Towards Social Interactions as Recursive MDPs

1st* Ravi Tejwani
Computer Science and AI Laboratory
Massachusetts Institute of Technology
Cambridge, MA
tejwanir@mit.edu

1st* Yen-Ling Kuo
Computer Science and AI Laboratory
Massachusetts Institute of Technology
Cambridge, MA
ylkuo@mit.edu

2nd Tianmin Shu
Computer Science and AI Laboratory
Massachusetts Institute of Technology
Cambridge, MA
tshu@mit.edu

3rd Boris Katz
Computer Science and AI Laboratory
Massachusetts Institute of Technology
Cambridge, MA
boris@mit.edu

4th Andrei Barbu
Computer Science and AI Laboratory
Massachusetts Institute of Technology
Cambridge, MA
abarbu@mit.edu

***Abstract*—While machines and robots must interact with humans, providing them with social skills has been a largely overlooked topic. This is mostly a consequence of the fact that tasks such as navigation, command following, and even game playing are well-defined, while social reasoning still mostly remains a pre-theoretic problem. We demonstrate how social interactions can be effectively incorporated into MDPs by reasoning recursively about the goals of other agents. In essence, our method extends the reward function to include a combination of physical goals (something agents want to accomplish in the configuration space, a traditional MDP) and social goals (something agents want to accomplish relative to the goals of other agents). Our Social MDPs allow specifying reward functions in terms of the estimated reward functions of other agents, modeling interactions such as helping or hindering another agent (by maximizing or minimizing the other agent's reward) while balancing this with the actual physical goals of each agent. Our formulation allows for an arbitrary function of another agent's estimated reward structure and physical goals, enabling more complex behaviors such as politely hindering another agent or aggressively helping them. Extending Social MDPs in the same manner as I-POMDPs extension would enable interactions such as convincing another agent that something is true. To what extent the Social MDPs presented here and their potential Social POMDPs variant account for all possible social interactions is unknown, but having a precise mathematical model to guide questions about social interactions has both practical value (we demonstrate how to make zero-shot social inferences and one could imagine chatbots and robots guided by Social MDPs) and theoretical value by bringing the tools of MDP that have so successfully organized research around navigation to shed light on what social interactions really are given their extreme importance to human well-being and human civilization.**

***Index Terms*—social interactions, robotics, markov decision processes, multi-agent**

I. Introduction

Progress on modeling social interactions and giving machines social goals, such as being particularly nice to a user, is significantly hampered by the lack of theoretical models which characterize what social interactions are. Great practical progress was made in robot navigation and sensing, with the introduction of MDPs [1] and POMDPs [2]. Defining the problem clearly allowed us as a field to understand what we can model and how to do so. Until we take this same step for social interactions, they will remain on shaky ground despite their importance to virtually every interaction humans engage in.

We introduce an extension of MDPs, which we term Social MDPs. In the process, we make several assumptions. First, that agents have physical goals and social goals, and their overall reward structure is some arbitrary combination of the two, potentially accompanied by other terms. Physical goals are precisely what MDPs can already express, a function of points in a configuration space. Social goals are a function of the estimate of the reward structure of another agent. For example, a reward that hinders another agent is some negative function of the estimated reward of that agent. Complicating matters is the fact that social rewards like beliefs can be recursive: an agent may want to help another agent help them. To model this, Social MDPs are recursive up to some depth, much like interactive POMDPs [3], I-POMDPs. Unlike I-POMDPs, Social MDPs are not recursive in terms of agent's beliefs about the state of the world. Instead, Social MDPs are recursive in terms of the rewards of the agents. This makes Social MDPs and I-POMDPs orthogonal and complementary. Social MDPs are specifically formulated to not interfere with the standard extension from MDPs to POMDPs, making it possible to include partial observability. While we do not develop a joint Social I-POMDP here, this is a reasonable extension which would cover far more of the space of social interactions, although one that is computationally challenging.

Our contributions are:

1) formulating Social MDPs where an agent's reward function is an arbitrary function of the recursive estimate of another agent's reward and a physical goal,
2) an implementation where that function is a linear transformation, which captures many notions of helping and hindering,
3) demonstrating that the model performs zero-shot social

* equal contribution

reasoning in agreement with a human subjects experiment, and

4) examples of the practical utility of recursive social reasoning,

In an anonymized online appendix[1] we fully enumerate all possible scenarios predicted by our model given an environment simple enough to allow doing so, demonstrating that it captures a diverse set of social behaviors. We also provide videos of the behavior of our model in all these scenarios.

II. Related Work

a) Modeling other agents: In order to interact with other agents effectively, an agent must be able to reason about the goals, preferences, and beliefs of other agents [4]. Theory-based models for social goal attribution [5], [6], [7], Bayesian inverse planning to infer an agent's goal given the observations of their behaviors [8], [9], and learning the reward functions of other agents [10] have been explored. Prior research also tried to recognize social interactions such as waving and hugging in videos where people are involved in group activities [11], [12]. These methods generally involve two separate stages [13]: 1) a social perception stage and 2) coordination or collaboration stage where agents interact. In contrast, Social MDPs constantly reevaluate the goals of other agents enabling them to adapt to changes in the plans of other agents. Moreover, MDPs are more efficient than POMDPs, even when solving them recursively. Social MDPs also allow for enumerating social situations by formally defining the space of what social interactions are, opening the doors to a more theoretical approach to social interactions.

b) Learning to interact with other agents: Interactive POMDPs [14], [15] (I-POMDPs) are extensions of POMDPs that recursively model the beliefs of other agents. Social MDP and I-POMDPs are orthogonal. Social MDPs allow agents to reason recursively about other agents' reward functions while I-POMDPs allow agents to reason recursively about other agent's beliefs about the state of the world. The two could in principle be combined, but while Social MDPs require solving a modest number of additional nested MDPs, I-POMDPs require significantly nested inference, and when the two are combined the problem quickly becomes intractable. [16] propose a different type of approach that does not require nested inference: learning a low-dimensional representation of another agent's strategy. This approach allows an agent to avoid another agent or to manipulate another agent into some mutually-beneficial behavior. Social MDPs, on the other hand, allow building the strategy of another agent directly into the reward function of an agent, enabling behaviors such as helping or hindering regardless of what the other agent is trying to achieve. Moreover, Social MDPs are zero-shot, while this prior approach is not. From the point of view of generalization and sample-efficient robotics, a zero-shot approach is preferable; in addition, it opens new doors for a more theoretical understanding of social interactions. We could combine Social MDPs with this prior work to build in latent representations of strategies into reward functions [17], [18] creating more efficient approximations of Social MDPs.

[1] See https://social-mdp.github.io

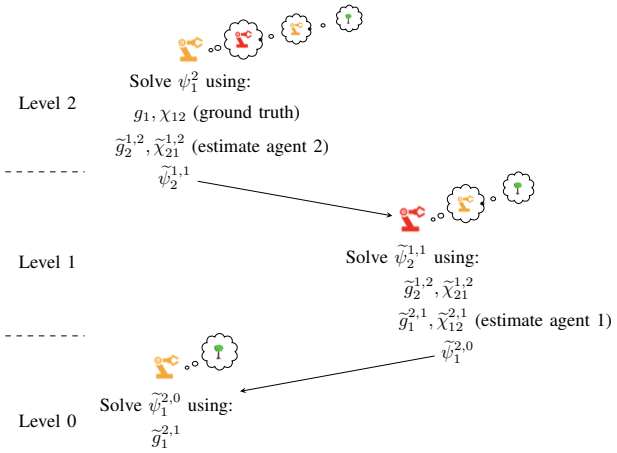


Fig. 1: An example of recursively solving Social MDP for the yellow robot at level 2 in a two-agent interaction scenario. We denote the yellow robot as agent 1 and the red robot as agent 2. At level 2, the yellow robot estimates the red robot's goals (both physical $\widetilde{g}_2^{1,2}$ and social $\widetilde{\chi}_{12}^{2,1}$) and social policy by assuming the red robot is running a level 1 Social MDP. Solving the social policy $\widetilde{\psi}_2^{1,1}$ of the red robot at level 1 requires the red robot to estimate the yellow robot's goals and policy by assuming the yellow robot is running a level 0 Social MDP, i.e., a regular MDP, so we can drop the estimation of $\widetilde{\chi}_{12}^{2,1}$ here. All these estimates are in agent 1's belief space and are updated at every time step.

Require: $l, s^t, a_J^t, \chi_{ij}, g_i$
  if $l = 0$ **then**
    solve MDP for agent i
  else
    for all $\widetilde{\chi}_{ji}^{i,l,t}, \widetilde{g}_j^{i,l,t}$ **do**
      compute
      $P(\widetilde{\chi}_{ji}^{i,l,t} | s^{t-1}, a_J^{t-1})$
      $P(\widetilde{g}_j^{i,l,t} | s^{1:t-1})$
      $\widetilde{\psi}_j^{i,l-1}(s^t, a_J^t, \widetilde{\chi}_{ji}^{i,l,t}, \widetilde{g}_j^{i,l,t})$
    end for
    compute $R_i^l(s^t, a_J^t, \chi_{iJ}, g_i)$
    compute $Q_i^l(s^t, a_J^t, \chi_{iJ}, g_i)$
    $\pi_i^l \leftarrow \text{argmax}_{a_i \in \mathcal{A}_i} Q_i^l$
  end if

Fig. 2: The algorithm to compute social policy ψ_i^l for agent i at level l and time t. We use the estimated social policy $\widetilde{\psi}_j^{i,l-1}$ at previous time step to update the estimated physical and social goal as described in section III-B1. At $t = 0$, we assume $P(\widetilde{g}_j^{i,l,t})$ and $P(\widetilde{\chi}_{jJ}^{i,l,t})$ are from uniform distributions. This algorithm is called at all recursion steps $\widetilde{\psi}_j^{i,l-1}$ to estimate social policy for the other agent $j \in J$. The estimated goals and policies are used to compute the rewards and Q values for selecting the actions.

III. Social MDPs

Social MDPs are recursive MDPs (Markov decision process) with nested estimates of other agent's goals inspired by hierarchical models of games [19] and nested MDP that reason

about the beliefs of other agents [20], [21], [22]. Figure 1 shows an example of recursively estimating the other agent's goals and policy in a two-agent scenario. Like other nested models, e.g I-POMDPs, Social MDPs have the notion of a level. A level 0 Social MDP is simply an MDP: agents reason about the map state. A level 1 Social MDP enables each agent to reason about the physical goals of other agents (those other agents are treated like level 0 agents). A level 2 Social MDP enables each agent to reason about the level 1 social goals of other agents. To perform this nested inference, agents must have access to another agents' physical and social goals. These goals are estimated by solving Social MDPs recursively at every level.

A level 0 agent can take physical actions, but cannot reason socially. A level 1 agent can take actions relative to another agent's physical goals, such as helping, hindering, stealing, etc. A level 2 agent can take actions relative to another agent's social and physical goals, such as avoiding an attempt to be hindered, recognizing that help is needed, joining in to help together. Levels deeper than 2 continue to describe meaningful interactions although we do not consider them here. It is unclear what level of recursion is required before agents exceed the social reasoning capacities of humans.

A. Formal definition of Social MDPs

A Social MDP for agent i with respect to all agents J consists of an arity (here we formulate the pairwise case) and a maximum level, l, and is defined as:

$$M_i^l = \langle \mathcal{S}, \mathcal{A}, T, \chi_{iJ}, g_i, R_i^l, \gamma \rangle \tag{1}$$

where

- $\mathcal{S}$ is a set of states in the environment where $s \in \mathcal{S}$.
- $\mathcal{A} = \mathcal{A}_J$ is the set of joint moves of all agents in J. a_i is an action for agent i.
- T is the probability distribution of going from state $s \in \mathcal{S}$ to next state $s' \in \mathcal{S}$ given actions of all agents in J: $T(s' \mid s, a_J)$.
- χ_{iJ} is agent i's social goal toward every other agent in J. For convenience, χ_{iJ} is a shorthand for $\bigcup_{j \in J, j \neq i} \chi_{ij}$.
- g_i is agent i's physical goal.
- R_i^l is the l-th level reward function for agent i based on its estimate of other agents' rewards.
- γ is a discount factor: $\gamma \in (0, 1)$.

a) Reward: Each agent has its own physical goal, e.g., going to a landmark, as well a social goal, e.g., helping or hindering other agents. What enables Social MDPs to go beyond regular MDPs is recursive nature of the reward function which can be written in terms of the estimated rewards of other agents. The immediate reward of an agent i at level l is computed as follows:

$$R_i^l(s, a_J, \chi_{iJ}, g_i) = r_i(s, a_i, g_i) + \sum_{j \in J, j \neq i} \chi_{ij} (\widetilde{R}_j^{i,l-1}(s, a_J, \widetilde{\chi}_{jJ}^{i,l}, \widetilde{g}_j^{i,l})) - c(a_i) \tag{2}$$

where $r(\cdot)$ is the static reward given the agent's own physical goal g_i, $\widetilde{R}_j^{l-1,i}(\cdot)$ is the estimated reward for agent j from agent i's point of view assuming agent j is a level $l-1$ agent, $c(\cdot)$ is the cost for taking an action. For negative levels, the reward is defined to be zero.

χ_{ij} is the social goal, it transforms the reward of another agent j into a goal that is part of the reward of the target agent i. In this paper, we instantiate the model with a linear transformation, so χ_{ij} is simply a reweighting of the estimated reward of the other agent. If it is a negative value, the target agent will attempt to minimize the reward of another agent, i.e. hindering. A positive value corresponds to helping. Social goals can be eliminated entirely by setting this weight to zero.

In order to estimate another agent's reward function, it needs to estimate that agent's physical and social goals. Throughout the paper, we use $\widetilde{\chi}_{jJ}^{i,l}$ and $\widetilde{g}_j^{i,l}$ to denote the estimated social and physical goals. The superscript i, l indicates agent i at level l is making the estimations.

We describe how to estimate the social and physical goals in section III-B1.

B. Planning for Social MDPs

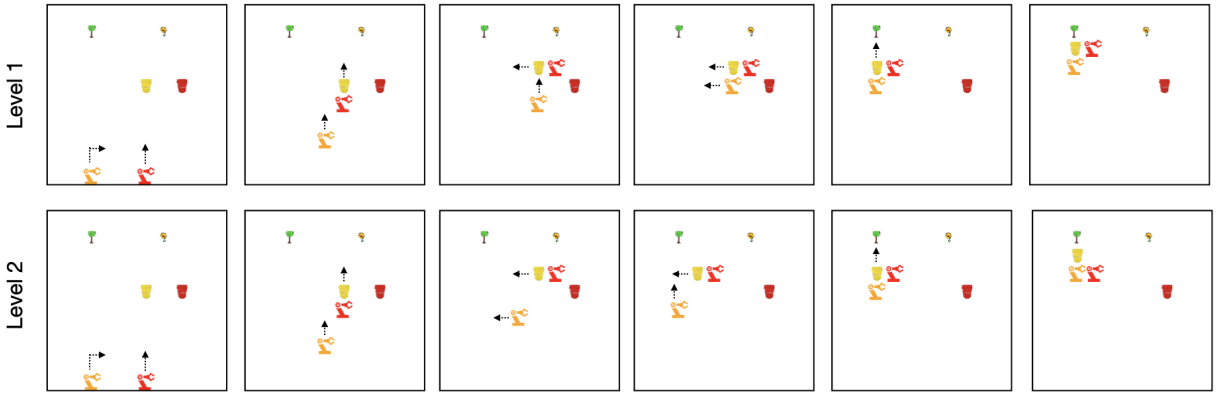
Analogous to MDPs, the Q function of Social MDPs is the sum of immediate reward and the expected value in the future.

$$Q_i^l(s, a_J, \chi_{iJ}, g_i) = R(s, a_i, \chi_{iJ}, g_i) + \gamma \sum_{s' \in S} T(s, a_J, s') V_i^l(s', \chi_{iJ}, g_i) \tag{3}$$

Since agent i is interacting with other agents $j \in J$, it needs to estimate what actions other agents are likely to take in order to compute its state-action value. Social MDPs take the expectation over the estimated goals and actions of agent j to compute $V_i^l(s', \chi_{iJ}, g_i)$:

$$\begin{aligned} V_i^l(s', \chi_{iJ}, g_i) &= \max_{a_i' \in \mathcal{A}_i} \left\{ E_{\widetilde{g}_j^{i,l}, \widetilde{\chi}_{jJ}^{i,l}, a_j'} [Q_i^l(s', a_J', \chi_{iJ}, g_i)] \right\} \\ &= \max_{a_i' \in \mathcal{A}_i} \left\{ \sum_{\substack{j \in J, \\ j \neq i}} \sum_{a_j' \in \mathcal{A}_j} \sum_{\widetilde{g}_j^{i,l}} \int_{\widetilde{\chi}_{ji}^{i,l}} \underbrace{P(\widetilde{g}_j^{i,l} | s^{1:t})}_{\substack{\text{estimate physical goal} \\ \text{(Eq. 6)}}} \right. \\ &\quad \left. \underbrace{P(\widetilde{\chi}_{ji}^{i,l} \mid s, a_J)}_{\substack{\text{estimate social goal} \\ \text{(Eq. 5)}}} \underbrace{\widetilde{\psi}_j^{i,l-1}(s', a_J', \widetilde{\chi}_{ji}^{i,l}, \widetilde{g}_j^{i,l})}_{\substack{\text{estimate social policy} \\ \text{(Eq. 7)}}} Q_i^l(\cdot) d\widetilde{\chi}_{ji}^{i,l} \right\} \end{aligned} \tag{4}$$

When solving agent i's MDP at level l, the estimated social and physical goals are further used to update the other agent j's social policy to the actions agent j may take. We denote the estimated social policy for agent j at reasoning level $l-1$ as $\widetilde{\psi}_j^{i,l-1} : \mathcal{S} \times \mathcal{A}_J \times \widetilde{\chi}_{jJ}^{i,l} \times \widetilde{g}_j^{i,l} \to [0, 1]$. Figure 2 summarizes the steps to compute the state-action values and select optimal actions for any level l at time step t. We first update the probability of the estimated goals of other agents using the observed state and the estimated policy from the previous time step. The updated probability of goals are used to update the policy of other agents and compute the reward and Q function of the target agent. The recursion happens at estimating the social policies of other agent at a lower level.



Scenario 28: Red wants to help yellow ($\chi_{\text{red,yellow}} = 2$)

Fig. 3: Example of zero-shot social interactions.The Social MDP gives the robots the ability to understand and predict relationships, thereby making far more efficient actions. The yellow robot wants to water the tree. Moving the yellow bucket is easy for the yellow robot, while moving the red bucket is hard for the yellow robot. The yellow robot performs inference to understand what the red robot is doing. With a level 1 Social MDP, the yellow robot assumes that the red robot has a physical goal, but not a social goal. With a level 2 Social MDP, the yellow robot assumes that the red robot has both a physical and social goal, then recursively estimates the social goal of the red robot (which is in turn modeled as a level 1 Social MDP)

1) Updating social and physical goals of other agents: An agent's estimate of another agent's social and physical goals at time step t and level l can be updated based on the actions performed by the agents. At time step $t = 0$, we use uniform distributions for social and physical goals.

The social goal, estimated at time step t, is updated after actions taken by all agents at the previous time step. This update is similar to the belief update in the POMDP framework but based on the estimated social policy of the other agent j:

$$P(\widetilde{\chi}_{ji}^{i,l,t} \mid s^{t-1}, a_J^{t-1}) \propto P(\widetilde{\chi}_{ji}^{i,t-1} \mid s^{t-2}, a_J^{t-2}) \sum_{\widetilde{g}_j^{i,l,t-1}} P(a_j^{t-1} \mid s^{t-1}, \widetilde{\chi}_{ji}^{i,l,t-1}, \widetilde{g}_j^{i,l,t-1}) \times T(s^{t-1}, a_J^{t-1}, s^t) \quad (5)$$

The physical goal g_j of agent j is estimated by i as follows, similar to [23] but marginalized over the estimated social goal as the agent is estimating the social goal at the same time.

$$P(\widetilde{g}_j^{i,l,t} | s^{1:t-1}) \propto \int_{\widetilde{\chi}_{ji}^{i,l,t}} P(s^{1:t-1} | \widetilde{g}_j^{i,l,t}, \widetilde{\chi}_{ji}^{i,l,t}) \cdot P(\widetilde{g}_j^{i,l,t}) \cdot P(\widetilde{\chi}_{ji}^{i,l,t})\, d\widetilde{\chi}_{ji}^{i,l,t} \quad (6)$$

2) Estimating social policies of other agents: The l-level social policy $\widetilde{\psi}_j^{i,l}$ of the agent j is predicted by i using the Q-function at level l-1:

$$\widetilde{\psi}_j^{i,l-1}(s, a_J, \widetilde{\chi}_{jJ}^{i,l}, \widetilde{g}_j^{i,l}) = \text{Softmax}(Q_j^{l-1}(s, a_J, \widetilde{\chi}_{jJ}^{i,l}, \widetilde{g}_j^{i,l})) \quad (7)$$

This is a softmax policy where use a temperature parameter τ to control how much the agent j follows the greedy actions. As shown in eq. 4, in order to use agent j's Q function at level l-1, it requires to compute agent i's Q function at level l-2, and so on. This involves solving Social MDPs recursively at levels $0, 1, \cdots, l$-1.

C. Time complexity

The time complexity of solving a Social MDP at level 0 is the same as that of solving an MDP. At level 1, an MDP must be solved for every agent independently in order to compute the likely physical goals of every other agent. Assume that the number of models considered for each pair of agent at each level is bounded by a number M (based on the number of social and physical goals to consider). Solving a Social MDP at level l requires solving $O(M(A-1)^2 l)$ MDPs, where A is the number of agents. Social MDPs form a tree with branching factor $A-1$ as every agent must compute the pairwise social goal of every other agent until level 0 where the tree bottoms out. There are many potential speedups that can alleviate this runtime to allow for efficient inference even in the face of many agents. For example, a distance horizon could be used where far away agents could simply be considered non-interacting. Similar to [24], it is also possible to speed up the algorithm by amortized inference over goals and relations by training a neural net to recognize goals and relations as initial guesses and refine them through probabilistic inference.

IV. Results

We apply the Social MDP framework to a multi-agent grid world inspired by previous studies on social perceptions [9], [5], [25]. The 10×10 world consists of two agents, a yellow robot and red robot, two physical landmarks, a flower and tree, and two objects, a yellow water can and red water can. The yellow agent has a low cost for moving the yellow watering can, while it has a high cost for moving the red watering can. Robots can have a physical goal of moving the water can to a target plant. Robots can have a social goal of helping or hindering to different degrees. In the grid world, agents can move in four direction (left, right, up, down) or choose not to move.

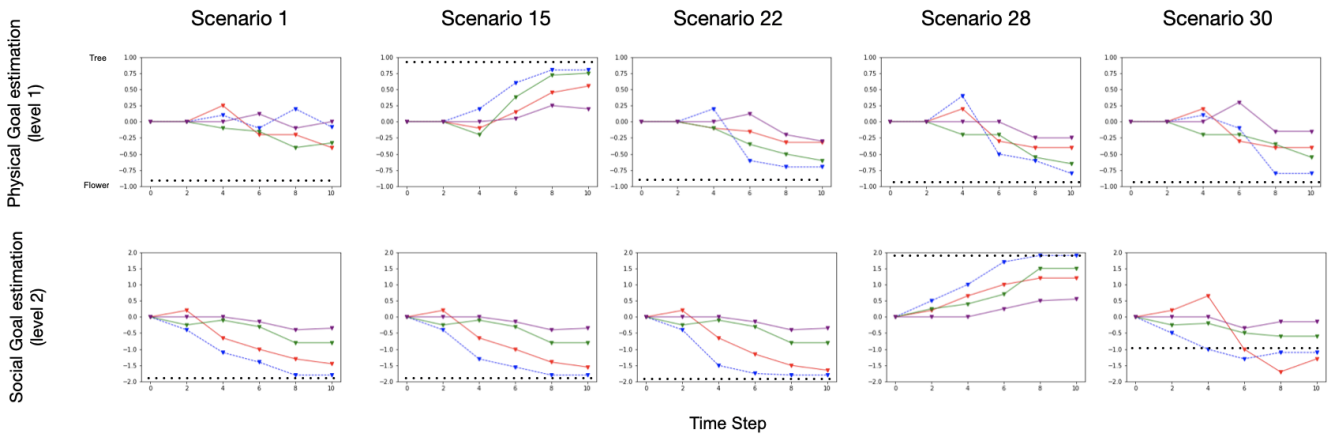
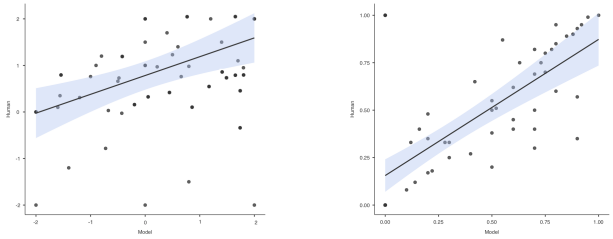


Fig. 4: A deep dive into how humans and each model interprets the five experiment scenarios (refer Appendix for results for all experiment scenarios) at both levels at each time step (in red Human scores, in blue our Social MDP scores, in green Inverse planning [9], in purple the cue-based model [23], and **in dotted the ground truth**). The goal of each model is to interpret how one agent perceives another. (top) At level one, an agent has a belief over the physical goal of another agent. Humans and models predict what this belief is (the degree to which the agent believes that the other agent is heading toward the tree or the flower). Note that all models perform rather well and follow human judgements. (bottom) At level two, an agent has a belief over the physical and social goals of another agent. Humans and models predict what the beliefs of the agents are about the social goals of other agents. In other words, to what degree does this agent think that the other agent is hindering or helping them. Here our model fits human data much better because of its recursive nature. At deeper levels, our model is capable of capturing social interactions and social inferences that other models cannot. Other models are confused, and so predict that there is a very weak or non-existent social goal in most cases while our model follows human judgements.



(left) Weight of social goals (right) Weight physical goals

Fig. 5: Twelve human subjects, and our model, the Social MDP, watched and scored 10 videos at different snapshots. These videos consist of five scenarios where robots reason at either level 1 or level 2 (presented to the users in randomized order). The straight black line represents the best linear fit to the data, and the light blue band around the line shows the uncertainty in the linear fit. The light blue band represents a 95% confidence interval. (left) Models and humans were asked to predict how social the agents were and the valence of the interaction (was it positive or negative). Non-social settings have a weight of 0, while adversarial settings have a social weight of -2, overwhelming the physical goal of any agent. Humans and machines predict similar social goals both in terms of value and magnitude. (right) Models and humans were asked to predict a weight factor on the physical goal, how much does this agent care about its physical goal. At 0, the physical goal is ignored. At 1, it is weighted equally with a social goal also set at 1. Human and model scores are again highly correlated. Our model is able to effectively generate trajectories that humans recognize as being social interactions. It is also able to predict the type of social interaction that humans believe occurred.

98 different experiment scenarios[2] are systematically created

[2]Interactions for the experimental scenarios can be viewed at https://social-mdp.github.io/scenarios

	Social MDP (ours)	Inverse Planning	Cue-based
Social Goal	**0.85**	0.78	0.20
Physical Goal	**0.76**	0.71	0.07

Fig. 6: The coefficient of correlation with 95% confidence interval between machine judgements and ground truth(final 2 time steps) for all the 98 experiment scenarios (each scenario has agents having either the same or different physical goals along with one of 7 different scaling factors on each of their social goals (-2, -1, -0.5, 0, 0.5, 1, 2)). Refer to appendix for detailed results for each scenario. We provide two baselines and our own approach. The cue-based model is described in [23]. The inverse planning model is described in [9]. Social MDPs produce better alignment with ground truth than other models and do not require training like the cue-based model.

in this grid world. Each scenario has agents as having either the same physical goal or different physical goals and one of 7 different scaling factors on each of their social goals (-2, -1, -0.5, 0, 0.5, 1, 2) ($2 * 7 * 7 = 98$ scenarios). All 98 experiment scenarios correspond to reasonable interactions between agents. The degree to which this is true in more complex environments and the degree to which systematically unfolding the model in more complex environments always results in what humans would describe as social interactions is an important topic for future work

Each agent's reward for reaching its physical goal is based on that agent's geodesic distance from the goal after taking an action [9]. This physical reward function is parameterized by ρ and δ that determines the scale and shape of the physical reward: $r_i(s, a, g_i) = \max\left(\rho\left(1 - \text{distance}(s, a, g_i)/\delta\right), 0\right)$. We set the cost, c, of an action a, to 1 for grid moves and 0.1 to staying in place while ρ and δ were set to 1.25 and 5, respectively.

To quantitatively establish the quality of the social inferences

made by the Social MDPs, we compare human judgements of 12 subjects against those of two baseline models: inverse planning [9] and a recent cue-based model [23]. Humans and models had to estimate the physical and social goals of agents in these environments when the agents were acting both as level one agents (unaware that the other agents are also social) and as level two agents (who could account for the fact that the other agents are social). In fig. 5 we show the raw judgements of humans and of our models, along with a best linear fit. The performance of all models against human judgements, was measured through correlation coefficient at 95% confidence level, for social goal estimation (r = 0.89 for the Social MDP vs. r = 0.81 for the Inverse Planning model vs. r = 0.23 for the Cue-based model) and physical goal estimation (r = 0.78 for the Social MDP vs. r = 0.72 for the Inverse Planning model vs. r = 0.08 for the Cue-based model). Our model performs considerably better than other models. This is even more evident in the deep dive shown in fig. 4. For level one agents, agents that are social but that assume that other agents are not social, all models agreed with human judgements. Yet, for level two agents, agents that are social and can assume that other agents are also social, our models are far better aligned with human judgements.

V. Conclusion

Social MDPs are a first step toward a theory of social interactions that fits within the established frameworks we have in robotics. They can perform zero-shot social recognition and planning for diverse situations. The fact that MDPs can be extended in a natural way that is also computationally tractable to account for many social interactions by nesting inference and allowing models to take arbitrary functions of the estimated rewards of other agents has not been noted before. Our experiments clearly show that Social MDPs are superior to prior models and account for more social interactions.

We have only begun to explore what Social MDPs can represent in this work. The environment we consider is very simple, yet, at the same time, more than enough to differentiate Social MDPs from other models. We have unrolled Social MDPs only two levels, what exists at deep levels is still unclear. It is likely that humans do not perform deeply-nested recursive reasoning to carry out social interactions, although, what the cutoff is, and if Social MDPs are close enough to a human's mental model to allow for measuring that cutoff is unknown.

We would like to see in the future that any MDP-based system can be augmented to be social by a straightforward extension with Social MDPs. Much like virtually any approach can be easily augmented to partially-observed environments using POMDPs. Social MDPs and POMDPs are compatible, exploring their combinations and the implications of partial observability for social interactions remains as future work.

References

[1] R. Bellman, "A markovian decision process," *Journal of mathematics and mechanics*, vol. 6, no. 5, pp. 679–684, 1957.

[2] K. J. Åström, "Optimal control of markov processes with incomplete state information," *Journal of Mathematical Analysis and Applications*, vol. 10, no. 1, pp. 174–205, 1965.

[3] P. J. Gmytrasiewicz and P. Doshi, "Interactive pomdps: Properties and preliminary results," in *International Conference on Autonomous Agents: Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems-*, vol. 3, 2004, pp. 1374–1375.

[4] S. V. Albrecht and P. Stone, "Autonomous agents modelling other agents: A comprehensive survey and open problems," *Artificial Intelligence*, vol. 258, pp. 66–95, 2018.

[5] C. L. Baker, N. D. Goodman, and J. B. Tenenbaum, "Theory-based social goal inference," in *Proceedings of the thirtieth annual conference of the cognitive science society*. Cognitive Science Society, 2008.

[6] C. L. Baker and J. B. Tenenbaum, "Modeling human plan recognition using bayesian theory of mind," *Plan, activity, and intent recognition: Theory and practice*, vol. 7, pp. 177–204, 2014.

[7] N. Rabinowitz, F. Perbet, F. Song, C. Zhang, S. A. Eslami, and M. Botvinick, "Machine theory of mind," in *International conference on machine learning*. PMLR, 2018, pp. 4218–4227.

[8] C. L. Baker, R. Saxe, and J. B. Tenenbaum, "Action understanding as inverse planning," *Cognition*, vol. 113, no. 3, pp. 329–349, 2009.

[9] T. D. Ullman, C. L. Baker, O. Macindoe, O. Evans, N. D. Goodman, and J. B. Tenenbaum, "Help or hinder: Bayesian models of social goal inference," MASSACHUSETTS INST OF TECH CAMBRIDGE DEPT OF BRAIN AND COGNITIVE SCIENCES, Tech. Rep., 2009.

[10] D. Hadfield-Menell, S. J. Russell, P. Abbeel, and A. Dragan, "Cooperative inverse reinforcement learning," *Advances in neural information processing systems*, vol. 29, pp. 3909–3917, 2016.

[11] A. Patron-Perez, M. Marszalek, I. Reid, and A. Zisserman, "Structured learning of human interactions in tv shows," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 12, pp. 2441–2453, 2012.

[12] M. S. Ryoo and J. K. Aggarwal, "Spatio-temporal relationship match: Video structure comparison for recognition of complex human activities," in *2009 IEEE 12th international conference on computer vision*. IEEE, 2009, pp. 1593–1600.

[13] X. Puig, T. Shu, S. Li, Z. Wang, J. B. Tenenbaum, S. Fidler, and A. Torralba, "Watch-and-help: A challenge for social perception and human-ai collaboration," in *The International Conference on Learning Representations*, 2021.

[14] P. Doshi and P. J. Gmytrasiewicz, "Monte carlo sampling methods for approximating interactive pomdps," *Journal of Artificial Intelligence Research*, vol. 34, pp. 297–337, 2009.

[15] P. Doshi and D. Perez, "Generalized point based value iteration for interactive pomdps." in *AAAI*, 2008, pp. 63–68.

[16] A. Xie, D. P. Losey, R. Tolsma, C. Finn, and D. Sadigh, "Learning latent representations to influence multi-agent interaction," in *Conference on Robot Learning (CoRL)*, 2020.

[17] M. Igl, L. Zintgraf, T. A. Le, F. Wood, and S. Whiteson, "Deep variational reinforcement learning for POMDPs," in *Proceedings of the 35th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, J. Dy and A. Krause, Eds., vol. 80. PMLR, 10–15 Jul 2018, pp. 2117–2126.

[18] A. X. Lee, A. Nagabandi, P. Abbeel, and S. Levine, "Stochastic latent actor-critic: Deep reinforcement learning with a latent variable model," in *Neural Information Processing Systems (NeurIPS)*, 2020.

[19] C. F. Camerer, T.-H. Ho, and J.-K. Chong, "A cognitive hierarchy model of games," *The Quarterly Journal of Economics*, 2004.

[20] I. Shpitser, R. J. Evans, T. S. Richardson, and J. M. Robins, "Introduction to nested markov models," *Behaviormetrika*, 2014.

[21] W. Yoshida, R. J. Dolan, and K. J. Friston, "Game theory of mind," *PLoS Comput Biol*, vol. 4, no. 12, p. e1000254, 2008.

[22] T. N. Hoang and K. H. Low, "Interactive pomdp lite: Towards practical planning to predict and exploit intentions for interacting with self-interested agents," in *Twenty-Third International Joint Conference on Artificial Intelligence*, 2013.

[23] T. Shu, M. Kryven, T. D. Ullman, and J. B. Tenenbaum, "Adventures in flatland: Perceiving social interactions under physical dynamics," in *42d proceedings of the annual meeting of the cognitive science society*, 2020.

[24] A. Netanyahu, T. Shu, B. Katz, A. Barbu, and J. B. Tenenbaum, "Phase: Physically-grounded abstract social events for machine social perception," in *The AAAI Conference on Artificial Intelligence*, 2020.

[25] C. Baker, R. Saxe, and J. Tenenbaum, "Bayesian theory of mind: Modeling joint belief-desire attribution," in *Proceedings of the annual meeting of the cognitive science society*, vol. 33, no. 33, 2011.