Multi Agent Systems, Lecture 4, November 29, 2006

Summary by Ola Rozenfeld

December 8, 2006

1 Social Systems: The Golden Mean Problem

Suppose we have the following social interaction system: a strategic form game in which the utility of each agent from a given outcome depends not only on the outcome itself, but also on the agent's "goal", where the goal is chosen from a finite fixed set. This obviously can model many real-life situations.

Formally, let $N = \{1..n\}$ be the set of agents, S the strategy set (the same to all agents) and G_{soc} the goal set (also the same to all agents). For every agent $i \in N$ and every $g \in G_{soc}$, $u^i_g \to [0,1]$ is the utility function of agent i with the goal g.

We are interested in the following question: is it possible to restrict the strategy set S to $\overline{S}$ so that in the restricted game, every agent will have a safety level of at least ϵ for every goal?

Formally, we look for $\overline{S} \subset S$ such that for every $i \in N$ and every $g \in G_{soc}$ there exists $s \in \overline{S}$ such that for every $s_{-i} \in \overline{S}$, $u^i_g(s, s_{-i}) \geq \epsilon$.

Obviously, such solution is not always achievable – it depends on our requirement for ϵ. First of all, although we demand the strategy space to be symmetric, we place no restriction on the agents' utility functions; in particular, they can be strictly competitive. For instance, in the matching pennies game, if we view it as an instance of this problem (where the only goal of both agents is "to win the game"), we cannot guarantee both agents a safety level above 0. So, obviously, any instance of the golden mean problem can have a solution only if for every goal, there exists at least one outcome in which every agent with this goal is satisfied to the degree of at least ϵ, i.e. cooperation must be possible.

The problem is called "the golden mean problem", since on one hand we want to restrict the agents, so that their goals can be safely achieved (without assuming any rationality of other agents – this is why we use the safety level measurement); on the other hand we don't want to restrict the agents too much, because we do want to leave the possibility of all their goals being achieved.

The main objective of this section is to prove the following result:

Theorem 1 *The golden mean problem is NP-complete.*

First, we argue that the problem is in NP: we can guess $\overline{S}$ and then verify that for each agent and goal there exists a safety level strategy which yields at least ϵ. The verification takes time linear in the size of the input (the numbers $u_g^i(s)$ for every $s \in S^n$).

Now, we must show that the problem is NP-hard. It will suffice to show that for a restriction of the problem, where we have only 2 agents, we restrict the utility values to only $\{0, 1\}$ and we assume that the game is symmetric, i.e. $u_g^I(s,t) = u_g^{II}(t,s)$ for all $s, t \in S$ and $g \in G_{soc}$. In this restricted version $\epsilon = 1$ (since the only possible values are now 0 and 1, and with $\epsilon = 0$ the problem is always trivially solvable).

W.l.o.g. we can assume that $S = \{1, ..., m\}$ and $G_{soc} = \{1, ..., k\}$ for some numbers m, k. Then, all we have to specify as an input to this restricted problem is: m, k and for each $1 \leq g \leq k$ and each pair $(i,j), 1 \leq i,j \leq m$ the number $u_g(i,j) \in \{0,1\}$ which represents the payoff of agent I with the goal g in the outcome (i,j). We assume that the input is given as k matrices M_g of dimensions $m \times m$ of numbers in $\{0,1\}$. The solution to the problem then is to find such a set T of numbers in $\{1, ..., m\}$ so that when all the matrices M_g are restricted to contain only the rows and columns in T, in each restricted matrix there is a row of all 1's. If such T exists, we say that the instance of the problem is solvable.

We show that the (restricted) problem is NP-hard using a polynomial time reduction from SAT.

Definition 1 *A CNF (conjunctive normal form) formula is a propositional formula of the following form:* $c_1 \wedge c_2 \wedge ... \wedge c_n$ *where each clause* c_i *is of the form* $l_1^i \vee l_2^i \vee ... \vee l_k^i$ *and each literal* l_j^i *is either a propositional variable* x *or a negation of such variable* $\overline{x}$.

Definition 2 *The SAT problem: given a set of propositional variables* X *and CNF formula* f *over the variables in* X*, determine whether there exists an assignment for the variables in* X *that satisfies* f.

The SAT problem is known to be NP-hard. Therefore, if we present a polynomial time reduction from SAT to golden-mean, it will follow that the golden-mean problem is NP-hard as well.

We must describe a polynomial time computable function φ which transforms an instance f of the SAT problem into an instance $\varphi(f)$ of the golden mean problem, so that f is satisfiable if and only if $\varphi(f)$ is solvable. We define φ as follows: S, the set of strategies, will be the set of literals of the SAT instance ($S = \{x, \overline{x} | x \in X\}$); G_{soc}, the set of goals will correspond to the set of clauses in f; for each clause c_i of f, the corresponding matrix M_i is defined as follows: $M_i(l_j, l_k) = 1$ if l_j appears in c_i and $l_j \neq \overline{l_k}$, and 0 otherwise.

In words, this can be described by the following process: we start with a matrix of 0's; then for each literal l_j that appears in the clause c_i we place 1's in the raw j of the matrix; we then place 0 in any pair of conflicting literals $x, \overline{x}$.

Obviously, this reduction is polynomial time computable; now, we must show that it satisfies the desired property, namely that f is satisfiable if and only if $\varphi(f)$ is solvable.

Direction one: if f is satisfiable, consider the total set T of literals that receive the value "true" in the satisfying assignment θ of f (formally, $T = \{x | \theta(x) = true\} \cup \{\overline{x} | \theta(x) = false\}$). We argue that T is a solution to $\varphi(f)$: since θ is a satisfying assignment of f, it satisfies every clause of f; i.e. in every clause c_i there exists a literal

$l_j \in T$. Since an assignment is by definition non conflicting (does not give a variable both the values $true$ and $false$), we have that $\overline{l_j} \notin T$; then, from the definition of M_i we have that the j'th row contains all 1's.

Direction two: suppose that $\varphi(f)$ is solvable, and let T be the solution to $\varphi(f)$. We define the assignment θ to f as follows: $\theta(x) = true$ if for some matrix M_i (restricted to T) the row j of $l_j = x$ is all 1's; $\theta(x) = false$ if for some matrix M_i (restricted to T) the row j of $l_j = \overline{x}$ is all 1's; the other values of θ are not important. From the definition of M_i's we have that the assignment θ is valid; since T is a solution to $\varphi(f)$, in every matrix M_i (restricted to T) there exists a row of all 1's, which means that the clause c_i is satisfied by θ, which completes our proof.

2 The Development Of Social Conventions

This section presents the paper "On The Emergence Of Social Conventions: Modeling, Analysis And Simulations" by Yoav Shoham and Moshe Tennenholtz (available at Moshe's website). The paper deals with the development of social conventions in a repeated setting. Sometimes social conventions, such as traffic laws, are decided upon in advance; however, in some situations this is inachievable or undesireable. For example, choosing a particular operating system or file format is a convention; we don't want to limit the players to a particular choice in advance – rather we prefer that the players repeatedly make their own choices, while constantly observing the choice of others. During such process, social conventions might naturally develop; below we will explore settings where such development occurs and estimate the time it takes.

First, we formalize the definition of a social law and social convention – by these we simply mean a restriction on the agents' actions:

Definition 3 *A social law sl is a restriction on the set of actions available to the agents. A game G and a social law sl induce a subgame G_{sl} of G that is a restriction of G to the actions which are not prohibited by sl.*

Definition 4 *A social convention is a social law that limits the players to a single action.*

Next, we formalize the meaning of a *rational* social law (or convention) – one that, in some sense, gives all the agents a good payoff:

Definition 5 *Let G be a game, and V a game variable (e.g.: the surplus in the best Nash equilibrium of G, or: the max-min value of G). Let $<$ be an ordering on the possible values of V. A social law sl is rational with respect to V if $V(G) < V(G_{sl})$.*

For example, the social convention "Cooperate" in the Prisoner's Dilemma game is rational with respect to both max-min value and the best Nash equilibrium value.

We will now model the following process: k agents are repeatedly randomly chosen from of a large number of agents to play a game. The payoffs of the game are not known to the agents in advance. At any time, an agent doesn't see his opponents' actions, neither does know whom is he playing against; also, he cannot see games in which he does not participate. The only thing an agent knows is his own actions in all games he participated in, and the payoffs he received – therefore, whenever he gets selected to play, he must decide on his action based on this information alone.

Definition 6 *An n-k-g iterative game consists of a set of n agents, k-person game and an unbounded sequence of ordered tuples of k agents selected with uniform distribution from the given agents.*

Definition 7 *An* action selection function *(or a* learning rule*) is a function from the game's history to an action which is both local and oblivious.*

A local action selection function depends only on the history of the agent – his actions and the corresponding payoffs he received. For example, it does not depend on other players' actions and their corresponding payoffs – we want to explore situations where this information is not necessarily available to the agents.

An oblivious action selection function is one that does not depend on the names of the strategies. The motivation behind the obliviousness requirement is as follows: suppose our agents are a set of robots. We are interested in situations where a system designer (who programs the robots' learning rules) cannot anticipate in advance every situation in which the robots might have to make a choice (e.g. move on the right side of the road or on the left one); therefore, he will have to program rules such as "if you have several choices, each of which has been tried this many times and has yielded this much payoff, then next time make the following choice" – i.e. oblivious learning rules.

We will concentrate on a specific class of games, called *social agreement games*:

Definition 8 *(didn't appear in the lecture) A* social agreement game *is a symmetric 2*2 game*

x,x	*u,v*
v,u	*y,y*

where the following holds:

- $x > 0$ *or* $y > 0$
- $u < 0$ *or* $v < 0$
- *if* $x > 0$ *and* $y > 0$ *then* $x = y$

This is a subclass of games that includes games such as the classical coordination game:

1, 1	-1,-1
-1,-1	1, 1

and a version of the Prisoner's dilemma:

1, 1	-3, 3
3,-3	-2,-2

Definition 9 *An HCR (Highest Cumulative Reward) is a learning rule that specifies that an agent switches to a new action if and only if the total payoff obtained from this action in the last ℓ iterations is highest among all actions and strictly higher than the action currently selected.*

The memory limitation of the agent, ℓ, is a parameter of the algorithm. In the following proofs we will assume that ℓ refers to the total number of last iterations, including those played by other agents; so, after ℓ iterations elapse where an agent is not selected, all that he remembers is the action he decided to play.

We now state the main theorem:

Theorem 2 *Let $n \geq 4$. Given an n-k-g iterative game, where g is a social agreement game, placing no constraints on the initial choices of action by all agents, and assuming all agents employ the HCR rule, then the following holds:*

- *For every $\epsilon > 0$, there exists a bounded number $M(\epsilon)$ such that if the system runs for M iterations then the probability of a social convention being reached is greater than $1 - \epsilon$.*
- *Once a convention is reached, it will never be left.*
- *If a social convention is reached, it will not lead to a payoff lower than the (pure!) max-min value of the game g.*
- *If a social convention exists for g that is rational w.r.t. max-min value, then the social convention reached will be rational w.r.t. max-min value*

Proof: We will show the proof for the pure coordination game – the general case is proved in a similar manner. First, observe that there always exists a pair of agents with identical strategies. Then, notice that the following process can be generated with a probability $p = \frac{1}{f(n)}$ and leads to a rational social convention in $g(n)$ iterations, where both $f(n)$ and $g(n)$ are bounded by $O(n^{poly(\ell,n)})$. The process is defined as follows: a pair of agents (i, j) with the same strategy is selected and meet each other until the rest of the agents forget their past. Afterwards, i meets a member $x \neq j$, and then meets j again. The last step is repeated once for each $x \neq j$. It is easy to see that this process will bring to a rational social convention (all agents will adopt the same strategy). As a result, if the system runs for $M = k \cdot g(n) \cdot f(n)$ iterations, the probability that a rational social convention will not be reached is at most $\left(1 - \frac{1}{f(n)}\right)^{k \cdot f(n)} = \left(\left(1 - \frac{1}{f(n)}\right)^{f(n)}\right)^{k} \simeq e^{-k}$. Taking $k > -\log(\epsilon)$ yields the desired result. The other claims of the theorem are straightforward given the definition of HCR and social agreement games.