Chapter 8

Game Theory

"Game theory is a sort of umbrella or 'unified field' theory for the rational side of social science, where 'social' is interpreted broadly, to include human as well as non-human players (computers, animals, plants)."

– Robert Aumann, 1987

8.1 Introduction

In this chapter we look at a distributed system from a different perspective. Nodes no longer have a common goal, but are *selfish*. The nodes are not byzantine (actively malicious), instead they try to benefit from a distributed system – possibly without contributing.

Game theory attempts to mathematically capture behavior in strategic situations, in which an individual's success depends on the choices of others.

Remarks:

- Examples of potentially selfish behavior are file sharing or TCP. If a packet is dropped, then most TCP implementations interpret this as a congested network and alleviate the problem by reducing the speed at which packets are sent. What if a selfish TCP implementation will not reduce its speed, but instead transmit each packet twice?

- We start with one of the most famous games to introduce some definitions and concepts of game theory.

8.2 Prisoner's Dilemma

A team of two prisoners (players u and v) are being questioned by the police. They are both held in solitary confinement and cannot talk to each other. The prosecutors offer a bargain to each prisoner: snitch on the other prisoner to reduce your prison sentence.

u / v		Player u	
		Cooperate	Defect
Player v	Cooperate	1 1	0 3
	Defect	3 0	2 2

Table 8.1: The prisoner's dilemma game as a matrix.

- If both of them stay silent (*cooperate*), both will be sentenced to one year of prison on a lesser charge.
- If both of them testify against their fellow prisoner (*defect*), the police has a stronger case and they will be sentenced to two years each.
- If player u defects and the player v cooperates, then player u will go free (snitching pays off) and player v will have to go to jail for three years; and vice versa.
- This two player game can be represented as a matrix, see Table 8.1.

Definition 8.2 (game)**.** *A game requires at least two rational players, and each player can choose from at least two options (**strategies**). In every possible outcome (**strategy profile**) each player gets a certain payoff (or cost). The payoff of a player depends on the strategies of the other players.*

Definition 8.3 (social optimum)**.** *A strategy profile is called social optimum (SO) if and only if it minimizes the sum of all costs (or maximizes payoff).*

Remarks:

- The social optimum for the prisoner's dilemma is when both players cooperate – the corresponding cost sum is 2.

Definition 8.4 (dominant)**.** *A strategy is dominant if a player is never worse off by playing this strategy. A dominant strategy profile is a strategy profile in which each player plays a dominant strategy.*

Remarks:

- The dominant strategy profile in the prisoner's dilemma is when both players defect – the corresponding cost sum is 4.

Definition 8.5 (Nash Equilibrium)**.** *A Nash Equilibrium (NE) is a strategy profile in which no player can improve by unilaterally (the strategies of the other players do not change) changing its strategy.*

Remarks:

- A game can have multiple Nash Equilibria.
- In the prisoner's dilemma both players defecting is the only Nash Equilibrium.
- If every player plays a dominant strategy, then this is by definition a Nash Equilibrium.
- Nash Equilibria and dominant strategy profiles are so called solution concepts. They are used to analyze a game. There are more solution concepts, e.g. correlated equilibria or best response.
- The best response is the best strategy given a belief about the strategy of the other players. In this game the best response to both strategies of the other player is to defect. If one strategy is the best response to any strategy of the other players, it is a dominant strategy.
- If two players play the prisoner's dilemma repeatedly, it is called iterated prisoner's dilemma. It is a dominant strategy to always defect. To see this, consider the final game. Defecting is a dominant strategy. Thus, it is fixed what both players do in the last game. Now the penultimate game is the last game and by induction always defecting is a dominant strategy.
- Game theorists were invited to come up with a strategy for 200 iterations of the prisoner's dilemma to compete in a tournament. Each strategy had to play against every other strategy and accumulated points throughout the tournament. The simple Tit4Tat strategy (cooperate in the first game, then copy whatever the other player did in the previous game) won. One year later, after analyzing each strategy, another tournament (with new strategies) was held. Tit4Tat won again.
- We now look at a distributed system game.

8.3 Selfish Caching

Computers in a network want to access a file regularly. Each node $v \in V$, with V being the set of nodes and $n = |V|$, has a demand d_v for the file and wants to minimize the cost for accessing it. In order to access the file, node v can either cache the file locally which costs 1 or request the file from another node u which costs $c_{v \leftarrow u}$. If a node does not cache the file, the cost it incurs is the minimal cost to access the file remotely. Note that if no node caches the file, then every node incurs cost ∞. There is an example in Figure 8.6.

Remarks:

- We will sometimes depict this game as a graph. The cost $c_{v \leftarrow u}$ for node v to access the file from node u is equivalent to the length of the shortest path times the demand d_v.

- Note that in undirected graphs $c_{u \leftarrow v} > c_{v \leftarrow u}$ if and only if $d_u > d_v$. We assume that the graphs are undirected for the rest of the chapter.

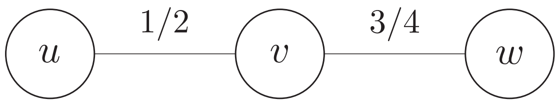


Figure 8.6: In this example we assume $d_u = d_v = d_w = 1$. Either the nodes u and w cache the file. Then neither of the three nodes has an incentive to change its behavior. The costs are 1, 1/2, and 1 for the nodes u, v, w, respectively. Alternatively, only node v caches the file. Again, neither of the three nodes has an incentive to change its behavior. The costs are 1/2, 1, and 3/4 for the nodes u, v, w, respectively.

Algorithm 8.7 Nash Equilibrium for Selfish Caching

$S = \{\}$ //set of nodes that cache the file
repeat
  Let v be a node with maximum demand d_v in set V
  $S = S \cup \{v\}, V = V \setminus \{v\}$
  Remove every node u from V with $c_{u \leftarrow v} < 1$
until V = {}

Theorem 8.8. *Algorithm 8.7 computes a Nash Equilibrium for Selfish Caching.*

Proof. Let u be a node that is not caching the file. Then there exists a node v for which $c_{u \leftarrow v} \leq 1$. Hence, node u has no incentive to cache.

Let u be a node that is caching the file. We now consider any other node v that is also caching the file. First, we consider the case where v cached the file before u did. Then it holds that $c_{u \leftarrow v} > 1$ by construction.

It could also be that v started caching the file after u did. Then it holds that $d_u \geq d_v$ and therefore $c_{u \leftarrow v} \geq c_{v \leftarrow u}$. Furthermore, we have $c_{v \leftarrow u} > 1$ by construction. Combining these implies that $c_{u \leftarrow v} \geq c_{v \leftarrow u} > 1$.

In either case, node u has no incentive to stop caching. □

Definition 8.9 (Price of Anarchy)**.** *Let NE_- denote the Nash Equilibrium with the highest cost (smallest payoff). The **Price of Anarchy** (PoA) is defined as*

$$PoA = \frac{\text{cost}(NE_-)}{\text{cost}(SO)}.$$

Definition 8.10 (Optimistic Price of Anarchy)**.** *Let NE_+ denote the Nash Equilibrium with the smallest cost (highest payoff). The **Optimistic Price of Anarchy** (OPoA) is defined as*

$$OPoA = \frac{\text{cost}(NE_+)}{\text{cost}(SO)}.$$

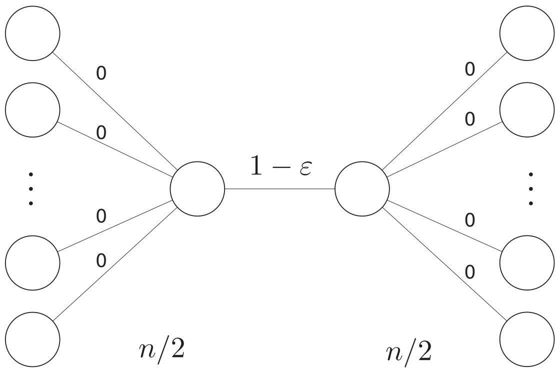


Figure 8.12: A network with a Price of Anarchy of $\Theta(n)$.

Remarks:

- The Price of Anarchy measures how much a distributed system degrades because of selfish nodes.
- We have $PoA \geq OPoA \geq 1$.

Theorem 8.11. *The (Optimistic) Price of Anarchy of Selfish Caching can be* $\Theta(n)$.

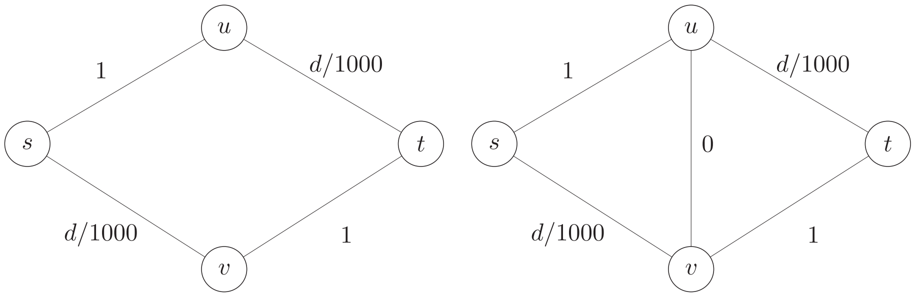
Proof. Consider a network as depicted in Figure 8.12. Every node v has demand $d_v = 1$. Note that if any node caches the file, no other node has an incentive to cache the file as well since the cost to access the file is at most $1-\varepsilon$. Wlog let us assume that a node v on the left caches the file, then it is cheaper for every node on the right to access the file remotely. Hence, the total cost of this solution is $1 + \frac{n}{2} \cdot (1-\varepsilon)$. In the social optimum one node from the left and one node from the right cache the file. This reduces the cost to 2. Hence, the Price of Anarchy is $\frac{1+\frac{n}{2}\cdot(1-\varepsilon)}{2} \underset{\varepsilon \to 0}{=} \frac{1}{2} + \frac{n}{4} = \Theta(n)$. □

8.4 Braess' Paradox

Consider the graph in Figure 8.13, it models a road network. Let us assume that there are 1000 drivers (each in their own car) that want to travel from node s to node t. Traveling along the road from s to u (or v to t) always takes 1 hour. The travel time from s to v (or u to t) depends on the traffic and increases by 1/1000 of an hour per car, i.e., when there are 500 cars driving, it takes 30 minutes to use this road.

Lemma 8.14. *Adding a super fast road (delay is 0) between u and v can increase the travel time from s to t.*

Proof. Since the drivers act rationally, they want to minimize the travel time. In the Nash Equilibrium, 500 drivers first drive to node u and then to t and 500 drivers first to node v and then to t. The travel time for each driver is 1 + 500 / 1000 = 1.5.



(a) The road network without the shortcut (b) The road network with the shortcut

Figure 8.13: Braess' Paradox, where d denotes the number of drivers using an edge.

To reduce congestion, a super fast road (delay is 0) is built between nodes u and v. This results in the following Nash Equilibrium: every driver now drives from s to v to u to t. The total cost is now $2 > 1.5$. □

Remarks:

- There are physical systems which exhibit similar properties. Some famous ones employ a spring. YouTube has some fascinating videos about this. Simply search for "Braess Paradox Spring".
- We will now look at another famous game that will allow us to deepen our understanding of game theory.

8.5 Rock-Paper-Scissors

There are two players, u and v. Each player simultaneously chooses one of three options: rock, paper, or scissors. The rules are simple: paper beats rock, rock beats scissors, and scissors beat paper. A matrix representation of this game is in Table 8.15.

u v		Player u		
		Rock	Paper	Scissors
Player v	Rock	0 0	1 -1	-1 1
	Paper	-1 1	0 0	1 -1
	Scissors	1 -1	-1 1	0 0

Table 8.15: Rock-Paper-Scissors as a matrix.

Remarks:

- None of the three strategies is a Nash Equilibrium. Whatever player u chooses, player v can always switch her strategy such that she wins.
- This is highlighted in the best response concept. The best response to e.g. scissors is to play rock. The other player switches to paper. And so on.
- Is this a game without a Nash Equilibrium? John Nash answered this question in 1950. By choosing each strategy with a certain probability, we can obtain a so called mixed Nash Equilibrium. Indeed:

Theorem 8.16. *Every game has a mixed Nash Equilibrium.*

Remarks:

- The Nash Equilibrium of this game is if both players choose each strategy with probability 1/3. The expected payoff is 0.
- Any strategy (or mix of them) is a best response to a player choosing each strategy with probability 1/3.
- In a pure Nash Equilibrium, the strategies are chosen deterministically. Rock-Paper-Scissors does not have a pure Nash Equilibrium.
- Unfortunately, game theory does not always model problems accurately. Many real world problems are too complex to be captured by a game. And as you may know, humans (not only politicians) are often not rational.
- In distributed systems, players can be servers, routers, etc. Game theory can tell us whether systems and protocols are prone to selfish behavior.

8.6 Mechanism Design

Whereas game theory analyzes existing systems, there is a related area that focuses on designing games – mechanism design. The task is to create a game where nodes have an incentive to behave "nicely".

Definition 8.17 (auction)**.** *One good is sold to a group of bidders in an auction. Each bidder v_i has a secret value z_i for the good and tells his bid b_i to the auctioneer. The auctioneer sells the good to one bidder for a price p.*

Remarks:

- For simplicity, we assume that no two bids are the same, and that $b_1 > b_2 > b_3 > \dots$

Definition 8.19 (truthful)**.** *An auction is truthful if no player v_i can gain anything by not stating the truth, i.e., $b_i = z_i$.*

Algorithm 8.18 First Price Auction

1: every bidder v_i submits his bid b_i
2: the good is allocated to the highest bidder v_1 for the price $p = b_1$

Theorem 8.20. *A First Price Auction (Algorithm 8.18) is not truthful.*

Proof. Consider an auction with two bidders, with bids b_1 and b_2. By not stating the truth and decreasing his bid to $b_1 - \varepsilon > b_2$, player one could pay less and thus gain more. Thus, the first price auction is not truthful. □

Algorithm 8.21 Second Price Auction

1: every bidder v_i submits his bid b_i
2: the good is allocated to the highest bidder v_1 for $p = b_2$

Theorem 8.22. *Truthful bidding is a dominant strategy in a Second Price Auction.*

Proof. Let z_i be the truthful value of node v_i and b_i his bid. Let $b_{\max} = \max_{j \neq i} b_j$ is the largest bid from other nodes but v_i. The payoff for node v_i is $z_i - b_{\max}$ if $b_i > b_{\max}$ and 0 else. Let us consider overbidding first, i.e., $b_i > z_i$:

- If $b_{\max} < z_i < b_i$, then both strategies win and yield the same payoff $(z_i - b_{\max})$.
- If $z_i < b_i < b_{\max}$, then both strategies lose and yield a payoff of 0.
- If $z_i < b_{\max} < b_i$, then overbidding wins the auction, but the payoff $(z_i - b_{\max})$ is negative. Truthful bidding loses and yields a payoff of 0.

Likewise underbidding, i.e. $b_i < z_i$:

- If $b_{\max} < b_i < z_i$, then both strategies win and yield the same payoff $(z_i - b_{\max})$.
- If $b_i < z_i < b_{\max}$, then both strategies lose and yield a payoff of 0.
- If $b_i < b_{\max} < z_i$, then truthful bidding wins and yields a positive payoff $(z_i - b_{\max})$. Underbidding loses and yields a payoff of 0.

Hence, truthful bidding is a dominant strategy for each node v_i. □

Remarks:

- Let us use this for Selfish Caching. We need to choose a node that is the first to cache the file. But how? By holding an auction. Every node says for which price it is willing to cache the file. We pay the node with the lowest offer and pay it the second lowest offer to ensure truthful offers.
- Since a mechanism designer can manipulate incentives, she can implement a strategy profile by making all the strategies in this profile dominant.

Theorem 8.23. *Any Nash Equilibrium of Selfish Caching can be implemented for free.*

Proof. If the mechanism designer wants the nodes from the caching set S of the Nash Equilibrium to cache, then she can offer the following deal to every node not in S: If not every node from set S caches the file, then I will ensure a positive payoff for you. Thus, all nodes not in S prefer not to cache since this is a dominant strategy for them. Consider now a node $v \in S$. Since S is a Nash Equilibrium, node v incurs cost of at least 1 if it does not cache the file. For nodes that incur cost of exactly 1, the mechanism designer can even issue a penatly if the node does not cache the file. Thus, every node $v \in S$ caches the file. □

Remarks:

- Mechanism design assumes that the players act rationally and want to maximize their payoff. In real-world distributed systems some players may be not selfish, but actively malicious (byzantine).
- What about P2P file sharing? To increase the overall experience, BitTorrent suggests that peers offer better upload speed to peers who upload more. This idea can be exploited. By always claiming to have nothing to trade yet, the BitThief client downloads without uploading. In addition to that, it connects to more peers than the standard client to increase its download speed.
- Many techniques have been proposed to limit such free riding behavior, e.g., tit-for-tat trading: I will only share something with you if you share something with me. To solve the bootstrap problem ("I don't have anything yet"), nodes receive files or pieces of files whose hash match their own hash for free. One can also imagine indirect trading. Peer u uploads to peer v, who uploads to peer w, who uploads to peer u. Finally, one could imagine using virtual currencies or a reputation system (a history of who uploaded what). Reputation systems suffer from collusion and Sybil attacks. If one node pretends to be many nodes who rate each other well, it will have a good reputation.

Chapter Notes

Game theory was started by a proof for mixed-strategy equilibria in two-person zero-sum games by John von Neumann [Neu28]. Later, von Neumann and Morgenstern introduced game theory to a wider audience [NM44]. In 1950 John Nash proved that every game has a mixed Nash Equilibrium [Nas50]. The Prisoner's Dilemma was first formalized by Flood and Dresher [Flo52]. The iterated prisoner's dilemma tournament was organized by Robert Axelrod [AH81]. The Price of Anarchy definition is from Koutsoupias and Papadimitriou [KP99]. This allowed the creation of the Selfish Caching Game [CCW$^+$04], which we used as a running example in this chapter. Braess' paradox was discovered by Dietrich Braess in 1968 [Bra68]. A generalized version of the second-price auction is the VCG auction, named after three successive papers from first Vickrey, then Clarke, and finally Groves [Vic61, Cla71, Gro73]. One popular example of selfishness in practice is BitThief – a BitTorrent client that successfully downloads without uploading [LMSW06]. Using game theory economists try to understand markets and predict crashes. Apart from John Nash, the Sveriges Riksbank Prize (Nobel Prize) in Economics has been awarded many times to game theorists. For example in 2007 Hurwicz, Maskin, and Myerson received the prize for "for having laid the foundations of mechanism design theory".

This chapter was written in collaboration with Philipp Brandes.

Bibliography

[AH81] Robert Axelrod and William Donald Hamilton. The evolution of cooperation. *Science*, 211(4489):1390–1396, 1981.

[Bra68] Dietrich Braess. Über ein paradoxon aus der verkehrsplanung. *Unternehmensforschung*, 12(1):258–268, 1968.

[CCW+04] Byung-Gon Chun, Kamalika Chaudhuri, Hoeteck Wee, Marco Barreno, Christos H Papadimitriou, and John Kubiatowicz. Selfish caching in distributed systems: a game-theoretic analysis. In *Proceedings of the twenty-third annual ACM symposium on Principles of distributed computing*, pages 21–30. ACM, 2004.

[Cla71] Edward H Clarke. Multipart pricing of public goods. *Public choice*, 11(1):17–33, 1971.

[Flo52] Merrill M Flood. Some experimental games. *Management Science*, 5(1):5–26, 1952.

[Gro73] Theodore Groves. Incentives in teams. *Econometrica: Journal of the Econometric Society*, pages 617–631, 1973.

[KP99] Elias Koutsoupias and Christos Papadimitriou. Worst-case equilibria. In *STACS 99*, pages 404–413. Springer, 1999.

[LMSW06] Thomas Locher, Patrick Moor, Stefan Schmid, and Roger Wattenhofer. Free Riding in BitTorrent is Cheap. In *5th Workshop on Hot Topics in Networks (HotNets), Irvine, California, USA*, November 2006.

[Nas50] John F. Nash. Equilibrium points in n-person games. *Proc. Nat. Acad. Sci. USA*, 36(1):48–49, 1950.

[Neu28] John von Neumann. Zur Theorie der Gesellschaftsspiele. *Mathematische Annalen*, 100(1):295–320, 1928.

[NM44] John von Neumann and Oskar Morgenstern. *Theory of games and economic behavior*. Princeton university press, 1944.

[Vic61] William Vickrey. Counterspeculation, auctions, and competitive sealed tenders. *The Journal of finance*, 16(1):8–37, 1961.