Hidden Actions Analysis in p2p Systems

Weiyu Zhang, Oleg Naroditsky

December 6, 2009

Currency Based Incentive

- Peers would earn currency by contributing resources, and spend currency to obtain resources. e.g. MojoNation, Karma

Currency Based Incentive

- Peers would earn currency by contributing resources, and spend currency to obtain resources. e.g. MojoNation, Karma
- Golle et al. (2001) shows if peers are charged an amount proportional to their gap between uploads and downloads, strict Nash equilibrium exists.

Currency Based Incentive

- Peers would earn currency by contributing resources, and spend currency to obtain resources. e.g. MojoNation, Karma
- Golle et al. (2001) shows if peers are charged an amount proportional to their gap between uploads and downloads, strict Nash equilibrium exists.
- Friedman et al. (2006) shows if peers play threshold strategy, Nash equilibrium also exists. The amount of money in the system can affect the system efficiency.

Currency Based Incentive

- Peers would earn currency by contributing resources, and spend currency to obtain resources. e.g. MojoNation, Karma
- Golle et al. (2001) shows if peers are charged an amount proportional to their gap between uploads and downloads, strict Nash equilibrium exists.
- Friedman et al. (2006) shows if peers play threshold strategy, Nash equilibrium also exists. The amount of money in the system can affect the system efficiency.
- Problems : whitewashing attack.

Hidden Actions in p2p Systems

- "Hidden Actions" means strategic behaviour of individuals with "selfish" goals, while their actions are hidden from the rest of the network. e.g. Request forward.

Hidden Actions in p2p Systems

- "Hidden Actions" means strategic behaviour of individuals with "selfish" goals, while their actions are hidden from the rest of the network. e.g. Request forward.
- We'll use a currency-based mechanism to motivate agents to exert costly effort while their actions are *hidden* from the principal.
  Example: Hidden actions in OR.

Principal-Agent Model

- Principal employs n agents. Each agent $i \in N$ has an action $a_i \in \{0, 1\}$, and an cost $c(a_i) \geq 0$ i.e. $c(0) = 0, c(1) = c$.

Principal-Agent Model

- Principal employs n agents. Each agent $i \in N$ has an action $a_i \in \{0, 1\}$, and an cost $c(a_i) \geq 0$ i.e. $c(0) = 0, c(1) = c$.
- The outcome of the game depends on the *technology* $t : A_1 \times \cdots \times A_n \to [0, 1]$. $t(a_1, \ldots, a_n)$ represents the probability of success.

Principal-Agent Model

- Principal employs n agents. Each agent $i \in N$ has an action $a_i \in \{0, 1\}$, and an cost $c(a_i) \geq 0$ i.e. $c(0) = 0, c(1) = c$.
- The outcome of the game depends on the *technology* $t : A_1 \times \cdots \times A_n \to [0, 1]$. $t(a_1, \ldots, a_n)$ represents the probability of success.
- Principal gains $v > 0$ from a project success, and gains nothing from a project failure.

Principal-Agent Model

- Principal employs n agents. Each agent $i \in N$ has an action $a_i \in \{0, 1\}$, and an cost $c(a_i) \geq 0$ i.e. $c(0) = 0, c(1) = c$.
- The outcome of the game depends on the *technology* $t : A_1 \times \cdots \times A_n \to [0, 1]$. $t(a_1, \ldots, a_n)$ represents the probability of success.
- Principal gains $v > 0$ from a project success, and gains nothing from a project failure.
- Contract: commitment to pay agent i $p_i \geq 0$ if the project succeed, and 0 otherwise.

Special technology: Read-once Network

- Read-once network is a subclass of the technology that could be represented as a source-sink graph. Each agent controls the presence of an edge (probabilistically) and project succeed if there is a s-t path.

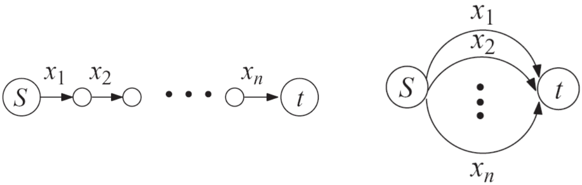

Figure: And technology and Or technology

- If the technology is anonymous (symmetric), then $t(a) = t(m)$ where m is the total number of agents who exert efforts.
- Some technology is not read-once network, e.g. MAJORITY

Hidden Action v.s. Observable Action

- In observable action case, principal could control the action of agents by giving them incentives at the beginning.
- In hidden action case, principal could also choose which agents should be contracted to exert effort. Otherwise, the incentives are promise, and the principal cannot observe the agents' action.
- The incentive is studied when the joint actions of agents are at Nash Equilibrium.

Utility in Hidden Action Model

- For agent i, utility under the profile actions $a = (a_1, \ldots, a_n)$

$$u_i(a) = p_i t(a) - c(a_i)$$

Utility in Hidden Action Model

- For agent i, utility under the profile actions $a = (a_1, \ldots, a_n)$

$$u_i(a) = p_i t(a) - c(a_i)$$

- For principal, expected utility is evaluated when joint actions a are at Nash Equilibrium.

$$u_{ha}(a, v) = t(a)(v - \sum_{i|a_i=1} p_i)$$

Utility in Hidden Action Model

- For agent i, utility under the profile actions $a = (a_1, \ldots, a_n)$

$$u_i(a) = p_i t(a) - c(a_i)$$

- For principal, expected utility is evaluated when joint actions a are at Nash Equilibrium.

$$u_{ha}(a, v) = t(a)(v - \sum_{i|a_i=1} p_i)$$

Definition

$\Delta_i(a_{-i}) = t(1, a_{-i}) - t(0, a_{-i}) \leq 1$ is marginal contribution.

Lemma

Given profile actions a_{-i}, the best strategy of agent i is $a_i = 1$ if $p_i \geq \frac{c}{\Delta_i(a_{-i})}$ and $a_i = 0$ if $p_i \leq \frac{c}{\Delta_i(a_{-i})}$.

Optimal Contract for Principal

- In observable-actions cases, principal can induce effort with $p_i = c$ to agent i. Principal total utility is social welfare

$$u_{oa}(v, a) = t(a)v - \sum_{i|a_i=1} c$$

Optimal Contract for Principal

- In observable-actions cases, principal can induce effort with $p_i = c$ to agent i. Principal total utility is social welfare

$$u_{oa}(v, a) = t(a)v - \sum_{i|a_i=1} c$$

- In hidden-actions case, principal need $p_i = \frac{c}{\Delta_i(a_{-i})} \geq c$ to induce effort. Principal's expected utility is

$$u_{ha}(v, a) = t(a)(v - \sum_{i|a_i=1} \frac{c}{\Delta_i(a_{-i})})$$

Price of Unaccountability

- Given the project evaluation v, principal can choose the optimal set of agents to induce the effort in both observable and hidden action cases which optimize the expected utility $u(a, v)$.

$$v^* = \arg\max_v u(v, a) \quad U^*(v) = u(v^*, a)$$

Price of Unaccountability

- Given the project evaluation v, principal can choose the optimal set of agents to induce the effort in both observable and hidden action cases which optimize the expected utility $u(a, v)$.

$$v^* = \arg\max_{v} u(v, a) \quad U^*(v) = u(v^*, a)$$

- Price of unaccountability is the worst ratio between the social welfare in observable action and hidden action cases.

$$POU(t) = \sup_{v>0} \frac{U^*_{oa}(v)}{U^*_{ha}(v)}$$

Structure Technology

- The probability that agent i succeed in its tasks $(0 < \gamma < \frac{1}{2})$

$$f_i(a_i) = \begin{cases} \gamma & : a_i = 0 \\ 1-\gamma & : a_i = 1 \end{cases}$$

- In AND technology, if m of n agents exert effort, then

$$t(m) = \gamma^{m-n}(1-\gamma)^n$$

- In OR technology, if m of n agents exert effort, then

$$t(m) = 1 - (1-\gamma)^{m-n}\gamma^n$$

- To induce the efforts of $m+1$ agents, principal need to pay
  - c to all $m+1$ agents in observable action.
  - $\frac{c}{t(m+1)-t(m)}$(promise) to all $m+1$ agents in hidden action.

Numerical Results(1)

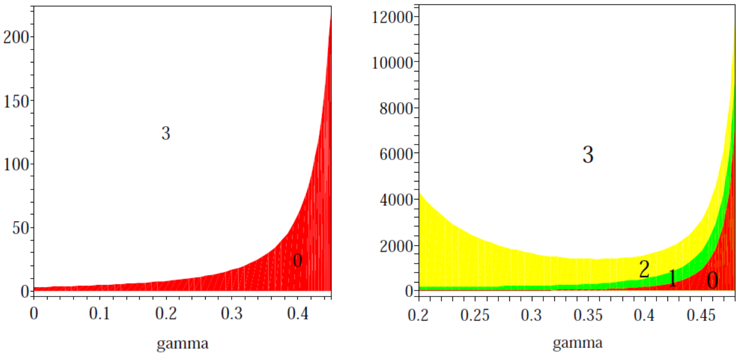

Figure: Number of agents in the optimal contract of the AND(left) and OR(right) technologies with 3 players. No phase transition in intermediate state in AND technology.

Numerical Results(2)

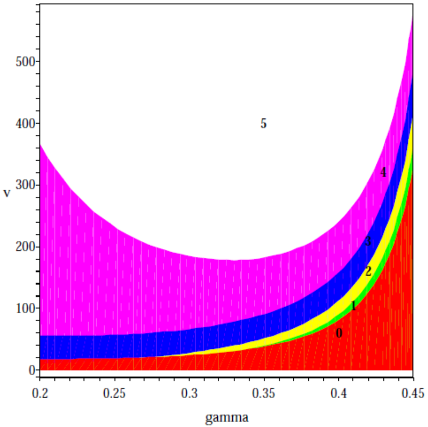

Figure: Number of agents in the optimal contract of the MAJORITY technology with 5 players. The first transition is from 0 to simple majority when γ is very small.

Theorems

Theorem

For AND technology, the optimal contract has a single transition $v*$ with no agent for $v < v*$ and all n agents for $v > v*$. POU is obtained at transition point of hidden action case, and is

$$POU = (\frac{1}{\gamma} - 1)^{n-1} + (1 - \frac{\gamma}{1-\gamma})$$

Theorem

For OR technology, there exist finite positive values $v_1 < v_2 < \cdots < v_n$ s.t when $v_k < v < v_{k+1}$ contracting with k agents is optimal.

Theorem

Solving optimal contract for general read-once networks is # P-hard.