Resource Allocation in Cloud Computing

K Subash Chandra Reddy[1] , Akanksha Singh[2]

Abstract— **Resource Allocation, a component of Resource Management, is of crutial important given its outcome effects the efficiency of resource utilization, profit generation for the parties involved and overall efficiency of the cloud-services itself. We are gonna discuss various Game Theory Concepts utilized at various stages of Resource Allocation in Cloud Computing and analyze the same.**

I. INTRODUCTION

Cloud Computing environment provisions the supply of computing resources on the basis of demand, as and when needed. It builds upon advances of virtualisation and distributed computing to support cost efficient usage of computing resources, emphasizing on resource scalability and on-demand services.

In the face of selfish agents who attempt to maximize their own utility, Ketchpel [1] suggested a "two agent auction" mechanism for coalition formation. Shapley Value is used to determine the utility an agent receives for his membership in the coalition. Coalition formation in the state of uncertainty pertaining to agent's lack of control in perfect consumption of resources is proposed by Hanna and Zreik [2]. This leads to an uncertain coalition value and a probabilistic model was proposed taking the issues into account.

Multi-Instance two-player zero-sum game for coalition formation and resource allocation in the face of uncertainity is proposed by Pillai and Shrisha [3]. While restricting to non-overlapping and fixed size, coalitions are formed prior to arrival of task requests from clients. This pre-computation naturally results in shorter response times to user requests. Extending this solution approach to VM allocations in Cloud Computing [4] and relaxing the coalition size and overlap constraints, a demand-aware topology-aware resource allocation mechanism was proposed by Pillai and Shrisha. Second-priced Auction Mechanism with the assistance of pricing and truth telling mechanism is suggested by Wui-Yu, Guan-Yu and Hung-Yu [5].

A hierarchical cooperative game model is presented by Niyato [9]. Stocastic Linear Programming Game Approach is proposed, which takes into account uncertainty of internal users from each provider into account during resource sharing and coalition formation. Handling requests for bundles of virtual machines (VMs), scenario referred to as Combinatorial auction-based allocation is proposed by Zaman [10]. Also proposed an alternative to fixed-price schemes which also provides natural load balancing as a side effect.

Lampe [11] takes it a bit ahead by concurrently dealing with both determination of equilibrium prices and distribution of virtual machine instances to the underlying physical hosts in the data center. Proposed two approaches, an optimal allocation approach which takes exponential time and a heuristic approach which is practical with its own limitations.

II. BACKGROUND & TERMINOLOGY

A. Coalitions

A coalition is a pact or treaty among individuals or groups, during which they cooperate in joint action, each in their own self-interest, joining forces together for a common cause. This alliance may be temporary or a matter of convenience. Coalitions, in cloud computing environment, are formed when individual agents can't cater to client's requests efficiently, and so form coalitions, which results in efficient use of resources, increased utility to the players involved.

An agent following the economic principle of rationality will attempt to form a coalition which will maximize its own utility.

B. Stable Coalition

For P to be stable , there must not be any other partition P' forming coalitions $C'_1, C'_2, ..., C'_m$ such that $\exists C'_i \in P'\ \forall a_j \in C'_i$

$$u(P', a_j) > u(P, a_j) \tag{1}$$

C. Utility of Agent and Coalition

In any game, utility represents the motivations of players. The more the utility of a game, the more it preferable by the player. An agent's (a) utility will be the immediate benefits he will be having for joining a coalition and a coalition's (C) utility will be the benefit the whole coalition will be having after its formation. Utility of a coalition is given by

$$u(C) = \sum_{a \in C} u(a) \tag{2}$$

D. Zero-Sum game

A Zero-Sum game is a mathematical representation of a situation in which each participant's gain or loss of utility is exactly balanced by the losses or gains of the utilities of the other participants.

E. Uncertainty principle in game theory

Uncertainty is a situation which involves imperfect and/or unknown information. In game theory, a player is unknown of other players moves, so his decisions are just based on his assumptions regarding the moves the other player might take up. An example of uncertainty in cloud computing is when the agents can't exactly control the consumption of resources, which there by results in uncertain coalition utility value.

F. Equilibrium Price

Equilibrium is the state in which market supply and demand balance each other and, as a result, prices become stable. Generally, when there is too much supply for goods or services, the price goes down, which results in higher demand. The balancing effect of supply and demand results in a state of equilibrium. Equilibrium Price in case of a two-player game is the combination if moved-away from, will always result in reduced utility to either of the players.

III. COMPARISION AND ANALYSIS

In cloud computing, as mentioned above, the resource allocation plays an important role in the performance of the entire system and also the level of customer satisfaction provided by the system. At the same time, profit generation is also of great importance to the service provider. So the resource allocation should be economical on both views i.e., on the end user and the service provider perspective.

The main problems related to cloud computing are the network bandwidth, response time, minimum delay in data transfer and minimum transfer cost for data. In this section, we are gonna look at various strategies of game theory used to efficiently handle the resource allocation problem.

A. Coalitions

There are two important challenges where coalitions are concerned

1) Determining the coalition of agents willing to work together for the benefit of all the members of the coalition.
2) How the joint reward should be divided among the agents in the coalition,

Hanna [2] discusses coalition in the scenario where both the utility of agent as well as the coalition are uncertain. Here each task is assumed to consist fixed number of subtasks. An agent, whose target is to maximize his expected reward, has two options at any given time. Either execute a subtask, which reduces his chances to execute another subtask as his resources will be depleted or ignore and save for the future. Agents select subtasks using Markov Decision Process that allows for optimal subtask selection policy and then coordinate their decisions using auction protocol. This results in agents announcing their payoff and allocate each task to coalition with minimal payoff.

While uncertainty in agents reward in [2] is because of agent's lack of control on resource consumption, Steven [1] presents an uncertainty scenario cause of the absence of a valuation function which determines the utility of a coalition. This results in different expectations of this value by different agents, thereby prevents in reaching a consensus for a coalition formation.

Agents exchange initial offers and each agent forms a preference list based on their expected profit. Agents enters into a two agent auction, where through a bidding process the agents in question decide upon an agreeable value one is willing to pay other to be in charge of the coalition, and form coalition of size two. Next iteration takes place with the pairs formed in iteration one as individual agents until the coalition size, if fixed, is met.

Niyato [9] presents a new uncertainty in the context of resource and revenue sharing with coalition formation. That is, the internal demand of data center on when the service has to be offered to the public cloud users. Explained in a different way, since the cooperation to form coalition incurs a certain cost, it may not be worth for some provider to join coalition. This demand uncertainty can affect the cooperation decision of cloud providers.

A hierarchical cooperative game model to handle the uncertainty. First, a stochastic linear programming approach is proposed, which takes the random internal demand of cloud providers into account to determine the optimal services i.e., available virtual machines to be offered to the public cloud users. Second, the resources and revenues are shared by a coalition game among the relevant cloud providers. Niyato proposes two Coalitional Structures

1) Stable Coalitional Structure: The individual profits of all providers are maximized thereby stability is guaranteed cause no player has an incentive to change the decision.
2) Optimal Coalitional Structure: The providers can form coalition to maximize the total profit instead of individual profit.

Given Optimal Coalitional structure tries to maximizes overall coalition utility, its value is not lower than that of stable coalition structure. Even though Optimal Coalitional Structure seems like the best bet, it fails to be practical as some members get high utility compared to others, thereby providing incentive for some to leave the coalition and compromise stability. Hence, utility of optimal coalitional structure serves as a theoretical upper bound to ulitity of stable coalitional structure.

While [1],[2] and [9] handle different uncertainties and deal with coalition formation after receiving user request and deciding on the appropriate coalition which will generate an agreeable utility for the agents involved, sometime at a risk to the agent in charge of the coalition, Parvathy [4] suggests a pre-computation model before the request is received.

By taking advantage of type of VMs that may be requested, coalitions are formed to host VMs before the actual request arrives. This presents a new uncertainty, that is coalition formation in the face of uncertain task information as coalitions are formed before the client's tasks are received.

Parvathy aims to provide an alternative to the integer programming approach used traditionally to solve payoff matrix for optimization like the one discussed in [9], as integer programming has a high complexity and goes out of bounds if the number of machines and requests, which corresponds to constraints and variables respectively, increase even by a modest amount.

A preference list is formed for each agent using the uncertainty principle of game theory. Given that the near-optimal coalition according to one agent may not be favored by another agent, creating a preference list quickens the coalition formation as the agent can now go on to checking the feasibility of next coalition in the list, thereby shortening response time to client's requests.

On receiving a client request, if an existing open coalition is capable of handling the request, it is assigned. Else, a coalition is forcibly formed out of ready agents and dissolved after the competition of the task. This approach not only results in minimal allocation time but also minimal resource wastage attributing to the fact that preconfigured coalitions will be assigned to tasks after ensuring minimum wastage.

B. *Equilibrium Price Auctions*

Maximization of auction profits is a challenge for cloud provider as it involves concurrent determination of equilibrium prices and distribution of virtual machines instances to the underlying physical hosts in the data center. Lampe [11] proposed a linear optimization approach as well as a heuristic based approach to tackle this Equilibrium Price Auction Allocation Problem. The profit for provider is the difference between revenue from the served bids and the operating costs of the PM instances.

Linear Optimization approach proposed turned out to be not practical given solving the Linear Program(LP) by Branch and Bound in worst case results in exponential time complexity for optimal solution. Heuristic Approach, assumes the initial prices for all VM's that will maximize the expected profit from the server bids. Though this reduces the complexity to polynomial, the optimality of the solution is dependent on the initial assumption, which adds some uncertainty/approximation into the solution approach. Yet this is first solution approach that concurrently solves the challenge of pricing and distributing VM instances based on an auction scheme.

C. *Combinatorial Auction*

Zaman [10] presents solution to a scenario where bids are placed on combinations of items, rather than just individual items. By incorporating combinatorial auction-based mechanisms for allocating VM instances, results in an alternate to fixed-price schemes which fail to reflect the equilibrium prices that arise from market demand and supply. This makes sure that a situation where, only subset of requested VMs are allocated, doesn't occur. Also ensures load balancing overtime as users with low valuations for VM instances will choose a time frame that doesn't conflict with that of high valuation users.

Three mechanisms are proposed that solve the virtual machine allocation problem (VMAP).

1) Fixed-price mechanism
2) Combinatorial Auction-Linear Programming
3) Combinatorial Auction-Greedy

Fixed-price is the mechanism currectly used by several cloud service providers where VM instances are allocated to the users in a first-come, first-served bases until the resources are exhausted. Both the combinatorial approaches are approximations as winner determination problem of combinatorial auctions is NP-hard.

In approach 2, a linear program is formulated that maximizes the sum of users' valuations and randomized rounding is done until the constraints are violated to select the winner. Approach 3 works by ranking the users in decreasing order of their 'bid density', and then greedily allocating from the top of the list. Experimental results suggest combinatorial-greedy approach clearly outperforms fixed-price mechanism in terms of resource utilization, revenue generated and allocation efficiency.

D. Agent's Rewards

The value of a coalition is dependent on which agent is performing the valuation, i.e., this value will depend upon his perceptions of the coalition. As by Steven [1],

$$v_A(AB) \neq v_B(AB) \neq v(AB) \tag{3}$$

Here, as in 2-agent situation, one agent may offer another agent a "share" of utility larger than the total obtained from the coalition due to its estimation.

While according to Hanna [2], forming a coalition to execute a task is a necessary but not sufficient constraint to obtain a reward and the agents' reward must be subjected to the task execution and not only to the coalition formation and task allocation.

While in [1], in case of failure of execution of task, only the manager will be bearing the loss , in [2], the agent whose share of work is incomplete will bear the consequences.

The reward function defined by [2] is

$$g_k : E_k(T^i) \Rightarrow R^+ \tag{4}$$

where g_k represents a_k's payoff for executing t. The final expected payoff of an agent is dependent on the probability to join a certain coalition.

To take into more realistic situations, Parvathy [3] says that the distribution of payoff also depends on the resources that sub-tasks requires, i.e. giving more payoff weightage to the tasks requiring more resources.

The reward function introduced in the paper, has a payoff function calculation taking into account capability of the agent, while also keeping track of necessity of the task.

In [4], while extending to cloud coomputing scenario, Parvathy defined the reward function in terms of capability of a host machine, coalition's resource capability and task's resource necessity, thus taking into account capability of not only agent but also the coalition which is performing the given task. Negative payoff may be awarded to the coalition if the task is left incomplete.

IV. SUMMARY

Pointing out the highlights, we have looked at uncertainties pertaining to different reasons including resource allocation, coalition formation and maximizing profit. While in coalition based scenario, Approach proposed by parvathy [4] turned out to be interesting by reducing response time and minimizing resource wastage. Zaman [10] came up with a scenario of VMs as bundles, which when handled by combinatorial auction, automatically provides load balancing and increased revenue generation.

An observation from the above discussion is, while it emphasizes the importance of resource allocation in cloud computing as a non-trivial and complex problem, it provides many approaches, be it either efficient or approximated, to handle the scenarios as efficiently as possible.

REFERENCES

[1] Steven Ketchpel, Forming Coalitions in the Face of Uncertain Rewards.

[2] H. Hosam, Z. Khaldoun, Planning Coalition Formation under Uncertainty: Auction Approach.

[3] Parvathy S Pillai, Shrisha Rao, A Resource Allocation Mechanism Using Coalition Formation and The Uncertainty Principle of Game Theory.

[4] Parvathy S Pillai, Shrisha Rao, Resource Allocation in Cloud Computing Using the Uncertainty Principle of Game Theory.

[5] Wei-Yu Lin, Guan-Yu Lin, Hung-Yu Wei, Dynamic Auction Mechanism for Cloud Resource Allocation.

[6] Patricia Takako Endo, Andre Vitor de Almeida Palhares, Nadilma Nunes Pereira, Resource allocation for distributed cloud: concepts and research challenges.

[7] Jalaparti, Virajith, Nguyen, Giang D, Cloud Resource Allocation Games.

[8] Donald F. Ferguson, Christos Nikolaou, Jakka Sairamesh, Yechiam Yemini, Economic models for allocating resources in computer systems.

[9] Dusit Niyato, Athanasios V. Vasilakos, Zhu Kun, Resource and Revenue Sharing with Coalition Formation of Cloud Providers: Game Theoretic Approach.

[10] Sharrukh Zaman, Daniel Grosu, Combinatorial Auction-Based Allocation of Virtual Machine Instances in Clouds.

[11] Ulrich Lampe, Melanie Siebenhaar, Apostolos Papageorgiou, Maximizing Cloud Provider Profit from Equilibrium Price Auctions.