Program equilibria and complexity

H. Vartiainen

February 1, 2007

1 Introduction

- That computers are able to communicate in the "new economy" affects strategic behavior of men in a non-standard way.

- Example: Two players in e-bay auction who program a computer to play the game for them.

  - If the computers are programmed in an *open access* environment, where the source code of any program is public, then a computer can be programmed to *scan* the other player's program before implementing its actions.

  - This affects the optimal programming strategies. Programmer A needs to take into account not only what action player B's computer is programmed to take but also whether the B's computer has been programmed to scan A's computer before implementing its action.

- Resulting *program equilbrium* (Tennenholz, 2004) is the theme of this paper.

- Model programs by finite automata.

- A program equilibrium need not exists. However, this is true only if no computational limitations.

- Main interest is in program equilibria under low complexity costs.

- Conjecture:

  1. In coordination games, beneficial coordination is guaranteed with zero cost. Computers thus enhances welfare as coordination failures cannot occur.

  2. In competitive games á la Matching Pennies the converse holds; competition becomes more intense and leads to additional loss of resources.

2 Model

- Two players 1 and 2

- Symmetric "underlying game" with action set A.

- A finite automaton $M = (\delta, \tau, Q, q^0)$ of player i
  - $\delta : Q \to A$
  - $\tau : Q \times X \to Q$,
  - finite set of states Q,
  - initial state $q^0 \in Q$.

- $|M|$ the size of machine M, measured by the cardinality of Q.

- Denote the set of all finite machines $\mathcal{M}$ (countably infinite).

- Finite alphabet X.

- Given input $x = (x^0, ..., x^n)$, machine $M = (\delta, \tau, Q, q^0)$ implements action $\delta[x] \in A$, where $\delta[x] = \delta(q)$ for $q \in Q$ such that $q = q^n$ and $\tau(q^k, x^k) = q^{k+1}$ for all $k = 0, ..., n-1$.

- M is enconded and distinguished by a string $\xi(M)$ of alphabets. Assume:

A1 If equally long $\xi(M)$ and $\xi(M')$ differ only in the final entry, then $\delta_i[x] \neq \delta_i'[x]$ for all x.

A2 If $|M| > |M'|$, then the length of $\xi(M)$ is at least $\xi(M')$.

- Player i's vNM preferences are represented by a utility function $u$of the form

$$u_i(a_i, a_j) - c(M), \tag{1}$$

where function $c : \mathcal{M} \to \mathbb{R}_+$ is the *complexity cost* such that

$$c(M) \geq c(M) \;\; \text{if} \;\;\; |M| \geq |M| \, .$$

- The machine game is defined as follows. First players choose machines M_1 and M_2. Then machine M_i takes as input $\xi(M_j)$, the code of machine M_j, $j \neq i$. Finally, machine $M_i = (\delta_i, \tau_i, Q_i, q_i^0)$ implements $\delta_i[\xi(M_j)]$. Given machines (M_1, M_2) player i's poayff is

$$u_i(\delta_i[\xi(M_j)], \delta_j[\xi(M_i)]) - c(M_i).$$

- Let $\sigma$is a probability distribution on $\mathcal{M}$. Given pair

(σ_1, σ_2), the expected payoff of i is

$$
\begin{aligned}
& v_i(\sigma, \sigma_j) \\
= \quad & \sum_{M_1' \in \mathcal{M}} \sum_{M_2 \in \mathcal{M}} \sigma_1(M_1)\sigma_2(M_2)u_i(\delta[\xi(M_j)], \delta_j[\xi(M \\
& - \sum_{M_i' \in \mathcal{M}} \sigma_i(M_i)c(M_i)
\end{aligned}
$$

Strategy (σ_1, σ_2) forms a *program equilibrium* if

$$v_i(\sigma_i, \sigma_j) \geq v_i(\sigma_i', \sigma_j), \text{ for all } \sigma_i', \text{ for all } i = 1, 2.$$

- If $c(M) = 0$ for all M, then the agent i is said to be *complexity neutral*. In such case (σ_1, σ_2) forms a program equilibrium if $u_i(\sigma_i, \sigma_j) \geq u_i(\sigma_i', \sigma_j)$, for all σ_i', for all $i = 1, 2$.

3 Complexity Neutral Agents

Matching Pennies:

		1	
		A	B
2	A	$1, -1$	$-1, 1$
	B	$-1, 1$	$1, -1$

Take any mixed strategy σ. We argue that a best response by j is a pure strategy or does not exist.

Lemma 1 *For any $M_i \subset \mathcal{M}$, there is M_j such that*

$$\begin{aligned} u_i(\delta_i[\xi(M_j)], \delta_j[\xi(M_i)]) &= -1, \\ u_j(\delta_i[\xi(M_j)], \delta_j[\xi(M_i)]) &= 1. \end{aligned}$$

Lemma 2 *For any $\{M_i^1, ..., M_i^k\} \subset \mathcal{M}$, there is M_j such that*,

$$\begin{aligned} u_i(\delta_i[\xi(M_j)], \delta_j[\xi(M_i^l)]) &= -1, \\ u_j(\delta_i[\xi(M_j)], \delta_j[\xi(M_i^l)]) &= 1. \end{aligned}$$

for all $l = 0, ..., k$.

- For any finite set $S \subset \mathcal{M}$, there is M_j that is a best response against all $M \in S$.

- This holds, in particular, for any finite subset of the support of i's strategy σ.

Proposition 3 *Let agents be complexity neutral. There is no program equilibrium in the Matching Pennies game..*

4 Complexity Averse Agents

B c is unbounded.

- For any $n \in \mathbb{N}$ there is $M \in \mathcal{M}$ such that $c(M) > n$. Now there is a finite subset S of $\mathcal{M}$such that

$$\max_{M \in S} \min_{a_i, a_j} [u_i(a_i, a_j) - c(M)] > \max_{\substack{a_i, a_j, \\ M \notin S}} [u_i(a_i, a_j) - c(M)].$$

- Thus only machines in S can possibly be chosen in program equilibrium. Since S is finite and payoffs continuous with respect to the randomization over S, a program equilibrium exists.

Proposition 4 *Under B, a program equilibrium exists.*

4.1 Small Complexity Costs

- Let $c(\cdot) = \xi \cdot y(\cdot)$, where $\xi > 0$ is a scalar and $y(\cdot)$ is nondercreasing, continuous and bounded. Interest in the limit $\xi \rightarrow 0$.

Proposition 5 *In Matching Pennies, the program equilibrium payoffs approach* -1 *when* $\xi \rightarrow 0$.

- Define a *coordination game*:

$$\begin{array}{ccc|c|c|} & & \multicolumn{1}{c}{1} & \multicolumn{1}{c}{} \\ & & \multicolumn{1}{c}{A} & \multicolumn{1}{c}{B} \\ \cline{3-4} 2 & A & 0,0 & 1,1 \\ \cline{3-4} & B & 1,1 & 0,0 \\ \cline{3-4} \end{array}$$

Proposition 6 *In the coordination game, the payoff maximizing symmetric program equilibrium payoffs approach* 1 *when* $\xi \rightarrow 0$.