Self-Ratings and Peer Review

Leonie Baumann

McGill University

Setting

A principal has to assign a prize and wants to give it to the best agent among many agents:

Setting

A principal has to assign a prize and wants to give it to the best agent among many agents:

- An employer has to hire one applicant,
  a manager has to promote one employee,
  a committee has to assign a prize to one candidate...

Setting

A principal has to assign a prize and wants to give it to the best agent among many agents:

- An employer has to hire one applicant,
  a manager has to promote one employee,
  a committee has to assign a prize to one candidate...

The principal does not know the true value of any applicant, only knows the distribution of values. Agents (exactly) know their own value and might have knowledge about other candidates:

Setting

A principal has to assign a prize and wants to give it to the best agent among many agents:

- An employer has to hire one applicant,
  a manager has to promote one employee,
  a committee has to assign a prize to one candidate...

The principal does not know the true value of any applicant, only knows the distribution of values. Agents (exactly) know their own value and might have knowledge about other candidates:

- Two people who have worked together know each other's abilities, ...

Setting

A principal has to assign a prize and wants to give it to the best agent among many agents:

- An employer has to hire one applicant,
  a manager has to promote one employee,
  a committee has to assign a prize to one candidate...

The principal does not know the true value of any applicant, only knows the distribution of values. Agents (exactly) know their own value and might have knowledge about other candidates:

- Two people who have worked together know each other's abilities, ...

Agents have a **knowledge network**.

Goal of the Paper

Design a mechanism such that the principal always assigns the prize to the best agent in every equilibrium (full implementation),

Goal of the Paper

Design a mechanism such that the principal always assigns the prize to the best agent in every equilibrium (full implementation), when

- agents send private and costless messages to the principal with statements about themselves and about their neighbors (application and references),

Goal of the Paper

Design a mechanism such that the principal always assigns the prize to the best agent in every equilibrium (full implementation), when

- agents send private and costless messages to the principal with statements about themselves and about their neighbors (application and references),
- agents can lie in each statement but only to a certain extent

Goal of the Paper

Design a mechanism such that the principal always assigns the prize to the best agent in every equilibrium (full implementation), when

- agents send private and costless messages to the principal with statements about themselves and about their neighbors (application and references),
- agents can lie in each statement but only to a certain extent ($\Rightarrow$ revelation principle fails), and

Goal of the Paper

Design a mechanism such that the principal always assigns the prize to the best agent in every equilibrium (full implementation), when

- agents send private and costless messages to the principal with statements about themselves and about their neighbors (application and references),
- agents can lie in each statement but only to a certain extent ($\Rightarrow$ revelation principle fails), and
- the principal does not dispose of the prize and does not use monetary transfers.

Goal of the Paper

Design a mechanism such that the principal always assigns the prize to the best agent in every equilibrium (full implementation), when

- agents send private and costless messages to the principal with statements about themselves and about their neighbors (application and references),
- agents can lie in each statement but only to a certain extent ($\Rightarrow$ revelation principle fails), and
- the principal does not dispose of the prize and does not use monetary transfers.

A mechanism specifies a probability of getting the prize for every agent for every set of messages the principal might receive.

Two Specific Applications

Peer review processes in academia:

- researchers apply with a paper, proposal... to a conference, journal, institution... for presentation, awards, publication, grants,... and they write referee reports about competitors

Two Specific Applications

Peer review processes in academia:

- researchers apply with a paper, proposal... to a conference, journal, institution... for presentation, awards, publication, grants,... and they write referee reports about competitors

Employee evaluation processes in firms:

- 360 degree feedback: self- and peer-evaluations (e.g. co-workers) are used to decide on payments and promotion
- in 2013, 34% among 112 companies, including Boeing, Monsanto, PepsiCo, FedEx and Dell use 360 degree feedback (survey by 3D Group)

Two Specific Applications

Peer review processes in academia:

- researchers apply with a paper, proposal... to a conference, journal, institution... for presentation, awards, publication, grants,... and they write referee reports about competitors

Employee evaluation processes in firms:

- 360 degree feedback: self- and peer-evaluations (e.g. co-workers) are used to decide on payments and promotion
- in 2013, 34% among 112 companies, including Boeing, Monsanto, PepsiCo, FedEx and Dell use 360 degree feedback (survey by 3D Group)

How to interpret applications/self-evaluations and reports/peer-evaluations to always identify the most deserving candidate? Which “mechanism”?

Contribution and Preview of Results

- new framework for a specific mechanism design/ implementation problem with a network setting

Contribution and Preview of Results

- new framework for a specific mechanism design/ implementation problem with a network setting
- show that traditional mechanisms for full implementation (relying on Maskin-/ Bayesian-monotonicity) and a mechanism which assigns prize as function of either applications only or references only fail

Contribution and Preview of Results

- new framework for a specific mechanism design/ implementation problem with a network setting
- show that traditional mechanisms for full implementation (relying on Maskin-/ Bayesian-monotonicity) and a mechanism which assigns prize as function of either applications only or references only fail
- provide first mechanism for full implementation in this setting for a class of networks

Contribution and Preview of Results

- new framework for a specific mechanism design/ implementation problem with a network setting
- show that traditional mechanisms for full implementation (relying on Maskin-/ Bayesian-monotonicity) and a mechanism which assigns prize as function of either applications only or references only fail
- provide first mechanism for full implementation in this setting for a class of networks
- a mechanism which assigns the prize as a function of best applications and worst references fully implements iff network is complete

Contribution and Preview of Results

- new framework for a specific mechanism design/ implementation problem with a network setting
- show that traditional mechanisms for full implementation (relying on Maskin-/ Bayesian-monotonicity) and a mechanism which assigns prize as function of either applications only or references only fail
- provide first mechanism for full implementation in this setting for a class of networks
- a mechanism which assigns the prize as a function of best applications and worst references fully implements iff network is complete
- for a class of incomplete networks (sufficient conditions), an extension of the mechanism fully implements, if agents are partially honest

Contribution and Preview of Results

- new framework for a specific mechanism design/ implementation problem with a network setting
- show that traditional mechanisms for full implementation (relying on Maskin-/ Bayesian-monotonicity) and a mechanism which assigns prize as function of either applications only or references only fail
- provide first mechanism for full implementation in this setting for a class of networks
- a mechanism which assigns the prize as a function of best applications and worst references fully implements iff network is complete
- for a class of incomplete networks (sufficient conditions), an extension of the mechanism fully implements, if agents are partially honest
- the mechanism fully implements in expectation for every network, if communication is noisy

Related Literature

Mechanism design for allocation, implementation, and persuasion problems with **(partially) verifiable information**, one principal and many agents.

Related Literature

Mechanism design for allocation, implementation, and persuasion problems with **(partially) verifiable information**, one principal and many agents.

- Lipman and Seppi (1995), Glazer and Rubinstein (2001), Deneckere and Severinov (2008), Ben-Porath and Lipman (2012), Kartik and Tercieux (2012), Koessler and Perez-Richet (2017):
  No underlying knowledge network.
- This paper: an agent's private information and message space depend on her position in a network.

Related Literature

Mechanism design for allocation, implementation, and persuasion problems with **(partially) verifiable information**, one principal and many agents.

- Lipman and Seppi (1995), Glazer and Rubinstein (2001), Deneckere and Severinov (2008), Ben-Porath and Lipman (2012), Kartik and Tercieux (2012), Koessler and Perez-Richet (2017):
  No underlying knowledge network.
- This paper: an agent's private information and message space depend on her position in a network.

Implementation with **partially honest** agents.

Related Literature

Mechanism design for allocation, implementation, and persuasion problems with **(partially) verifiable information**, one principal and many agents.

- Lipman and Seppi (1995), Glazer and Rubinstein (2001), Deneckere and Severinov (2008), Ben-Porath and Lipman (2012), Kartik and Tercieux (2012), Koessler and Perez-Richet (2017):
  No underlying knowledge network.
- This paper: an agent's private information and message space depend on her position in a network.

Implementation with **partially honest** agents.

- Dutta and Sen (2012), no underlying knowledge network. Korpela (2014), Bayesian setting.
- This paper: knowledge network, no integer mechanisms, no incentive-compatibility.

Mechanism Design and **Networks**.

Mechanism Design and **Networks**.

- Renou and Tomala (2012): implementation for different communication networks among agents and mechanism designer
- Dziubiński, Sankowski, Zhang (2016): optimal network protection when the network is unknown
- Bloch and Olckers (2018): elicit full ordinal ranking, one desirable equilibrium, incentive-compatibility
- This paper: communication network is a star with principal as center; network is public knowledge; cardinal problem, all equilibria, no incentive compatibility

Model

One Principal and Many Heterogeneous Agents

- A principal P has to assign a prize to one agent.

One Principal and Many Heterogeneous Agents

- A principal P has to assign a prize to one agent.
- Set of agents: $N = \{1, ..., n\}$ with $n \geq 3$

One Principal and Many Heterogeneous Agents

- A principal P has to assign a prize to one agent.
- Set of agents: $N = \{1, ..., n\}$ with $n \geq 3$
- Every agent wants the prize.

One Principal and Many Heterogeneous Agents

- A principal P has to assign a prize to one agent.
- Set of agents: $N = \{1, ..., n\}$ with $n \geq 3$
- Every agent wants the prize.
- Agents are heterogeneously suited to receive the prize:

One Principal and Many Heterogeneous Agents

- A principal P has to assign a prize to one agent.
- Set of agents: $N = \{1, ..., n\}$ with $n \geq 3$
- Every agent wants the prize.
- Agents are heterogeneously suited to receive the prize:
  - an agent's suitability for the prize is measured by her distance to the ideal point

One Principal and Many Heterogeneous Agents

- A principal P has to assign a prize to one agent.
- Set of agents: $N = \{1, ..., n\}$ with $n \geq 3$
- Every agent wants the prize.
- Agents are heterogeneously suited to receive the prize:
  - an agent's suitability for the prize is measured by her distance to the ideal point
  - agent i's distance to the ideal is $d_i \in [0, 1]$, 0 is best, 1 is worst

One Principal and Many Heterogeneous Agents

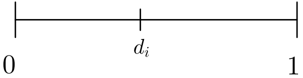
- A principal P has to assign a prize to one agent.
- Set of agents: $N = \{1, ..., n\}$ with $n \geq 3$
- Every agent wants the prize.
- Agents are heterogeneously suited to receive the prize:
  - an agent's suitability for the prize is measured by her distance to the ideal point
  - agent i's distance to the ideal is $d_i \in [0, 1]$, 0 is best, 1 is worst

    iid from continuous distribution with full support on $[0, 1]$ (common knowledge),

One Principal and Many Heterogeneous Agents

- A principal P has to assign a prize to one agent.
- Set of agents: $N = \{1, ..., n\}$ with $n \geq 3$
- Every agent wants the prize.
- Agents are heterogeneously suited to receive the prize:
  - an agent's suitability for the prize is measured by her distance to the ideal point
  - agent i's distance to the ideal is $d_i \in [0, 1]$, 0 is best, 1 is worst

    iid from continuous distribution with full support on $[0, 1]$ (common knowledge),



One Principal and Many Heterogeneous Agents

- A principal P has to assign a prize to one agent.
- Set of agents: $N = \{1, ..., n\}$ with $n \geq 3$
- Every agent wants the prize.
- Agents are heterogeneously suited to receive the prize:
  - an agent's suitability for the prize is measured by her distance to the ideal point
  - agent i's distance to the ideal is $d_i \in [0, 1]$, 0 is best, 1 is worst

    iid from continuous distribution with full support on $[0, 1]$ (common knowledge),
  - focus on $d_i \neq d_j$ for all $j \neq i$ (prob. 1)

One Principal and Many Heterogeneous Agents

- A principal P has to assign a prize to one agent.
- Set of agents: $N = \{1, ..., n\}$ with $n \geq 3$
- Every agent wants the prize.
- Agents are heterogeneously suited to receive the prize:
  - an agent's suitability for the prize is measured by her distance to the ideal point
  - agent i's distance to the ideal is $d_i \in [0, 1]$, 0 is best, 1 is worst

    iid from continuous distribution with full support on $[0, 1]$ (common knowledge),
  - focus on $d_i \neq d_j$ for all $j \neq i$ (prob. 1)
- P's utility is maximized if and only if the prize is assigned to the *global minimum* g with $d_g = \min_{i \in N} d_i$

Information Structure

- P does not know d_i of any $i \in N$, only the distribution.

Information Structure

- P does not know d_i of any $i \in N$, only the distribution.
- Every i perfectly knows d_i and the distances of other agents whom she is linked to:

Information Structure

- P does not know d_i of any $i \in N$, only the distribution.
- Every i perfectly knows d_i and the distances of other agents whom she is linked to:
    - Undirected "knowledge" network, defined by a set of links L among agents. L is common knowledge.

Information Structure

- P does not know d_i of any $i \in N$, only the distribution.
- Every i perfectly knows d_i and the distances of other agents whom she is linked to:
  - Undirected "knowledge" network, defined by a set of links L among agents. L is common knowledge.
  - If link $ij \in L$, then i perfectly knows d_j, and j perfectly knows d_i.

Information Structure

- P does not know d_i of any $i \in N$, only the distribution.
- Every i perfectly knows d_i and the distances of other agents whom she is linked to:
  - Undirected "knowledge" network, defined by a set of links L among agents. L is common knowledge.
  - If link $ij \in L$, then i perfectly knows d_j, and j perfectly knows d_i.
  - If link $ij \notin L$, then i does not know d_j and vice versa, they only know the distribution.

Information Structure

- P does not know d_i of any $i \in N$, only the distribution.
- Every i perfectly knows d_i and the distances of other agents whom she is linked to:
  - Undirected "knowledge" network, defined by a set of links L among agents. L is common knowledge.
  - If link $ij \in L$, then i perfectly knows d_j, and j perfectly knows d_i.
  - If link $ij \notin L$, then i does not know d_j and vice versa, they only know the distribution.
  - Set of neighbors of agent i: $N_i = \{j | ij \in L\}$.
    Assume: Every agent has at least one neighbor.

Information Structure

- P does not know d_i of any $i \in N$, only the distribution.
- Every i perfectly knows d_i and the distances of other agents whom she is linked to:
  - Undirected "knowledge" network, defined by a set of links L among agents. L is common knowledge.
  - If link $ij \in L$, then i perfectly knows d_j, and j perfectly knows d_i.
  - If link $ij \notin L$, then i does not know d_j and vice versa, they only know the distribution.
  - Set of neighbors of agent i: $N_i = \{j | ij \in L\}$.
    Assume: Every agent has at least one neighbor.
- Agent i has type $\theta_i = (d_i, (d_j)_{j \in N_i})$ with type space $\Theta_i = [0,1]^{|N_i|+1}$, a type profile is θ and space of type profiles Θ

Information Structure

- P does not know d_i of any $i \in N$, only the distribution.
- Every i perfectly knows d_i and the distances of other agents whom she is linked to:
  - Undirected "knowledge" network, defined by a set of links L among agents. L is common knowledge.
  - If link $ij \in L$, then i perfectly knows d_j, and j perfectly knows d_i.
  - If link $ij \notin L$, then i does not know d_j and vice versa, they only know the distribution.
  - Set of neighbors of agent i: $N_i = \{j | ij \in L\}$.
    Assume: Every agent has at least one neighbor.
- Agent i has type $\theta_i = (d_i, (d_j)_{j \in N_i})$ with type space $\Theta_i = [0,1]^{|N_i|+1}$, a type profile is θ and space of type profiles Θ
- given θ_i, i's expectation that others' types are $\theta_{-i} \in \Theta'_{-i}$ is conditional probability $p(\Theta'_{-i} | \theta_i)$ derived from distribution of distances

Mechanism

- P chooses, announces and commits to a mechanism for the prize allocation, before agents choose their message strategies

Mechanism

- P chooses, announces and commits to a mechanism for the prize allocation, before agents choose their message strategies
- a mechanism is a pair (M, π) where M is the space of message profiles and π an outcome function (prize allocation rule)

Mechanism

- P chooses, announces and commits to a mechanism for the prize allocation, before agents choose their message strategies
- a mechanism is a pair (M, π) where M is the space of message profiles and π an outcome function (prize allocation rule)
  - $M = \prod_i M_i$ with agent i's message space $M_i = \Theta_i$

Mechanism

- P chooses, announces and commits to a mechanism for the prize allocation, before agents choose their message strategies
- a mechanism is a pair (M, π) where M is the space of message profiles and π an outcome function (prize allocation rule)
  - $M = \prod_i M_i$ with agent i's message space $M_i = \Theta_i$
  - any $m_i \in M_i$ is $m_i = \left(m_{ii}, (m_{ij})_{j \in N_i}\right)$ with $m_{ii}, m_{ij} \in [0, 1]$ for all $j \in N_i$

Mechanism

- P chooses, announces and commits to a mechanism for the prize allocation, before agents choose their message strategies
- a mechanism is a pair (M, π) where M is the space of message profiles and π an outcome function (prize allocation rule)
  - $M = \prod_i M_i$ with agent i's message space $M_i = \Theta_i$
  - any $m_i \in M_i$ is $m_i = \left(m_{ii}, (m_{ij})_{j \in N_i}\right)$ with $m_{ii}, m_{ij} \in [0, 1]$ for all $j \in N_i$
  - m_{ii} is i's *application*, m_{ij} is i's *reference* about j

Mechanism

- P chooses, announces and commits to a mechanism for the prize allocation, before agents choose their message strategies
- a mechanism is a pair (M, π) where M is the space of message profiles and π an outcome function (prize allocation rule)
  - $M = \prod_i M_i$ with agent i's message space $M_i = \Theta_i$
  - any $m_i \in M_i$ is $m_i = \left(m_{ii}, (m_{ij})_{j \in N_i}\right)$ with $m_{ii}, m_{ij} \in [0, 1]$ for all $j \in N_i$
  - m_{ii} is i's *application*, m_{ij} is i's *reference* about j
  - $\pi : M \to [0, 1]^n$ such that $\sum_{i \in N} \pi_i(m) = 1$ for all $m \in M$

Mechanism

- P chooses, announces and commits to a mechanism for the prize allocation, before agents choose their message strategies
- a mechanism is a pair (M, π) where M is the space of message profiles and π an outcome function (prize allocation rule)
  - $M = \prod_i M_i$ with agent i's message space $M_i = \Theta_i$
  - any $m_i \in M_i$ is $m_i = \left(m_{ii}, (m_{ij})_{j \in N_i}\right)$ with $m_{ii}, m_{ij} \in [0,1]$ for all $j \in N_i$
  - m_{ii} is i's *application*, m_{ij} is i's *reference* about j
  - $\pi : M \to [0,1]^n$ such that $\sum_{i \in N} \pi_i(m) = 1$ for all $m \in M$
  - $\pi(m)$ is a probability distribution over all agents with $\pi_i(m)$ being the probability with which i receives the prize given message profile m

Mechanism

- P chooses, announces and commits to a mechanism for the prize allocation, before agents choose their message strategies
- a mechanism is a pair (M, π) where M is the space of message profiles and π an outcome function (prize allocation rule)
  - $M = \prod_i M_i$ with agent i's message space $M_i = \Theta_i$
  - any $m_i \in M_i$ is $m_i = \left(m_{ii}, (m_{ij})_{j \in N_i}\right)$ with $m_{ii}, m_{ij} \in [0,1]$ for all $j \in N_i$
  - m_{ii} is i's *application*, m_{ij} is i's *reference* about j
  - $\pi : M \to [0,1]^n$ such that $\sum_{i \in N} \pi_i(m) = 1$ for all $m \in M$
  - $\pi(m)$ is a probability distribution over all agents with $\pi_i(m)$ being the probability with which i receives the prize given message profile m
- M is fixed and P only chooses π, so we refer to π as the mechanism

Mechanism

- P chooses, announces and commits to a mechanism for the prize allocation, before agents choose their message strategies
- a mechanism is a pair (M, π) where M is the space of message profiles and π an outcome function (prize allocation rule)
  - $M = \prod_i M_i$ with agent i's message space $M_i = \Theta_i$
  - any $m_i \in M_i$ is $m_i = \left(m_{ii}, (m_{ij})_{j \in N_i}\right)$ with $m_{ii}, m_{ij} \in [0,1]$ for all $j \in N_i$
  - m_{ii} is i's *application*, m_{ij} is i's *reference* about j
  - $\pi : M \to [0,1]^n$ such that $\sum_{i \in N} \pi_i(m) = 1$ for all $m \in M$
  - $\pi(m)$ is a probability distribution over all agents with $\pi_i(m)$ being the probability with which i receives the prize given message profile m
- M is fixed and P only chooses π, so we refer to π as the mechanism
- the announcement of π induces a static game $\Gamma(\pi)$ among the agents

Static Game $\Gamma(\pi)$ among the Agents

- agent i's action set at θ_i is message set $M_i(\theta_i) \subset M_i$

Static Game $\Gamma(\pi)$ among the Agents

- agent i's action set at θ_i is message set $M_i(\theta_i) \subset M_i$
  - $m_i \in M_i(\theta_i)$ if and only if

Static Game $\Gamma(\pi)$ among the Agents

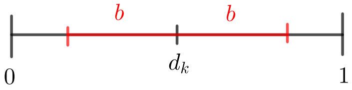
- agent i's action set at θ_i is message set $M_i(\theta_i) \subset M_i$
  - $m_i \in M_i(\theta_i)$ if and only if $m_{ik} \in [\max\{0, d_k - b\}, \min\{d_k + b, 1\}]$ for all $m_{ik} \in m_i$

Static Game $\Gamma(\pi)$ among the Agents

- agent i's action set at θ_i is message set $M_i(\theta_i) \subset M_i$
  - $m_i \in M_i(\theta_i)$ if and only if $m_{ik} \in [\max\{0, d_k - b\}, \min\{d_k + b, 1\}]$ for all $m_{ik} \in m_i$ with exogenous $b \in (0, \frac{1}{2})$ which is common knowledge

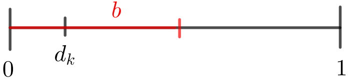
Static Game $\Gamma(\pi)$ among the Agents

- agent i's action set at θ_i is message set $M_i(\theta_i) \subset M_i$
  - $m_i \in M_i(\theta_i)$ if and only if $m_{ik} \in [\max\{0, d_k - b\}, \min\{d_k + b, 1\}]$ for all $m_{ik} \in m_i$ with exogenous $b \in (0, \frac{1}{2})$ which is common knowledge



Static Game $\Gamma(\pi)$ among the Agents

- agent i's action set at θ_i is message set $M_i(\theta_i) \subset M_i$
  - $m_i \in M_i(\theta_i)$ if and only if $m_{ik} \in [\max\{0, d_k - b\}, \min\{d_k + b, 1\}]$ for all $m_{ik} \in m_i$ with exogenous $b \in (0, \frac{1}{2})$ which is common knowledge



Static Game $\Gamma(\pi)$ among the Agents

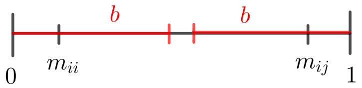
- agent i's action set at θ_i is message set $M_i(\theta_i) \subset M_i$
  - $m_i \in M_i(\theta_i)$ if and only if $m_{ik} \in [\max\{0, d_k - b\}, \min\{d_k + b, 1\}]$ for all $m_{ik} \in m_i$ with exogenous $b \in (0, \frac{1}{2})$ which is common knowledge
  - agents can lie about each true distance maximally by $\pm b$

Static Game $\Gamma(\pi)$ among the Agents

- agent i's action set at θ_i is message set $M_i(\theta_i) \subset M_i$
  - $m_i \in M_i(\theta_i)$ if and only if $m_{ik} \in [\max\{0, d_k - b\}, \min\{d_k + b, 1\}]$ for all $m_{ik} \in m_i$ with exogenous $b \in (0, \frac{1}{2})$ which is common knowledge
  - agents can lie about each true distance maximally by $\pm b$
  - agent i "proves" to be better than j if $m_{ij} - m_{ii} > 2b$

Static Game $\Gamma(\pi)$ among the Agents

- agent i's action set at θ_i is message set $M_i(\theta_i) \subset M_i$
  - $m_i \in M_i(\theta_i)$ if and only if $m_{ik} \in [\max\{0, d_k - b\}, \min\{d_k + b, 1\}]$ for all $m_{ik} \in m_i$ with exogenous $b \in (0, \frac{1}{2})$ which is common knowledge
  - agents can lie about each true distance maximally by $\pm b$
  - agent i "proves" to be better than j if $m_{ij} - m_{ii} > 2b$



Static Game $\Gamma(\pi)$ among the Agents

- agent i's action set at θ_i is message set $M_i(\theta_i) \subset M_i$
    - $m_i \in M_i(\theta_i)$ if and only if $m_{ik} \in [\max\{0, d_k - b\}, \min\{d_k + b, 1\}]$ for all $m_{ik} \in m_i$ with exogenous $b \in (0, \frac{1}{2})$ which is common knowledge
    - agents can lie about each true distance maximally by $\pm b$
    - agent i "proves" to be better than j if $m_{ij} - m_{ii} > 2b$

Static Game $\Gamma(\pi)$ among the Agents

- agent i's action set at θ_i is message set $M_i(\theta_i) \subset M_i$
  - $m_i \in M_i(\theta_i)$ if and only if $m_{ik} \in [\max\{0, d_k - b\}, \min\{d_k + b, 1\}]$ for all $m_{ik} \in m_i$ with exogenous $b \in (0, \frac{1}{2})$ which is common knowledge
  - agents can lie about each true distance maximally by $\pm b$
  - agent i "proves" to be better than j if $m_{ij} - m_{ii} > 2b$
  - messages are sent privately to the principal

Static Game $\Gamma(\pi)$ among the Agents

- agent i's action set at θ_i is message set $M_i(\theta_i) \subset M_i$
  - $m_i \in M_i(\theta_i)$ if and only if $m_{ik} \in [\max\{0, d_k - b\}, \min\{d_k + b, 1\}]$ for all $m_{ik} \in m_i$ with exogenous $b \in (0, \frac{1}{2})$ which is common knowledge
  - agents can lie about each true distance maximally by $\pm b$
  - agent i "proves" to be better than j if $m_{ij} - m_{ii} > 2b$
  - messages are sent privately to the principal
- agent i's strategy $\hat{m}_i$ specifies one $m_i \in M_i(\theta_i)$ for each $\theta_i \in \Theta_i$, pure strategies only; strategy profile is $\hat{m}$, others' strategies are $\hat{m}_{-i}$

Static Game $\Gamma(\pi)$ among the Agents

- agent i's action set at θ_i is message set $M_i(\theta_i) \subset M_i$
  - $m_i \in M_i(\theta_i)$ if and only if
    $m_{ik} \in [\max\{0, d_k - b\}, \min\{d_k + b, 1\}]$ for all $m_{ik} \in m_i$
    with exogenous $b \in (0, \frac{1}{2})$ which is common knowledge
  - agents can lie about each true distance maximally by $\pm b$
  - agent i "proves" to be better than j if $m_{ij} - m_{ii} > 2b$
  - messages are sent privately to the principal
- agent i's strategy $\hat{m}_i$ specifies one $m_i \in M_i(\theta_i)$ for each $\theta_i \in \Theta_i$, pure strategies only; strategy profile is $\hat{m}$, others' strategies are $\hat{m}_{-i}$
- agent i's exp. utility (probability of winning) at θ_i from $m_i \in M_i(\theta_i)$ given $\hat{m}_{-i}$ is

$$\Pi_i(m_i, \hat{m}_{-i}|\theta_i) = \int_{\theta_{-i}} \pi_i(m_i, \hat{m}_{-i}(\theta_{-i}))\, dp(\theta_{-i}|\theta_i).$$

Equilibrium of $\Gamma(\pi)$ and Full Implementation

- A strategy profile $\hat{m}$ is a Bayesian Nash equilibrium of $\Gamma(\pi)$ if and only if for all i, $m_i \in M_i(\theta_i)$, and θ_i

$$\Pi_i(\hat{m}_i(\theta_i), \hat{m}_{-i}|\theta_i) \geq \Pi_i(m_i, \hat{m}_{-i}|\theta_i).$$

Equilibrium of $\Gamma(\pi)$ and Full Implementation

- A strategy profile $\hat{m}$ is a Bayesian Nash equilibrium of $\Gamma(\pi)$ if and only if for all i, $m_i \in M_i(\theta_i)$, and θ_i

$$\Pi_i(\hat{m}_i(\theta_i), \hat{m}_{-i}|\theta_i) \geq \Pi_i(m_i, \hat{m}_{-i}|\theta_i).$$

- Mechanism π **fully implements** in L, if **every equilibrium** $\hat{m}$ of $\Gamma(\pi)$ is such that $\pi_g(\hat{m}(\theta)) = 1$ with prob. 1.

Equilibrium of $\Gamma(\pi)$ and Full Implementation

- A strategy profile $\hat{m}$ is a Bayesian Nash equilibrium of $\Gamma(\pi)$ if and only if for all i, $m_i \in M_i(\theta_i)$, and θ_i

$$\Pi_i(\hat{m}_i(\theta_i), \hat{m}_{-i}|\theta_i) \geq \Pi_i(m_i, \hat{m}_{-i}|\theta_i).$$

- Mechanism π **fully implements** in L, if **every equilibrium** $\hat{m}$ of $\Gamma(\pi)$ is such that $\pi_g(\hat{m}(\theta)) = 1$ with prob. 1.

- Goal: Design π such that π fully implements for as many L as possible.

Full Implementation

Full Implementation in the Complete Network

Full Implementation in the Complete Network

Definition (Applications-And-References Mechanism π^{so})

Full Implementation in the Complete Network

Definition (Applications-And-References Mechanism π^{so})

For any $m \in M$,

Full Implementation in the Complete Network

Definition (Applications-And-References Mechanism π^{so})

For any $m \in M$,

first, identify all agents who send the best application ($B_1 \subseteq N$):

Full Implementation in the Complete Network

Definition (Applications-And-References Mechanism π^{so})

For any $m \in M$,

first, identify all agents who send the best application ($B_1 \subseteq N$):

$$B_1 \subseteq N \text{ such that } i \in B_1 \text{ iff } m_{ii} = \min_{k \in N} m_{kk}$$

Full Implementation in the Complete Network

Definition (Applications-And-References Mechanism π^{so})

For any $m \in M$,

first, identify all agents who send the best application ($B_1 \subseteq N$):

$$B_1 \subseteq N \text{ such that } i \in B_1 \text{ iff } m_{ii} = \min_{k \in N} m_{kk}$$

Second, choose $B_2 \subseteq B_1$ such that

Full Implementation in the Complete Network

Definition (Applications-And-References Mechanism π^{so})

For any $m \in M$,

first, identify all agents who send the best application ($B_1 \subseteq N$):

$$B_1 \subseteq N \text{ such that } i \in B_1 \text{ iff } m_{ii} = \min_{k \in N} m_{kk}$$

Second, choose $B_2 \subseteq B_1$ such that

- if $\min_{k \in N} m_{kk} > 0$, then $B_2 = B_1$.

Full Implementation in the Complete Network

Definition (Applications-And-References Mechanism π^{so})

For any $m \in M$,

first, identify all agents who send the best application ($B_1 \subseteq N$):

$$B_1 \subseteq N \text{ such that } i \in B_1 \text{ iff } m_{ii} = \min_{k \in N} m_{kk}$$

Second, choose $B_2 \subseteq B_1$ such that

- if $\min_{k \in N} m_{kk} > 0$, then $B_2 = B_1$.
- if $\min_{k \in N} m_{kk} = 0$, then P looks at the references which agents in B_1 receive:

Full Implementation in the Complete Network

Definition (Applications-And-References Mechanism π^{so})

For any $m \in M$,

first, identify all agents who send the best application ($B_1 \subseteq N$):

$$B_1 \subseteq N \text{ such that } i \in B_1 \text{ iff } m_{ii} = \min_{k \in N} m_{kk}$$

Second, choose $B_2 \subseteq B_1$ such that

- if $\min_{k \in N} m_{kk} > 0$, then $B_2 = B_1$.
- if $\min_{k \in N} m_{kk} = 0$, then P looks at the references which agents in B_1 receive:
  - worst reference which i receives: $\bar{r}_i = \max_{j \in N_i} m_{ji}$

Full Implementation in the Complete Network

Definition (Applications-And-References Mechanism π^{so})

For any $m \in M$,

first, identify all agents who send the best application ($B_1 \subseteq N$):

$$B_1 \subseteq N \text{ such that } i \in B_1 \text{ iff } m_{ii} = \min_{k \in N} m_{kk}$$

Second, choose $B_2 \subseteq B_1$ such that

- if $\min_{k \in N} m_{kk} > 0$, then $B_2 = B_1$.
- if $\min_{k \in N} m_{kk} = 0$, then P looks at the references which agents in B_1 receive:
  - worst reference which i receives: $\bar{r}_i = \max_{j \in N_i} m_{ji}$
  - $i \in B_2$ iff

Full Implementation in the Complete Network

Definition (Applications-And-References Mechanism π^{so})

For any $m \in M$,

first, identify all agents who send the best application ($B_1 \subseteq N$):

$$B_1 \subseteq N \text{ such that } i \in B_1 \text{ iff } m_{ii} = \min_{k \in N} m_{kk}$$

Second, choose $B_2 \subseteq B_1$ such that

- if $\min_{k \in N} m_{kk} > 0$, then $B_2 = B_1$.
- if $\min_{k \in N} m_{kk} = 0$, then P looks at the references which agents in B_1 receive:
  - worst reference which i receives: $\bar{r}_i = \max_{j \in N_i} m_{ji}$
  - $i \in B_2$ iff $i \in B_1$ and

Full Implementation in the Complete Network

Definition (Applications-And-References Mechanism π^{so})

For any $m \in M$,

first, identify all agents who send the best application ($B_1 \subseteq N$):

$$B_1 \subseteq N \text{ such that } i \in B_1 \text{ iff } m_{ii} = \min_{k \in N} m_{kk}$$

Second, choose $B_2 \subseteq B_1$ such that

- if $\min_{k \in N} m_{kk} > 0$, then $B_2 = B_1$.
- if $\min_{k \in N} m_{kk} = 0$, then P looks at the references which agents in B_1 receive:
  - worst reference which i receives: $\bar{r}_i = \max_{j \in N_i} m_{ji}$
  - $i \in B_2$ iff $i \in B_1$ and $\bar{r}_i = \min_{k \in B_1} \bar{r}_k$,

Full Implementation in the Complete Network

Definition (Applications-And-References Mechanism π^{so})

For any $m \in M$,

first, identify all agents who send the best application ($B_1 \subseteq N$):

$$B_1 \subseteq N \text{ such that } i \in B_1 \text{ iff } m_{ii} = \min_{k \in N} m_{kk}$$

Second, choose $B_2 \subseteq B_1$ such that

- if $\min_{k \in N} m_{kk} > 0$, then $B_2 = B_1$.
- if $\min_{k \in N} m_{kk} = 0$, then P looks at the references which agents in B_1 receive:
  - worst reference which i receives: $\bar{r}_i = \max_{j \in N_i} m_{ji}$
  - $i \in B_2$ iff $i \in B_1$ and $\bar{r}_i = \min_{k \in B_1} \bar{r}_k$,

    i.e. her worst reference is the least bad among the worst references of agents in $B_1(m)$

Full Implementation in the Complete Network

Definition (Applications-And-References Mechanism π^{so})

For any $m \in M$,

first, identify all agents who send the best application ($B_1 \subseteq N$):

$$B_1 \subseteq N \text{ such that } i \in B_1 \text{ iff } m_{ii} = \min_{k \in N} m_{kk}$$

Second, choose $B_2 \subseteq B_1$ such that

- if $\min_{k \in N} m_{kk} > 0$, then $B_2 = B_1$.
- if $\min_{k \in N} m_{kk} = 0$, then P looks at the references which agents in B_1 receive:
  - worst reference which i receives: $\bar{r}_i = \max_{j \in N_i} m_{ji}$
  - $i \in B_2$ iff $i \in B_1$ and $\bar{r}_i = \min_{k \in B_1} \bar{r}_k$,

    i.e. her worst reference is the least bad among the worst references of agents in $B_1(m)$ ("min-max", "best worst" reference).

Full Implementation in the Complete Network

Definition (Applications-And-References Mechanism π^{so})

For any $m \in M$,

first, identify all agents who send the best application ($B_1 \subseteq N$):

$$B_1 \subseteq N \text{ such that } i \in B_1 \text{ iff } m_{ii} = \min_{k \in N} m_{kk}$$

Second, choose $B_2 \subseteq B_1$ such that

- if $\min_{k \in N} m_{kk} > 0$, then $B_2 = B_1$.
- if $\min_{k \in N} m_{kk} = 0$, then P looks at the references which agents in B_1 receive:
  - worst reference which i receives: $\bar{r}_i = \max_{j \in N_i} m_{ji}$
  - $i \in B_2$ iff $i \in B_1$ and $\bar{r}_i = \min_{k \in B_1} \bar{r}_k$,

    i.e. her worst reference is the least bad among the worst references of agents in $B_1(m)$ ("min-max", "best worst" reference).

Finally, let $\pi_i^{so} = \frac{1}{|B_2|}$ for all $i \in B_2$.

Incentives in $\Gamma(\pi^{so})$ in any L

Given π^{so}, every agent would like to send the best application, and to receive the min-max reference, if the best application is zero.

Incentives in $\Gamma(\pi^{so})$ in any L

Given π^{so}, every agent would like to send the best application, and to receive the min-max reference, if the best application is zero.

$\Rightarrow$ Incentives to fully exaggerate about oneself positively,

Incentives in $\Gamma(\pi^{so})$ in any L

Given π^{so}, every agent would like to send the best application, and to receive the min-max reference, if the best application is zero.

$\Rightarrow$ Incentives to fully exaggerate about oneself positively, and negatively about all neighbors, if own best application is zero.

Incentives in $\Gamma(\pi^{so})$ in any L

Given π^{so}, every agent would like to send the best application, and to receive the min-max reference, if the best application is zero.

$\Rightarrow$ Incentives to fully exaggerate about oneself positively, and negatively about all neighbors, if own best application is zero.

Any m_i with $m_{ii} = \max\{0, d_i - b\}$ and $m_{ij} = \min\{d_j + b, 1\}$ for all $j \in N_i$ is a dominant message.

Incentives in $\Gamma(\pi^{so})$ in any L

Given π^{so}, every agent would like to send the best application, and to receive the min-max reference, if the best application is zero.

$\Rightarrow$ Incentives to fully exaggerate about oneself positively, and negatively about all neighbors, if own best application is zero.

Any m_i with $m_{ii} = \max\{0, d_i - b\}$ and $m_{ij} = \min\{d_j + b, 1\}$ for all $j \in N_i$ is a dominant message.

Definition (Dominance)

A message $m_i \in M_i(\theta_i)$ is dominant at θ_i, if m_i is weakly better than any $m_i' \in M_i(\theta_i)$ for all m_{-i} which agent i believes the others can send.

“Good” dominant strategy equilibrium of $\Gamma(\pi^{so})$ for all L

“Good” dominant strategy equilibrium of $\Gamma(\pi^{so})$ for all L

Lemma

For all L, $\hat{m}_{ii} = \max\{0, d_i - b\}$ and $\hat{m}_{ij} = \min\{d_j + b, 1\}$ for all $j \in N_i$, all θ_i and all i is a dominant strategy equilibrium of $\Gamma(\pi^{so})$

"Good" dominant strategy equilibrium of $\Gamma(\pi^{so})$ for all L

Lemma

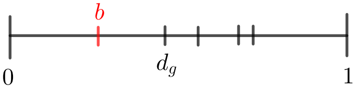
For all L, $\hat{m}_{ii} = \max\{0, d_i - b\}$ and $\hat{m}_{ij} = \min\{d_j + b, 1\}$ for all $j \in N_i$, all θ_i and all i is a dominant strategy equilibrium of $\Gamma(\pi^{so})$ such that $\pi_g^{so} = 1$ for all θ.

"Good" dominant strategy equilibrium of $\Gamma(\pi^{so})$ for all L

Lemma

For all L, $\hat{m}_{ii} = \max\{0, d_i - b\}$ and $\hat{m}_{ij} = \min\{d_j + b, 1\}$ for all $j \in N_i$, all θ_i and all i is a dominant strategy equilibrium of $\Gamma(\pi^{so})$ such that $\pi_g^{so} = 1$ for all θ.

Given $\hat{m}$, if

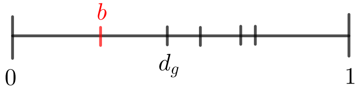


"Good" dominant strategy equilibrium of $\Gamma(\pi^{so})$ for all L

Lemma

For all L, $\hat{m}_{ii} = \max\{0, d_i - b\}$ and $\hat{m}_{ij} = \min\{d_j + b, 1\}$ for all $j \in N_i$, all θ_i and all i is a dominant strategy equilibrium of $\Gamma(\pi^{so})$ such that $\pi_g^{so} = 1$ for all θ.

Given $\hat{m}$, if



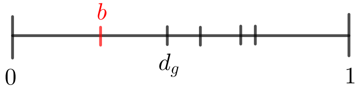
then $m_{gg} < m_{jj}$ for all $j \neq g$.

"Good" dominant strategy equilibrium of $\Gamma(\pi^{so})$ for all L

Lemma

For all L, $\hat{m}_{ii} = \max\{0, d_i - b\}$ and $\hat{m}_{ij} = \min\{d_j + b, 1\}$ for all $j \in N_i$, all θ_i and all i is a dominant strategy equilibrium of $\Gamma(\pi^{so})$ such that $\pi_g^{so} = 1$ for all θ.

Given $\hat{m}$, if



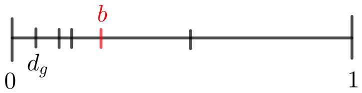
then $m_{gg} < m_{jj}$ for all $j \neq g$. Thus, $\pi_g^{so} = 1$.

"Good" dominant strategy equilibrium of $\Gamma(\pi^{so})$ for all L

Lemma

For all L, $\hat{m}_{ii} = \max\{0, d_i - b\}$ and $\hat{m}_{ij} = \min\{d_j + b, 1\}$ for all $j \in N_i$, all θ_i and all i is a dominant strategy equilibrium of $\Gamma(\pi^{so})$ such that $\pi_g^{so} = 1$ for all θ.

Given $\hat{m}$, if

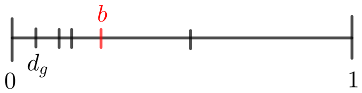


"Good" dominant strategy equilibrium of $\Gamma(\pi^{so})$ for all L

Lemma

For all L, $\hat{m}_{ii} = \max\{0, d_i - b\}$ and $\hat{m}_{ij} = \min\{d_j + b, 1\}$ for all $j \in N_i$, all θ_i and all i is a dominant strategy equilibrium of $\Gamma(\pi^{so})$ such that $\pi_g^{so} = 1$ for all θ.

Given $\hat{m}$, if



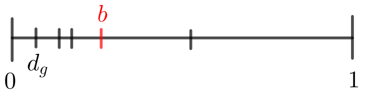
then $m_{gg} = 0$ and $\bar{r}_g < \bar{r}_j$ for all $j \neq g$.

"Good" dominant strategy equilibrium of $\Gamma(\pi^{so})$ for all L

Lemma

For all L, $\hat{m}_{ii} = \max\{0, d_i - b\}$ and $\hat{m}_{ij} = \min\{d_j + b, 1\}$ for all $j \in N_i$, all θ_i and all i is a dominant strategy equilibrium of $\Gamma(\pi^{so})$ such that $\pi^{so}_g = 1$ for all θ.

Given $\hat{m}$, if



then $m_{gg} = 0$ and $\bar{r}_g < \bar{r}_j$ for all $j \neq g$. Thus, $\pi^{so}_g = 1$.

Mechanism π^{so} and the Complete Network

Proposition

Mechanism π^{so} fully implements if and only if L is complete.

Mechanism π^{so} and the Complete Network

Proposition

Mechanism π^{so} fully implements if and only if L is complete.

Consider L complete.

Mechanism π^{so} and the Complete Network

Proposition

Mechanism π^{so} fully implements if and only if L is complete.

Consider L complete.

If $\pi_g^{so} < 1$,

Mechanism π^{so} and the Complete Network

Proposition

Mechanism π^{so} fully implements if and only if L is complete.

Consider L complete.

If $\pi_g^{so} < 1$, then g can deviate to $m'_{gg} = \max\{0, d_g - b\}$

Mechanism π^{so} and the Complete Network

Proposition

Mechanism π^{so} fully implements if and only if L is complete.

Consider L complete.

If $\pi_g^{so} < 1$, then g can deviate to $m'_{gg} = \max\{0, d_g - b\}$ and $m'_{gj} = \min\{d_j + b, 1\}$ for all $j \neq g$ such that $\pi_g^{so'} = 1$.

Mechanism π^{so} and the Complete Network

Proposition

Mechanism π^{so} fully implements if and only if L is complete.

Consider L complete.

If $\pi_g^{so} < 1$, then g can deviate to $m'_{gg} = \max\{0, d_g - b\}$ and $m'_{gj} = \min\{d_j + b, 1\}$ for all $j \neq g$ such that $\pi_g^{so'} = 1$.

Full implementation!

Mechanism π^{so} and Incomplete Networks

If L is incomplete, there exist equilibria such that $\pi_g^{so} < 1$ for a positive measure set of type profiles.

Mechanism π^{so} and Incomplete Networks

If L is incomplete, there exist equilibria such that $\pi_g^{so} < 1$ for a positive measure set of type profiles.

Intuition:
References by agents who are indifferent between all their messages because they know they will lose might determine the prize winner.

Mechanism π^{so} and Incomplete Networks

If L is incomplete, there exist equilibria such that $\pi_g^{so} < 1$ for a positive measure set of type profiles.

Intuition:
References by agents who are indifferent between all their messages because they know they will lose might determine the prize winner.

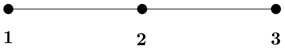
Consider

Mechanism π^{so} and Incomplete Networks

If L is incomplete, there exist equilibria such that $\pi^{so}_g < 1$ for a positive measure set of type profiles.

Intuition:
References by agents who are indifferent between all their messages because they know they will lose might determine the prize winner.

Consider

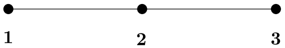


Mechanism π^{so} and Incomplete Networks

If L is incomplete, there exist equilibria such that $\pi^{so}_g < 1$ for a positive measure set of type profiles.

Intuition:
References by agents who are indifferent between all their messages because they know they will lose might determine the prize winner.

Consider



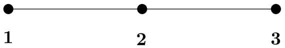
Suppose $i = 1, 3$ send dominant message $m_{ii} = 0$ and $m_{i2} = 1$.

Mechanism π^{so} and Incomplete Networks

If L is incomplete, there exist equilibria such that $\pi_g^{so} < 1$ for a positive measure set of type profiles.

Intuition:
References by agents who are indifferent between all their messages because they know they will lose might determine the prize winner.

Consider



Suppose $i = 1, 3$ send dominant message $m_{ii} = 0$ and $m_{i2} = 1$.

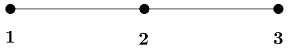
Any m_2 is a best response for 2 because 2 knows that he loses as he cannot send a best application.

Mechanism π^{so} and Incomplete Networks

If L is incomplete, there exist equilibria such that $\pi_g^{so} < 1$ for a positive measure set of type profiles.

Intuition:
References by agents who are indifferent between all their messages because they know they will lose might determine the prize winner.

Consider



Suppose $i = 1, 3$ send dominant message $m_{ii} = 0$ and $m_{i2} = 1$.

Any m_2 is a best response for 2 because 2 knows that he loses as he cannot send a best application.

If 2 sends $m_{23} = d_1$ and $m_{21} = d_3$, then 3 falsely gets the prize.

Full Implementation with Partially Honest Agents

- Full implementation with π^{so} in L incomplete fails because agents who know they are losing are indifferent between references.

Full Implementation with Partially Honest Agents

- Full implementation with π^{so} in L incomplete fails because agents who know they are losing are indifferent between references.
- Tie-breaking rule for this case such that references are "meaningful".

Full Implementation with Partially Honest Agents

- Full implementation with π^{so} in L incomplete fails because agents who know they are losing are indifferent between references.
- Tie-breaking rule for this case such that references are "meaningful".
- Assume an agent only lies, if lying increases her exp. prob. of winning, otherwise she says the truth. (Dutta and Sen, 2012)

Full Implementation with Partially Honest Agents

- Full implementation with π^{so} in L incomplete fails because agents who know they are losing are indifferent between references.
- Tie-breaking rule for this case such that references are “meaningful”.
- Assume an agent only lies, if lying increases her exp. prob. of winning, otherwise she says the truth. (Dutta and Sen, 2012)
- Lexicographic preferences: Agents first care about maximizing exp. prob. of winning, and second about truth-telling.

Full Implementation with Partially Honest Agents

- Full implementation with π^{so} in L incomplete fails because agents who know they are losing are indifferent between references.
- Tie-breaking rule for this case such that references are "meaningful".
- Assume an agent only lies, if lying increases her exp. prob. of winning, otherwise she says the truth. (Dutta and Sen, 2012)
- Lexicographic preferences: Agents first care about maximizing exp. prob. of winning, and second about truth-telling.
- The game induced by π with partially honest agents and everything else as in base game is denoted by $\Gamma^h(\pi)$.

Full Implementation with Partially Honest Agents

- Full implementation with π^{so} in L incomplete fails because agents who know they are losing are indifferent between references.
- Tie-breaking rule for this case such that references are "meaningful".
- Assume an agent only lies, if lying increases her exp. prob. of winning, otherwise she says the truth. (Dutta and Sen, 2012)
- Lexicographic preferences: Agents first care about maximizing exp. prob. of winning, and second about truth-telling.
- The game induced by π with partially honest agents and everything else as in base game is denoted by $\Gamma^h(\pi)$.

Strategy profile $\hat{m}$ is an equilibrium of $\Gamma^h(\pi)$ if and only if for all i, all θ_i and all $m_i \in M_i(\theta_i)$

Full Implementation with Partially Honest Agents

- Full implementation with π^{so} in L incomplete fails because agents who know they are losing are indifferent between references.
- Tie-breaking rule for this case such that references are "meaningful".
- Assume an agent only lies, if lying increases her exp. prob. of winning, otherwise she says the truth. (Dutta and Sen, 2012)
- Lexicographic preferences: Agents first care about maximizing exp. prob. of winning, and second about truth-telling.
- The game induced by π with partially honest agents and everything else as in base game is denoted by $\Gamma^h(\pi)$.

Strategy profile $\hat{m}$ is an equilibrium of $\Gamma^h(\pi)$ if and only if for all i, all θ_i and all $m_i \in M_i(\theta_i)$

1. $\Pi(\hat{m}_i(\theta_i), \hat{m}_{-i}|\theta_i) \geq \Pi(m_i, \hat{m}_{-i}|\theta_i)$, and

Full Implementation with Partially Honest Agents

- Full implementation with π^{so} in L incomplete fails because agents who know they are losing are indifferent between references.
- Tie-breaking rule for this case such that references are "meaningful".
- Assume an agent only lies, if lying increases her exp. prob. of winning, otherwise she says the truth. (Dutta and Sen, 2012)
- Lexicographic preferences: Agents first care about maximizing exp. prob. of winning, and second about truth-telling.
- The game induced by π with partially honest agents and everything else as in base game is denoted by $\Gamma^h(\pi)$.

Strategy profile $\hat{m}$ is an equilibrium of $\Gamma^h(\pi)$ if and only if for all i, all θ_i and all $m_i \in M_i(\theta_i)$

1. $\Pi(\hat{m}_i(\theta_i), \hat{m}_{-i}|\theta_i) \geq \Pi(m_i, \hat{m}_{-i}|\theta_i)$, and
2. $\hat{m}_i(\theta_i) = \theta_i$, if $\Pi(\theta_i, \hat{m}_{-i}|\theta_i) \geq \Pi(m_i, \hat{m}_{-i}|\theta_i)$.

A tweak to π^{so} is needed for equilibrium existence.

If P uses π^{so} and agents are partially honest,
equilibrium existence of $\Gamma^h(\pi^{so})$ is not guaranteed for some L because of lexicographic preferences. (see paper)

$\Rightarrow$ need to adjust π^{so} to recover equilibrium existence for all L!

Definition (Applications-References-Honesty Mechanism π^{soh})

Definition (Applications-References-Honesty Mechanism π^{soh})

For any $m \in M$,
choose B_1 and B_2 in same way as for π^{so}, i.e. best application and min-max reference if the best application is zero.

Definition (Applications-References-Honesty Mechanism π^{soh})

For any $m \in M$,
choose B_1 and B_2 in same way as for π^{so}, i.e. best application and min-max reference if the best application is zero.

First, choose $B_1 \subseteq N$ such that $i \in B_1$ iff $m_{ii} = \min_{k \in N} m_{kk}$,

Definition (Applications-References-Honesty Mechanism π^{soh})

For any $m \in M$,
choose B_1 and B_2 in same way as for π^{so}, i.e. best application and min-max reference if the best application is zero.

First, choose $B_1 \subseteq N$ such that $i \in B_1$ iff $m_{ii} = \min_{k \in N} m_{kk}$,
second, choose $B_2 \subseteq B_1$ such that

- if $\min_{k \in N} m_{kk} > 0$, then $B_2 = B_1$,
- if $\min_{k \in N} m_{kk} = 0$, then $i \in B_2$ iff $i \in B_1$ and $\bar{r}_i = \min_{k \in B_1} \bar{r}_k$.

Definition (Applications-References-Honesty Mechanism π^{soh})

For any $m \in M$,
choose B_1 and B_2 in same way as for π^{so}, i.e. best application and min-max reference if the best application is zero.

First, choose $B_1 \subseteq N$ such that $i \in B_1$ iff $m_{ii} = \min_{k \in N} m_{kk}$,

second, choose $B_2 \subseteq B_1$ such that

- if $\min_{k \in N} m_{kk} > 0$, then $B_2 = B_1$,
- if $\min_{k \in N} m_{kk} = 0$, then $i \in B_2$ iff $i \in B_1$ and $\bar{r}_i = \min_{k \in B_1} \bar{r}_k$.

Third, choose $B_3 \subseteq B_2$ such that $i \in B_3$ iff 1) or 2) is satisfied:

1. $m_{ij} - m_{ii} > 2b$ for all $j \in N_i$ (prove better than all neighbors)
2. $m_{ii} \neq m_{ji}$ or $m_{ij} \neq m_{jj}$ for some $j \in N_i$ (conflict with a neighbor)

Definition (Applications-References-Honesty Mechanism π^{soh})

For any $m \in M$,
choose B_1 and B_2 in same way as for π^{so}, i.e. best application and min-max reference if the best application is zero.

First, choose $B_1 \subseteq N$ such that $i \in B_1$ iff $m_{ii} = \min_{k \in N} m_{kk}$,

second, choose $B_2 \subseteq B_1$ such that

- if $\min_{k \in N} m_{kk} > 0$, then $B_2 = B_1$,
- if $\min_{k \in N} m_{kk} = 0$, then $i \in B_2$ iff $i \in B_1$ and $\bar{r}_i = \min_{k \in B_1} \bar{r}_k$.

Third, choose $B_3 \subseteq B_2$ such that $i \in B_3$ iff 1) or 2) is satisfied:

1. $m_{ij} - m_{ii} > 2b$ for all $j \in N_i$ (prove better than all neighbors)
2. $m_{ii} \neq m_{ji}$ or $m_{ij} \neq m_{jj}$ for some $j \in N_i$ (conflict with a neighbor)

If $B_3 \neq \emptyset$, then $\pi_i^{soh} = \frac{1}{|B_3|}$ for all $i \in B_3$.

Definition (Applications-References-Honesty Mechanism π^{soh})

For any $m \in M$,
choose B_1 and B_2 in same way as for π^{so}, i.e. best application and min-max reference if the best application is zero.

First, choose $B_1 \subseteq N$ such that $i \in B_1$ iff $m_{ii} = \min_{k \in N} m_{kk}$,

second, choose $B_2 \subseteq B_1$ such that

- if $\min_{k \in N} m_{kk} > 0$, then $B_2 = B_1$,
- if $\min_{k \in N} m_{kk} = 0$, then $i \in B_2$ iff $i \in B_1$ and $\bar{r}_i = \min_{k \in B_1} \bar{r}_k$.

Third, choose $B_3 \subseteq B_2$ such that $i \in B_3$ iff 1) or 2) is satisfied:

1. $m_{ij} - m_{ii} > 2b$ for all $j \in N_i$ (prove better than all neighbors)
2. $m_{ii} \neq m_{ji}$ or $m_{ij} \neq m_{jj}$ for some $j \in N_i$ (conflict with a neighbor)

If $B_3 \neq \emptyset$, then $\pi_i^{soh} = \frac{1}{|B_3|}$ for all $i \in B_3$.

If $B_3 = \emptyset$, then "punishment allocations"... (see paper)

For all equilibria of $\Gamma^h(\pi^{soh})$ which we present $B_3 \neq \emptyset$ with probability 1.

For all equilibria of $\Gamma^h(\pi^{soh})$ which we present $B_3 \neq \emptyset$ with probability 1.

Incentives in $\Gamma^h(\pi^{soh})$:

- Every agent wants to send the best application,
  receive the min-max reference if the best application is zero,
  and prove better than all neighbors or conflict with a neighbor.

For all equilibria of $\Gamma^h(\pi^{soh})$ which we present $B_3 \neq \emptyset$ with probability 1.

Incentives in $\Gamma^h(\pi^{soh})$:

- Every agent wants to send the best application,
  receive the min-max reference if the best application is zero,
  and prove better than all neighbors or conflict with a neighbor.

Claim

For every L, strategy profile $\hat{m}^h$ is an equilibrium of $\Gamma^h(\pi^{soh})$ such that $\pi_g^{soh} = 1$ for all θ.

Equilibrium strategy profile $\hat{m}^h$ such that $\pi_g^{soh} = 1$

Equilibrium strategy profile $\hat{m}^h$ such that $\pi_g^{soh} = 1$

A *non-minimal* agent i has $d_i > d_j$ for some $j \in N_i$.

Equilibrium strategy profile $\hat{m}^h$ such that $\pi_g^{soh} = 1$

A *non-minimal* agent i has $d_i > d_j$ for some $j \in N_i$.

A *local minimum* agent i has $d_i < d_j$ for all $j \in N_i$.

Equilibrium strategy profile $\hat{m}^h$ such that $\pi_g^{soh} = 1$

A *non-minimal* agent i has $d_i > d_j$ for some $j \in N_i$.
A *local minimum* agent i has $d_i < d_j$ for all $j \in N_i$.

Strategy profile $\hat{m}^h$:

Equilibrium strategy profile $\hat{m}^h$ such that $\pi_g^{soh} = 1$

A *non-minimal* agent i has $d_i > d_j$ for some $j \in N_i$.
A *local minimum* agent i has $d_i < d_j$ for all $j \in N_i$.

Strategy profile $\hat{m}^h$:

- any *non-minimal* agent i says the truth

Equilibrium strategy profile $\hat{m}^h$ such that $\pi_g^{soh} = 1$

A *non-minimal* agent i has $d_i > d_j$ for some $j \in N_i$.
A *local minimum* agent i has $d_i < d_j$ for all $j \in N_i$.

Strategy profile $\hat{m}^h$:

- any *non-minimal* agent i says the truth
- any local minimum i with *partial information* (not linked to all $j \neq i$)

Equilibrium strategy profile $\hat{m}^h$ such that $\pi_g^{soh} = 1$

A *non-minimal* agent i has $d_i > d_j$ for some $j \in N_i$.
A *local minimum* agent i has $d_i < d_j$ for all $j \in N_i$.

Strategy profile $\hat{m}^h$:

- any *non-minimal* agent i says the truth
- any local minimum i with *partial information* (not linked to all $j \neq i$)
  - best possible application,

Equilibrium strategy profile $\hat{m}^h$ such that $\pi_g^{soh} = 1$

A *non-minimal* agent i has $d_i > d_j$ for some $j \in N_i$.
A *local minimum* agent i has $d_i < d_j$ for all $j \in N_i$.

Strategy profile $\hat{m}^h$:

- any *non-minimal* agent i says the truth
- any local minimum i with *partial information* (not linked to all $j \neq i$)
    - best possible application,
    - worst possible references,

Equilibrium strategy profile $\hat{m}^h$ such that $\pi_g^{soh} = 1$

A *non-minimal* agent i has $d_i > d_j$ for some $j \in N_i$.
A *local minimum* agent i has $d_i < d_j$ for all $j \in N_i$.

Strategy profile $\hat{m}^h$:

- any *non-minimal* agent i says the truth
- any local minimum i with *partial information* (not linked to all $j \neq i$)
  - best possible application,
  - worst possible references,

    if $d_j - d_i \leq 2b$ for some $j \in N_i$ (some neighbor's distance is closeby)

Equilibrium strategy profile $\hat{m}^h$ such that $\pi_g^{soh} = 1$

A *non-minimal* agent i has $d_i > d_j$ for some $j \in N_i$.
A *local minimum* agent i has $d_i < d_j$ for all $j \in N_i$.

Strategy profile $\hat{m}^h$:

- any *non-minimal* agent i says the truth
- any local minimum i with *partial information* (not linked to all $j \neq i$)
  - best possible application,
  - worst possible references,
    if $d_j - d_i \leq 2b$ for some $j \in N_i$ (some neighbor's distance is closeby)
  - truthful references,

Equilibrium strategy profile $\hat{m}^h$ such that $\pi_g^{soh} = 1$

A *non-minimal* agent i has $d_i > d_j$ for some $j \in N_i$.
A *local minimum* agent i has $d_i < d_j$ for all $j \in N_i$.

Strategy profile $\hat{m}^h$:

- any *non-minimal* agent i says the truth
- any local minimum i with *partial information* (not linked to all $j \neq i$)
  - best possible application,
  - worst possible references,
    if $d_j - d_i \leq 2b$ for some $j \in N_i$ (some neighbor's distance is closeby)
  - truthful references,
    if $d_j - d_i > 2b$ for every $j \in N_i$ (all neighbors' distances are far)

Equilibrium strategy profile $\hat{m}^h$ such that $\pi_g^{soh} = 1$

A *non-minimal* agent i has $d_i > d_j$ for some $j \in N_i$.
A *local minimum* agent i has $d_i < d_j$ for all $j \in N_i$.

Strategy profile $\hat{m}^h$:

- any *non-minimal* agent i says the truth
- any local minimum i with *partial information* (not linked to all $j \neq i$)
  - best possible application,
  - worst possible references,
    if $d_j - d_i \leq 2b$ for some $j \in N_i$ (some neighbor's distance is closeby)
  - truthful references,
    if $d_j - d_i > 2b$ for every $j \in N_i$ (all neighbors' distances are far)
- any local minimum i with *full information* (linked to all $j \neq i$)

Equilibrium strategy profile $\hat{m}^h$ such that $\pi_g^{soh} = 1$

A *non-minimal* agent i has $d_i > d_j$ for some $j \in N_i$.
A *local minimum* agent i has $d_i < d_j$ for all $j \in N_i$.

Strategy profile $\hat{m}^h$:

- any *non-minimal* agent i says the truth
- any local minimum i with *partial information* (not linked to all $j \neq i$)
  - best possible application,
  - worst possible references,
    if $d_j - d_i \leq 2b$ for some $j \in N_i$ (some neighbor's distance is closeby)
  - truthful references,
    if $d_j - d_i > 2b$ for every $j \in N_i$ (all neighbors' distances are far)
- any local minimum i with *full information* (linked to all $j \neq i$)
  - best possible application and worst possible references,
    if $d_j - d_i \leq 2b$ for some $j \in N_i$,

Equilibrium strategy profile $\hat{m}^h$ such that $\pi_g^{soh} = 1$

A *non-minimal* agent i has $d_i > d_j$ for some $j \in N_i$.
A *local minimum* agent i has $d_i < d_j$ for all $j \in N_i$.

Strategy profile $\hat{m}^h$:

- any *non-minimal* agent i says the truth
- any local minimum i with *partial information* (not linked to all $j \neq i$)
  - best possible application,
  - worst possible references,
    if $d_j - d_i \leq 2b$ for some $j \in N_i$ (some neighbor's distance is closeby)
  - truthful references,
    if $d_j - d_i > 2b$ for every $j \in N_i$ (all neighbors' distances are far)
- any local minimum i with *full information* (linked to all $j \neq i$)
  - best possible application and worst possible references,
    if $d_j - d_i \leq 2b$ for some $j \in N_i$,
  - the truth, if $d_j - d_i > 2b$ for every $j \in N_i$

Full Implementation with π^{soh}: Sufficient Conditions on L

Full Implementation with π^{soh}: Sufficient Conditions on L

Theorem

Mechanism π^{soh} fully implements with partially honest agents, if L is connected and for all $i \in N$ and all $j \in N_i$ is true that

Full Implementation with π^{soh}: Sufficient Conditions on L

Theorem

Mechanism π^{soh} fully implements with partially honest agents, if L is connected and for all $i \in N$ and all $j \in N_i$ is true that

- $N_j \setminus i \subset N_i$ *(i is linked to all neighbors $k \neq i$ of j)*
  or

Full Implementation with π^{soh}: Sufficient Conditions on L

Theorem

Mechanism π^{soh} fully implements with partially honest agents, if L is connected and for all $i \in N$ and all $j \in N_i$ is true that

- $N_j \backslash i \subset N_i$ *(i is linked to all neighbors $k \neq i$ of j)*
  or
- *j is linked to every $k \neq j$ (j has full information)*

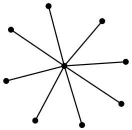
Full Implementation with π^{soh}: Sufficient Conditions on L

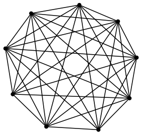
Theorem

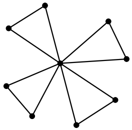
Mechanism π^{soh} fully implements with partially honest agents, if L is connected and for all $i \in N$ and all $j \in N_i$ is true that

- $N_j \backslash i \subset N_i$ *(i is linked to all neighbors $k \neq i$ of j)*
  or
- *j is linked to every $k \neq j$ (j has full information)*

E.g. the complete graph, the star, the "Dutch windmill", ...





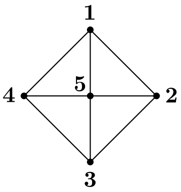


Full Implementation with π^{soh} is Non-Monotonic in Links

Claim

There is L which is a supergraph of the star and a subgraph of the complete graph such that π^{soh} does not fully implement in L.

There is an equilibrium such that
if $0 < d_1 < d_2 < d_3 < d_4 < d_5 \leq b$, then



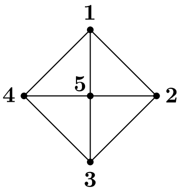
Full Implementation with π^{soh} is Non-Monotonic in Links

Claim

There is L which is a supergraph of the star and a subgraph of the complete graph such that π^{soh} does not fully implement in L.

There is an equilibrium such that
if $0 < d_1 < d_2 < d_3 < d_4 < d_5 \leq b$, then

- 5 says the truth,



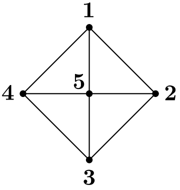
Full Implementation with π^{soh} is Non-Monotonic in Links

Claim

There is L which is a supergraph of the star and a subgraph of the complete graph such that π^{soh} does not fully implement in L.

There is an equilibrium such that
if $0 < d_1 < d_2 < d_3 < d_4 < d_5 \leq b$, then

- 5 says the truth,
- $m_{ii} = 0$ for all $i = 1, 2, 3, 4$



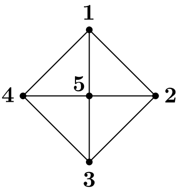
Full Implementation with π^{soh} is Non-Monotonic in Links

Claim

There is L which is a supergraph of the star and a subgraph of the complete graph such that π^{soh} does not fully implement in L.

There is an equilibrium such that
if $0 < d_1 < d_2 < d_3 < d_4 < d_5 \leq b$, then

- 5 says the truth,
- $m_{ii} = 0$ for all $i = 1, 2, 3, 4$
- $m_{1j} = d_j + b$ for all $j \in N_1$



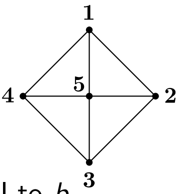
Full Implementation with π^{soh} is Non-Monotonic in Links

Claim

There is L which is a supergraph of the star and a subgraph of the complete graph such that π^{soh} does not fully implement in L.

There is an equilibrium such that
if $0 < d_1 < d_2 < d_3 < d_4 < d_5 \leq b$, then

- 5 says the truth,
- $m_{ii} = 0$ for all $i = 1, 2, 3, 4$
- $m_{1j} = d_j + b$ for all $j \in N_1$
- references by 2,3,4, about all their neighbors are equal to b.

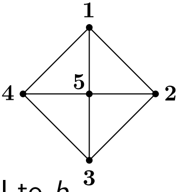


Full Implementation with π^{soh} is Non-Monotonic in Links

Claim

There is L which is a supergraph of the star and a subgraph of the complete graph such that π^{soh} does not fully implement in L.

There is an equilibrium such that
if $0 < d_1 < d_2 < d_3 < d_4 < d_5 \leq b$, then



- 5 says the truth,
- $m_{ii} = 0$ for all $i = 1, 2, 3, 4$
- $m_{1j} = d_j + b$ for all $j \in N_1$
- references by 2,3,4, about all their neighbors are equal to b.
- 1 and 3 send best application and have min-max reference of b $\Rightarrow B_3 = \{1, 3\}$

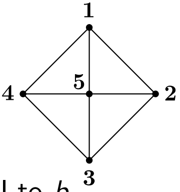
Full Implementation with π^{soh} is Non-Monotonic in Links

Claim

There is L which is a supergraph of the star and a subgraph of the complete graph such that π^{soh} does not fully implement in L.

There is an equilibrium such that
if $0 < d_1 < d_2 < d_3 < d_4 < d_5 \leq b$, then

- 5 says the truth,
- $m_{ii} = 0$ for all $i = 1, 2, 3, 4$
- $m_{1j} = d_j + b$ for all $j \in N_1$
- references by 2,3,4, about all their neighbors are equal to b.
- 1 and 3 send best application and have min-max reference of b $\Rightarrow B_3 = \{1, 3\}$



The property of full implementation not monotonic in links added.
More information and communication possibilities not always beneficial.

Conclusion

- develop mechanisms to allocate a prize to the best agent, when there are self-ratings, peer reviews and a limit to lying
- achieve full implementation for the complete network, and for a larger class of networks, if agents are partially honest
- full implementation via untruthful equilibria
  $\Rightarrow$ focus on truthful revelation is not necessary

Open questions:

- how can the principal ensure a specific limit to lying?
- other tie-breaking rules for indifference, e.g. favoritism? Imperfect knowledge about neighbors and network? ...

Unsuccessful Mechanisms I

1. "Consensus" mechanisms relying on Maskin-/Bayesian-monotonicity
    - standard mechanisms for full implementation

Unsuccessful Mechanisms I

1. "Consensus" mechanisms relying on Maskin-/Bayesian-monotonicity
    - standard mechanisms for full implementation
    - One property: If there are no conflicting statements (consensus), then prize is given to agent who the consensus says is best.

Unsuccessful Mechanisms I

1. "Consensus" mechanisms relying on Maskin-/Bayesian-monotonicity
    - standard mechanisms for full implementation
    - One property: If there are no conflicting statements (consensus), then prize is given to agent who the consensus says is best.
    - relies on truthful revelation in equilibrium; what about "false" consensus equilibria?

Unsuccessful Mechanisms I

1. "Consensus" mechanisms relying on Maskin-/Bayesian-monotonicity
   - standard mechanisms for full implementation
   - One property: If there are no conflicting statements (consensus), then prize is given to agent who the consensus says is best.
   - relies on truthful revelation in equilibrium; what about "false" consensus equilibria? M-/B-monotonicity serves to rule out this out.

Unsuccessful Mechanisms I

1. "Consensus" mechanisms relying on Maskin-/Bayesian-monotonicity
    - standard mechanisms for full implementation
    - One property: If there are no conflicting statements (consensus), then prize is given to agent who the consensus says is best.
    - relies on truthful revelation in equilibrium; what about "false" consensus equilibria? M-/B-monotonicity serves to rule out this out.
    - M-/B-monotonicity is not satisfied in our setting:

Unsuccessful Mechanisms I

1. "Consensus" mechanisms relying on Maskin-/Bayesian-monotonicity
   - standard mechanisms for full implementation
   - One property: If there are no conflicting statements (consensus), then prize is given to agent who the consensus says is best.
   - relies on truthful revelation in equilibrium; what about "false" consensus equilibria? M-/B-monotonicity serves to rule out this out.
   - M-/B-monotonicity is not satisfied in our setting:
     It requires that agents' preference orderings over outcomes change across type realizations/states.

Unsuccessful Mechanisms I

1. "Consensus" mechanisms relying on Maskin-/Bayesian-monotonicity
    - standard mechanisms for full implementation
    - One property: If there are no conflicting statements (consensus), then prize is given to agent who the consensus says is best.
    - relies on truthful revelation in equilibrium; what about "false" consensus equilibria? M-/B-monotonicity serves to rule out this out.
    - M-/B-monotonicity is not satisfied in our setting:
      It requires that agents' preference orderings over outcomes change across type realizations/states. In our setting, every agent always prefers a higher over a lower probability of getting the prize.

Unsuccessful Mechanisms I

1. "Consensus" mechanisms relying on Maskin-/Bayesian-monotonicity
    - standard mechanisms for full implementation
    - One property: If there are no conflicting statements (consensus), then prize is given to agent who the consensus says is best.
    - relies on truthful revelation in equilibrium; what about "false" consensus equilibria? M-/B-monotonicity serves to rule out this out.
    - M-/B-monotonicity is not satisfied in our setting:
      It requires that agents' preference orderings over outcomes change across type realizations/states. In our setting, every agent always prefers a higher over a lower probability of getting the prize.
    - failure of M-/ B- monotonicity implies that full implementation would be impossible in our setting, if there was no limit to lying (Maskin (1999), Jackson (1991))

Unsuccessful Mechanisms I

1. "Consensus" mechanisms relying on Maskin-/Bayesian-monotonicity
   - standard mechanisms for full implementation
   - One property: If there are no conflicting statements (consensus), then prize is given to agent who the consensus says is best.
   - relies on truthful revelation in equilibrium; what about "false" consensus equilibria? M-/B-monotonicity serves to rule out this out.
   - M-/B-monotonicity is not satisfied in our setting:
     It requires that agents' preference orderings over outcomes change across type realizations/states. In our setting, every agent always prefers a higher over a lower probability of getting the prize.
   - failure of M-/ B- monotonicity implies that full implementation would be impossible in our setting, if there was no limit to lying (Maskin (1999), Jackson (1991))
   - I use mechanisms for full implementation where agents lie in equilibrium (no truthful revelation).

Unsuccessful Mechanisms II

2. Applications Only

Unsuccessful Mechanisms II

2. Applications Only
    - "Give prize to agent with 'best' application"

Unsuccessful Mechanisms II

2. Applications Only
   - “Give prize to agent with ’best’ application”
   - Each agent has incentives to positively exaggerate about herself.

Unsuccessful Mechanisms II

2. Applications Only
    - "Give prize to agent with 'best' application"
    - Each agent has incentives to positively exaggerate about herself.
    - if all are very good (close to 0), "bunching" (all applications) at 0 $\lightning$

Unsuccessful Mechanisms II

2. Applications Only
   - "Give prize to agent with 'best' application"
   - Each agent has incentives to positively exaggerate about herself.
   - if all are very good (close to 0), "bunching" (all applications) at 0 $\lightning$
   - One reference for every agent is necessary for full implementation!

Unsuccessful Mechanisms II

2. Applications Only
   - "Give prize to agent with 'best' application"
   - Each agent has incentives to positively exaggerate about herself.
   - if all are very good (close to 0), "bunching" (all applications) at 0 ↯
   - One reference for every agent is necessary for full implementation! Network is important.

Unsuccessful Mechanisms II

2. Applications Only
    - “Give prize to agent with 'best' application”
    - Each agent has incentives to positively exaggerate about herself.
    - if all are very good (close to 0), “bunching” (all applications) at 0 $\lightning$
    - One reference for every agent is necessary for full implementation! Network is important.
3. References Only

Unsuccessful Mechanisms II

2. Applications Only
    - "Give prize to agent with 'best' application"
    - Each agent has incentives to positively exaggerate about herself.
    - if all are very good (close to 0), "bunching" (all applications) at 0 $\lightning$
    - One reference for every agent is necessary for full implementation! Network is important.
3. References Only
    - "Give prize to agent who has the 'best' reference"

Unsuccessful Mechanisms II

2. Applications Only
   - “Give prize to agent with ’best’ application”
   - Each agent has incentives to positively exaggerate about herself.
   - if all are very good (close to 0), “bunching” (all applications) at 0 ↯
   - One reference for every agent is necessary for full implementation! Network is important.
3. References Only
   - “Give prize to agent who has the ’best’ reference”
   - Each agent has incentives to negatively exaggerate about her neighbors.

Unsuccessful Mechanisms II

2. Applications Only
   - "Give prize to agent with 'best' application"
   - Each agent has incentives to positively exaggerate about herself.
   - if all are very good (close to 0), "bunching" (all applications) at 0 ↯
   - One reference for every agent is necessary for full implementation! Network is important.
3. References Only
   - "Give prize to agent who has the 'best' reference"
   - Each agent has incentives to negatively exaggerate about her neighbors.
   - if all are very bad (close to 1), "bunching" (all references) at 1 ↯

Unsuccessful Mechanisms II

2. Applications Only
   - “Give prize to agent with 'best' application”
   - Each agent has incentives to positively exaggerate about herself.
   - if all are very good (close to 0), “bunching” (all applications) at 0 ↯
   - One reference for every agent is necessary for full implementation! Network is important.

3. References Only
   - “Give prize to agent who has the 'best' reference”
   - Each agent has incentives to negatively exaggerate about her neighbors.
   - if all are very bad (close to 1), “bunching” (all references) at 1 ↯

“Applications only” and “references only” fail for different states $\Rightarrow$ a mechanism which relies on both is successful!

Full Implementation with Noisy Communication

- instead of partial honesty, suppose communication is noisy

Full Implementation with Noisy Communication

- instead of partial honesty, suppose communication is noisy
- denote game with noisy communication, everything else as in the base game, as Γ^n

Full Implementation with Noisy Communication

- instead of partial honesty, suppose communication is noisy
- denote game with noisy communication, everything else as in the base game, as Γ^n
- noisy communication:
  i with θ_i still chooses $m_i \in M_i(\theta_i)$ as before,
  but the principal receives $\tilde{m}_{ik} = m_{ik} + \epsilon_{ik}$ for $k = i, j \in N_i$
  where ϵ_{ik} is iid from a distribution with full support on $[-m_{ik}, 1 - m_{ik}]$
  with mean 0 and strictly decreasing likelihood for larger errors

Full Implementation with Noisy Communication

- instead of partial honesty, suppose communication is noisy
- denote game with noisy communication, everything else as in the base game, as Γ^n
- noisy communication:
  i with θ_i still chooses $m_i \in M_i(\theta_i)$ as before,
  but the principal receives $\tilde{m}_{ik} = m_{ik} + \epsilon_{ik}$ for $k = i, j \in N_i$
  where ϵ_{ik} is iid from a distribution with full support on $[-m_{ik}, 1 - m_{ik}]$
  with mean 0 and strictly decreasing likelihood for larger errors
- thus for any m_{ik}, $\tilde{m}_{ik}$ can take any value in $[0, 1]$ with higher likelihood for values closer to m_{ik}

Full Implementation with Noisy Communication

- instead of partial honesty, suppose communication is noisy
- denote game with noisy communication, everything else as in the base game, as Γ^n
- noisy communication:
  i with θ_i still chooses $m_i \in M_i(\theta_i)$ as before,
  but the principal receives $\tilde{m}_{ik} = m_{ik} + \epsilon_{ik}$ for $k = i, j \in N_i$
  where ϵ_{ik} is iid from a distribution with full support on $[-m_{ik}, 1 - m_{ik}]$
  with mean 0 and strictly decreasing likelihood for larger errors
- thus for any m_{ik}, $\tilde{m}_{ik}$ can take any value in $[0, 1]$ with higher likelihood for values closer to m_{ik}
- If P applies the prize allocation function to $\tilde{m}$,
  then for every θ_i there is a pos. prob. that $\tilde{m}_i$ is a winning message

Full Implementation in Expectation for all L

- For any $\hat{m}_{-i}$ and any θ_i,
  m_i such that $m_{ii} = \max\{0, d_i - b\}$ and $m_{ij} = \min\{d_j + b, 1\}$ for all $j \in N_i$ uniquely maximizes i's exp. prob. of sending the best application, and receiving the min-max reference if the best application is zero.

Full Implementation in Expectation for all L

- For any $\hat{m}_{-i}$ and any θ_i, m_i such that $m_{ii} = \max\{0, d_i - b\}$ and $m_{ij} = \min\{d_j + b, 1\}$ for all $j \in N_i$ uniquely maximizes i's exp. prob. of sending the best application, and receiving the min-max reference if the best application is zero.

Proposition

For any L, the unique equilibrium of $\Gamma^n(\pi^{so})$ is $\hat{m}_{ii} = \max\{0, d_i - b\}$ and $\hat{m}_{ij} = \min\{d_j + b, 1\}$ for all $j \in N_i$ and i. Thus π^{so} fully implements in expectation.

Full Implementation in Expectation for all L

- For any $\hat{m}_{-i}$ and any θ_i,
  m_i such that $m_{ii} = \max\{0, d_i - b\}$ and $m_{ij} = \min\{d_j + b, 1\}$ for all $j \in N_i$ uniquely maximizes i's exp. prob. of sending the best application, and receiving the min-max reference if the best application is zero.

Proposition

For any L, the unique equilibrium of $\Gamma^n(\pi^{so})$ is $\hat{m}_{ii} = \max\{0, d_i - b\}$ and $\hat{m}_{ij} = \min\{d_j + b, 1\}$ for all $j \in N_i$ and i. Thus π^{so} fully implements in expectation.

π^{so} fully implements in expectation, because $E[\tilde{m}] = \hat{m}(\theta)$ and $\pi_g^{so}(\hat{m}(\theta)) = 1$ for all θ.

Applications Only: Formal Proof

Suppose $\hat{m}$ is an eq. and if $d_g > b$, then $\pi_g^s < 1$ with pos. prob.

Applications Only: Formal Proof

Suppose $\hat{m}$ is an eq. and if $d_g > b$, then $\pi_g^s < 1$ with pos. prob. Then g expects with pos. prob. to be g and that $\pi_g^s < 1$.

Applications Only: Formal Proof

Suppose $\hat{m}$ is an eq. and if $d_g > b$, then $\pi_g^s < 1$ with pos. prob. Then g expects with pos. prob. to be g and that $\pi_g^s < 1$. If g deviates to $m'_{gg} = d_g - b$, then g sends the best appl. and $\pi_g^{s'} = 1$.

Applications Only: Formal Proof

Suppose $\hat{m}$ is an eq. and if $d_g > b$, then $\pi_g^s < 1$ with pos. prob. Then g expects with pos. prob. to be g and that $\pi_g^s < 1$. If g deviates to $m'_{gg} = d_g - b$, then g sends the best appl. and $\pi_g^{s'} = 1$. Agent g does not lose in any other case which g expects with pos. prob.

Applications Only: Formal Proof

Suppose $\hat{m}$ is an eq. and if $d_g > b$, then $\pi_g^s < 1$ with pos. prob. Then g expects with pos. prob. to be g and that $\pi_g^s < 1$. If g deviates to $m'_{gg} = d_g - b$, then g sends the best appl. and $\pi_g^{s'} = 1$. Agent g does not lose in any other case which g expects with pos. prob. $\lightning$

Applications Only: Formal Proof

Suppose $\hat{m}$ is an eq. and if $d_g > b$, then $\pi_g^s < 1$ with pos. prob. Then g expects with pos. prob. to be g and that $\pi_g^s < 1$. If g deviates to $m'_{gg} = d_g - b$, then g sends the best appl. and $\pi_g^{s'} = 1$. Agent g does not lose in any other case which g expects with pos. prob. $\lightning$

Suppose $\hat{m}$ is an eq. and if $d_g \leq b$, then $\pi_i^s < \frac{1}{|D|}$ for some $i \in D$ with pos. prob.

Applications Only: Formal Proof

Suppose $\hat{m}$ is an eq. and if $d_g > b$, then $\pi_g^s < 1$ with pos. prob. Then g expects with pos. prob. to be g and that $\pi_g^s < 1$. If g deviates to $m'_{gg} = d_g - b$, then g sends the best appl. and $\pi_g^{s'} = 1$. Agent g does not lose in any other case which g expects with pos. prob. ↯

Suppose $\hat{m}$ is an eq. and if $d_g \leq b$, then $\pi_i^s < \frac{1}{|D|}$ for some $i \in D$ with pos. prob. Then $i \in D$ expects with pos. prob. that $\pi_i^s < \frac{1}{|D|}$.

Applications Only: Formal Proof

Suppose $\hat{m}$ is an eq. and if $d_g > b$, then $\pi_g^s < 1$ with pos. prob. Then g expects with pos. prob. to be g and that $\pi_g^s < 1$. If g deviates to $m'_{gg} = d_g - b$, then g sends the best appl. and $\pi_g^{s'} = 1$. Agent g does not lose in any other case which g expects with pos. prob. $\lightning$

Suppose $\hat{m}$ is an eq. and if $d_g \leq b$, then $\pi_i^s < \frac{1}{|D|}$ for some $i \in D$ with pos. prob. Then $i \in D$ expects with pos. prob. that $\pi_i^s < \frac{1}{|D|}$. If i deviates to $m'_{ii} = 0$, then i sends a best appl. and $\pi_i^{s'} \geq \frac{1}{|D|}$.

Applications Only: Formal Proof

Suppose $\hat{m}$ is an eq. and if $d_g > b$, then $\pi_g^s < 1$ with pos. prob. Then g expects with pos. prob. to be g and that $\pi_g^s < 1$. If g deviates to $m'_{gg} = d_g - b$, then g sends the best appl. and $\pi_g^{s'} = 1$. Agent g does not lose in any other case which g expects with pos. prob. $\lightning$

Suppose $\hat{m}$ is an eq. and if $d_g \leq b$, then $\pi_i^s < \frac{1}{|D|}$ for some $i \in D$ with pos. prob. Then $i \in D$ expects with pos. prob. that $\pi_i^s < \frac{1}{|D|}$. If i deviates to $m'_{ii} = 0$, then i sends a best appl. and $\pi_i^{s'} \geq \frac{1}{|D|}$. Agent i does not lose in any other case which i expects with pos. prob.

Applications Only: Formal Proof

Suppose $\hat{m}$ is an eq. and if $d_g > b$, then $\pi_g^s < 1$ with pos. prob. Then g expects with pos. prob. to be g and that $\pi_g^s < 1$. If g deviates to $m'_{gg} = d_g - b$, then g sends the best appl. and $\pi_g^{s'} = 1$. Agent g does not lose in any other case which g expects with pos. prob. $\lightning$

Suppose $\hat{m}$ is an eq. and if $d_g \leq b$, then $\pi_i^s < \frac{1}{|D|}$ for some $i \in D$ with pos. prob. Then $i \in D$ expects with pos. prob. that $\pi_i^s < \frac{1}{|D|}$. If i deviates to $m'_{ii} = 0$, then i sends a best appl. and $\pi_i^{s'} \geq \frac{1}{|D|}$. Agent i does not lose in any other case which i expects with pos. prob. $\lightning$

Applications Only: Formal Proof

Suppose $\hat{m}$ is an eq. and if $d_g > b$, then $\pi_g^s < 1$ with pos. prob. Then g expects with pos. prob. to be g and that $\pi_g^s < 1$. If g deviates to $m'_{gg} = d_g - b$, then g sends the best appl. and $\pi_g^{s'} = 1$. Agent g does not lose in any other case which g expects with pos. prob. ↯

Suppose $\hat{m}$ is an eq. and if $d_g \leq b$, then $\pi_i^s < \frac{1}{|D|}$ for some $i \in D$ with pos. prob. Then $i \in D$ expects with pos. prob. that $\pi_i^s < \frac{1}{|D|}$. If i deviates to $m'_{ii} = 0$, then i sends a best appl. and $\pi_i^{s'} \geq \frac{1}{|D|}$. Agent i does not lose in any other case which i expects with pos. prob. ↯

Suppose $\hat{m}$ is an eq. and if $d_g \leq b$, then $\pi_i^s > \frac{1}{|D|}$ for some $i \in D$ with pos. prob.

Applications Only: Formal Proof

Suppose $\hat{m}$ is an eq. and if $d_g > b$, then $\pi_g^s < 1$ with pos. prob. Then g expects with pos. prob. to be g and that $\pi_g^s < 1$. If g deviates to $m'_{gg} = d_g - b$, then g sends the best appl. and $\pi_g^{s'} = 1$. Agent g does not lose in any other case which g expects with pos. prob. ↯

Suppose $\hat{m}$ is an eq. and if $d_g \leq b$, then $\pi_i^s < \frac{1}{|D|}$ for some $i \in D$ with pos. prob. Then $i \in D$ expects with pos. prob. that $\pi_i^s < \frac{1}{|D|}$. If i deviates to $m'_{ii} = 0$, then i sends a best appl. and $\pi_i^{s'} \geq \frac{1}{|D|}$. Agent i does not lose in any other case which i expects with pos. prob. ↯

Suppose $\hat{m}$ is an eq. and if $d_g \leq b$, then $\pi_i^s > \frac{1}{|D|}$ for some $i \in D$ with pos. prob. But then some $j \in D$ expects $\pi_i^s < \frac{1}{|D|}$ with pos. prob. ↯

Back

References Only: Formal Proof

Consider L complete.
If $d_g < 1 - b$ and $\pi_g^o < 1$,
g deviates to $m'_{gj} = \min\{d_j + b, 1\} = \bar{r}'_j > d_g + b \geq \bar{r}_g$ for all $j \neq g$ and $\pi_g^{o'} = 1$.
If $d_g \geq 1 - b$ and $\pi_i^o < \frac{1}{n}$,
i deviates to $m'_{ij} = 1 \geq \bar{r}_i$ for all $j \neq i$ and $\pi_i^{o'} \geq \frac{1}{n}$.

Back

Consensus relying on Maskin-/Bayesian-Monotonicity: Proof

- Rule 1:
  If there are no conflicting statements across all agents, allocate the prize according to the consensus.
- Rule 2:
  If $n-1$ agents claim state θ with outcome $\pi(\theta)$ and agent i claims state θ' with $\pi(\theta') \neq \pi(\theta)$,
  then P chooses $\pi(\theta)$, if i strictly prefers $\pi(\theta')$ in state θ, and $\pi(\theta')$, if i weakly prefers $\pi(\theta)$ in state θ.

Take our model with L complete. Consider a false consensus claiming θ with $\pi(\theta)$ when the true state is θ'.
For this not to be an equilibrium, there must be i who deviates to claim θ' with $\pi(\theta')$ and weakly prefers $\pi(\theta)$ in state θ. If this is the case, then i also weakly prefers $\pi(\theta)$ in state θ' and i does not deviate.

Back

Complete Graph: Formal Poof

Proposition

If L is complete, mechanism π^{so} *fully implements the principal's objective.*

Complete Graph: Formal Poof

Proposition

If L is complete, mechanism π^{so} fully implements the principal's objective.

Consider L complete.

Complete Graph: Formal Poof

Proposition

If L is complete, mechanism π^{so} fully implements the principal's objective.

Consider L complete.

Suppose $\hat{m}$ is eq. and $\pi_g^{so} < 1$ for some θ.

Complete Graph: Formal Poof

Proposition

If L is complete, mechanism π^{so} fully implements the principal's objective.

Consider L complete.

Suppose $\hat{m}$ is eq. and $\pi_g^{so} < 1$ for some θ.

Then g can deviate to $m'_{gg} = \max\{0, d_g - b\}$ and $m'_{gj} = \min\{d_j + b, 1\}$ for all $j \neq g$.

Complete Graph: Formal Poof

Proposition

If L is complete, mechanism π^{so} fully implements the principal's objective.

Consider L complete.

Suppose $\hat{m}$ is eq. and $\pi_g^{so} < 1$ for some θ.

Then g can deviate to $m'_{gg} = \max\{0, d_g - b\}$ and $m'_{gj} = \min\{d_j + b, 1\}$ for all $j \neq g$.
If $d_g > b$, then $m'_{gg} < d_j - b$ for all $j \neq g$,

Complete Graph: Formal Poof

Proposition

If L is complete, mechanism π^{so} fully implements the principal's objective.

Consider L complete.

Suppose $\hat{m}$ is eq. and $\pi_g^{so} < 1$ for some θ.

Then g can deviate to $m'_{gg} = \max\{0, d_g - b\}$ and $m'_{gj} = \min\{d_j + b, 1\}$ for all $j \neq g$.
If $d_g > b$, then $m'_{gg} < d_j - b$ for all $j \neq g$,
and if $d_g \leq b$, then $m'_{gg} = 0$ and $\bar{r}_j > d_g + b$ for all $j \neq g$.

Complete Graph: Formal Poof

Proposition

If L is complete, mechanism π^{so} fully implements the principal's objective.

Consider L complete.

Suppose $\hat{m}$ is eq. and $\pi_g^{so} < 1$ for some θ.

Then g can deviate to $m'_{gg} = \max\{0, d_g - b\}$ and $m'_{gj} = \min\{d_j + b, 1\}$ for all $j \neq g$.
If $d_g > b$, then $m'_{gg} < d_j - b$ for all $j \neq g$,
and if $d_g \leq b$, then $m'_{gg} = 0$ and $\bar{r}_j > d_g + b$ for all $j \neq g$.
In each case $\pi_g^{so'} = 1$ and $\hat{m}$ was not an eq.

Back

π^{so} partially implements for all other L: Formal Proof

Claim

For all L,
$\hat{m}_{ii} = \max\{d_i - b, 0\}$ *and* $\hat{m}_{ij} = \min\{d_j + b, 1\}$ *for all* $j \in N_i$*,* θ_i *and* i *is a dominant strategy equilibrium of* $\Gamma(\pi^{so})$ *such that* $\pi_g^{so} = 1$ *for all* θ*.*

π^{so} partially implements for all other L: Formal Proof

Claim

For all L,
$\hat{m}_{ii} = \max\{d_i - b, 0\}$ and $\hat{m}_{ij} = \min\{d_j + b, 1\}$ for all $j \in N_i$, θ_i and i is a dominant strategy equilibrium of $\Gamma(\pi^{so})$ such that $\pi_g^{so} = 1$ for all θ.

$\hat{m}$ is a dom. strat. eq. because $\hat{m}_i(\theta_i)$ is a dom. message for all θ_i and i.

π^{so} partially implements for all other L: Formal Proof

Claim

For all L,
$\hat{m}_{ii} = \max\{d_i - b, 0\}$ and $\hat{m}_{ij} = \min\{d_j + b, 1\}$ for all $j \in N_i$, θ_i and i is a dominant strategy equilibrium of $\Gamma(\pi^{so})$ such that $\pi_g^{so} = 1$ for all θ.

$\hat{m}$ is a dom. strat. eq. because $\hat{m}_i(\theta_i)$ is a dom. message for all θ_i and i.

$\pi_g^{so}(\hat{m}(\theta)) = 1$ for all θ because

π^{so} partially implements for all other L: Formal Proof

Claim

For all L,
$\hat{m}_{ii} = \max\{d_i - b, 0\}$ and $\hat{m}_{ij} = \min\{d_j + b, 1\}$ for all $j \in N_i$, θ_i and i is a dominant strategy equilibrium of $\Gamma(\pi^{so})$ such that $\pi^{so}_g = 1$ for all θ.

$\hat{m}$ is a dom. strat. eq. because $\hat{m}_i(\theta_i)$ is a dom. message for all θ_i and i.

$\pi^{so}_g(\hat{m}(\theta)) = 1$ for all θ because
if $d_g > b$, $m_{gg} < mjj$ for all $j \neq g$, and

π^{so} partially implements for all other L: Formal Proof

Claim

For all L,
$\hat{m}_{ii} = \max\{d_i - b, 0\}$ *and* $\hat{m}_{ij} = \min\{d_j + b, 1\}$ *for all* $j \in N_i$, θ_i *and* i *is a dominant strategy equilibrium of* $\Gamma(\pi^{so})$ *such that* $\pi_g^{so} = 1$ *for all* θ.

$\hat{m}$ is a dom. strat. eq. because $\hat{m}_i(\theta_i)$ is a dom. message for all θ_i and i.

$\pi_g^{so}(\hat{m}(\theta)) = 1$ for all θ because
if $d_g > b$, $m_{gg} < mjj$ for all $j \neq g$, and
if $d_g \leq b$, $m_{gg} = 0$ and $\bar{r}_g = d_g + b < \min\{d_j + b, 1\} = \bar{r}_j$ for all $j \neq g$.

π^{so} partially implements for all other L: Formal Proof

Claim

For all L,
$\hat{m}_{ii} = \max\{d_i - b, 0\}$ *and* $\hat{m}_{ij} = \min\{d_j + b, 1\}$ *for all* $j \in N_i$, θ_i *and* i *is a dominant strategy equilibrium of* $\Gamma(\pi^{so})$ *such that* $\pi_g^{so} = 1$ *for all* θ.

$\hat{m}$ is a dom. strat. eq. because $\hat{m}_i(\theta_i)$ is a dom. message for all θ_i and i.

$\pi_g^{so}(\hat{m}(\theta)) = 1$ for all θ because
if $d_g > b$, $m_{gg} < mjj$ for all $j \neq g$, and
if $d_g \leq b$, $m_{gg} = 0$ and $\bar{r}_g = d_g + b < \min\{d_j + b, 1\} = \bar{r}_j$ for all $j \neq g$.
In each case $B_2 = \{g\}$.

Back

π^{so} does not fully implement for all L: Formal Proof

Claim

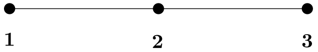
For some L, there is a dominant strategy equilibrium of $\Gamma(\pi^{so})$ such that $\pi_g^{so} = 0$ for some θ.

π^{so} does not fully implement for all L: Formal Proof

Claim

For some L, there is a dominant strategy equilibrium of $\Gamma(\pi^{so})$ such that $\pi_g^{so} = 0$ for some θ.

Consider $b = .2$, N and L s.t.

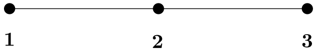


π^{so} does not fully implement for all L: Formal Proof

Claim

For some L, there is a dominant strategy equilibrium of $\Gamma(\pi^{so})$ such that $\pi_g^{so} = 0$ for some θ.

Consider $b = .2$, N and L s.t.



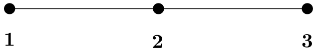
Suppose always $\hat{m}_{ii} = \max\{d_i - b, 0\}$ and $\hat{m}_{i2} = \min\{d_2 + b, 1\}$ for $i = 1, 3$.

π^{so} does not fully implement for all L: Formal Proof

Claim

For some L, there is a dominant strategy equilibrium of $\Gamma(\pi^{so})$ such that $\pi_g^{so} = 0$ for some θ.

Consider $b = .2$, N and L s.t.



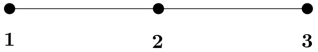
Suppose always $\hat{m}_{ii} = \max\{d_i - b, 0\}$ and $\hat{m}_{i2} = \min\{d_2 + b, 1\}$ for $i = 1, 3$. For $i = 2$, $\hat{m}_{ii} = \max\{d_i - b, 0\}$ and $\hat{m}_{ij} = \min\{d_j + b, 1\}$ for $j = 1, 3$,

π^{so} does not fully implement for all L: Formal Proof

Claim

For some L, there is a dominant strategy equilibrium of $\Gamma(\pi^{so})$ such that $\pi^{so}_g = 0$ for some θ.

Consider $b = .2$, N and L s.t.



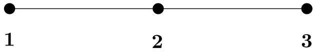
Suppose always $\hat{m}_{ii} = \max\{d_i - b, 0\}$ and $\hat{m}_{i2} = \min\{d_2 + b, 1\}$ for $i = 1, 3$. For $i = 2$, $\hat{m}_{ii} = \max\{d_i - b, 0\}$ and $\hat{m}_{ij} = \min\{d_j + b, 1\}$ for $j = 1, 3$, if not both $d_2 > 0.8$ and $d_1, d_3 < 0.2$.

π^{so} does not fully implement for all L: Formal Proof

Claim

For some L, there is a dominant strategy equilibrium of $\Gamma(\pi^{so})$ such that $\pi_g^{so} = 0$ for some θ.

Consider $b = .2$, N and L s.t.



Suppose always $\hat{m}_{ii} = \max\{d_i - b, 0\}$ and $\hat{m}_{i2} = \min\{d_2 + b, 1\}$ for $i = 1, 3$. For $i = 2$, $\hat{m}_{ii} = \max\{d_i - b, 0\}$ and $\hat{m}_{ij} = \min\{d_j + b, 1\}$ for $j = 1, 3$, if not both $d_2 > 0.8$ and $d_1, d_3 < 0.2$.

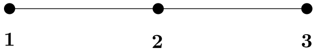
If $d_2 > 0.8$ and $d_1, d_3 < 0.2$, then 2's best application will always be worse for any messages 1 and 3 can send, because $d_i + b < d_2 - b$ for $i = 1, 3$.

π^{so} does not fully implement for all L: Formal Proof

Claim

For some L, there is a dominant strategy equilibrium of $\Gamma(\pi^{so})$ such that $\pi_g^{so} = 0$ for some θ.

Consider $b = .2$, N and L s.t.



Suppose always $\hat{m}_{ii} = \max\{d_i - b, 0\}$ and $\hat{m}_{i2} = \min\{d_2 + b, 1\}$ for $i = 1, 3$. For $i = 2$, $\hat{m}_{ii} = \max\{d_i - b, 0\}$ and $\hat{m}_{ij} = \min\{d_j + b, 1\}$ for $j = 1, 3$, if not both $d_2 > 0.8$ and $d_1, d_3 < 0.2$.

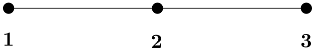
If $d_2 > 0.8$ and $d_1, d_3 < 0.2$, then 2's best application will always be worse for any messages 1 and 3 can send, because $d_i + b < d_2 - b$ for $i = 1, 3$. Thus any message is dominant for 2 and suppose in this case $\hat{m}_{23} = d_1$ and $\hat{m}_{21} = d_3$.

π^{so} does not fully implement for all L: Formal Proof

Claim

For some L, there is a dominant strategy equilibrium of $\Gamma(\pi^{so})$ such that $\pi^{so}_g = 0$ for some θ.

Consider $b = .2$, N and L s.t.



Suppose always $\hat{m}_{ii} = \max\{d_i - b, 0\}$ and $\hat{m}_{i2} = \min\{d_2 + b, 1\}$ for $i = 1, 3$. For $i = 2$, $\hat{m}_{ii} = \max\{d_i - b, 0\}$ and $\hat{m}_{ij} = \min\{d_j + b, 1\}$ for $j = 1, 3$, if not both $d_2 > 0.8$ and $d_1, d_3 < 0.2$.

If $d_2 > 0.8$ and $d_1, d_3 < 0.2$, then 2's best application will always be worse for any messages 1 and 3 can send, because $d_i + b < d_2 - b$ for $i = 1, 3$. Thus any message is dominant for 2 and suppose in this case $\hat{m}_{23} = d_1$ and $\hat{m}_{21} = d_3$.Then $\hat{m}$ is a dom. strat. eq. and if θ s.t. $d_1, d_3 < .2$ and $d_2 > .8$, then $\pi^{so}_g = 0$.

Back