CS295 Introduction to Algorithmic Game Theory

Instructor: Ioannis Panageas Scribed by: Saeed, Farhan and Chakraborty, Vishal

Lecture 7. Price of Anarchy / Lecture 8. Complexity Classes (half)

1 Introduction

1.1 Definition of Price of Anarchy

Price of Anarchy can be described as the performance of a system when agents choose the worst decision. This can be described in the formula shown in the introduction lecture:

$$PoA = \frac{\text{Performance of worst case Nash Equilibrium}}{\text{Optimal performance if the agents don't decide on their own}}$$

The price of anarchy is basically a percentage of how worse a system can get at it's worst Nash equilibrium. It is based off of Braess's paradox in which adding more roads may not improve flow in certain scenarios.

2 Price of Anarchy between different routes

2.1 Finding Nash equilibrium between two routes

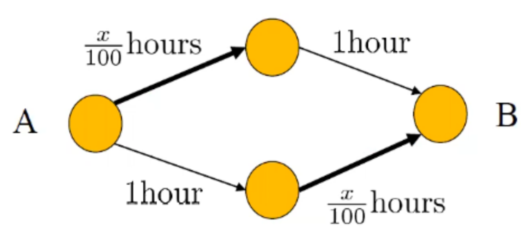

Figure 1: Truck route from A to B

Suppose 100 drivers commute from point A to B. There is a difference between the two paths where going up would give you $\frac{x}{100}$ hours of travel time on the first node whereas the bottom middle node has a travel time that is a constant of 1 hour. Between these middle nodes, they switch in the time it takes from the parent node A. The nash equilibrium that can be achieved in this case is 50 drivers go on top and 50 drivers go on the bottom route. This brings us a 1.5 hour travel time as our Nash equilibrium.

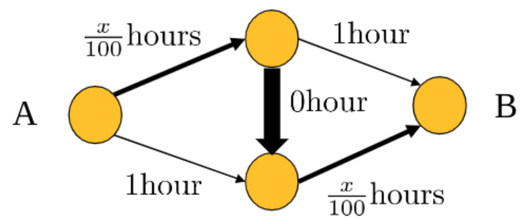

Figure 2: Truck route from A to B with a portal to take you from one route to the other

2.2 A magical portal appears

Adding a fast link to a problem is not always the solution. This is the case when we add a portal that takes you from one route to the other. We end up with a worst case Nash Equilibrium of everyone going to the top route initially, then going to the bottom. We end up using more time due to everyone going the same route! This is an example of Braess's paradox, where adding more roads doesn't always benefit the agents.

In this case, the performance of the worst case Nash Equilibrium is 2 hours. The optimal performance if the agents don't get to choose what routes to take on their own is 1.5 hours. When we plug these values into the Price of Anarchy equation that was introduced last section, we will get $\frac{4}{3}$.

3 Non-atomic selfish routing

3.1 A simple example

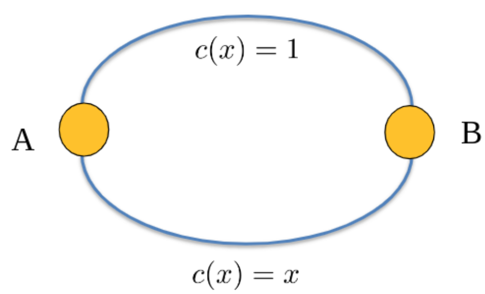

Figure 3: Pigou network graph

In this example, we have a graph that is called a pigou network, and you can take either vertices to go to B. One vertex has a constant cost of 1 while the other has a cost of x. The optimized cost is for half the population to go to the bottom and top vertices, which gives us $\frac{3}{4}$. The worst case Nash equilibrium is the cost of 1 if 100% use the top route. The Price of Anarchy in this situation is the same as the last example, $\frac{4}{3}$.

3.2 Why is there a pattern?

This is what a non-atomic selfish routing game is! It is defined as a graph $G(v, e)$ with source destination pairs $(s_1, t_1), ..., (s_k, t_k)$. We route traffic r_i from $s_i \rightarrow t_i$. The cost of each edge e is continuous and non-decreasing, $c_e(.) \geq 0$. Flow is an equilibrium in the game if all traffic is routed to the cheapest path.

$$\text{Cost of path: } c_p(f) = \sum_{e \in p} c_e(f)$$

$$\text{Social cost} := \sum_p f_p c_p(f)$$

The cost of path is simply the cost of an edge between two vertices while Social cost is the sum of all the path costs. Note: Equilibrium flow exists and is unique!

Definition 3.1 *Linear costs*
Linear costs are of the form $c_e(x) = a_e * x + b_e$.

This definition applies to all Nash costs. This definition will apply to the following theorem:

Theorem 3.1 *Roughgarden-Tardos 00', PoA for linear costs*
For every network with linear costs:

$$\textit{Cost of Nash Flow} \leq \frac{4}{3} * \textit{Cost of Optimal Flow}$$

No matter how the graph looks like or the number of sources or sinks, the cost of the Nash flow is at most $\frac{4}{3}$ multiplied by the cost of optimal flow.

Proof: Let us prove the previous theorem. Let f^* be a Nash flow and f be another flow. We first show and prove the Variational Inequality:

$$\sum_e f^* c_e(f_e^*) \leq \sum_e f_e c_e(f_e^*)$$

First, observe that f^* is an equilibrium flow. The strategies are the paths. So, if a path p has a flow positive, then people are choosing the path. Otherwise the flow is deviated.

$$f^* \implies \text{if } f_p^* > 0 \text{ then } c_p(f^*) \leq c_{p'}(f^*) \text{ for all } p^{'}$$

The Nash equilibrium flow has a property that every flow you're using should effectively have the same cost. It is an analog that every strategy you play should give you the same payoff in a normal game play set. Therefore all paths p that have a positive flow, $f_p^* > 0$, will have the same cost L. Hence:

$$\sum_p f_p^* c_p(f^*) = L * F \text{ where } F = \sum_p f_p^* \text{ is the total flow}$$

F is defined as the total flow, which can be thought of as population/agent that goes through the flow.

$$c_p \geq L$$

For every path, the cost of a path in the Nash flow is at least L.
We can conclude that:

$$\sum_p f_p c_p(f^*) \geq L \sum_p f_p = L * F$$

the sum of all path multiplied by the flow of the path will be at least the Nash Flow multiplied by the total flow. To put it all together:

$$\sum_e f_e c_e(f_e^*) = \sum_p f_p c_p(f^*) \geq L * F = \sum_p f_p^* c_p(f^*) = \sum_e f_e^* c_e(f^*)$$

Now that we have proved the inequality, let us prove Theorem 3.1.
We get that:

$$\sum_e f^* c_e(f^*) \leq \sum_e f_e c_e(f) + \sum_e f_e(c_e(f^*) - c_e(f))$$

Let f be the minimum flow. From this inequality, the right hand side is the same. The left hand side we need to change the f^* to f. We also have that:

$$\sum_e f_e(c_e(f^*) - c_e(f)) \leq \frac{1}{4} \sum_e f_e^* c_e(f^*)$$

The right hand side of the inequality we started with is bounded to the right hand side from the same inequality times $\frac{1}{4}$. Now we're going to use that we have linear costs.

- Case where $c_e(f^*) < c_e(f)$ which trivially is the last equation.
- The opposite case where $c_e(f^*) \geq c_e(f) \implies f_e^* \geq f_e$. Linear costs implies that, $LHS = a_e f_e(f_e^* - f_e)$ and RHS $\geq \frac{1}{4} a_e {f_e^*}^2$. In other words, we are comparing two numbers: $(f_e^* - f_e)$ and ${f_e^*}^2$.

Because $xy - y^2 \leq \frac{x^2}{4}$ it implies that LHS $\leq$ RHS.
We can conclude that:

$$\sum_e f^* c_e(f^*) \leq \sum_e f_e c_e(f) + \sum_e f^* c_e(f^*)$$

is equivalent to:

$$\sum_e f^* c_e(f^*) \leq \frac{4}{3} \sum_e f_e c_e(f)$$

■

Theorem 3.2 *Roughgarden-Tardos 02', PoA for polynomial costs*
For every network with polynomial costs with degree d:

$$\textit{Cost of Nash Flow} \leq \theta(\frac{d}{\log d}) * \textit{Cost of Optimal Flow}$$

Pigou is tight as we have shown in two previous examples.

4 Price of Anarchy in Congestion Games

4.1 What is a congestion game?

A congestion game is a game with n sets of players and E sets of edges, facilities and bins. Each player i has set of strategies $S_i \subset 2^E$. Every cost function of edge e is defined as $c_e : \{1, 2, ..., n\} \rightarrow \mathbb{R}^+$.

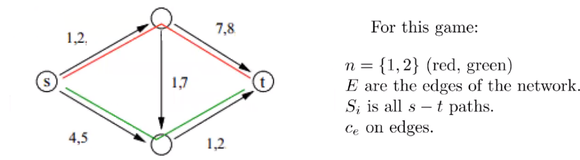

Figure 4: Example of a congestion game

Theorem 4.1 *Christodoulou-Koutsoupias, PoA for linear costs*
For every network with linear costs:

$$\textit{Cost of Nash Flow} \leq \frac{5}{2} * \textit{Cost of Optimal welfare}$$

The proof with this theorem is similar to the last one where we will start with a variational inequality.
Proof: Let l^* be a Nash equilibrium in which i uses P_i and assume i deviates to $\tilde{P}_i$. It holds:

$$\sum_{e \in P_i} c_e(l_e^*) \leq \sum_{e \in P_i \cap \tilde{P}_i} c_e(l_e^*) + \sum_{e \in \tilde{P}_i \setminus P_- i} c_e(l_e^* + 1)$$

Because the right hand side is non-decreasing, we can set the right side to:

$$\leq \sum_{e \in P_i \cap \tilde{P}_i} c_e(l_e^* + 1) + \sum_{e \in \tilde{P}_i \setminus P_- i} c_e(l_e^* + 1) = \sum_{e \in \tilde{P}_i} c_e(l_e^* + 1)$$

Here we bounded the cost of the player i at the Nash with the cost of the player i if they deviate. Consider any configuration $\tilde{l}$ where each agent j uses path $\tilde{P}_j$. Summing for all agents i:

$$\sum_{i \in [n]} \sum_{e \in P_i} c_e(l_e^*) \leq \sum_{i \in [n]} \sum_{e \in \tilde{P}_i} c_e(l_e^* + 1)$$

This inequality is similar to the last inequality. Now, we need to change this l^* to the social cost of $\tilde{l}$. We can remove the first summation by multiplying the cost of every edge by the number of people using this edge, which is the load of the edge $\tilde{l}_e$..

$$= \sum_{e \in \tilde{P}_i} \tilde{l}_e c_e(l_e^* + 1)$$

Now we need to break the sum by using the fact that the cost function is linear.

$$\sum_{e \in \tilde{P}_i} a_e \tilde{l}_e (l_e^* + 1) + b_e \tilde{l}_e$$

Because $y(z+1) \leq \frac{5}{3}y^2 + \frac{1}{3}z^2$ for natural y, z, we can say that:

$$\leq \sum_{e \in \tilde{P}_i} a_e (\frac{5}{3}\tilde{l}_e^2 + \frac{1}{3}{l_e^*}^2) + b_e \tilde{l}_e$$

Observe that:

$$\frac{5}{3}C(\tilde{l}) = \frac{5}{3}\sum_{i \in [n]}\sum_{e \in \tilde{P}_i} c_e(l_e^*) = \sum_{e \in \tilde{P}_i} \frac{5}{3}a_e \tilde{l}_e^2 + \frac{5}{3}b_e \tilde{l}_e \geq \sum_{e \in \tilde{P}_i} \frac{5}{3}a_e \tilde{l}_e^2 + b_e \tilde{l}_e$$

Therefore:

$$C(l^*) \leq \frac{5}{3}C(\tilde{l}) + \frac{1}{3}\sum_e a_e {l_e^*}^2$$

$$\leq \frac{5}{3}C(\tilde{l}) + \frac{1}{3}C(l^*)$$

$$C(l^*) \leq \frac{5}{2}C(\tilde{l})$$

This is tightly bounded and applies for polynomial cost functions that the PoA is exponential in d.

■

5 Price of Anarchy and Balls & Bins

5.1 Definition

Consider a set of n balls and n bins $e_1, ..., e_n$. Each ball i chooses a bin j and pays the load of the bin j. We define the social cost as the maximum load, or the maximum balls that can be in the bin.

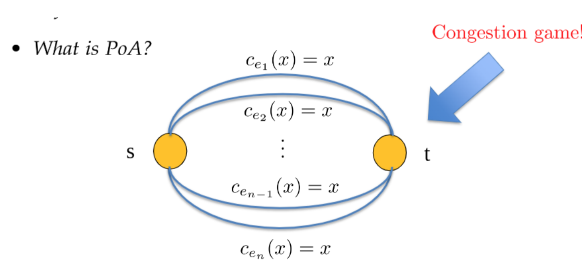

Figure 5: Balls and Bins visually

Theorem 5.1 *Koutsoupias-Papadimitriou, PoA for Balls & Bins*
$PoA = \Omega(\frac{\ln n}{\ln \ln n})$

Proof: We will use the second moment method. First, we set every ball in a different bin. Hence, the optimal social cost is 1. The Nash equilibrium ends up to be $(\frac{1}{n}, ..., \frac{1}{n})$ which is symmetric. With high probability, we show that uniform gives max load of $\Omega(\frac{\ln n}{\ln \ln n})$ which implies that the expected max load is $\Omega(\frac{\ln n}{\ln \ln n})$.

- Claim 1: Bin i has $k << n$ balls with probability at least:

$$\binom{n}{k}\frac{1}{n^k}(1-\frac{1}{n})^{n-k} \geq \frac{1}{n^k}\binom{n}{k}^k \frac{1}{e} = \frac{1}{ek^k}$$

Choosing $k = \frac{\ln n}{3 \ln \ln n}$ we have $k^k \leq (\ln n)^k = (\ln n)^{\frac{\ln n}{3 \ln \ln n}} = n^{\frac{1}{3}}$ Claim 1: Thus bin i has at least $\frac{\ln n}{3 \ln \ln n}$ balls with probability at least $\frac{1}{en^1/3}$. Let X_i be the indicator that bin i has at least $\frac{\ln n}{3 \ln \ln n}$ balls and X will be the expected as the number of all bins with at least $\frac{\ln n}{3 \ln \ln n}$ balls.

$$X = \sum_i X_i \Rightarrow E[X] = \sum_i E[X_i]$$

Observe that $E[X] \geq \frac{n^{\frac{2}{3}}}{e} >> 1$ but this does not imply $X \geq 1$ with high probability. We need to argue about the variance. Chebyshev's inequality gives us $\mathbf{Pr}[|X - E[X]| \geq tE[X]] \leq \frac{\text{Var}[X]}{t^2 E^2[X]}$, thus $\mathbf{Pr}[X = 0] \leq \mathbf{Pr}[|X - E[X]| \geq tE[X]] \leq \frac{\text{Var}[X]}{t^2 E^2[X]}$

$$\mathbf{Pr}[X = 0] \leq \mathbf{Pr}[|X - E[X]| \geq tE[X]]$$

From negative correlation we have that $\text{Var}[X] \leq \sum_i \text{Var}[X_i]$. Moreover $\text{Var}[X_i] = E[X_i^2] - E^2[X_i] \leq E[X_i^2] = E[X_i] \leq 1$. We can conclude that $\mathbf{Pr}[X = 0] \leq \frac{n}{e^2 n^{\frac{4}{3}}} = \frac{n^{-\frac{1}{3}}}{e^2}$
Therefore:

$$\mathbf{Pr}[X \geq 1] = 1 - \mathbf{Pr}[X = 0] \geq 1 - \frac{n^{-\frac{1}{3}}}{e^2} \to 1$$

■

End Lecture 7 notes

6 Section 8 start

7 Standard Complexity Classes

P (polynomial) is a set of decision problems for which some algorithm can provide an answer for in polynomial time. NP (non deterministic polynomial) is a set of all decision problems where if the answer is "yes", then we can verify that the answer is correct in polynomial time. co-NP is similar to NP but we can verify if the answer "no" is correct in polynomial time instead of "yes".

7.1 Problems in P

Examples of problems in P include:

- Decision version of shortest path (yes or no)
- Decision version of finding the maximum number in a list (yes or no)
- is n a prime number (yes or no)

7.2 Problems in NP

Problems in NP include problems where ”yes” can be verified in polynomial time. One famous problem in NP is called the travelling salesman problem (TSP), where we are given a weighted graph and have to find the shortest route that visits every vertex once and returns to where we started.

7.3 Equilibrium Computation

Computing Nash Equilibrium is not a decision problem. The goal is to find a function that maps games to mixed strategies. An example for this is to add two numbers and find an outcome.

8 Function Complexity Classes

FP is the set of function problems for which some algorithm can provide an answer in polynomial time. FNP is the set of all function problems where the validity of an input, output pair can be verified in polynomial time with an algorithm of some kind. TFNP is the subclass of FNP where existence of a solution is guaranteed to exist for every input.