International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622
NATIONAL CONFERENCE on Developments, Advances & Trends in Engineering Sciences
(NCDATES- 09th & 10th January 2015)

RESEARCH ARTICLE OPEN ACCESS

High Performance Resource Allocation Strategies for Computational Economies

Mr.A.Arun Kumar, Mr.E.SureshBabu
M.Tech Student, CMR Engineering College, Hyderabad Telangana, India
Asst.Professor,Dept of CSE,CMR Engineering College, Hyderabad, Telangana, India

Abstract
Utility computing models have long been the focus of academic research, and with the recent success of commercial cloud providers, computation and storage is finally being realized as the fifth utility. Computational economies are often proposed as an efficient means of resource allocation, however adoption has been limited due to a lack of performance and high overheads. In this paper, we address the performance limitations of existing economic allocation models by defining strategies to reduce the failure and reallocation rate, increase occupancy and thereby increase the obtainable utilization of the system. The high-performance resource utilization strategies presented can be used by market participants without requiring dramatic changes to the allocation protocol. The strategies considered include overbooking, advanced reservation, just-in-time bidding, and using substitute providers for service delivery. The proposed strategies have been implemented in a distributed met scheduler and evaluated with respect to Grid and cloud deployments. Several diverse synthetic workloads have been used to quantity both the performance benefits and economic.

Existing System

A global computation market could be realized by a high-performance federated architecture that spans both Grid and cloud computing providers, this type of architecture necessitates the use of economic aware allocation mechanisms driven by the underlying allocation requirements of cloud providers. Computational economies have long been touted as a means of allocating resources in both centralized and decentralized computing systems. Proponents of computational economies generally cite allocation efficiency, scalability, clear incentives, and well-understood mechanisms as advantages. However, adoption of economies in production systems has been limited due to criticisms relating to, among other things, poor performance, high latency, and high overheads. Moreover, there is an opportunity cost to reserving resources during a negotiation, as they will not be available for other negotiations that begin during the interval of the first negotiation. This type of scenario is clearly evident in auction or tender markets, however it can also be seen in any negotiation in which parties are competing against one another for the goods on offer. In any case, this wasteful negotiation process is expensive in both time and cost and therefore reduces the overall utilization of the system.

Disadvantages

- These strategies can be employed either through allocation protocols and/or by participants, to increase resource occupancy and therefore optimize overall utilization.
- The application of two general principles to largely address these inefficiencies: first, avoid commitment of resources, and second, avoid repeating negotiation and allocation processes. We have distilled these principles into five high-performance resource utilization strategies, namely: overbooking, advanced reservation, just-in-time (JIT) bidding, progressive contracts, and using substitute providers to compensate for encouraging oversubscription.

Proposed System

The earliest published computational market was the futures market that enabled users to bid for compute time on a shared departmental machine. Over time these market based architectures have grown from distributed computational economies, such as Spawn and Enterprise to modern brokers, met schedulers and distributed architectures such as Nimrod/G DRIVE and SORMA. Looking forward, there is great research interest in the creation of federated computing platforms encapsulating different computation providers. DRIVE, the system used for the experimental work in this paper, is one example of a a federated met scheduler and is designed around the idea of “infrastructure free” secure cooperative markets. Another prominent example is InterCloud which features a generic market model to match requests with providers using different negotiation protocols (including auctions), in which context, our strategies could largely be applied. Another alternative approach is spot pricing while this approach is in some ways similar to an auction (users set a max price), the fundamentals of operation are sufficiently different that a different set of strategies would be needed.

Advantages

- The previously used in computational domains as a way to increase utilization and profit. In overbooking is used to some extent to compensate for "no shows" and poorly estimated task duration.
- The first projects to define basic advanced reservation architecture to support QoS reservations over heterogeneous resources.
- The additional flexibility specified by some consumers, additionally these architectures have realized different reservation aware scheduling algorithms

Modules Description

Opportunities and High Utilization Strategies

The life cycle of economic negotiation presents a number of opportunities to implement utilization improving policies and strategies before, during, and after negotiation. In a traditional auction, providers auction resources by soliciting consumer's bids, at the conclusion of the auction an agreement is established to provide resources for the winning price, when the agreement expires the resources are returned.

Flexible Advanced Reservations

The task of planning job requirements is becoming more complex, requiring fine grained coordination of interdependent jobs in order to achieve larger goals. Often tasks require particular resources to be available at certain times in order to run efficiently. For example, a task may require temporary data storage while executing and more permanent storage after completion. Tasks may also require coordinated execution due to dependencies between one another.

Post auction

Auction latency may restrict providers participating in future negotiations due to a lack of knowledge of the outcome of ongoing or previous negotiations. There are two general approaches to mitigate this issue, that is, providers can reserve resources for the duration of the negotiation immediately, or they can wait for the result of the allocation before reservation.

Virtual Organization

Allocation in DRIVE is abstracted through an economic market which allows any economic protocol to be used. DRIVE features a novel "co-op" architecture, in which core met scheduling services are hosted on participating resource providers as a condition of joining the Virtual Organization.

Penalty Functions

The following penalty functions into two distinct penalty types: Constant penalties are fixed penalties that are statically defined irrespective of any other factors, whereas Dynamic penalties are based on a no static variable designed to reflect the value of a violation.

Overbooking Strategy

The high dynamic workload and the batch model, the peak system utilization approaches the maximum available capacity of the testbed when providers can bid beyond capacity. The average utilization and percentage of tasks allocated for all workloads is more than double that of the guaranteed strategy which highlights the value of overbooking. The allocation improvement exhibited in the batch workload represents the single biggest gain of any strategy and results in near optimal allocation and utilization.

CONCLUSIONS

The utility model employed by commercial cloud providers has remotivated the need for efficient and responsive economic resource allocation in high-performance computing environments. While economic resource allocation provides a well studied and efficient means of scalable decentralized allocation it has been stereotyped as a lowperformance solution due to the resource commitment overhead and latency in the allocation process. The high utilization strategies proposed in this paper are designed to minimize the impact of these factors to increase occupancy and improve system utilization. The high utilization strategies have each been implemented in the DRIVE metascheduler and evaluated using a series of batch and interactive workloads designed to model different scenarios, including multiple high throughput, short job duration workloads in which auction mechanisms typically perform poorly. The individual strategies, and the combination of the different strategies, was shown to dramatically improve occupancy and utilization in a highperformance situation. The increase in allocation rate was shown to be up to 286 percent for the dynamic workloads and 109 percent for the batch model.
In addition to occupancy and utilization improvements these strategies also provide advantages under differing

economic conditions. For example, the use of substitute providers was shown to be more price agnostic than other strategies due to the decreased allocation rate when a linear bidding strategy is used. Provider revenue also increased with the use of the proposed strategies, in part due to the increased allocation rate obtained. Finally, the effect of enaltiesontotal revenuewasshownto be heavily dependent on the penalty function used. The bid difference penalty, which represents the impact of the contract breach, resulted in only a small loss of total revenue across all providers. These results highlight that while these strategies can dramatically improve allocation performance, participants must fully consider the negative effects of the strategy used and associated penalty functions in order to optimize revenue.

REFERENCES

[1] I.E. Sutherland, “A Futures Market in Computer Time,” Comm. ACM, vol. 11, no. 6, pp. 449-451, 1968.

[2] K. Chard and K. Bubendorfer, “Using Secure Auctions to Build A Distributed Meta-Scheduler for the Grid,” Market Oriented Grid and Utility Computing, series Wiley Series on Parallel and Distributed Computing, R. Buyya and K. Bubendorfer, eds., pp. 569-588, Wiley, 2009.

[3] K. Chard, K. Bubendorfer, and P. Komisarczuk, “High Occupancy Resource Allocation for Grid and Cloud Systems, a Study With Drive,” Proc. 19th ACM Int’l Symp. High Performance Distributed Computing (HPDC ’10). pp. 73-84, 2010,

[4] C.A. Waldspurger, T. Hogg, B.A. Huberman, J.O. Kephart, and W.S. Stornetta, “Spawn: A Distributed Computational Economy,” IEEE Trans. Software Eng., vol. 18, no. 2, pp. 103-117, Feb. 1992.

[5] T.W. Malone, R.E. Fikes, K.R. Grant, and M.T. Howard, “Enterprise: A Market-Like Task Scheduler for Distributed Computing Environments,” The Ecology of Computation, pp. 177- 205, Elsevier Science Publishers (North-Holland), 1988.

[6] R. Buyya, D. Abramson, and J. Giddy, “Nimrod/g: An Architecture for a Resource Management and Scheduling System in a Global Computational Grid,” Proc. Fourth Int’l Conf. High Performance Computing in Asia-Pacific Region (HPC Asia ’00), pp. 283-289, 2000.