In *AAMAS workshop on Autonomous Robots and Multirobot Systems* (ARMS 2022), Online, May 2022

Quantifying Human Rationality in Ad-hoc Teamwork

Yair Hanina[1], Reuth Mirsky[1,2], William Macke[2], Peter Stone[2]

[1] Department of Computer Science, Bar Ilan University, Israel.
yairmofet@gmail.com, mirskyr@cs.biu.ac.il
[2] Department of Computer Science, The University of Texas at Austin, USA
{*wmacke,pstone*}*@cs.utexas.edu*

Abstract. Ad-hoc teamwork is defined as the task of collaborating with teammates without pre-coordination. When the ad hoc agent is a robot that needs to collaborate with people, it cannot assume that its teammates will behave optimally or legibly. By providing a means to learn human policies in ad-hoc teamwork, this work will help create robots that can adapt to a new human agent and work together to achieve a common goal. We focus on a simple, yet powerful model for representing agents using the concept of bounded rationality. Our preliminary results exemplify how such a model can be used in a domain from the ad-hoc teamwork literature called "the tool fetching domain".

1 Introduction

Learning autonomous agents are becoming increasingly capable of solving complex tasks, but encounter many challenges when required to solve such tasks as a team, especially if some of the team members are human. In such human-robot teaming scenarios, the robot often learns how to act prior to the actual interaction, while assuming it has full knowledge of the teammates and the task at hand. However, this process is not suitable to many real world scenarios, where the robot interacts with some of its teammates for the first time and might have not collaborated with them in the past. As a topical example, service robots might be deployed to assist medical teams in an epidemic outbreak. Such heterogeneous robots might be deployed without any prior coordination about each other's capabilities to assist, but they will only be effective if they are able to work together, with the medical team, and with patients, without the need to be explicitly provided with coordination strategies in advance.

Ad-hoc teamwork (AHT) is the task of collaborating with teammates without pre-coordination, which was defined as a formal challenge by Stone et al. [17]. However, when the AHT agent is a robot that needs to collaborate with people or previously unmet agents, it cannot assume that its teammates will behave optimally or legibly [4, 18], and asking too many questions about its teammate's policies risks excessively interrupting the teammate, and thus hindering the collaboration. This work aims to address this challenge by learning how suboptimal human policies tend to be in a specific AHT scenario.

By providing a means to learn human policies in AHT, this work will help create robots that can adapt to any human agent and work together to achieve a common goal. The approach used in the work is optimization using example human trajectories. We focus on a simple, yet powerful model for representing agents with bounded rationality. The main contributions presented in this paper are as follows:

- To the best of our knowledge, this is the first work to investigate whether it is possible to quantify human rationality for artificial agents' decision making in the context of the AHT challenge.
- It presents a basic algorithm that can fit the parameters of a bounded rationality model to learn human behaviour.
- The paper provides preliminary results showing the fitting of the bounded rationality model in a specific domain.

Moving forward, we aim to use the learned model to decide on the action that the robot should take in response to the human behavior in ad-hoc human-robot interactions.

2 Background

In this work, we rely on two main research thrusts and combine them together: AHT, and bounded rationality.

2.1 Ad-hoc Teamwork

Ad-hoc teamwork is defined as the challenge to create an autonomous agent that is able to efficiently and robustly collaborate with previously unknown teammates on tasks to which they are all individually capable of contributing as team members [17]. Previous work investigated the formation of AHT with human team-members [7,9], and Stone et al. [17] formally defined the challenge of designing AHT artificial agents. There are two main properties that distinguish AHT from other multiagent systems. First, it assumes that all teammates strive to be collaborative [6]. Second, the properties of the environment and of the teammates cannot be changed by the AHT agent, and the behavior of the teammates is not necessarily known in advance. The AHT agent's task is to reason and plan towards the team's goals under these conditions.

Subsequent works proposed to model teammates by mapping them to one out of a set of types [1, 14] or by modeling them directly [2]. For approaches which employ a learned type set, learned decision trees were common [3]. More recently, AHT research modeled the information pertaining to teammates' behavior by using classes of types into which any new teammate is classified. Such methods have been used to learn a neural network-based encoder that maps observations of teammates to an embedding of the agent's type [12, 13, 20, 21]. While these works do narrow the gap between learning agents collaborating in simulation and real-world AHT, they have not evaluated human users or human

decision makers as one or more of the teammates in their tasks. Recently, Suriadinata et al. [18] presented an experiment that evaluates human behavior in AHT under three different conditions of instructions prior to the interaction, where the users' performance was evaluated in terms of optimality and legibility. This paper showed that the optimality and legibility values are significantly different between the conditions, thus highlighting the importance of acquiring a model that encompasses different human behaviors.

2.2 The Tool Fetching Domain

One of the simulated domains created to evaluate the performance of AHT agents is called *the tool fetching domain* [11]. This domain is a grid world with workstations, in which there are two teammates – one agent, the worker, needs to reach a specific workstation, while the other agent, the fetcher, needs to fetch the worker an appropriate tool from a toolbox, according to the workstation the worker goes to.

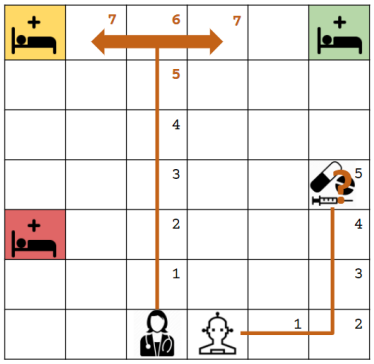

Fig. 1: A depiction of the Tool Fetching domain. The robot is the fetcher that needs to fetch the right medicine to the physician, according to the room the physician is heading for.

This setup means that the fetcher's goal depends on the goal of the worker, and hence its choice of actions relies on its understanding of the worker's goal. For example, as depicted in Figure 3, if the physician (the worker) walks North 6 steps, the fetcher cannot decide whether to pick up the medicine for the patient in the yellow room or for the patient in the green room.

In Suriadinata et al. [18], a human user played the role of the worker, and the fetcher was an artificial agent which was modelled using the baseline agent from Mirsky et al. [11]. That artificial agent is defined as follows: if there is no ambiguity regarding the goal of the worker, the artificial agent acts optimally to minimize the number of actions it takes to fetch the right tool (e.g., if the physician in Figure 3 turns east, it is clear that the fetcher needs to fetch the

medicine for the patient in the green room). If there is some ambiguity regarding the goal of the worker, there are two options: if there is an action that is optimal for all goals with a probability higher than zero, then the fetcher will take that action (e.g., no matter what action the physician takes, the robot needs to first reach the toolbox and hence it will know what to do for the first five steps). Otherwise, the fetcher cannot be certain about which action to take and it needs more information to act optimally. In this situation, the fetcher will wait in its current location until there is at least one optimal action that is common to all likely goals. Notice that this policy depends on the fetcher's mental model of the worker's policy: the fetcher needs to have some assumptions about the behavior of the worker. For the sake of the described experiment, the worker was assumed to behave optimally, but the final results show that people did not always act optimally nor legibly. This result motivates the need to have a better model of human behavior in AHT scenarios, such as the tool fetching domain.

2.3 Modeling Human Rationality

Many approaches exist to represent human rationality and predict how people would behave under specific interaction protocols such as negotiations, autonomous driving, games, and more [15]. These approaches leverage game theoretical models, deep learning, case-based reasoning, and more.

In this work, we focus on a specific approach called *bounded rationality*, originally interpreted by Simon [16] and which draws from psychological and behavioral studies [8]. This approach assumes that people have a limited ability to compute the outcome of every single decision they can make and thus they use a heuristic policy, which might not choose an optimal strategy. Bounded rationality means that when humans make decisions, their rationality is limited by the outcomes from the actions, the difficulty of the challenge, the cognitive capabilities, and the time available to make the decision. In this approach, the human decision maker is often represented using a *utility function* that describes a biased or a sub-rational heuristic policy. One of the common models of bounded rationality is called *Quantal Response*.

Quantal response assumes that humans will try to maximize their utility function, but will noisily estimate each strategy's expected utility [5, 10]. One approach to represent quantal response is using the logit quantal response, as shown in Equation 1, where P represents the set of all possible policies that the human can follow, $u(p)$ is the expected utility from policy $p \in P$, λ is the *rationality parameter*, and $\mathbb{P}(p_i)$ is the probability that the human would follow policy $p_i \in P$.

$$\mathbb{P}(p_i) = \frac{exp[\lambda \cdot u(p_i)]}{\sum_{p_j} exp[\lambda \cdot u(p_j)]} \tag{1}$$

When the rationality parameter $\lambda = 0$, the decision-maker acts randomly, using a uniform random choice over the possible policies regardless of their expected utility. When $\lambda \to \infty$, the quantal response function converges to a utility-maximizing policy, meaning that the human acts optimally given the utility function u.

3 Bounded Rationality in AHT

In this work, we take the quantal response model as our underlying representation of human behavior in the tool fetching domain, and we fit the rationality parameter based on the human examples gathered in previous work [18].

3.1 Policy Representation

One of the main challenges in utilizing this model, is the need to enumerate all possible policies that the human can take. Unlike single-shot games where the size of the policy set equals the size of possible actions, in a sequential decision making problem like the tool fetching domain, this set can be large or even infinite: consider a policy p_0, where a human repetitively takes one step north then one step south and never reaches a goal. As the expected utility of p_0 is zero (as the human never reaches its goal), it can be easily discarded, but any prefix of p_0 which is followed by a sequence of actions that reaches the goal is a policy with a utility higher than zero and thus should be taken into consideration.

To overcome this challenge, we use a standard Markov Decision Process (MDP) representation for the modeled agent [19]. An *MDP* is a tuple $M = \langle S, A, T, R \rangle$ such that S is a set of states of the environment; A is the set of actions that the agent can execute; $T : S \times A \times S \rightarrow [0, 1]$ is the transition function representing the probability of reaching s' when the agent is at state s and taking action a; and $R : S \rightarrow \mathbb{R}$ is the reward of the agent for reaching state s. A *policy* of an agent is $p(s, a)$ is a function that returns the probability of the agent to choose action $a \in A$ when in state $s \in S$.

With this general definition of a policy, and given the Markovian assumption of the MDP model, we can break down the quantal response model into a set of state-dependant functions, $\mathbb{P}(a, s)$, such that the policy p is now replaced with a specific action a that the human can take in a specific state s. As shown in Equation 2, each state s requires a different value of λ_s, and $\mathbb{P}$ is defined as the probability of taking an action a_i in a specific state s.

$$\mathbb{P}(a_i, s) = \frac{exp[\lambda_s \cdot u(s, a_i)]}{\sum_{a_j} exp[\lambda_s \cdot u(s, a_j)]} \tag{2}$$

3.2 Fitting the State-dependent Rationality Parameter

We can now predict the probability of a person taking action a_i in state s using Equation 2, which depends on a state-dependent rationality parameter λ_s. To learn the value of this parameter in a domain, we can fit it using samples of human behavior as observed in previous actions taken by people in state s. We start by defining a simpler measure, $R(s)$, which is the ratio of the number of times people acted optimally from state s to the number of times they visited (and thus acted from) s. This calculation is shown in Equation 3, where $Opt(a, s) = 1$ if a is an optimal action for state s, and 0 otherwise.

Algorithm 1 Rationality Parameter for State Algorithm

1: **Input**: Trajectories Ξ, States S, constant ϵ
2: $R \leftarrow []$
3: **for** $s_i \in S$ **do**
4: $\quad R[s] \leftarrow 0$
5: **for** $\xi \in \Xi$ **do**
6: $\quad s, a \leftarrow \xi$
7: $\quad$ visits(s) += 1
8: $\quad$ **if** Opt(a,s) **then**
9: $\quad\quad$ opt(s) += 1
10: **for** $s \in S$ **do**
11: $\quad$ **if** $visits(s) \neq 0$ **then**
12: $\quad\quad R[s] \leftarrow \frac{opt(s)}{visits(s)}$
13: $\quad \lambda(s) \leftarrow \frac{1}{|\log{(R(s)+\epsilon)}|} - \frac{1}{|\log{\epsilon}|}$
return R

$$R(s) = \frac{\sum_{a \in A}(Opt(a,s))}{visits(s)} \tag{3}$$

The value of R is between $[0, 1]$, and in the original quantal response formulation $\lambda \in [0, \infty)$, so we use $R(s)$ to compute λ_s as shown in Equation 4:

$$\lambda_s \propto \frac{1}{|\log{(R(s)+\epsilon)}|} - \frac{1}{|\log{\epsilon}|} \tag{4}$$

The value of ϵ is an arbitrarily small number. In our experimental results, we experimented with several values of ϵ and eventually chose the value of $\epsilon = 10^{-5}$. Given each s and a predefined ϵ, the value of λ_s can now be calculated using sampling. Algorithm 1 shows how to calculate the rationality parameter for each state in a state space S. For a state space S and a set of trajectories Ξ such that each trajectory is a sequence of states and actions $\xi = \{s_0, a_0, s_1, a_1, \ldots\}$, the algorithm first counts the number of times the agent visited a state (line 7) and the times it acted optimally (lines 8-9). The ratio between these two values gives us the rationality value for each state (line 12), as described in Equation 3. Notice that for states that were not visited in the sampled trajectories, the R value is undefined, so we set it to be 0. This R value is then used to compute λ_s for each state, according to Equation 4.

4 Empirical Results

We elicit the rationality values from Suriadinata et al. [18], using the trajectories sampled in the experiment using MTurk. The experiment was approved by the University of Texas at Austin (IRB 2020-07-0012). In this experiment, 45 participants played in 16 setups of the tool-fetching domain with varying board

sizes (ranging from a 3×5 grid to a 6×10 grid), number of goals (ranging from 2 to 4) and varying goal locations. The order in which the 16 setups were presented to each participant was randomized. Overall, the experiment consisted of $45 * 16 = 720$ independent runs.

In Section 3 we defined R as an auxiliary value for calculating λ. For simplicity, when presenting our results we will use R with a clear value interval of $[0, 1]$, however it can be converted to λ using Equation 4.

At each instance, the participant played the role of the worker, and the fetcher was an artificial agent as described in Section 2.2. The run ends when the worker reaches the correct goal and chooses the action "work". In this domain, an optimal action of the worker is an action that keeps it on a shortest possible trajectory from its current location to its goal.

For each setup, there are 45 samples. We accumulated the total number of actions and the number of optimal actions taken at each state, and we used these values to compute the rationality value R from Equation 3 for all of the states visited by a worker in all 45 runs. For states that were never visited by workers, we set the rationality value to be 0.

An example of one such setup with 4 goal stations and computed R values for all states is illustrated in Figure 3 as a heatmap. The worker's starting point is labeled using W, the toolbox is labeled as T, the fetcher is labeled as F and the four goals as labeled using G, where the goal of the worker is labeled with G^*. The fill color of each cell defines the probability to take an optimal action (lighter is closer to R value of 1). Unexplored states are the states filled in light gray color. The states with the brightest color are states in which participants always moved correctly. Some states have a distinct color compared to their environment (for example, state $(8, 4)$ with an R value of 0 meaning that every time this state was visited, the worker took a non-optimal action). These drastic changes in color usually represent cases where only one (or a few) participant visited that state and took a non-optimal action in that visit. We expect that with a larger sample size, the color transition between states would be smoother. When a state has a medium-brightness color it means that most actions were optimal and that there are more visits in order to make incorrect actions less significant to the overall color of the state.

Figure 3 is a histogram of all states across the 16 board setups, and the R value computed for each of these setups based on the participant trajectories. Across the 16 board setups of varying sizes, there are 870 distinct states in total, out of which 399 states were visited at least once. The bins of the histogram are split by skips of 0.1. This histogram shows that participants acted optimally and have an R value close to 1 in about 60% of the visited states (237 states). About 30% of the visited states have R values varying from 0.9 to 0.5, stressing the need for a bounded rationality model to capture human behavior in such domains. Lastly, in about 10% of the visited states (40 states) the rationality value is very low and has a value between 0 to 0.1. As mentioned before, in most of these cases these states were only visited once by a single participant who took a non-optimal action.

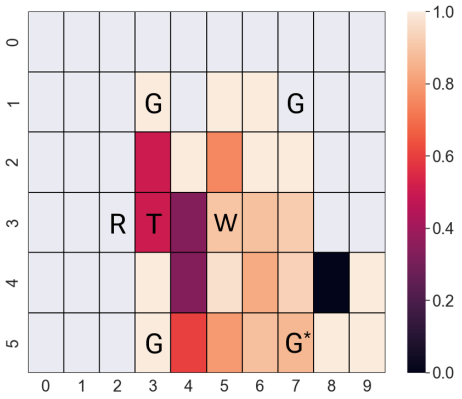

Fig. 2: A heatmap of one of the experiments, repersenting the R value of each state on the board. G cells are the goals, G^* is the true goal of the worker, R is the fetcher, W is the worker, and T is the location of the toolbox. The probability of taking a rational action is set by the state's color. Darker colors mean that there the action taken in the specific state were usually of always non-optimal. Light-grey represent states that were never visited by participants in the experiment.

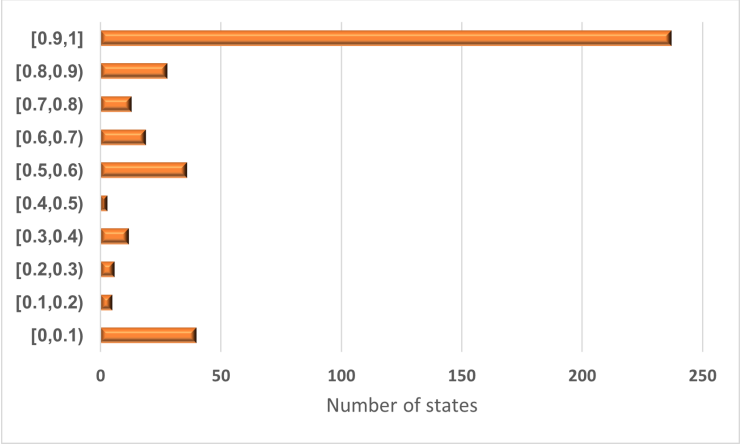

Fig. 3: A histogram of all the states in all 16 experiments, showing the number of states with an R value in each interval of size 0.1. States which were not visited are not included in the counting.

5 Conclusion

In this paper, we presented a bounded rationality model for state-dependent policies of human agents. This variation of the quantal response model can provide a prediction about the optimality of human participants, and can be incorporated

back into AHT agents that need to address the optimality of their teammates. We shows some preliminary results on quantifying the rationality parameter in the "Tool Fetching Domain", showing that the participants' rationality value varied significantly for different states.

As mentioned in the empirical section, by increasing the number of participants, we expect the rationality value of different states to smooth so it provides a closer estimate to the real rationality value for the bounded rationality model. In addition, as some states were never visited by the participants and might never get visited even with additional trials, we consider using Gaussian blur in future work to propagate a rationality value to unvisited states.

The preliminary results presented do not discuss the difference in rationality values when participants were given different instructions, but Suriadinata et al. [18] did identify a general decrease in optimality when participants were given more elaborate instructions. We expect to see similar trends when explicitly computing the rationality value, and we will examine whether this decrease in optimality is more common in specific states, for example closer to goal stations.

Finally, we intend to leverage the learned quantal response model and feed it back into the design of an AHT agent that needs to collaborate with people.

References

1. S. V. Albrecht and P. Stone. Reasoning about hypothetical agent behaviours and their parameters. In *AAMAS*, pages 547–555, 2017.
2. S. Barrett, A. Rosenfeld, S. Kraus, and P. Stone. Making friends on the fly: Cooperating with new teammates. *Artificial Intelligence*, October 2016.
3. S. Barrett, A. Rosenfeld, S. Kraus, and P. Stone. Making friends on the fly: Cooperating with new teammates. *Artificial Intelligence*, 242:132–171, 2017.
4. A. D. Dragan, K. C. Lee, and S. S. Srinivasa. Legibility and predictability of robot motion. In *2013 8th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 301–308. IEEE, 2013.
5. J. K. Goeree, C. A. Holt, and T. R. Palfrey. Quantal response equilibrium and overbidding in private-value auctions. *Journal of Economic Theory*, 104(1):247–272, 2002.
6. B. J. Grosz and S. Kraus. The evolution of Sharedplans. In *Foundations of Rational Agency*, volume 14 of *Applied Logic Series*. Springer Netherlands, 1999.
7. J. E. Just, M. R. Cornwell, and M. N. Huhns. Agents for establishing ad hoc cross-organizational teams. In *Proceedings. IEEE/WIC/ACM International Conference on Intelligent Agent Technology, 2004.(IAT 2004).*, pages 526–530. IEEE, 2004.
8. D. Kahneman. Maps of bounded rationality: Psychology for behavioral economics. *American economic review*, 93(5):1449–1475, 2003.
9. R. Kildare. Ad-hoc online teams as complex systems: agents that cater for team interaction rules. 2004.
10. R. D. McKelvey and T. R. Palfrey. Quantal response equilibria for normal form games. *Games and economic behavior*, 10(1):6–38, 1995.
11. R. Mirsky, W. Macke, A. Wang, H. Yedidsion, and P. Stone. A penny for your thoughts: The value of communication in ad hoc teamwork. *Good Systems-Published Research*, 2020.

12. N. Rabinowitz, F. Perbet, F. Song, C. Zhang, S. M. A. Eslami, and M. Botvinick. Machine theory of mind. In *International Conference on Machine Learning*, pages 4218–4227. PMLR, 2018.
13. A. Rahman, N. Höpner, F. Christianos, and S. V. Albrecht. Towards open ad hoc teamwork using graph-based policy learning. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139. PMLR, 2021.
14. M. Ravula, S. Alkoby, and P. Stone. Ad hoc teamwork with behavior switching agents. In *IJCAI*, 2019.
15. A. Rosenfeld and S. Kraus. Predicting human decision-making: From prediction to action. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 12(1):1–150, 2018.
16. H. A. Simon. Rational choice and the structure of the environment. *Psychological review*, 63(2):129, 1956.
17. P. Stone, G. A. Kaminka, S. Kraus, and J. S. Rosenschein. Ad hoc autonomous agent teams: Collaboration without pre-coordination. In *Twenty-Fourth AAAI Conference on Artificial Intelligence*, 2010.
18. J. Suriadinata, W. Macke, R. Mirsky, and P. Stone. Reasoning about human behavior in ad hoc teamwork. In *Adaptive and learning Agents Workshop at AAMAS 202*, 2021.
19. R. S. Sutton and A. G. Barto. *Reinforcement learning: An introduction*. MIT press, 2018.
20. 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*. PMLR, 2020.
21. L. Zintgraf, S. Devlin, K. Ciosek, S. Whiteson, and K. Hofmann. Deep interactive Bayesian reinforcement learning via meta-learning. *arXiv:2101.03864 [cs]*, 2021.