary methods, like replication (limited to two or three), or check-pointing. We should consider that even in a roaming workflow, each time it is only on one mobile phone, unlike service request that may be in more than one mobile phone request list. This is for avoiding congestion in the network. The components of our system are depicted in figure 2. Details of interaction are explained in the text.

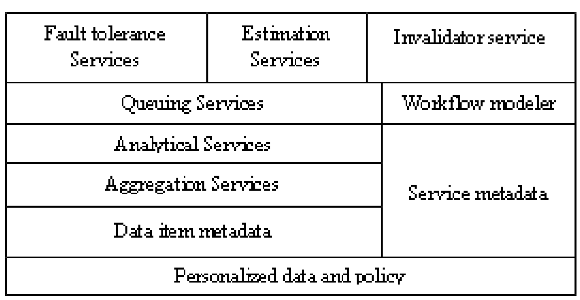

Fig. 2. Software system proposed for providing service grid on mobile phones

To improve performance of our system, cache services should be used for services, and data items, and favorite services and data items. To advertise services, they preserve the history of requests and will omit the advertiser, in case the request for that service is decreased. Each advertisement also has an expiration date, it also has a rate. According to these data advertising could be omitted. In our system, also each request has QoS requirement related to the rate of service, and it is identified by user, according to reliability of service they need.

6. Evaluation

To evaluate our system, we should first have other systems to compare with. Unfortunately there is no such system to compare with, and also previous works on grid were not suitable for this platform, for unique characteristics of this platform. But we compare the proposed system with nonfunctional requirement of this specific platform.

First being lightweight, is the main nonfunctional requirement, it is supported by our layered architecture. Modularity of proposed system provides flexibility, and adaptability. This modular system, could provide even lighteweighness, which means someone who has a mobile phone with a higher memory capacity and processing, can install more modules, and enjoy optimized applications, and less battery power usage. Our proposed system also provides adaptability, that means the threshold of caches, and other parameters could be set according to special devices requirements.

Our mechanisms are optimized and less computational, and more distribution makes it more suitable for mobile phone platforms. Also our platform is scalable, since our mechanisms are configured for distribution, and do not depend on the central component.

7. Conclusion

Today mobile phones, are widespread, however there is no software system to provide service grid capabilities on them. Most previous works were focused on looking at mobile phones as terminal devices. We changed the view and looked at mobile phones as servers. We defined the concept of society network, which could be leveraged to enhance mobile phone service grid capability. Our contribution was redefining mechanisms of service grid modules on mobile phones. Due to the unique characteristics of mobile phones, previous solutions were not suitable, and we defined the mechanism ourselves. Our publish service was not done globally, like grid, but we used locally publishing; we used self aware service and data items concept. Our approach was adaptable, and makes the performance better over time. We used caches to improve performance. Our architecture provides flexibility, adaptability, and performance for mobile phones. We identified new scenarios for using mobile phones as service providers, which was focused on aggregation. We talked about details of mechanisms for workflow, which are configured for specific characteristics of mobile phones.

Our work could be enhanced by investigating thresholds for different mobile phones. Also simulating our framework and investigate how granular each service, and workflow should be, could be issued for future investigations.

Acknowledgements

This research was created through a financial grant offered by Iran Telecommunication Research Canter.

References

[1] F.Magoules, J.Pan, K.A.Tan, A.Kumar, "Introduction to grid computing", Taylor & Francis Group, 2009.

[2] H.Sunaga, T.Oka, K.Ueda, and H.Matsumura , “P2P-based Grid Architecture for Homology Searching”, Proc. IEEE. 2005.

[3] J.Cohen W.Lee J.Darlington A. S.McGough, “A Service-oriented Utility Grid Architecture Utilising Pay-per-use Resources ”, Proc. IEEE, 2006.

[4] S.Kurkovsky, Bhagyavati, A.Ray. “Modeling a Grid-Based Problem Solving Environment for Mobile Devices”, Proceedings of the International Conference on Information Technology: Coding and Computing 2004.

[5] A.Coronato, Gi. D.Pietro: MiPeG: A middleware infrastructure for pervasive grids. Future Generation Comp. Syst. 24(1): 17-29 (2008).

[6] D.Mengistu, P.Davidsson, L.Lundberg, “A Grid Based Simulation Environment for Mobile Distributed Applications”, Proceeding International Conference on Multimedia and Ubiquitous Engineering, 2007.

[7] W.Y.Zeng Y.L. Zhao, J.W. Zeng , W.Song , “Mobile Grid Architecture Design and Application”, Proceeding 4th International Conference on Wireless Communications, Networking and Mobile Computing, 2008.

[8] K. Katsaros and G.C. Polyzos, "Evaluation of scheduling policies in a Mobile Grid architecture," Proceeding International Symposium on Performance Evaluation of Computer and Telecommunication Systems, 2008.

[9] Chunlin Li, Layuan Li, “Energy constrained resource allocation optimization for mobile grids”, Energy constrained resource allocation optimization for mobile grids. In Press, Corrected Proof, Available online 24 June 2009.

[10] M. Parashar and J-M. Pierson, "When the Grid becomes Pervasive: A Vision on Pervasive Grids,", Proceedings of the 8th Hellenic European Research on Computer Mathematics & its Applications Conference, 2007.

[11] R. Moreno, “A Hybrid Mechanism for Resource/Service Discovery in Ad-Hoc Grids”, Journal Future Generation Computer Systems , 2008.

[12] A. Litke, D. Skoutas, K. Tserpes, T.A. Varvarigou: Efficient task replication and management for adaptive fault tolerance in Mobile Grid environments. Future Generation Comp. Syst. 23(2): 163-178, 2007.

[13] L. Son and M. Calitz, “A Model for Collaboration Services between Mobile Devices,” *Proceeding the South African Telecommunications Networks and Applications Conference*, 2006.

Authors

Meisam HEJAZINIA is currently MBA student at Sharif University of Technology. His research interests include enhancing distributed system capability on mobile phones. He received his MS degree in computer engineering from Faculty of Computer and Information Technology Engineering of Amirkabir University of Technology in 2009. He received BS degree in computer engineering, and another BS degree in Information Technology Engineering, from Amirkabir University of Technology. He is author of 10 articles in the field of distributed processing on mobile phones.

Mohammadreza RAZZAZI is currently an Associate Professor at Amirkabir University of Technology. His research interests include distributed processing on mobile phones, software development on mobile phones, grid computing, software engineering best practices, and computational geometry. He received his MS in computer science at Stanford University, and Ph.D. degree in Computer Science from University of California, Santa Barbara. He is author of more than 90 conference papers, and more than 10 Journal papers.