Learning Social Preferences in Games

Ya'akov Gal, Avi Pfeffer, Barbara J. Grosz
Harvard University

Francesca Marzo
University of Siena

Motivation

- Computer agents are becoming an integral part of people's lives.
- Computer and people are making decisions together.
- People's behavior in social environments is varied and complex.
- We need to build agents that interact successfully with people in these situations.

The Challenge

- People's behavior is affected by a multitude of variables:
  - social preferences, e.g. selfish/altruistic players
  - types of environments, e.g. cooperative/non cooperative
  - social context, e.g. who needs whom
- People make mistakes !

Problem and proposed Solution

- Difficult for analytical approaches (e.g. Game Theory) to capture diversity of behavior.
- To build a *socially adaptive* agent, we need to
  - define social factors in a precise framework.
  - learn them through observing people interact.
  - be able to model different types of people in different types of scenarios.
  - account for people's mistakes.

Our Hypothesis

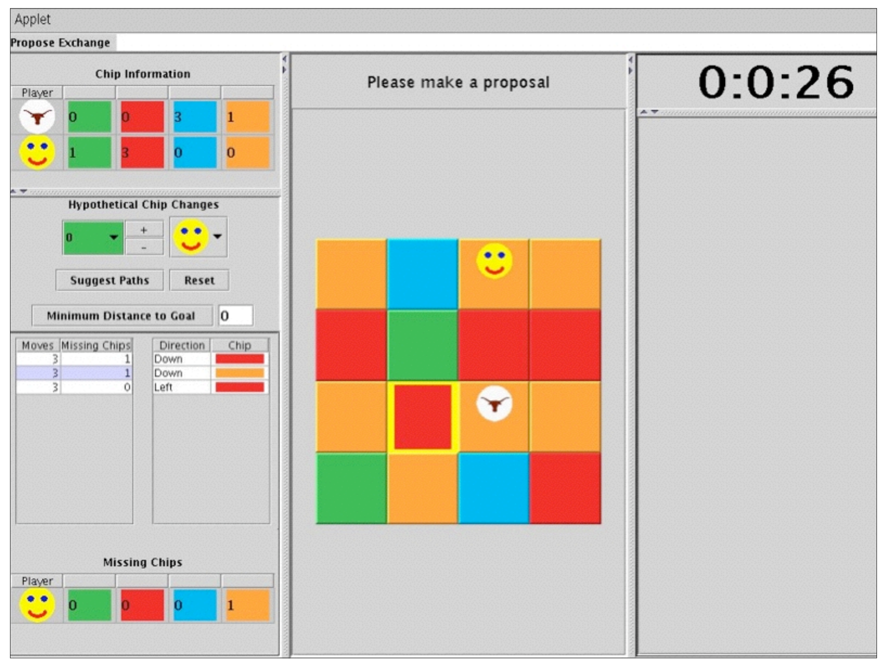
- Agents need to learn social preferences to interact with people.
- A socially competent agent will
  - be more successful (in terms of outcome) than an analytical agent.
  - be able to generalize to people and situations it has not seen before.

Our Approach

- Use a game for testing decision-making in groups comprised of people and computer agents.
- Formalize a social utility function that depends on social preferenes, such as individual benefit, social welfare, advantageous inequality.
- Build a model of human play that incorporates social utility.
- Evaluate model by using it to play against people.

Our Framework

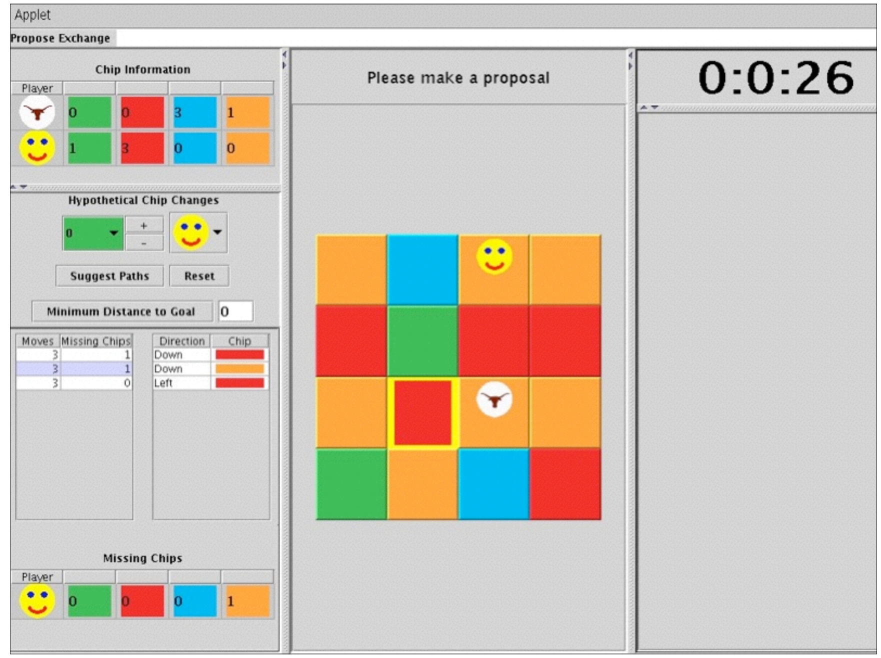
- Colored Trails (CT) [Grosz and Kraus '04]- a computer board game for testing theories of negotiation.
- CT can support humans and machines playing together.
- Each player has a goal; certain resources are needed to reach it.
- Players are allocated resources and can exchange resources in order to reach the goal.

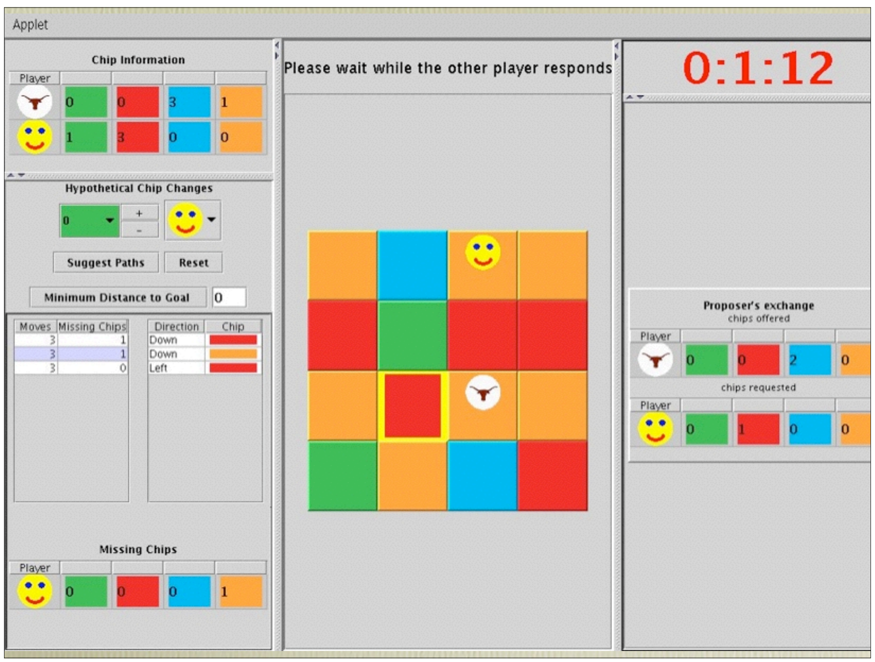


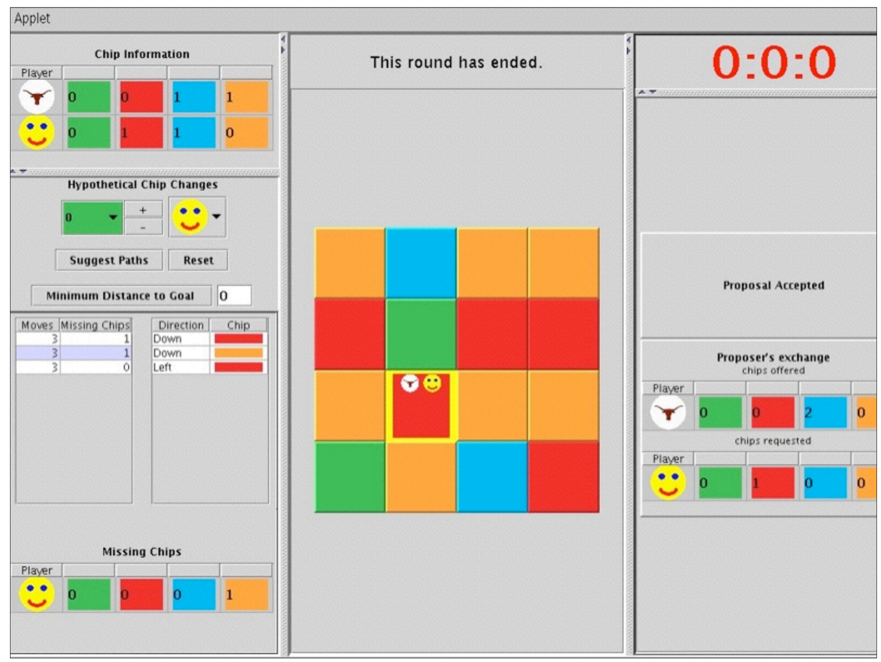
The Scenario

- Game flow comprises of
  1. Allocator makes a proposal.
  2. Deliberator responds to proposal.
  3. Movement towards to goal
- Score depends on distance from goal and number of chips at the end of the game.
- game is non cooperative.

Division of Engineering and Applied Sciences
Harvard University





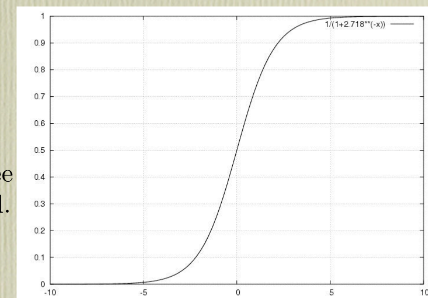


Social preferences in CT

- Reference points
  - No Negotiaion Alternative NN_D, NN_A
  - Proposed Outcome PO_D, PO_A
- Social preferences of Deliberator are defined in terms of outcomes
  - Selfishness $PO_D - NN_D$
  - Social Welfare $(PO_D + PO_A) - (NN_D + NN_A)$
  - Advantage of Outcome $PO_D - PO_A$
  - Advantage of Trade $(PO_D - NN_D) - (PO_A - NN_A)$

Modeling the Deliberator

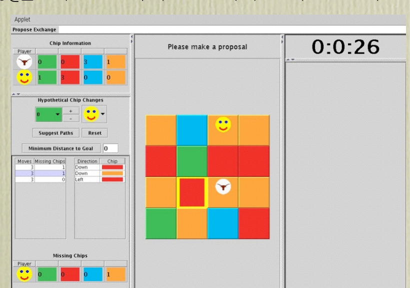
- Given exchange x, *social* utility for Deliberator $u(x)$ is a weighted sum of its social preferences.
- Probability of acceptance of exchange e is $P(accept|x) = \frac{1}{1+e^{-u(x)}}$
- Utility also measures the degree to which a decision is preferred.



Using the model to make a proposal

- Allocator will propose the exchange that maximizes its outcome.

$$e = \text{argmax}_{x \in E} P(accept|x) PO_A(x) + (1 - P(accept|x) NA_A)$$



Modeling different Deliberators

- We use a mixture model of Deliberator types.
- For each type of Deliberator there is a different
  - social utility function
  - probability of accepting a given proposal
- Allocator will propose deal that maximizes expected utility over all possible Responder types.

Data Collection

- used 32 subjects over 2 trials.
- each round consisted of 8 CT games, which paired up different people; altogether 192 data points.
- each data consisted of game description, proposal, and response.
- games were played in various contexts.
- game performance determined payment for subjects.
- parameters learned using EM and gradient descent algorithm.

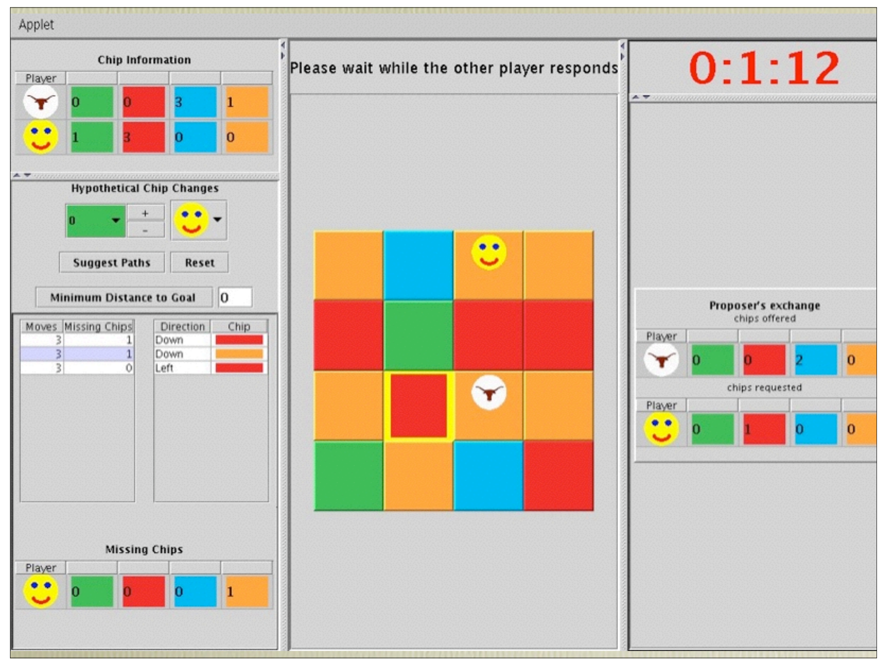
Model evaluation

- We used two groups, each consisting of 5 human subjects and 3 computer players.
- We played 21 different games.
- Each game was played 4 times; once between two people; 3 times between computer Allocator and a human Deliberator.
- We aggregated the rewards of subjects and each type of computer players.

Types of Computer Allocators

- Social Agent - used our social utility model to make an offer.
- Nash equilibrium (NE)
- Nash bargaining (NB)

$$e = \text{argmax}_{x \in E}(PO_D(x) - NN_D)(PO_A(x) - NN_A)$$



Results

Model	Total outcome	Exchanges accepted	Exchnges declined	No offers
Social agent	2880	16	5	0
NE	2100	13	8	0
NB	2400	14	2	5
Human	2440	16	1	4

Example

- Social agent's offer was accepted, the NE offer was declined

Model	Allocator score	Deliberator Score
No Negotiation	75	150
Social Agent	170	170
NE	180	155
NB	150	190

Division of Engineering and Applied Sciences
Harvard University

Related and Future work

- Using CT as testbed for agents designed by humans [Grosz, Kraus, Talman and Stossel '04]
- Modeling repeated interactions between players in CT.
- Limited visibility of board/chips.
- Using CT as a turing-test framework.

Division of Engineering and Applied Sciences
Harvard University

Conclusion

-
- Computers that reason about social behavior are more adept to playing with people.
- Social behavior must be learned.
- Our approach
  - used a framework where computers and people interact together.
  - learned a model of human play.
  - our computer outperformed traditional game theoretic players.

Division of Engineering and Applied Sciences
Harvard University

Thanks to the Harvard AI research group !

{gal, avi, grosz, marzo}@eecs.harvard.edu

Example

- Most declines for social agent occurred when it was already better off.

Model	Proposer score	Responder Score
no offer	180	35
SP	200	15