Deceptive Reinforcement Learning by Dealing with Multiple Rewards with Single Q-Table

Yingnan Shi 1025903, Junchao Wang 1032268, Qingfeng Xu 967175

Supervisor: Tim Miller

Credit Points: 25

Type of project: Research Project

Subject Code: COMP90055

I certify that

- this thesis does not incorporate without acknowledgement any material previously submitted for a degree or diploma in any university; and that to the best of my knowledge and belief it does not contain any material previously published or written by another person where due reference is not made in the text.

- where necessary I have received clearance for this research from the University's Ethics Committee and have submitted all required data to the School

- the thesis is 4842 words in length (excluding text in images, table, bibliographies and appendices)

Deceptive Reinforcement Learning by Dealing with Multiple Rewards with Single Q-Table

Yingnan Shi 1025903, Junchao Wang 1032268, Qingfeng Xu 967175

Supervisor: Tim Miller

Abstract

In this paper, we research the problem of deceptive reinforcement learning by dealing with multiple rewards with a single Q-Table. One of the key characters of reinforcement learning is interacting with the reward from the environment. These rewards are present as a reward function in the reinforcement learning model. Deceptive Reinforcement Learning can be used to keep reward functions private by showing the fake path to the observer. However, all of the reward function information is still required during the whole learning process and existing methods have to train multiple Q-tables for corresponding goals(Yang, et al., 2020). Therefore, we define the problem of dealing with multiple rewards with a single Q-Table, and design two models which have unique characters to solve it. Both models are based on reinforcement learning and dissimulation (A type for deceptive that 'hides the truth'(Whaley, 1982)(Bell, 2003)). Two models are evaluated and compared with other types of models. The result proves that our models show a deceptive policy comparing with an honest agent and less time complexity and information leak comparing with other deceptive reinforcement learning.

1 Introduction

Deception is induced by a deceiver that designs and projects a misperception with real representation (Whaley, 1982). Nowadays, deception is widely used as a tool in national defense (Masters and Sardina, 2017), games (Anderson et al., 2018), cybersecurity (Huang and Zhu, 2019) and etc. Deceptive path planning is one of the applications of deception. For example, there are several potential targets and our force in going to raid one of them, which is called real goal, and the other goals are considered as bogus ones. The enemy is able to observe our army by satellite and we want them to realize the real target as late as possible. This can be perfectly represented and solved by reinforcement learning (RL).

Reinforcement learning is a form of machine learning that emphasizes how computer agents act based on the environment they are in to maximize the expected rewards they gain from their actions. Q-learning is the RL technique adopted in this study that by iteratively updating the future rewards of each state, a Q table can be learned. Q table is a simple lookup table, namely, it records the maximum cumulative future rewards of each state that can guide the agents to the best actions.

There are two mainstream types of methods in deceptive path planning. The first one is a Model-Based Deceptive Pathing. This method requires a way to calculate an optimal last deceptive point (LDP) (Masters and Sardina, 2017). LDP is a point that any sub-sequential actions after that point are truthful, thus not deceptive. However, this method is less general since the model requires a model of the problem, whereas model-free RL can be used with or without a model

Another type is called Model-Free Method (Sutton and Barto, 2018). It only relies on real samples from the environment and never uses generated predictions of next state and next reward to alter behavior (although they might sample from experience memory, which is close to be a model). Namely, we expect the model to learn by itself, therefore reinforcement learning is used.

In this paper, two models are presented: one is based on an inadmissible heuristic to the real goal as a distraction, and another is similar to the ambiguity model (Yang et al., 2020) that choosing the

actions that maximize the entropy from the observer's angle of vision. But our model only uses a one-q table. Since Yang's model needs to train a Q-table for each real goal and bogus goals, it may fall into the curse of time complexity when there are too many goals, our model rescues it from it, with a little trade-off of deception ability. Furthermore, since only one table is trained, the total amount of information is reduced, thus less risk of information leak.

There are three evaluation metrics used. The proportion of exposed paths, probability of the real goal in the observer's point of view, and cost of the pathing. The results show that comparing to ambiguity model(Yang et al., 2020) , the heuristic model has the lowest cost but most exposed paths and highest probability of the real goals, and the entropy model has very close performance in these three metrics even the time of training is n times less than the ambiguity model where n is the number of goals.

2 Background

Deceptive path planning is an intersection of two disciplines within Computer Science: path-planning and goal recognition (Masters and Sardina, 2017). There are many algorithms solving path-planning problems, for example, A-star search (Hart et al., 1968) can guarantee the optimality of the solution, but rather totally truthful in the view of an intelligent observer. An observer predicts agents' actions by modeling the agents in some way (Banerjee and Peng, 2003).

As an essential indicator of intelligence (Alloway et al., 2015), deception, is the key part of deceptive path planning that keeps the goal not recognizable as long as possible during the pathing to against observers.

From the perspective of psychologists, there are two types of deceptions, dissimulation (hiding the real) and simulation (showing the false) (Whaley, 1982). In the scenario of this paper, we focus on dissimulation, that we are trying to preserve real goal privacy by hiding the reward function of reinforcement learning, in two ways, making inadmissible heuristics and maximizing entropy from the observer's angle of vision. Also, reinforcement learning with multiple shared rewards is a hot topic. Douglas promotes an integrated interaction model which deals with multiple reward function and goals property (Gouglas and Guisi, 2016).

2.1 Inadmissible heuristics

The idea of making inadmissible heuristics is from the paper named deceptive game of Anderson et al., they proposed that making a reward structure that offers the agent a suboptimal policy can be seen as a deceptive strategy. Thus, making an inadmissible heuristic, for example, the inverse distance from an optimal policy, would result in deception. There are three categories of deception in their games to trap AI agents, greedy trap, smoothness trap, and generality trap. The idea of a greedy trap contributes the most inspiration to the heuristic model in our study. The way to do it is designing a game with small rewards that attract the agents into actions that makes future larger reward unattainable. For example, in Super Mario, if a very large reward of taking coins is given to the AI agents during training, it is very likely that the agents are going to run out of time by collecting all possible coins, instead of moving to the destination. This kind of deception strategy is a simulation, if this idea is applied to reinforcement learning, in which the reward function is giving extra rewards to the bogus goals, the agents would result in two sequences of actions. The first sequence happens when the extra rewards to the bogus goal are relatively lower, the result would be exactly the same as the solution of A star search, which is the truthful optimal path. The second sequence would lead the agent to the trap of fake goals when the rewards given to the fake goals are relatively higher, namely, the agent would never be able to go to the real goal. Therefore, in later studies, we would alter this idea of a greedy trap into dissimulation.

2.2 Ambiguity Model

In the paper of Yang et al., they designed two models, both of them are dissimulations. One is to maximize entropy in the observer's views, which is called ambiguity model, and another is to make

irrational choices. Both of the models acquire q-tables for each fake goal and the real goal.

The ambiguity model chooses the action by comparing all q-tables. It chooses the actions that have high Q-values for both real reward function and bogus reward functions, which means the highest entropy is left to the observer so that the information gain they can get from calculating the probability of each reward functions is minimized. When a reward function is very unlikely to be the real goal, it is then pruned during the calculation.

One of the advantages of our models upon Yang et al.'s models is that there is only 1 q table against multiple q-tables. The number of q-tables (say N) in ambiguity model and irrational model is equivalent to the number of bogus goals plus one and the one stands for the real goal. In reality, sometime N can be very large, for example, in cybersecurity, the number of routers can be very large. Another benefit is that since only 1 q table is trained, the total information amount is reduced, which brings lower probability of information leak.

2.3 Deceptive Planning

Masters and Sardina (2017) formalize deceptive path-planning and proposed the concept of last deceptive point (LDP), after which the path of an agent is found to be optimal thus truthful. The last deceptive point (LDP) is measured by density (proportion of path completion), and this can be used to rank the agents by their deception ability. For every path, there must be one and only one last deceptive point (LDP), where it can be the starting point if the path is completely truthful, or somewhere near to the real goal if the path is deceptive.

They also designed an observer that calculates the probability of all potential goals $P(G|\vec{o})$. If an action is deceptive, then $P(g_r|o) \leq P(g|o)$ for all $g \in G \setminus g_r$ and g_r denote the real goal.

3 Models

The model can be described as defining a reinforcement learning path planning problem *P,* a root agent *A* knows the all goals, map and reward functions of problem *P* and a deceptive path planer agent *A'.* The observers is *O* .The root agent *A* generates a Q-Table *T* to provide for deceptive path planer *A', A'*design a deceptive path based on T and show the path observer *O*, which makes the real goal hard to be discriminated.

3.1 Entropy Method

The Entropy model is based on the ambiguity model proposed by Yang et al.(2020). The main conceptual idea is selecting the most ambiguous action at each state while keep moving towards the real goal. However, instead of training multiple Q-tables against multiple reward functions, only one Q-table is generated by these reward functions. As a model free environment, the agent learns its action at each state and the action it takes might make some fake goals insensible. Therefore, these fake goals are pruned from the possible goal set G as the trajectory exploring. When G is empty besides from the real goal, a naïve path-planning strategy is implemented according to the high Q-values for the real goal which is optimal in the cost of heading to the real goal. The ambiguity is measured by entropies which are calculated by the corresponding Q-values at a state s for all the remained goals in G.
First of all, the Q-table is pretrained. For the action a taking the agent from the state s to state s', its Q-value is updated with the equation below:

$$Q(s,a) = Q(s,a) + \alpha[r + \gamma max_{a'} Q\left(s', a'\right) - Q(s,a)] \quad [1]$$

Where α is the learning rate and γ is the discount. We receive r if it is the reward for any of the reward functions in the MDP, or r will be the negative cost of the action a.
The Euclidean distance is introduced as a heuristic function for the fake goal pruning process. The intuition is, for an observer, if the agent is moving away from a certain goal, it might not be the real destination. Therefore, the observation is a tuple

(s, a) where s is the current state and a is the action that the agent takes. $D(s, g_i)$ represents the Euclidean distance between s and $g_i \in G$ where G is the remained goal set. For an action a taking the agent from state s to state s':

$$Disdiff(g_i|(s,a)) = D(s,g_i) - D(s',g_i) \quad [2]$$

If $Disdiff(g_i|(s,a)) < 0$, g_i will be pruned from G once the action a is taken. However, if $Disdiff(g_i|(s,a)) > 0$, g_i will meet the requirement of fake goal reconsideration process which takes the goal back to G.

At each state s, Q-values indicate the value of taking an action. We select the Q-value in the direction $dir(s, g_i)$ as $Q(s, g_i)$. The $dir(s, g_i)$ is calculated by (as showed in Fig1, assuming each state has 8 possible actions):

$$x_{diff} = x_{gi} - x_s \quad y_{diff} = y_{gi} - y_s$$

$$dir_x = \begin{cases} 1, if\, x_{diff} > 0 \\ 0, if\, x_{diff} = 0 \\ -1, if\, x_{diff} < 0 \end{cases}, \quad [3]$$

$$dir_y = \begin{cases} 1, if\, y_{diff} > 0 \\ 0, if\, y_{diff} = 0 \\ -1, if\, y_{diff} < 0 \end{cases}$$

$$dir(s, g_i) = (d_x, d_y)$$

Where coordinates for s and g_i are (x_s, y_s) and (x_{gi}, y_{gi}) respectively.

For example, the agent is at the state s_1 shown in Fig1 and the coordinate is (18,10). One of our goals g_1 is at (8,20). Therefore, $dir(s_1, g_1) = (-1,1)$ and $Q(s_1, g_1) = 87.686$.

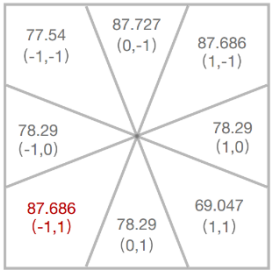


Figure 1 Example state of a Q-table

For each possible action $a_i \in A$ taking the agent from S to s', all reward functions are considered for this observation (s,a_i), then a temporary fake goal pruning process is taken for the observation. with remained goals at state s', calculate the probability of heading to each $g_i \in G$ using the formula (3) below. We repeat the same process for each possible action $a_i \in A$:

$$P(g_i|(s,a)) = \frac{Q(s\prime,g_i)}{\sum_{j=1}^{n} Q(s\prime,g_j)} \quad [4]$$

In which n is total number of remained goals in G for the observation (s, a).

This gives us probability distributions of heading to each remained reward function for each possible action at the current state s.

Our model selects the most ambiguous action which has the most uncertainty for an observer. Demonstrated by Shannon in 1948, the higher entropy the action has, the higher uncertainty it possesses. Therefore, calculate the entropy for each $a_i \in A$ with the probability distributions mentioned above, the action with the highest entropy is selected as the next move a_{actual}:

$$a_{actual} = argmax_{a \in A}(\sum_{i=1}^{n} -P(g_i|(s,a) * log_2(P(g_i|(s,a))) \quad [5]$$

Where n is total number of remained goals in G for the observation (s, a).

After a_{actual} is taken, we execute a formal fake goal pruning process for this observation (s,a_i). After that, a fake goal reconsideration process is also implemented to take back the goals pruned at early steps to the remained goal set G. However, this reconsideration step might result in that the G might never be empty. Therefore, we specify that if the pruned fake goal is the last fake goal in G, the reconsideration step will be stopped.

When all fake goals are pruned from G and there is only the real goal remained in G, the current node n_{LDP} is demonstrated as the last deceptive point by Masters and Sardina (2017). Then, the agent executes a naïve path-planning approach

and selects the highest Q-value for the real goal. In this way, the path from n_{LDP} to the real goal g_{real} is optimal in cost but with zero capacity of deception.

The pruning process is essential in this model for two reasons: firstly, it ensures that the agent keeps moving forward to the real goal and finally reaches it; secondly, it prevents the agent lingering repeatedly among all fake goals and tries to be as deceptive as it can. Demonstrated by Fig2(a), the agent pruned the top-left fake goal at point *a*, pruned the fake goal on the right-hand side at point *b* and pruned the last fake goal at point *c*. However, in Fig2(b), none of three fake goals are pruned during the path exploring, the agent keeps moving left and right because it always tries to be ambiguous for all three fake goals and the real goal. In general, reducing the repeated zigzag results in a lower path cost.

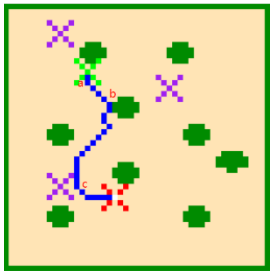


Figure 2(a) With Pruning

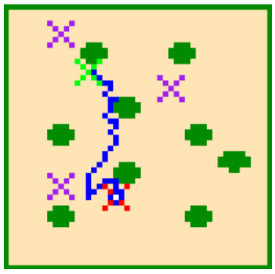


Figure 2(b) Without Pruning

Fig 2 An example of the entropy model in path planning. The agent navigates from the green starting point to the real destination (red), using bogus destinations (purple)

3.2 Heuristic Model

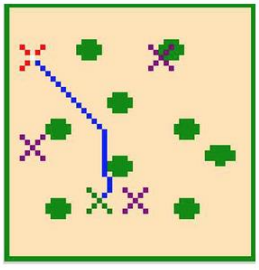
Heuristic Model is based on the assumption that making an inadmissible heuristic would result in deception (Anderson et al., 2018). One of the ways to do it is designing a game with small rewards that attract the agents into actions that makes future larger reward unattainable. If we plug this idea into reinforcement learning, then each bogus goal and real goal has a reward function and their set is denoted as *R,* we set the $r = r_t + \sum_i r_i$ where *r* is the reward function adopted in Q learning, r_t is the reward function for the true goal, and $r_i \in R$ is the reward function of each goal. That would either have no influence to the pathing comparing to *A star* if the reward for true reward is overwhelming, or the agent would be stuck to a bogus reward and never be able to reach the true goal. Instead of doing this, in order to guarantee the agent is deceptive and the pathing is converging finally, the extra rewards to the bogus goals are formulated to reward shaping as a heuristic, so the policy $\Pi = (S,A,T,r,F,\gamma)$ and $F = (s, s')$. When updating q table:

$$Q(s,a) = Q(s,a) + \alpha[r + F(s,s') + \gamma max_{a'} Q(s',a') - Q(s,a)] \quad [6]$$

There can be many ways to define *F*, but to make it more general, in this paper, *F* is defined by formula:

$$F = \sum_i D_i \quad [7]$$

where D_i is the Euclidean distance to the corresponding bogus goal. For specific environments, weights can be considered for each bogus goals when summing the distances, so that the more weighted goal would be more attractive to the agent, but that is less general and need specific strategy analysis from the user. By formulate heuristic to reward shaping, the agent can be guaranteed to be both deceptive and real goal realizable.



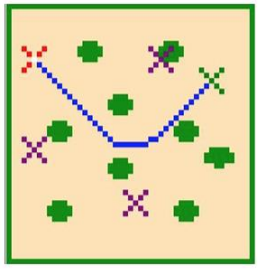


Figure 3: An example of heuristic model with different starting points (green crosses). The purple crosses are bogus goals, the red crosses are the real goals. A remarkable feature of heuristic model is that the agent would try to be as close as possible to all the bogus goals until it reaches the LDP, and the LDP has similar distances to each bogus goal due to the reward shaping function.

4. Computational Evaluation

In this section, we use a related path planning framework (Masters and Sardina, 2017) to test and evaluate our two models introduced in section 3. There are two goals in this experiment: firstly, test the models' capacity of deception by comparing them with the honest baseline model and one of the previous works; secondly, measure the path cost of models by comparing them with the honest model which is optimal in the path cost. Both items are key indexes for a successful deceptive path planning model.

4.1 Experiment Design

In Section 3, we introduced two models: Heuristic model and Entropy model. To evaluate the path planning algorithms and visualize the performances, both models are implemented with the P4 framework. At the same time, the ambiguity model (Yang, el al., 2020) and Astar model are implemented as control groups.

Independent variables

The details of the four models implemented in the experiment are:

1. Dummy Model, which implements the Astar algorithm to find a policy in a given problem. This agent can be treated as an honest model with the lowest path cost and zero capacity of deception.
2. Original Model, which is a successful deceptive path planning model by using reinforcement learning. However, multiple Q-Tables are trained to deal with multiple reward functions.
3. Heuristic Model, as introduced in section 3.1.
4. Entropy Model, as introduced in section 3.2.

Measurements

There are three measurement metrics introduced in the experiment:

(1) The proportion of exposed paths against the density. This represents that: at each density, the proportion of agents who have already exposed their real goal to the observer. For a model, the proportion is calculated by:

$$Proportion\ of\ exposed\ paths = \frac{Number\ of\ agents\ who\ have\ exposed\ the\ real\ goal}{Total\ number\ of\ agents\ experimented}$$

(2) The probability of the real goal against the density. This measures that: at each density, how likely is the real goal to be the real destination from an observer's view. This measurement is based on the naïve intention recognition algorithm by implementing the notion of cost difference in the path-planning problem (Kulkarni et al., 2019).

Both (1) and (2) are used to measure the capacity of deception of models.

(3) The cost ratio. It is the average path cost over the dummy model which is optimal in the path cost aspect. We desire an excellent deceptive model at lower cost.

One more thing to mention is 'density'. The node at density = x%, means the node at position = (x%*length (Path)). For example: for a path with length = 200, an agent at density = 15% means it is at the 30^{th} node of this path. During this experiment, density can also be treated as the percentage of the path exposed to observers. It ranges from 0 to 100% and means the agent has reached the destination when density = 100%. Usually, the higher density that the model performs deceptively, the more deceptive it is.

Experiment Parameters

We test our models on three different maps:

(1) The map without any obstacles
(2) The map with three large obstacles
(3) The map with some random but smaller obstacles

All three maps are in the same size (49*49) and have one real goal with a number of fake goals. However, 11 different locations of goals were set for each map. Therefore, there are totally 33 different experiment configurations for each model. For Measurement (1) and Measurement (2), we record the results at every 10% density. For Measurement (3), only the average path cost ratios were recorded.

4.2 Result

The Figure 1 shows the proportion of exposed paths as the y-axis and the x-axis is density. In general, the lower the y value is, the better capacity of deception. For this metric, it is obvious that the Original model performs the best, then the Entropy model, next is the Heuristic model. There are three conclusions acquired from Figure 4: (1)

Even though the performance of the Entropy model and Heuristic model are not as good as the Original model, they still perform much better than the Dummy agent, which proves that they are deceptive but in different degrees. (2) For both Heuristic and Entropy model, they perform well at early stages of their paths. As showed in Fig4, the difference between the Entropy model and the Original model is smaller than 0.1 when density<50%.

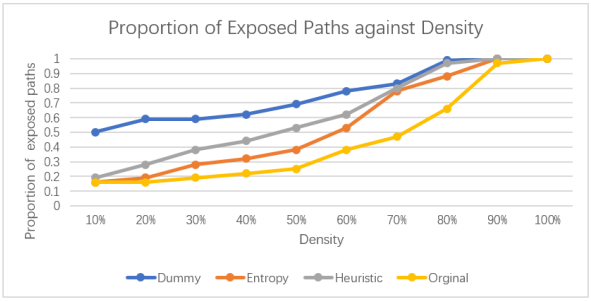


Figure 4 Proportion of Exposed Paths against Density

The Figure 5 presents the average probability of the real goal as the y-axis against the density as the x-axis. The smaller the y value is, the better capacity of deception is. There is an important check point on y-axis = 0.5. Since when the probability of the real goal is greater than 0.5, it means the it is greater than the probabilities of any other fake goals. It makes no difference to an observer between the probabilities of the real goal as long as they are greater than 0.5 and the observer will always correctly figure out the real destination.
By analyzing the result under y=0.5, the Entropy model performs the best, then the Original model and Heuristic model, which is a similar conclusion as the proportion of exposed paths against the density metric. However, in this evaluation metric, the Heuristic model and Entropy model show more competitive performance and the later one is even the most deceptive one. The reason why the results from Measurement (1) and Measurement (2) are not consistent with each other is that: at each density, there is a smaller number of agents exposed from the Original model. However, the variance of the probabilities of the real goal from the Original model is larger than the one of the Entropy model, which results in a larger average value of the probability of the real goal.
One limitation of the metric in Fig5 is that the higher probability of the real goal does not always mean the higher chance of exposing the real goal to the observer. For example, there are three goals: g1 is the real goal, g2 and g3 are bogus ones. Two probability distributions for (g1, g2, g3) are (0.4,0.5,0.1) and (0.35,0.33,0.32). Even though 0.4>0.35, the agent with the second probability distribution where probability of g1 is smaller exposes its real goal to the observer. Therefore, this metric is just one aspect to evaluate the model's capacity of deception.

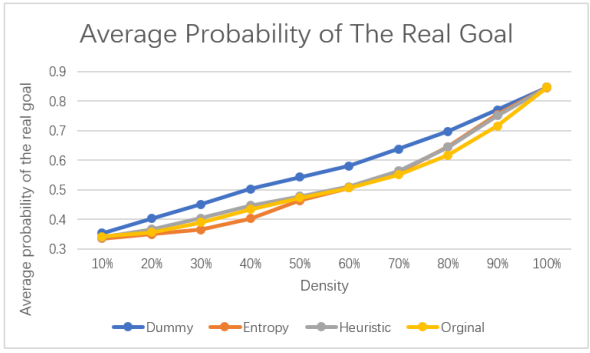


Figure 5 Average Probability of the Real Goal

The capacity of the deception is not the only aspect in our problem since a very skewed path which lingers among fake goals can be very deceptive but very cost expensive as well. Therefore, the third measurement reveals the cost of each model. Since the Dummy model is based on the Astar path planning algorithm, it leads to be optimal in cost. In the Figure 6, the y-axis Cost ratio of each model is calculated by comparing their average path cost with the average cost of the Dummy model. In our experiment, the Entropy model and the Original model share similar costs while the cost of Heuristic model is smaller than these two models. This demonstrates that even though the Heuristic model shows less deception, it can be implemented in the situation where lower cost required.

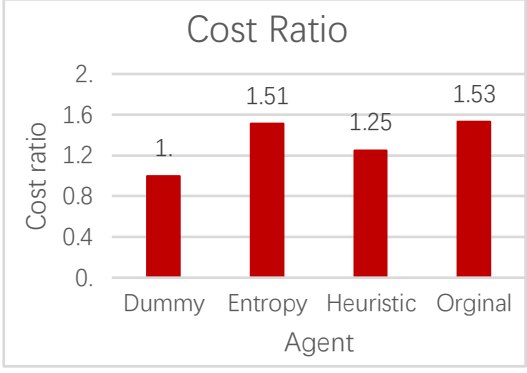


Figure 6 Cost Ratio

Overall, the capacity of deception rank for four models is (from the worst to the best):

Dummy < Heuristic < Entropy ≈ Original

The rank for the path cost (from the worst to the best):

Entropy ≈ Original < Heuristic < Dummy

5. Discussion and Future Work

In this paper, we presented two models for hiding the real reward function in reinforcement learning from an outside observer. The Heuristic model is based on the reward shaping of Q-learning (Vered and Kaminka, 2017), while the Entropy model finds the most ambiguous action through comparing the entropy of each state. Different from previous reinforcement deceptive path-planning methods, only one integrated Q-table is trained against multiple reward functions, which leads to the advantages of less training time and less information leak. However, a limitation of Entropy model is that Euclidean distance is implemented for the fake goal pruning process and entropy model still performance various on different maps. Therefore, in future work, we will apply pruning in value iteration process rather than path exploring. Also generalizing model for heuristic model to perform stably in different environments. Besides, Turing test (Turing, 1950) involving human evaluation will also be introduced for the model evaluation section as a complement of our computational experiment.

Acknowledgement

We thank the reviewers and their comments, especially our supervisor Tim Miller who has been guiding us with patience for the whole semester. We also thank our families and friends who give us inspiration and spiritual motivation.

Reference

Alloway, T.P., McCallum, F., Alloway, R.G. and Hoicka, E., 2015. Liar, liar, working memory on fire: Investigating the role of working memory in childhood verbal deception. Journal of experimental child psychology, 137, pp.30-38.

Alan, M., Turing. 1950. Computing machinery and intelligence. Mind, 59(236), pp.433-433.

Anderson, D., Stephenson, M., Togelius, J., Salge, C., Levine, J. and Renz, J., 2018, April. Deceptive games. In International Conference on the Applications of Evolutionary Computation (pp. 376-391). Springer, Cham.

Banerjee, B. and Peng, J., 2003. Countering deception in multiagent reinforcement learning. In Proceedings of the Workshop on Trust, Privacy, Deception and Fraud in Agent Societies at AAMAS-03, Melbourne, Australia (pp. 1-5).

Bell, J.B., 2003. Toward a theory of deception. International journal of intelligence and counterintelligence, 16(2), pp.244-279.

Whaley, B., 1982. Toward a general theory of deception. The Journal of Strategic Studies, 5(1), pp.178-192.

Guisi, D.M., Ribeiro, R., Teixeira, M., Borges, A.P. and Enembreck, F., 2016, June. Reinforcement Learning with Multiple Shared Rewards. In ICCS (pp. 855-864).

Huang, Y. and Zhu, Q., 2019, October. Deceptive reinforcement learning under adversarial manipulations on cost signals. In International Conference on Decision and Game Theory for Security (pp. 217-237). Springer, Cham.

Kulkarni, A., Srivastava, S. and Kambhampati, S., 2019, July. A unified framework for planning in adversarial and cooperative environments. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 33, pp. 2479-2487).

Masters, P. and Sardina, S., 2017, May. Cost-based goal recognition for path-planning. In Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems (pp. 750-758).

Masters, P. and Sardina, S., 2017, August. Deceptive Path-Planning. In IJCAI (pp. 4368-4375).
Shannon, C.E., 1948. A mathematical theory of communication. Bell system technical journal, 27(3), pp.379-423.

Sutton, R.S. and Barto, A.G., 2018. Reinforcement learning: An introduction. MIT press.

Vered, M. and Kaminka, G.A., 2017. Heuristic online goal recognition in continuous domains. arXiv preprint arXiv:1709.09839.

Yang, Y., Liu, Z., Masters, P., Miller, P., 2020, Deceptive Reinforcement Learning for Preserving the Privacy of Reward Functions. Retrived from: https://piazza.com/ class_profile/get_resource/k440kkcj3o9516/k74k4sal74n4fw