Free-rider problem in peer-to-peer networks

Sumitra Ganesh

What is the Free-rider problem

When no individual is willing to bear the cost of something when he expects that someone else will bear the cost instead.

The Free-rider's reasoning

If other's work hard, I can share in the good result even if I don't put in any effort.

If others do not work hard I have no reason to work hard either because other people will share what the results of my work while I myself will get only a small portion of what I produce.

Two Player game

A	B	Work Hard	Free Ride
Work Hard		(2,2)	(0,3)
Free Ride		(3,0)	(1,1)

In a one-shot game the dominant strategy for both Players is to “Free ride”. In a repeated game to free-ride is no longer the dominant strategy if the game has a large probability to continue.

Free riding on P2P networks

An extensive analysis of user traffic on Gnutella revealed the following [1] :

- Quantity of files shared : Nearly 70% of users share no files.
- Quantity vs "Desirability" : Top 1% of the peers provided 47% of the answers, top 25% provided 98%.
- The queries are concentrated on particular topics : top 1% of the queries accounted for 37% of the total queries, top 25% for over 75% of all queries.

Game Theoretic Approach

Users = Players : Users have positive utilities for files downloaded and negative utilities (costs) for files uploaded/shared.

Central Server/P2P software (Napster/Kazaa) : Means of implementing the mechanism

Aim : To design mechanism that forces the users to a non free-riding equilibrium by providing appropriate incentives/disincentives to the players.

Utility of a user

Positive Utility

- Files downloaded
- Search results (quantity/bandwidth/variety)
- Kicks

Negative utility

- Delay in download
- Disk Space and Bandwidth required for sharing
- Time for which the files have to be shared

Problem Statement

Aim : To design a payment mechanism that :

- Forces the seller to reveal his true cost and provides compensation
- Forces the buyer to reveal his true value and charges him appropriately.
- Include a possible revenue model for service provider

VCG Mechanism

Problem Setup

Good = (file, level) : There are different levels at which a file can be provided (different levels of bandwidth)

Double-sided auction : There are buyers and sellers – Each player is both a buyer and a seller of goods.

Central server/Software : Auctioneer who makes the allocation according to the mechanism

Resource constraints : The total bandwidth available with any user is limited.

Nomenclature :

The file *k* at level *l* is transferred from User *i* to User *j*

The value of *(k,l)* to User *j* is $d_{j,(k,l)}$

The cost incurred by User *i* is $c_{i,(k,l)}$

Allocation function :

$x_{i\to j,(k,l)}$ = 1 ...if *(k,l)* is transferred from User *i* to User *j*

$x_{i\to j,(k,l)}$ = 0...otherwise

Constraints

$x_{i \to j,(k,l)} = 0$ if $d_{j,(k,l)} = 0$ (1)

(user *j* does not request for the file)

$x_{i \to j,(k,l)} = 0$ if $c_{i,(k,l)} = \infty$ (2)

(user *i* does not have the file)

Each user provides files at some level l_i

$$\sum_{j \neq i,k} x_{i \to j,(k,l_i)} + \sum_{j \neq i,k,l} x_{j \to i,(k,l)} \leq K_{l_i}$$

uploads *downloads*

Value function of a user (quasi-linear)

$$v_i(x) = -\sum_{j,k} x_{i \to j,(k,l)} c_{i,(k,l_i)} + \sum_{j,k,l} x_{j \to i,(k,l)} d_{i,(k,l)}$$

Utility function of a user

$$U_i(x) = v_i(x) - P_i(x)$$

Where the payments are defined by the mechanism

VCG Mechanism

Optimum Allocation Rule

$$x^* = \arg\max \sum_{i=1}^{N} v_i(x) = \arg\max V(x)$$

Payment Rule for a user

$$P_i(x^*) = V_{-i}(x^*) - V(x^*_{-i})$$

Revenue of auctioneer = $\sum_{i=1}^{N} P_i(x^*) \geq 0$

(Weakly Budget Balanced)

Assumptions

1. The value functions are quasi-linear.

2. The allocation is fixed and determined by a central agency.

3. The user has the freedom to specify the file and the level (of bandwidth) but does not get to choose the source from which he downloads. (Can modify the allocation function to be fractional probabilities)

The Good News

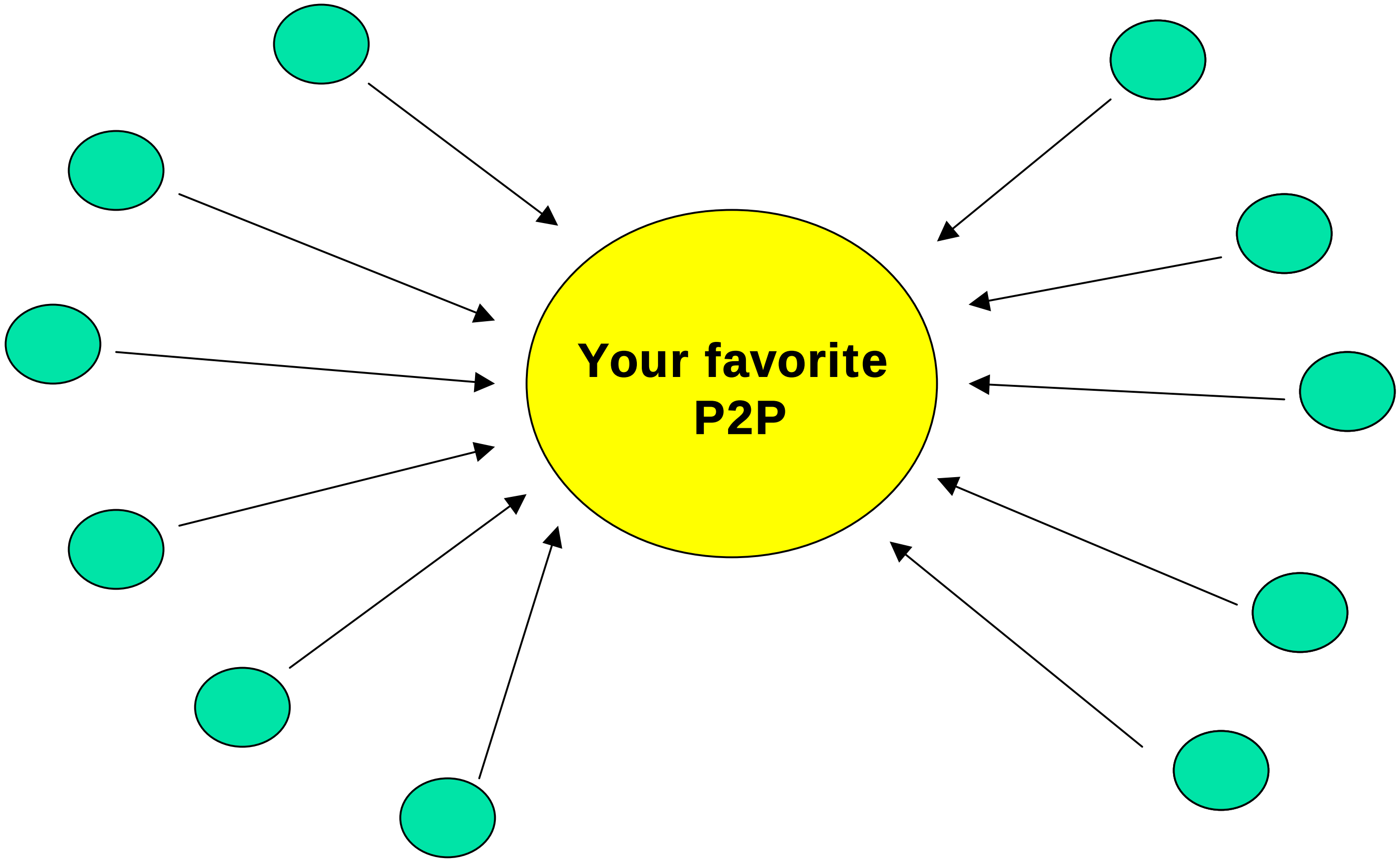
1. Allocative efficiency : maximizes the total value over all players
2. Strategy proof : Truth revelation is the dominant strategy

3. VCG Mechanisms are the *only* allocatively-efficient and strategy-proof mechanisms for agents with quasi-linear preferences and general valuation functions, among all direct revelation mechanisms.

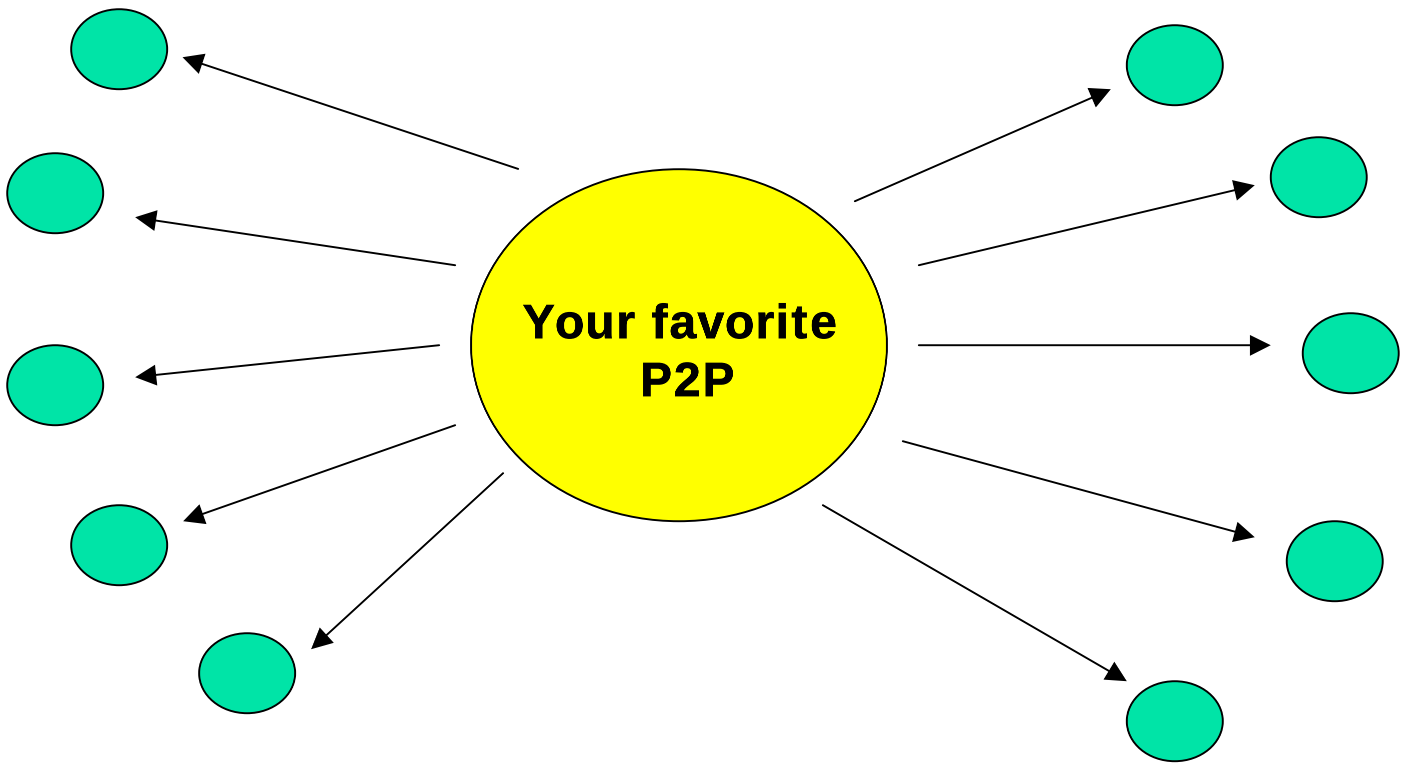
...And the bad

Computational and Communication complexity

- Users have to compute their values/costs
- Users have to communicate their values/costs
- Winner determination is computationally expensive
- Winner determination is completely centralized



Users have to compute and communicate their values/costs



The software has to compute and communicate the allocation and payments to the users

Questions

Local valuation problem : iterative/dynamic mechanism

Winner determination problem : Approximate VCG

What happens to the allocative-efficiency and strategy-proofness ??? [3]

What is the communication overhead ??

Repeated game scenario – how does this extend ???

References

1) E. Adar and B. Huberman. *Free riding on Gnutella*. First Monday, 5(10), 2000.
2) Kevin Leyton-Brown et. al. *Incentives for Sharing in Peer-to-Peer Networks.*

   http://robotics.stanford.edu/~kevinlb/recent_work.html
3) David Parkes. Chapter 2 and 3, *Iterative Combinatorial Auctions: Achieving Economic and Computational Efficiency* Ph.D. dissertation, University of Pennsylvania, May, 2001.

   http://www.eecs.harvard.edu/~parkes/mechdesign.html