Best-response dynamics (with applications to distributed protocols and mechanisms)*

Lecturer: Paolo Penna

February 24, 2014

1 Warm up

Which of these three games have **pure Nash equilibria (PNE)**?

	H	T
H	-1 1	1 -1
T	1 -1	-1 1

Matching Pennies

	B	S
B	1 2	0 0
S	0 0	2 1

Battle of Sexes

*The material of this lecture is taken from Nisan et al. (2011a) where you can find several other applications of best-response mechanisms. There you have a more precise, extensive, and formal description of best-response mechanisms, plus further pointers into the literature.

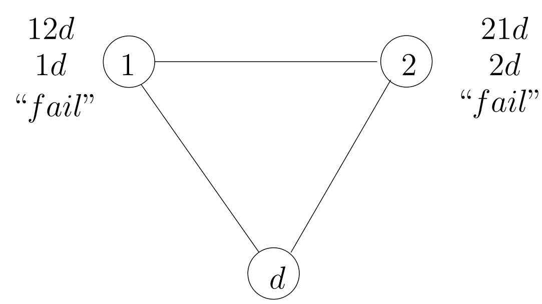
	A	S
A	-2, -2	0, -3
S	0, -3	-1, -1

Prisoners' Dilemma

> **(Synchronous) Best-Response Dynamics**: Players play their best response infinitely many times, one by one in a fixed order (round robin).

What happens for the three games above?

Example 1 Two nodes, 1 and 2, want to send traffic to another destination node d. Their strategy is to choose the *next hop* the traffic is sent to (one of the neighbors). The following picture shows the physical network and the preferences of each node (which path to use) near the corresponding node:



Each node prefers to reach d via the other node, but if they both send their own traffic to each other they fail (which is the least preferable option for both). ■

> **Question:** What happens if the two nodes move (play) always simultaneously? What happens if node 1 plays "1 → 2" at each step (while the other node plays best-response)?

Best Response:

1. No convergence in *asynchronous* settings.
2. Not incentive compatible.

For which games this does not happen?

Asynchronous Best-Response Dynamics: At each step an adversary activates an arbitrary subset of players who best respond to the current profile (the adversary also chooses a starting strategy profile). The adversary must activate each player an infinite number of times.

The choice of the adversary and the "response strategies" of each player determine an infinite sequence

$$s^0 \Longrightarrow s^1 \Longrightarrow \cdots s^t \Longrightarrow \cdots$$

If the game converges (after finitely many steps T we have $s^T = s^{T+1} = s^{T+2} = \cdots$) then the utility of each player i is $u_i(s^T)$. If the game keeps "oscillating" then we consider an upper bound on what the player can get (the worst case for us and the best for the player) that is $\limsup_{t\to\infty} u_i(s^t)$.

$$\begin{array}{rcl} \text{Base game } G & \Longrightarrow & \text{Repeated game } G^* \\ s_i \in S_i & & \text{response strategy } R_i() \in S_i \\ u_i(s) & & \text{total utility } \Gamma_i := \limsup\limits_{t\to\infty} u_i(s^t) \end{array}$$

Definition 2 *Best-response are* ***incentive compatible*** *for G if repeated best-responding is a Nash equilibrium for the repeated game G^*, that is, for every i*

$$\Gamma_i \geq \Gamma_i'$$

where Γ_i i the total utility when all players best respond and Γ_i' is the total utility when all but i best respond (starting from the same initial profile s^0 and applying the same activation sequence).

2 “Nice” Games

Consider this game (with a unique PNE):

		Player 2	
		A	B
Player 1	A	2, 1	0, 0
	B	3, 0	1, 2

Best response works as follows

$$(A,A) \overset{\textit{Player } 1}{\Longrightarrow} (B,A) \overset{\textit{Player } 2}{\Longrightarrow} (B,B) \overset{\textit{Player } 1}{\Longrightarrow} (B,B) \overset{\textit{Player } 2}{\Longrightarrow} (B,B) \cdots \Longrightarrow (B,B)$$

Player 1 improves if he/she does not best response (keep playing A):

$$(A,A) \overset{\textit{Player } 1}{\Longrightarrow} (A,A) \overset{\textit{Player } 2}{\Longrightarrow} (A,A) \overset{\textit{Player } 1}{\Longrightarrow} (A,A) \Longrightarrow \cdots \Longrightarrow (A,A)$$

> Convergence but no incentive compatibility

Exercise 1 For the following game

		Player 2	
		A	B
Player 1	A	1, 1	1, 1
	B	1, 1	1, 1

find best response strategies that *never converge* (keep oscillating between different profiles). Find other best response strategies for which we *do have convergence*. ■

> Two intuitions/ideas:
>
> 1. Introduce tie breaking rule.
>
> 2. Eliminate “useless” strategies.

2.1 Convergence

Consider this game

	A	B	C
a	$2, 1$	$0, 0$	$0, 0$
b	$1, 2$	$1, -1$	$-1, 1$
c	$-1, -2$	$-1, 1$	$1, -1$

Exercise 2 *Prove that for this game best-response dynamics converge to a unique PNE.*

Note that in the previous game no strategy is dominant and no strategy is dominated. Strategy C satisfies the following (weaker) definition:

Definition 3 (never best response (NBR)) *A strategy $s_i \in S_i$ is a* never best response (for tie breaking rule $\prec$) *if there is always another strategy that gives a better payoff or that gives the same payoff but is better w.r.t. to this tie breaking rule: for all s_{-i} there exists $s'_i \in S_i$ such that one of these holds*

1. $u_i(s_i, s_{-i}) < u_i(s'_i, s_{-i})$ *or*
2. $u_i(s_i, s_{-i}) = u_i(s'_i, s_{-i})$ *and* $s_i \prec_i s'_i$.

The following condition is enough to guarantee convergence:

Definition 4 (NBR-solvable) *A game G is* NBR-solvable *if iteratively eliminating NBR strategies results in a game with one strategy per player. That is, there exists a tie breaking rule $\prec$, sequence $p_1, \ldots, p_\ell$ of players, and a corresponding sequence of subsets of strategies $E_1, \ldots, E_\ell$ such that:*

1. *Initially $G_0 = G$ and $G_i + 1$ is the game obtained from G_i by removing the strategies E_i of player p_i;*
2. *Strategies E_i are NBR for $\prec$ in the game G_{i-1}.*

3. *The final game G_ℓ has one strategy for each player (this unique profile is thus a PNE for G).*

A sequence of players and of strategies as above is called an elimination sequence *for the game G.*

Exercise 3 *Prove that the game described at the beginning of this section is NBR-solvable. Provide also a bound on the parameter ℓ.*

Lemma 5 (rounds vs subgames) *Let $p_1, \ldots, p_\ell$ be the players of any elimination sequence for the game under consideration. Suppose that players $p_1, \ldots, p_k$ always best respond (according to the prescribed tie breaking rule $\prec$). Then, for any initial profile and for any activation sequence, every profile after the k^{th} round is a profile in the subgame G_k.*

Before proving the lemma we observe that it implies convergence:

Theorem 6 (convergence) *For NBR-solvable games best response (according to the prescribed tie breaking rule $\prec$) converge even in the asynchronous case.*

Proof. Take $k = \ell$ and observe that G_ℓ contains only one profile. $\square$

Proof of Lemma 5. Denote by $round_j$ the last time step of the j^{th} round in the activation sequence. Obviously for any t we have $s^t \in G_0 = G$. Now consider $t \geq round_1$ and observe that, since player p_1 has been activated at least once the corresponding strategy satisfies [1]

$$s^t_{p_1} \notin E_1$$

which is equivalent to $s^t \in G_1$ for all $t \geq round_1$.

To prove the analogous for player p_2 we observe that, in the 2^{nd} round player p_2 is activated and, since $s^t \in G^1$ and since p_2 plays best response, for $t \geq round_2$ we have $s^t_{p_2} \notin E_2$. Since we have previously proved $s^t_{p_1} \notin E_1$, this implies $s^t \in G_2$ for $t \geq round_2$.

We can then continue and prove, by induction, that after the k^{th} round player p_k does not play any strategy in E_k and thus $s^t \in G_k$ for all $t \geq round_k$. $\square$

[1] More in detail, if the player is activated at time t' then at time $t' + 1$ his/her profile is not in E_1; If the player is not activated at time t' then her strategy at time $t' + 1$ remains the same.

2.2 Incentive Compatible

Look (again) at this game:

		Player 2	
		A	B
Player 1	A	2, 1	0, 0
	B	3, 0	1, 2

> **Bad for incentive compatibility:** The unique PNE does not give Player 1 the highest possible payoff he/she can get in this game.

Definition 7 (NBR-solvable with clear outcome) *A NBR-solvable game G has a clear outcome if there exists a tie breaking rule $\prec$ such that the following holds. For every player i there exists an elimination sequence consisting of players $p_1, \ldots, p_a, \ldots, p_\ell$ and strategies $E_1, \ldots, E_a, \ldots, E_\ell$ (according to Definition 4) such that,*

1. *p_a denotes the* first appearance *of i in the sequence, that is,*

$$p_a = i \neq p_1, p_2, \ldots, p_{a-1};$$

2. *in the corresponding subgame*

$$G_{a-1} = G \setminus (E_1 \cup E_2 \cup \cdots \cup E_{a-1})$$

the PNE s^* is globally optimal for i, *that is,*

$$u_i(\hat{s}) \leq u_i(s^*) \quad \textit{for all } \hat{s} \in G_{a-1}.$$

(Recall that s^* is the unique profile in the final subgame G_ℓ.)

Theorem 8 (incentive compatibility) *For NBR-solvable games best response (according to the prescribed tie breaking rule $\prec$) are also incentive compatible.*

Proof. Compare the case in which all players best respond to the case in which player i does not best respond (while the others best respond). In particular, we consider the two sequences of profiles

$$\begin{array}{rl} \text{All best respond: } s^0 & \Longrightarrow \quad s^1 \Longrightarrow s^2 \Longrightarrow \cdots \Longrightarrow s^* \Longrightarrow s^* \cdots \\ \text{All but } i \text{ best respond: } s^0 & \Longrightarrow \quad \hat{s}^1 \Longrightarrow \hat{s}^2 \Longrightarrow \cdots \Longrightarrow \hat{s}^t \Longrightarrow \hat{s}^{t+1} \cdots \end{array}$$

We want to show that starting from some finite T the utility of i in the second sequence is not better than the "final" utility in the first sequence:

$$u_i(\hat{s}^t) \leq u_i(s^*) \quad \text{for all } t \geq T \tag{1}$$

This implies $\hat{\Gamma}_i \leq \Gamma_i$ that is the incentive compatibility condition (see Definition 2). Consider the elimination sequence of definition of NBR-solvable game (Definition 7) and let $p_k = i$ be the first occurrence of i in the sequence (i.e. $i \neq p_1, \ldots, i \neq p_{k-1}$):

$$\begin{array}{r|cccccc} \text{Player:} & p_1 & \cdots & p_{k-1} & i & p_{k+1} & \cdots \\ \text{NBR Strategies:} & E_1 & \cdots & E_{k-1} & E_k & E_{k+1} & \cdots \\ \text{Current Game:} & G_0 & \cdots & G_{k-2} & G_{k-1} & G_k & \cdots \end{array}$$

We know from Lemma 5 that after round $k-1$ the profile must be in the game G_{k-1} (since i does not appear in the elimination sequence before position k, all players $p_1, \ldots, p_{k-1}$ are different from i and thus they all play best response). Since the PNE s^* is globally optimal for i in this subgame, we have $u_i(s^t) \leq u_i(s^*)$ for all $t \geq round_{k-1}$. This proves Inequality (1) and thus the theorem. □

3 TCP Games

We begin with a toy example. Two players want to send data through a link of a certain capacity C. Each player i can select a sending rate s_i (the *strategy* of player i) in an interval $[0, M_i]$, and the channel policy (if capacity is exceeded some packets are dropped) determines the actual rate r_i for each player (this amount is the *payoff* of player i).

The following is an abstract view of what TCP prescribes to do:

Probing Increase Educated Decrease (PIED): Send exactly at the maximum rate that you can get (not more than that).

After gradually increasing the sending rate, at some point some packets are dropped. This is a way for player i to learn the maximum rate he/she can get without packets being dropped. PIED prescribes to send at this maximum rate, that is, to play

$$s_i^* := \max\{s_i \in [0, M_i] | \ \ r_i(s_i, s_{-i}) = s_i\},$$

where the actual rate $r_i()$ depends on the channel policy.

Exercise 4 *Explain why PIED is not incentive compatible if the channel policy is to divide the total capacity* proportionally *to the sending rate of the player (whenever their requests exceed C):*

$$r_i = C \frac{s_i}{\sum_j s_j}.$$

We introduce a channel policy that makes PIED incentive compatible and uses the whole channel capacity:

Strict Priority Queuing: Try to satisfy the players requests one-by-one in a fixed order:

$$\begin{aligned}
r_1 &\leftarrow \min(s_1, C) \\
r_2 &\leftarrow \min(s_2, C - r_1) \\
&\vdots \\
r_n &\leftarrow \min(s_n, C - r_1 - r_2 \cdots - r_{n-1})
\end{aligned}$$

Exercise 5 *Show that if the channel uses a Strict Priority Queuing policy then PIED converges and is incentive compatible.*

Fair Queuing:

1. Allocate the capacity C evenly among all sending "requests"

$$r_i \leftarrow \min(s_i, C/n)$$

2. Recursively allocate the residual capacity among all partially satisfied requests as in previous step:

$$C \leftarrow C - \sum_i r_i \qquad\qquad s_i \leftarrow s_i - r_i.$$

Exercise 6 *Show that if the channel uses a Fair Queuing policy then PIED converges and is incentive compatible.*

4 Mechanisms with Money

We run an auction for selling an item to the players, each player has his/her own valuation v_i for the item. Consider the 2^{nd}**-price auction** in which the highest bid wins the item and the price to pay is the 2^{nd}-highest bid. For example

$$bids: \quad 1, \underbrace{5}_{pays\ 3}, 3$$

The utility of the winner equals to the difference between the valuation and the price to pay (the others have zero utility). This auction "simulates" a **repeated** 1^{st}**-price auction**:

$$bid_1 = 0.1 \to bid_2 = 0.2 \to \cdots \to bid_3 = 3 \to bid_2 = 3.000001$$

where the winner pays his/her final bid.

Exercise 7 *Consider two bidders having* different *valuations and the case bids/valuations are discrete (integers). Describe repeated* 1^{st}*-price auction as a best-response dynamics and prove that it converges and is incentive compatible. Explain how you can deduce from this that* 2^{nd}*-price auction is truthful (reporting a bid different from the true valuation does not improve the utility of the corresponding player).*

A second type of auctions consider **cost-sharing** problems, whose simplest case is the following one. The cost for providing a service to any subset of users is 1 and this cost must be recovered from those who are interested (enough) in the service. We decide to have a payment scheme which divides the cost equally among all users who get the service, so the payment scheme is of the form

$$\begin{array}{cccc} 1 & & & \\ 1/2 & 1/2 & & \\ 1/3 & 1/3 & 1/3 & \\ 1/4 & 1/4 & 1/4 & 1/4 \\ \vdots & & & \end{array} \tag{2}$$

> **Moulin mechanism:** Service the largest set S of players that accept to pay $1/|S|$, and charge each of them this amount (the others get no service and pay nothing).[a]
>
> [a] A player accepts to pay a price p if his/her bid b_i is at least this price.

For example

$$bids: \quad \underbrace{1/5, 1/4}_{excluded}, \underbrace{1/2, 1}_{pay\ 1/2}$$

This is the 1^{st}**-price counterpart** of the above mechanism:

> Service the largest set S of players as in Moulin mechanism, and charge each serviced player **his/her bid**.

Consider the **repeated** version of the latter mechanism and ask each player to do the following:

> Submit the minimal bid $b_i \leq v_i$ for which you are included in S, given the current bids of the others, if such a bid exists; Otherwise, submit the largest bid $b_i \leq v_i$.

Here is what happens when the true valuations are $(1/5, 1/4, 1/2, 1)$ and players are activated in round robin fashion:

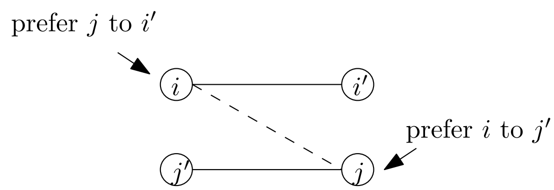
$$(0,0,0,0) \;\rightarrow\; (1/5,0,0,1) \rightarrow (1/5,1/4,0,1) \rightarrow (1/5,1/4,1/2,0) \rightarrow (1/5,1/4,\underbrace{1/2,1}_{S})$$

$$\rightarrow \cdots \qquad \cdots \rightarrow (1/5,1/4,\underbrace{1/2,1/2}_{S})$$

Exercise 8 *Prove that the repeated version of the 1^{st}-price mechanism gives the same result as Moulin mechanism with input the true valuations of the players.*

5 Stable Matchings

The general version of the problem considers n players, each of them having preferences over the others. A **stable matching** is a matching such that there are no two players who prefer each other to their matched partners, that is, nothing like this should happen:



Exercise 9 *Show that in general a stable matching may not exist.*

Exercise 10 *Show that a stable matching exists for the following bipartite restriction of the problem. Players are partitioned into **researchers** and **universities**. Universities have a common (same) rank of researchers, while researchers rank universities differently (e.g., based on salary, location, etc.). Describe an algorithm for computing a stable matching.*

The instances described in the previous exercise satisfy the following definition:

Acyclic Instances: There is no cycle of $\ell \geq 3$ players

$$i_1 \to i_2 \to \cdots \to i_\ell \to i_1$$

such that each player prefers the next one over the previous one.

An intuitive mechanism would be to let players propose to the others. A player i **makes a better offer to** j if i proposes to j and j prefers i over all players that currently propose to j. Then a player should try to make a better offer to the player he/she likes the most:

Mechanism for Stable Matching:

- Each player checks which players he/she can make a better offer to, and then proposes him/herself to the most preferred one in this set.

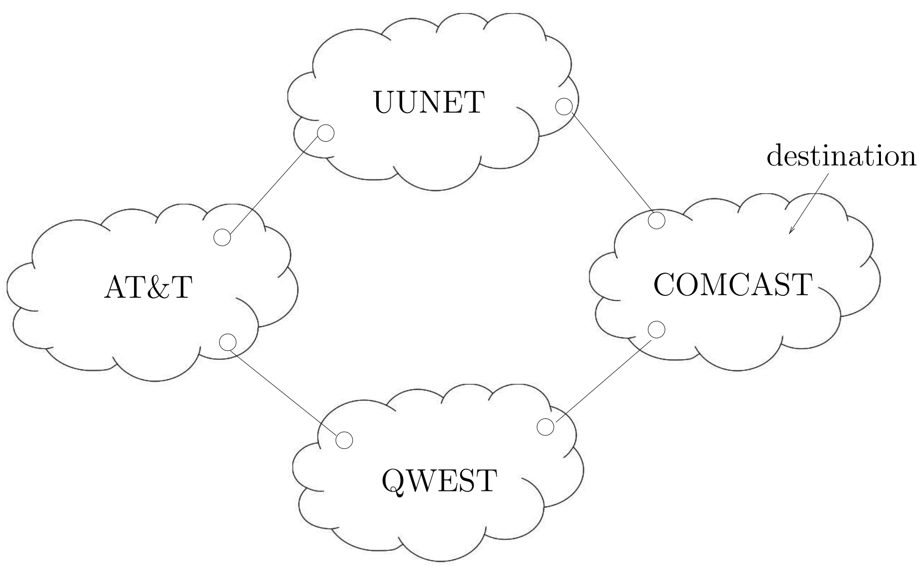
One can view the above mechanism as a best-response dynamics for the following **stable matching game**:

$$u_i(s) = \begin{cases} rank_i(s_i) & \text{if } i \text{ makes a better offer to } s_i \\ 0 & \text{otherwise} \end{cases}$$

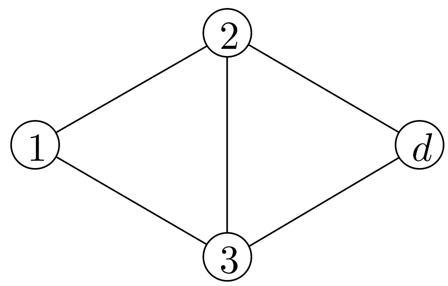
Exercise 11 *Prove that for acyclic instances the above mechanism converges and is incentive compatible (no player can get matched to a player he/she likes more by misreporting her preferences).*

6 BGP Games

Several Autonomous Systems are connected to each other:



The Border Gateway Protocol (BGP) specifies how to forward traffic. Each node in this graph chooses neighbor ("next hop"):



6.1 BGP "in Theory"...

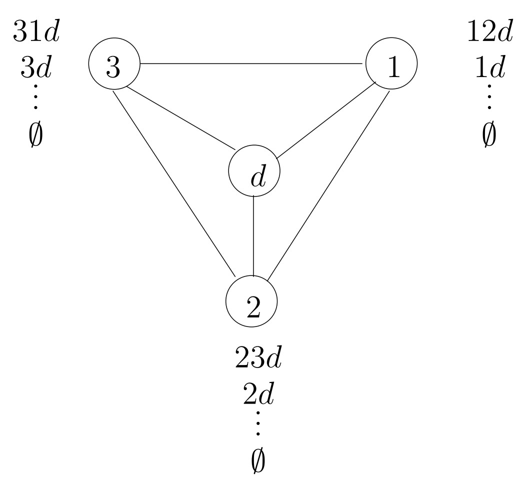
BGP game (static version)

1. Players = Nodes
2. Strategies = Neighbors
3. Strategy profile = Set of paths (or loops)
4. Utilities = Order over the paths connecting i to d

$$P_1 \prec_i P_2 \prec_i \cdots \prec_i P_k$$

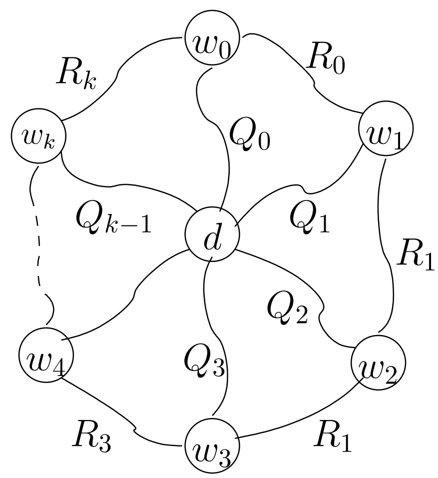
   and any path $\emptyset$ which does **not** connect i to d is strictly worse: $\emptyset \prec_i P_1$.

Consider this instance:



There is no PNE.

Dispute Wheel: every node prefers routing over the next one in the "wheel"



with preferences

$$Q_i \prec_{w_i} R_i Q_{i+1}$$

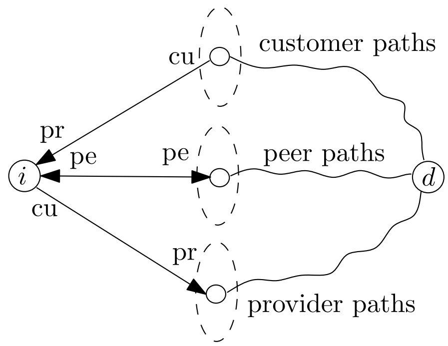
no convergence + no incentive compatible

6.2 ...BGP "in Practice"

Gao-Rexford Model $\Longrightarrow$ No Dispute Wheel $\Longrightarrow$ BPG Converges Incentive Compatible

There are two types of **commercial relationships** between ASs:

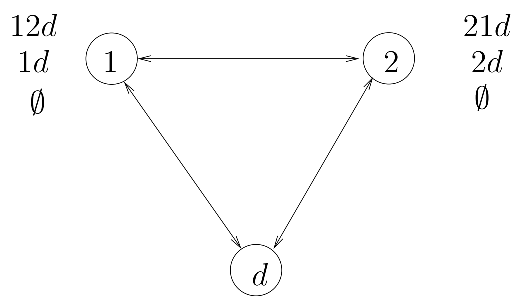
Each node i classifies paths according to its commercial relationship with the neighbor in the path (first hop): (1) **customer paths**, (2) **peer paths**, and (3) **provider paths**:



The top path is a customer path because the first hop is from i to a customer of i. Similarly, we have peer and provider paths (all neighbors of i can be grouped into these three classes). The preferences of each node i respect this classification:

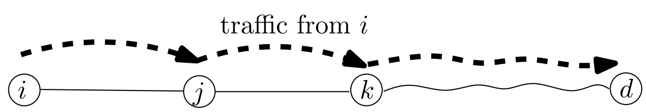
> **Gao-Rexford model (first version):**
> (GR1) provider paths $\prec$ peer paths $\prec$ customer paths

Dispute wheel is still possible:



> **Gao-Rexford model (second version):**
> (GR1) provider paths $\prec$ peer paths $\prec$ customer paths
> (GR2) transit traffic to/from my customers only

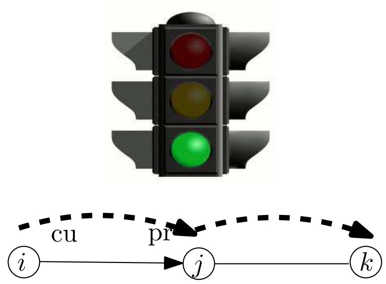
Consider this path:



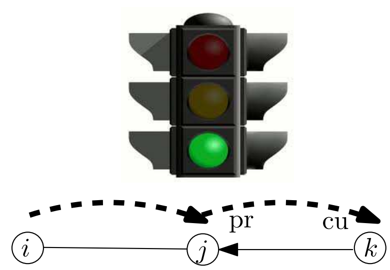
It may happen that node j does not allow **transit traffic** from node i:

- Node j chooses k as its next hop, but
- Node j does not forward the traffic coming from i to node j

There are **only two cases** where a node j allows transit traffic:

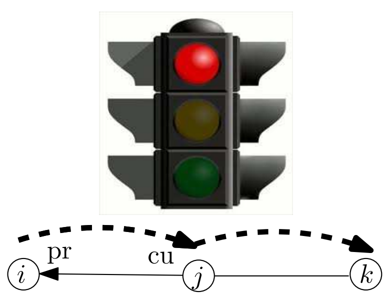


"from my customer"

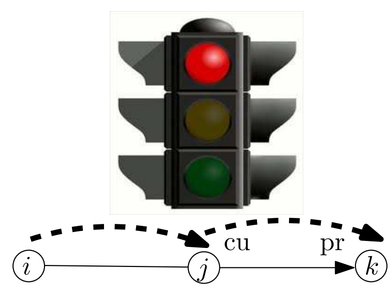


"to my customer"

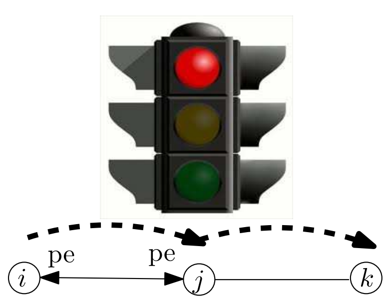
In all other cases a node j does **not allow transit traffic**:



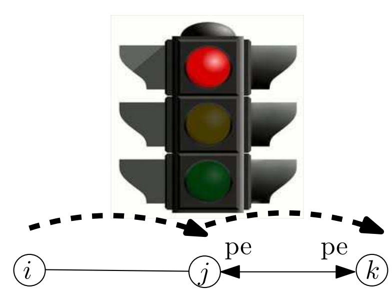
"from my provider"



"to my provider"



"from a peer"

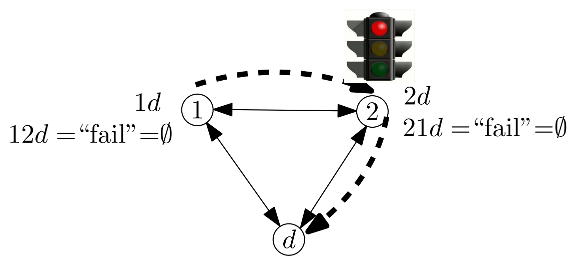


"to a peer"

No transit ⇒ zero utility

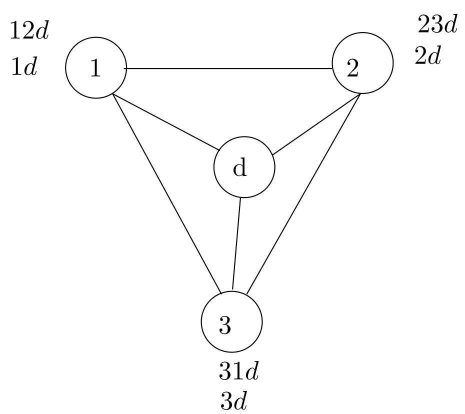
If node j does not allow transit traffic from node i then any path $P = i \to j \to \cdots d$ represents a "failure" for i which we denote with the symbol $\emptyset$. Such "failing" path have always the lowest utility 0.

Example 9 *Reconsider our previous example with all nodes having "peer-to-peer" relationships:*



Now we consider the path "12d" as a ***failure for node*** 1 *because its traffic will not be forwarded by node* 2*, though node* 2 *is forwarding its own traffic to* d*. Therefore the* ***preferences*** *of node* 1 *must be as shown in the picture. A similar argument holds for the path "21d" with respect to node* 2*.*

Exercise 12 Show that the following dispute wheel is still possible:



that is, these preferences do not necessarily violate conditions GR 1 and GR 2 (find the commercial relationships for which this is the case). ■

> **Gao-Rexford model (final version):**
> (GR1) $\emptyset \prec$ provider paths $\prec$ peer paths $\prec$ customer paths
> (GR2) transit traffic to/from my customers only
> (GR3) no customer-provider cycles

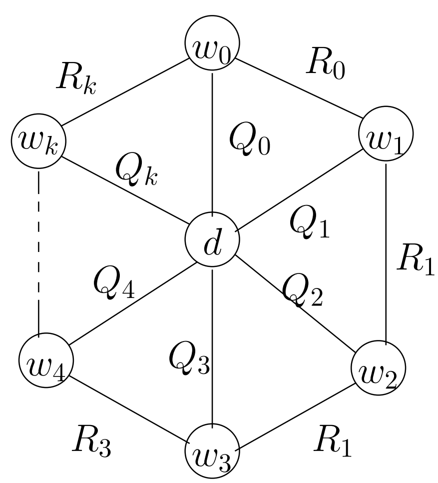
(GR3) says that no AS is indirectly a provider of itself.
(GR1) can be rewritten in terms of utilities as

$$0 = u_i(\emptyset) < u_i(\text{provider-path}) < u_i(\text{peer-path}) < u_i(\text{customer-path})$$

for any provider-path, any peer-path and any customer-path of i.

6.3 Gao-Rexford $\Longrightarrow$ No Dispute Wheel

We show that the network cannot contain nodes and paths that form a dispute wheel. We prove the result only for these simpler wheels (paths P_i and Q_i consist of a single link):



Recall that $\emptyset$ denotes any path that does not allow w_i to reach d (in particular if w_{i+1} does not allow transit traffic from w_i) and the utility is $u_{w_i}(\emptyset) = 0$. This and the preferences of the nodes

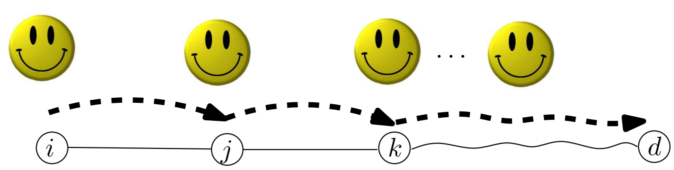
$$Q_i \prec_{w_i} R_i Q_{i+1}$$

imply that w_{i+1} must allow transit traffic from w_i. This is possible only in one of these two cases (GR2):

Exercise: show that in either case we must have a dispute wheel.

6.4 No Dispute Wheel $\Longrightarrow$ NBR-solvable with clear outcome

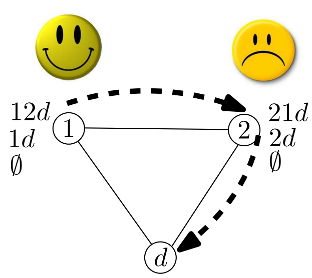
The key idea to construct an appropriate elimination sequence is to identify what we call "happy paths":



where

 = I got my top-ranked path

Here is an example of "unhappy" path (not all players are happy):

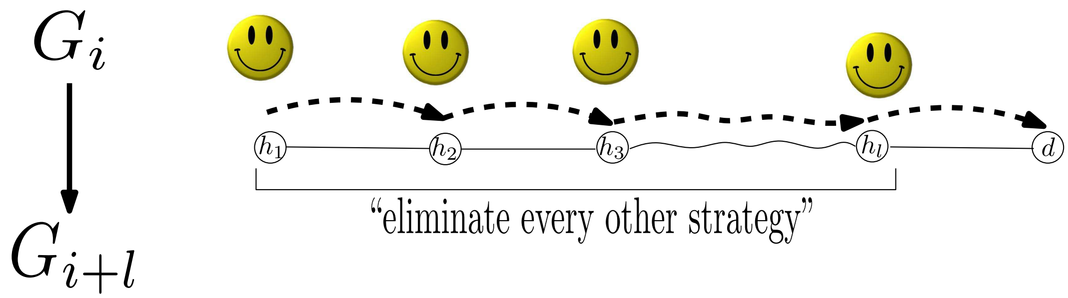


A path $h_1 \to h_2 \to \cdots \to h_l \to d$ in a subgame G_i is an **happy path** if this path gives the highest possible payoff to all of these nodes:

$$h_a \to h_{a+1} \to \cdots \to h_l \to d$$

is h_a's top ranked path among those that are available in the **sub**game G_i.

To see the idea of how happy paths give an elimination sequence:



The elimination sequence goes "**from right to left**":

1) h_l eliminates all strategies other than "$h_l \to d$" from the current subgame G_i and this gives us G_{i+1}. In this subgame G_{i+1} it is still true that the path is an happy path and thus h_{l-1} can eliminate all strategies other than "$h_{l-1} \to h_l$". We can continue until the first node in the happy path has eliminated all but the "$h_1 \to h_2$" strategy.

2) In the resulting subgame we find another happy path and repeat the previous step until there are no happy paths that start with a node with at least two strategies.

Suppose at the end of this process we included all nodes:

Every node belongs to some happy path. (3)

Then the final subgame consists of a game with one strategy per player. At each step we eliminate strategies that give the node a non-optimal payoff in the current subgame. So the starting game is NBR-solvable with clear outcome.

6.4.1 No Dispute Wheel ⇒ Condition (3)

We show that if there is no happy path then there must be a Dispute Wheel. Given that there is no happy path, starting from a node w_0 its top ranked path is *not* an happy path:

$$TR_{w_0} = w_0 \to i_1 \to \cdots w_1 \to i_a \to \cdots i_l \to d$$

and w_1 is the rightmost node (closest to d) for which the subpath

$$w_1 \to i_a \to \cdots \to i_l \to d$$

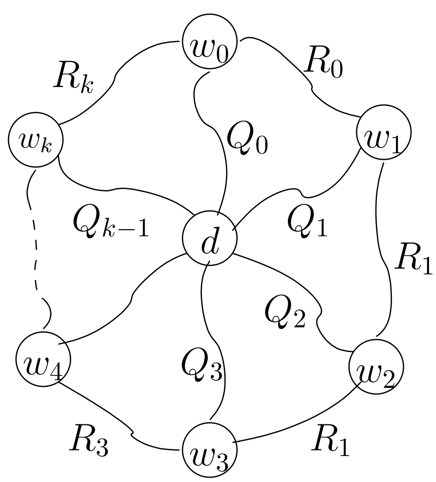
is not w_1's top ranked available path which is instead

$$TR_{w_1} = w_1 \to j_1 \to \cdots \to w_2 \to j_{a'} \to \cdots \to j_{l'} \to d$$

where w_2 is (again) the rightmost node in this path for which the corresponding subpath is not top ranked for it (this because there is no happy path). Since there is no happy path this can go on until we get some w_k such that

$$TR_{w_k} = w_1 \to n_1 \to \cdots w_{k+1} \to n_{a''} \to \cdots \to n_{l''} \to d$$

and w_{k+1} is one of the previously considered w_j's. For instance, if $w_{k+1} = w_0$ then we get the Dispute Wheel



by setting $R_iQ_{i+1} := TR_{w_i}$. If $w_{k+1} = w_s$ then we get a smaller Dispute Wheel with nodes $w_s, w_{s+1}, \ldots, w_k$.

BGP "in Practice" (Gao-Rexford model):   YES convergence + YES incentive compatible

References

The material of this lecture is taken from Nisan et al. (2011a) where you can find more details about the applications and further pointers into the literature. More details about the analysis of BGP and auctions can be find in Levin et al. (2008) and in Nisan et al. (2011b), respectively.

Hagay Levin, Michael Schapira, and Aviv Zohar. Interdomain routing and games. In *ACM Symposium on Theory of computing (STOC)*, page 57, 2008. URL http://www.cs.huji.ac.il/~schapiram/routing_games-full.pdf.

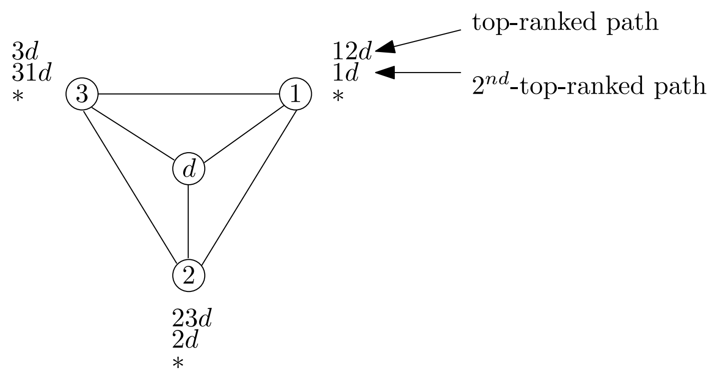
Noam Nisan, Michael Schapira, Gregory Valiant, and Aviv Zohar. Best-response mechanisms. In *Innovations in Computer Science (ICS)*, pages 155–165, 2011a. URL http://www.cs.huji.ac.il/~noam/BRM.pdf.

Noam Nisan, Michael Schapira, Gregory Valiant, and Aviv Zohar. Best-response auctions. In *Proceedings of the 12th ACM conference on Electronic commerce (EC)*, pages 351–360. ACM, 2011b. URL http://www.cs.huji.ac.il/~schapiram/ec087-nisan.pdf.

Exercises for Lecture 1

This exercise is on the games discussed in Example 1 in the lecture notes.

Exercise 1 (on BGP games) Consider the following simple instance of the "BGP games" (a network and the preferences of each node over the possible paths to d):



where the symbol "$*$" denotes any path connecting the node to d but different from the top two. The utilities correspond to the rank in the natural way:

> Top-ranked path has utility 3, second-top-ranked has utility 2, $*$ has utility 1, and the utility is 0 if the node does not reach d (e.g., if 3 and 1 point to each other, then their utility is 0).

Prove that the resulting game is NBR-solvable with clear outcome.

The next exercise is on the games in Section 3 of the lecture notes.

Exercise 2 (on TCP games) Consider a single channel of capacity C and two players with maximum sending rate M_1 and M_2. Prove that PIED converges and is incentive compatible (for any C and any M_1 and M_2).

Exercises for Lecture 2

Exercise 3 Consider auctions for selling one item and n bidders with valuations $v_1, \ldots, v_n$ of this item. Both the valuations and the possible bids belong to a set of discrete values:

$$\{0, \delta, 2\delta, \ldots, k\delta, \ldots\}$$

where $\delta > 0$.

Describe repeated 1^{st}-price auction as a best-response dynamics, and prove that it converges and is incentive compatible. Explain how you can deduce from this that 2^{nd}-price auction is truthful (reporting a bid different from the true valuation does not improve the utility of the corresponding player).
Note: You can assume that the auctioneer breaks ties in a fixed manner (if two or more bidders have the highest bid the auctioneer gives the item to the one with smallest index).

Exercise 4 Consider the game at page 5 of the lecture notes:

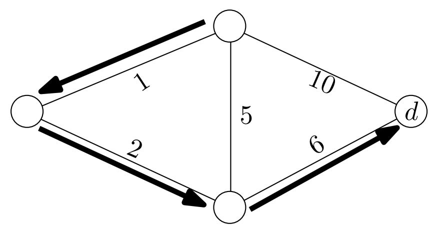
	A	B	C
a	2, 1	0, 0	0, 0
b	1, 2	1, -1	-1, 1
c	-1, -2	-1, 1	1, -1

Prove that best-response dynamics are **not incentive compatible** (i.e., show a starting state and an activation sequence for which one player can improve his/her total utility by deviating from best-response).

Exercise 5 Given a network $G = (V, E)$ in which every link $e \in E$ has some delay c_e and each node is a router, we would like to compute a tree directed towards a fixed (given) destination $d \in V$ having **minimal delays**:

> For every node i, the path from i to d in the tree has minimal delay (the delay of a path is the sum of all delays of its links), that is, every other path from i to d in G has at least the same delay.

Like in BGP, each node can only choose the next hop, i.e., select an outgoing link towards one of its neighbors. The following picture shows an instance and the corresponding tree with minimal delays:



Describe a distributed protocol (what each node i is prescribed to do) for computing a tree having minimal delays on any network with any delays on the links. Prove that your protocol converges in a finite number of *rounds* to the required solution.

Note: We are not assuming the nodes to be selfish or to have a particular preference order over the paths (you can assume they follow what you prescribe them to do). The setting is asynchronous meaning that an adversary chooses the initial state (each node links to some neighbor) and which nodes are activated at each time step (only active nodes execute your protocol in that time step, but each node is activated infinitely many times).