Colored Trails: A Formalism for Investigating Decision-making in Strategic Environments

Ya'akov Gal and **Barbara J. Grosz**
Division of Engineering and Applied Sciences
Harvard University
Cambridge, MA 02138

Sarit Kraus[*]
Dept. of Computer Science
Bar Ilan University
Ramat Gan 52900
Israel

Avi Pfeffer and **Stuart Shieber**
Division of Engineering and Applied Sciences
Harvard University
Cambridge, MA 02138

Abstract

Colored Trails is a research testbed for analyzing decision-making strategies of individuals or of teams. It enables the development and testing of strategies for automated agents that operate in groups that include people as well as computer agents. The testbed is based on a conceptually simple but highly expressive game in which players, working individually or in teams, make decisions about how to deploy their resources to achieve their individual or team goals. The complexity of the setting may be increased along several dimensions by varying the system parameters. The game has direct analogues to real-world task settings, making it likely that results obtained using Colored Trails will transfer to other domains. We describe several studies carried out using the formalism, which investigated the effect of different social settings on the negotiation strategies of both people and computer agents. Using machine learning, results from some of these studies were used to train computer agents. These agents outperformed other computer agents that used traditional game theoretic reasoning to guide their behavior, showing that CT provides a better basis for the design of computer agents in these types of settings.

1 Introduction

As heterogeneous group activities of computer systems and people become more prevalent, it is important to understand the decision-making strategies people deploy when interacting with computer systems. Colored Trails (CT) is a test-bed for investigating the types of decision-making that arise in task settings where the key interactions are among goals (of individuals or of groups), tasks required to accomplish those goals, and resources. The CT architecture allows games to be played by groups comprising people, computer agents, or heterogeneous mixes of people and computers. The purpose of the CT framework is to enable to design, learn and evaluate players' decision-making behavior as well as group dynamics in settings of varying complexity.

The rules of CT are simple, abstracting from particular task domains, enabling investigators to focus on decision-making rather than the specification of domain knowledge. In this respect CT is similar to the games developed in behavioral economics [1]. However, unlike behavioral economics games, CT provides a clear analog to multi-agent task settings, can represent games that are larger in size, and provides situational contexts and interaction histories in which to make decisions.

The CT environment allows a wide range of games to be defined. Games may be made simple or complex along several dimensions including the number of players and size of the game; information about the environment available to different players; information about individual agents available publicly to all players, to subgroups, or only privately; the scoring rules for agents; the types of communication possible among agents; and the negotiation protocol between agents.

At the heart of CT is the ability of players to communicate with each other, enabling them to commit to and retract bargaining proposals and to exchange resources. The conditions of these exchanges, group dynamics and players' behavior towards others are some aspects that can be investigated in these types of settings.

1.1 Rules of the Game

CT is played by two or more players on a rectangular board of colored squares with a set of chips in colors chosen from the same palette as the squares. For each game of CT, any number of squares may be designated as the goal. Each player's piece is located initially in one of the non-goal squares, and each player is given a set of colored chips. A piece may be moved into an adjacent square, but only if the player turns in a chip of the same color as the square.

Players are removed from the game if they reach the goal state or have been dormant for a given number of moves, as specified by a game parameter. When all players have been removed, the game is declared over and each player's score is computed. The scoring function of a CT game can depend on the following criteria: the position of a player on the board; the number of chips the player possesses; the number of moves made by the player throughout the game; the score of other players in the game. It is possible to vary the ex-

[*]Sarit Kraus is also affiliated with the Institute for Advanced Computer Studies at the University of Maryland (UMIACS).

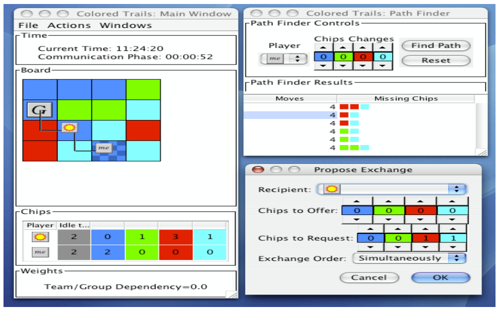

Figure 1: A snapshot of a two-player game

tent to which the scoring function depends on any of these parameters.

The game controller makes available to each player a list of suggested paths to the goal that are displayed in a panel on the screen. These paths are optimized for a given chip distribution and player, as queried by the player, such that they represent the best route given a player's objectives. The ability to access this information is contingent on a player's ability to view the board and chips, as specified by the game parameters.

A snapshot of a two-player game is presented in Figure 1. Here, the Main Window panel shows the board game, player icons and the goal state, as well as the chip distribution for the players. In this game, both *me* and *sun* players lack chips to get to the goal. The *me* player has queried the Path Finder panel and has chosen a path, outlined on the board, for which it lacks a red and cyan chip. It is about to ask the *sun* player for these chips, using the Propose Exchange panel.

Players in CT negotiate with each other during specified communication phases. Each message has a list of fields containing the information embedded in the message. Messages may be of the following types.

1. Propose an exchange.
2. Commit to a proposal.
3. Retract a proposal (i.e., a previous commitment).
4. Request/suggest a path to the goal.
5. Send chips to a player.

Note that messages (1) through (4) pass information between players while message (5) transfers chips between players. By setting the game parameters, agreements reached during the communication phase may or may not be binding. For example, a player whose offer was accepted by another player may need to initiate the sending of chips should it[1] wish to fulfill its commitment.

1.2 Analogy with Task Domains

There is a correspondence between CT play and the planning and execution of tasks by a group of agents. Colors correspond to agent capabilities and skills required by tasks: an agent's possession of a chip of a certain color corresponds to having a skill available for use at a time; not all agents get all colors much as different agents have different capabilities and availability. Paths through the board correspond to performing complex tasks, the constituents of which are individual tasks requiring the skills of the corresponding color. Various kinds of requirements on goals and paths followed correspond to different types of group activities and collaborative tasks. For instance, the degree to which an agent's score depends on the performance of other agents may be used to distinguish collaborative teamwork from situations in which group members act in other ways. Also, requiring only that a certain number of agents get to a goal square might correspond to the need for agents to allocate tasks to subgroups or form coalitions to accomplish the actions in a recipe for a collaborative task. Varying the amount of the board an agent can "see" corresponds to varying information about task recipes or resource requirements.

In addition, the inter-dependence between players can be varied. For example, the scoring function may stipulate a *reward dependence* by having the scores of a player depend in some way on the scores of other agents. Second, a *task dependence* arises whenever players lack the chips they need to reach their goals and must depend on other players supplying those chips.

In this paper, we present several studies that were carried out using the CT framework at Harvard and Bar Ilan Universities. Each study is presented in increasing order of the complexity of the CT setting. All results are statistically significant in the 95% confidence interval range. Three types of players interacted in these studies: people, computer agents designed by the experimenters, and computer agents designed by human subjects.

Section 2 describes how a model of human behavior in one-shot games was devised and evaluated using a machine learning approach in a simple CT setting. Section 3 describes a model for negotiation between computer agents in a setting in which agents were uncertain about others' resources as well as their level of helpfulness. Section 4 outlines a study which investigated the effect of reward dependency on the behavior of people and of agents.

2 Learning Social Preferences in One-shot Games

Research in behavioral economics [1] has established that a multitude of sociological factors affect people's behavior when they interact with others. In particular, people have been shown to exhibit preferences for choices that affect others as well as themselves and to vary in the extent to which these factors affect their play. Traditional game-theoretic models cannot naturally capture the diversity of this behavior [5]. In a series of studies [3; 6], we showed that computer agents that explicitly represented social preferences in their model and learned their extent of influence on people's behavior were able to outperform traditional game-theoretic models. In particular, they were able to generalize to new situations in the game as well as to new players.

[1] we use gender neutral pronouns to refer to players in the game, be they computer agents or people

2.1 The CT set-up

We used a version of CT in which two players played on 4x4 boards with a palette of 4 colors. Each player had full view of the board as well as the other player's tiles. The distribution of tiles at the onset of the game was designed such that (1) at least one of the players could reach the goal after trading with the other player; (2) it was not the case that both players could reach the goal without trading.

The scoring function was chosen so that while getting to the goal was by far the most important component, if a player couldn't get to the goal it was preferable to get as close to the goal as possible. Furthermore, a player's outcome was determined solely by its own performance.

In each game, one player deemed the *allocator* was allowed to propose an offer for exchange of tiles to the other player, deemed the *deliberator*. The deliberator could either accept or reject the allocator's offer. If the allocator did not make an offer, then both players were left with their initial allocation of tiles. The deliberator was not allowed to counter the allocator's offer with another proposal. The score that each player received if no offer was made was identical to the score each player received if the offer was rejected by the deliberator. We referred to this event as the *no negotiation alternative*. The score that each player received if the offer was accepted by the deliberator was referred to as the *proposed outcome* score. Under the conditions specified above, each game consisted of a one-shot negotiation deal between the two players, and a deliberator's reply to the exchange proposed by the allocator completely determines the final outcome of the game.

2.2 Model Construction

Our model predicted whether the deliberator will accept a given proposal, given a CT game. The inputs to the model are NN_A and NN_D, the no-negotiation alternative scores for the allocator and deliberator, and PO_A and PO_D, the proposed outcome scores for the allocator and deliberator.

To develop the model, we introduced the following features, which represented possible social preferences that might affect the deliberator for a given deal. Each feature was derived using the no-negotiation alternative and proposed outcome scores.

- self interest $PO_D - NN_D$
- social welfare $(PO_D + PO_A) - (NN_D + NN_A)$
- advantageous inequality $PO_D - PO_A$
- fair trade $(PO_D - NN_D) - (PO_A - NN_A)$

Given any proposed exchange x, a particular deliberator's utility $u(x)$ is a weighted sum of these features. The weights measure the relative importance of each of the social preferences to the deliberator.

We captured the fact that people make mistakes by implementing a noisy decision function for the deliberator. We defined the probability of acceptance for a particular exchange x by a deliberator as $P(accept \mid x, t) = \frac{1}{1+e^{-u(x)}}$. This probability converges to 1 as the utility from an exchange becomes large and positive, and to 0 as the utility becomes large and negative.

The model assumed that people reason about the same types of social factors, but that individuals weigh them differently. We used a mixture model over types of people, with a probability distribution $P(t)$ over the set of types. Each type t was associated with its own set of social preference weights, defining a utility function u_t.

Given that we have a model describing the deliberator's behavior, the next step was to incorporate this model into a computer agent that played with humans. In our framework, the computer agent played the allocator and a human played the deliberator. The strategy of the allocator was to propose the deal that maximized its expected utility. The expected utility was the sum of the allocator's utility of the proposal times the probability the proposal is accepted, plus the allocator's no-negotiation alternative score times the probability the proposal is rejected. We took the expectation of this sum with respect to all of the deliberator utility functions.

To learn the parameters of the model, we estimated the distribution $P(T)$ over deliberator types, and for each type $t \in T$, we estimated the feature weights. We did this by interleaving two optimization procedures, a version of the EM algorithm [2] and the gradient descent technique. We began by placing an arbitrary distribution over deliberator types and setting the feature weights with particular parameter values.

2.3 Experimental Setup and Results

A total of 42 subjects participated in the experiment, 32 in the data-collection phase and 10 in the evaluation phase. Each phase was composed of playing a number of rounds of different games. A central server was responsible for matching up the participants at each round and for keeping the total score for each subject in all of the rounds of the experiment. Participants were paid in a manner consistent with the scoring function in the game. For example, a score of 130 points gained in a round earned a $1.30 payment. We kept a running score for each subject, revealed at the end of the experiment.

In the data-collection phase, 16 subjects played consecutive CT games against each other Each subject played 24 CT rounds, making for a total of 192 games played. The initial settings (board layout, tile distribution, goal and starting point positions) were different in each game. For each round of the game, we recorded the board and tile settings, as well as the proposal made by the allocator, and the response of the deliberator. The data obtained was then used to learn a mixture model of human play, which included 2 types with probabilities $(0.36, 0.64)$. The feature weights learned for each type were $(3.00, 5.13, 4.61, 0.46)$ and $(3.13, 4.95, 0.47, 3.30)$ for individual-benefit, aggregate-utility, advantage-of-outcome and advantage-of-trade. According to the learned model, both types assigned high weights for social welfare, while still being competitive; one of the types cares more about advantage-of-outcome, and the other type cares more about advantage-of-trade.

The evaluation study consisted of two groups, each involving 3 human subjects and 3 computer players. At each round, eight concurrent games of CT were played in which members of the same group played each other. One of the human subjects, designated as an allocator, played another human subject, designated as a deliberator; each computer player, designated as an allocator, played another human subject, designated as a deliberator.

The computer players, only playing allocators, were agents capable of mapping any CT game position to some proposed exchange. Agent SP proposed the exchange with the highest expected utility, according to our learned social preferences model. Agent NE proposed the exchange corresponding to the Nash equilibrium strategy for the allocator. Agent NB proposed the exchange corresponding to the Nash bargaining strategy for the allocator, consisting of the exchange that maximized the product of each player's individual benefit.

The game settings, including board layout, start and goal positions, and initial tile distributions, were the same for all of the games played by members of the same group. Therefore, at each round there were 4 matching CT games being played by the eight members of each group.

The following table presents the results of the evaluation phase for each of the models used in the experiment.

Model	Total Reward	Proposals Accepted	Proposals Declined	No Offers
SP	2880	16	5	0
NE	2100	13	8	0
NB	2400	14	2	5

It lists the total monetary reward, the number of proposals accepted, the number of proposals rejected, and the number of times no offer was proposed. The SP agent had achieved a significantly higher utility than the other computer agents. It also had the highest number of accepted proposals, along with the allocations proposed by humans. The performance of NE was the worst of the three. The computer allocator labeled NE always proposed the exchange that corresponded to the allocator's strategy in the (unique) sub-game perfect Nash equilibrium of each CT game. This resulted to offering the best exchange for the allocator, out of the set of all of the exchanges that are not worse off to the deliberator. As a consequence, many of the exchanges proposed by this agent were declined. We hypothesize this was because they were not judged as fair by the human deliberator. This result closely follows the findings of behavioral game theory. The computer allocator labeled NB consistently offered more to the deliberator than the NE player did for the same game, when the board and tile distribution enabled it. Because NB tended to offer quite favorable deals to the deliberator, they were accepted more than the other computer players, provided that an offer was made. but its overall reward was less than SP.

While we have focused on one particular game for practical reasons, the learned models we used were cast in terms of general social preferences, which did not depend on the specific features of the game and were shown to be exhibited by people in many types of interactions.

3 Modeling Agents' Helpfulness in Uncertain Environments

When agents depend on each other to achieve their goals, they need to cooperate in order to succeed, i.e. to perform actions that mutually benefit each other. In open systems, there is no central control for agents' design, and therefore others' willingness to cooperate is unknown. To establish cooperative relationships in such systems, agents must identify those that are helpful and reciprocate their behavior, while staying clear of those that are unhelpful. However, in open environments it is difficult to identify the degree of helpfulness of other agents based solely on their actions. This is further made difficult if agents constantly change their strategies.

In this work [7], we built a model which explicitly represented and reasoned about agents' level of helpfulness. The model characterized helpfulness along two dimensions: cooperation (the tendency to propose mutually beneficial exchanges of resources) and reliability (the tendency to fulfill commitments).

We used a version of CT in which two or four players played on boards of different sizes. Each player had knowledge of the scoring function and full view of the board but could not see the other player's chips. Agreements reached during the communication phase were not binding and thus agents could deceive each other by not fulfilling their commitments. A player was declared "out-of-game" if it reached the goal state or if it stayed dormant for 3 moves, at which point its score was computed. Each player's outcome depended solely on its own performance.

3.1 Model Construction

We wanted the model to be able to generalize to environments which varied the number of players, the size of the board-game, and the task dependency between players. To do this, the model explicitly reasoned about others' level of helpfulness, rather than their utility functions.

We described agents' helpfulness along two dimensions with range $[0, 1)$.

- Cooperation (c) - measured an agent's willingness to share resources with others in the game through initiating and agreeing to beneficial proposals.
- Reliability (r) - measured agents' willingness to keep their commitments in the game through delivering the chips they had agreed to.

Given some action a, opponent j, and state s, an agent's utility function depended on the following features.

- The helpfulness measure of agent i, denoted P_i.
- Agent i's estimate of the agent j's helpfulness, denoted P_j. This was estimated as the fraction of times j was cooperative and reliable when interacting with i in the past, decayed by a discount factor.
- The expected value of taking action a given the state of the environment s, denoted $EV_i(a \mid s)$. This quantity estimated the likelihood of getting to the goal, negatively correlated with the number of chips lacked by the agent.
- The expected cost of future ramifications of taking action a, denoted $EC_i(a)$. This function rewarded actions that were beneficial to agent j and punished actions that reneged on commitments.

We constructed a utility function that was a linear combination of these features associated with weights that were tuned empirically. Agents negotiated using this utility function at each communication phase in the game, by performing each action in the subset of actions that fulfilled the following conditions: there were no two actions in the subset that conflicted (for example, two exchange proposals that offered the same chips); the combined utility value for the agent from each action in the subset was highest compared to any other subset with non-conflicting actions. Using this utility function, agents' behavior was contingent on their perception of others, as well as their own helpfulness.

3.2 Experimental Design

We used two class of agents in our study. The first consisted of two types: Multiple-Personality (MP) and Single-Personality (SP) agents. Both MP and SP class agents use the model described earlier to make their decisions. However, the cooperation and reliability levels of an SP agent were constant, whereas an MP agent adopted different measures of cooperation and reliability for each personality type of its opponents based on a matching scheme, derived empirically. Both MP and SP agents were adaptive: they changed their behavior as a function of their estimate of others' helpfulness, given the history of their observations. However, the MP agent adopted a unique measure of helpfulness for each player, whereas the measure of helpfulness for the SP agent was constant.

Another class of agents was Peer-Designed (PD) agents, created by graduate-level computer science students at Bar Ilan University who were not given any explicit instructions regarding agents' strategies and reasoning processes.

We classified PD and SP agents as either "helpful" or "unhelpful". Helpful SP agents were those that engaged in cooperative exchanges more than 50% of the time and reneged on their commitments less than 20% of the time. We expected helpful agents to realize opportunities for exchange with each other more often than unhelpful agents and to exceed them in performance, as measured by the score in the game. We also expected that in some cases, unhelpful agents would be able to take advantage of the vulnerability of those helpful agents that allow themselves to be exploited. We hypothesized that the MP agent would be able to identify and reciprocate helpful agents more quickly than SP or PD agents, while staying clear of agents that are unhelpful. As a result, the MP agent would perform better than all other agents in the game.

We evaluated the MP agent by playing a series of repeated games with the other agents in the systems. We allowed agents to update their model of others from game to game. Each agent's final outcome was the aggregate of its scores in all of the games it participated in.

In our experiment we executed 5,040 games, played in 1,080 rounds of three consecutive games each. The board games we used in each round varied the task dependency relationships between players. The players in each game included a MP agent, two SP agents, and one of the PD agents. Each group of four players played all possible task dependency roles, to control for any effect brought about by dependency relationships. Table 1 presents the average score for the MP agent when playing against helpful and unhelpful agents across all games. The scores reported in the table sum over the other players in the game.

The average score achieved by the MP agent was significantly higher than all other agents, regardless of their level of helpfulness. Also, the MP agent's score when playing against helpful agents (170.6) was higher than its score when playing against unhelpful agents (142.5). Helpful agents also benefited from cooperating with the MP agent: their performance was significantly higher than their unhelpful counterparts (114.8 vs. 98.2).

	MP agent	PD and SP agents
Helpful	170.6	114.8
Unhelpful	142.5	98.2

Table 1: Average performance of MP agent against helpful/unhelpful agents (3 repeated games)

Exchange Type	Helpful agents	Unhelpful agents
Reciprocal	60%	25%
Idle	20%	39%

Table 2: Percentage of exchange types proposed by MP agent

Further investigation revealed that the MP agent engaged in cooperative exchanges with helpful agents significantly more often than the other agents, while the amount of time the MP agent remained idle when dealing with unhelpful agents was longer than the amount of time other agents remained idle.

Another hypothesis was that any group of agents would increase its overall social welfare when playing with an MP agent, because the MP agent would help them to realize more beneficial exchanges. To evaluate this claim, we ran a series of 2-player repeated games that included SP and PD type agents, but did not include MP agents, and compared it to the performance of each agent type after after including an MP agent in the group. The results are described in Figure 2. The performance of helpful and unhelpful agents increased significantly when interacting with the MP agent. As expected, this increase was more profound for helpful SP and PD agents.

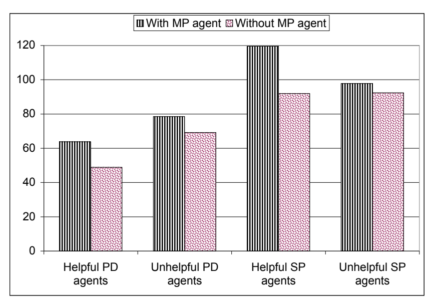

Figure 2: Performance with/without MP agent

4 The Influence of Reward Dependencies on Decision Making

In the work presented here [4], we investigated the effect of social dependencies on players' behavior in CT. We varied the

		# reached goal	Private score
People	$RD = 0$	2.47	151.3
	$RD > 0$	2.8	252.34
PD	$RD = 0$	1.1	82.4

Table 3: Results for people vs. PD agents

social dependency between players by including a ”reward dependency factor” RD in the scoring function; If RD was zero, a player's score was independent of the performance of other players; if it was non-zero, a player's score was a combination of that player's individual score and a weighted average of the individual scores of all the other players.

We hypothesized that (1) higher RD will increase the amount of cooperation between agents. In particular, we expected agents to give other agents chips more frequently and to ask for fewer chips in exchange when RD is higher. (2) when RD weight is high, agents will score higher and reach the goal more often.

The CT games were played by groups of four players, the board was 6x6, the palette was 5 colors, and there was a single goal square for all players. Players were able to see the full board but not each others' chips. This restriction separates decisions about chip exchanges from scoring information, thereby making helpful behavior distinct from score optimization computations.

Two classes of experiments were performed, one involving 4-player groups of people and the other two involving 4-player groups of computer agents. The human player groups were drawn from a population of upperclass and master's computer science students at Bar Ilan University who were not experts in negotiation strategies nor in economic theories directly relevant to agent design (e.g., game theory, decision theory). We compared their performance with that of Peer Designed (PD) agents, who were constructed in a similar fashion as described in Section 3.

A comparison of the results in Table 3 for human players when $RD = 0$ with those when $RD > 0$ supports this hypothesis. The average private score of all games played by people in which $RD > 0$ was significantly higher than in the games where $RD = 0$ In addition, the number of human players who reached the goal in games in which $RD > 0$ was significantly higher than for games with $RD = 0$ This shows that people realized more opportunities for exchange when their performance depended on others. Thus, the main hypotheses regarding RD are supported by the results of games played by people. Interestingly, the PD designs were not influenced by the reward dependencies, and agents did not act significantly different in either condition. This suggests that implicit mention of reward-dependence in the design specification may not affect behavior. In contrast, this same incidental mention of RD in instructions to people playing the game did engender different behavior as discussed below.

Another interesting discovery was that the average private score for people was significantly higher than the average private score of the PDs in these games. Furthermore, the average number of people reaching the goal in these games was significantly higher than the average number of PDs reaching the goal. Further investigation revealed that this was because people were significantly more likely to engage in cooperative exchanges.

5 Conclusion and Future Work

In this paper, we have motivated the need for understanding the decision-making strategies people deploy when computer systems are among the members of the groups in which they work. It has reviewed several studies, all using the CT framework, which analyzed the effects of various social settings on people's behavior, and built computer agents to match people's expectations in these settings.

In the future, we plan to extend the CT formalism in several realms. First, we are designing a system for evaluation of CT models, which would be able to dynamically configure dependent and independent variables, run a series of CT games using an online database of game configurations, game status, and results.

Second, we are constructing a model for repeated negotiation between players which reasons about the reputation of agents. This model will incorporate such features as reward and punishment, and the affinity of players to each other based on their actions. It will learn the extent to which these features affect decision making through incorporating observations of people's play.

6 Acknowledgments

The research reported in this paper has been supported in part by National Science Foundation grants no. 0222914, IIS-0222892 and IIS-0091815.

References

[1] C.F. Camerer. *Behavioral Game Theory: Experiments in Strategic Interaction*. Princeton University Press, 2003.

[2] A.P. Dempster, N.M. Laird, and D.B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39(1), 1977.

[3] Y. Gal, A. Pfeffer, F. Marzo, and B. Grosz. Learning social preferences in games. In *Proc. 19th National Conference on Artificial Intelligence (AAAI)*, 2004.

[4] B. Grosz, S. Kraus, S. Talman, and B. Stossel. The influence of social dependencies on decision-making. Initial investigations with a new game. In *Proc. 3rd International Joint Conference on Multi-agent systems (AAMAS)*, 2004.

[5] J.H. Kagel and A.E. Roth, editors. *The hanbook of experimental economics*. Princeton University Press, 1995.

[6] F. Marzo, Y. Gal, A. Pfeffer, and B. Grosz. Social preferences in relational contexts. In *IV Conference on Collective Intentionality*, 2004.

[7] S. Talman, Y. Gal, M. Hadad, and S. Kraus. Adapting to agents' personalities in negotiation. In *Proc. 4th International Joint Conference on Multi-agent systems (AAMAS)*, 2005.