Survey of Game Thereotic Approaches to Cloud Computing

RICARDO MARIN JR, University of Virginia

The main topic for this Capstone project is surveying Game-Theoretic Approaches in Cloud Computing. In this approach, the game theory model is used that best mimics how current tenants act under a certain policy. This allows for the creation of new strategies (e.g., pricing structures) to improve the cloud computing performance such as traffic congestion control, resource efficiency improvement, resource competition avoidance and so on. This project will explore recently published peer-reviewed articles to best understand the current state of knowledge on using game-theoretic approaches in improving the cloud performance and will provoke new ideas to be discussed for future research. This paper will achieve said goal by categorizing many peer-reviewed articles based on their objectives to see the current value in these approaches.

1 INTRODUCTION

As the need for Cloud resources has grown over the last decade, so has the competition between providers. As such, there has been a tremendous amount of research performed that try to further optimize cloud systems. One prominent segment of the research performed have used game theory as a modeling tool to model both existing and proposed systems. Game theory is the study of mathematical models of strategic interaction among rational decision-makers. Essentially, game theory allows for predictions on what could happen to existing system if change is introduced. There are a few different ways game theory can be utilized. There is cooperative game, where players work together for a collective maximum benefit, and non-cooperative game, where players work individually for their own maximum benefit. There are four approaches that have used game theory in a prominent way that will be discussed: Task Scheduling, Load Balancing, Pricing, and Resource Allocation.

2 RELATED WORK

A 2019 paper by Sane et al. is a survey of game theoretic approaches for cloud computing security issues [39]. While this paper looks at game theoretic approaches, my paper does not look at possible security issues at all. Furthermore, [1] is a survey that focuses solely on task scheduling. [36] solely looks at load balancing, with a heavy emphasis on the game theoretic approaches. [31] is a survey of game theory with regards to multi-access edge computing. [33] is a survey on resource allocation overall, briefly mentioning game theory. [28] also briefly mentions game theory in its survey of resource allocation techniques. [22] is a survey of spot pricing techniques used in cloud computing, included ones based on game.

3 APPROACHES USED

This paper will explore four main game theoretic approaches that researchers have explored in order to better the usage of Cloud Computing for both tenants and providers. All papers discussed were published within the last decade with many ideas taking a different approach solving the same problem.

3.1 Task Scheduling

In cloud computing, task scheduling is defined as assigning a task to different resource nodes corresponding to perform appropriate strategies for performance reasons. [50]. Essentially, the more efficient the resources are used, the more tasks that can be run. This is highly beneficial for both the provider and its tenants as the better a task scheduling algorithm can become, the more tasks can be run and the less time it takes waiting for a task. Currently, the "State of the Art" algorithms are "Max-Min" and "Min-Min". Briefly, the Max-Min algorithm takes the task with the maximum expected time and assigns it to the resource with the smallest execution time [5]. By contrast, the Min-Min algorithm selects the task with the smallest expected time and assigns it to the resource with the quickest execution.

3.1.1 An energy management based task scheduling algorithm. In the Yang et al. paper from 2017, an algorithm that improves energy management is proposed. They used cooperative game theory along with sequential game thing to develop the task scheduling algorithm. This paper uses the idea of a balanced scheduling algorithm as a basis for the proposed algorithm. As such, the paper found that nodes with a balanced ability are able to achieve a higher computation power. This paper used the the task processing time delay as a utility function of the game, with the tasks acting as "players" [50]. This algorithm also behaves the same without respect to compute resources, meaning it provides benefits to both large and small number of resources. Energy management is a major benefit to cloud providers as minimizing the energy used can significantly lower costs for providers. However, since processing cost is ignored, the actual processing time needed may be higher than usual and offset gains.

3.1.2 Real Time Task-Scheduling. In the 2019 Patra et el. paper, a model for real time task scheduling is developed. The tasks are sorted based on deadline and them broken into groups of two alongside a scheduler. The scheduler then

Author’s address: Ricardo Marin Jr, rickym@virginia.edu, University of Virginia.

"plays" the two tasks. They use both a non cooperative game model along with a cooperative game model, finding the non cooperative model gave better results. This makes sense, since the tasks don't care about each other's waiting time, focusing solely on waiting the least amount of time [34] Completion time and waiting time are both the main "payoffs", with the waiting time of tasks wanting to be minimized. However, this neglects to look at energy consumption, which can be very high for providers, therefore very costly. Also, this approach scales with number of virtual machines, so smaller systems should not use this approach. In the future, adding additional constraints should further improve this algorithm.

3.1.3 Task Scheduling with Fog Computing. Fog Computing is an extension of Cloud Computing with the exception being that most processing is done in a decentralized enviromnent. Instead of using centralized data servers, fog computing uses a device such as a router to handle tasks. This has allowed for very low latency tasks, however the resources capacity and power is very minimal compared to using large data centers. Therefore, in the 2020 Arisdakessian et al paper, researchers used a new schedule approach that utilized game theory. The unique aspect that this paper brought was it utilized game theory differently than other related works. Instead of considering business related parameters, the researchers used delay constraints and resource utilization in order to maximize QoS [3]. Instead of maximizing profits for the provider that some other papers do [48], this paper focuses on QoS for the tenant. By doing so, this paper's approach is better for tenants. By taking account of preferences, this technique actually improves execution time compared to the SOTA.

3.1.4 Power Efficiency . Power efficiency can also be a valuable metric that cloud providers try to maximize. In a 2017 paper by Lei Zhang and Jin-he Zhou, energy efficiency is the key metric that the proposed task scheduling algorithm maximizes. The paper mentions that QOS is usually researched, while the growing energy demand is rarely examined [51]. This paper tries to accomplish optimal tasks scheduling while still guaranteeing the QoS that tenants need. In their non-cooperative game model, servers act as players with the utility is unit power efficiency. As more computing resources are used, the unit power consumption increases, but QoS also increases. Therefore, this technique forces the servers to find the best task proportion while maximizing the power efficiency. This is important because while tasks may be scheduled to be executed quicker than this algorithm, the power consumption may be very high. By making power consumption a decision decider, providers can still guarantee high QoS while minimizing their own costs.

3.1.5 Earliest Finish Time. In an improvement over an existing algorithm (Heterogeneous Earliest Finish Time), Samadi et al take into account load balancing. Originally, the HEFT algorithm minimizes overall execution time, but neglects budget constraint and load balancing. In this approach, the scheduler minimizes the data transfers that may be called for by the HEFT algorithm [38]. This algorithm is comprised in four stages. In the first stage, a "load threshold" is assigned. Second, datasets are clustered into different data centers based on certain dependencies. Since certain datasets may be needed by multiple tasks, they are grouped such that they will stay together. This is important because this reduces the possible data transfers that may be needed saving executing time. Next, A graph is creating that shows a ranking for execution. If a task is on the same level of another task, this means that the two tasks can be executed at the same time without any worry of dependencies. Finally, game theory is used to actually choose the optimal VM to run the tasks. Each task can choose one VM, while VMs can hold up to a certain quota. Preferences that the tasks decide from include processing power, memory, and storage capacity. The least impact choice gets the highest rank. This algorithm beats the original HEFT by 26%, solely by looking at the VM allocation in the first phase. This crucial step was a major improvement over the existing algorithms and shows the connection between task scheduling and load balancing.

3.1.6 Eagle: Data Center Scheduler. Eagle is a data center scheduler that uses Succinct State Sharing to partition notes dynamically. A new technique, Sticky Batch Probing provides job awareness and avoids stragglers. Eagle outperforms other state of the art scheduling solutions. [8]

3.1.7 Energy-Efficient Mechanisms. This paper is a operating model of a cloud computing server [2]. A mechanism is budget balanced with a low communication complexity. This also allows for a better Price of Anarchy, meaning the social welfare of a system does not degrade as previous mechanisms.

3.1.8 Cost-Aware Task Assignment . This paper focuses on the cost of data centers in a distributed, non-cooperative environment [27]. Each agent is given a estimated cost of tasks to minimize while following QoS constraints. A BRA optimizes the utility function to reach a Nash equilibrium.

3.2 Load Balancing

Load Balancing is a technique that distributes work evenly throughout all of the nodes available. [32]. This means that when work is received, there exists an algorithm that tells where the work should be completed as to not overload any of the existing nodes. This is an important problem because a key metric used by cloud providers is the resource utilization ratio. This is essential to prevent a node overload which creates bottlenecks within the system. However, this problem is non-deterministic, therefore there is not believed to be an "optimal" solution.

3.2.1 *Particle Swarm Optimization and Game Theory.*

In a 2019 Mrhari and Hadi paper, a new load balancing algorithm called SASPSOLB is introduced, promising to find a near optimal to the load balancing problem[32]. The researchers used a non cooperative game theory model alongside an existing optimization called SPSO that solves the game problem. Particle swarm optimization is an optimization method that can converge as a group looking at aggregation such as bird flocking or a swarm of insects [52]. This optimization is used for solving the non cooperative game which aims to minimal response time as the game. This algorithm is nearly 3x faster at distributing tasks to nodes than GALB, one of the most common load balancing algorithms used. However, this method may not be the most efficient or the most scalable. Therefore, the impact that this algorithm brings is limited in real world use.

3.2.2 *Load Balancing Using SGMLB.*

In a paper published in 2020 by Swathy et al, an approach using the Stackelberg game theoretical model was introduced to best deploy tasks [45]. The Stackelberg model is an economic model in which a player is a "leader" while the other player is a "follower". This means the "leader" makes the first decision with the "follower" following, meaning the leader gets the most benefits. In this load balancing approach, the "leader" deploys tasks one by one reliably, while the follower (host) selects the minimum utility function value. By having the leader allocate tasks based on best price and computing availability of its hosts, tasks are deployed at a fast rate and resources are utilized 60% more efficiently than the current SOTA.

3.2.3 *Cooperative Game with Cost-efficiency.*

In a 2014 paper, Song et al. utilizes a cooperative game approach among processing nodes in order to improve fairness of the cloud system [43]. This approach assign processing nodes into groups, with each group having a "load manager". These load managers are responsible for playing the game in order to minimize cost. Since this is a cooperative game, the load managers work together trying to benefit the collective group. Compared to two older algorithms, this approach has a much better fairness index for users and processors. Indeed, at a 70% utilization, average completiion time of tasks for users is significantly lower than the standard two algorithms provide.

3.2.4 *Fair Load Balancing.*

In 2016, Xiao et al proposed a dynamic non-cooperative game model that promised fairness [46]. This approach is for large scale distributed computing systems as it uses reinforcement learning to improve the game model. This paper focuses heavily on heter-scheduler collaboration, which is responsible for accepting or rejecting jobs in order to maximize its own profit. By modeling a scheduling problem using a non-cooperative game where each heter-scheduler is wanting to minimize the response time of its jobs, a balanced load is actually ensured among all schedulers since there would be less profit otherwise. The machine learning aspect of this approach is able to learn the game and further strengthen the algorithms.

3.2.5 *Energy-Aware Load Balancing.*

Yang et al. took a completely different approach than the previous papers, instead focusing on minimizing energy consumption of data centers [49]. The motivating factor for this is that reducing number of available resources can reduce QoS, so trying to keep the same QoS while using less power is very ideal. However, as will be discussed later on, average response time may be slower than other load balancing techniques. The energy model keeps power consumption of computing resources proportional to its current workload. However, the game theory model has QoS as a constraint. Therefore, the model will not let response time be long enough such that it would interfere or breach a SLA with a tenant. In this cooperative game, minimum performance is guaranteed. Load rate is a major factor for the average response time, while the QoS level achieved greatly impacts the energy consumption of a cloud center. This paper is very important for cloud users who do not need the highest QoS, as the providers could pass the savings down. Indeed, the providers have a massive financial benefit to implement an energy aware load balancing algorithm for at least some data centers since the energy cost savings would be substantial. A possible issue with this approach is demand spikes. If a data center is in a low power mode and there is a massive surge of traffic, the time to scale up may be very damaging to both the tenant and the provider.

3.2.6 *Core Resilience.*

K-cores are maximal induced subgraphs where all vertices have degree of at least k [30]. This paper looks at algorithms for k-core modifications using a game-theoretic approach. By doing so, the paper's algorithm outperforms other solutions. This work can be used for the resilience of cloud networks.

3.2.7 *Incentivizing Collaboration.*

This paper proposes a mechanism based on localized social welfare [42]. This means individual effort also benefits neighboring agents. While this paper is not directly applied to cloud computing, the ideas can be brought to a computing nature.

3.2.8 *Online-Learning Congestion Control.*

This paper introduces a rate-control protocol which outperforms traditional TDP variants [10]. It also shows a stable global rate configuration always exists.

3.2.9 *Edge Offloading.*

In this paper, tasks are divided into subtasks to form into a set of groups[20]. This reduces task processing time by around 30%.

3.2.10 Multi-User Computation Offloading. This paper formulated the distributed computation offloading problem as a multi-user game [6]. Results show this distributed computation offloading algorithm scales as user size increases.

3.3 Pricing

Pricing has been a significant point of research in recent years as price is a major factor for budget-aware tenants [23]. Therefore, optimizing pricing strategies that can attract and retain more tenants while still making profit is highly sought after by providers.

3.3.1 Pricing in a Duopoly Market. Li et al. was one of the first papers to explore pricing competition in a cloud computing market and the first to use a game theoretic approach towards studying pricing [25]. In this approach, the researchers modeled a duopoly cloud market, which makes since given the Amazon Web Service and Azure market dominance at the time. In this non-cooperative game, cloud providers charge a fixed price to all possible tenants per time used, not amount of resources used. An interesting result found was then when providers have the same capacities available, they are forced charging the same price as each other. This paper was very early to tackle a game theoretic approach to pricing, but it paved the way for future studies to be conducted.

3.3.2 Strategic Bidding. Sowmya and Sundarraj's 2012 paper gives an incredible insight on how tenants view pricing. They model bidding strategies of tenants who are trying to buy cloud instances [44]. Then, they used actual Amazon data to verify their model and discovered that most tenants decide to defect (place a high bid). However, if two tenants are to meet again, cooperating (placing a low bid) is actually more useful. This paper evaluated real-time data to verify their model which is atypical of a majority of the papers discussed. This paper shows how tenants are likely to be as high as they can, even when doing so may mean paying more than the fixed "on-demand" cost. .

3.3.3 Pricing Negotiation. The paper by Tapale et al. takes a different approach to the bidding process from earlier above. Instead, there is an existence of a resource broker that connects the tenant's wants to what any provider can give [14]. This is an interesting dynamic as the tenant tries to use the most resources at minimal cost while the provider tries to maximize its own profit by increasing resource cost. This differs to the previous paper as multiple tenants were negotiating with a single provider compared to this technique in which one tenant does not care about the provider used as long as price is minimized. However, if a provider would consistently undercut its competitors by accepting low offers, its resource utilization rate may skyrocket. This could force the provider to increase its threshold for acceptance. This game helps tenants and providers increase their own profits while maximizing utilization.

3.3.4 Multi-attribute Price Bidding. This 2018 Hu et al. paper takes the bidding process even further. In the previous two papers, price was the only determining factor on whether a tenant or provider would accept each other's offer. However, QoS is a major factor that this technique includes as an attribute for the bidding process [19]. This approach allows tenants the ability to stipulate certain QoS metrics (bandwidth, latency, and reputation). This paper focuses solely on one approach: one customer and multiple cloud providers. Although they explain how the approach would work in other conditions, some more research would be highly valuable in this area. Nevertheless, in the one tenant condition, a customer submits their highest price for one resource. Multiple providers then submit their attributes and resources process to the customer. Finally, the customer selects the most idea set of attributes and prices that fit within their ideal criteria. This approach is very beneficial for the customer as many times they don't need the lowest latency rate or the highest bandwidth that a provider can provide. Providers do not also have to give their prices beforehand, which means new tenants may not have an expectation of what an instance may cost. This means that, in some cases, providers can further maximize their profits.

3.3.5 Pricing Dynamics. A major topic when it comes to choosing a Cloud Provider as a tenant is cost. In a 2016 paper, Do et al. analyzes the competition that exists between maximizing profits and gaining new tenants [9]. This model comes by two stages of competition; first, Cloud Providers compete as they try to maximize profits, and second, cloud users select the best performance and price that fits their needs. This is quite the opposite from the Hu et al. paper, as possible tenants know the costs before making their decision. There are two costs that a tenant has to handle: service cost and delay costs. Therefore while the cloud service providers play a non-cooperative static game where they compete with each other, tenants play an evolutionary game to get the best deal possible. However, this paper limits this approach to a duopoly market. Therefore, this approach should be studied more in an oligopoly market, since the number of major cloud providers continues to rise and the level of competition rises between the providers.

3.3.6 Profit Maximization. A 2020 Zhu et al. paper focuses primarily on maximizing the profits of SaaS and IaaS providers. This technique counters the bidding techniques discusses above along with the fixed pricing model structure [54]. This is a unique situation where IaaS providers maximize profits by having affordable virtual machines while SaaS providers focuses more on complying with SLA contracts. Therefore, there is conflict between maximizing profits between the two ways to run. The unique aspect of this paper is that is tackles the SLA in its pricing game, the first paper to do. As such, IaaS providers and SaaS providers play leader and follower respectively with SLA being a major factor to determine share for the SaaS provider. This approach maximizes the IaaS profit as the prices can dynamically increase as the demand for VMs increase. With an auction based system, the bids are too uncertain, meaning that there are times where profit is very low even if demand is high. However, this paper neglects to look at the competition between multiple IaaS providers, which is a key aspect for tenants as seen in [14]. Therefore, further work should be done to combine the results in the two studies to best determine the impact of SLA and multi IaaS Provider selection for profits.

3.3.7 Availability Knob. The Availability Knob provided flexible, user-defined availability in IaaS clouds, giving more control to customers [40]. Due to more efficient markets, provider costs are reduced and profits are increased. Game theory is used to derive incentive compatible pricing.

3.3.8 Bidding the Cloud. Amazon EC2 uses spot pricing to sell capacity, where user bids over this price are accepted [53]. Running different optimal bidding strategies show spot pricing reduces user cost by 90% while slightly increasing completion time.

3.3.9 Cloud Federations. This paper looks at showing that Cloud Federations would benefit both Cloud Providers and tenants [7]. Cloud Providers participate in a game by changing federation policies to maximize individual profits.

3.3.10 Multiple User Competition. This paper focuses on different strategies that tenants make in order to maximize value [26]. Solving this problem led to an iterative proximal algorithm that configures a proper request strategy.

3.3.11 Price Bidding Configurations. This paper focuses on pricing bidding strategies for resource usage [24]. Each tenant has a function which combines net profit with time efficiency. An iterative proximal algorithm is formulated to compute a Nash equilibrium solution.

3.3.12 Weighted Voting. This paper proposes a new cooperative game in which agents form coalitions to share gains [29]. Agents are able to obtain higher shares than people who played non cooperatively.

3.4 Resource Allocation

Resource Allocation is similar to load balancing, but with a wider focus on metrics such as bandwidth allocation and link congestion. Resource Allocation ensures that there is enough resources to handle tasks needed by a provider's tenants. A major concern that this area of research tries to tackle is the under-utilization of computing resources

3.4.1 Fair Resource Allocation. In one of the first papers to thoroughly improve resource allocation, Wei et al. uses Binary Integer Programming along with game theory to find a fair allocation algorithm that satisfies QoS agreements. This is a non cooperative game as tasks try to solve their problem independently. A cost is assigned by number of resources needed. Since the game doesn't know the end allocation in the beginning, there has to be an evolutionary process that handles the changes. A multiplex allocation minimizes the efficiency loss at each evolutionary step. However, this paper makes significant assumptions that limit the use case. For example, bandwidth cost is not included in the model. This means that the same load balancing problem that the earlier Samedi et al. tried to solve.

3.4.2 Bandwidth Sharing Incentives. A 2014 paper by Shen et al. took a unique approach to the pricing problem described in the previous section by adding bandwidth sharing policies [41]. The premise of this idea is that tenants may have unused bandwidth that can impact the overall network. Although a majority of tenants may not even know that they are hurting the network indirectly, some may, which can be very unfair. Therefore, the researchers proposed a payment structure where cost is based on consumed bandwidth. This causes tenants to be cooperative towards each other indirectly as their motivation is to minimize costs. However, this has tremendous benefits for both the provider and the tenant in other ways. By tenants using uncongested network links, providers are less likely to have SLA violations while tenants have better performance. A major concern that cloud providers have is about "min-guarantees", best described as the minimum service that a provider will ensure at all times. Any violation of this can cause significant profit loss to the provider and possibly the loss of the customer. Therefore, there is high motivation for a provider to implement a technique like the one described to ensure all contracts are fulfilled.

3.4.3 Further Fair Resource Allocation. Also from 2014, researchers took a different approach on reducing resource waste. However, instead of a main focus on the provider being able to optimize for their "min-guarantees", this technique focuses on giving no one better resources than others [47]. By doing so, the researchers try to maximize the minimum consumption among resources while also minimizing uneven consumption. First, the researchers design a new resource management system. The allocation of resources is time based, where decision moments decide whether the work can be done in a certain time slot. Each user has a maximum share fraction of the total capacity, but there are also three major properties this paper strives to satisfy: Sharing incentive (splitting the total resources equally), envy freeness (tenants do not prefer another tenant's allocation), and the Pareto efficient (increasing the resource of a tenant without decreasing anothers is impossible). The proposed algorithm (FUGA) satisfies all three of these properties. The game that tenants play is to share resources impartiality. This can further maximize resource utilization rate. This is a very interesting approach because it neglects the idea of tenant-specific SLAs. Since everyone has the same allocation of resources, providers would not be able to charge more for QoS when it comes to having a min-guarantee.

3.4.4 *Improved Performance through Coalition-formation.*

While the previous paper took a "fair" allocation process, Pillai et al. took a maximum performance based approach towards solving the resource allocation process [35]. This approach tries to guarantee certain tenant requirement like other papers discussed above. In this approach, resources act as "agents" in a cooperative game in which coalitions are formed in order to ensure a task can be done. For example, an agent might need to form a coalition with two or more agents to execute a job. Therefore, agents try to maximize the minimum payoff. Overall, the approach provided lower task allocation time, while preventing resource wastage. This is an interesting dynamic as this algorithm is not the best at assigning tasks. Therefore, while the job execution time was faster than other algorithms, this approach has processing time for actually assigning the coalitions needed.

3.4.5 *Coral-Reefs Optimization.*

One paper combined a unique biological-based algorithm along with game theory to best optimize elastic cloud resource allocation [12]. The coral-reef algorithm simulates a coral reef where coral fight for space, grow and reproduce, and fight with other corals [37]. This algorithm allows for the optimization of complex systems. In this paper, cloud elasticity is modeled using this method. This ability for cloud providers to have is essential to make it appear that the cloud is without bottlenecks. Therefore, satisfying SLAs while using the least amount of resources is essential to all providers. This paper tackles the using the mentioned coral reef optimization an a non cooperative game that best identifies the best resource reallocation. A set of VMs in a host machine act as players where the strategy is whether to migrate or not. The cost value has the incentive for player not to have to activate a new Host Machine. The whole procedure works by having the game simulation satisfy the number of requests generated. Overall, this solution satisfies the demands of tenants while also maximizing the profit for cloud providers. In future work, this coral reef approach should used for other optimization problems other than resource allocation such as task scheduling.

3.4.6 *Mean-Field Games.*

This paper looks at the last level cache sharing problem in large scale cloud networks [16]. The paper shows that a mean-field-taking strategy is a stable strategy and a equilibrium is reached as number of players grows. The optimal price for resources converge to the optimal price of the mean-field game.

3.4.7 *User Allocation in Edge Computing Environment.*

This paper seeks to minimize network latency and energy consumption with minimum overall system cost [18]. While the optimal solution is NP-hard, the proposed EUAGame formulate the problem as a game which accomplishes a Nash Equilibrium.

3.4.8 *Smart Cloud Storage Service Selection.*

This paper tries to make the selection of a cloud provider easier to customers, with game theory being used for promoting the truth-telling providers [11]. This helps reduce the confusion that multi-objective nature that providers and tenants have.

3.4.9 *Network Games.*

This paper is the foundation of a methodology for reasoning about network games. Game theory has been used to analyze network design and this paper maps the state space of a network game to a much smaller space [4].

3.4.10 *Efficient and Fair Allocation Mechanism.*

This paper proposes a server-based approach in which each server allocates by maximizing a per-server utility function [21]. This follows desirable properties such as bottleneck fairness and sharing incentive. This can also be implemented in a distributed fashion.

3.4.11 *Complete Information Sharing.*

This paper regards cooperative game using complete information sharing [13]. Using a new allocation rule called mood value, user satisfaction is equalized with a distribution of the available resources.

3.4.12 *Mean-Field Games.*

Last level cache sharing problems in cloud networks is examined in this paper [17]. Each player implements a mean-field-taking strategy; a successful strategy in both finite and infinite player scenarios. This also reaches the optimal prices for resources.

3.4.13 *Fair Network Bandwidth Allocation.*

In this paper, the main objectives are to guarantee bandwidth for VMs based on requirements and to share bandwidth in proportion to the weight of VMs[15]. Using a bargaining game, a new asymmetric allocation algorithm is proposed to meet those objectives.

4 CONCLUSION

Numerous papers have been presented to show the new and upcoming approaches to tackle many existing problems within Cloud Computing. It is evident that newer research has been built up from prior research, which has lead to great innovations for both tenants and users. Techniques for task scheduling include vast improvements in power efficiency, while techniques for load balancing will allow for tasks to be executed significantly quicker allowing for the execution of even more tasks. There has been many new pricing strategies introduced and while some build upon each other, some maximize for a specific metric that benefits providers and/or users. Finally, improvements in resource allocation techniques can bring massive benefits to both tenants and cloud providers such as reduced costs and improved utilization/efficiency rates. Overall, these techniques show major promise given how game theory was used in order to derive many of the papers' conclusions. Whether either modeling existing systems or verifying a proposed algorithm, game theory has allowed researchers to have great confidence that these proposed algorithms will actually work outside of a theoretical environment.

5 FUTURE WORK

Future work could look more at how multi attribute decision making impacts many of these algorithms. Many of these papers were only able to look at a certain attribute or look at maximizing/minimizing a certain characteristic. While this would be immensely difficult, this would be one step further in putting many of these algorithms in the real world. Using optimization algorithms such as the coral reef algorithm can further model the real world strategies of both users and tenants with game theory sounds especially promising and further research should conducted using the technique.

ACKNOWLEDGMENTS

To Professor Shen and Professor Xu, I thank you sincerely for the help with the proposal of this incredible idea. I have learned so much about all the research that takes place with cloud computing and the scope of this project would never had made it this far without you both.

REFERENCES

[1] A. Ananth and K. C. Sekaran. 2014. Game theoretic approaches for job scheduling in cloud computing: A survey. In *2014 International Conference on Computer and Communication Technology (ICCCT)*. 79–85. https://doi.org/10.1109/ICCCT.2014.7001473

[2] Antonios Antoniadis, Andrés Cristi, Tim Oosterwijk, and Alkmini Sgouritsa. 2020. A General Framework for Energy-Efficient Cloud Computing Mechanisms. In *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems* (Auckland, New Zealand) *(AAMAS '20)*. International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, 70–78.

[3] Sarhad Arisdakessian, Omar Abdel Wahab, Azzam Mourad, Hadi Otrok, and Nadjia Kara. 2020. FoGMatch: An Intelligent Multi-Criteria IoT-Fog Scheduling Approach Using Game Theory. *IEEE/ACM Transactions on Networking* 28, 4 (2020), 1779–1789.

[4] Guy Avni, Shibashis Guha, and Orna Kupferman. 2017. An Abstraction-Refinement Methodology for Reasoning about Network Games. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*. 70–76. https://doi.org/10.24963/ijcai.2017/11

[5] Upendra Bhoi, Purvi N Ramanuj, et al. 2013. Enhanced max-min task scheduling algorithm in cloud computing. *International Journal of Application or Innovation in Engineering and Management (IJAIEM)* 2, 4 (2013), 259–264.

[6] X. Chen, L. Jiao, W. Li, and X. Fu. 2016. Efficient Multi-User Computation Offloading for Mobile-Edge Cloud Computing. *IEEE/ACM Transactions on Networking* 24, 5 (2016), 2795–2808. https://doi.org/10.1109/TNET.2015.2487344

[7] George Darzanos, Iordanis Koutsopoulos, and George D. Stamoulis. 2019. Cloud Federations: Economics, Games and Benefits. *IEEE/ACM Trans. Netw.* 27, 5 (Oct. 2019), 2111–2124. https://doi.org/10.1109/TNET.2019.2943810

[8] Pamela Delgado, Diego Didona, Florin Dinu, and Willy Zwaenepoel. 2016. Job-Aware Scheduling in Eagle: Divide and Stick to Your Probes. In *Proceedings of the Seventh ACM Symposium on Cloud Computing* (Santa Clara, CA, USA) *(SoCC '16)*. Association for Computing Machinery, New York, NY, USA, 497–509. https://doi.org/10.1145/2987550.2987563

[9] Cuong T Do, Nguyen H Tran, Eui-Nam Huh, Choong Seon Hong, Dusit Niyato, and Zhu Han. 2016. Dynamics of service selection and provider pricing game in heterogeneous cloud market. *Journal of Network and Computer Applications* 69 (2016), 152–165.

[10] Mo Dong, Tong Meng, Doron Zarchy, Engin Arslan, Yossi Gilad, Brighten Godfrey, and Michael Schapira. 2018. PCC Vivace: Online-Learning Congestion Control. In *15th USENIX Symposium on Networked Systems Design and Implementation (NSDI 18)*. USENIX Association, Renton, WA, 343–356. https://www.usenix.org/conference/nsdi18/presentation/dong

[11] C. Esposito, M. Ficco, F. Palmieri, and A. Castiglione. 2016. Smart Cloud Storage Service Selection Based on Fuzzy Logic, Theory of Evidence and Game Theory. *IEEE Trans. Comput.* 65, 8 (2016), 2348–2362. https://doi.org/10.1109/TC.2015.2389952

[12] Massimo Ficco, Christian Esposito, Francesco Palmieri, and Aniello Castiglione. 2018. A coral-reefs and game theory-based approach for optimizing elastic cloud resource allocation. *Future Generation Computer Systems* 78 (2018), 343–352.

[13] F. Fossati, S. Hoteit, S. Moretti, and S. Secci. 2018. Fair Resource Allocation in Systems With Complete Information Sharing. *IEEE/ACM Transactions on Networking* 26, 6 (2018), 2801–2814. https://doi.org/10.1109/TNET.2018.2878644

[14] RH Goudar, Manisha T Tapale, and Mahantesh N Biqe. 2017. Price negotiation for cloud resource provisioning. In *2017 International Conference On Smart Technologies For Smart Nation (SmartTechCon)*. IEEE, 1027–1032.

[15] J. Guo, F. Liu, J. C. S. Lui, and H. Jin. 2016. Fair Network Bandwidth Allocation in IaaS Datacenters via a Cooperative Game Approach. *IEEE/ACM Transactions on Networking* 24, 2 (2016), 873–886. https://doi.org/10.1109/TNET.2015.2389270

[16] Ahmed Farhan Hanif, Hamidou Tembine, Mohamad Assaad, and Djamal Zeghlache. 2016. Mean-Field Games for Resource Sharing in Cloud-Based Networks. *IEEE/ACM Trans. Netw.* 24, 1 (Feb. 2016), 624–637. https://doi.org/10.1109/TNET.2014.2387100

[17] A. F. Hanif, H. Tembine, M. Assaad, and D. Zeghlache. 2016. Mean-Field Games for Resource Sharing in Cloud-Based Networks. *IEEE/ACM Transactions on Networking* 24, 1 (2016), 624–637. https://doi.org/10.1109/TNET.2014.2387100

[18] Q. He, G. Cui, X. Zhang, F. Chen, S. Deng, H. Jin, Y. Li, and Y. Yang. 2020. A Game-Theoretical Approach for User Allocation in Edge Computing Environment. *IEEE Transactions on Parallel Distributed Systems* 31, 03 (mar 2020), 515–529. https://doi.org/10.1109/TPDS.2019.2938944

[19] J. Hu, K. Li, C. Liu, and K. Li. 2018. A Game-Based Price Bidding Algorithm for Multi-attribute Cloud Resource Provision. *IEEE Transactions on Services Computing* (2018), 1–1. https://doi.org/10.1109/TSC.2018.2860022

[20] M. Hu, Z. Xie, D. Wu, Y. Zhou, X. Chen, and L. Xiao. 2020. Heterogeneous Edge Offloading With Incomplete Information: A Minority Game Approach. *IEEE Transactions on Parallel and Distributed Systems* 31, 9 (2020), 2139–2154. https://doi.org/10.1109/TPDS.2020.2988161

[21] J. Khamse-Ashari, I. Lambadaris, G. Kesidis, B. Urgaonkar, and Y. Zhao. 2018. An Efficient and Fair Multi-Resource Allocation Mechanism for Heterogeneous Servers. *IEEE Transactions on Parallel and Distributed Systems* 29, 12 (2018), 2686–2699. https://doi.org/10.1109/TPDS.2018.2841915

[22] Dinesh Kumar, Gaurav Baranwal, Zahid Raza, and Deo Prakash Vidyarthi. 2018. A survey on spot pricing in cloud computing. *Journal of Network and Systems Management* 26, 4 (2018), 809–856.

[23] Ang Li, Xiaowei Yang, Srikanth Kandula, and Ming Zhang. 2011. Comparing public-cloud providers. *IEEE Internet Computing* 15, 2 (2011), 50–53.

[24] K. Li, C. Liu, K. Li, and A. Y. Zomaya. 2016. A Framework of Price Bidding Configurations for Resource Usage in Cloud Computing. *IEEE Transactions on Parallel Distributed Systems* 27, 08 (aug 2016), 2168–2181. https://doi.org/10.1109/TPDS.2015.2495120

[25] X. Li, B. Gu, C. Zhang, K. Yamori, and Y. Tanaka. 2014. Price competition in a duopoly IaaS cloud market. In *The 16th Asia-Pacific Network Operations and Management Symposium*. 1–4. https://doi.org/10.1109/APNOMS.2014.6996552

[26] C. Liu, K. Li, C. Xu, and K. Li. 2016. Strategy Configurations of Multiple Users Competition for Cloud Service Reservation. *IEEE Transactions on Parallel Distributed Systems* 27, 02 (feb 2016), 508–520. https://doi.org/10.1109/TPDS.2015.2398435

[27] S. Long, W. Long, Z. Li, K. Li, Y. Xia, and Z. Tang. 2020. A Game-based Approach for Cost-aware Task Assignment with QoS Constraints in Large Data Centers. *IEEE Transactions on Parallel and Distributed Systems* (2020), 1–1. https://doi.org/10.1109/TPDS.2020.3041029

[28] Sunilkumar S. Manvi and Gopal Krishna Shyam. 2014. Resource management for Infrastructure as a Service (IaaS) in cloud computing: A survey. *Journal of Network and Computer Applications* 41 (2014), 424 – 440. https://doi.org/10.1016/j.jnca.2013.10.004

[29] Moshe Mash, Yoram Bachrach, and Yair Zick. 2017. How to form winning coalitions in mixed human-computer settings. In *Proceedings of the 26th international joint conference on artificial intelligence (IJCAI)*. 465–471.

[30] Sourav Medya, Tiyani Ma, Arlei Silva, and Ambuj Singh. 2020. A Game Theoretic Approach For Core Resilience. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, Christian Bessiere (Ed.). International Joint Conferences on Artificial Intelligence Organization, 3473–3479. https://doi.org/10.24963/ijcai.2020/480 Main track.

[31] José Moura and David Hutchison. 2018. Game theory for multi-access edge computing: Survey, use cases, and future trends. *IEEE Communications Surveys & Tutorials* 21, 1 (2018), 260–288.

[32] Amine Mrhari and Youssef Hadi. 2019. A Load Balancing Algorithm in Cloud Computing Based on Modified Particle Swarm Optimization and Game Theory. In *2019 4th World Conference on Complex Systems (WCCS)*. IEEE, 1–6.

[33] Swapnil M Parikh. 2013. A survey on cloud computing resource allocation techniques. In *2013 Nirma University International Conference on Engineering (NUiCONE)*. IEEE, 1–5.

[34] Manoj Kumar Patra, Sampa Sahoo, Bibhudatta Sahoo, and Ashok Kumar Turuk. 2019. Game theoretic approach for real-time task scheduling in cloud computing environment. In *2019 International Conference on Information Technology (ICIT)*. IEEE, 454–459.

[35] Parvathy S Pillai and Shrisha Rao. 2014. Resource allocation in cloud computing using the uncertainty principle of game theory. *IEEE Systems Journal* 10, 2 (2014), 637–648.

[36] Shilpa V Pius, TS Shilpa, and Akkikavu Akkikavu. 2014. Survey on load balancing in cloud computing. In *International Conference on Computing, Communication and Energy Systems (ICCCES)*.

[37] S Salcedo-Sanz, J Del Ser, I Landa-Torres, S Gil-López, and JA Portilla-Figueras. 2014. The coral reefs optimization algorithm: a novel metaheuristic for efficiently solving optimization problems. *The Scientific World Journal* 2014 (2014).

[38] Yassir Samadi, Mostapha Zbakh, and Claude Tadonki. 2018. E-HEFT: enhancement heterogeneous earliest finish time algorithm for task scheduling based on load balancing in cloud computing. In *2018 International Conference on High Performance Computing & Simulation (HPCS)*. IEEE, 601–609.

[39] Bernard Ousmane Sane, Cheikh Saliou Mbacke Babou, Doudou Fall, and Ibrahima Niang. 2019. A Survey of Game Theoretic Solutions for Cloud Computing Security Issues. In *International Conference on Innovations and Interdisciplinary Solutions for Underserved Areas*. Springer, 1–12.

[40] Mohammad Shahrad and David Wentzlaff. 2016. Availability Knob: Flexible User-Defined Availability in the Cloud. In *Proceedings of the Seventh ACM Symposium on Cloud Computing* (Santa Clara, CA, USA) *(SoCC '16)*. Association for Computing Machinery, New York, NY, USA, 42–56. https://doi.org/10.1145/2987550.2987556

[41] H. Shen and Z. Li. 2014. New bandwidth sharing and pricing policies to achieve a win-win situation for cloud provider and tenants. In *IEEE INFOCOM 2014 - IEEE Conference on Computer Communications*. 835–843.

[42] Arunesh Sinha and Michael P. Wellman. 2019. Incentivizing Collaboration in a Competition. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems* (Montreal QC, Canada) *(AAMAS '19)*. International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, 556–564.

[43] S. Song, T. Lv, and X. Chen. 2014. A cooperative game method for load balancing in cloud based on cost-efficiency. In *2014 Sixth International Conference on Ubiquitous and Future Networks (ICUFN)*. 311–314. https://doi.org/10.1109/ICUFN.2014.6876803

[44] K. Sowmya and R. P. Sundarraj. 2012. Strategic Bidding for Cloud Resources under Dynamic Pricing Schemes. In *2012 International Symposium on Cloud and Services Computing*. 25–30. https://doi.org/10.1109/ISCOS.2012.28

[45] R Swathy, B Vinayagasundaram, G Rajesh, Anand Nayyar, Mohamed Abouhawwash, and Mohamed Abu Elsoud. 2020. Game theoretical approach for load balancing using SGMLB model in cloud environment. *Plos one* 15, 4 (2020), e0231708.

[46] Zheng Xiao, Zhao Tong, Kenli Li, and Keqin Li. 2017. Learning non-cooperative game for load balancing under self-interested distributed environment. *Applied Soft Computing* 52 (2017), 376–386.

[47] Xin Xu and Huiqun Yu. 2014. A game theory approach to fair and efficient resource allocation in cloud computing. *Mathematical Problems in Engineering* 2014 (2014).

[48] B. Yang, Z. Li, S. Chen, T. Wang, and K. Li. 2016. Stackelberg Game Approach for Energy-Aware Resource Allocation in Data Centers. *IEEE Transactions on Parallel and Distributed Systems* 27, 12 (2016), 3646–3658. https://doi.org/10.1109/TPDS.2016.2537809

[49] B. Yang, Z. Li, and S. Jiang. 2017. Cooperative Game Approach for Energy-Aware Load Balancing in Clouds. In *2017 IEEE International Symposium on Parallel and Distributed Processing with Applications and 2017 IEEE International Conference on Ubiquitous Computing and Communications (ISPA/IUCC)*. 9–16. https://doi.org/10.1109/ISPA/IUCC.2017.00012

[50] Jiachen Yang, Bin Jiang, Zhihan Lv, and Kim-Kwang Raymond Choo. 2020. A task scheduling algorithm considering game theory designed for energy management in cloud computing. *Future Generation computer systems* 105 (2020), 985–992.

[51] Lei Zhang and Jin-he Zhou. 2017. Task scheduling and resource allocation algorithm in cloud computing system based on non-cooperative game. In *2017 IEEE 2nd International Conference on Cloud Computing and Big Data Analysis (ICCCBDA)*. IEEE, 254–259.

[52] X. Zhao. 2010. An Enhanced Particle Swarm Optimization Algorithm with Passive Congregation. In *2010 International Conference on Machine Vision and Human-machine Interface*. 432–435. https://doi.org/10.1109/MVHI.2010.193

[53] Liang Zheng, Carlee Joe-Wong, Chee Wei Tan, Mung Chiang, and Xinyu Wang. 2015. How to Bid the Cloud. In *Proceedings of the 2015 ACM Conference on Special Interest Group on Data Communication* (London, United Kingdom) *(SIGCOMM '15)*. Association for Computing Machinery, New York, NY, USA, 71–84. https://doi.org/10.1145/2785956.2787473

[54] Zhengfa Zhu, Jun Peng, Kaiyang Liu, and Xiaoyong Zhang. 2020. A game-based resource pricing and allocation mechanism for profit maximization in cloud computing. *Soft Computing* 24, 6 (2020), 4191–4203.