To Bid or Not To Bid: A Hybrid Market-Based Resource Allocation Framework

Elizeu Santos-Neto, Kate Keahey

University of Chicago & Argonne National Labs

elizeu@cs.uchicago.edu,keahey@mcs.anl.gov

Abstract

In this work we present a market based resource allocation framework which combines the advantages of leasing reservation and auction proportional share mechanisms. The rationale behind the hybrid allocation policy is to have an efficient and flexible resource sharing environment that fulfill the conflicting user/provider requirements by exploiting the heterogeneity of client demand. Our solution builds on current advances of Virtualization and Grid technologies. The main contributions are: (I) A market-based framework that combines advantages of leasing and auction resource allocation models; (II) An analysis of a hybrid market based resource allocation policy, where users can choose between paying more to have a low-risk resource reservation or paying less and controlling resource shares via bids.

1. Target Use Cases: *Low Latency* vs. *Allocation Stability*

- Use Case #1 - Educational environments
  - Focus on resource *allocation stability.*
  - Allocation Stability = less variation as possible in an allocated resource share.
  - Eg. Nodes for a class: shares of 70% in 5 nodes (cpu, mem, disk) from 1PM to 2PM.
- Use Case #2 - Urgent computations
  - Focus on *low latency* in resource acquisition.
  - Low Latency = to have resources allocated as soon as possible.
  - Eg. Simulations of the impact of a hurricane needs to start executing no later than 40sec.

2. Why markets ?

- Resource allocation policies have to cope with scalability and complexity issues.
- Market models appear as a natural choice to differentiate client priorities.
- Aggregation of distributed resources demands self-organizing mechanisms.
- Examples of real markets for computational resources
  - Amazon EC2 & S3
  - Sun Grid
- These examples are commodity markets based on pay-per-use charging method

3. Current Approaches

- *Leased Reservation*
  - Guarantee resource allocation stability (eg. Faucets, Brokered Leasing)
- *Auction Proportional Share*
  - Reduce the latency in resource acquisition (eg. Tycoon).

4. More about our context

- Distributed Resource Allocation Issues
  - Multiple users and providers
  - Users want to reduce latency and maximize throughput (min cost)
  - Providers want to increase utilization (max profit)
- Policies - Market Models
  - Currency is a natural way to define priority among clients
  - Demands an infrastructure (i.e. banks, auditing mechanisms, etc)
  - It is arguable whether a real market is the final solution
- Mechanisms - Virtualization
  - Isolation among clients (security & performance)
  - Fine grain performance tuning
  - Allow conflicting configurations to coexist
  - Increases overhead

5. The Main Issue: Requirements are conflicting

- Current approaches do not address both at the same time
- *Leased Reservations* - Brokered Leasing [1]
  - Allocation stability with no guarantees about latency
- *Auction Proportional Share* - Tycoon [2]
  - Low latency with probabilistic guarantees about stability

6. What is the trade-off ?

- Allocation stability is exchanged for a higher latency tolerance.
- Low latency is exchanged for tolerance on variations on resource allocation.
- Pricing mechanism creates incentive for users to decide on which contract to use.

7. Our Approach

- To combine resource reservation and auctions
  - Use Case #1 prioritizes resource *allocation stability*
  - Use Case #2 prioritizes *low latency* in resource allocation.
- Hybrid Market Model
  - Policy = Leasing Reservations + Auction Proportional Share
  - Mechanism = Virtual Machines (eg. Virtual Workspaces [3])
  - Clients pay more for stability
  - System updates the price according to demand fluctuations.

8. Hybrid Market

- Leasing Reservation
  - Resource share may not be available at request time
  - Clients have the resource share allocated during the time requested
  - Clients are not preempted in this situation
- Auction Proportional Share
  - A resource share is immediately allocated
  - The client's bid determines her share
  - Clients may be outbid at anytime (this may cause preemption)

9. System Architecture

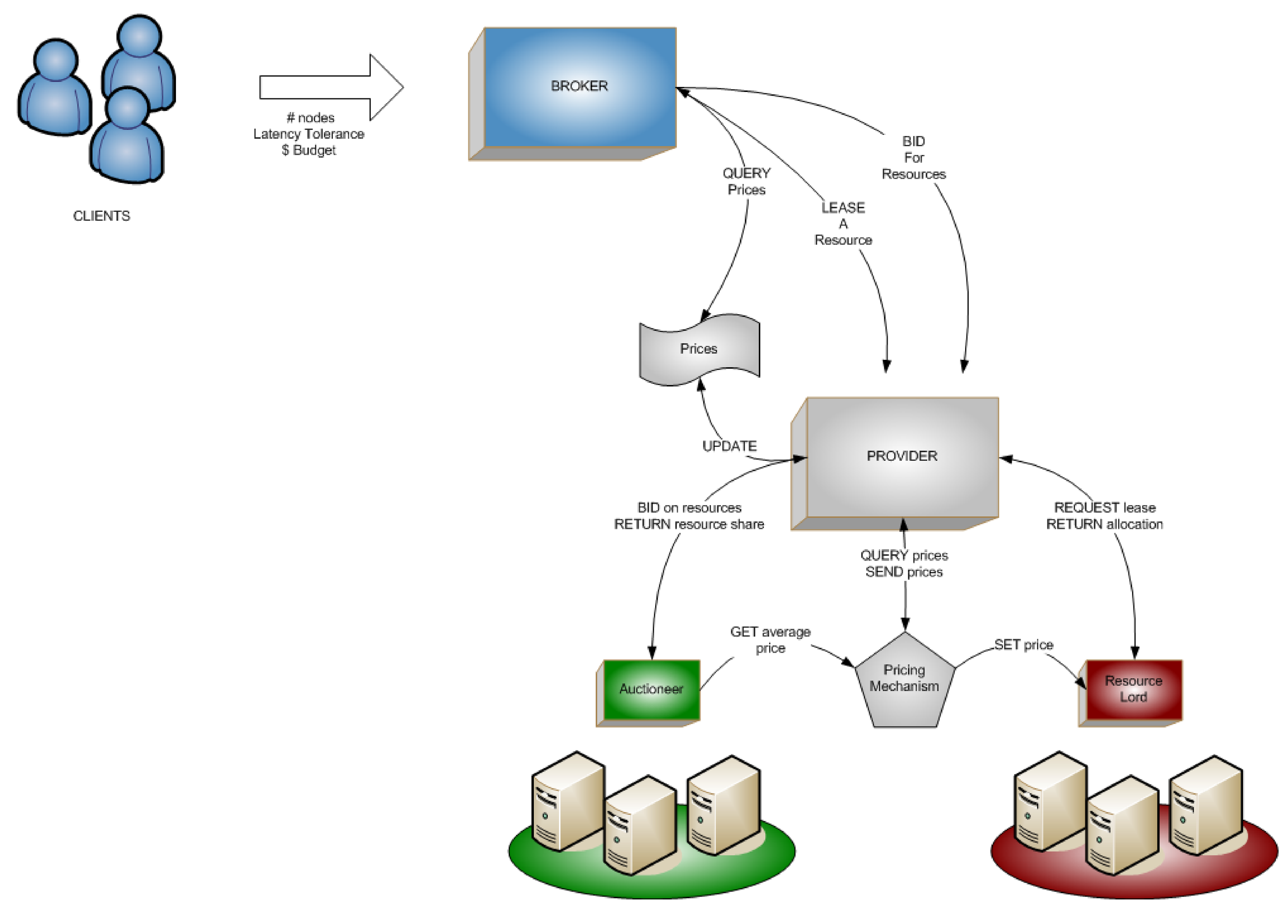

Figure 1: *System architecture overview*

9.1 Market Extensions

- Assumptions
  1. There exists a bank which regulates the funding rate of clients.
  2. Clients (or brokers) are in charge of analyzing the trade-off (leasing vs. auction).
  3. Leases and Bids have an expiration date.
  4. Resource stability is more valuable, since it is based on reservation.
- Resource Price under Auction (n is # bids placed on resource r):

$$price^r_{auction} = \sum_{j=1}^{n} bid^r_j \quad (1)$$

- Resource Price under Leasing

$$price^r_{lease} = c \times \frac{\sum_{j=1}^{m} price^j_{auction}}{m} \quad (2)$$

- Resource Share per client (under auction):

$$share^r_i(auction) = \frac{bid_i}{\sum_{j=1}^{n} bid_j} \quad (3)$$

- Resource Share per client (under leasing, resources are dedicated):

$$share^r_i(leasing) = 1 \quad (4)$$

- Leasing prices are updated according to the auction demand.

SC06, The International Conference for High Performance Computing, Networking, Storage, and Analysis, 11-17 November 2006, Tampa, FL, USA

- Prices and resource shares are iteratively updated:
  1. A bid arrives request arrives.
  2. A bid expires.

10. Simulations

- Each simulation consisted of 1000 steps.
- Number of resources under auction and leasing contracts is equal.
- Poisson distributions were used to simulate request rate and job sizes.
- Clients are funded with the same amount at each simulation step.
- Demand was varied by regulating the average request rate and average job size.
- Clients use a simple strategy:
  - they use their available balance to contract a number of nodes for a certain interval.

10.1 Fairness

- Fairness: a client must get what she pays for.
- Lease
  - Average share under leasing should be close to 1 (resource is dedicated).
  - Price should be always higher then Auction price.
- Auction
  - Average share is expected to decrease as demand increases.
  - Price should be always lower than Leasing price

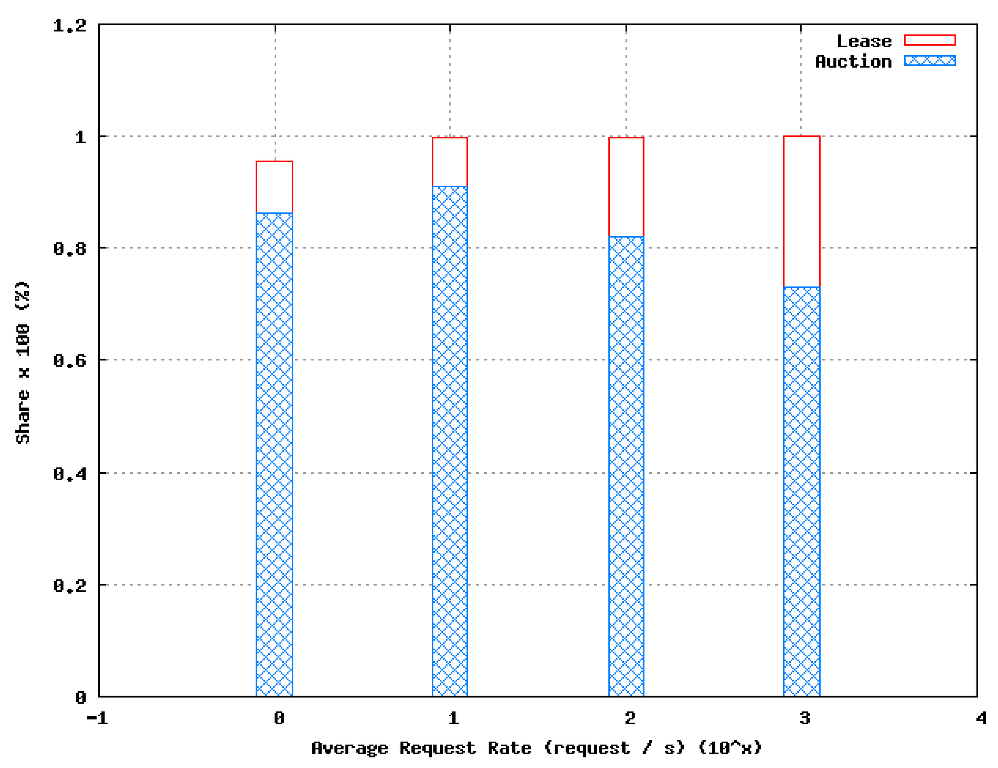

Figure 2: *Resource share as a function of request rate*

- As demand increases, Leasing resource shares do not vary, in contrast to Auction resource share.

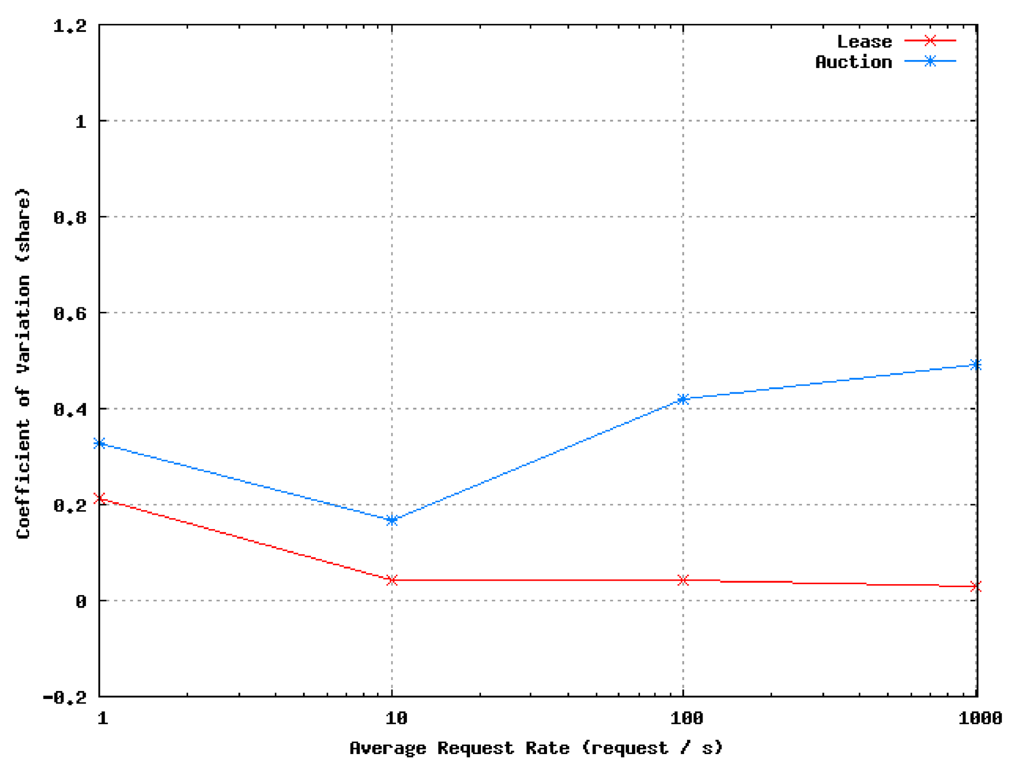

Figure 3: *Allocation Stability as a function of request rate*

- Allocation stability is guaranteed for leasing (over demand cases)

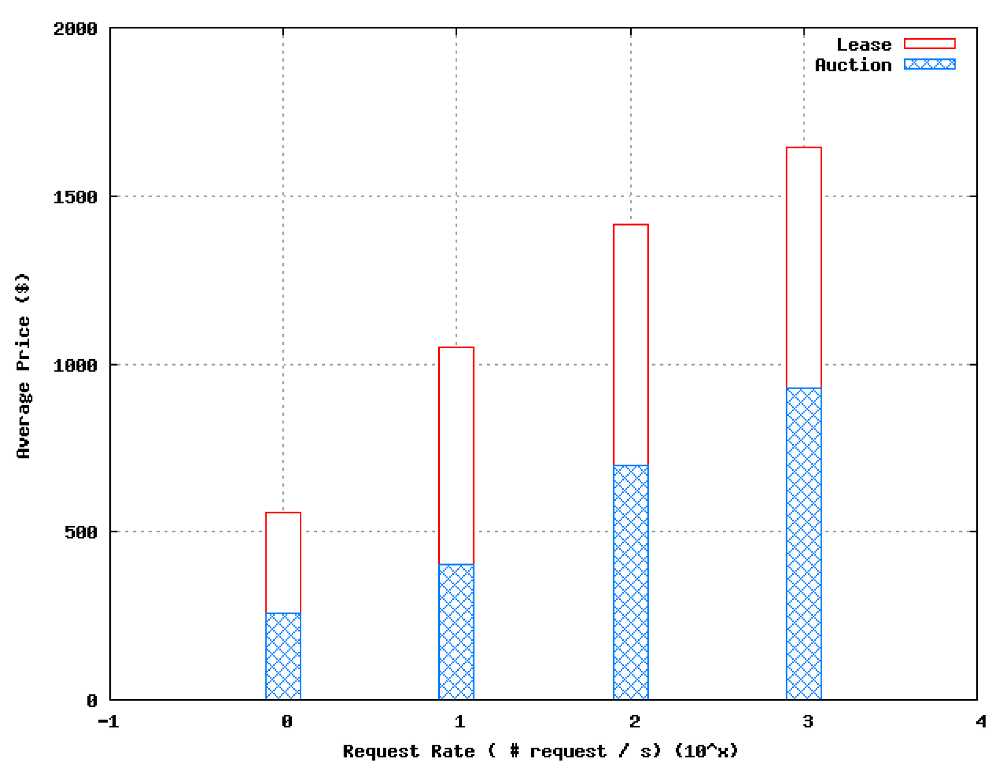

Figure 4: *Resource price as a function of request rate*

- Price increases with demand (as expected)
- Leasing prices are always higher than Auction (stability was defined as more valuable).

10.2 Market Price Adaptation

- In average, price tends to stabilize over time.

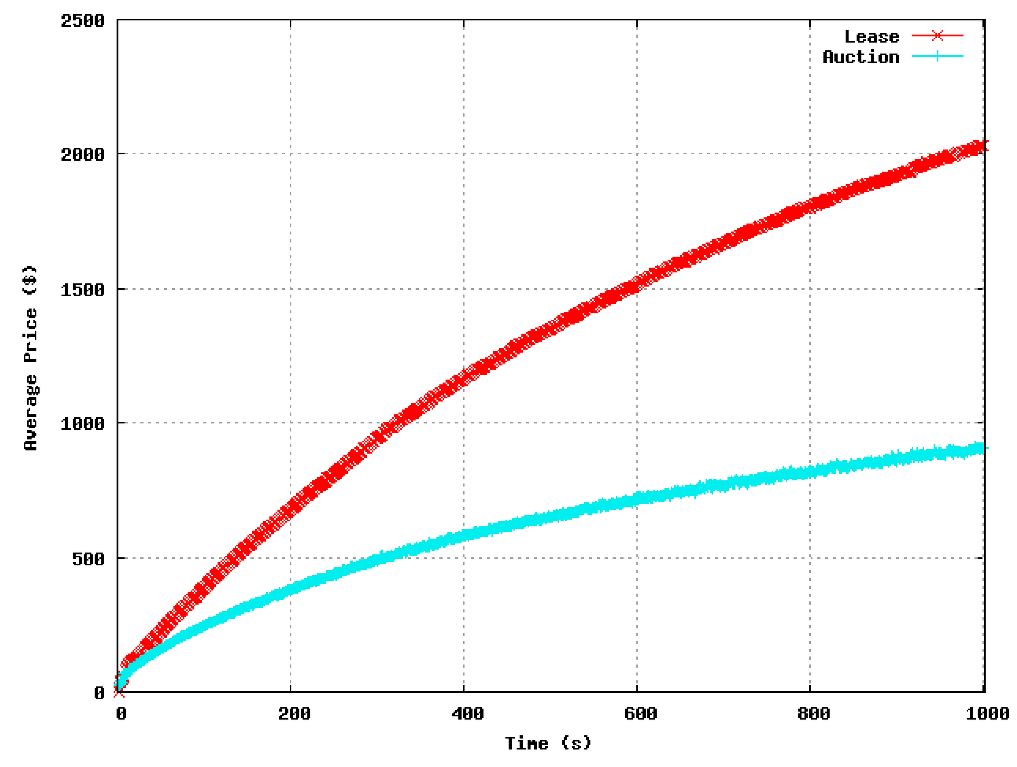

Figure 5: *Average leasing and auction prices over time*

- Higher the demand, higher the demand and longer it takes to stabilize.

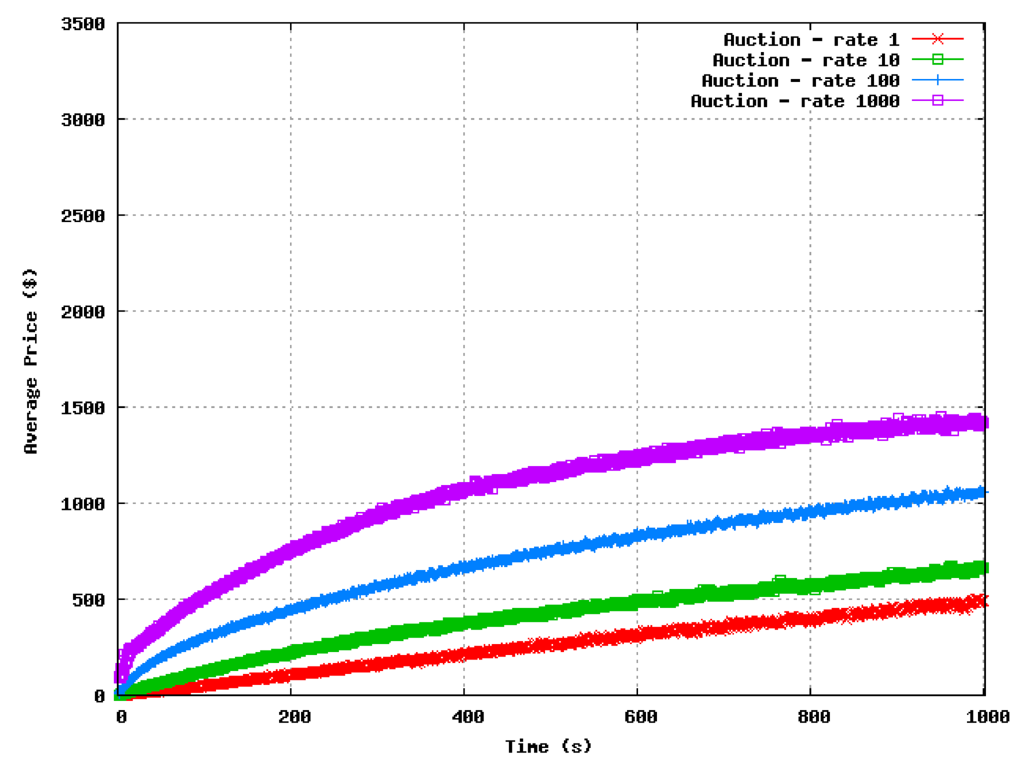

Figure 6: *Average auction price grouped by request rate*

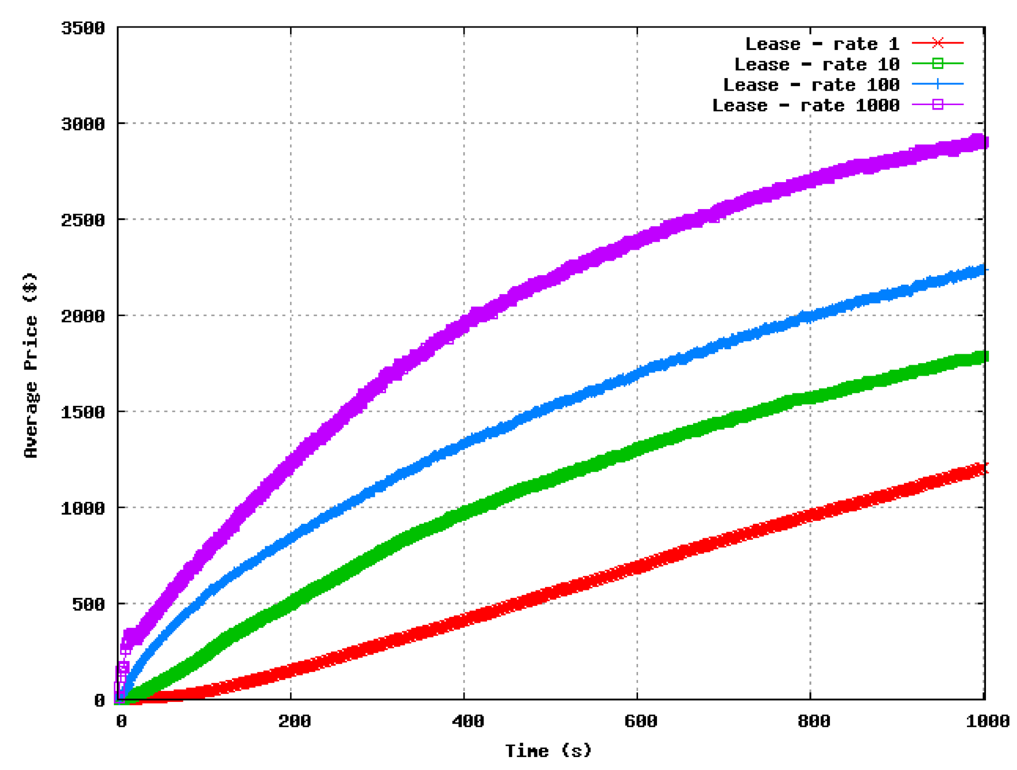

Figure 7: *Average leasing price grouped by request rate*

- Current funding scheme makes the price to inflate.
- An alternative for price adaptation is to consider demand/supply excesses.

11. Conclusions and Future Directions

- Our target use cases present conflicting requirements (stability & low latency)
- Combining Leasing Reservations and Auction Proportional Share seems to be a promising strategy
- A more comprehensive analysis is needed to refine our market model.
  - Number of requests of each class
  - Variation of the resource pool size on demand
  - Price equilibrium and client satisfaction would be interesting to compare among approaches
- Experiment with different funding schemes (eg. self recharging currency [4])
- Different client strategies (eg. price prediction mechanisms).

References

[1] D. Irwin, J. Chase, L. Grit, A. Yumerefendi, and D. Becker, "Sharing networked resources with brokered leases," in *Proceedings of USENIX Annual Technical Conference*, 2006.

[2] K. Lai, B. A. Huberman, and L. Fine, "Tycoon: A distributed market-based resource allocation system." Technical Report arXiv:cs.DC/0404013, April 2004.

[3] K. Keahey, I. Foster, T. Freeman, and X. Zhang, "Virtual workspaces: Achieving quality of service and quality of life in the grid," *Scientific Programming Journal (Special Issue: Dynamic Grids and Worldwide Computing)*, vol. 13, no. 4, pp. 265–76, 2005.

[4] D. Irwin, J. Chase, L. Grit, and A. Yumerefendi, "Self-recharging virtual currency," in *Proceedings of the 2005 ACM SIGCOMM Workshop on Economics of Peer-To-Peer Systems*, August 2005.