Social Feedback for Robotic Collaboration

Emily Wu, Brown University

October 6, 2015

1   Introduction

Collaboration is a process that relies heavily on communication for its success. When humans collaborate, this communication is both obvious and implicit—we not only instruct and request aid from each other, but we also tacitly monitor our partners for signs of approval and understanding while producing these signals ourselves. However, in robotics, many of these vital components of successful communication are missing or lost. The absence of these signals likely account for numerous failures in human-robot collaborative tasks. I propose to create a framework that allows for these missing components to be added to human-robot collaborative tasks, which should result in faster, more accurate, and less frustrating human-robot collaboration experience.

To describe this process of back-and-forth signaling, we use the term *social feedback*. Social feedback refers to the communication and monitoring of state. Examples of the communication of state include explicit statements, such as "I'd like this object", and implicit behaviors, like looking at the object that is desired. Examples of monitoring state include explicitly asking one's partner, "Which object would you like", or implicitly monitoring their behaviors for signs of confusion or unhappiness. Human do both naturally to improve the flow and clarity of the conversation, which in turn improves the success of the conversation. We aim to arm our robot with these communicative skills.

By enabling robots with social feedback, we hope to see several improvements in many domains. Overall, we expect a better human-robot interactive experience: the robot will feel more responsive to human feedback by providing cues of understanding, and be more engaging as it actively communicates with the human, instead of simply obeying its commands. Because of this, we also expect novice users to feel more comfortable collaborating with the robot, as the collaboration will be more analogous to collaboration with humans. Additionally, communicating with the human will allow the robot to use the human as a resource to help it recover from failures. Overall, we expect these changes to result in increased speed and accuracy in human-robot tasks.

2   Existing Work

Previous work has been published towards this goal. In our Wu et al. [2015], we describe a Partially Observable Markov Decision Process (POMDP) [Kaelbling et al., 1998] model of an object handoff task where the robot is capable of explicit monitoring—i.e., asking yes or no questions.

2.1   Domain: Object Handoff

The domain we tested our system on involves the robot handing a requested object to a human. Our Baxter robot is situated by a table with objects located in reach. A human participant requests Baxter hand them an object using natural language and gesture. Depending on whether social feedback is enabled, Baxter can ask the human questions to learn more about which object the human desires.

2.2   POMDPs

We formulate this domain as a POMDP. POMDPs are similar to traditional Markov Decision Processes (MDPs), except the true state is not known, as must be estimated from observatons. Our POMDP is parameterized by seven variables, $\{S, A, T, R, \Omega, O\}$.

S describes the set of states, which in our domain is a tuple $(\omega, \mathcal{O})$. The set of objects on the table is $\mathcal{O}$, and $\omega \in \mathcal{O}$ is the object the human desires. Each object in $\mathcal{O}$ is parameterized by its name, its location, and a

vocabulary of words that describe it. The desired object ω is not known, and can only be inferred via observations. However, we consider $\mathcal{O}$ to be completely observable and to change deterministically.

A is the set of actions that our robot can take. This set always includes the actions `PICK`, where the robot picks and object and hands it to the human, and `WAIT`, where the robot does nothing. If social feedback is enabled, the robot is also capable of choosing the action `ASK(x)` where x is an object property derived from one of the words of the object vocabulary (e.g., "Would you like a wooden object?")

T is the transition function, which describes the probability of transitioning to another state given an action: $p(s'|s, a)$. In this domain, we restrict the set of objects on the object to only change if an object is picked up. Similarly, the desired object only changes if that object is picked, and probability is redistributed uniformly across the remaining objects.

R is the reward function, which takes as input a state and an action and outputs a real-valued reward, incentivizing and disincentivizing the robot appropriately. In this domain, we choose a high positive reward for the robot picking the correct object, and a large negative reward for picking the incorrect object. In addition, we reward `WAIT` actions with a small negative reward and any `ASK(x)` actions are given a slightly higher penalty to disincentivize the robot from asking too many questions and annoying the human.

Ω describes the set of all observations. Observations here are a tuple of (l, g), where l is the human's language as interpreted by speech recognition, and g is the human's gesture, as determined through the position of the human's shoulder and wrist compared to the location of objects on the table.

O is the observation function, which gives the probability $p(o|s, a)$—the probability of witnessing a certain observation given the state that it occured in and the action that triggered it. We consider language and gesture to be conditionally independent given the true state, and can therefore consider them separately. Language we interpret (incrementally) according to a unigram model with probabilities specified by the object's vocabulary. The probability of a gesture given the object is indicated is intepreted to a normal distribution centered at the object. Our observation function also provides a special case for when the human responds "yes" or "no". This is interpreted to be an answer to the last question the robot asked (which is stored as an additional state variable for bookkeeping purposes), `ASK(x)`. The observation of "yes" is more likely for objects that have property `x` and less likely for objects that do not. For "no" it is the opposite. In addition, we model the probability of any communication from the participant to be higher if the robot asks a question the previous turn.

2.3   POMDP Solver: Belief Sparse Sampling

We formulate this POMDP using the Brown UMBC Reinforcement Learning And Planning (BURLAP) Library [MacGlashan, 2015] and use Belief Sparse Sampling [Kearns et al., 2002] to solve the POMDP. However, Belief Sparse Sampling was too slow to be used in user trials, so many optimizations, including caching several thousand simulated interactions, are necessary to allow the code to run at an acceptable pace. For more details about these optimizations, see Wu et al. [2015].

3   Continuing Work

The preliminary work described above serves mostly as a framework and proof of concept for adding a full set of social-feedback tools. For my thesis, I propose improvements to fix weaknesses of the current system and new features to create a more fully social robot.

3.1   Additional avenues of social feedback

Our previous work is incomplete in that the robot's only means of interacting socially with the user was through yes/no questions. In continuing work, we will add new avenues such as eye gaze and gesture, allow the robot to indicate objects by looking and pointing at them. These actions are invaluable for their ability to disambiguate objects that cannot be disambiguated by speech alone, as well as providing a faster and less disruptive means of communication than asking a question. We will also explore more complicated questions that allow the robot learn more about the human's state as well as communicate more about its own. In addition, we will equip the robot with ways to communicate various levels of failure to the human, such as failures to recognize the human's speech, failure to parse it, or failure to act, which will allow the human to repond more helpfully.

3.2 Representing human state

In our existing model, the human's actions are interpreted as observations to allow us to estimate the human's desired object, ω. However, we also acknowledge that our actions affect another parameter, which is which object the human believes we are about to hand them. This distribution over $\mathcal{O}$ we call b_H. We add this to our state, so our new state representation is a tuple of $(\omega, \mathcal{O}, b_H)$.

Adding this additional variable causes our state space to increase significantly. In order to remain computationally tractable, we will make the assumption that b_H begins at a known distribution and changes deterministically. Specifically, we update b_H as if it were a Bayes filter [Eldon, 2015] where the hidden state is the object the robot intends to hand over, and the observations are the robot's actions. In this way, b_H mirrors the belief vector of the belief MDP derived from our POMDP, which we notate as b_R.

By tracking b_H, the robot is able to determine how its actions are perceived by the human. In conjuction with b_R, we can ensure that our state is successfully communicated with the human by taking actions such that b_R, our belief about what the human wants, matches b_H, our estimate of what the human believes we will hand over.

3.3 Incremental Actions

The responsiveness of our system is also important for fluid interaction, so the robot always has an accurate model of the human's current state. For this reason, we interpret language and gesture incrementally. In order to make full use of this incremental interpretation, we should also act incrementally. In order to accomplish this, we will design several incremental actions, which allow the robot to change its actions if new knowledge is made available about the human's state. This means, for every avenue social feedback (i.e., speech, gesture, and gaze), we have both STOP and CONTINUE actions which allow the robot to either continue its current action or change its current action. Crucially, this will allow the robot to cancel the pick if the human provides new information that change the robot's estimate.

3.4 Real-time POMDP Solutions

In order to have incremental interpretation and responses, we must be able to solve our POMDP in real time. To achieve this, we plan to use the Partially Observable Sparse Sampling (POSS) algorithm described in **cite**. This newly developed algorithm is available in the BURLAP library and performs well in domains with large observation spaces such as ours. Solving the POMDP in real time will give us the benefit of acting responsively while freeing us from the restrictions imposed by caching pre-computed solutions.

3.5 New Domains

In this work, we will also explore new domains to test our social feedback system. The current object handoff task is deficient because the task is too simple and clarifications are rarely needed. We will explore additional domains that are more taxing on the communication skills of both human and robot. One task we are considering are object rearrangment, where the robot must rearrange objects according to a pattern the human knows. In this more difficult domain, we expect the benefits of social feedback will be more apparent, and lead to increased speed and accuracy over a baseline implementation of the task. This more difficult domain will also better represent challenges a robot collaborating in a human environment are likely to face.

4 Results

The results from our previous paper are summarized in Table 1. With our small sample size of 6 participants, two trials each, we see a slight improvement in speed, though a decrease in accuracy. These preliminary studies are too brief to make statements regarding the effectiveness of social feedback, but results are promising for future work.

We have since performed additional user studies that suggest that users find the robot more likeable and competent when it uses social feedback, as well as affirmed the small improvement in speed.

We expect adding more avenues of feedback will result in greater improvements to speed and accuracy, and that new, more difficult domains will accentuate these improvements.

Metric	No Feedback	Feedback
Picks Correct	12/13	12/14
EUED Time	17.33s	13.33s
IOD Time	2.59%	3.18%

Table 1: Experimental Results. EUED time is time from the end of the first utterance in the request until the end of delivery of the desired object and IOD time is time spent delivering incorrect objects.

5 Conclusion

Our existing work provides a strong beginning for our explorations in socially enabled robotics. In my thesis work, I will expand upon our existing approach, adding new avenues for social communication and allowing the robot to better monitor the human's state, to fully equip our robot to naturally and fluidly collaborate with humans. The improvements I propose will allow the robot to communicate its state with the human and monitor the human's state to allow it to provide key information to the human and request help when it is needed. These improvements will add robustness to robotic operation that will, with the help of humans, allow it to tackle new tasks it could not solve without human cooperation.

References

Emily Wu, Yuxin Han, David Whitney, John Oberlin, James MacGlashan, and Stefanie Tellex. Robotic social feedback for object specification. *AAAI: Artificial Intelligence for Human-Robot Interaction*, 2015.

Leslie P. Kaelbling, Michael L. Littman, and Anthony R. Cassandra. Planning and acting in partially observable stochastic domains. *Artificial Intelligence*, 101(1-2), 1998.

James MacGlashan. *Brown UMBC Reinforcement Learning and Planning Library*, 2015. `burlap.cs.brown.edu`.

Michael Kearns, Yishay Mansour, and Andrew Y Ng. A sparse sampling algorithm for near-optimal planning in large markov decision processes. *Machine Learning*, 49(2-3):193–208, 2002.

Miles Eldon. Incrementally Interpreting Multimodal Referring Expressions in Real Time. Undergraduate Thesis, May 2015.