Deceptive Reinforcement Learning for Preserving the Privacy of Reward Functions

Yue Yang* , **Zhengshang Liu*** , **Peta Masters** , **Tim Miller**
School of Computing and Information Systems
The University of Melbourne, Parkville, 3040, Australia.
{yuey16,zhengshangl}@student.unimelb.edu.au, {peta.masters,tmiller}@unimelb.edu.au

Abstract

In this paper, we study the problem of deceptive reinforcement learning to preserve the privacy of a reward function. Reinforcement learning is the problem of finding a behaviour policy based on rewards received from exploratory behaviour. A key ingredient in reinforcement learning is a *reward function*, which determines how much reward (negative or positive) is given and when. However, in some situations, we may want to keep a reward function private; that is, to make it difficult for an observer to determine the reward function used. We define the problem of privacy-preserving reinforcement learning, and present two models for solving it. These models are based on *dissimulation* – a form of deception that 'hides the truth'. We evaluate our models both computationally and via human behavioural experiments. Results show that the resulting policies are indeed deceptive, and that participants can determine the true reward function less reliably than that of an honest agent.

1 Introduction

In this paper, we study the problem of deceptive reinforcement learning for preserving the privacy of a reward function. Reinforcement learning is a framework and collection of methods for an agent to learn a behaviour policy by interacting with the environment and responding to positive and negative rewards [Sutton and Barto, 2018]. Recent advances in deep reinforcement learning have led to a surge of interest from the research community [Mnih *et al.*, 2015]. An important part of the reinforcement learning framework is the *reward function*, which determines when and how much reward (negative or positive) is given for each possible behaviour in a system. This defines the goals of the agent.

However, there are situations in which we may want to preserve the privacy of goals. Masters and Sardina [2017b] give the example of a convoy escorting a VIP to a secret destination, with an observer that plans to deploy an assassin. We do not want an observer to be able to determine the final destination. In reinforcement learning, this corresponds to being able to determine what reward function was used to learn a policy.

Deceptive behaviour is an act or a change of current state that could provide misleading information to hide the truth [Bell, 2003]. Bell defines two general types of deception: *dissimulation*, in which someone 'hides the truth' to avoid revealing information; and *simulation*, in which someone 'shows the false' to entice someone to believe something that is not true.

Models of deceptive planning have been proposed in recent years [Masters and Sardina, 2017b; Keren *et al.*, 2016; Kulkarni *et al.*, 2018a], all based around the idea of preserving goal privacy using dissimulation. However, these are in the context of model-based planning and require reasoning about the model structure to inform the dissimulation, so are not applicable to model-free methods such as model-free reinforcement learning.

In this paper, we propose a more general model of dissimulation for preserving goal privacy that *is* applicable for model-free MDPs. We present two methods: one based on ambiguity, in which the agent selects actions that maximise the entropy from the observer's point of view; and one based on Masters and Sardina [2019]'s model for intention recognition using irrationality, in which action selection is a weighted sum of optimal honest behaviour and some 'irrational' behaviour. These methods can be implemented using pre-trained Q-functions, and require no model. Since the Q-function provides, for each state, a measure of expected future rewards, it provides a general representation of the possibilities for action selection [Sutton and Barto, 2018].

We evaluate our models by using a naïve intention recognition algorithm and via a human subject experiment with non-naïve 69 participants. The intention recognition algorithm and our participants were asked to estimate the likelihood of different destinations in a path planning simulation. The results show that our agents are effective at hiding their reward compared to an honest again, but that, like an honest agent, the true reward function becomes clearer as more actions are executed. The irrationality model deceives more than the

*These two authors contributed equally.

ambiguity model, but receives less discounted expected reward on its real reward function.

2 Background and Related Work

Deception, psychologists broadly agree, is a pejorative term for the fostering or maintainance of false belief in the minds of others [Carson, 2010]. Computer science has necessarily widened the definition, first, to accommodate mindless machines incapable of belief (as such) and second, to allow for the emerging realisation (particularly from the field of social robotics) that deception is a fundamental aspect of intelligent behaviour, frequently beneficial not only to the deceiver but to the deceived [Shim and Arkin, 2013; Wagner and Arkin, 2011].

Much work on deceptive AI focusses on aspects such as its detection [Avrahami-Zilberbrand and Kaminka, 2014], its ethical implications [Arkin *et al.*, 2012] and the qualities in a target (or mark) that make a deceptive act most likely to succeed [Ettinger and Jehiel, 2010].

The first, and to our knowledge, still the only *general* theory of deception, which focuses not on prevention but on the objective method by which deception may be achieved, comes from military strategists Bell and Whaley [1982; 2003; 1982]. They define deception non-judgementally as "the distortion of perceived reality" and maintain that it can *only* be achieved, in one of two ways: by simulation ("showing the false") or dissimulation ("hiding the true").

2.1 Deceptive Planning

In planning, deception is frequently associated with security and has lately become almost synonymous with privacy-protection [Chakraborti *et al.*, 2019]. Dissimulation in this context becomes the task of obscuring intent by maximising a plan's ambiguity [Kulkarni *et al.*, 2018b; Keren *et al.*, 2016]. The notion of obscuring intent assumes an observer engaged in intention recognition; and deceptive planning is commonly (though not exclusively)[1] conceived as an inversion of the intention recognition task [Dragan *et al.*, 2014; Keren *et al.*, 2016; Masters and Sardina, 2017b].

In this paper, we invert a type of cost-based goal recognition [Ramirez and Geffner, 2010]. To generate a probability distribution over goals, they compare each goal's *cost difference*, that is, the difference between the optimal cost of a plan via observed actions and the optimal cost of any alternative plan. The lower the cost difference, the higher the probability. Vered *et al.* [2016] take a similar approach but instead of cost difference use the ratio between the optimal cost of reaching each goal via the observations and the optimal cost per se. They propose two heuristics to minimise the computational effort in the context of online recognition, one of which suggests pruning a goal from consideration if observations deviate too far from the optimal behaviour.

Masters and Sardina [2017b] apply Bell and Whaley's theory to path-planning. They assume a naïve observer, modelled as a probabilistic goal recognition system. Observations $\vec{o}$ are passed in and a probability distribution across potential goals $P(G|\vec{o})$ is returned. They define deception at two levels of granularity: step and path. A step is deceptive if, at that step, the probability of the real goal g_r does not dominate the probability of some other goal, that is, $P(g_r|\vec{o}) \leq P(g)|\vec{o})$ for all $g \in G \setminus g_r$. They observe that *every* path must have one last deceptive point (LDP), even if it is the starting point. Masters and Sardina's approach is applicable only to model-based path planning problems, so does not generalise to MDPs.

In later work, Masters and Sardina [2019] investigate the relationship between deception and rationality. Observing that the usual definition of rationality is problematic when the ground truth is unknown, they define a rationality measure (RM) with which to compare the relative rationality of competing plans. If the plan so far is optimal for *at least one* of the possible goals, then $RM = 1$, with its value decreasing as optimality decreases with respect to *all* possible goals.

2.2 Inverse Reinforcement Learning and Imitation Learning

The idea of privacy-preserving deceptive reinforcement learning is related to *inverse reinforcement learning* [Ng and Russell, 2000] and *imitation learning* [Ziebart *et al.*, 2008]. Inverse reinforcement learning is the problem of inferring a reward function given traces of an agent's behaviour in a variety of circumstances and the sensory input to the agent. Imitation learning [Ziebart *et al.*, 2008] is similar to inverse reinforcement learning, but instead of inferring a reward function, the aim is to infer a *policy*. These methods are typically used to learn a reward function by observing e.g., a human complete the same task many times. The problem that we define in this paper could be framed as the problem of producing a policy that makes it difficult to perform inverse reinforcement or imitation learning. However, there are two key differences. First, in this paper, we aim to simply deceive for a single trace of behaviour, whereas these inverse learning problems require either a known optimal policy from which to generate traces, or a set of traces of behaviour. Despite this, there is clearly a related problem that is of interest in studying the problem of deception as obfuscating inverse reinforcement learning. Second, we define a set of possible reward functions, whereas inverse reinforcement learning starts with the set of all reward functions. While we could frame our problem in a similar way, it is uncommon for an observer not to have a model of likely goals for an actor, so we believe this is a reasonable assumption for now, but could serve as interesting future work.

3 Models

In this section, we present two models for deceptive reinforcement learning. The first is based on the ambiguous selection of actions that suggest multiple reward functions. The second is influenced by irrational behaviour:

[1] See [Kulkarni *et al.*, 2018a] for an argument against.

the policy selects an action that maximises the weighted sum between optimal behaviour and behaviour that is irrational for all possible reward functions.

3.1 Problem Formalism

We first define our base problem.

Definition 1 (Markov Decision Process (MDP) [Puterman, 2014])**.** An MDP is a tuple $\Pi = (S, A, T, r, \gamma)$, in which S is a set of states, A is a set of actions, $T(s, a, s')$ is a transition function from $S \times A \to 2^S$, which defines the probability of action a going to state s' from state s, $r(s, a, s')$ is the *reward* received for the transition from executing action a in state s and ending up in state s', and γ is the discount factor. The task is to synthesise a *policy* $\pi : S \to A$ from states to actions that maximises expected reward over trajectories in π for problem Π.

A Q-function $Q : S \times A \to \mathbb{R}$ defines the value of selecting an action a from state s, written $Q(s, a)$. A policy π is then defined as $\pi(s) = \arg\max_{a \in A} Q(s, a)$.

Definition 2 (Deceptive Reinforcement Learning)**.** A *deceptive reinforcement learning problem* is a tuple $\Pi = (S, A, T, r, \mathcal{R}, \gamma)$, in which S, A, T, r, and γ are the same as in Definition 1, and $\mathcal{R}$ is a set of possible reward functions such that $r \in \mathcal{R}$, which model the set of reward functions that an observer may believe are true. The task is to synthesise a *policy* $\pi : S \to A$ that maximises expected reward over trajectories in π while also making it difficult for an observer to determine which $r \in \mathcal{R}$ is the reward function against which π is optimised.

Clearly, this is a dual objective problem with no optimal solution: to achieve any reward, an agent must reveal part of its reward function and therefore fails to deceive; meanwhile, it is trivial to deceive by simply behaving randomly but this fails to maximise the reward.

In the remainder of this section, we present two solutions: an *ambiguity model* and an *irrationality model*.

We assume that there are pre-trained Q-functions for all reward functions in R_n. We use Q_{r_i} to represent one trained on bogus reward function r_i.

3.2 Ambiguity Model

The ambiguity model is based on the path planning framework proposed by Masters and Sardina [2017b], described in Section 2. The main conceptual idea is that the agent behaves ambiguously by selecting actions that have high Q-values not only for the real reward function but also for multiple bogus reward functions. As the trajectory progresses, fewer reward functions remain sensible, so these are pruned from consideration. Eventually, the policy selects actions only optimal for the true reward function. The final point before this occurs conforms to Masters and Sardina's *last deceptive point*.

The main idea is for our policy to generate sequences of actions that have positive reward for several reward functions, including the true reward function. For this, we need a measure of how far a sequence of observed behaviour diverges from optimal behaviour.

Observations $\vec{o}$ consist of a sequence of tuples (s, a). We measure how far observations diverge from the optimal solution by summing the Q-differences:

$$\Delta_{r_i}(\vec{o}) = \sum_{(s,a) \in \vec{o}} \left(Q_{r_i}(s, a) - \max_{a' \in A} Q_{r_i}(s, a') \right) \tag{1}$$

This formula is based on the definition of cost difference by Ramirez and Geffner [2010], and quantifies the divergence of $\vec{o}$ from the optimal solution by calculating, for each state-action pair (s, a) in $\vec{o}$, how far the action a is from the optimal choice for r_i. If $\vec{o}$ follows a sequence that is optimal for reward function r_i, then $\Delta_{r_i}(\vec{o}) = 0$, and any sub-optimal behaviour has a negative divergence.

Other definitions are possible; what matters is that Δ allows us to compare behaviour with respect to optimality. For example, we could the cost ratio rather than cost difference, as done by Vered *et al.* [2016]:

$$\Delta'_{r_i}(\vec{o}) = \sum_{(s,a) \in \vec{o}} \frac{Q_{r_i}(s, a)}{\max_{a' \in A} Q_{r_i}(s, a')} \tag{2}$$

In this case, however, if $\vec{o}$ is optimal, then $\Delta'_{r_i}(\vec{o}) = 1$, and for any sub-optimal behaviour, $0 \leq \Delta'_{r_i}(\vec{o}) < 1$. This is a property that we exploit later.

Using the definition of Δ from Equation 1, the probability that reward function r_i is the true reward function r is defined using a Boltzmann distribution:

$$P(r_i \mid \vec{o}) = \frac{exp\{\Delta_{r_i}(\vec{o})\}}{\sum_{r_j \in \mathcal{R}} exp\{\Delta_{r_j}(\vec{o})\}} \cdot P(r_i) \tag{3}$$

This implies that if $\vec{o}$ is far from optimal for r_i compared to other reward functions in $\mathcal{R}$, its probability will be lower relative to the other reward functions. This gives us a probability distribution over all reward functions in $\mathcal{R}$ outlining which reward function is most likely given the observations from an observer. The term $P(r_i)$ is the prior probability that r_i is the true reward function, which can be uniform over $\mathcal{R}$ if this is unknown.

Our model uses this probability distribution to minimise information gain by the observer using Shannon entropy[2] [Shannon, 1948] each time an action is chosen. Given a sequence of observations $\vec{o}$, our model chooses the action that minimises the information gain for the observer:

$$\pi^D(\vec{o}, s) = \arg\min_{a \in A} -\kappa \sum_{r_i \in \mathcal{R}} P(r_i \mid \vec{o} \cdot a) \times log_2(P(r_i \mid \vec{o} \cdot a)) \tag{4}$$

in which κ is a normalising term.

Thus, an agent following policy π^D will move ambiguously between all of the Q-functions to maximise entropy. However, sometimes a particular reward can become so irrational that it would be clear to an observer that this is no longer a likely reward function. We exclude such reward functions from the entropy calculation.

[2]Shannon entropy measures *information gain*. Increasing uncertainty lowers information gain and increases entropy.

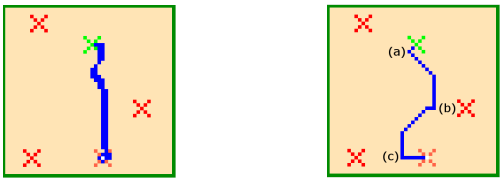
We formalise this notion by re-evaluating the bogus reward functions at each step of the plan, and excluding those would be irrational. We define the *advantage* of action a for reward function r_i as:

$$A_{r_i}(s,a) = Q_{r_i}(s,a) - R_{r_i}(\vec{o})$$

in which $R_{r_i}(\vec{o})$ is the residual expected rewards received so far in sequence $\vec{o}$, obtained by summing $Q(s',a') - Q(s,a)$ for each contiguous state-action pair in $\vec{o}$. Thus, $R_{r_i}(\vec{o})$ represents the value of having arrived at state s minus the reward of executing a, while $A_{r_i}(s,a)$ represents the advantage provided by action a in the current state s.

Intuitively, $A_{r_i}(s,a) < 0$ implies that action a is moving 'away' from the rewards given by r_i, and is therefore less rational. Note this is not the same as an *advantage function* [Baird, 1994], which measures the advantage of an action over the optimal action. We measure the advantage of taking an action relative to the value of the current state. This is based on the pruning heuristic from [Vered and Kaminka, 2017].

A reward function is pruned from the entropy calculation (set $\mathcal{R}$ in Equation 4) if $A_{r_i}(s,a) < \delta$, in which δ is a pruning parameter. If $\delta = 0$, a reward function is pruned because it offers no advantage from the current state. If $\delta < 0$, the pruning would be less aggressive, allowing some actions that do not offer an advantage. If $\delta = -\infty$, we would never prune a reward function. Each time a policy is applied, all reward functions are considered for all actions, so a pruned reward function will be re-considered at a later step. This may have the negative effect that all but the true one are pruned. In implementation, a minimum number of policies can be specified.



(a) Without pruning (b) With pruning

Figure 1: An example of the ambiguity model in path planning. The agent navigates from the green starting point to the real destination (orange), using bogus destinations (red).

Figure 1 illustrates a path planning problem in which the agent must navigate from the green start point to the orange destination. The bogus destinations are red. In Figure 1a, the agent tries to minimise information gain for all goals without pruning. It is difficult to see, but the thicker line in Figure 1a compared to Figure 1b is the agent zigzagging repeatedly left-to-right. In Figure 1b, at the first turn, labelled (a), the destination at the top left is pruned, while at turn (b), the destination on the right is pruned, and turn (c) prunes the destination at the bottom left. This delivers a shorter path because it avoids zigzagging behaviour from trying to maximise the entropy of all destinations.

3.3 Irrationality Model

The irrationality model is based on the intention recognition model in [Masters and Sardina, 2019]. In this irrationality model, the *deceptive Q-value* of an action is a weighted sum of its optimal Q-value and a *irrationality measure*. The higher the weight on the optimal Q-value, the less deceptive the behaviour.

First, we define the *irrationality measure* for an observation sequence, which is dependent on the history of a sequence of actions, rather than a single action. This is useful since an action may appear rational in a one state, but not in the context of a longer sequence.

Definition 3 (Irrationality Measure)**.** For an observed sequence of state-action pairs $\vec{o}$, the *irrationality measure* of $\vec{o}$ with respect to reward function r_i is:

$$IM(\vec{o}) = 1 - \max_{r_i \in \mathcal{R}} \Delta'_{r_i}(\vec{o}) \tag{5}$$

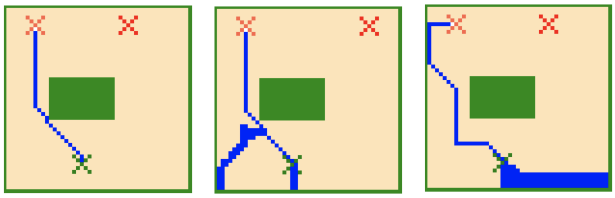
in which Δ'_{r_i} is the alternative divergence function (Equation 2). This definition is similar to the definition of rationality for path planning outlined by Masters and Sardina [2019].

Under this definition, a sequence $\vec{o}$ that has a low value for *all* reward functions has a high IM — it is irrational not to make progress towards at least one goal. We take the minimum of all reward functions: if the sequence is rational for *any* of the possible reward functions, then it is deemed rational by an observer who does not know the true reward function.

The goal of the agent is to maximise its expected reward as well as its irrationality. We achieve this with a parameter α ($0 \leq \alpha \leq 1$) as the weight factor to define the importance of the Q-value versus the irrationality. The deceptive policy π^D is defined as the weighed sum of the optimal Q-value and the irrationality measure:

$$\pi^D(\vec{o}, s) = \arg\max_{a \in A} \; (1-\alpha) \underbrace{Q'_r(s,a)}_{\text{Optimal}} + \alpha \underbrace{IM(\vec{o} \cdot (s,a))}_{\text{Irrational}} \tag{6}$$

in which $Q'_r(s,a)$ is $Q_r(s,a)$ normalised against other actions $a' \in A$ to the range [0,1]. The higher the value of α, the lower the weight given to the Q-value and the more irrational the agent's behaviour.



(a) $\alpha = 0$ (b) $\alpha = 0.15$ (c) $\alpha = 0.3$

Figure 2: An example of the irrationality model in path planning. As before the starting point is green, real destination orange and bogus destination red.

Figure 2 illustrates the irrationality model in a path planning setting. When $\alpha = 0$, we get honest behaviour.

As α increases, rationality decreases. For $\alpha = 0.15$ and 0.3, the agent moves away from both destinations. For $\alpha = 0.3$, the blue block in the bottom corner reflects the agent's excessive irrationality. As the agent moves towards its true destination, its behaviour becomes more rational, capturing a similar idea to that of the last deception point in Masters and Sardina [2017b]: it becomes more difficult to deceive as one 'approaches' a goal. To modify this, one can restrict $\vec{o}$ to the last n actions.

4 Computational Evaluation

In this section, we present a computational evaluation our two models from Section 3 using an existing framework for path planning [Masters and Sardina, 2017b]. The goals of the experiment are to quantify how deceptive our models are compared to an honest baseline, and to determine the 'cost' of deception based on how far the deceptive behaviour deviates from the honest optimal model.

4.1 Experiment Design

We implemented the ambiguity and irrationality models defined in Section 3 using the P4 framework for evaluating path planning algorithms,[3] which includes model-based planning algorithms for deceptive path planning and evaluation measures.

Independent variables We evaluated four different agents in our experiments:

1. an 'honest' agent, which uses the optimal policy for the real reward function;
2. one deceptive agent that uses the ambiguity models; and

3-4. two deceptive agents using the irrationality model, with $\alpha = 0.3$ and $\alpha = 0.5$ respectively.

Reward functions gave a reward of 10,000 for reaching the destination state. Negative rewards model actions costs, with -1 for an action up, down, left, or right, and $-\sqrt{2}$ for any diagonal move. All agents use a discounted reward of $\gamma = 1$. For the ambiguity model, we set pruning parameter $\delta = 0$. Q-functions were implemented as Q-tables and were learnt using value iteration [Sutton and Barto, 2018] until convergence.

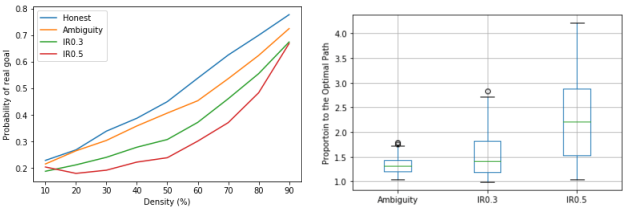
Measures We applied two measures: (1) the total path cost, which is the inverse of the discounted reward; and (2) the probability assigned to the true reward function, calculated using a naïve intention recognition algorithm for path planning [Masters and Sardina, 2017a], which applies the notion of cost difference in the strict context of path-planning.

Experiment parameters We designed five different map types for the simulator: (1) 49×49 with no obstacles, such as in Figure 1; (2) 49×49 with some large obstacles, such as in Figure 2; (3) 49×49 map with random and high density obstacles; (4) 100×100 with 'archipelago' (a small number of large island obstacles); (5) 100×100 with many rooms and corridors. For each map type, we defined eight different variations by changing the number of goals (three or five), distribution of goals, and the position of real goal, leading to a total of 40 maps.

Each model was applied to all 40 maps. For each path generated, the intention recognition measure was taken at nine 'checkpoints': every 10% along the path.

4.2 Results

Figure 3 presents an overview of the results for the computational experiments. Figure 3a, plots the average probability of the real goal at each point for the maps with five destinations. The results for maps with three destinations shows the same trend, so are omitted for brevity. In all cases, the intention recognition algorithm becomes more accurate as observation density (percentage of the path revealed) increases. However, there is a clear trend that it performs worse on the deceptive models. The irrationality models are more deceptive but this must be considered alongside path costs.



(a) Computational evaluation (five destinations)

(b) Path costs, proportional to honest path

Figure 3: Overview of computational evaluation results

Figure 3b shows the path costs for the three models, as a proportion of the length of optimal (honest) path. The ambiguity model arrives at the destination with fewer actions than either irrationality model. This is important because in addition to being deceptive, the objective of deceptive reinforcement learning is to maximise discounted expected rewards. In some cases, irrationality model with $\alpha = 0.5$ was more than four times as long. If we give higher priority to the expected reward for the real reward function, we may prefer the ambiguity model or to use the irrationality model with a lower value of α.

5 Human Behavioural Evaluation

In this section, we describe a human behavioural experiment undertaken to measure the ability to deceive people, rather than algorithms. In the computational experiment, the intention recognition algorithm used is 'naïve', in that it assumes that the agent it is observing is behaving honestly and rationally, and therefore, does not consider deception.

5.1 Experiment Design

The experiment design was similar to that used for the computational evaluation, with two exceptions: (1) in-

[3] https://bitbucket.org/ssardina-research/p4-simulator/

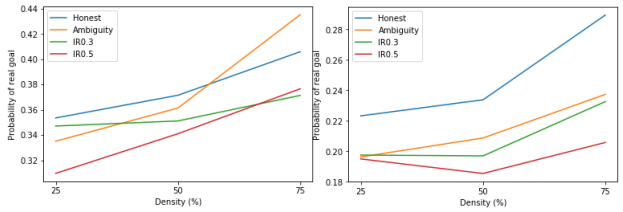
stead of the intention recognition algorithm, we ask human participants to estimate the goal distribution; and (2) the human participants were provided with only a random selection of the maps and methods. Participants were 'aware'; that is, they were explicitly instructed before the experiment that the agent may sometimes try to hide its true destination, but sometimes not.

Our experiment consisted of 480 individual stimuli: 40 maps $\times$ four possible models producing behaviour = 160 map-path pairs. We generated checkpoints at 25%, 50%, and 75%, leading to 480 samples. Each participant was tasked with studying 38 randomly selected stimuli.

We recruited 69 participants via Amazon Mechanical Turk, a crowd-sourcing platform that is commonly used for obtaining data for human-subject experiments [Buhrmester *et al.*, 2018]. Participants were paid US$4 for completing all tasks, which took on average 11.5 minutes. Participants were aged 20-55 ($\mu = 32$), and of the 69 participants, 15 were female, 54 were male, and none chose to specify their gender manually. Ethics approval was obtained prior to the experiment from the [Institute name omitted to preserve anonymity], with ethics approval number [Omitted].

5.2 Results

Figure 3 presents an overview of the results for the human subject evaluation. We see similar results to that of the naïve intention recognition algorithm, except that the human subjects were overall less accurate than the naïve model, even for honest behaviour. This is understandable as the optimal behaviour is straightforward for an algorithm to calculate, but less so for a human. At the first checkpoint, by which point participants have seen 25% of the path, the accuracy is close to random.



(a) Human subject evaluation (three destinations)

(b) Human subject evaluation (five destinations)

Figure 4: Results for the naïve intention recognition algorithms and human subject estimates against all four models.

Interestingly, for the ambiguity model in maps with three goals, the human subjects were more accurate than for the honest model, but this is mostly accounted for by noisy data – the difference is less than 3%.

Overall, we see that our models are effective at deceiving the human observers for the path planning task, but the effectiveness is is only in the region of 6-10% at all stages of the path.

5.3 Limitations

There are several limitations with our study. First, while path planning is a good application for human behavioural experiments (people are good at intention recognition for path planning due to strong spatial reasoning), it is only one domain and further experimentation on different types of domains is necessary, as well as different assumptions; e.g., partially-observable environments, multiple rewards available. Second, the naïve intention recognition model we used to evaluate deception in the computational evaluation is not as sophisticated as our model of deception; that is, it does not mitigate for the fact that it is being deceived. This is difficult to mitigate because we need a level of separation between the methods and the evaluation metrics, and the only suitable model of which we are aware is the irrationality model [Masters and Sardina, 2019], on which our model is based. Third, there was only minimal incentive for our experimental participants, which is not reflective of some applications where failing to identify deception can have devastating outcomes.

6 Discussion and Future Work

In this paper, we presented two models for preserving the privacy of reward functions in reinforcement learning. The first model is based on acting ambiguously by taking actions that work towards rewards from multiple reward functions, while the second is based on selecting semi-rational actions, where part of the selection of an action weights actions that are not rational for any of the reward functions. Through computational and human evaluation in a path planning task, we have shown that the models can deceive both naïve intention recognition algorithms and human subjects.

In future work, we will apply this model to tasks other than path planning. Further, we will investigate this model in policy-based reinforcement learning, in which we do not have Q-functions, but learn a policy directly.

References

[Arkin *et al.*, 2012] Ronald C. Arkin, Patrick Ulam, and Alan R. Wagner. Moral decision making in autonomous systems: enforcement, moral emotions, dignity, trust, and deception. 100(3):571–589, 2012.

[Avrahami-Zilberbrand and Kaminka, 2014] Dorit Avrahami-Zilberbrand and Gal A Kaminka. Keyhole adversarial plan recognition for recognition of suspicious and anomalous behavior. In *AAAI Workshop on Plan, Activity, and Intent Recognition*, pages 87–121, 2014.

[Baird, 1994] Leemon C Baird. Reinforcement learning in continuous time: Advantage updating. In *Proceedings of 1994 IEEE International Conference on Neural Networks*, pages 2448–2453. IEEE, 1994.

[Bell and Whaley, 1982] J Bowyer Bell and Barton Whaley. *Cheating: Deception in War & Magic, Games & Sports*. St Martin's Press, 1982.

[Bell, 2003] J. Bowyer Bell. Toward a theory of deception. *International Journal of Intelligence and Counterintelligence*, 16(2):244–279, 2003.

[Buhrmester *et al.*, 2018] Michael D Buhrmester, Sanaz Talaifar, and Samuel D Gosling. An evaluation of amazon's mechanical turk, its rapid rise, and its effective use. *Perspectives on Psychological Science*, 13(2):149–154, 2018.

[Carson, 2010] Thomas L Carson. *Lying and deception: Theory and practice*. Oxford University Press, 2010.

[Chakraborti *et al.*, 2019] Tathagata Chakraborti, Anagha Kulkarni, Sarath Sreedharan, David E Smith, and Subbarao Kambhampati. Explicability? legibility? predictability? transparency? privacy? security? the emerging landscape of interpretable agent behavior. In *ICAPS*, volume 29, pages 86–96, 2019.

[Dragan *et al.*, 2014] Anca D Dragan, Rachel M Holladay, and Siddhartha S Srinivasa. An analysis of deceptive robot motion. In *Robotics: science and systems*, page 10, 2014.

[Ettinger and Jehiel, 2010] David Ettinger and Philippe Jehiel. A theory of deception. *American Economic Journal: Microeconomics*, 2(1):1–20, 2010.

[Keren *et al.*, 2016] Sarah Keren, Avigdor Gal, and Erez Karpas. Privacy preserving plans in partially observable environments. In *Proceedings of IJCAI'16*, pages 3170–3176, 2016.

[Kulkarni *et al.*, 2018a] Anagha Kulkarni, Matthew Klenk, Shantanu Rane, and Hamed Soroush. Resource bounded secure goal obfuscation. In *AAAI Fall Symposium on Integrating Planning, Diagnosis and Causal Reasoning*, 2018.

[Kulkarni *et al.*, 2018b] Anagha Kulkarni, Siddharth Srivastava, and Subbarao Kambhampati. A unified framework for planning in adversarial and cooperative environments. In *ICAPS Workshop on Planning and Robotics*, 2018.

[Masters and Sardina, 2017a] Peta Masters and Sebastian Sardina. Cost-based goal recognition for path-planning. In *AAMAS*, pages 750–758. IFAAMAS, 2017.

[Masters and Sardina, 2017b] Peta Masters and Sebastian Sardina. Deceptive path-planning. In *Proceedings of IJCAI'17*, pages 4368–4375, 2017.

[Masters and Sardina, 2019] Peta Masters and Sebastian Sardina. Goal recognition for rational and irrational agents. In *Proceedings of AAMAS'19*, pages 440–448. IFAAMAS, 2019.

[Mnih *et al.*, 2015] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529, 2015.

[Ng and Russell, 2000] Andrew Y Ng and Stuart J Russell. Algorithms for inverse reinforcement learning. In *ICML*, volume 1, page 2, 2000.

[Puterman, 2014] Martin L Puterman. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

[Ramirez and Geffner, 2010] Miquel Ramirez and Hector Geffner. Probabilistic plan recognition using off-the-shelf classical planners. In *Proceedings of AAAI'10*, pages 1121–1126, 2010.

[Shannon, 1948] Claude Elwood Shannon. A mathematical theory of communication. *Bell system technical journal*, 27(3):379–423, 1948.

[Shim and Arkin, 2013] Jaeeun Shim and Ronald C. Arkin. A taxonomy of robot deception and its benefits in HRI. In *IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 2328–2335, 2013.

[Sutton and Barto, 2018] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

[Vered and Kaminka, 2017] Mor Vered and Gal A Kaminka. Heuristic online goal recognition in continuous domains. In *IJCAI*, pages 4447–4454. AAAI Press, 2017.

[Vered *et al.*, 2016] Mor Vered, Gal A. Kaminka, and Sivan Biham. Online goal recognition through mirroring: Humans and agents. In *Conference on Advances in Cognitive Systems*, 2016.

[Wagner and Arkin, 2011] Alan R Wagner and Ronald C Arkin. Acting deceptively: Providing robots with the capacity for deception. *International Journal of Social Robotics*, 3(1):5–26, 2011.

[Whaley, 1982] Barton Whaley. Toward a general theory of deception. *The Journal of Strategic Studies*, 5(1):178–192, 1982.

[Ziebart *et al.*, 2008] Brian D Ziebart, Andrew Maas, J Andrew Bagnell, and Anind K Dey. Maximum entropy inverse reinforcement learning. In *Proceedings of AAAI'08*, volume 3, pages 1433–1438, 2008.