Effects of Learning State Dependence of Reward Weights on Trust and Team Performance in a Human-Robot Sequential Decision-Making Task

Shreyas Bhat*, Joseph B. Lyons†, Cong Shi‡ and X. Jessie Yang*
*Industrial and Operations Engineering, University of Michigan, Ann Arbor, MI
†Air Force Research Laboratory, Dayton, OH
‡Herbert Business School, University of Miami, Miami, FL

***Abstract*—In this paper, we evaluate two interaction strategies for a robot in a sequential decision-making task: one which uses a state-dependent reward function and the other that uses a state-independent (constant) reward function. Towards this, we present a study done on Amazon Mechanical Turk to learn the state-dependent reward function. Using this reward function, we compare the two strategies in simulation, where we also set the risk levels actively to induce a difference between the two strategies. Our results indicate that the interaction strategy using the state-dependent reward function results in better trust and team performance compared to that using the state-independent reward function, especially when more of the state space is explored. Thus, there may be merit in learning a more fine-grained reward function for a robot interacting with a human. The results of this study provide a starting point for a future human-subjects study.**

I. INTRODUCTION

With the advent of AI, humans and *smart* robots are increasingly teaming up to perform various tasks. Robots are starting to be seen more like teammates in a human-robot team rather than mere tools to be utilized by the human operator [1]–[3]. As in any team, trust in the robot is very important to ensure smooth teamwork and good performance of the team. Towards this, a large area of research has emerged focusing on developing dynamic models of trust [4]–[6], predicting human behavior through trust [7], [8], and using these predictions to modify robot behavior [9], [10].

Researchers have been interested in the idea of value/goal alignment which deals with aligning the values/goals of robots with that of their human counterparts [9], [11]–[13]. This is typically done by modeling the interaction as a reward maximization process and trying to learn the human's reward function through demonstrations or queries about their preferences about the task at hand. In an earlier work [8], our research team found that when a robot is starting from an uninformed prior about the human's reward function, then learning the reward function through the human's behavior is a good way to increase her trust in the robot. On the other hand, if the robot already has a good prior on the reward function of the human, further personalization through reward learning may not be so important for trust and team performance. This study tries to extend this prior work by learning a more fine-grained reward function that depends on the state of the human-robot team. Through a study conducted using Amazon Mechanical Turk (MTurk), we show the process of learning this state-dependence of the human's reward function. We then present two robot interaction strategies: one using a constant reward function set through the informed prior from the earlier study and the other using this state-dependent reward function and compare trust and team performance in simulation. We also provide a brief discussion on how to set the risk level actively in order to emphasize the differences between the two robot strategies. Our results indicate that the robot strategy using the more fine-grained state-dependent reward function results in a higher trust and team performance regardless of whether the simulated human uses the constant reward function or the state-dependent reward function. This study should be seen as a first step towards a human-subjects experiment designed to verify the simulation results.

The rest of the paper is organized as follows: Section II discusses the literature that informs our study, Section III provides the mathematical model, Section IV provides details of the MTurk study, Section V sets up the simulation study, Section VI provides major results from the simulation study, and finally, Section VII concludes the paper by discussing the implications of the results and the limitations of our study.

II. RELATED WORK

A. Trust-driven HRI

In recent years, there has been increased research interest in using trust as a guiding point to predict human behavior. Such predictions can be utilized by robots interacting with humans to change their behavior and ensure an appropriate level of trust. In one of the seminal papers, Xu and Dudek [14] presented a quantitative model of trust and used it to improve the performance of an autonomous drone through an interactive visual navigation system. The authors [15] then improved upon their model and demonstrated its use in enabling autonomous robots to actively gain trust. Floyd et al. [6] presented a framework for a robot to quickly learn trustworthy behaviors while interacting with humans by learning their preferences. Chen et al. [10], [16] modeled the interaction between a human and a robot as a Partially Observable Markov Decision Process (POMDP) with trust as

the unobservable state. They showed that a robot can show behaviors that gain human trust by solving this POMDP. Guo et al. [7], [17] used a reverse psychology human behavior model and showed that robots using such a model could intentionally deceive their human partners. Bhat et al. [18] improved this model by adding a trust-gaining reward term and showed its effectiveness at keeping the robots from engaging in deceptive behavior. In later studies [8], [9], the authors used a bounded rationality disuse model of human behavior and showed that using this model inherently nudges the robots away from deceptive behaviors even without the need for a trust-gaining reward term. Zahedi et al. [19] provided a computational model for capturing and modulating trust in iterated human-robot interactions. A robot uses this model to gain human trust using explicable actions and later maximize rewards using possibly inexplicable actions once enough trust has been built. In summary, there have been significant advances in the use of modeling techniques as applied to trust evolution in the context of HRI.

B. Value Alignment in HRI

The term "value alignment" has received a lot of attention among researchers in the field of HRI in the past few years, especially with the advancements in AI. Arnold et al. [11] considered whether pure inverse reinforcement learning (IRL) can lead to true value alignment or whether a more norm-enforced method is more appropriate for the task. Brown et al. [12] provided a simple test to check whether the values of the human and the robot are aligned. Hadfield-Menell et al. [20] proposed a Cooperative Inverse Reinforcement Learning (CIRL) framework for modeling and learning human rewards while working towards a common goal. Fisac et al. [21] and Malik et al. [22] proposed solutions to the CIRL problem that result in robot behaviors that actively teach the human about the task and effectively learn the human's preferences about how to complete the task. Further, Li et al. [23] found that adaptive policies adopted by an agent can not only result in better team performance in an interdependent Human-Autonomy Teaming context, but such strategies can achieve optimal performance faster relative to other strategies (e.g. static or random). In our earlier work [8], we used Bayesian IRL [24] to learn human's reward functions while assuming that they are independent of the state of the human-robot team. In this study, we try to extend that work by removing that assumption and actively learning the state-dependent reward functions of humans. For practical applications of HRI, this could be a critical step as goals in an HRI context may evolve over time.

III. PROBLEM FORMULATION

We consider a dyadic human-robot team in which a human soldier teams up with an intelligent drone to sequentially search N sites in a town for potential threats. At each site, the drone scans and reports a chance of threat being present inside the site. Additionally, it recommends which action to choose. The human can select one of two actions: breach the site directly or use a robotic armored rescue vehicle (RARV) for protection from threats. Breaching a site directly is faster but risky; the soldier can get harmed if a threat is encountered in this case. Using the RARV is time consuming since it takes time to deploy it, but it is risk-free since it will protect the soldier from harm in case any threat is present inside the site. We operationalize the harm and time consumption as points that get deducted from the team's score. The soldier starts with a health level H and a time level C. Each time a threat is encountered without protection, the soldier loses h points of health. Each time the RARV is deployed, the soldier loses c points of time.

The interaction between the human and the drone is modeled as a trust-aware Markov Decision Process (MDP) [18] consisting of states, actions, rewards, transition function, and a human behavior model.

A. States

The trust of the human t, the remaining health points H, and the remaining time points C as the states for the MDP.

B. Actions

The two actions available to the human are USE or NOT USE the RARV, operationalized as 1 and 0 respectively.

C. Reward function

We consider the reward function as a negative convex combination of the cost for losing health and the cost for losing time. In particular,

$$R(H,C,a,D) = -w(H,C)\mathscr{H}(a,D) - (1-w(H,C))\mathscr{C}(a). \tag{1}$$

Here, $w(H,C)$ is the reward weight associated with the cost of losing health, $\mathscr{H}(a,D)$ is the cost for losing health, $\mathscr{C}(a)$ is the cost for losing time, a is the action selected by the human, and D is a binary variable indicating the presence of threat. In particular, the functions $\mathscr{H}(a,D)$ and $\mathscr{C}(a)$ are given by:

$$\mathscr{H}(a,D) = hD(1-a), \tag{2}$$

$$\mathscr{C}(a) = ca, \tag{3}$$

where h is the cost for losing health and c is the cost for losing time. For our study, we set $h = c = 10$.

D. Transition function

We use the beta distribution trust model [4] with the reward-based performance metric [18] as the transition function for trust. Specifically, trust is modeled as a beta distributed random variable with parameters α and β which are updated based on positive and negative experiences with the recommendations.

$$t_i \sim Beta(\alpha_i, \beta_i) \text{ where} \tag{4}$$

$$\alpha_i = \alpha_{i-1} + p_i v^s, \tag{5}$$

$$\beta_i = \beta_{i-1} + (1-p_i)v^f. \tag{6}$$

Here $(\alpha_0, \beta_0, v^s, v^f)$ are the trust parameters associated with an individual, p_i is the reward-based binary performance of the recommendation at the i^{th} search site. Mathematically,

$$p_i = \begin{cases} 1, & \text{if } R(t,H,C,a_i^r,D) \geq R(t,H,C,1-a_i^r,D), \\ 0, & \text{otherwise.} \end{cases} \quad (7)$$

Here, R is the reward function (Eq. 1), a_i^r is the recommended action at the i^{th} search site.

We update the health and time according to the rules mentioned above. The soldier loses health when encountering a threat without protection from the RARV. The soldier loses time for using the RARV.

$$H_{i+1} = \begin{cases} H_i - h, & \text{if } a_i^h = 0 \text{ and } D_i = 1, \\ H_i, & \text{otherwise.} \end{cases} \quad (8)$$

$$C_{i+1} = \begin{cases} C_i - c, & \text{if } a_i^h = 0, \\ C_i, & \text{otherwise.} \end{cases} \quad (9)$$

Here, i is the site index, H_i and C_i are the health and time points remaining before the i^{th} interaction, a_i^h is the action chosen by the human at the i^{th} site, and D_i is a binary variable indicating the presence of threat at the i^{th} site.

E. Human behavior model

We use the Bounded Rationality Disuse Model of human behavior when interacting with recommender systems [8]. It states that the human chooses the recommended action with the probability equal to their level of trust on the recommendation system. With the remaining probability, the human chooses an action using the bounded rationality model, which means choosing an action with a probability proportional to the exponential of the expected reward associated with that action. Overall, for our case, the probabilities are:

$$P(a_i^h = a | a_i^r = a) = t_i + (1-t_i)p_i^a, \quad (10)$$

$$P(a_i^h = 1-a | a_i^r = a) = (1-t_i)(1-p_i^a). \quad (11)$$

Here, a_i^h is the action selected by the human, a_i^r is the action recommended by the robot, and t_i is the human's level of trust on the recommender at the i^{th} search site and p_i^a is defined as:

$$p_i^a \propto \exp\left(\kappa E[R(t_i, H_i, C_i, a, D_i)]\right), \quad (12)$$

where κ is the "rationality coefficient" of the human. Its value controls the level of randomness in the human's action choices: a higher value results in less randomness (more rationality) while a lower value results in more randomness. The robot solves the MDP using finite-horizon value iteration to generate its optimal recommendation.

IV. MTURK STUDY

This section describes the first phase of our overall study - data collection for learning the state dependence of reward weights. We used Amazon Mechanical Turk (MTurk) for collecting data. Our main insight was that assuming the bounded-rationality model of human behavior, we can find the tipping point of the threat level at which a majority of the population will change their choice of action from risk-taking to risk-averse. This threat level (d^*) can then be converted to the health reward weight using the equation below:

$$d^* = \frac{(1-w)c}{wh}. \quad (13)$$

For the derivation of the equation, we refer readers to Bhat et al. [9]. In essence, it is the threat level at which the one-step expected reward for losing health is the same as that for losing time. Therefore, the probability of choosing either action is 0.5.

So, the idea is to collect action-choice data for a set of states of (H,C) for a series of threat levels. We can then train a logistic regression model to find d^* and thus $w(H,C)$ from Eq. 1. Fig. 1 shows an example of such a query.

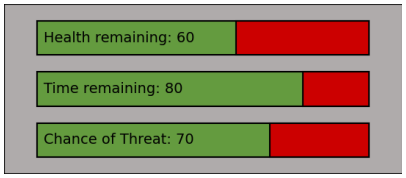


Fig. 1: A sample query asked to an MTurk worker. The worker was shown this information and asked whether s/he would choose to use the RARV or not.

A. Data Collection

We selected 6 health states, 6 time states, and 11 threat levels giving a total of 396 queries to be asked to MTurk workers. We randomly divided the queries into 12 separate surveys consisting of 33 queries each to ensure that each survey takes a reasonable amount of time to complete. Participants first provided informed consent at the beginning of the survey process, followed by watching an instructional video to learn the task, the two actions, and their consequences. At the end of the instructional video, participants were asked to choose the correct full form of RARV, failing which the participants were prohibited from continuing the survey. Those who correctly answered this question moved to the main part of the survey. Three attention-check questions were embedded in the survey to ensure data quality.

The study was approved by the Institutional Review Board at the University of Michigan (ID HUM00249731). All surveys were administered using Qualtrics. Each of the 12 surveys was completed by at least 10 MTurk workers. In total, we obtained 4092 query responses from 124 workers.

B. Data Cleaning

We realized that the data collected from MTurk workers was filled with sub-optimal action choices, possibly due to them not understanding the task, or not paying enough attention (although all the workers did pass the basic attentional check questions). To clean the data, we decided to remove the data from workers who had chosen some "obviously" sub-optimal action choices. These include two cases: (1) Using the RARV when the threat level is 0% and (2) Not using the

RARV when threat level is 100%, health is 10 and time is greater than 10. The first case is sub-optimal since the other action will definitely lead to the no health loss, no time loss outcome, while using the RARV will result in an unnecessary time loss. The second case is sub-optimal because if the RARV is not used in this case, the soldier's health level will drop to 0, ending the mission.

We identified the MTurk IDs of the workers corresponding to these sub-optimal cases and removed the corresponding data. This resulted in the removal of data from $83 \cup 10 = 85$ workers. Thus, our final dataset consisted of responses from 39 workers, corresponding to 1287 query responses.

C. Data Analysis

A logistic regression was trained using threat level as the predictor and the action choice as the target for each state (h,c) in the query set of the cleaned data. A sample logistic curve is shown in Fig. 2. The point at which the logistic curve reaches a value of 0.5 corresponds to d^* from Eq. 13. Thus, we get the raw reward weights for losing health $w(H,C)$ as a function of health remaining and time remaining at each of the queried states. This is shown as a heatmap in Fig. 3a.

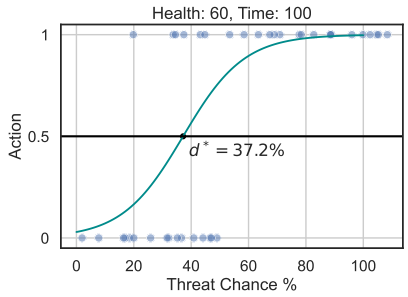


Fig. 2: Sample logistic curve learned from the collected data

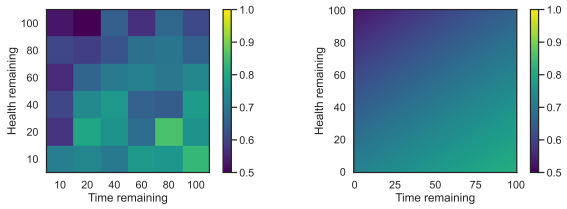
In order to get a smooth function for the health reward weight, we used the forward selection method of model selection using the Akaike Information Criterion (AIC). To do this, we incrementally added features from the set $\{H,C,H^2,C^2,HC\}$ to logistic regression models, computed the AIC, and stopped when the AIC increased. This gave us a logistic regression model with $\{H,C\}$ as the features. The final resulting model is given by,

$$w(H,C) = \frac{1}{1+\exp(0.26H - 0.17C - 0.79)}. \qquad (14)$$

A heatmap generated using this model is shown in Fig. 3b. Here, the reward weight increases with increasing value of time remaining and decreasing value of health remaining.

V. SIMULATION STUDY

We simulate interactions with two robot strategies: one using a constant reward weight of 0.81 and the other using the state-dependent reward weights 14. The value of 0.81 was chosen from an earlier study [8] that found that using this reward weight would result in similar trust to the case when the robot is actively learning reward weights. In this study we want to see if that is the case when we explore more of the state space.



(a) Raw data (b) Smoothed model

Fig. 3: Heatmaps showing (a) raw data of learned health reward weights at each queried state and (b) the smoothed function for the state dependence of reward weights

A. Simulating the human

Our simulated human sampled trust parameters taken from a dataset collected in an earlier study [18]. The simulated human thus maintained and updated her trust using the trust dynamics model (Eq. 4). After receiving a recommendation from the robot, the simulated human chooses an action using Eq. 10. We use $\kappa = 0.2$ throughout the simulation. The simulated human is assumed to be using the state-dependent reward function (Eq. 14). After observing the outcome of the action selection, the simulated human updates her trust parameters and reports the level of trust by sampling from the corresponding beta distribution.

The robot learns the trust parameters of each human using Maximum Likelihood Estimation (MLE) [8].

B. Setting threats and threat levels

We used the following strategy for setting the threats and threat levels. In general, the prior probability of threat presence was set to 0.7 meaning that there would be threats in $7/10$ search sites on average. This was to ensure that aligning the values of the robot with the human will be beneficial for trust, which is only important under high-risk scenarios [9]. Then, with 50% chance at any site independently, the threat was set randomly using a Bernoulli sample with this prior probability. To set the threat level after scanning the site by the drone, we sampled a beta distribution with a peak at 0.9 in case a threat is present and with a peak at 0.1 if a threat is not present.

With the remaining 50% chance, the threat and threat level were chosen *intelligently* to induce a difference in recommendations from the two robot strategies. As can be seen from Fig. 2, the value d^* is like a threshold threat level below which the action 0 is more likely and above which the action 1 is more likely. A similar behavior is also seen for recommendations [9]. We compute d^* for the constant reward weight and the state-dependent reward weight and set the threat level to be uniformly sampled between the two values. The threat is then sampled with a Bernoulli distribution with this threat level as the parameter. In this way, we make it more likely for the two recommendations to differ.

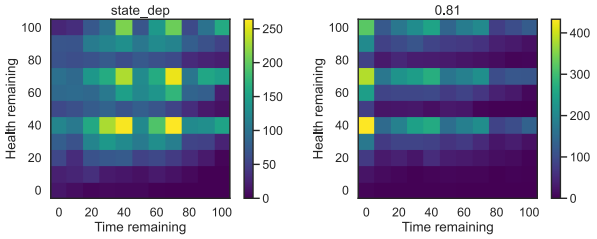
VI. RESULTS

We ran 100 independent simulations for 9 starting conditions of health and time selected from the set $\{100, 70, 40\}$. The results presented below are an average of all simulation runs. For each simulation run, the human-robot team sequentially searched through 10 search sites. In all line plots in this section, the 95% confidence interval is also plotted.

A. States Visited

Fig. 4 illustrates the frequency of visits to each state during the simulation runs. Brighter colors represent states visited more frequently, while darker colors represent those visited less often. As shown, the robot strategy with constant reward weights produces more horizontally oriented patterns of brighter colors (Fig. 4b). This pattern suggests that health remains relatively stable throughout the interaction, reflecting a more conservative approach by the team.

In contrast, the robot strategy with state-dependent rewards leads to a broader exploration of the state space, with a greater number of states shown in brighter colors 4a). This pattern indicates that the team is more willing to take risks, potentially sacrificing some health to save time.



(a) State dependent rewards (b) Constant rewards

Fig. 4: Counts of states visited for the two robot strategies

B. Trust Dynamics

Fig. 5 compares the trust reported by the simulated human for the two interaction strategies. It is clear that the state-dependent strategy of the robot results in higher trust.

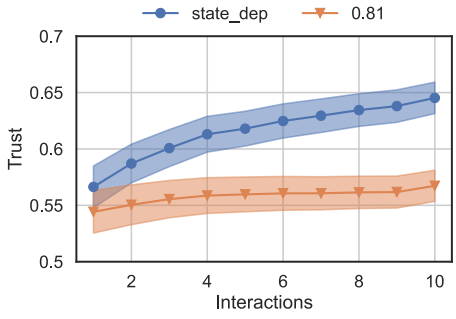


Fig. 5: Comparison of trust reported by the simulated human for the two robot strategies. Here, trust ranges from 0 to 1.

C. Performance

Fig. 6 compares the performance of the team for the two robot strategies. Performance is measured as the sum of the health points and the time points at each stage of the mission. As is evident from the graph, the state-dependent strategy performs better than the constant strategy for the robot.

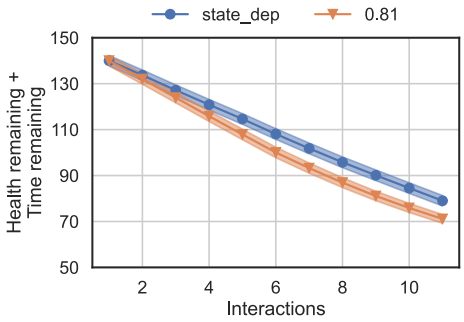


Fig. 6: Comparison of the performance of the team, as measured by the sum of the health points and the time points

Fig 7 splits out the performance into its components. As can be seen, the constant strategy prefers to save health, only losing about 10 points on average throughout the mission, while losing a lot of time. The state-dependent strategy, however, loses around 20 points of health to save time.

It should be noted that the results will remain the same as long as the human's reward weights for losing health satisfies $w > 0.5$, which is the case for the state-dependent reward function and the constant reward function. This is because the performance metric will be the same as long as this condition is satisfied. Therefore, even if the simulated human was using the constant reward function, the results still hold.

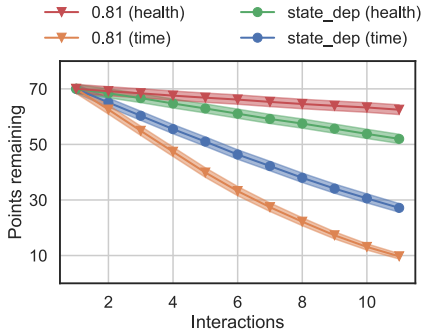


Fig. 7: Comparison of the health and time points across the interaction for the two robot strategies

VII. CONCLUSION

In this paper, we compared two interaction strategies for a robot working with a human teammate in a sequential decision-making task: one using a state-dependent reward function and the other using a state-independent (constant) reward function. Our simulation results indicate that the strategy using the state-dependent reward function results in higher reported trust and team performance compared to that using constant rewards. These improvements indicate the benefits of learning more fine-grained reward functions when interacting with humans. These results give us a starting point for a human-subjects study to verify whether the results translate to real-life.

We conducted a human-subjects study on Amazon Mechanical Turk to learn the state-dependence of the reward function. The learned reward function showed expected behavior: willing to take risks when the health is high and the time remaining is low and becoming more risk averse as the health decreases and the time remaining increases.

It will be interesting to see if humans feel more trust towards a robot that may recommend riskier actions in order to save time or if they prefer a more conservative approach with the robot using the constant rewards strategy. Another interesting thing to look into is the perceived performance of the team when interacting with the two robot strategies. In an earlier work [8], we found that some robot strategies lead to a higher perceived performance even though there was not any significant difference in objective performance of the team.

The results of this study should be viewed considering the following limitations. Firstly, using simulated humans means that their reward functions exactly match that of the robot. In reality, there could be individual differences in the reward functions of each human that the robot interacts with. Thus, the results on trust and performance could change from person-to-person. This limitation, in turn, leads to a possible direction for future research: personalization of the reward weights (Eq. 14) model during interaction, By doing so, there is potential to see improvements in trust and performance for all individuals that interact with such a "adaptive" robot. Secondly, although we sample trust parameters for the simulated human from an existing dataset, it still does not cover all possibilities. This could be addressed through a future human-subjects study.

ACKNOWLEDGEMENT

A part of this work supported by the Air Force Office of Scientific Research under grant number FA9550-23-1-0044.

REFERENCES

[1] J. B. Lyons, K. Sycara, M. Lewis, and A. Capiola, "Human–autonomy teaming: Definitions, debates, and directions," *Frontiers in Psychology*, vol. 12, 2021.

[2] T. O'Neill, N. McNeese, A. Barron, and B. Schelble, "Human–autonomy teaming: A review and analysis of the empirical literature," *Human Factors*, vol. 64, no. 5, pp. 904–938, 2022, pMID: 33092417.

[3] H. Chung, T. Holder, J. Shah, and X. J. Yang, "Developing a Team Classification Scheme for Human-Agent Teaming," *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 2024.

[4] Y. Guo and X. J. Yang, "Modeling and Predicting Trust Dynamics in Human–Robot Teaming: A Bayesian Inference Approach," *International Journal of Social Robotics*, vol. 13, no. 8, pp. 1899–1909, Dec. 2021.

[5] A. Xu and G. Dudek, "OPTIMo: Online Probabilistic Trust Inference Model for Asymmetric Human-Robot Collaborations," in *Proceedings of the Tenth Annual ACM/IEEE International Conference on Human-Robot Interaction*. Portland Oregon USA: ACM, Mar. 2015, pp. 221–228.

[6] M. W. Floyd, M. Drinkwater, and D. W. Aha, "Learning Trustworthy Behaviors Using an Inverse Trust Metric," in *Robust Intelligence and Trust in Autonomous Systems*, R. Mittu, D. Sofge, A. Wagner, and W. Lawless, Eds. Boston, MA: Springer US, 2016, pp. 33–53.

[7] Y. Guo, C. Shi, and X. J. Yang, "Reverse Psychology in Trust-Aware Human-Robot Interaction," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 4851–4858, Jul. 2021.

[8] S. Bhat, J. B. Lyons, C. Shi, and X. J. Yang, "Evaluating the Impact of Personalized Value Alignment in Human-Robot Interaction: Insights into Trust and Team Performance Outcomes," in *Proceedings of the 2024 ACM/IEEE International Conference on Human-Robot Interaction*. Boulder CO USA: ACM, Mar. 2024, pp. 32–41.

[9] ——, "Value Alignment and Trust in Human-Robot Interaction: Insights from Simulation and User Study," in *Discovering the Frontiers of Human-Robot Interaction*, R. Vinjamuri, Ed. Cham: Springer Nature Switzerland, 2024, pp. 39–63.

[10] M. Chen, S. Nikolaidis, H. Soh, D. Hsu, and S. Srinivasa, "Planning with Trust for Human-Robot Collaboration," in *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*. Chicago IL USA: ACM, Feb. 2018, pp. 307–315.

[11] T. Arnold, D. Kasenberg, and M. Scheutz, "Value alignment or misalignment - what will keep systems accountable?" in *AAAI Workshops*, 2017.

[12] D. S. Brown, J. J. Schneider, and S. Niekum, "Value alignment verification," in *International Conference on Machine Learning*, 2020.

[13] S. Milli, D. Hadfield-Menell, A. Dragan, and S. Russell, "Should robots be obedient?" in *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, ser. IJCAI'17. AAAI Press, 2017, p. 4754–4760.

[14] A. Xu and G. Dudek, "Trust-driven interactive visual navigation for autonomous robots," in *2012 IEEE International Conference on Robotics and Automation*. St Paul, MN, USA: IEEE, May 2012, pp. 3922–3929.

[15] ——, "Maintaining efficient collaboration with trust-seeking robots," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Daejeon, South Korea: IEEE, Oct. 2016, pp. 3312–3319. [Online]. Available: http://ieeexplore.ieee.org/document/7759510/

[16] M. Chen, S. Nikolaidis, H. Soh, D. Hsu, and S. Srinivasa, "Trust-aware decision making for human-robot collaboration: Model learning and planning," *J. Hum.-Robot Interact.*, vol. 9, no. 2, Jan. 2020.

[17] Y. Guo, X. J. Yang, and C. Shi, "Reward shaping for building trustworthy robots in sequential human-robot interaction," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2023, pp. 7999–8005.

[18] S. Bhat, J. B. Lyons, C. Shi, and X. J. Yang, "Clustering Trust Dynamics in a Human-Robot Sequential Decision-Making Task," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 8815–8822, Oct. 2022.

[19] Z. Zahedi, M. Verma, S. Sreedharan, and S. Kambhampati, "Trust-Aware Planning: Modeling Trust Evolution in Iterated Human-Robot Interaction," in *Proceedings of the 2023 ACM/IEEE International Conference on Human-Robot Interaction*. Stockholm Sweden: ACM, Mar. 2023, pp. 281–289.

[20] D. Hadfield-Menell, A. Dragan, P. Abbeel, and S. Russell, "Cooperative inverse reinforcement learning," in *Proceedings of the 30th International Conference on Neural Information Processing Systems*, ser. NIPS'16. Red Hook, NY, USA: Curran Associates Inc., 2016, p. 3916–3924.

[21] J. F. Fisac, M. A. Gates, J. B. Hamrick, C. Liu, D. Hadfield-Menell, M. Palaniappan, D. Malik, S. S. Sastry, T. L. Griffiths, and A. D. Dragan, "Pragmatic-pedagogic value alignment," in *Robotics Research*, N. M. Amato, G. Hager, S. Thomas, and M. Torres-Torriti, Eds. Cham: Springer International Publishing, 2020, pp. 49–57.

[22] D. Malik, M. Palaniappan, J. Fisac, D. Hadfield-Menell, S. Russell, and A. Dragan, "An efficient, generalized Bellman update for cooperative inverse reinforcement learning," 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. 3394–3402.

[23] H. Li, T. Ni, S. Agrawal, F. Jia, S. Raja, Y. Gui, D. Hughes, M. Lewis, and K. Sycara, "Individualized mutual adaptation in human-agent teams," *IEEE Transactions on Human-Machine Systems*, vol. 51, no. 6, pp. 706–714, 2021.

[24] D. Ramachandran and E. Amir, "Bayesian inverse reinforcement learning," in *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, ser. IJCAI'07. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2007, p. 2586–2591.