Algorithms, Games, and Networks	January 29, 2013

Lecture 5

Lecturer: Avrim Blum — *Scribe: Yu Zhao*

1 Overview

Today's topic is the **price of anarchy**, **potential functions** and **congestion games**. The corresponding readings are Sections 17, 19.3 in the AGT book.

2 Recap of Lemke-Howson algorithm

Given $n \times n$ symmetric game A, we want to find a symmetric Nash equilibrium. Consider $2n$ linear constraints on n variables:

- $A_i z \leq 1$ for all i
- $z_j \geq 0$ for all j

We call strategy i is "represented" if $A_i z = 1$ or $z_i = 0$.

The Lemke-Howson algorithm just start at $(0, \ldots, 0)$, move along the edge (relax one of $z_j = 0$ and move until hit some $A_i z = 1$), and then repeat taking strategy represented twice and relax constraint you didn't just hit until all strategies are represented.

Claim 1 *Lemke-Howson algorithm will end in finite steps at a symmetric Nash equilibrium.*

Proof: All states except the start state $(0, \ldots, 0)$ and the final state have only one strategy to be represented twice. Therefore all states have only one-way in and one-way out. Then the algorithm has no loop since every state has degree two and you can't go back to the original state. Since there are finite states, Lemke-Howson algorithm will have to eventually reach a state in which every state is represented once that is not all-zeroes. As we argued last time, this is a symmetric Nash equilibrium. ■

3 General setup

Now we switch to games with many players but structured, and examine different questions, such as how much do we lose in terms of overall "quality" of the solution compared to a global optimum, if players are self-interested? We will focus on pure-strategy equilibria. Each player i will have a set S_i of strategies (will also call them "actions") it can take, and the combined strategies chosen by all players $s = (s_1, \ldots, s_n)$ where $s_i \in S_i$ determines the payoffs that each player receives.

3.1 Definitions

Definition 2 (Games and Social Welfare/Cost) *For n players, each player i chooses strategy $s_i \in S_i$, the overall state $s = (s_1, \ldots, s_n) \in S = S_1 \times, \ldots, \times S_n$, and the utility function for player i is $u_i : S \to \mathbb{R}$, or the cost function for player i is $\text{cost}_i : S \to \mathbb{R}$. We define (Sum) Social Welfare of s as the sum of utilities over all players $\sum_{i=1}^{n} u_i(s)$. If cost, $\sum_{i=1}^{n} \text{cost}_i(s)$ is called Sum Social cost of s.*

For the rest of today we will talk about costs.

Definition 3 (Price of Anarchy) *The Price of Anarchy for a game is the ratio of the cost of the worst equilibrium in the game to the cost of the social optimum. The Price of Anarchy for a class of games is the highest such ratio over all games in the class.*

Definition 4 (Price of Stability) *The Price of Stability for a game is the ratio of the cost of the* best *equilibrium to the cost of the social optimum. The Price of Stability for a class of games is the highest such ratio over all games in the class.*

3.2 Example: Fair Cost-Sharing

Suppose there are n players in weighted directed graph G. Player i wants to get from s_i to t_i, and each edge e has cost c_e. Players will share the cost of each edge they use equally with others who are also using it. We call this kind of problem *Fair Cost-Sharing*.

In Figure 1(a), there are two equilibria: A bad equilibrium is that all use the edge of cost n, a good one is that all use the edge of cost 1, which is also the social optimum. This specific game has a Price of Anarchy of n and Price of Stability of 1. This example shows that the Price of Anarchy for fair cost sharing on general graphs is $\geq n$. This turns out to be tight.

Fact 5 *In Fair Cost-Sharing game, Price of Anarchy is always $\leq n$.*

Proof: In every Nash Equilibrium of this game, the cost of player i must be less than or equal to the cost of the shortest path from s_i and t_i, otherwise he/she can choose the

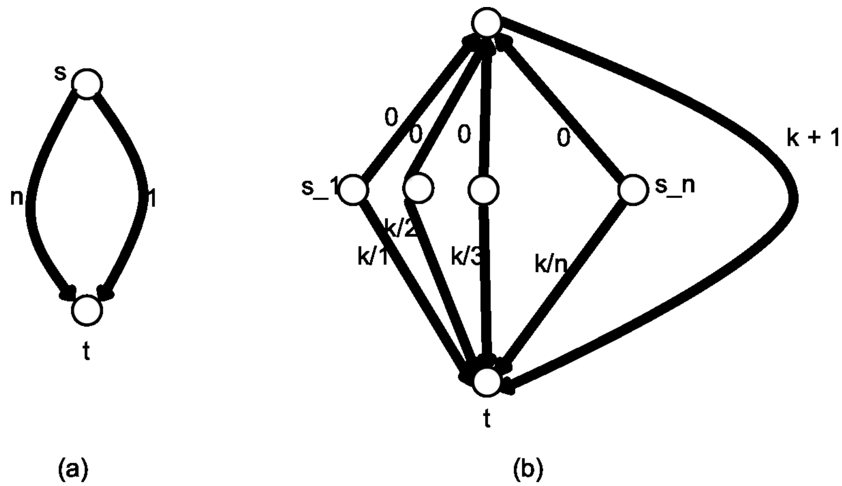

Figure 1: Fair Cost-Sharing

shortest path to get a lower cost. On the other hand, the cost of Social Optimium must be larger than or equal to the maximum of the lengths of these shortest path from s_i to t_i. The sum is at most a factor of n larger than the max. Therefore Price of Anarchy is always $\leq n$. ■

There is one more interesting example in Figure 1(b). The cost of social optimium is $k+1$. However there is only one Nash Equilibrium which has cost

$$k/1 + k/2 + k/3 + \ldots k/n \approx k \ln n$$

This example shows that the Price of Stability for fair cost sharing $= \Omega(\log n)$.

4 Exact Potential Game

In fact, the Price of Stability for fair cost-sharing is $O(\log n)$ too. For ths we will use the fact that fair cost-sharing is an *exact potential game.*

Definition 6 (Exact potential game) *G is an exact potential game if there exists a function $\Phi(s)$ (potential function) such that for all players i, for all states $s = (s_1, \ldots, s_i, \ldots, s_n)$, for all possible moves to state $s' = (s_1, \ldots, s'_i, \ldots, s_n)$,*

$$\text{cost}_i(s') - \text{cost}_i(s) = \Phi(s') - \Phi(s)$$

Recall that a "state" is a vector of pure strategies, one for each player. Notice that if G is an exact potential game, then there must exist a pure-strategy Nash equilibrium since the state at minimum Φ has the property that no player has any incentive to deviate (else it would lower Φ). Futhermore, we can reach a pure-strategy Nash equilibrium by simple best-response dynamics. Each move is guaranteed to reduce the potential function.

Claim 7 *Fair cost-sharing is an exact potential game.*

Proof: We define potential

$$\Phi(s) = \sum_e \sum_{i=1}^{n_e(s)} \frac{c_e}{i}$$

where $n_e(s)$ is the number of players using edge e in state s.

If player changes from path p to path p', he/she would pay $c_e/(n_e(s)+1)$ for each new edge, and gets back $c_e/n_e(s)$ for each old edge. so $\Delta \text{cost}_i = \Delta\Phi$ ■

What is the gap between potential and cost?

$$\text{cost}(s) = \sum_{n_e(s)>0} c_e \le \Phi(s) = \sum_e \sum_{i=1}^{n_e(s)} \frac{c_e}{i} \le (1+\ln n)\text{cost}(s)$$

If we start at socially optimal state OPT and do best-response dynamics from there until reach Nash equilibrium s, we have

$$\text{cost}(s) \le \Phi(s) \le \Phi(\text{OPT}) \le \log(n) \times \text{cost}(\text{OPT})$$

So Price of Stability $= O(\log n)$.

5 Summary of Fair Cost Sharing

For fair cost-sharing in every game we have:

- $\forall$ Nash equilibrium s, $\text{cost}(s) \le n \times \text{cost}(\text{OPT})$
- $\exists$ Nash equilibrium s, $\text{cost}(s) \le \log(n) \times \text{cost}(\text{OPT})$

And there exists games s.t.

- $\exists$ Nash equilibrium s, $\text{cost}(s) \ge n \times \text{cost}(\text{OPT})$
- $\forall$ Nash equilibrium s, $\text{cost}(s) \ge \log(n) \times \text{cost}(\text{OPT})$

Furthermore, the potential function satisfies:

$$\text{cost}(s) \le \Phi(s) \le \log(n) \times \text{cost}(s)$$

6 Congestion Games more generally

A *Congestion Game* is defined by n players and m resources (e.g., resources could be edges in a network). Each player i chooses a set of resources (e.g. a path) from collection S_i of allowable sets of resources (e.g. paths from s_i to t_i). Cost of resource j is a function $f_j(n_j)$ of the number n_j of players using it. Cost incurred by player i is the sum, over all resources being used, of the cost of the resource.

The generic potential function of congestion games is $\sum_j \sum_i^{n_j} f_j(i)$. Best-response dynamics may take a long time to reach a Nash equilibrium, but if gap between Φ and cost is small, it can get to an approximate Nash equilibrium fast (if we define approximate Nash equilibria in an additive way like on the homework).

We just saw that every conjestion game is an exact potential game [Rosenthal '73]. I turns out that the converse is true as well [Monderer and Shapley '96]. For any exact potential game, we can define resources to view it as a congestion game.