Teaching Multiple Learning Agents by Environment-Dynamics Tweaks

Hang Xu
Nanyang Technological University
Singapore, 639798
hang017@e.ntu.edu.sg

Ridhima Bector
Nanyang Technological University
Singapore, 639798
ridhima001@e.ntu.edu.sg

Zinovi Rabinovich
Nanyang Technological University
Singapore, 639798
zinovi@ntu.edu.sg

ABSTRACT

In recent years, inducing a desired behaviour in learning agents by exercising some degree of influence over their experiences has received extensive treatment. While there are several offshoots of this problem, it is commonly viewed within the framework of Curriculum Learning: devising a sequence of tasks that gradually induce the desired behaviour. Majority of these methods assume value alignment between the teacher and the learner, though some notable exceptions exist. One such exception is the method of Behaviour Cultivation (BC) that induces a desired behaviour by tweaking the environment dynamics. BC does not assume value alignment, and has been shown to be indispensable, i.e., not reproducible by other teaching methods. Unfortunately, classical BC is an open loop method, i.e., blind to the progress of the learner, and lacks the ability to teach group of agents.

In this paper, we combine the Behaviour Cultivation core with the recent advances of Curriculum MDPs. This allows us to address several shortcomings of the classical BC, while preserving its strengths, such as the freedom from the teacher-learner value alignment. Our model exploits the knowledge of the learner population adaptation process to induce and proliferate a desired behaviour throughout the population. We term our model BC-MDP, and experimentally show its effectiveness, and retention of key positive features of both BC and Curriculum-MDP.

KEYWORDS

Reinforcement Learning, Multi-agent System, Behaviour Cultivation, Curriculum MDP

1 INTRODUCTION

Thaler [28] views a "nudge" as a way to predictably alter people's behaviour by controlling its context. Social change [18], government intervention [19], health care [4], economics [9] and even human-computer interaction [8] have enjoyed its use. Generally though, "nudges" are manually designed based on psychological studies. However, the advent of big data technologies brought an increased pressure to automate nudge generation. In this respect, the proximity of human reasoning to that of Reinforcement Learning (RL) agents (see e.g., [14, 33]), suggests the use of RL teaching paradigms to obtain "nudges".

Majority of RL teaching paradigms can be roughly grouped into three categories: by demonstration or advice (e.g., [2, 3, 15]), by incentives (e.g., [17, 32]), and by environment dynamics design (e.g., [16, 20]). The first two categories explicitly build their approaches on the assumption that the learner has the motivation or interest to accept the teacher's input. However, in many social change scenarios, this assumption is difficult to satisfy. In contrast, environment design approaches need no such assumption. E.g., in Behaviour Cultivation (BC) [24], the teacher and the learner may differ in behaviour preferences, i.e., no value alignment is assumed. Furthermore, BC is a conservative and seeks to minimize environment changes. Alas, unlike its value-aligned counterparts (e.g., Curriculum-MDP [21]), BC constructs an open-loop control sequence of environment modifications, disregarding the learner's actual progress.

In this paper, to achieve a teaching method that is both free of value-alignment and closed-loop, we merge the cores of the Behaviour Cultivation and the Curriculum-MDP approaches. Naturally termed Behaviour Cultivation Markov Decision Process (BC-MDP), our model also retains the conservative view of environment modifications, and seeks minimal effective changes. Motivated by Massive Open Online Courses (MOOCs), we further generalize our teaching by considering the cost of its simultaneous, but independent, application to multiple learners. To summarise, BC-MDP supports the following features simultaneously: a) ability to balance teaching effort, i.e., the amount of environment modifications, with the effectiveness of this modification to drive the learner's behaviour change (i.e., teaching success); b) considering the above balance in the context of a population of learners; c) providing means to control the structural/cognitive complexity of the teaching strategy.

The rest of the paper is organized as follows. Section 2 introduces related works. Sections 3 and 4 gradually build and formalise Behaviour Cultivation MDP model. Experimental support is given in Section 5, followed by limitation and future work discussion in Section 6. Section 7 concludes the work.

2 RELATED WORK

A reinforcement learning agent's behaviour stems solely from its experience in the environment that it inhabits, i.e., how the environment state changes under the agent's action and the reward obtained as the result. The research interest here lies in studying the effect that limited controls of that experience can have on the agent's learned behaviour.

For instance, Curriculum Learning (CL) [6] assumes that no fine-grain control of a single environment is available, and devises a *training sequence* of distinct environments with transferable experiences. The sequence terminates with a *target* environment that the agent is actually supposed to inhabit. The design of the sequence, either manual [20, 23] or automated [27], hastens the agent in obtaining the set of experiences most relevant for optimal behaviour in the target environment. Furthermore, it is possible to construct the *training sequence* on-the-fly, responding to the actual progress of the learner [21]. This is achieved by treating the learner as a dynamic system in which the state is the learner's behaviour and the actions are the training sequence elements. This latter view we borrow to construct our BC-MDP model.

Now, it is not always possible to entirely redesign the environment, and only limited access to its elements is available to the teacher to form the learner's experiences. For instance, Zhang [30–32] considers limited access to the learner's reward function. The teacher wants the learner to adopt a particular behaviour, and seeks the smallest possible change to the learner's reward to incentivize the behaviour adoption. In particular, this means that the teacher pays some cost for the learner's performance. This aspect is absent from CL methods, but is a part of our BC-MDP model.

Reward modulation, however, is neither sufficient for all tasks nor universally available. Fortunately, we can also influence learner's experience by modulating the environment's transition function, i.e., how the environment changes in response to the learner's action. This entails access to the hyper-parameters of the environment dynamics, and has been used in both [16] and [24]. The former searches through the space of possible hyper-parameter setting to facilitate the learner's attainment of its goals. This search changes the environment off-line wrt the learner's experience gathering. In contrast, the latter work, the Behaviour Cultivation method, designs a sequence of hyper-parameter changes that take effect *during* the learner's progress, much in the same way that CL methods do. Notably, both of the aforementioned works consider that changing environment hyper-parameters comes at a cost, and seek to minimize it. Our BC-MDP model follows suit.

Finally, we must position our work with respect to teaching multiple agents. Some examples of doing so exist. E.g., [10, 22] take the route of accessing the reward function of the system, while [26, 29] adopt the CL approach with its strong environment redesign approach. We are not aware of any works that address this issue by modulating environment dynamics, i.e., the teaching method that [16, 24] and our BC-MDP model adopt. Since our model already combines several non-trivial modelling decisions, at this time, we have decided to address only one aspect of teaching a multi-agent system. Specifically, we focus on the need to balance the teaching effort when applied to multiple *independent* learners. Intuitively, we design a MOOC server that administers multiple copies of an interactive scenario.

3 INTUITIVE MODEL SUMMARY

We formulate the problem as a two-level interactive MDP architecture. In this architecture, the learning process is formulated as an Markov Decision Process (MDP) while the teaching process is designed based on the Curriculum-MDP. As shown in Figure 1, the learner *acts in* the environment to explore an optimal behavior policy while the teacher *acts on* the environment dynamics in response to the learners' specific behaviour type. In other words, to induce learners to follow a desired behaviour, the teacher considers the behaviour distribution over the learner population, and accordingly designs a teaching strategy to modulate environment dynamics. In this work, the teaching model aims to generate a time-dependent mapping from learners' behaviour to teacher's action, at the same time, achieve a balance between teaching effort with teaching accuracy.

4 FORMAL MODEL

In this section, we formally define the learner's and the teacher's modus operandi, and derive the teacher's optimisation criteria. Together they form our Behaviour-Cultivation MDP (BC-MDP) model.

4.1 BC-MDP: Learner-Teacher Interaction Model

We follow the Curriculum MDP (CMDP) approach [21], as far as the structure of the interaction between the teacher and the learner is concerned. In particular, we view the learner as a reinforcement learning agent that faces a Markovian environment, and the teacher as an algorithm that controls the hyper-parameters of that environment. However, we significantly differ from CMDP in what parameters of the environment are accessible to the teacher, and the purpose of the teacher's actions. Formally, the following is assumed.

4.1.1 Learner's Environment. Our assumption is that a learner follows some RL algorithms and interacts with a Markovian environment captured by the standard tuple $< S, A, T, r, p_0 >$, where: S is the set of environment states; p_0 is a distribution over the set S from which the initial state of the environment is sampled; A is the set of actions available to the learner; $r : S \times A \times S \longrightarrow \mathbb{R}$ is the reward function, where $r(s, a, s')$ is the benefit extracted by the learner by performing action a in state s and causing the environment to transit to state s'; and, finally, T is the probabilistic state transition function with hyper-parameters coming from some space U. The latter needs a few more details. We assume that the transition function has the form $T : U \longrightarrow \Delta(S)^{S\times A}$, so that $T(s'|s, a; u)$ denotes the probability of the environment state changing from s to s', given that the learner selected action a and the transition was parameterised by $u \in U$.

The goal of the learner is to acquire a behaviour policy, i.e., a probabilistic mapping from environment states to actions, that maximises the expected cumulative reward. We assume that the policy can be parameterised, and thus the probability of the learner taking action $a \in A$ in state $s \in S$ is denoted by $\pi(a|s; \theta)$, where $\theta \in \Theta$ are policy parameters that come from some parameter space Θ. We note that the learner is not aware of environment transition function hyper-parameters, or their changes during its learning progress.

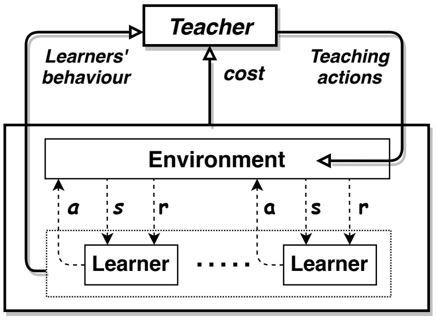


Figure 1: Two-level interactive MDP architecture: learning process is an MDP where *a, s, r* represent learner's action, state and reward; teaching process is a higher-level MDP where state space consists of learners' policies and action space includes all the teaching actions.

4.1.2 Teacher's Environment. We assume that the teacher has full access to the hyper-parameters of a learner's environment, and can modulate them with the confines of the set U. In addition, we consider it possible for the teacher to have access to the learner's policy at regular intervals, but not to modify them directly. Such would be the case of MOOCs, where the skills of students (their behaviour policy) are obtained by regular testing. Following, Curriculum-MDPs [21], we thus formulate the teacher's environment as a higher-level Markovian process. Formally, it is defined by the tuple $< \Theta, U, F, D_0, \theta^*, Cost >$, where:

- Θ, the space of learner policy parameterisations, is treated here as the state space over which the teacher operates. We denote by $\theta^* \in \Theta$ a policy that the teacher considers idealistic in some sense.
- D_0 is a distribution over Θ so that $D_0(\theta)$ is the probability that a learner's initial behaviour is $\theta \in \Theta$.
- U, the space of all hyper-parameters of the learner's environment, becomes the action space of the teacher. We assume that some u_0 exists that describes some "natural" environment response.
- F is the teacher's environment probabilistic transition function, and it captures how a learner changes its behaviour. Essentially, $F(\theta'|\theta, u)$ is the probability that the learner shifts its behaviour parameter from θ to θ' while inhabiting a learner's environment with hyper-parameter $u \in U$.
- $Cost : \Theta \times U \rightarrow \mathbb{R}$ is a function that represents the teaching cost, and combines the teaching effort and the teaching success (as we describe later in Section 4.2 and Table 1), dictated by the hyper-parameters of the learner's environment that the teacher chose to enforce, and how this effected the learner's behaviour parameters.

Unlike the learner, we assume that the teacher has only a finite number of teaching iterations to "tune" the learner's environment, which we denote by T_{max}. Furthermore, rather than treating D_0 as uncertainty in the initial position of a single learner, we treat it as a statistic of a population of learners. That is, the teacher is facing a large number of learners simultaneously, with full knowledge of the current behaviour policy of each one of them, and the overall statistic of those policies given by D_0. We assume that the teacher wants to minimize the total expected teaching cost over the finite horizon. To this end, we allow the teacher to use a non-stationary, probabilistic strategy, so that $\sigma_t(u|\theta)$ denotes the probability that the teacher will use parameterisation $u \in U$ at iteration t to influence a learner with behaviour $\pi(\cdot|\cdot;\theta)$.

4.2 BC-MDP: Strategy Optimality Criteria

Let us now, provide a formal definition of the teacher's optimisation problem beyond its verbal description in Section 4.1.2. Let us begin from the design of the teaching cost function $Cost(u, \theta)$, and follow up with the construction of the overall optimality criteria for a teacher's strategy.

4.2.1 Classical Behaviour Cultivation Cost. As was mentioned in the Section 1, we adopt the teaching cost design from Behaviour Cultivation (BC) [24]. We recap it here for background completeness.

BC recognizes that the choice of environment parameterisation u is only relevant, as long as it has a desired effect on the learner's experience. That is BC does *not* separate between the (teaching) effort it took to build a particular environment variation (i.e. the difference between u and u_0) and (teacher's) success in influencing the learner (i.e., the difference between the current policy $\pi(\cdot|\cdot;\theta)$ and the ideal policy $\pi(\cdot|\cdot;\theta^*)$). Rather, BC compares the *combinations* of u and θ with u_0 and θ^*. To achieve this, BC uses the Kullback-Leibler Divergence Rate (KLR) [25] and compares the overall environment dynamic under u and θ with the dynamic under u_0 and θ^*. Formally, the cost is defined by:

$$\begin{aligned} Cost(u, \theta') &= KLR(P_{u,\theta'}(s', a'|s, a)||P^*(s', a'|s, a)) \\ &= \sum_{s,a} q(s, a) \sum_{s',a'} P_{u,\theta'}(s', a'|s, a) \log \frac{P_{u,\theta'}(s', a'|s, a)}{P^*(s', a'|s, a)} \end{aligned}$$

Here, $P_{u,\theta'}(s', a'|s, a) = T(s'|s, a; u)\pi(a'|s'; \theta')$, i.e., overall environment dynamics when a learner follows the policy $\pi(\cdot|\cdot;\theta')$ in the environment parameterised by $u \in U$. Similarly, $P^*(s', a'|s, a) = T(s'|s, a; u_0)\pi(a'|s'; \theta^*)$, and $q(s, a)$ is the stationary distribution of $P_{u,\theta'}$.

4.2.2 BC-MDP Optimality Design. We adopt BC's cost function as well. However, we significantly differ from BC in a way that the cost is aggregated. BC focused on a single learner, assumed learning progress to be deterministic and used an open-loop teaching strategy. None of these assumptions are present in BC-MDP, which complicates its optimality criterion for the teacher's strategy.

First, recall that we treat D_0 as a statistic over a population of learners. In fact, we must introduce a sequence of distributions D_t, which denote the population statistic after the teaching iteration t. Hence, for $t \in [0 : T_{\max}]$ we have:

$$D_t(\theta') = \sum_{\theta,u} D_{t-1}\sigma_t(u|\theta)F(\theta'|\theta, u)$$

In particular, it means that the teacher's strategy needs to be constructed with population-wide effects in mind. Hence, teaching cost is aggregated both in time, and in expectation over the engendered population dynamic. Formally, this makes the teacher's goal to solve:

$$\min_{\sigma_t} \sum_{t=1}^{T_{\max}} \mathbb{E}_{u\sim\sigma_t,\theta'\sim D_t}[Cost(u, \theta')], \tag{1}$$

where

$$\mathbb{E}_{u\sim\sigma_t,\theta'\sim D_t}[Cost(u, \theta')] = \sum_{\theta'} \sum_{u,\theta} D_{t-1}(\theta)\sigma_t(u|\theta)F(\theta'|\theta, u)KLR(P_{u,\theta'}||P^*) \tag{2}$$

However, we found the optimisation problem in Equation 1 to be unstable. We resolve this issue by introducing a regularisation component. Intuitively, we seek to reduce the cognitive effort required of the teacher to tune the environment individually for each possible learner behaviour. As such a cognitive effort can be conveniently captured by the complexity of the teacher's strategy, we deploy information theoretic regularizer to control it. Namely, we introduce a term $I_t(u, \theta)$ based on mutual information between u and θ into the optimisation criterion. Formally, we define I_t as follows:

$$I_t = \sum_{u,\theta} \sigma_t(u|\theta) D_{t-1}(\theta) \log \frac{\sigma_t(u|\theta) D_{t-1}(\theta)}{\sigma_t(u) D_{t-1}(\theta)} \tag{3}$$

where $\sigma_t(u) = \sum_\theta \sigma_t(u|\theta) D_{t-1}(\theta)$. Choosing a strategy that minimizes this term, the teacher is forced to "compress" its strategy and use similar u for multiple learner behaviours.

Putting them all together, we obtain the teacher's optimality criterion for non-stationary teaching strategies:

$$\min_{\sigma_t} \sum_{t=1}^{T_{\max}} \left\{ \mathbb{E}_{u \sim \sigma_t, \theta' \sim D_t} [Cost(u, \theta')] + \beta I_t \right\}, \tag{4}$$

where β is an importance parameter to control the effect of strategy compression.

Now, for cases such as therapeutic environment design [7], the non-stationary $\sigma_t(u|\theta)$ is the preferred form. E.g., in a group therapy, more aggressive means of influence may become appropriate to handle "straggling" members of the group at later stages of the course. However, in some domains, such as traffic management or green game design [11], a stationary $\sigma(u|\theta)$ is more suitable, because it represents an application of the law, which may be considered immutable during its application. The optimality criterion for stationary $\sigma(u|\theta)$ is:

$$\min_{\sigma} \sum_{t=1}^{T_{\max}} \left\{ \mathbb{E}_{u \sim \sigma, \theta' \sim D_t} [Cost(u, \theta')] \right\} + \beta I_0, \tag{5}$$

In summary, the optimization problem (taking non-stationary $\sigma_t(u|\theta)$ as example) with all constraints can be described as follows. We solve the optimization problem by standard application of Lagrange multipliers.

$$\boxed{\begin{aligned}
&\arg\min_{\sigma_t(u|\theta)} \sum_{t=1}^{Tmax} \{ \sum_{\theta'} \sum_{u,\theta} D_{t-1}(\theta) \sigma_t(u|\theta) F(\theta'|\theta, u) KLR(P_{u,\theta'} || P^*) \\
&\qquad + \beta \sum_{u,\theta} \sigma_t(u|\theta) D_{t-1}(\theta) \log \frac{\sigma_t(u|\theta) D_{t-1}(\theta)}{\sigma_t(u) D_{t-1}(\theta)} \} \\
&\qquad s.t \\
&D_t(\theta') = \sum_{\theta,u} D_{t-1}(\theta) \sigma_t(u|\theta) F(\theta'|\theta, u) \\
&\sigma_t(u) = \sum_{\theta} \sigma_t(u|\theta) D_{t-1}(\theta) \\
&\sum_u \sigma_t(u|\theta) = 1
\end{aligned}}$$

5 EXPERIMENT

In this section, we evaluate the performance of the proposed BC-MDP model and the designed optimality criterion.

5.1 Experiment Setup

The BC-MDP is theoretically feasible in both discrete and continuous domains. In this experiment, we focus on the discrete domain to clearly demonstrate the BC-MDP's performance without intricate variables. We seek a simplest, but quickly extendable, environment that allows us to evaluate the BC-MDP model.

5.1.1 Learning Process Environment.

For the learning process, we build a 4 × 4 windy grid-world as the environment domain as Figure 2. It is necessary to note that the wind application is a mechanism to modulate the environment dynamics. In each state, the agent moves in one of the four cardinal directions. If the state is experiencing the *up/down* wind, the agent's next state will be shifted upward/downward by the wind strength. Additionally, it is important to note that the learner's timeframe is different from the teacher's timeframe. In more details, a teaching iteration t corresponds to several learning time steps. The length of learning duration is unified for all the learners, and it is fixed for each teaching iteration.

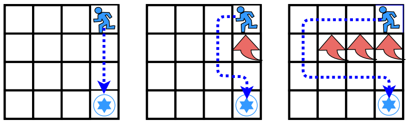


Figure 2: Environment of learning process: 4×4 windy grid-world. The red arrow represents the wind application.

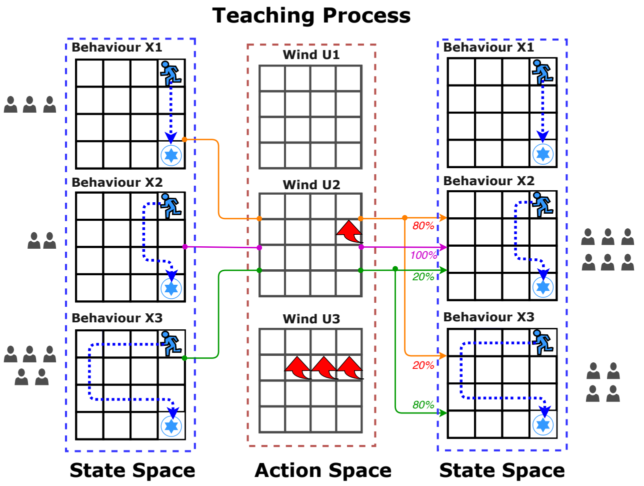


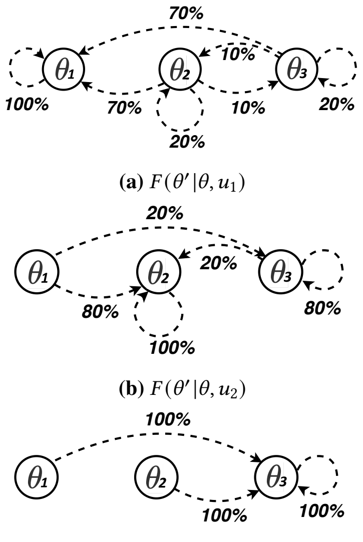
Figure 3: Environment of teaching process: When the teacher implements specific wind on the grid-world, learners accordingly update their behaviour policies following corresponding transition probabilities.

5.1.2 Teaching Process Environment.

For the teaching process, we intentionally define a simple teaching *environment* with artificially small size of Θ and U. As shown in Figure 3, the teaching *environment* includes the learning process, where its state space Θ is the set of learner's possible behaviour policies, and its action space U consists of all kinds of wind applications. In addition, there are two assumptions in this initial experiment. First, the teacher is assumed to know learners' behaviour policies at each teaching iteration, as well as the population behaviour distribution. The second assumption is that the teacher has prior knowledge of the policy transition probability $F(\theta|\theta', u)$. $F(\theta|\theta', u)$ represents the effect of teaching action u on the learner's policy transition, which is designed as Figure 4.

Teaching Terminology	Meaning	Notation
action	The modification action on the environment transition function (dynamics)	$u \in U$
strategy	The time-dependent mapping from learning agents' behaviour to the teaching agent's action	$\sigma_t(u\|\theta)$
accuracy	The proportion of the learning agents following the desired behavior	$D_t(\theta^*)$
cost	The individual deviation caused by the tweaked environment and the undesired policy	$KLR(u, \theta')$
success	The similarity between individual behaviour policy with the desired behaviour	–
effort	The amount of modifications on the environment transition function (dynamics)	–
complexity	The cost of implementing simultaneous but independent teaching actions	–

Table 1: Terminology list about Teaching Process



(a) $F(\theta'|\theta, u_1)$

(b) $F(\theta'|\theta, u_2)$

(c) $F(\theta'|\theta, u_3)$

Figure 4: Policy transition probability: effect of teaching actions on the learner's policy transition.

5.2 Baseline

The BC-MDP model is developed based on the classical BC and the Curriculum-MDP. As described in Section 4, our objective is quite different from the goal of curriculum policy, thus rendering Curriculum-MDP an unfit benchmark in this work. Instead, the classical BC is the ideal benchmark due to the same optimization objective. However, it is not designed for the population cultivation. In order to enable the classical BC work for learner population, we tweak its cost function as:

$$\sum_{\theta'} \sum_{u,\theta} D_{t-1}(\theta)\sigma_t(u)F(\theta'|\theta, u)KLR(u, \theta')$$

Here, $\sigma_t(u)$ is a time-dependent sequence of teaching actions, which is blind to the learner's specific behaviour policy.

5.3 Results

In this section, we evaluate the performance of the BC-MDP via comparing with the classical BC. The responsive feature of the BC-MDP will be discussed and the corresponding effect will be analyzed. Then, we explore the effect of mutual information (i.e., the regularization in the optimality criterion) on the teaching strategy generation. For unambiguous discussion, Table 1 is provided to explain terminologies involved.

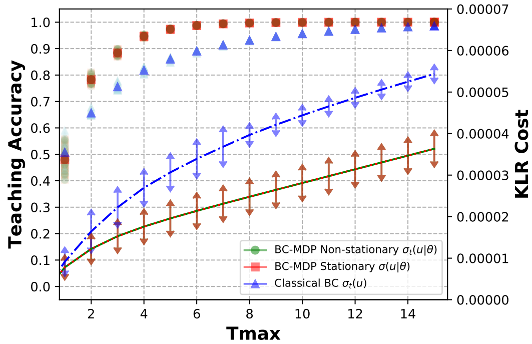


Figure 5: Comparison of the BC-MDP and the classical BC in different T_{max} settings. The scatter represents the final teaching accuracy and the line represents the cumulative KLR cost.

5.3.1 *Effect of Responsive Feature.*

Evaluation in Different $\mathbf{T_{max}}$. The first experiment is to evaluate the teaching performance in different T_{max} settings. It is implemented for a limited-size learner population. Here, we set the initial behaviour distribution as $D_0 = [0.3, 0.2, 0.5]$ and accordingly extract 50 samples whose group consists of 100 learners.

Figure 5 displays the final teaching accuracy and the overall teaching cost generated by the BC-MDP and the classical BC. According to Table 1, the teaching cost is defined as the deviation caused by the tweaked environment and the undesired policy, which measures the balance between teaching effort with teaching success. Therefore, the lower teaching cost means the better balance performance. Since the teaching cost is computed by KLR, the cumulative teaching cost is labeled as *KLR cost* in the figure. Based on Figure 5, we can draw three conclusions as follows.

First, focusing on the teaching accuracy, when the number of teaching iteration is large enough (e.g., $T_{max} = 15$), both the BC-MDP and the classical BC have good teaching performance. However, when T_{max} is limited, the BC-MDP obviously achieves better final teaching accuracy. For example, when $T_{max} = 6$, the BC-MDP non-stationary $\sigma_t(u|\theta)$ achieves 98.75% accuracy and stationary $\sigma(u|\theta)$ achieves 98.70% accuracy, in contrast, the classical BC only reaches 89.04% accuracy. In other words, the BC-MDP improves the teaching accuracy by 10.59%. As shown, the advantage of BC-MDP on teaching accuracy is more prominent as the T_{max} becoming less.

Second, focusing on the overall teaching cost, it is obvious that the BCMDP achieves lower teaching cost whatever the T_{max} settings. As shown in Figure 5, when $T_{max} = 15$, teaching accuracies are similar but differences of teaching costs are obvious. This means that BC-MDP takes less teaching effort when achieving the same accuracy performance. As mentioned before, the teaching cost measures the balance, thus, the proposed BC-MDP is more advantageous in balancing the overall teaching success with the teaching effort.

Third, combining the analysis of both teaching accuracy and teaching cost, we can conclude an optimal T_{max} that is able to achieve good teaching accuracy with the least total effort. As Figure 5, for BC-MDP, the optimal maximum teaching time is $T_{max} = 8$, where the teaching accuracy converges to a stable value while the cumulative teaching cost is the least in horizontal comparison. In contrast, the classical BC has $T_{max} = 15$ as the optimal maximum teaching time. As a result, the BC-MDP makes obvious improvement in teaching efficiency, which utilizes less teaching time to achieve the better teaching accuracy as well as the less total teaching effort.

In addition, the BC-MDP non-stationary teaching strategy $\sigma_t(u|\theta)$ is expected to perform better than the stationary teaching strategy $\sigma(u|\theta)$ due to its responsive ability to the teaching iteration t. However, Figure 5 shows that performances of $\sigma_t(u|\theta)$ and $\sigma(u|\theta)$ are almost the same. The reason is that the experiment design is simple and the policy transition probability is constant in respect of teaching iterations. Since $\sigma_t(u|\theta)$ and $\sigma(u|\theta)$ are quite similar in teaching performance, we use stationary $\sigma(u|\theta)$ as the representation of BC-MDP in following discussion.

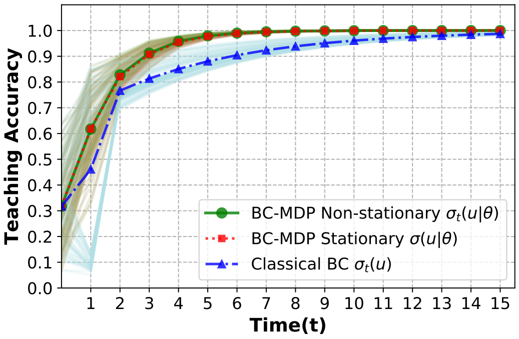
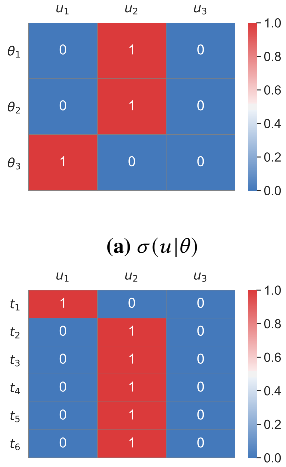


Figure 6: Comparison of the BC-MDP and the classical BC with different initial behaviour distributions. The line represents the teaching accuracy during the teaching process; The scatter marks the accuracy value at each teaching iteration.

Evaluation with Different $\mathbf{D_0(\theta)}$. The second experiment is to demonstrate that, no matter what the initial behaviour distribution, the responsive feature enables the BC-MDP perform better than the classical BC. We implement the experiment with fixed $T_{max} = 15$, and test 100 samples whose initial behaviour distributions are randomly set.

Figure 6 shows the teaching process of the BC-MDP with comparison of the classical BC. As shown, even though both the BC-MDP and the BC achieve final teaching accuracy more than 98.5%, the BC-MDP has a better teaching speed during the process. For example, the BC-MDP successes in nudging 95.83% learners at 4^{th} teaching iteration while the classical BC takes more than 10 teaching iterations to reach 95.10% accuracy. It means the BC-MDP is able to complete a specific goal with less teaching iterations, thus, the better teaching efficiency is achieved by BC-MDP.



	u_1	u_2	u_3
θ_1	0	1	0
θ_2	0	1	0
θ_3	1	0	0

(a) $\sigma(u|\theta)$

	u_1	u_2	u_3
t_1	1	0	0
t_2	0	1	0
t_3	0	1	0
t_4	0	1	0
t_5	0	1	0
t_6	0	1	0

(b) $\sigma_t(u)$

Figure 7: Heatmap representation of teaching strategies in (a) BC-MDP and (b) classical BC, when $T_{\max} = 6$.

Analysis via Teaching Strategy Examples. We will discuss a teaching strategy example in detail, in order to explain results shown in Figure 5 and Figure 6. Heatmaps in Figure 7 represent teaching strategies generated by the BC-MDP and the classical BC.

First, we analyze the teaching strategy generated by the BC-MDP as Figure 7a. As denoted in policy transition probabilities given in Figure 4, θ_1 has 80% probability to transit to the desired θ_2 when trained by u_2. Such high transition probability is the main reason why the teacher chooses u_2 for training θ_1. Besides, for learners with behaviour type θ_3, it is a *two-step* teaching strategy: the teacher first transforms θ_3 to θ_1 by taking u_1 and thereafter takes u_2 to cultivate them to transit to the desired θ_2. Such two-step teaching matches the policy transition probability given in Figure 4. θ_3 only has 20% probability to become θ_2 via u_2, thus, the u_2 is not an efficient teaching action. Instead, since the effort of taking u_1 is less than that of u_2, θ_3 is first trained by the less-effort u_1 and becomes an intermediate behaviour θ_1. Since a specific teaching action is adopted responding to the learner's behaviour type θ, the learner can be cultivated specifically with higher transition efficiency. As a result, BC-MDP has a good performance in teaching accuracy even when the T_{max} is limited.

Then, for the classical BC, one teaching action is applied for all the learners at each teaching iteration. First, since there are 50% learners who have initial behaviour θ_3 according to $D_0(\theta) = [0.3, 0.2, 0.5]$, the teacher takes u_1 at the first teaching iteration t_1. As explained before, it is an efficient teaching action for the majority of learners who follow θ_3. However, the side effect is that the 20% θ_2 will departure from the desired behaviour when trained by u_1, which has negative effect on the teaching accuracy. It explains why the teaching accuracy of the classical BC decreases at $t = 1$. Then, the teacher takes u_2 for all the learners until the last teaching iteration. For learners who follow θ_1 and θ_2, u_2 is an ideal teaching action due to 80% and 100% transition probability as shown in Figure 4. However, u_2 is of low efficiency for cultivating the minority who follows θ_3 due to the low transition probability. As a result, more teaching iterations are required to cultivate θ_3. This is the main reason why the classical BC performs poorly when T_{max} is limited but performs well when T_{max} is large enough.

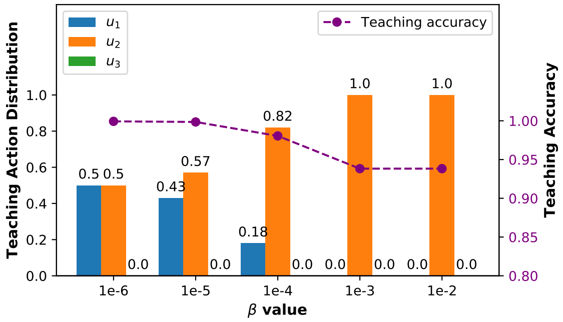


Figure 8: Effect of the regularizer (mutual information) in different β settings: the bar represents the teaching action distribution; the dot line means the final teaching accuracy.

5.3.2 *Effect of Mutual Information.*

As description in Section 4, the mutual information I_t controls the complexity of the teacher's strategy. It compresses the teacher's strategy to use similar u for multiple learner behaviours, where the parameter β controls the effect of strategy compression. To evaluate the effect of mutual information, we fixed $T_{max} = 10$ and set $D_0(\theta) = [0.3, 0.2, 0.5]$.

Figure 8 shows teaching action distribution and teaching accuracy in different β settings. For stationary teaching strategy, when the β is small, it successes in nudging most learners due to the responsive feature playing the leading role. With the increasing of β, the teaching action diversity is decreasing until only one teaching action adopted. At the same time, the teaching accuracy is becoming worse due to the less responsive teaching action. Therefore, the mutual information controls the responsive level between the teaching strategy with learners' behaviours. In this way, it influences the trade-off between teaching accuracy with the teaching complexity (i.e., the structural/cognitive complexity of the teaching strategy).

6 LIMITATION AND FUTURE WORK

This work experimentally demonstrates that the proposed BC-MDP preserves strengths of the BC, and furthermore performs better on teaching efficiency for population cultivation. Nevertheless, the model is developed with assumptions which may cause a gap between theoretical model and practical applications. To further improve the feasibility and scalability, promising directions of the future work are described as follows.

Agent Modeling: It is assumed that the teacher exactly knows the behaviour policy of the individual learner, and also knows the behaviour distribution of the population. In some applications, this assumption can be satisfied, such as MOOCs. However, it is difficult to meet such assumption in other scenarios, such as traffic management. To widen the breath of applicability, we would like to adopt agent modeling approaches [1, 12] that enable the teacher learn the learner's behaviour policy via interactions.

Policy Transition Probability: Another assumption is that the teacher has prior knowledge on learners' policy transition probabilities. With this assumption, the population cultivation is regarded as a planning problem. In the future work, we would like to extend such planning problem into a learning problem, where the teacher needs to learn the transition probabilities when designing the teaching strategy. In this way, the proposed BC-MDP model would be feasible for more complex environment where preliminary experiments are difficult to implement.

Extension to Teamwork: Currently, the proposed BC-MDP model is suitable for scenarios where multiple learners are independent with each other. It means that there is no cooperation or communication among the learner population. And learners have no joint objective, instead, they seek the individual optimal reward. In the further work, we would like to further explore the teamwork cultivation, where agents cooperate with each other to achieve a common goal, such as swarm system [5, 13].

Effect on Learning Process: The teaching model can be integrated with variant learning algorithms, even though this work does not discuss this part deeply. In the future experiment, we would like to integrate the teaching algorithm with different learning algorithms, such as deterministic algorithms (i.e., dynamic programming) and stochastic algorithm (i.e., neural network). The effect of the BC-MDP on the learning process would be explored further, such as the improvement on learning speed.

7 CONCLUSION

We proposed a teaching model BC-MDP for population behaviour cultivation. The model is built as a two-level interactive MDP architecture with responsive features, which generates a time-dependent and time-independent mapping from the learner's specific behaviour to the teaching action. We designed an optimality criterion to balance the teaching accuracy with the teaching effort and complexity. Our empirical evaluation supports the feasibility and effectiveness of the BC-MDP, as well as its retention of positive features from the BC. Moreover, the BC-MDP is more advantageous compared with the classical BC, especially on teaching accuracy within limited teaching iterations, and on teaching balance between the accuracy with total effort. This work demonstrates that the proposed BC-MDP model is a potentially effective model for automate nudge generation.

8 ACKNOWLEDGEMENTS

This work was supported by SCSE NTU SUG "Choice Manipulation and Security Games".

REFERENCES

[1] Stefano V Albrecht and Peter Stone. 2018. Autonomous agents modelling other agents: A comprehensive survey and open problems. *Artificial Intelligence* 258 (2018), 66–95.

[2] Ofra Amir, Ece Kamar, Andrey Kolobov, and Barbara Grosz. 2016. Interactive teaching strategies for agent training. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence*.

[3] Brenna D Argall, Sonia Chernova, Manuela Veloso, and Brett Browning. 2009. A survey of robot learning from demonstration. *Robotics and autonomous systems* 57, 5 (2009), 469–483.

[4] Anneliese Arno and Steve Thomas. 2016. The efficacy of nudge theory strategies in influencing adult dietary behaviour: a systematic review and meta-analysis. *BMC public health* 16, 1 (2016), 676.

[5] Levent Bayındır. 2016. A review of swarm robotics tasks. *Neurocomputing* 172 (2016), 292–321.

[6] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. 2009. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*. 41–48.

[7] Etienne Burdet, Yanan Li, Simone Kager, Karen Sui-Geok Chua, Asif Hussain, and Domenico Campolo. 2018. Interactive robot assistance for upper-limb training. In *Rehabilitation Robotics*. Elsevier, 137–148.

[8] Ana Caraban, Evangelos Karapanos, Daniel Gonçalves, and Pedro Campos. 2019. 23 Ways to Nudge: A Review of Technology-Mediated Nudging in Human-Computer Interaction. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. ACM, 503.

[9] Rachel Davis, Rona Campbell, Zoe Hildon, Lorna Hobbs, and Susan Michie. 2015. Theories of behaviour and behaviour change across the social and behavioural sciences: a scoping review. *Health Psychology Review* 9, 3 (2015), 323–344.

[10] Lachlan Dufton and Kate Larson. 2009. Multiagent policy teaching. *In Proceedings of the 8th International Conference on Autonomous Agents and MultiAgent Systems* (2009).

[11] Fei Fang, Peter Stone, and Milind Tambe. 2015. When security games go green: Designing defender strategies to prevent poaching and illegal fishing. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence*.

[12] Aditya Grover, Maruan Al-Shedivat, Jayesh K Gupta, Yura Burda, and Harrison Edwards. 2018. Learning policy representations in multiagent systems. *arXiv preprint arXiv:1806.06464* (2018).

[13] Maximilian Hüttenrauch, Sosic Adrian, Gerhard Neumann, et al. 2019. Deep reinforcement learning for swarm systems. *Journal of Machine Learning Research* 20, 54 (2019), 1–31.

[14] Julian Jara-Ettinger, Hyowon Gweon, Laura E Schulz, and Joshua B Tenenbaum. 2016. The naïve utility calculus: Computational principles underlying commonsense psychology. *Trends in cognitive sciences* 20, 8 (2016), 589–604.

[15] Parameswaran Kamalaruban, Rati Devidze, Volkan Cevher, and Adish Singla. 2019. Interactive Teaching Algorithms for Inverse Reinforcement Learning. *arXiv preprint arXiv:1905.11867* (2019).

[16] Sarah Keren, Luis Pineda, Avigdor Gal, Erez Karpas, and Shlomo Zilberstein. 2017. Equi-Reward Utility Maximizing Design in Stochastic Environments. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*. 4353–4360.

[17] W Bradley Knox and Peter Stone. 2013. Learning non-myopically from human-generated reward. In *Proceedings of the international conference on Intelligent user interfaces*. 191–202.

[18] Mark Kosters and Jeroen Van der Heijden. 2015. From mechanism to virtue: Evaluating Nudge theory. *Evaluation* 21, 3 (2015), 276–291.

[19] Susan Michie and Robert West. 2013. Behaviour change theory and evidence: a presentation to Government. *Health Psychology Review* 7, 1 (2013), 1–22.

[20] Sanmit Narvekar, Jivko Sinapov, Matteo Leonetti, and Peter Stone. 2016. Source task creation for curriculum learning. In *Proceedings of the 15th International Conference on Autonomous Agents and Multiagent Systems*. 566–574.

[21] Sanmit Narvekar, Jivko Sinapov, and Peter Stone. 2017. Autonomous Task Sequencing for Customized Curriculum Design in Reinforcement Learning. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*. 2536–2542.

[22] Sriraam Natarajan, Gautam Kunapuli, Kshitij Judah, Prasad Tadepalli, Kristian Kersting, and Jude Shavlik. 2010. Multi-agent inverse reinforcement learning. In *2010 Ninth International Conference on Machine Learning and Applications*. 395–400.

[23] Bei Peng, James MacGlashan, Robert Loftin, Michael L Littman, David L Roberts, and Matthew E Taylor. 2016. An empirical study of non-expert curriculum design for machine learners. In *Proceedings of the IJCAI Interactive Machine Learning Workshop*.

[24] Zinovi Rabinovich, Lachlan Dufton, Kate Larson, and Nick Jennings. 2010. Cultivating desired behaviour: Policy teaching via environment-dynamics tweaks. *In Proceedings of the 9th International Conference on Autonomous Agents and MultiAgent Systems* (2010), 1097–1104.

[25] Ziad Rached, Fady Alajaji, and L Lorne Campbell. 2004. The Kullback-Leibler divergence rate between Markov sources. *IEEE Transactions on Information Theory* 50, 5 (2004), 917–921.

[26] Golden Rockefeller, Patrick Mannion, and Kagan Tumer. 2019. Curriculum Learning for Tightly Coupled Multiagent Systems. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*. 2174–2176.

[27] Felipe Leno Da Silva and Anna Helena Reali Costa. 2018. Object-oriented curriculum generation for reinforcement learning. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*. 1026–1034.

[28] Richard H Thaler and Cass R Sunstein. 2009. *Nudge: Improving decisions about health, wealth, and happiness*. Penguin.

[29] Weixun Wang, Tianpei Yang, Yong Liu, Jianye Hao, Xiaotian Hao, Yujing Hu, Yingfeng Chen, Changjie Fan, and Yang Gao. 2019. From Few to More: Large-scale Dynamic Multiagent Curriculum Learning. *arXiv preprint arXiv:1909.02790* (2019).

[30] Haoqi Zhang, Yiling Chen, and David C Parkes. 2009. A general approach to environment design with one agent. In *Twenty-First International Joint Conference on Artificial Intelligence*.

[31] Haoqi Zhang and David C Parkes. 2008. Value-Based Policy Teaching with Active Indirect Elicitation. In *Twenty-Third AAAI Conference on Artificial Intelligence*, Vol. 8. 208–214.

[32] Haoqi Zhang, David C Parkes, and Yiling Chen. 2009. Policy teaching through reward function learning. In *Proceedings of the 10th ACM conference on Electronic commerce*. ACM, 295–304.

[33] Brian D Ziebart, Andrew L Maas, J Andrew Bagnell, and Anind K Dey. 2009. Human Behavior Modeling with Maximum Entropy Inverse Optimal Control. In *AAAI Spring Symposium: Human Behavior Modeling*, Vol. 92.