

Harvard University
CS 186

Lecture 3

Lecture Notes by Jeremy Nixon

February 4, 2015

1 Overview

1. Folk Theorem
2. Prisoner's Dilemma Tournament
3. P2P File Sharing

1.1 Sections

1. Hongyao - Wed 3-4 PM, MD 323
2. Shosh Thu 1-2 PM, Littauer M-6
3. InYoung Thur 5-6 PM, MD 223

1.2 Office Hours

1. Hongyao - Mon 5-6:30 MD 123
2. Shosh Thu Mon 10-11:30 AM, Littauer G26
3. InYoung Sun 7:30-9 PM Kirkland Dining Hall

2 Prisoner's Dilemma Theory

In the finite prisoner's dilemma, because we will defect in the final period, by backwards induction the only subgame perfect equilibrium is defecting (Nash) in every period.

The distinction that we make between this finite version of the game and an infinite version of the game is what makes it possible to use more complex strategies.

Grim trigger: A player cooperates until they are defected against, and then they defect in every future period.

Claim: The Grim Trigger is a Nash Equilibrium of G^∞ .

Notice: infinite sum with high δ has more potential.

$$v + \delta v + \delta^2 v + \dots = \frac{v}{1-\delta}$$

We can write down the discounted utility for player 1 after there is a defect period - she gets 5 in the period she defects in plus 1 for the rest of the periods, discounted.

In equilibrium, the utility of player 2 is $\frac{3}{1-\delta}$

$$5 + \frac{\delta}{1-\delta}$$

The claim is that for large enough δ , $\frac{3}{1-\delta} \geq 5 + \frac{\delta}{1-\delta}$

Next question - is this a subgame perfect equilibrium? Is this grim trigger effect a credible threat?

Why, in the finite game, does this argument not work? It's that the fact that in an infinitely repeated game, there is no end to start backwards induction from.

3 Folk Theorem

Profile a is *enforceable* if $s^* \in \text{NE}(G)$ s.t. $u_i = u_i(a) \geq u_i s^* = e_i$

Here e_i can be thought of as the enforcing play. All that is necessary is that the utility of the enforcing play is low.

Theorem: There exists a subgame perfect equilibrium of G^∞ where a is played in equilibrium.

Strategy:

(Coop) Play a_i if everyone played a in past.

(Punish) all ways play $s^* i$.

The claim is that this is a Subgame Perfect Equilibrium. We're going to make certain of this by making sure that there is no single profitable equilibrium.

We're going to solve the problem of the infinite number of subgames by breaking the game into a number of classes of subgame.

$$h \in H_1 : \text{no deviation}$$

If you deviate, the best possible thing you'll get in that period will be v_{max} . But into the future, you just get the punishing payoff (discounted).

$$v_{max} + \delta * \frac{e_i}{1-\delta} \leq \frac{v_i}{1-\delta}$$

Check the reading and look at the open loop concept - this is the open loop Nash play.

The difference between this and the grim trigger is that in this case, you are triggered by any player defecting, and in the other game you were only affected if one other player deviated. With this trigger, if you defect you want to cooperate in future. In the other game, you wanted to defect in the other periods.

4 Prisoner's Dilemma Tournament

Woohoo!!! I won!

5 P2P File Sharing

As more people join the system they contribute resources. You care about resilience to resources and scalability.

As we started to get amazon streaming and netflix streaming, the relative share of internet traffic involved in peer to peer file sharing decreased dramatically.

1. Protocol: Messages that can be sent actions that can be taken over the network. These are the rules in a game theoretical system.
 2. Client is a strategy
 3. User is an agent
1. Get a list of IP addresses of peers from set of known peers (no server)

Share versus Free Riding - Players can choose between uploading bandwidth and not uploading. The cost of sharing comes from the resources that are taken from you, worried about illegality, etc.

Despite this, we did see a lot of sharing in the gnutella system. A lot of people were doing substantial amounts of sharing.

Why would anyone share? People may program their client to make people share - you write the system in a way that any downloaded files are automatically uploaded. Unfortunately, there is competition amongst the clients, so other people will build clients that don't use the user's upload capacity. It's easy to imagine an advertising business model or something that made it advantageous for people to design clients that did not force uploading.

5.1 Kazaa

Kazaa was built by the people who made Skype - it promoted cooperation so that you share your statistics and other people can preferentially cooperate you. But people would publish numbers that looked very good.

5.2 BitTorrent

The breakthrough was with BitTorrent - approximately 85% of P2P traffic in the US. Allowed file sharing.

The key innovation in BitTorrent is that you break the file into many small pieces. Then you run a repeated game between the users. It allows for tit-for-tat like behavior, saying that "If you let me download, I'll reciprocate."

Flash crowd - you get a lot of people in a burst sharing a particular file.

1. Download from as many people as possible.
2. You're dividing your upload capacity into chunks. For some of your upload capacity, you allow people to download from you based on who you can reciprocate with. The final part of your upload capacity is subject to optimistic unchoking. If you don't optimistically unchoke, there is no way for new peers to enter the system. You also may discover a new peer with something useful.

5.2.1 Types of attack

1. Free riders: Never let anybody download from you, and use optimistic unchoking for everybody. That's what the BitThief algorithm did. Keep asking for peers from tracker, grow neighborhood quickly. Never upload! It's about 2-4x slower than tit for tat, but if you really don't want to use any of your upload you can. Here the defense is simple - you don't let the same IP address request new peer interactions for half an hour.
2. Strategic Piece Revealer - Reference client: tell neighbors about new pieces, use rarest piece first. Using rarest piece first makes the rarest pieces more common, so that other people can help upload rare pieces. New Strategy: reveal most common piece that reciprocating peer does not have! Protect a monopoly on pieces, keep others interested. This lets you extract as much from them as possible - you have the critical piece that

they need. Holding on to these rare pieces forces others to be more cooperative than they would be otherwise. This is not an equilibrium analysis - if everybody does this, then the system becomes 12% worse

3. BitTyrant - a return on investment approach. I'm going to give you just enough of my upload capacity to get you to reciprocate back with you. And then I'm going to use my bandwidth to work with more other people.