2019

Helping an Agent Reach a Different Goal by Action Transfer in Reinforcement Learning

Yuchen Wang
University of Wollongong, yw808@uowmail.edu.au

Fenghui Ren
University of Wollongong, fren@uow.edu.au

Minjie Zhang
University of Wollongong, minjie@uow.edu.au

Helping an Agent Reach a Different Goal by Action Transfer in Reinforcement Learning

Abstract

© 2019, Springer Nature Switzerland AG. Reinforcement learning agents can be helped by the knowledge transferred from experienced agents. This paper studies the problem of how an experienced agent helps another agent learn when they have different learning goals by action transfer. This problem is motivated by the widely existing situations where agents have different learning goals and only action transfer is available to agents. To tackle the problem, we propose an approach to facilitate the transfer of actions that are right to a learning agent's goal. Experimental results show the effectiveness of the proposed approach in transferring right actions to an agent and helping the agent learn to reach a different goal.

Disciplines

Engineering | Science and Technology Studies

Publication Details

Wang, Y., Ren, F. & Zhang, M. (2019). Helping an Agent Reach a Different Goal by Action Transfer in Reinforcement Learning. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 11919 LNAI 15-27.

Helping an Agent Reach a Different Goal by Action Transfer in Reinforcement Learning

Yuchen Wang, Fenghui Ren, and Minjie Zhang

School of Computing and Information Technology,
University of Wollongong, Wollongong, NSW, 2522, Australia
yw808@uowmail.edu.au, {fren, minjie}@uow.edu.au

Abstract. Reinforcement learning agents can be helped by the knowledge transferred from experienced agents. This paper studies the problem of how an experienced agent helps another agent learn when they have different learning goals by action transfer. This problem is motivated by the widely existing situations where agents have different learning goals and only action transfer is available to agents. To tackle the problem, we propose an approach to facilitate the transfer of actions that are right to a learning agent's goal. Experimental results show the effectiveness of the proposed approach in transferring right actions to an agent and helping the agent learn to reach a different goal.

Keywords: Different Goals, Action Transfer, Reinforcement Learning

1 Introduction

Reinforcement Learning (RL) has been widely used for an autonomous agent to learn to reach its goal in sequential decision-making tasks [7]. An RL agent might need a long learning time. To improve learning, transferring knowledge from experienced agents to learning agents has been widely studied [9].

Transferring different kinds of knowledge has various requirements for agents. This paper considers the transfer of *actions*, which requires agents to only have a common action set. Compared with transferring other kinds of knowledge, the requirement for action transfer is considered to be minimal [10]. This provides much flexibility. For example, agents giving and receiving actions could use different knowledge representations and learning algorithms.

In this paper, we study the problem of how an experienced agent helps another agent learn when they have different learning goals by action transfer. This problem would widely exist in the real world. For example, Alice knows how to reach her travel destination. When Bob loses his way, Alice might help Bob reach his travel destination efficiently. However, the destinations of Alice and Bob might be different. Also, Bob might not understand Alice's detailed expressions due to various reasons. In this situation, an understandable way for Alice to help is to point out some directions that Bob could follow. Here "point out" indicates "transfer", and "directions to follow" indicates "actions".

Several knowledge transfer approaches have been proposed to help an agent learn in the different-goal situation [4, 6, 12]. These approaches require learning agents to access and understand the knowledge of source agents. However, this requirement might not be satisfied in many applications, especially when humans are helping or learning [10]. Some knowledge transfer approaches with only action transfer have been proposed [1, 3, 9, 10, 13, 15]. However, these approaches require agents giving and receiving actions to have the same learning goal, which is not satisfied in the different-goal situation. Therefore, how an experienced agent helps another agent learn when they have different learning goals by action transfer remains as a challenging problem.

To tackle this problem, we ask below questions: (Q1) what actions are right to be transferred to help a learning agent in the different-goal situation? (Q2) do right actions exist? (Q3) if right actions exist, how an experienced agent finds them? and (Q4) if a right action exists, but an experienced agent cannot decide the rightness of this right action, could the agent still be able to transfer this action? Hereafter, action transfer are called *action advice*, agents giving/receiving advice are called *teachers/students*. These names often appear in the action transfer literature. We propose an action advice approach to answer the above questions. For (Q1), we define an agent's goal, describe what makes the different-goal situation, and define a teacher's right/wrong advice (Section 2). For (Q2), we define the concept of policy-similar states, at which right advice exists (Section 3.1). For (Q3), we propose a method that enables a teacher to decide if a state is policy-similar by finding right advice (Section 3.3). For (Q4), we propose a method that enables a teacher to give right advice at states which are policy-similar, but could not be decided as policy-similar by the teacher (Section 3.4). Experimental results show the effectiveness of the proposed action advice (action transfer) approach used in the different-goal situation.

2 Problem Formulation

In this section, we first give the background, including Markov Decision Process (MDP) and action advice framework. Then, we formulate this paper's problem. **Background** RL has been widely used to solve sequential decision-making tasks. Markov decision process [5] has been widely used as the model of an RL task. An MDP is described by a tuple $< S, A, T, R >$, where S is the set of states, A is the set of actions, $T : S \times A \times S \rightarrow [0, 1]$ is the transition function, $R : S \times A \rightarrow \mathbb{R}$ is the reward function. An agent needs to learn an optimal policy π, which is a mapping from S to A. Following π maximises the expected reward: $V(s) = E[\sum_{t=0}^{\infty} \gamma^t r^t | s_0 = s]$, $\forall s \in S$, where $\gamma \in [0, 1)$ is a discount factor, r^t is the reward at time step t, V is the expected reward value function.

The action advice framework [8] includes two types of agents: Teacher and Student. A teacher has learned an optimal policy π[1]. When a student is learning, the teacher could help the student learn by giving advice. The advice at a state

[1] We follow a general setting where π is optimal. Considering sub-optimal π is not the main issue in this paper, and would be left as future work.

s is an action $a \in \pi(s)$, i.e., an optimal action to take at s based on the optimal policy learned by the teacher.

Formulation of the Problem We first define the goal of an agent. Then, we describe what makes the different-goal situation, and clarify why current action advice approaches are not applicable in the different-goal situation.

Definition 1 (Agent Goal). *Given an agent in an MDP with a state space S, let V^* be the maximised expected reward value function, the goal of the agent is a state $g \in S$ where $V^*(g) \geq V^*(s), \forall s \in S$*[2]*.*

An agent receives the maximum expected reward among all states when the agent reaches its goal. The optimal policy learned by the agent guides the agent to its goal from other states.

Let t and u be a teacher and a student, g_t and g_u be their goals. The different-goal situation can be denoted as $g_t \neq g_u$. Basically, $g_t \neq g_u$ means that a teacher and a student need to solve different MDPs. Two MDPs are different when they have difference in any of S, A, T or R. In this paper, we focus on a specific kind of difference that makes $g_t \neq g_u$: *two different MDPs share the same S, A, T, R^-, and have different R^+*, where R^+: $S \times A \to \mathbb{R}_{>0}$, R^-: $S \times A \to \mathbb{R}_{<0}$. The MDPs with this kind of difference could model a bunch of different, but similar tasks in the real world. For example, different navigation tasks on land share the same S (land space), A (actions available on land), T (execution results of actions), R^- (e.g., battery consumption), and have different R^+ (different navigation goals). Enabling agents in different navigation tasks to advise each other would be beneficial to these agents.

Let π_g be the optimal policy for reaching a goal g. A teacher has learned π_{g_t}. A student has not learned π_{g_u}, and would need action advice from the teacher to learn π_{g_u}. To help the student learn, the advised actions should be optimal for reaching g_u. The optimal/non-optimal advised actions can be defined as follows:

Definition 2 (Right/Wrong Advice). *Let g_t and g_u be the goals of a teacher and a student, π_g be the optimal policy for reaching a goal g. At a state s, an advised action $a \in \pi_{g_t}(s)$ is right/wrong advice when $a \in \pi_{g_u}(s)/a \notin \pi_{g_u}(s)$.*

In the same-goal situation, $g_t = g_u$. Then, $\pi_{g_t} = \pi_{g_u}$, which means $\forall s \in S, \pi_{g_t}(s) = \pi_{g_u}(s)$. Hence, we have $\forall s \in S, \forall a \in \pi_{g_t}(s), a \in \pi_{g_u}(s)$. This clarifies that in the same-goal situation, at any state, any advised action (optimal to g_t) is right advice (optimal to g_u). However, in the different-goal situation, $g_t \neq g_u$. Then, $\pi_{g_t} \neq \pi_{g_u}$, which means $\exists s \in S$, $a \in \pi_{g_t}(s) \wedge a \notin \pi_{g_u}(s)$. This indicates that at some states, some advised actions might be wrong advice. In current action advice approaches, a teacher does not decide if its advice is right to a student. Then, the teacher might give wrong advice, which would mislead the student. Hence, this paper's problem is to study how a teacher gives right advice to a student in the different-goal situation. The notation used in this paper is shown in Table 1.

[2] There are multiple goals when multiple states have the same maximum V value. The technical details for multi-goal and one-goal situations are generally the same. We only describe the one-goal situation for clear description.

Table 1: Notation

Notation	Meaning
S, s, g	a state space, a state, a goal state of an agent
π_g, $\pi_g(s)$	an optimal policy for g, optimal actions to take at s for reaching g
$PS(g_1, g_2)$	policy-similar states of two agents with goals g_1 and g_2
$OR^{\pi_g}(s, a)$	optimally reachable states of a state s and an action a under a policy π_g

3 Action Advice in the Different-Goal Situation

This section first summarises two aims of the proposed approach by defining agents' policy-similar states. Then, the proposed approach is described in detail.

3.1 Policy-Similar States and Aims of the Proposed Approach

For an agent with a goal g, g indicates a unique π_g [7]. Hence, for a teacher and a student with g_t and g_u be their goals, π_{g_t} and π_{g_u} are well-defined. Based on π_{g_t}, π_{g_u} and Definition 2, the states where right advice exists are also well-defined. Those states can be defined as follows:

Definition 3 (Policy-Similar States). *For a state space S, let g_t and g_u be the goals of a teacher and a student respectively, π_g be the optimal policy for reaching a goal g, the policy-similar states of the agents are a set of states:*

$$PS(g_t, g_u) = \{s \in S \backslash \{g_u\} | (\exists a)[a \in \pi_{g_t}(s) \wedge a \in \pi_{g_u}(s)]\} \quad (1)$$

The term *policy-similar* describes that a teacher and a student can take at least one same optimal action to reach the agents' different goals. At a policy-similar state, right advice could be given if the teacher knows which action (indicated by the teacher's policy) is optimal to the student's goal. Note that PS is not known by any agent because an agent only knows its own goal and policy. PS is computed within $S \backslash \{g_u\}$. g_u is excluded because when at g_u, a student has reached its goal and does not need to take actions or get advice. Based on PS, we summarise the aims of the proposed approach as follows:
Aim 1: To give right advice at a state s, the first aim is *to enable a teacher to decide if s is in PS, i.e., to decide if* $\exists a[a \in \pi_{g_t}(s) \wedge a \in \pi_{g_u}(s)]$.
Aim 2: If a teacher could decide that $\forall s \in PS, \exists a[a \in \pi_{g_t}(s) \wedge a \in \pi_{g_u}(s)]$, the teacher could give right advice at maximum number of states. However, the above decision would be hard to made. This is because finding all PS would require the knowledge of both π_{g_t} and π_{g_u}, which would be infeasible for a teacher who only knows π_{g_t}. Let PS_d be the states that could be decided in PS by a teacher. We expect $PS_d \subset PS$, i.e., there would be some states $PS \backslash PS_d$ that are policy-similar, but could not be decided as policy-similar by the teacher. To give right advice at more states than just at PS_d, the second aim is *to enable a teacher to give right advice at states $PS \backslash PS_d$*.

The number of policy-similar states would relate to the settings of agents' goals. This will be experimentally investigated in Section 4.1.

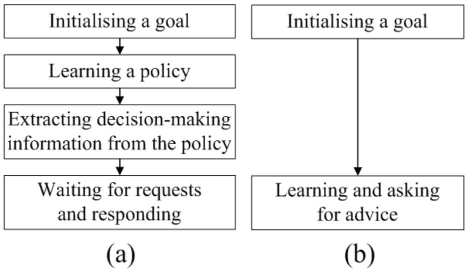

Initialising a goal
Learning a policy
Extracting decision-making information from the policy
Waiting for requests and responding
(a)
Initialising a goal
Learning and asking for advice
(b)

Fig. 1: The procedures of (a) a teacher and (b) a student.

3.2 Overview of the Proposed Action Advice Approach

To tackle the aims summarised in Section 3.1, we propose an action advice approach whose overview is shown in Fig. 1. Fig. 1(a) shows the procedure of a teacher. After initialising a goal, the teacher learns an optimal policy for reaching its goal. Then, from the policy, the teacher extracts the decision-making information used for deciding if an action is optimal to a student's goal (for Aim 1, described in Section 3.3). Next, the teacher starts to wait for requests from a student and will respond by giving or not giving advice. In Fig. 1(b), a student first initialises its goal. Then, the student starts to learn and will ask the teacher for advice (for Aim 2, described in Section 3.4).

3.3 Formulation and Extraction of Decision-Making Information

We first formulate the decision-making information used for deciding if an action is optimal to a student's goal. Then, we show the extraction of this decision-making information from the policy learned by a teacher.

Formulation of Decision-Making Information For a teacher and a student, from the teacher's perspective, any state $g_p \in S$ might be the student's goal, and the student may ask for advice to reach g_p from another state s. Hence, the teacher needs to decide if $\pi_{g_t}(s)$ provides optimal actions for reaching g_p from s. The decision-making information can be formulated in below definition:

Definition 4 (Optimally Reachable States). *Let π_g be the optimal policy for reaching a goal g, g_t be the goal of a teacher. For a state space S, a state s, and an action $a \in \pi_{g_t}(s)$, the optimally reachable states of (s,a) under π_{g_t} are a set of states:*

$$OR^{\pi_{g_t}}(s,a) = \{g_p \in S | a \in \pi_{g_t}(s) \wedge a \in \pi_{g_p}(s)\} \tag{2}$$

$\forall g_p \in OR^{\pi_{g_t}}(s,a)$, a is an optimal action for reaching both g_t and g_p. When a student is learning and asking for advice to reach g_u at s, the teacher can make its decision on whether to give advice based on following rules:

$$Decide(s, g_u) = \begin{cases} \text{give action advice } a, & \text{if } \exists a \in \pi_{g_t}(s)[g_u \in OR^{\pi_{g_t}}(s,a)] \\ \text{no advice}, & \text{otherwise} \end{cases} \tag{3}$$

If $\exists a \in \pi_{g_t}(s)[g_u \in OR^{\pi_{g_t}}(s,a)]$, a is decided to be optimal to the student's goal, and will be given as right advice by the teacher. Otherwise, the teacher cannot find right advice, and hence does not give advice.

Extraction of Decision-Making Information Next, we introduce the extraction of $OR^{\pi_{g_t}}$ from a teacher's policy π_{g_t}. To simplify notation, we use π to denote π_{g_t}, OR^{π} to denote $OR^{\pi_{g_t}}$.

We first use below equation to get an optimally reachable state of (s,a):

$$O(s,a) = \{s | s \in \hat{T}(s,a) \backslash \{s\} \wedge |\hat{T}(s,a) \backslash \{s\}| = 1\} \tag{4}$$

where $\hat{T}(s,a)$ indicates the states that an agent may travel to after taking a at s. $|\hat{T}(s,a)\backslash\{s\}| = 1$ means that the agent will travel to only one state other than the current state s. We use o_{sa} to denote the only state in $O(s,a)$. a is optimal for reaching o_{sa} from s. This is because the teacher has learned that o_{sa} is the state to reach before the teacher can optimally reach g_t. If there is another action $a_b \notin \pi(s)$ that could make the teacher better reach o_{sa}, the teacher would have learned that $a_b \in \pi(s)$, which contradicts with $a_b \notin \pi(s)$. When $|\hat{T}(s,a)\backslash\{s\}| > 1$, a might not be optimal for reaching $\hat{T}(s,a)\backslash\{s\}$. Detailed analysis on the optimality of a when $|\hat{T}(s,a)\backslash\{s\}| > 1$ is beyond the scope of this paper, and would be studied in future work.

For o_{sa}, we can apply Equation (4) to get $O(o_{sa},a'), a' \in \pi(o_{sa})$. We use $o_{saa'}$ to denote the only state in $O(o_{sa},a')$. As a is an optimal action to reach o_{sa} from s, a' is an optimal action to reach $o_{saa'}$ from o_{sa}, we have that a is an optimal action to reach $o_{saa'}$ from s because "is an optimal action to reach" is a transitive relation. Hence, $o_{saa'}$ is also an optimally reachable state of (s,a). Following the above analysis, we can get a sequential sets of optimally reachable states. To do so, we introduce the below equation:

$$N^{\pi}(S') = \{o_{s'a'} | \forall s' \in S', \forall a' \in \pi(s')\} \tag{5}$$

$N^{\pi}(\{o_{sa}\})$ indicates the optimally reachable states to reach by taking every action in $\pi(o_{sa})$. $N^{\pi}(\cdot)$ can be regarded as a function, and can be applied to the returned states of $N^{\pi}(\{o_{sa}\})$. We use $N^{\pi}_k(\{o_{sa}\})$ to denote repeatedly applying $N^{\pi}(\cdot)$ for k times from $\{o_{sa}\}$. Based on the transitive relation, the states in $N^{\pi}_k(\{o_{sa}\})$ are optimally reachable states of (s,a). Hence, we have:

$$OR^{\pi}(s,a) = \{o_{sa}\} \cup N^{\pi}_1(\{o_{sa}\}) \cup \cdots \cup N^{\pi}_k(\{o_{sa}\}) \cup \cdots \tag{6}$$

As we also have $N^{\pi}_k(\{o_{sa}\}) = N^{\pi}_{k-1}(N^{\pi}_1(\{o_{sa}\})) = N^{\pi}_{k-1}(\bigcup_{a'}\{o_{saa'}\})$ where $a' \in \pi(o_{sa})$, Equation (6) can be written in a recursive form:

$$\begin{aligned} OR^{\pi}(s,a) &= \{o_{sa}\} \cup \bigcup_{a'}\{o_{saa'}\} \cup \cdots \cup N^{\pi}_{k-1}(\bigcup_{a'}\{o_{saa'}\}) \cup \cdots \\ &= \{o_{sa}\} \cup \bigcup_{a'}\left[\{o_{saa'}\} \cup \cdots \cup N^{\pi}_{k-1}(\{o_{saa'}\}) \cup \cdots\right] \\ &= \{o_{sa}\} \cup \bigcup_{a'} OR^{\pi}(o_{sa},a') \end{aligned} \tag{7}$$

A teacher can use Equation (7) to extract OR^{π} after the learning of π.

3.4 Learning and Asking Process

In the different-goal situation, there should be a way to let a teacher know which state a student wants to reach. We enable the student to send state signals to the teacher. However, the number of state signals that can be sent at a state is limited by a transmission capacity c. We consider that utilising c would help to achieve Aim 2 (see Section 3.1). To do so, we first define an agent's *sub-goals*:

Definition 5 (Sub-Goal). *Given an agent in an MDP with a state space S, let V be the agent's experted reward value function, a state $s \in S$ is a sub-goal of the agent when $V(s) > \tau$, where τ is a threshold.*

A sub-goal indicates certain amount of expected reward ($> \tau$), and could be regarded as "close" to g_u. Reaching states with higher V value means that the student would be "closer" to g_u. The optimal actions for reaching g_u and sub-goals might be the same. When the student asks to reach g_u at a policy-similar state s, but the teacher does not know s is policy-similar, the student could utilise the transmission capacity c (if $c > 1$) by asking to reach sub-goals. If the teacher knows optimal actions to reach the sub-goals, those optimal actions might be right advice to the student. Even if the given advice were wrong, this would not badly hurt the student's learning because at least the wrong advice leads the student to states "close" to g_u. Based on the above analysis, we propose a learning and asking process of a student shown in Algorithm 1.

Algorithm 1: Learning and Asking Process of a Student

```
   Input: state space $S$, transmission capacity $c$, sub-goal threshold $\tau$.
1  Initialises $C(s) \leftarrow 0$, $N(s) \leftarrow \emptyset$, $A(s) \leftarrow \emptyset$, $\forall s \in S$; /* $C(s)$: number of times the
     student has asked for advice at a state $s$, $N(s)$: sub-goals to which no advice
     has been received at $s$, $A(s)$: advice that has been received at $s$ */
2  foreach episode do
3      repeat
4          $s \leftarrow$ Observes the current state;
5          if $A(s) \neq \emptyset$ then $a_{adv} \leftarrow A(s)$ and go to Line 14;
6          while $C(s) < c$ do
7              $g_{sub} \leftarrow \arg\max_s V(s), s \in S \land s \notin N(s) \land V(s) > \tau$;
8              if $g_{sub} \neq \emptyset$ then
9                  $a_{adv} \leftarrow Ask(s, g_{sub})$; $C(s) \leftarrow C(s) + 1$;
10                 if  $a_{adv} = \emptyset$ then
11                     $N(s) \leftarrow N(s) \cup \{g_{sub}\}$;
12                 else
13                     $A(s) \leftarrow a_{adv}$;
14         Takes $a_{adv}$ if $a_{adv} \neq \emptyset$. Otherwise, use $\epsilon$-greedy to select an action to
             take. Then, updates learning information, and updates $s$ to next state;
15     until s is the student's goal;
```

4 Experiments

In this section, we first present experimental settings. To set up agents' goals, we investigate the influence of specific goals settings on the number of policy-similar states. Then, we conduct two experiments to evaluate the proposed approach.

4.1 Experimental Settings

Domain The current action advice approaches are applied in domains with the same-goal situation [1, 3, 10, 15]. For example, [10] uses Mountain Car and Pac-Man. In Mountain Car, agents' goal is to reach the top of a mountain. In Pac-Man, agents' goal is to earn points while avoiding being caught. These domains are not suitable for evaluating approaches in the different-goal situation.

In this paper, the experiments are conducted in a grid-world domain (shown in Fig. 2(a)). Grid-world domains have been used in various RL problems [4, 7, 14]. The state space can be represented by a set of locations. An agent's goal is a specific target location that the agent learns to reach. When agents have different target locations, the agents are said to have different goals.

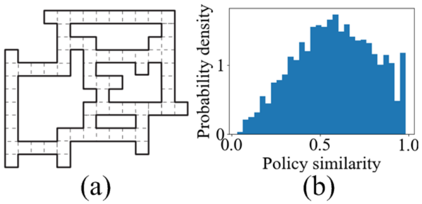


Fig. 2: (a) The navigation map, (b) policy similarity distribution.

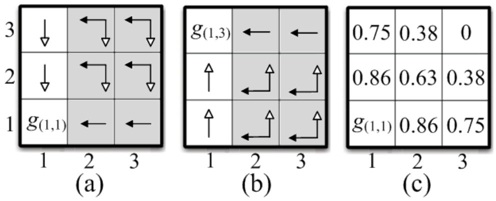


Fig. 3: Examples of the calculation of policy similarity in a simple navigation map.

Settings of Goals To set up a different-goal situation, we can choose a pair of different states as goals. One is for a teacher, and the other is for a student. According to Definition 3, a goal pair indicates a number of policy-similar states, which indicates the maximum number of states where right advice exists. This maximum number would influence the performance of the proposed action advice approach. Hence, the goal pairs in the settings should indicate various numbers of policy-similar states. For a state space S, there are $|S|$ goals and $|S|(|S|-1)$ goal pairs. For each goal, we can get the corresponding optimal policy by using a learning algorithm. Then, for each goal pair, we can get the corresponding policy-similar states. The number of these states is then divided by $|S|-1$ to get its normalisation, named as *policy similarity*. Fig. 3 shows examples of the calculation of policy similarity in a simple navigation map. A state is represented by coordinates (x, y). Fig. 3(a) and Fig. 3(b) show optimal actions, denoted as arrows, for reaching goals $g_{(1,1)}$ and $g_{(1,3)}$ respectively. The solid arrows indicate optimal actions to both $g_{(1,1)}$ and $g_{(1,3)}$, while the hollow arrows indicate optimal actions to either $g_{(1,1)}$ or $g_{(1,3)}$. The shaded states are policy-similar, and the policy similarity of $(g_{(1,1)}, g_{(1,3)})$ is 0.75 (6/8). Fig. 3(c) shows policy similarity values when one goal is $g_{(1,1)}$ and the other goal is a state $g_{(x,y)}$ other than $g_{(1,1)}$. The value shown on $g_{(x,y)}$ is the policy similarity of $(g_{(1,1)}, g_{(x,y)})$. We can see that the policy similarity ranges in $[0, 1)$, and some goal pairs indicate the same policy similarity. For the navigation map (Fig. 2(a)) that we use, we calculate the policy similarity values of all goal pairs, and the distribution is shown in Fig. 2(b). For each policy similarity value, we randomly choose 30 goal pairs as the settings of goals.

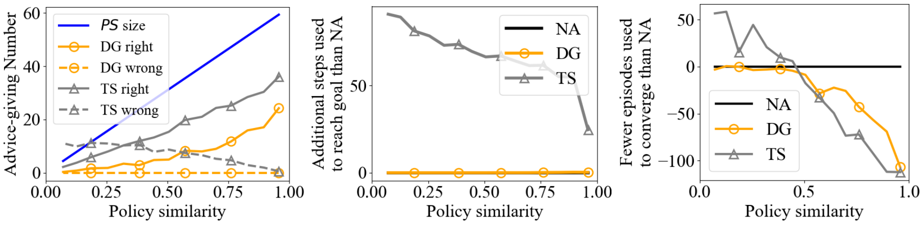
Settings of Two Experiments Q-Learning [11] is used as the learning algorithm due to its popularity. All learning tests are performed for 5000 episodes, with a learning rate of 0.02, a discount factor of 0.99, an exploration factor of 0.01 in ϵ-policy. An agent receives a reward of +200 for reaching its goal, and -1 for each action execution. States transitions are stochastic with a 0.1 probability of failure to an agent's actions. The action set is {Up, Down, Left, Right}. In each state, actions heading towards a wall are not available to an agent. This is to remove a goal which is the same for all agents: avoiding colliding with walls. The settings of action advice approaches used in experiments are as follows.

Experiment 1. The first experiment is to test if a teacher could find the optimal actions to a student's goal (see Aim 1 in Section 3.1). The experiment includes one teacher and one student. The teacher is trained to learn an optimal policy for reaching the teacher's goal before the learning of the student. Three action advice approaches are applied for comparison: (1) the proposed approach which considers the Different-Goal situation (DG); (2) a state-of-the-art Teacher-Student approach (TS) [10]; and (3) No-Advice (NA). TS represents previous action advice approaches developed for the same-goal situation. NA can be regarded as a baseline approach in the different-goal situation. The transmission capacity c in DG is set to 1, which means that at each state, the student can ask for advice to reach only one state, i.e., the student's goal.

Experiment 2. The second experiment is to test if a teacher could give right advice at states where optimal actions to a student's goal exist, but could not be found by the teacher (see Aim 2 in Section 3.1). The transmission capacity c ranges in $\{1, 8, 32\}$. When $c > 1$, the student can ask for advice to reach sub-goals (see Definition 5). The threshold τ for getting sub-goals is set to 0. As positive reward originates from the student's goal, at sub-goals with positive V, the student has found some ways to its goal. Then, reaching one of those sub-goals would be an option when the student does not get advice to its goal. The action advice approaches used in this experiment are DG and NA.

4.2 Results and Analysis

Experiment 1. Fig. 4(a) shows the average advice-giving results of DG and TS. We can see that DG always produces right advice and does not produce wrong advice. This means that by using DG, the teacher successfully finds optimal actions to the student's goal. The amount of right advice increases when policy similarity gets higher. This is because more policy-similar states indicate more states where optimal actions to the student's goal exist. By contrast, TS may produce wrong advice, especially when policy similarity is low. This is because the teacher using TS does not decide the optimality of advised actions. Fig. 4(b) shows the average additional steps used by the student to reach its goal compared with NA. We can see that when applying DG, the student takes almost the same steps to reach its goal as applying NA. This means that the student learns the optimal policy to its goal under most goal pair settings. By contrast, when applying TS, the student takes more steps, especially when policy similarity is low. This is because wrong advice misleads the student, and the student learns a worse policy than applying DG and NA. Fig. 4(c) shows the average fewer episodes used to converge compared with NA. We can see that when applying DG, the student's learning takes fewer episodes to converge, especially when policy similarity is high. The improvement is because taking right advice reduces the exploration space of the student. Taking more right advice results in faster learning. By contrast, when applying TS, although the student learns faster than applying DG when policy similarity is high, the policy learned by the student is worse. When policy similarity gets lower, the learning episodes required to converge grow faster, and the student learns an even worse policy.



(a) (b) (c)

Fig. 4: (a) Advice-giving results, (b) additional steps used to reach goals than NA, (c) fewer episodes used to converge than NA.

Experiment 2. Fig. 5(a) shows the average advice-giving results of DG with various transmission capacities. The result with $c = 1$ indicates the number of states where the teacher finds the optimal actions to the student's goal. When $c > 1$, we can see that the teacher gives right advice at more states than $c = 1$. Larger c results in more right advice given to the student, and results in faster learning speed (shown in Fig. 5(c)). The results indicate that the teacher successfully gives right advice at states where optimal actions to the student's goal exist, but could not be found by the teacher. This is because the optimal actions to the student's goal and sub-goals are possible to be the same. This possibility is 1 when $c = 1$, but would reduce when c gets larger. Fig. 5(a) shows that wrong advice is given when $c = 32$. As a result, Fig. 5(b) shows that the policy learned by the student is a little bit worse than NA when $c = 32$. Fig. 5(b) also shows that when $c = 1$, the policy learned might be a little bit worse than the optimal policy. This indicates that when only right advice is given, the student has a small probability to learn a sub-optimal policy. The investigation on this interesting phenomenon will be left as future work.

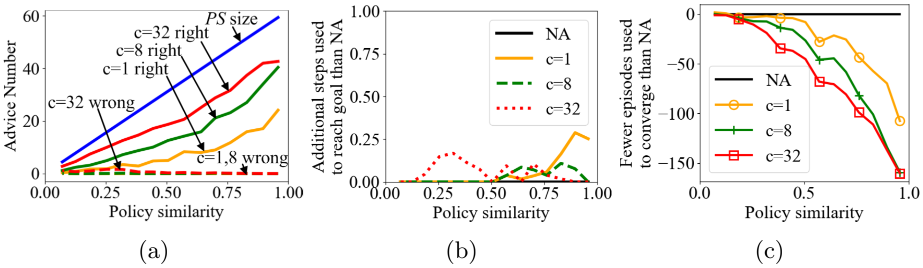


Fig. 5: (a) Advice-giving results, (b) additional steps used to reach goals than NA, (c) fewer episodes used to converge than NA.

5 Related Work

Knowledge Transfer (KT) has been widely used to improve reinforcement learning [9]. Several KT approaches include helping an agent learn in the different-goal situation by, e.g., action set transfer [6], policy transfer [4], MDP distribution transfer [12]. These approaches require learning agents to access and understand the knowledge in source agents. In this paper, agents cannot access the knowledge of each other. The only requirement for agents is a common action set, which enables agents to conduct action transfer (action advice).

Some action advice approaches have been proposed. Chernova and Veloso [2] enabled an agent to ask a human when the agent was uncertain of what actions to take. Torrey *et al.* [10] proposed a teacher-student framework which introduced a limitation on the number of times a teacher could provide advice. Amir *et al.* [1] proposed a jointly-initiated approach which reduced the attention cost of teachers. Zhan *et al.* [15] introduced a multi-teacher advice model where multiple bits of advice from multiple teachers were combined by a majority vote to improve a student's learning. Da Silva *et al.* [3] proposed a simultaneous learning and action advice approach. Ye *et al.* [13] proposed an approach that could reduce the impact of false advice provided by malicious agents. However, the above studies assume that the teacher and student have the same goal, which differs from the different-goal situation that we consider.

6 Conclusion

In this paper, we propose an approach which enables a teacher to help a student learn when they have different goals by action advice (action transfer). Experimental results show the effectiveness of the proposed approach. In future work, we plan to investigate how to conduct action advice in situations where different goals are caused by different S, A, T, R^+, R^- in MDPs. We also plan to study the influence of sub-optimal advice and various state transition functions on the optimality of advised actions. Another issue is to investigate why right advice might lead to sub-optimal policies learned by a student. This phenomenon appears in the results of Experiment 2.

Acknowledgement

This research is supported by a DECRA Project (DP140100007) from Australia Research Council (ARC), a UPA and an IPTA scholarships from University of Wollongong, Australia.

References

1. Amir, O., Kamar, E., Kolobov, A., Grosz, B.J.: Interactive teaching strategies for agent training. In: Proceedings of the 25th International Joint Conferences on Artificial Intelligence. pp. 804–811 (2016)
2. Chernova, S., Veloso, M.: Confidence-based policy learning from demonstration using gaussian mixture models. In: Proceedings of the 6th International Joint Conference on Autonomous Agents and Multiagent Systems. pp. 1315–1322 (2007)
3. Da Silva, F.L., Glatt, R., Costa, A.H.R.: Simultaneously learning and advising in multiagent reinforcement learning. In: Proceedings of the 16th Conference on Autonomous Agents and Multiagent Systems. pp. 1100–1108 (2017)
4. Fernández, F., Veloso, M.: Probabilistic policy reuse in a reinforcement learning agent. In: Proceedings of the fifth International Ioint Conference on Autonomous Agents and Multiagent Systems. pp. 720–727. ACM (2006)
5. Puterman, M.L.: Markov decision processes: discrete stochastic dynamic programming. John Wiley & Sons (2014)
6. Sherstov, A.A., Stone, P.: Improving action selection in mdp's via knowledge transfer. In: Proceedings of the 20th National Conference on Artificial Intelligence. vol. 5, pp. 1024–1029 (2005)
7. Sutton, R.S., Barto, A.G.: Reinforcement learning: An introduction. MIT Press (1998)
8. Taylor, M.E., Carboni, N., Fachantidis, A., Vlahavas, I., Torrey, L.: Reinforcement learning agents providing advice in complex video games. Connection Science **26**(1), 45–63 (2014)
9. Taylor, M.E., Stone, P.: Transfer learning for reinforcement learning domains: A survey. Journal of Machine Learning Research **10**, 1633–1685 (2009)
10. Torrey, L., Taylor, M.: Teaching on a budget: Agents advising agents in reinforcement learning. In: Proceedings of the 12th International Conference on Autonomous Agents and Multiagent Systems. pp. 1053–1060 (2013)
11. Watkins, C.J., Dayan, P.: Q-learning. Machine Learning **8**(3-4), 279–292 (1992)
12. Wilson, A., Fern, A., Ray, S., Tadepalli, P.: Multi-task reinforcement learning: a hierarchical bayesian approach. In: Proceedings of the 24th International Conference on Machine Learning. pp. 1015–1022. ACM (2007)
13. Ye, D., Zhu, T., Zhou, W., Philip, S.Y.: Differentially private malicious agent avoidance in multiagent advising learning. IEEE Transactions on Cybernetics (2019)

14. Yu, C., Zhang, M., Ren, F., Tan, G.: Multiagent learning of coordination in loosely coupled multiagent systems. IEEE Transactions on Cybernetics **45**(12), 2853–2867 (2015)
15. Zhan, Y., Ammar, H.B., Taylor, M.E.: Theoretically-grounded policy advice from multiple teachers in reinforcement learning settings with applications to negative transfer. In: Proceedings of the 25th International Joint Conference on Artificial Intelligence. pp. 2315–2321 (2016)