RESEARCH ARTICLE OPEN ACCESS

# Modified Truthful Greedy Mechanisms for Dynamic Virtual Machine Provisioning and Allocation in Clouds

[1]B.Sudha,

Assistant Professor,Voorhees College,Vellore.

----------------------------------------************************----------------------------

## Abstract:

In Cloud Computing models, Virtual Machines (VM) can be dynamically provisioned according to demand and released when not needed. Efficient Virtual Machine (VM) provisioning and allocation allows the cloud providers to effectively utilize their available resources and obtain higher profits. The existing static VM provisioning may not guarantee economically efficient allocation and thus cannot guarantee maximum revenue for cloud providers. A better solution would be to take into account the users' demand and dynamically provision VM instances. In the recent times, the cloud providers have introduced auction-based models for VM provisioning and allocation which allow users to submit bids for their requested VMs. MM Nejad, L. Mashayekhy and D Grosu have formulated a dynamic VM provisioning and allocation problem for the auction-based model as an integer program considering multiple types of resources. They have designed truthful greedy and optimal mechanisms for the problem such that the cloud provider provisions VMs based on the requests of the winning users and determines their payments. Since Virtual Machines are created by Hypervisors in the underlying physical machines and this would result in overhead and ultimately would impact the overall performance of the cloud. We propose that this overhead could be factored inthe design of Truthful Greedy mechanisms. It is hoped that our proposed idea can achieve better results in terms of revenue for the cloud provider.

*Keywords*— **cloud computing; truthful mechanism; virtual machine provisioning; dynamic resource allocation; hypervisor; Cloud provider**

----------------------------------------************************----------------------------

## 1. Introduction

### 1.1 What Cloud providers offer

The number of enterprises and individuals that are outsourcing their workloads to cloud providers has increased rapidly in recent years. Cloud providers form a large pool of abstracted, virtualized, and dynamically scalable resources allocated to users based on a pay-as-you-go model. These resources are provided as three different types of services: infrastructure as a service (IaaS), platform as a service (PaaS), and software as a service (SaaS). IaaS provides CPUs, storage, networks and other low level resources, PaaS provides programming interfaces, and SaaS provides already created applications. In this paper, the focus is on IaaS where cloud providers offer different types of resources in the form of VM instances. IaaS providers such as Microsoft Azure and Amazon EC2 offer four types of VM instances: small (S), medium (M), large (L), and extra-large (XL).

| VM Type | CPU | RAM | DISK Space |
|---|---|---|---|
| Small | 1 | 1 GB | 50 GB |

| | | | |
|---|---|---|---|
| Medium | 2 | 2 GB | 100 GB |
| Large | 4 | 4 GB | 200 GB |
| Extra Large | 8 | 8 GB | 400 GB |

Figure 1: Typical virtual machine types offered by cloud providers

## 1.2 The challenge for cloud providers:

Cloud providers face many decision problems when offering IaaS to their customers. One of the major decision problems is how to provision and allocate VM instances. Cloud providers provision their resources either statically or dynamically, and then allocate them in the form of VM instances to their customers. In the case of static provisioning, the cloud provider pre-provisions a set of VM instances without considering the current demand from the users, while in the case of dynamic provisioning, the cloud provider provisions the resources by taking into account the current users' demand. Due to the variable load demand, dynamic provisioning leads to a more efficient resource utilization and ultimately to higher revenues for the cloud provider.

The aim is to facilitate dynamic provisioning of multiple types of resources based on the users' requests. To sell the VM instances to users, cloud providers can employ fixed-price and auction-based models. In the fixed-price model, the price of each type of VM instance is fixed and pre-determined by the cloud provider, while in the auction-based model, each user bids for a subset of available VM instances (bundle) and an auction mechanism decides the price and the allocation. In this study, we consider the design of mechanisms for auction-based settings. In the auction-based models, users can obtain their requested resources at lower prices than in the case of the fixed-price models. Also, the cloud providers can increase their profit by allowing users to bid on unutilized capacity.

## 1.3 Our Contribution

Virtual machines are created by Hypervisors in the underlying physical machines. It is obvious that there is some overhead involved in when Virtual machines are created by Hypervisors. We propose to include this overhead factor in the work done by MM Nejad, L. Mashayekhy and D Grosu in their paper [1]. By factoring in the cost associated with the design of Truthful Greedy mechanism for Dynamic Virtual Machine provisioning and allocation, the cloud provider can utilize their resources optimally with increased overall cloud performance. Thus it would not only result in better clientsatisfaction but also may result in higher profit in the end. We show that one way to factor in the overhead is to adjust the bidding amount based on the requested count of VMs irrespective of the types in the bundle. Our main goal is to extend the design of Truthful Greedy Mechanism [1] to factor in the underlying virtualization cost; however how to factor it in in the mechanism is open for further study and experiment

## 1.4 Related Work:

MM Nejad, L. Mashayekhy and D Grosu in their paper [1] dealt with this problem in detail. Their mechanism is based on users requesting bundles of VM instances of different types. They consider a set of users and a set of items (VM instances), where each user bids for a subset of items (bundle). Since several VM instances of the same type are available to users, the problem can be viewed as a multi-unit combinatorial auction. Each user has a private value (private type) for her requested bundle. In their model, the users are single minded, that means each user is either assigned her entire requested bundle of VM instances and she pays for it, or she does not obtain any bundle and pays nothing. The users are also selfish in the sense that they want to maximize their own utility. It may be beneficial for them to manipulate the system by declaring a false type (i.e., different bundles or bids from their actual request). That is a user may report lower valuation to pay less or a higher valuation to enhance the chances of winning the bid. One of the key properties of a provisioning and allocation mechanism is to give incentives to users so that they reveal their true valuations for the bundles. In general, greedy algorithms do not necessarily satisfy the properties required to achieve truthfulness (also called incentive-compatibility or strategy-proof-ness) and they need to be specifically designed to satisfy those properties. They designed truthful greedy mechanisms that solve the VM provisioning and allocation problem in the presence of multiple types of resources (e.g., cores, memory, storage, etc.). The mechanisms allocate resources to the users such that the social welfare (i.e., the sum of users' valuations for the requested bundles of VMs) is maximized. Their proposed mechanisms allow dynamic provisioning of VMs, and do not require pre-provisioning the VMs. As a result, cloud providers can fulfill dynamic market demands efficiently. A key property of their proposed mechanisms is the consideration of multiple types of resources when provisioning the VMs, which is the case in real cloud settings.

## 1.5 Simplified Strategy proof-ness or incentive compatible

A mechanism is truthful if truthful reporting is a dominant strategy for the users, that is, the users maximize their utilities by truthful reporting independently of what the other users are reporting. There are several papers dealing with Strategy proof-ness or incentive compatiblemechanisms.

Consider a cloud-client i in group $g_j$with valuation bid $v_i$. Let $b_{min}= \min\{b_k | k \in g_i\}$

When the client is in winning group, her/his utility $u_i$would be

a. $u_i=v_i-b_{min}$ if $b_i>b_{min}$
b. $u_i= 0$ if $b_i = b_{min}$
c.

The simplified form of it in the context of our proposed idea is given below.Let us assume that the cloud providers offer the following types VMs

| VM Type | CPU | RAM | DISK Space |
|---|---|---|---|
| Small | 1 | 1 GB | 50 GB |
| Medium | 2 | 2 GB | 100 GB |
| Large | 4 | 4 GB | 200 GB |
| Xtra Large | 8 | 8 GB | 400 GB |

Figure 2: An example for Virtual Machine types and associated resources.

The requests from clients could look like the following:

| Cloud Client | Requested Types | Bid amount | Resource need From Cloud provider's view |
|---|---|---|---|
| User1 | {1,2,1,0} | $20 K | 9 CPUs |
| User2 | {5,2,0,0} | $30 K | 9 CPUs |
| User3 | {9,0,0,0} | $40 K | 9 CPUs |
| User4 | {7,1,0,0} | $45 K | 9 CPUs |
| User5 | {1,0,0,1} | $50 K | 9 CPUs |
| | | | |
| User6 | {4,2,1,0} | $35 K | 12 CPUs |
| User7 | {8,2,0,0} | $40 K | 12 CPUs |
| User8 | {12,0,0,0} | $35 K | 12 CPUs |
| User9 | {10,1,0,0} | $65 K | 12 CPUs |
| User10 | {4,0,0,1} | $45 K | 12 CPUs |
| | | | |
| User11 | {7,2,1,0} | $25 K | 15 CPUs |
| User12 | {11,2,0,0} | $50 K | 15 CPUs |
| User13 | {15,0,0,0} | $70 K | 15 CPUs |
| User14 | {13,1,0,0} | $75 K | 15 CPUs |
| User15 | {7,0,0,1} | $85 K | 15 CPUs |

Figure 3: The bundles with bid amount as well as normalized resource request from cloud provider's point of view

Note that we assumed the requests from the cloud clients contain only CPUs. Other resources types are ignored for simplicity of explaining the truthful mechanism here. The bidding amount is the maximum that the user is willing to pay if he/she wins the bid. That is, from cloud client point of view, it is the maximum value he/she may derive if the bid is successful.

For a moment, let us consider Users 1 to 5. Though they request different types of VMs of various combinations, from cloud provider point of view, they are all requesting 9 CPUs. If we assume that the cloud provider has only 35 CPUs to offer at a particular point of time, there could be only three winning users (that is, users 3, 4 and 5) in this group of five members. Users 1 and 2 miss out. The payment can be based on the minimum bidding amount in within the group which, in this case, is $20 K.

This method is incentive-compatible because if a user is trying to lie about the true value that she/he may derive out the requested bundle if the bid is successful by bidding with very high value, there is possibility that his/her bid is least one and when the bid is successful, the he would end up paying the quoted bid amount as payment. It is also possible that the user may bid for very low value. In this case, there are very high chances that co-bidders quote higher value than his/her, the user may end up in losing out. So in the end, cloud provider would get $60 K (i.e. 20 K from each of winning users)

Let us assume that we have another set of users 6 to 10. The same way, within this group the winning users would be 9 and 10 and their payment for each would be $35. Thus cloud provider would get $70 K in total from the clients.

Now to maximize the profit, the cloud provider needs to consider all groups and only one group can be a winning group among them. The mechanism used by the cloud provider to allocate available resource would become complicated if there are hundreds of groups and with each group having many competing clients.

The problem would become much more complicated when there are multiple resource types to consider. The above example is only with CPU but there are other resource types such as RAM, DISK space, I/O Ports etc. Cloud provider need to check the requested resources against the available capacity at hand.

## 2 .Modified Greedy Mechanism

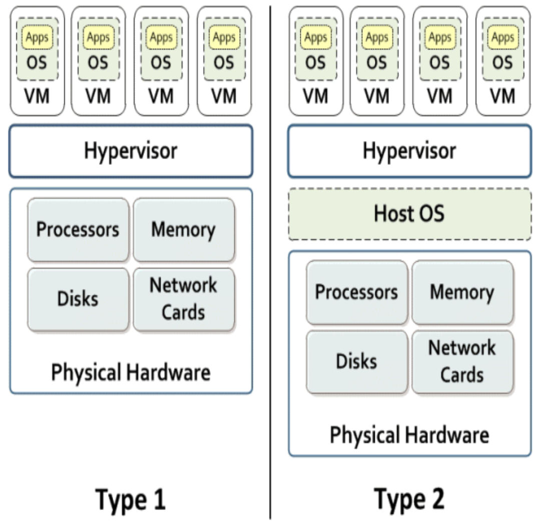


Figure 4: Hypervisors, an example from IBM website

Virtual machines are created by Hypervisors in the underlying physical machines. It is obvious that there is some overhead involved in when Virtual machines are created by Hypervisors. We propose to include this overhead factor in the work done by MM Nejad, L. Mashayekhy and D Grosu in their paper [1]. By factoring in the cost associated with the design of Truthful Greedy mechanism for Dynamic Virtual Machine provisioning and allocation, the cloud provider can utilize their resources optimally with increased overall cloud performance. Thus it would not only result in better client satisfaction but also may result in higher profit in the end. We show that one way to factor in the overhead is to adjust the bidding amount based on the requested count of VMs irrespective of the types in the bundle. Our maingoal is to extend the design of Truthful Greedy Mechanism [1] to factor in the underlying virtualization cost, however how to factor it in in the mechanism is open for further study and experiment.

Let us take previously stated example above. And let us assume that overhead cost for setting up each VM is $ 1K.

User 1 is requesting four VMs (1 small, 2 medium, 1 large). User 2 is requesting 7 VMs in total with 5 small and 2 medium types. However from cloud provider point of view, they both are requesting 9 CPUs in total.

| Cloud Clinet | Requested Types | Bid amount from client | Count of VMs | Adjusted Bid(=bid + overhead cost) | Resource need From Cloud provider's view |
|---|---|---|---|---|---|
| User1 | {1,2,1,0} | $20 K | 4 | $24 K | 9 CPUs |
| User2 | {5,2,0,0} | $30 K | 7 | $37 K | 9 CPUs |
| User3 | {9,0,0,0} | $40 K | 9 | $49 K | 9 CPUs |
| User4 | {7,1,0,0} | $45 K | 8 | $53 K | 9 CPUs |
| User5 | {1,0,0,1} | $50 K | 2 | $52 K | 9 CPUs |
| | | | | | |
| User6 | {4,2,1,0} | $35 K | 7 | $35 K | 12 CPUs |
| User7 | {8,2,0,0} | $40 K | 10 | $40 K | 12 CPUs |
| User8 | {12,0,0,0} | $35 K | 12 | $35 K | 12 CPUs |
| User9 | {10,1,0,0} | $65 K | 11 | $65 K | 12 CPUs |
| User10 | {4,0,0,1} | $45 K | 5 | $45 K | 12 CPUs |
| | | | | | |
| User11 | {7,2,1,0} | $25 K | 10 | $25 K | 15 CPUs |
| User12 | {11,2,0,0} | $50 K | 13 | $50 K | 15 CPUs |
| User13 | {15,0,0,0} | $70 K | 15 | $70 K | 15 CPUs |
| User14 | {13,1,0,0} | $75 K | 14 | $75 K | 15 CPUs |
| User15 | {7,0,0,1} | $85 K | 8 | $85 K | 15 CPUs |

Figure 4: Explains how the bidding amount can be factored in.

By adjusting the bid amount by overhead cost associated with each requested bundle is done in order to optimally allocate the resources at hand for cloud provider. With this adjustment, Truthful Greedy Mechanism would result in higher payment from the winning clients. It would also improve inoverall cloud performance thus would help in improved client satisfaction ultimately.

Note that $1 K per VM is only an assumption that we took to demonstrate our proposed mechanism. Also is the case with the adjustment of bidding amount. However it is open for the cloud provider to adopt any other suitable method that would effectively factor in the overhead associated with setting up of Virtual Machines by Hypervisors.

The downside with this approach is the actual bidding amount from the client is not the value used for determining the eventual payment from the client but the adjusted bidding amount. However it can be made transparent to the bidding clients that bidding amount would eventually be adjusted by a factor based on VM types in the bids.

Truthful Greedy Allocation algorithm and payment algorithm are explained in detail in [1].

## 3 Conclusion:

We addressed the overhead associated with setting up VMs in already proposed Truthful Greedy Mechanism for Dynamic VM Provisioning and Allocation in clouds by factoring it in the bidding amount from the clients.Our proposed Modified Truthful Greedy Mechanism would result in higher payment from the winning clients and thus this would result in higher profit for the cloud provider. It would also improve in overall cloud performance thus would help in improved client satisfaction ultimately from the cloud provider point of view. As a recommendation, Modified Truthful Greedy Mechanism proposed by us is the best choice since it would yield the highest revenue among the many proposed greedy mechnisms. We plan to implement a prototype allocation system in an experimental cloud computing system to further investigate the performance of our proposed mechanisms.

## 4 ACKNOWLEDGMENTS :

This paper is an extended version of [1] submitted on the category of IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, VOL. 26, NO. 2, FEBRUARY 2015

## 5 References:


[1] M.M. Nejad, L. Mashayekh and D. Grouu, "Truthful Greedy Mechanisms for Dynamic Virtual Machine Provisioning and Allocation in Clouds" IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, VOL. 26, NO. 2, FEBRUARY 2015

[2] F. Wu and N. Vaidya, "A Strategy-Proof Radio Spectrum Auction Mechanism in Non-cooperative

Wireless Networks" IEEE Transactions on , Volume 12 (5) – Mar 21, 2013

[3] Windows Azure. [Online]. Available: http://www.windowsazure.com/en-us/pricing/calculator/

[4] Amazon EC2 Instance Types. [Online]. Available: http://aws.amazon.com/ec2/instance-types/

[5] N. Jain, I. Menache, J. Naor, and J. Yaniv, "A truthful mechanism for value-based scheduling in cloud computing," Theory of Computing Systems, pp. 1–19, 2013.[Online]. Available: http://dx.doi.org/10.1007/s00224-013-9449-0

[6] Z.Kong,C.-Z.Xu,andM.Guo,"Mechanismdesignforstochastic virtual resource allocation in non-cooperative cloud systems," in Proc. 4th IEEE Intl. Conf. on Cloud Computing, 2011, pp. 614–621.

[7] Y. Wang, A. Nakao, and A. V. Vasilakos, "Heterogeneity playing key role: Modeling and analyzing the dynamics of incentive mechanisms in autonomous networks," ACM Transactions on Au¬tonomous and Adaptive Systems, vol. 7, no. 3, p. 31, 2012.

[8] D. Ardagna, B. Panicucci, and M. Passacantando, "Generalized Nash equilibria for the service provisioning problem in cloud systems," IEEE Transactions on Services Computing, vol. 6, no. 4, pp. 429–442, 2013.

[9] V.DiValerio,V.Cardellini,andF.LoPresti, "Optimalpricing and service provisioning strategies in cloud systems: a Stackelberg game approach," in Proc. of the 6th IEEE Intl. Conf. on Cloud Computing, 2013, pp. 115–122.

[10] X. Zhou, S. Gandhi, S. Suri, and H. Zheng, "ebay in the sky: strategy-proof wireless spectrum auctions," in Proc. of the 14th ACM Intl. Conf. on Mobile Comp. and Networking, 2008, pp. 2–13.

[11] X. Zhou and H. Zheng,"Trust: A general framework for truthful double spectrum auctions,"in Proc. ofIEEEINFOCOM2009,2009, pp. 999–1007.

[12] G. S. Kasbekar and S. Sarkar, "Spectrum auction framework for access allocation in cognitive radio networks," IEEE/ACM Transactions on Networking, vol. 18, no. 6, pp. 1841–1854, 2010.

[13] F. Wu and N. Vaidya, "A strategy-proof radio spectrum auction mechanism in noncooperative wireless networks," IEEE Transac¬tions on Mobile Computing, vol. 12, no. 5, pp. 885–894, 2013.

[14]J.Jia,Q.Zhang,Q.Zhang,andM.Liu, "Revenuegenerationfor truthful spectrum auction in dynamic spectrum access," in Proc. 10th ACM Intl. Symp. on Mobile Ad Hoc Networking and Computing, 2009, pp. 3–12.

[15] S. Zaman and D. Grosu, "Combinatorial auction-based dynamic vm provisioning and allocation in clouds," in Proc. 3rd IEEE Intl. Conf. on Cloud Comp. Tech. and Sci., 2011, pp. 107–114.

[16] ——, "Combinatorial auction-based allocation of virtual machine instances in clouds," J. of Parallel and Distributed Computing, vol. 73, no. 4, pp. 495–508, 2013.

[17] H. Kellerer, U. Pferschy, and D. Pisinger, Knapsack Problems.Springer, 2004.

[18] A. Mu'alem and N. Nisan,"Truthful approximation mechanisms for restricted combinatorial auctions," in Proc. 18th Nat. Conf. on Artificial Intelligence, 2002, pp. 379–384.

[19] D. Lehmann, L. O´callaghan, and Y. Shoham, "Truth revelation in approximately efficient combinatorial auctions," Journal of the ACM, vol. 49, no. 5, pp. 577–602, 2002.

[20]M.M.Nejad,L.Mashayekhy,andD.Grosu, "Afamilyof truthful greedy mechanisms for dynamic virtual machine provisioning and allocation in clouds," in Proc. of the 6th IEEE Intl. Conf. on Cloud Computing, 2013, pp. 188–195.