Optimal Protocol Design in Networks with Selfish Users

Tim Roughgarden (Stanford)

includes joint work with:

Ho-Lin Chen (Stanford), Gregory Valiant (UC Berkeley)

Inefficiency in Large Networks

Fact: optimization in/of large networks is hard.

- optimal allocation of network resources
- optimal production of resources (e.g., topology)

Two reasons:

- implementation constraints: need distributed algorithms and protocols
- economic constraints: end users non-cooperative

Consequence: optimality often unachievable.

Optimal Protocol Design

High-level goal: identify "optimal solution" subject to joint implementation + economic constraints.

Feasible solution: a distributed protocol meeting the implementation constraints.

Observation: a protocol induces a game among the end users (economic constraints).

- game depends on choice of protocol, but underlying optimization problem does not

Optimal Protocol Design (con'd)

Objective function: minimize worst-case efficiency loss in game induced by protocol.

Efficiency loss: given game, optimization problem, and equilibrium concept (e.g. Nash), can define:

$$\text{Price of Anarchy (POA)} = \frac{\text{cost(worst eq)}}{\text{cost(OPT)}}$$

(or infinity if no equilibria exist)

The Meta-Problem

Goal: design protocols (s.t. implementation constraints) to minimize worst-case POA.

Why bother?:

- rigorous notion of an "optimal" protocol
- quantify trade-offs between different objectives (e.g., fairness vs. efficiency)
- quantify trade-offs between different design constraints (e.g., state required vs. efficiency)

Cost-Sharing Protocols

This talk: illustrate paradigm via design of optimal cost-sharing protocols.

  - [Chen/Roughgarden/Valiant 07]

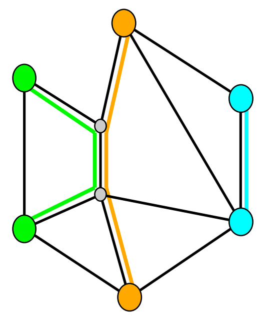
Related work:

- coordination mechanisms for scheduling
  - [Christodoulou/Koutsoupias/Nanavati ICALP 04]
  - [Immorlica/Li/Mirrokni/Schulz WINE 05]
  - [Azar et al 07]
- resource allocation [Johari/Tsitsiklis 07]

A Network Formation Model

Given: $G = (V,E)$, fixed costs c_e

- k players = vertex pairs (s_i,t_i)
- each picks an s_i-t_i path



Cost model:

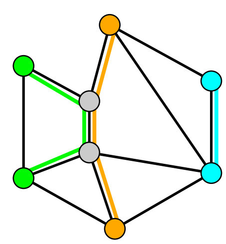
- cost of outcome = sum of costs of edges used by at least one player

Cost sharing: insist that this cost is passed on to players ("budget-balance")

7

Shapley Cost Sharing

Protocol design: How should multiple players on a single edge split costs?



Shapley cost sharing:
Players using e share costs evenly:

$$c_i(P) = \sum_{e \in P} c_e / k_e$$

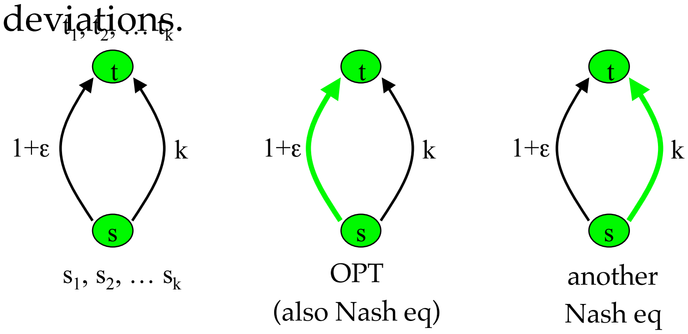
[Anshelevich et al FOCS 04]

- players' objectives: minimize individual cost
- global objective: minimize total network cost

8

Multiple Nash Equilibria

Recall: Nash equilibrium = choice of path for each player s.t. no profitable unilateral deviations.



Corollary: worst-case POA of Shapley = k.

The Optimizer's Mantra

Question #1: can we do better?

- want (much) smaller worst-case POA

Question #2: subject to what?

- i.e., what are the implementation constraints?

In Defense of Shapley

Essential properties: (non-negotiable)

- "budget-balanced" (total cost shares = cost)
- "separable" (cost shares defined edge-by-edge)
- pure-strategy Nash equilibria exist

Bonus good properties: (negotiable)

- "uniform" (same definition for all networks)
- "fair" (characterizes Shapley)

Priority Protocols

Priority Cost-Sharing Protocol:

- order the players (arbitrarily)
- full cost of edge charged to its "earliest" user

Observation: always have a unique (up to ties) Nash equilibrium.

- iterated removal of dominated strategies

Priority Protocols

Priority Cost-Sharing Protocol:

- order the players (arbitrarily)
- full cost of edge charged to its "earliest" user

Observation: always have a unique (up to ties) Nash equilibrium.

- iterated removal of dominated strategies

Fact: POA radically better! (in undirected graphs)

- $\Theta(\log k)$ in single-sink networks [Imase/Waxman 91]
- $O(\log^2 k)$ in general [Awerbuch/Azar/Bartal 96]

Can We Do Better?

Non-Uniform Improvement: (undirected, 1 sink)

- order players via Prim's MST algorithm
- worst-case POA = 2! [easy fact: this is best possible]

Uniform protocols: more practical.

- Can we still get a constant worst-case POA?

Key question: what are the alternatives to the Shapley and priority protocols?

In Defense of Shapley (reprise)

Ambitious goal: characterize all protocols that satisfy first 4 properties of Shapley:

- "budget-balanced" (total cost shares = cost)
- "separable" (cost shares defined edge-by-edge)
- pure-strategy Nash equilibria exist
  - "stability" --- a complex, "global" constraint
- "uniform" (same definition for all networks)

Potential Funtions

Defn: Φ (function from outcomes to reals) is a potential function if for all outcomes S, players i, and deviations by i from S:

$$\Delta\Phi = \Delta c_i$$

- "tracks" deviations by players
- assures existence of a Nash eq (consider global min)
- not necessarily a natural objective function
- [Beckman/McGuire/Winsten 56], [Rosenthal 73], [Monderer/Shapley 96]

Potential Funtion for Shapley

Claim: every Shapley network design game has a potential function (hence Shapley is stable).

Proof: Define $f(S) = 1 + 1/2 + 1/3 + \ldots 1/|S|$ and $\Phi(P_1,\ldots,P_k) = \Sigma\, c_e\, f(S_e)$. QED.

Potential Funtion for Shapley

Claim: every Shapley network design game has a potential function (hence Shapley is stable).

Proof: Define $f(S) = 1 + 1/2 + 1/3 + \ldots 1/|S|$ and $\Phi(P_1,\ldots,P_k) = \Sigma\, c_e\, f(S_e)$. QED.

Next: identify other (stable) protocols amenable to potential function method.

Characterization, Part I

Definition: a cost-sharing protocol is positive if it always assigns strictly positive cost shares.

- precludes priority protocols

Characterization, Part I

Definition: a cost-sharing protocol is positive if it always assigns strictly positive cost shares.

- precludes priority protocols

Characterization, Part I: every stable positive protocol is induced by a potential function.

- of the form $\Sigma\, c_e\, f(S_e)$ for a suitable function f
- in correspondence with interior of k-simplex
- similar but distinct from "weighted Shapley" protocol
  - [Chen/Roughgarden SPAA 06]

Concatenation

Definition: For any two cost sharing schemes ξ_1 and ξ_2, the concatenation of ξ_1 and ξ_2 is

$$(\xi_1 \oplus \xi_2)(i, S) = \begin{cases} \xi_1(i, S \cap A_1) & \text{if } i \in A_1 \\ \xi_2(i, S) & \text{if } S \subseteq A_2 \\ 0 & \text{otherwise.} \end{cases}$$

Notes:

- If ξ_1 and ξ_2, are stable, then so is their concatenation.
- Priority protocols = concatenation of 1-player protocols.

Characterization (final)

Theorem: a uniform cost-sharing protocol is stable if and only if it is the concatenation of potential-based cost-sharing protocols.

Characterization (final)

Theorem: a uniform cost-sharing protocol is stable if and only if it is the concatenation of potential-based cost-sharing protocols.

Application #1: every such protocol has worst-case POA = $\Omega(\log k)$, even in single-sink networks.

- non-trivial proof: group players according to "weight", use one of two types of bad examples

Corollary: priority protocols are optimal!!

23

Directed Graphs

Fact: every protocol (even non-uniform) has worst-case POA = k in directed networks.

Shapley: best Nash equilibrium within $H_k \approx \ln k$ of optimal solution. ("price of stability")

24

Directed Graphs

Fact: every protocol (even non-uniform) has worst-case POA = k in directed networks.

Shapley: best Nash equilibrium within $H_k \approx \ln k$ of optimal solution. ("price of stability")

Application #2: every uniform and stable protocol has POS ≥ H_k in directed networks.

- follows from "monotonicity" of stable protocols

Corollary: the Shapley protocol is optimal!!

- fairness comes for free!

Open Questions

- quantifiable efficiency vs. fairness trade-offs for undirected networks
- non-uniform methods in directed networks
- characterization theorem and/or lower bounds for scheduling mechanisms
- new applications (selfish routing, queuing disciplines, etc., etc.)