Survey Paper
Algorithmic Game Theory

Pavan Kumar Appikatla
UFID: 51971306

Introduction

Algorithmic game theory is a field of study that examines the mathematical models used to analyze and optimize the behavior of autonomous agents playing games. This survey paper seeks to provide an overview of algorithmic game theory, its key concepts, and its applications. The paper will discuss topics such as how algorithms are used to solve games, how game theory is used to explain and predict the behavior of autonomous agents, and how game theory can be used for decision-making. It will also look at the implications of algorithmic game theory for the design of artificial intelligence, autonomous agents, and decision-making processes. Finally, the paper will explore the potential applications of algorithmic game theory in areas such as economics, artificial intelligence, and robotics.

Concepts of Solution

Most of the work in Algorithmic Game theory is motivated by networks like internet and auctions motivate much of the work in AGT.[1]

The research in Algorithmic Game Theory designs applications through optimization problems and searches for optimal solutions and upper and lower bounds on the guaranteed approximations. [1]

Designing Systems

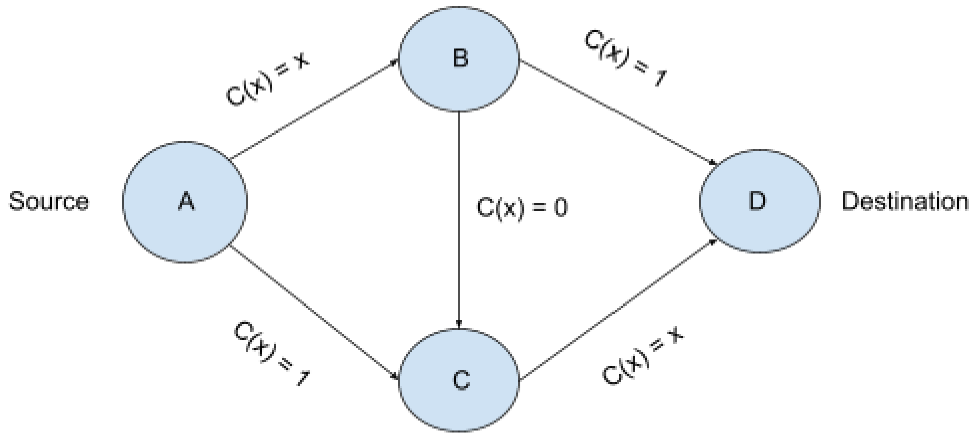
In designing systems we have participants who are both autonomous and strategic. They have their own interests. So, irrespective of the strategic behaviour of the participants, we need to design an algorithm that functions well.

The participants' goals may not be aligning with the goals of the designer. This creates an additional bit of complexity. In any game/contest/auction, the rules of the game are extremely important. Even in the case of strategic players, the designer never assumes that the players behave in a way away from their interests. Keeping in view of the strategic behaviour of the participants, the algorithm design must be taken care of. [1]

Mechanism Design

When is selfish behaviour not going to affect the design.

Braess's Paradox: Braess's paradox states that selfish behaviour by everyone in a contest/game doesn't always lead to optimum output for everybody.



Braces Network

Price of Anarchy: If the best possible system performance considering the strategic behaviour is taken as benchmark. In the Braess's network it is 4/3. The price of Anarchy value being 1 is the most desirable outcome of a strategic behaviour.
Anything closer to 1 is considered as the optimum output and can be stated as the system is more robust to selfish behaviour. [1] [10][11]

How do strategic play reach equilibrium

For example games like rock-paper-scissors, don't have a real deterministic state of equilibrium.

Nash Equilibrium

If each player in the game randomises the strategy uniformly, then they are said to achieve (mixed) Nash Equilibrium. Meaning, each player achieves optimal outcome without actually deviating from the initial strategy through which the game was started.

In fact, every game can have a state of Nash equilibrium.

Every bimatrix game, like rock-paper-scissors, has Nash Equilibrium.

Algorithmic Mechanism Design

The aim of Algorithmic Mechanism Design is to design and develop optimization algorithms for those problems where the input to such problems is unknown to the developer initially.

Such inputs are based on the interests of the participants. Best example to explain this is the process of auctioning.

The input data in this case is the interest of the participants of the bid to purchase the goods or service for a particular monetary value. This is the maximum value that is chosen by the bidder for the goods on auction.

The optimization algorithm takes care of the efficient handling of this process thereby maximising the overall outcome of the auction i.e, the maximum bid value is chosen as an outcome. Here the “mechanism” is the set of rules followed by the algorithm in getting the optimised solution to the problem.

Auction Theory

In establishing the complex dependency between the input data and the behaviour of the participants, an example of first-price auction can be considered. In this kind of bidding, every bidder submits a single bid in a sealed cover. A bidder who submits maximum value for the goods or services is declared as the winner. All the bidders in this kind of bidding submits a bid value lower than the maximum willingness to pay. The strategy is employed by bidders to keep the value of bid to the lowest possible subject to winning the bid. While preparing such bid values, every bidder needs to guess the value of other bidders so as to decrease the bid value to the lowest possible without losing the bid to the opponents. [7]

Suppose each bidder(i) wishes to submit a value of Vi, which is the maximum amount that the bidder is willing to pay for the goods. This is called private evaluation as the seller has no idea about what the Vi value is going to be and even the other bidders have no idea about what this value is going to be.

Utility Model:
For a bidder who loses the auction, the utility value is ‘0’.
For a bidder who wins, the utility value = Vi - price.

There is another auction process discussed, which is the **second-price auction**. In this process, the highest bidder will be declared the winner of the auction. However, the value of the bid that is considered is the second highest bid that is received. This process allows the bidders to cast their bidding with a freehand as the process itself takes care of lowering the value of such a bid to the next lower bid. Major e-commerce giants like ebay and Amazon follow the second-price auction for the majority of the auctions if not all.

Each bidder(i) has a dominant strategy that is to maximise the value of the good. The bidders are expected to bid the true values irrespective of what the other bidders are bidding. The contrast between first price bid and second price bid is that in the former, each bidder bids a value less than the actual valuation whereas in the latter, each bidder freely submits the actual valuation of the bid without being concerned about the other bidders.

Claim 1: The second-price auctions are independent of the bid values of the other bidders.

Proof:
1. Let V_i be the valuation of the goods and i be the successful bidder and the value of other bids be b-i.
2. To prove that the maximum utility value is achieved by bidding V_i.
3. Let X = Max (b_j) where j != i
4. In the second price auction, the utility of i is either ‘0’ (when b_i < B)
   Or V_i - B (when B <= b_i)
5. Case 1: When V_i < B, maximum possible utility is ‘0’ is achieved by $b_i = V_i$
6. Case 2: When V_i >= B, maximum possible utility is V_i - B which is achieved by $b_i = V_i$.
7. Thus from points 5 & 6, no matter what the strategy of the bidders is, the bid is going to maximise the utility.

Claim 2: In 2nd price bid, every truth telling bidder gets non negative utility.

The main emphasis here is to estimate the worst case approximations and the bounds on the complexities. This was first proposed by Nisan and Ronen [2].

“Incentive-compatible” efficient computation vs “classical” efficient computation:
The auctioning process consists of two parts. First part being the selection of the winner of the bid. This is called “Allocation Algorithm”. Second part takes care of charging the bids. This is called “Payment Algorithm”. The payment algorithm charges the unsuccessful bidders with ‘0’ and the successful bidder with the second highest bid value.

The argument here is that the auction is successful in the scenario: If all the bidders submits the true estimated value of the good or service. In this way, each bidder maximises the “net-value” of the good or service. Thus, based on the assumption that all the bidders submitted the bids truthfully, the second-price auction (Vickrey) [3].

The algorithm of allocation is considered implementable only if the allocation algorithm and payment algorithm when coupled judiciously yields maximum output for the auction.

In the case of single bid auction, the allocation algorithm is straightforward and is implementable whereas the second bid algorithm is not straightforward. From this it has been concluded that some of the algorithms are not implementable. This gives rise to a question, whether the algorithms that are implementable are less powerful than some algorithms that are arbitrary while solving optimization problems.

Worst-case equilibria:
The study of Koutsoupias et al. describes that the expected cost for every player i during Nash Equilibrium is always either less than or equal to 2 - [1/a] time of the optimal value. [4]

The basic idea under the proof is that always minimisation of the cost is tried by the agents.

$$Cost_i \leq tr_k / b + (b-1/b)\, tr_i$$

Shortest Path problem

Suppose the graph contains n edges representing the actors in question. Each agent has some secret data that makes up the edge weights. Here ti and I are her two examples of special edges. The shortest path is obtained when each agent honestly discloses its type or edge weight. If the resulting path is not the shortest, we can adjust the payment P so that the agent's payment pi is zero. [5]

Problems on minimum spanning trees can be solved using this. Optimising the calculation of the payment function is still an open problem.

Selfish Routing

Selfish routing is a strategy used in networks to optimize the cost of routing a packet from one node to another by choosing the path that minimizes the cost to the individual node, while disregarding the cost of other nodes in the network. It is a form of routing that does not consider the impact on other nodes in the network, and thus can lead to a suboptimal result for the entire network. Selfish routing is usually characterized by nodes making decisions based on their own interests, rather than considering the interests of other nodes or the entire network. This can lead to inefficient routing, increased traffic congestion, and increased resource utilization.

The algorithm understanding can be described as follows:

1. Initialize the network graph and weight matrix with the source and destination nodes.
2. Select a source node and calculate the shortest path between the source and destination.
3. Calculate the cost of each link in the path and determine the path with the lowest cost.
4. Calculate the cost of travelling on each edge in the network.
5. For each edge in the network, calculate the cost of travelling on it and add it to the total cost.
6. Select the path with the lowest total cost and ensure that there is no link congestion.
7. Calculate the cost of travelling on each edge in the path and add it to the total cost.
8. Adjust the weight of each link in the path according to the cost and the congestion.
9. Repeat steps 2-8 for each source and destination node in the network.
10. Output the lowest cost path from the source to the destination.

Recent Developments

Algorithmic game theory is an area of research that seeks to understand the interaction between computational strategies and game theory. The recent advances in this field have focused on developing algorithms for finding Nash equilibria in games, understanding the complexity of computing equilibria and strategies, and exploring the computational implications of different game structures.

One of the most exciting recent developments in algorithmic game theory is the development of efficient algorithms for computing Nash equilibria in games with multiple players. These algorithms employ a combination of linear programming, convex optimization, and machine learning techniques to compute approximate equilibria in games that are too complex for traditional methods. Recent work has also explored the implications of game complexity on the computational cost of computing strategies and equilibria.

This research has led to a better understanding of the complexity of computing Nash equilibria in games with multiple players, as well as the implications for the computational cost of computing strategies and equilibria in different game structures. In addition, recent work has looked into the application of algorithmic game theory to auction design. This research has explored the use of algorithms to design auctions that are more efficient and yield higher revenue.

This work has shown that auctions designed using algorithms can achieve better performance than traditional auction designs. Finally, recent research has also explored the use of algorithmic game theory to improve the design of online markets. This research has focused on developing algorithms that can automatically detect and respond to different types of market manipulation, as well as algorithms that can learn how to optimise market design.

-> Securing from Block chain attacks have been under study recently. An arbitrary attacker wants to maximise the utility function. [8]

-> Nash equilibrium has been utilised in the field of charging battery operated vehicles in order to maximise the output by taking minimal time for charging. This study considers various parameters like electric requirements, battery requirements and constraints that are hindering the output of the battery etc. [9]

Conclusion

In conclusion, algorithmic game theory has been an interesting and growing field of study over the years. Algorithmic game theory has provided a way to model and analyze the interactions between players in a game, as well as the strategies and outcomes of those interactions. We have seen a variety of applications for algorithmic game theory such as in artificial intelligence, multi-agent systems, and economics. As the field continues to develop, we can expect to see more applications of algorithmic game theory in the future paving a way to solve real world problems in the field of science, technology and medicine as well. Furthermore, algorithmic game theory has the potential to provide valuable insights into how to design better games and optimise strategies for players.

References:

[1] Algorithmic Game Theory∗ Tim Roughgarden† May 12, 2009

[2] N. Nisan and A. Ronen. Algorithmic mechanism design. Games and Economic Behavior, 35(1/2):166– 196, 2001.

[3] W. Vickrey. Counterspeculation, auctions, and competitive sealed tenders. Journal of Finance, 16(1):8– 37, 1961.

[4] Koutsoupias E, Papadimitriou C. Worst-case equilibria. Computer Science Review. 2009;3(2):65-69.

[5] Nisan N, Ronen A. Algorithmic mechanism design (extended abstract).

Proceedings of the thirty-first annual ACM symposium on Theory of computing - STOC '99. 1999;.

[6] Roughgarden T, Tardos É. How bad is selfish routing?. Journal of the ACM. 2002;49(2):236-259.

[7] Algorithmic Game Theory Edited by Noam Nisan, Hebrew University of Jerusalem, Tim Roughgarden, Stanford University, E ́ v a T a r d o s, Cornell University and Vijay V. Vazirani, Georgia Institute of Technology

[8] Dey S. Securing Majority-Attack in Blockchain Using Machine Learning and Algorithmic Game Theory: A Proof of Work. 2018 10th Computer Science and Electronic Engineering (CEEC). 2018;.

[9] Cao C, Chen B. Generalized Nash equilibrium problem based electric vehicle charging management in distribution networks. International Journal of Energy Research. 2018;42(15):4584-4596.

[10] Algorithmic Game Theory, Wikipedia.

[11] Papadimitriou, Christos (2001), "Algorithms, games, and the Internet", Proceedings of the 33rd ACM Symposium on Theory of Computing (STOC '01),