Integrating a Financial Option Based Model with GridSim for Pricing Grid Resources

David Allenotor #1, Ruppa K. Thulasiram *2

Department of Computer Science, University of Manitoba
Winnipeg, R3T 2N2, Canada
1 dallen@cs.umanitoba.ca

* Department of Computer Science, University of Manitoba
Winnipeg, R3T 2N2, Canada
2 tulsi@cs.umanitoba.ca

***Abstract*—Analysis of grid resources utilization from real grid trace data shows the feasibility of a financial option based model for pricing grid resources to attract more users for profitability for the grid provider while providing high Quality of Service (QoS) to users. However, in the absence of grid resource pricing benchmarks, we simulate grid resources usage in order to justify our pricing model using GridSim toolkit. In this work we integrate a financial option based pricing model with GridSim framework and use it as a grid simulation tool to price grid compute resources.**

I. Introduction/Background

Grid resources such as CPU cycles, memory, network bandwidths, throughput, computing power, disks, processor, and various measurements and instrumentation tools are non-storable compute commodities. Pricing these compute commodities is challenging because of the specific characteristics of the grid resources [1]: heterogeneity of resources (geographically dispersed ownership and time zones affects availability of resources) and volatility of resources since they exist as compute cycles). These two characteristics (among others) accounts for a high level of flexibilities (uncertainty and fluctuation) of resources. The trace data analysis in [2] shows that in the presence of such flexibilities, guaranteeing resource availability and profitability to users and grid services providers respectively is hard to determine using traditional methods such as Discounted Cash Flow (DCF) or Net Preset Value (NPV) [3].

Currently, there is no cost for using grid resources especially for research purposes. However, due to the large interest in grid for public computing, grid computing is experiencing a mushrooming of many service providers. Amazon, for example, introduced a Simple Storage Service S3 [4] system and the Elastic Compute Cloud (EC2) [5] for grid users. Amazon's S3 provides data-intensive, low cost, and highly available data storage system. EC2 provides on-demand computing resource as a virtual machine. One of the drawbacks of these services is that the resource prices are not flexible. Some other initiatives include AppNexus [6], GoGrid [7], Google App Engine [8], Microsoft Azure Services [9], and Joyent Accelerator [10]. Requirement for flexibility in grid resource usage is seen from the choices made available to users. Such choices include the decision to use the grid resources at a time in the present or at sometime in the future. It is hard to make decision using NPV or DCF without losing the realistic value of the decision [3]. To price the grid resources, (i) we treat them as grid compute commodities *gcc* and apply financial option theory to determine best exercise time for the use of the resources. (ii) we apply the real option concept to capture uncertainties that abound in making the decision to exercise the option (to use grid resources), and (iii) capture the uncertainties in the provision of QoS using fuzzy logic.

A financial option is defined (see, for example [11]) as the right to buy or to sell an underlying asset that is traded in an exchange for an agreed-on sum. The right to buy or sell an option may expire if the right is not exercised on or before a specific period and the option buyer loses the premium paid at the beginning of the contract. The exercise price *(strike price)* mentioned in an option contract is the stated price at which the asset can be bought or sold at a future date. A *call option* grants the holder the right (but not obligation) to buy the underlying asset at the specified strike price. On the other hand, a *put option* grants the holder the right to sell the underlying asset at the specified strike price. An *American option* can be exercised any time during the life of the option contract; a *European option* can only be exercised at expiry. Options are derivative securities because their value is a derived function from the price of some underlying asset on which the option is written. They are risky securities because the price of their underlying asset at any future time can not be predicted with certainty. This means the option holder has no assurance the option will bring profit before expiry.

Yeo and Buyya [12] provided a pricing function to improve utility for the producer on the basis that only jobs that have higher budgets will be accepted in the cluster workload. The pricing function also supports four essential requirements for pricing of utility-driven cluster resources: flexibility, fairness, dynamism, and adaptivity. In this article, the assumption that users will provide the exact computational requirements (faithfully) is only a hypothetical case. However, it is hard to capture users' honest computational requirements even in the presence of incentives [13]. In our work, our measure of flexibility for the use of grid resources is characterized by user opportunities from the decisions to utilize the resources. Such flexibilities can be accurately captured using real options so that asset prices could be computed using financial options pricing techniques.

A real option provides a choice from a set of alternatives. To hold a real option means to have a certain possibility for a given time to either choose for, or against investment decision. In [14], we designed a financial option model and proposed a justification for price adjustments in using a global base prices and a control factor in p_f [1]. Further study [2] showed that without unnecessarily charging for grid resource usage, we can satisfy both the user in terms of the Quality of Service (QoS) and the providers' incentives (profitability).

II. Integration Architecture GridSim

In [1], we designed a pricing architecture – abstract representation that comprises of grid services layer and the price optimization layer. In this architecture, the grid services layer houses the middleware while our pricing model integrates on the top layer of the middleware. In the current study, we integrate the top layer of our pricing model (price and usage optimization level) onto the top layer of GridSim [15].

Figure 1 shows the six layered architecture of the GridSim. It is a toolkit that administrates time-variable resource assignments. The first layer (at the bottom of Figure 1) consists of Java Virtual Machine (JVM). The JVM manages events and components interaction in the GridSim. The second layer consists of the infrastructure components such as network and resource hardware. This layer also enables the design and integration of user interfaces. The third and fourth layers are responsible for the simulation and modeling of computational grid entities. Simulation of the grid resource brokers takes place in the fifth layer. The top layer consists of the grid scenario, user requirements, I/O interface, and application configuration. In Figure 2, we show our pricing architecture (which we integrate into the GridSim toolkit in Figure 3) to price the grid resources using financial option based model [1]. The integration of our financial option-based pricing model to the GridSim toolkit is at the top levels of the GridSim toolkit architecture.

We develop user application that runs trinomial lattice (to compute option price) and optimize resources usage (reverse Dutch auction) and deployed to the user code layer of the GridSim toolkit. A classical Dutch auction begins with a high initial price which is constantly reduced until all items are sold. The price reduction happens through incremental discounts. In our approach using the inverse Dutch auction, the time between successive bids is similar to the time steps of the trinomial lattice. In the inverse auction – grid resource users begins with an initial willingness to use the grid resources (that is, to exercise the option). This means that the users are willing to pay a regular price without any discount (from the view point of the grid resource provider, this amount is a high price) and the provider continues to increase the willingness to reach the capacity of the resources by increasing the discounts to the users. If the grid resource capacity is less than the known limit we note the reserve price and then continue to offer more resources.

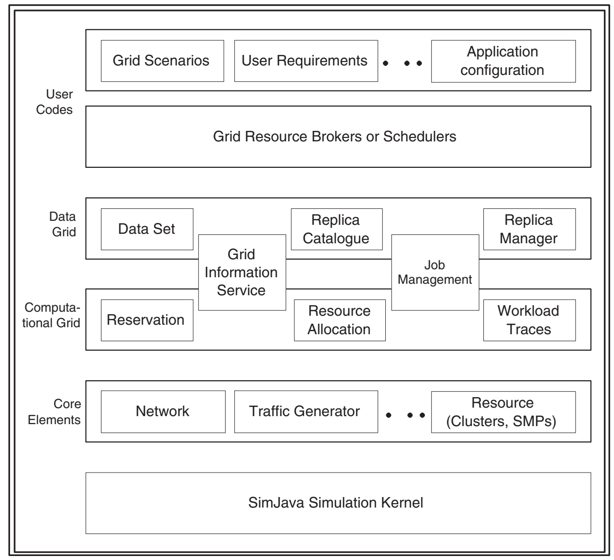


Fig. 1. GridSim Toolkit Layered Architecture

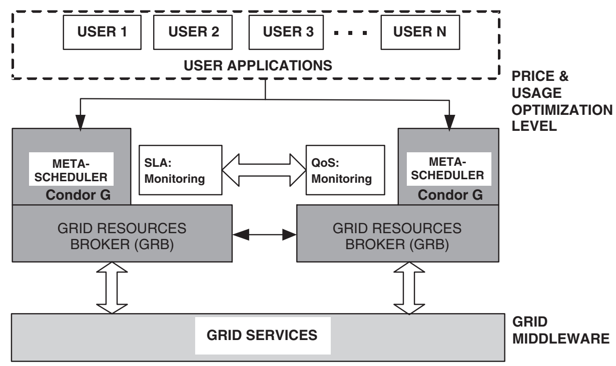


Fig. 2. Pricing Architecture [1]

III. Pricing Grid Resources in GridSim

The resultant architecture is given in Figure 3.

A. Algorithm

1. Begin: GridSim;
2. Begin: Create grid scenario;
/* Create the environment scenario
and initialize the GridSim Toolkit*/
3. Start: for each grid resource do; /*R_i*/
4. Create new processing elements; /*PE*/
5. Create new machines;/*M_i*/
6. Create new resources R_i; /*where R_i have one or more M_i that also have one or more PEs */
7. End: Create grid scenario
8. Begin: Create users' scenario

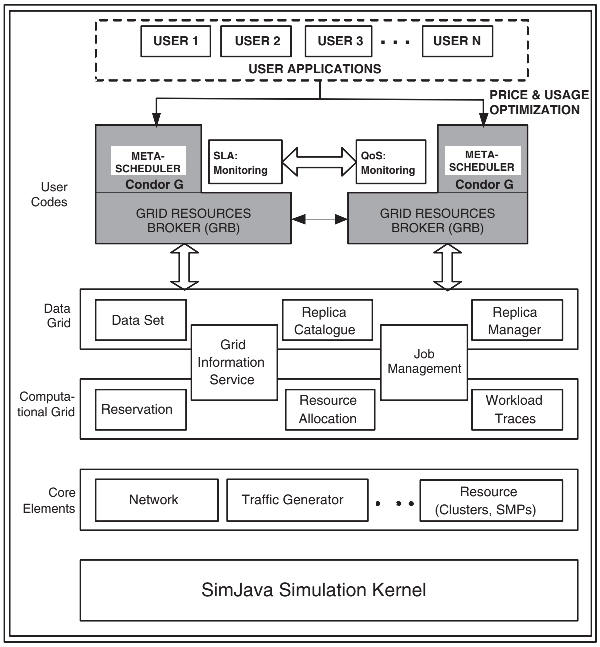


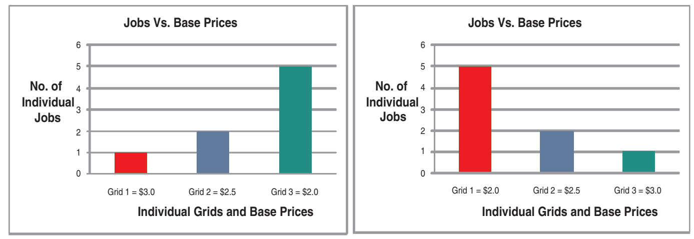
Fig. 3. Integrated Architecture

```
      for each user do
        create a grid task;
  End: users' scenario;
Begin: bid and trinomial;
   for each user do
    for grid resource do
        resource bid and utilization;
          Apply trinomial;
            compute option value;
    End: resource use and trinomial;
  Start GridSim simulation;
   Obtain simulation data;
  End GridSim simulation;
End:GridSim;
```

B. Results and Discussions

The algorithm in Section III-A shows the integration procedure in our simulation in Figure 3. The simulation starts with GridSim initialization and the creation of GridSim resources with different configurations. This is followed by the creation of GridSim users. Each user has different requirements from other users. The modeling of the GridSim resource starts with the CPUs (also called Processing Elements (PEs)). Each PE is created with several Million Instructions Per Second (MIPS) and the PEs combine to form one machine while several machines forms one single CPU node in the GridSim simulation. The second phase of the simulation setup is the specification for the trinomial lattice for option pricing. We configure the parameters as follows: Asset Price S, Strike Price K, Time to Maturity T, Interest Rate r, Number of Steps N, Interval time between 2 steps Δt, and Volatility σ.

We show preliminary results for 3 grids, G_1, G_2, and G_3; 3 users, user$_1$, user$_2$, and user$_3$, and 15 grid resources. We let the users provide specifications for their jobs and apply the inverse Dutch auction to allocate the jobs based the users' needs. Figure 4 shows job distribution in various grids with stated base prices. Figure 5 shows the computed option values for



(a) (b)

Fig. 4. User Jobs Distribution Based on Inverse Dutch Auction

Grid3 in Figure 4 (a) at various time steps. For example a time step of 16 in Figure 5 indicates a 3 month period a user may choose to exercise the option (utilize the grid resource). Each time step is a complete simulation carried out using similar base price of \$2.00 in Grid3. As the computed option values remains lower compared to the specified base prices, early exercise is profitable. Figure 6 shows computed option values for Grid1 in Figure 4 (a) at various time steps. In Figure 6, option values occurs at a relatively lower option value than the specified base prices. A user may decide to wait longer before exercising. Exercising late does not lead to profitability for the user because option values may become higher. For example, at higher time step (24 and 32 in Figure 5), the results of our trinomial option value computation converges and hence it does not profit waiting beyond these time steps. The jobs that ran in the two grids in Figure 4 despite a higher base prices could be due to the user requirements (time constrained, or deadline constrained job requirements).

From these option values, we can extract the underlying asset price, with these asset prices in conjunction with the control factor p_f used in the computations, we can also determine the QoS for the user and the level of cost recovering for the service provider. Yeo and Buyya [12] quantify QoS and profit level for the user and producer respectively. Their motive is to achieve a higher profitability for the service provider. In our work we focus on developing a model to bring an equilibrium between user satisfaction and cost recovering on the infrastructure expenses incurred by the providers. Hence, there is no direct comparison of our result with [12].

IV. Conclusion

The work presented puts forward an important idea: feasibility of a financial option based model for pricing grid resources with emphasis to attract more users for cost recovering on grid infrastructure for the grid provider while we uncompromis QoS for the users. We also present and justify the integration of our pricing architecture with GridSim which we used in our simulation.

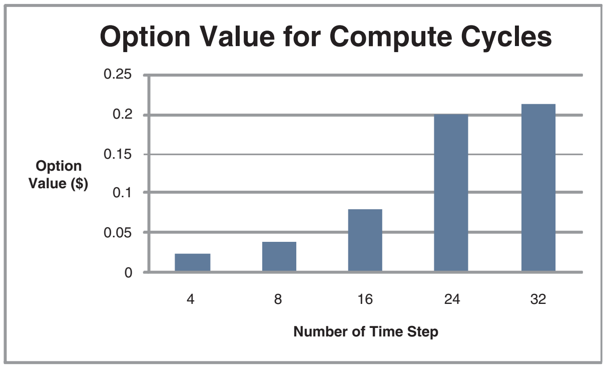


Fig. 5. Option Value at Base Price of $2.00. in Grid3

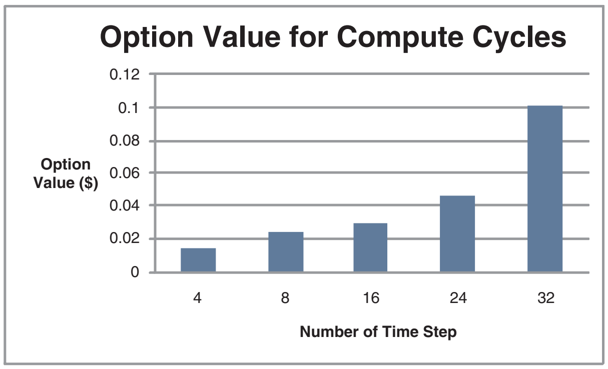


Fig. 6. Option Value at Base Price of $2.00 in G1

We conclude this work by identifying some of the important issues to be addressed in extending this work. We base our current suggestions on our current pricing architecture and the need to address the general problem of pricing grid resources using a financial option-based model. The following are extensions of the current work. More rigorous simulation is required (currently on-going) to determine the relationship between time of exercise and the cost of grid resource, job completion time, average resources processing times, average resource cost with relation to users budget in terms of cost and completion times.

Acknowledgement

We acknowledge the project team members of the Grid Computing and Distributed Systems (GRIDS) Laboratory, University of Melbourne for using their GridSim toolkit. The second author acknowledges gratefully the University Research Grant Program (URGP) of University of Manitoba and Natural Sciences and Engineering Research Council (NSERC) of Canada for the partial financial support.

References

[1] D. Allenotor and R. Thulasiram, "Grid resources pricing: A novel financial option-based quality of service-profit quasi-static equilibrium model," in *Proc. (CD-ROM) 9th IEEE/ACM IntI Conf. on Grid Computing (Grid 2008). September 29-October 01, Tsukuba, Japan.*, September 2008.

[2] D. Allenotor, R. Thulasiram, and P. Thulasiraman, "A financial option based grid resources pricing model: Towards an equilibrium between quality of service for users and profitability for service providers," in *4th IEEE Intl. Conf. on Grid and Pervasive Computing (GPC 2009). May 04-08 Geneva.*, May 2009.

[3] D. M. Chance and P. P. Peterson, *Real Options and Investment Valuation*. The Research Foundation of AMIR, 1st. Ed., 2002.

[4] M. Palankar, A. Onibokun, A. Iamnitchi, and M. Ripeanu, "Amazon s3 for science grids: a viable solution?" in *Poster: 4th USENIX Symposium on Networked Systems Design and Implementation (NSDI'07)*, April 2007.

[5] Amazone. (2009, March) Amazon Elastic Compute Cloud (Amazon EC2). [Online]. Available: http://aws.amazon.com/ec2/

[6] AppNexus. (2009, July) AppNexus. [Online]. Available: http://www.appneus.com/

[7] GoGrid. (2009, July) Cloud Hosting: Instant Windows and Linux Cloud Servers. [Online]. Available: http://www.gogrid.com/

[8] Google. (2009, July) App Engine. [Online]. Available: http://code.google.com/intl/de-DE/appengine/

[9] M. Corporation. (2009, July) Azure Services Platform. [Online]. Available: http://code.google.com/intl/de-DE/appengine/

[10] Joyent. (2009, July) Accelerator. [Online]. Available: http://joyent.com/accelerator/

[11] J. Hull, *Options, Futures, and Other Derivatives*. Prentice Hall, 6th. Ed., 2006.

[12] C. S. Yeo and R. Buyya, "Pricing for utility-driven resource management and allocation in clusters," *International Journal of High Performance Computing Applications*, vol. 21, no. 4, pp. 405–418, 2007.

[13] A. Mutz, R. Wolski, and J. Brevik, "Eliciting Honest Value Information in a Batch-Queue Environment," in *The 8th IEEE/ACM IntI Conf. on Grid Computing. September 1921, Austin, Texas, USA.*, September 2007.

[14] D. Allenotor and R. K. Thulasiram, "G-FRoM: Grid Resources Pricing A Fuzzy Real Option Model," in *Proc. (CD-ROM) 3rd IEEE Int' Conf. on e-Science and Grid Computing (eScience 2007) Bangalore, India, December 10-13*, 2007.

[15] A. Sulistio, U. Cibej, S. Venugopal, B. Robic, and R. Buyya, "A toolkit for modelling and simulating data grids: an extension to gridsim," *Concurrency and Computation: Practice and Experience (CCPE)*, vol. 20, no. 13, pp. 1591–1609, 2008.