An e-Resource Trading Paradigm for Computational Grids

Kiran Kumar Pattanaik†, Rabin Singh and Gadadhar Sahoo
{kkpatnaik, rabinsingh, gsahoo}@bitmesra.ac.in
Department of Computer Science, Birla Institute of Technology (Deemed University), Mesra, India

Summary
The concept of coupling internet-wide computational resources (high-end computers and low-end personal computing devices) to form a huge pool of compute resources that would provide cost-effective renting services is not new. Various approaches and economics have been initiated for the resource management in Grid, but so far no economic initiatives have been taken for creation of resource pool. Moreover the mechanisms proposed so far do not guarantee a minimum expected return-on-investment for the resource providers no matter how costly their services are as they are primarily governed by volunteering first and then generate revenue policy. This may cause the resource consumers maximize their time/budget/Quality-of-Service objective and leaving the resource providers' – making the system one sided. In this paper we propose a distributed Resource Market Place concept that is based on dynamic cost model and adopts economic institution paradigm for Compute Market creation and resource management (discovery and scheduling) in the Internet scale distributed resource pool. Our proposed model ensures both the resource providers and consumers maximize their objectives through different auctioning strategies and is scaleable.

Key words:
Grids, Computational Economics, Compute Agent, Auction Server, Proxy Auction Server.

1. Introduction

Resource discovery is a very important function of the resource management. Discovery services are used by the scheduling systems to obtain information about resource availability. This can be query based or agent based. Most contemporary grid systems follow the former approach. Use of idle or underutilized computers to perform resource intensive processing has been adopted by a number of projects and commercial ventures such as SETI@Home project [1], Entropia (Entropia Inc.) [2], Condor [14]. Condor has a centralized periodic push based data store and implements query based mechanism for resource availability information on a network. The Resource brokers in Globus [5] discover resources by querying the information service (MDS) for resource availability. Globus implements discovery services the same way as Condor does. Nimrod-G is grid enabled resource management and

scheduling system based on the concept of computational economy. Nimrod-G [3] uses the MDS services provided by Globus for resource discovery. GRACE [7] components coexisting with Globus implements the trading protocols to interact with resource owners and negotiate for access to resources at low cost. Most of the resource management architectures employ query based resource discovery and push or pull dissemination of resource status. Support for fault tolerance is mostly through replication of the components or recovering using persistent state. In this paper we propose a model that adopts economic institution paradigm for the resource pool creation and discovery of suitable resource from the pool. All the above systems do not scale well and suffer from single point of failure of the centralized information pool. We propose a bid filtering mechanism that rejects any unnecessary low bids thus making the system able to handle many auction requests without considerable increase in unmanageable traffic . In addition the dissemination mechanism used here is demand driven rather than conventional push or pull based.

The rest of the paper is organized as follows. Section 2 describes the related work. The proposed economy driven resource management architecture is presented in section 3. In section 4 we present the protocols. Section 5 discusses the simulation results and finally section 6 concludes with future work.

2. Related Work

Harnessing computing power of geographically distributed high-end resources and Internet-wide home computers to perform useful processing is not a new concept. It has been adopted successfully by SETI@Home project, ProcessTree [4], Entropia, Globus and Legion [6]. The Grid Architecture for Computational Economy (GRACE) adopts economic paradigm for resource management and scheduling for high-end Grid computing systems and claim that with suitable changes in implementation architecture and the underlying infrastructure it can be extended to low-end machines as well. A few other systems such as Popcorn [8] build market-oriented environments to harness the processing power of small network-of-computers configuration. The

Manuscript received July 5, 2007
Manuscript revised July 25, 2007
† Corresponding author

Compute Power Market architecture [9] claims to offer a true market-oriented Internet-scale computational Grid. There has been no attempt in devising an economy driven paradigm for the market (resource pool) creation and a mechanism to deal with the unnecessary low bids. Our proposed approach blends the basic idea behind these attempts with economic principles and goes a step further to create a market-oriented computational Grid that is scalable and guarantees that both the resource providers and consumers maximize their objectives through different auctioning strategies.

3. The Resource Management Framework

3.1 Overview

We define the Grid is a marketplace for resource providers and the service users. It motivates the providers (resources of any individual or enterprise intranets) to join the marketplace by creating an assured return-on-investment scenario through auctioning rather than first volunteer and then generate revenue approach. On the other hand the Grid resource users also able to maximize their objectives through reverse auction thus creating a win-win scenario. This guarantees maximization of objectives of the providers and the users and creates a healthy competitive market environment. According to these properties we present the abstract view of Grid Resource Marketplace (GRMP) (Fig. 1).

In the GRMP there are a core circle formed by Compute Agents (CA) and their associated resources. The users with a variety of resource requirements and the prospective resource providers have access to the GRMP through the Market Interface (MI). GRMP is dynamic in the sense that resources can join a CA and leave the CA dynamically. Similarly the CAs can also join and leave the marketplace dynamically. A CA can be perceived as a passive agent in the GRMP, in that it acts as a mediator between users and the resource providers (Fig. 1). Several CAs are grouped together to form virtual groups (VG) for better management. The group management protocols are not described here for simplicity. The number of CAs in a VG varies from group to group and depends on the sharing principles. The MI consists of Auction Server (AS) and Proxy Auction Servers (PAS) (Fig. 2). The purpose of AS and PAS are to carryout auctions:

1. For discovering suitable resource(s)–budget constrained, deadline, QoS - for the user willing to use the Grid resource.
2. For discovering a representative CA incase a provider is willing to participate in the market.

There can be more than one PAS and one AS in the entire marketplace. The presence of PAS is to work on behalf of the AS in the sense that the PAS forwards only the valid bids to the AS and filters unnecessary low bids. This enables the AS less burdened in handing multiple bids in a small time and decreases the bid response time thus making the system scalable.

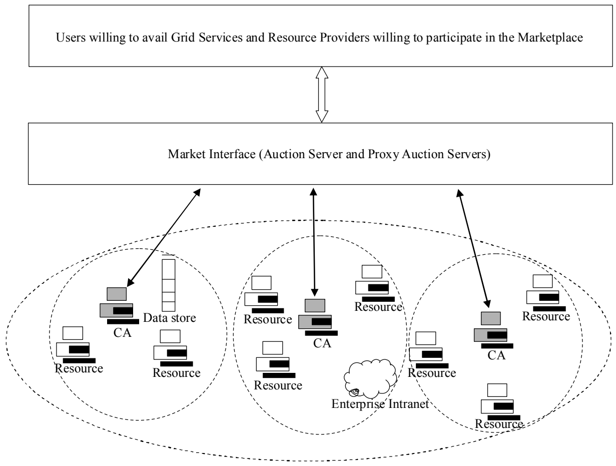

Fig.1 Grid Resource Market Place (GRMP).

3.2 Proposed Auction Framework

The main components in this framework are the Users, Providers, Compute Agents, Proxy Auction Servers and Auction Server (Fig. 2).

The AS holds an auction among the CAs interested in participating in the bid process when:

1. Provider willing to participate in the GRMP submits request to AS seeking the highest possible return-on-investment (Fig. 2 and Fig. 3 (a))
2. User willing to use the Grid resource submits a request with AS seeking the lowest possible charges for task execution (reverse auction) (Fig. 2 and Fig. 3 (b)).

First of the above two points is the prime motivation behind the market creation. The economics behind this is, a CA to have a larger provider base and play a volume game (example of minimum margin and larger customer base for a company). CA is a representative for a resource in the GRMP based on some agreed percentage (outcome of an auction) of earnings to be given to the provider it is representing.

This school of thought forces a CA to siphon a greater percentage of its earnings to its associated providers and retain a small pie. Hence, there is always a competition between the CAs to have a larger provider base to maximize their gross earnings. The agreed value is in most cases higher than the expectation of the provider because the agreement between the available CAs and the willing provider is accomplished through an auction process.

4. Auction Protocols

The prime players in GRMP are the users of resources, CAs and the providers. Users want services at the least possible cost whereas providers want the best possible return-on-investment. In this scenario CA is the major role player in meeting the requirements of both users and the

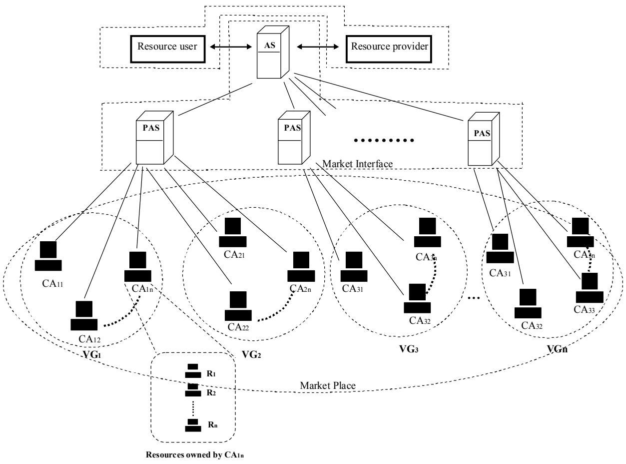

Fig. 2 Model for the proposed architecture

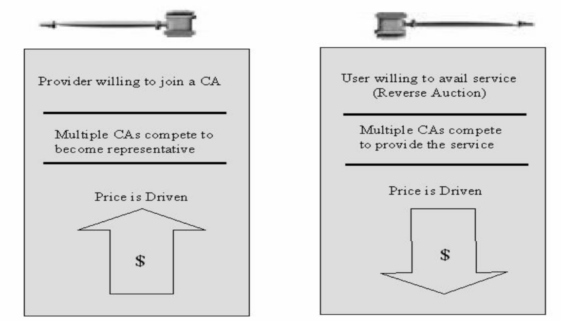

Fig. 3(a) Auction process and Fig. 3(b) Reverse Auction process.

providers. The AS serves as an auction gateway for both users and the providers. In case of users it holds a reverse auction and in case of providers it holds an ascending auction. Similarly the PAS stores the local maximal bid value and local minimal bid value for provider and user respectively. Besides this the PAS also stores the global maximal bid value and global minimal bid value supplied by AS. It is assumed that the CAs know to which of PASs to connect. With this local and global information a PAS is able to filter out unnecessary bid information. Steps 8 and 7 in figures 5 and 6 respectively correspond to comparison between the current value with the new bid value for a service (user seeking minimum service charge for task execution). For a provider willing to join a CA, the "<" condition becomes ">" in the steps 8 and 7 in Fig. 5 and Fig. 6 respectively (provider seeking maximum return-on-investment). The PASs enable the AS to be less burdened with unnecessary low bids and make the system scalable. Applying this concept, the CAs, the AS, the PAS acts as follows. It is also assumed that each CA, each PAS and an

```
while (auction_held)
{
   curr_val=request_curr_val(proxy_auc_server);
   bid_val=determine_bid_val(curr_val);
   send_bid (bid_val, proxy_auc_ser);
   wait (some_time);
}
```

Fig. 4 Procedure for CA

```
1.  bid_val=init_val;
2.  while(auction_held)
3.  {
4.   if (!time_out)
5.   send_curr_val (proxy_auc_servers);
6.   if (bid_val=receive_bid())
7.   {
8.     if (bid_val < curr_val)
9.     {
10.      curr_val=bid_val;
11.      return_success(curr_val,proxy_auc_server);
12.    }
13.    else return_decline(curr_val,proxy_auc_server);
14.   }
15. }
```

Fig. 5 Procedure for AS

```
while(auction_held)
{
  if(receive_request())
  return curr_val;
  if(bid_val=receive_bid())
    {
      if(bid_val < curr_val)
      curr_val=bid_val;
      send_bid(curr_val, auction_server);
      return_success(curr_val, CA);
    }
   else return_decline(curr_value, CA);
 }
```

Fig. 6 Procedure for PAS

AS are time synchronized and the data they retain are consistent for the corresponding auctions. Secondly, the framework uses Digital signature to authenticate each other and 128 bit key RSA encryption to encrypt the bid information. It uses the TRAM reliable multicast protocol of JRMS1.1 [17] to periodically send updated bid information from one sender (Auction Server) to multiple receivers (Proxy Auction Servers).

5. Design of Framework and Evaluations

5.1 Framework Design

The classes and methods that compose our framework are (Fig. 7):

1. **User_Message:** It is a GUI based class in which user gives its requirement to AS. It includes user name, user id, budget, deadline, QoS and pass these as arguments to the AS.
2. **Machine_Message:** A resource willing to join the market submits its minimum expected return-on-investment requirement, resource type and other terms and conditions to the AS.
3. **Auction_Server:** AS accepts input from users and resource providers and processes them.
4. **Proxy_Auction_Server:** PAS accepts bids from the prospective bidder CAs and processes on behalf of the AS.
5. **Auction:** AS and PAS perform the auction process.
6. **Compute_Agent:** CAs are the managers in the market for both users they serve and the resources they represent.
7. **CA_Characterics:** It represents the properties of CA such as machineList, userList, managementPolicy, cost, and the time zones the resources belong to.
8. **Accumulator:** Accumulator is a place holder where CAs keep their weights on the basis of their capabilities and a coordinator is elected based on the maximum of all the weights in the virtual group.
9. **CA_Calendar:** The calendar maintains arrival and finish times of tasks as well as the resource entry and exit times for accounting purposes.
10. **Machine:** A machine represents a resource under a CA.
11. **Processing_Element:** A machine can have one or more processing elements. Machine rating is specified in terms of MIPS.
12. **Perf_Benchmark:** This parameter represents the performance benchmark for a resource and this is used for identifying the capabilities of the resource.
13. **Alloc_Policy:** It is an abstract class used by the time shared and space shared policy.

Codes from users of the Grid are executed at providers' sites that ensure users safety as defined at the CA. Users' code is convoluted (Example: Entropia) at CA before scheduling at any resource to maintain the properties of the source programs intact.

5.2 Simulation and results

We evaluate the proposed method form two aspects through simulation using GridSim [18] simulator. One is bid versus number of bidders and the other being the network overhead. A Tree based network of depth equal to three with AS being the root of the tree and PASs being the intermediate nodes and the CAs are the leaf nodes (Fig. 2). We divide simulation into two parts. The first part is concerned with computational resource market creation and the second with the trading of resources among the potential consumers.

The resources are characterized by Resource Name (string), Resource Architecture (string), Resource Operating System (string), Number of Processing Elements (integer), MIPS of each Processing Element (integer), Time Zone of the Resource (double), Minimum Processing Cost of the Resource (double), Primary Memory Size (integer), Secondary Memory (integer).

A CA is characterized by it association with PAS, Reputation (Trading History).

Initially during the market creation phase there are no computational resources associated with CAs. When a computational resource wants to join the market, first registers with the AS. The AS initiates an auction process where the bidders (CAs) bid randomly at random timing. Here it is important to note that a bid is in terms of sharing a percentage of earnings with the prospective grid resource. Example: For instance, the earnings from a resource is \$ X (The value is governed by supply and demand and hence dynamic). A bid will be a percentage of the earnings that a CA will factor in the event of it representing that computational resource in the market. In such an approach, the bidding is likely to mark an upward trend with the possibility of the resource owners getting the best price for their return-on-investment in a given market situation. The parameters we focus that characterize an auction are bids against the number of bidder CAs (Fig. 8).

Figure 8 indicate as the number of participant CAs increase the amount of percentage of revenue sharing between CA and the resource owners also changes. It is observed that the share of resource owners is considerably increased thus enabling the owners to get better return-on-investment as opposed to the conventional method of volunteering first and then generate revenue approach. This encourages more and more resources to participate in the market and that eventually will bring down the cost of resource usage.

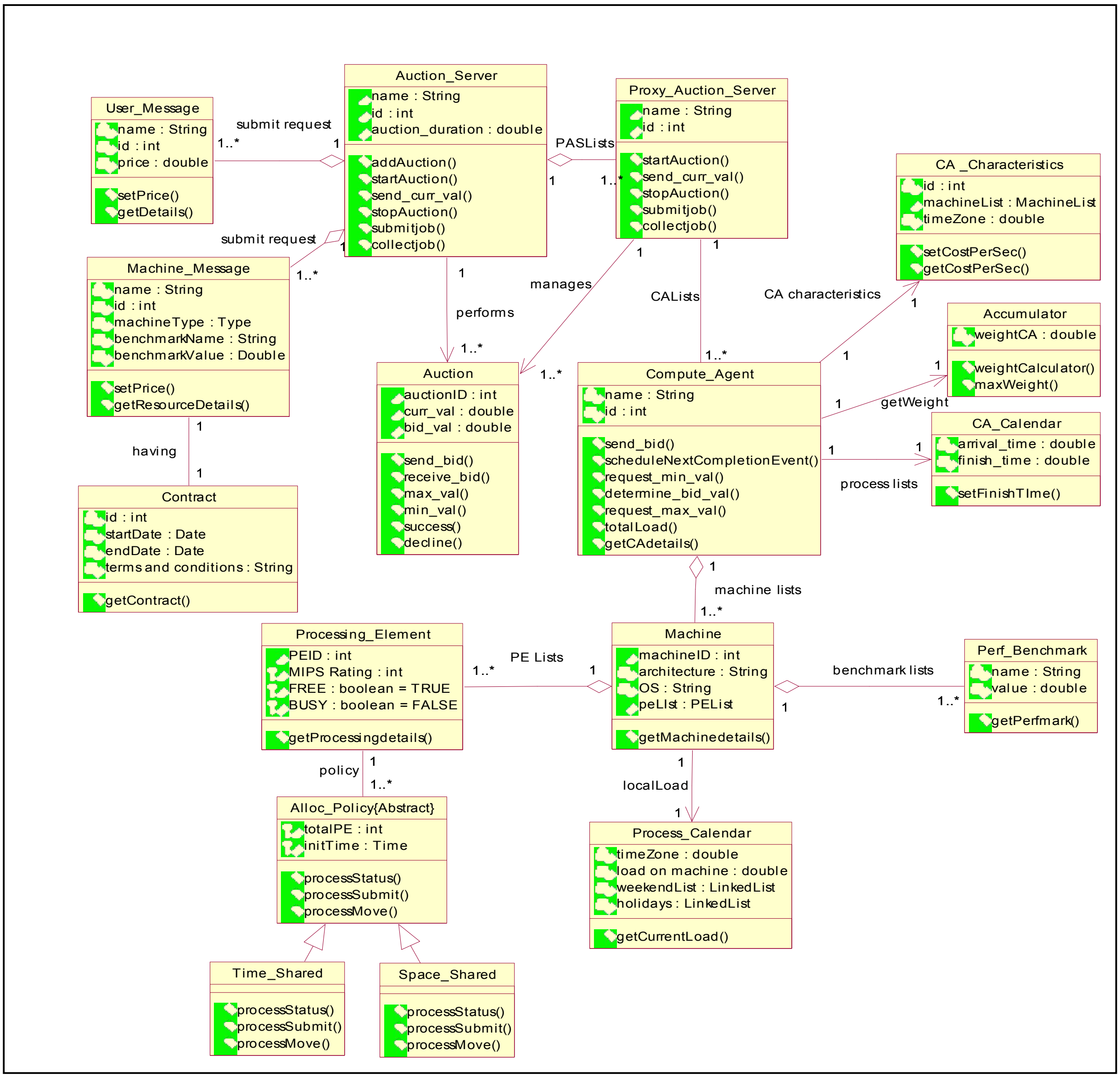

Fig. 7 Class diagram for the economy driven resource management framework

Figure 9 indicate as the number of participant CAs increase the usage cost for a resource changes. It is service to the users and relatively lesser profit margin for the CAs.
However as the number of resources associated with a CA increases the average profit margin is comparable (depends
on supply-demand) to the case of less resources and higher profit margins. The bid filtering mechanism provided in the PAS restricts the traffic to the AS thus enabling the system capable of managing large market (Fig. 11).

6. Conclusion

In this paper, we have described economy driven resource management framework for the Grid that adopts economic institution paradigm for Compute Market creation, resource management and scheduling of computations across the Compute Market. Our proposed model tends to motivate the computer owners contribute their resources only after through an auction process that assures the best possible return-on-investment in a given market scenario.

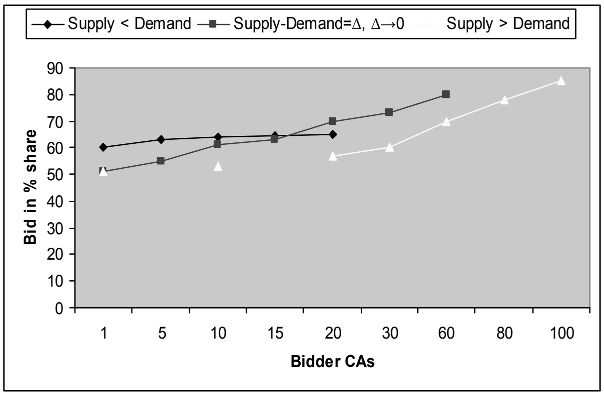

Fig. 8 CAs bidding for acquiring a resource following an auction announcement by AS and their corresponding PAS.

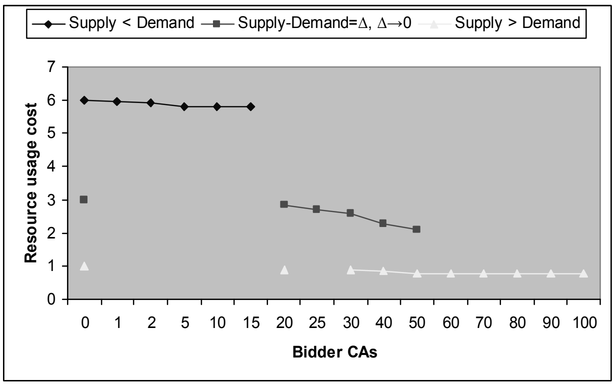

Fig. 9 CAs bidding for the usage charges of a resource following an auction announcement by AS and their corresponding PAS.

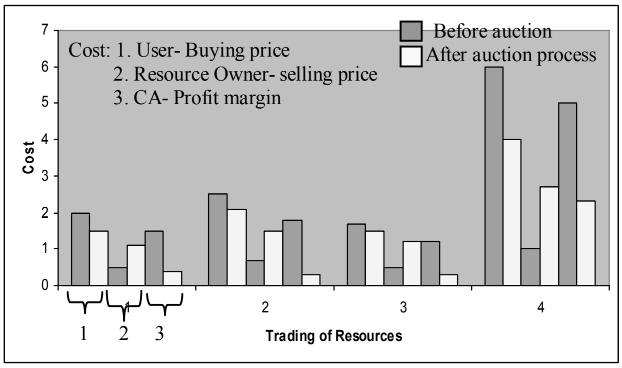

Fig. 10 Relation between the cost of providing a service, acquiring a resource and profit margin of CA for different resources before and after an auction process.

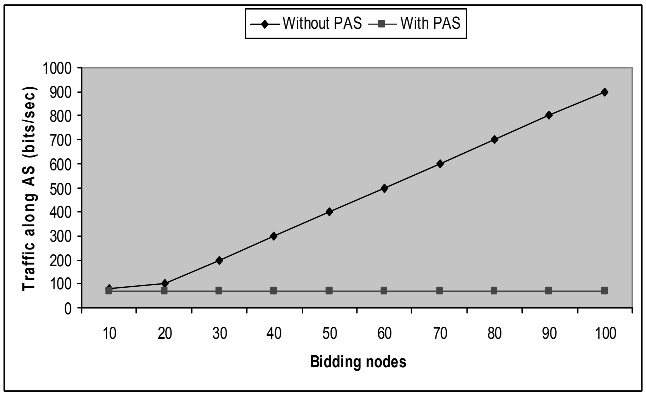

Fig. 11 Traffic along AS as the number of bidders increase.

We have presented different auctioning strategies that enable both the resource providers and consumers maximize their objectives. The bid filtering mechanism in our proposed model makes it a scalable framework by discarding unnecessary bids. This makes the AS handle a large number of auctions. We evaluated the proposed framework through simulation but the practical market will have more complex and heterogeneous properties. We expect the main features obtained by the simulation may be applicable to practical resource trading market scenario. Our results suggest that the proposed approach has a greater bearing on the creation of a practical and scaleable competitive computational resource marketplace. Our future work is towards generating the demand and supply profiles of resources where both sellers and buyers submit bids which are then ranked highest to lowest. This will help a better estimation of bids.

References

[1] http://setiathome.ssl.berkeley.edu/
[2] Entropia Inc. – http://www.entropia.com
[3] Rajkumar Buyya, David Abramson and Jon Giddy, Nimrod G: An Architecture for a Resource Management and System in a Global Computational Grid, 4th International Conference on High Performance Computing in Asia-Pacific Region (HPC Asia 2000), China.
[4] ProcessTree-http://www.processtree.com/, Distributed Science Inc, Nov. 2000.
[5] Ian Foster and Carl Kesselman, Globus: A Metacomputing Infrastructure Toolkit, International Journal of Supercomputer Applications, 11(2): 115-128, 1997.
[6] Steve Chapin, John Karpovich, Andrew Grimshaw, The Legion Resource Management System, 5th Workshop on Job Scheduling Strategies for Parallel Processing, April 1999.
[7] Rajkumar Buyya, Jonathan Giddy, David Abramson, An Economy Grid Architecture for Service-Oriented Grid Computing, 10th IEEE International Heterogeneous Computing Workshop (HCW 2001), In conjunction with IPDPS 2001, San Fransisco, USA, April 2001.
[8] Noam Nisan, Shmulik London, Ori Regev, Noam Camiel, Globally Distributed Computation over the Internet – The POPCORN project, International Conference on Distributed Computing Systems (ICDCS'98) 1998.
[9] Rajkumar Buyya and Sudharshan Vazhkudai, Compute Power Market: Towards a Market-Oriented Grid – http://www.computepower.com
[10] R.P. McAfee and J. McMillan, Auctions and Bidding: Journal of Economic Literature, 25(2):699-738, 1987.
[11] R. Buyya, D. Abramson, and S. Venugopal, The Grid Economy: Special Issue on Grid Computing, Proceedings of the IEEE, 93930:698-714, March 2005.
[12] D. Grosu and A. Das, Auction Based Resource Allocation Protocols in Grids: In Proceedings of the 16th IASTED International Conference on Parallel and Distributed Computing and Systems, page 20-27, Cambridge, MA, USA, November 2004.
[13] Kaviraju Ramana Dyapur and Kiran Kumar Patnaik, Transaction Oriented Computing (Hive Computing) using GRAM-Soft: In the proceedings of 5th International Conference on Computational Sciences (ICCS 2005), Atlanta, GA, USA, May 2005.
[14] R.J.M. Boer, Resource Management in the Condor System, Master Thesis, 1996.
[15] R. Buyya and M. Murshed, GridSim: A Toolkit for the Modeling and Simulation of Distributed Resource Management and Scheduling for Grid Computing, Technical Report, Monash University, Nov. 2001. To appear in the Journal of Concurrency and Computation: Practice and Experience (CCPE), 1-32pp, Wiley Press, May 2002.
[16] R. Buyya, D. Abramson, and J. Giddy, A Case for Economy Grid Architecture for Service-Oriented Grid Computing, Proceedings of the International Parallel and Distributed Processing Symposium: 10th IEEE International Heterogeneous Computing Workshop (HCW 2001), April 23, 2001, San Francisco, California, USA, IEEE CS Press, USA, 2001.
[17] N. Ranaldo, G. Tretola and E. Zimeo, Hierarchical and Reliable Multicast Communication for Grid Systems, Parallel Computing: Current & Future Issues of High-End Computing, Proceedings of the International Conference ParCo 2005, J¨ulich, NIC Series, Vol. 33, ISBN 3-00-017352-8, pp. 137-144, 2006.
[18] R. Buyya and M. Murshed. (2002) GridSim: a toolkit for the modeling and simulation of the distributed resource *and Computation-Practice and Experience* 14(13-15) pp.1507-1542.
[19] Kiran Kumar Pattanaik and Gadadhar Sahoo, Economy Driven Resource Management Architecture for Compute Grids. To appear in the Proceedings of the 9th International Conference on Enterprise and Information Systems (ICEIS 2007), Funchal-Madeira, Portugal, June 2007.

K. K. Pattanaik received Diploma in Electronics and Telecommunication Engineering from State Council of Technical Education (Orissa) in 1992 and Bachelor of Engineering in Electrical and Electronics Engineering from Kuvempu University, Shimoga in 1997 and Master of Technology in Computer Science and Engineering from Motilal Nehru National Institute of Technology (formerly MNREC), Allahabad in 2003. He is now with the Department of Computer Science and Engineering, Birla Institute of Technology (Deemed University), Mesra, Ranchi, INDIA in the capacity of Lecturer. His research interests are Distributed Systems and Resource Management issues in Grid Computing.

Rabin Singh received B.Tech. in Computer Science and Engineering from UP Technical University in 2004. He is a post-graduate student of Computer Science at Birla Institute of Technology (Deemed University), Mesra, Ranchi, India. His research interests are Grid Computing and Computer Networks.

Gadadhar Sahoo received MSc in Mathematics from Utkal University, Bhubaneswar in 1978 and PhD in Computational Mathematics from Indian Institute of Technology Kharagpur, India in 1983. He is now with the Department of Computer Science and Engineering, Birla Institute of Technology (Deemed University), Mesra, Ranchi, INDIA in the capacity of Professor and Head. His research interests are Parallel Computing, DNA Computing and Distributed Systems.