Automatic Curriculum for Unsupervised Reinforcement Learning

Yucheng Yang
Department of Mathematics and
Computer Science
Eindhoven University of Technology
Eindhoven, The Netherlands
y.yang@tue.nl

Tianyi Zhou
Department of Computer Science
University of Maryland, College Park
College Park, United States
tianyi@umd.edu

Lei Han
Tencent AI Lab
Shenzhen, China
lxhan@tencent.com

Meng Fang
Department of Computer Science
University of Liverpool
Liverpool, United Kingdom
Eindhoven University of Technology
Eindhoven, The Netherlands
Meng.Fang@liverpool.ac.uk

Mykola Pechenizkiy
Department of Mathematics and
Computer Science
Eindhoven University of Technology
Eindhoven, The Netherlands
m.pechenizkiy@tue.nl

ABSTRACT

Unsupervised reinforcement learning (URL) relies on carefully designed training objectives rather than task rewards to learn general skills. However, we lack quantitative evaluation metrics for URL but mainly rely on visualizations of trajectories for comparison. Moreover, most URL methods choose to optimize a single training objective, which may hinder later-stage learning and the development of new skills. To bridge these gaps, we first introduce a combination of metrics that can evaluate diverse properties of URL. We show that balancing these metrics in URL leads to better performance and trajectories with empirical evidence and theoretical insights. Next, we develop an automatic curriculum that uses a non-stationary multi-armed bandit algorithm to select URL objectives for different learning episodes, resulting in a balanced improvement on all the metrics. Extensive experiments in different environments demonstrate the advantages of our method in achieving promising and balanced performance on multiple metrics when compared to recent URL methods.

KEYWORDS

Unsupervised Reinforcement Learning; Auto curriculum learning; Intrinsic Motivation

ACM Reference Format:
Yucheng Yang, Tianyi Zhou, Lei Han, Meng Fang, and Mykola Pechenizkiy. 2024. Automatic Curriculum for Unsupervised Reinforcement Learning. In *Proc. of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2024), Auckland, New Zealand, May 6 – 10, 2024*, IFAAMAS, 18 pages.

1 INTRODUCTION

Reinforcement learning (RL) has recently achieved remarkable success in autonomous control [25] and video games [33]. Its mastery of Go [43] and large-scale multiplayer video games [46] has drawn growing attention. However, a primary limitation for the current RL is that it is highly task-specific and easily overfitting to the training task, while it is still challenging to acquire basic skills that are generalisable across tasks. Moreover, most RL methods still suffer from sparse rewards and insufficient exploration in many tasks. To overcome these weaknesses, intrinsic motivations [35] have been introduced to pre-train RL agents in earlier stages even without a task assigned. Unsupervised RL (URL) does not rely on any extrinsic task rewards, and its primary goal is to encourage exploration and develop versatile skills that can be adapted to downstream tasks.

Although URL provides additional objectives and rewards to train fundamental and task-agnostic skills, it lacks quantitative evaluation metrics and yet relies mainly on visualizations of trajectories [7, 13, 42] to demonstrate its effectiveness. One significant drawback of visualizations is that they require prior knowledge to represent at most two or three more important dimensions of the state, which contradicts the concept of 'unattended' RL. Although URL learned skills can be evaluated through downstream tasks by their extrinsic rewards [27], this requires further training and can be prone to overfitting or bias towards specific tasks. A key challenge in developing evaluation metrics for URL is how to cover different expectations or preferable properties for the agent, which usually cannot be all captured by a single metric. Recently, the concept of disentanglement is introduced in [24] to evaluate the informativeness and separability of learned skills, and state coverage to evaluate how well the state space is explored. However, their implementation of the state coverage needs prior knowledge of the environment to partition it into equally divided bins, and there lacks theoretical justification for the new disentanglement metrics. In addition, how to balance multiple metrics in the evaluation is an open challenge. Therefore, it is critical to develop a set of metrics that can provide a complete and precise evaluation of an URL agent.

In contrast to the ambiguity of evaluation metrics for current URL, the existing intrinsic rewards for URL are quite specific and focused, e.g., the novelty/uncertainty of states [6, 38, 39], the entropy of state distribution [29, 31, 34], and the mutual information between states and skills [13, 20], which are task-free and can provide dense feedback. For example, as shown later, agent learning with a single intrinsic reward for exploration could be hindered

Proc. of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2024), N. Alechina, V. Dignum, M. Dastani, J.S. Sichman (eds.), May 6 – 10, 2024, Auckland, New Zealand. © 2024 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org).

from further exploration since its novelty approximation is limited to local regions. For most URL methods, only one intrinsic reward is used. They mainly differ on instantiations, e.g., how to define the novelty, how to estimate the state entropy or mutual information, etc. However, the quality of instantiations is significantly affected by the error of modeling the environmental dynamics or possibility density functions. Moreover, in order to achieve consistent improvement on multiple evaluation metrics and balance their trade-offs, training with a single intrinsic reward is not enough. Hence, it is necessary for URL to take multiple intrinsic rewards into account as training objectives.

In this paper, we take a first step towards **quantitative multi-criteria evaluations** of URL by proposing a set of complementary evaluation metrics that can cover different preferred capabilities of URL, e.g., on both exploration and skill discovery. In the case studies, we show that URL achieves balanced and high scores over all the proposed metrics, which meet our requirements of a promising pre-trained agent. In contrast, excelling on only one metric cannot exclude certain poorly learned URL policies. **Regarding the training objectives**, we consider multiple existing intrinsic rewards and choose the most helpful one in each learning stage for maximizing the proposed evaluation metrics. To this end, we develop a curriculum of URL whose training objective is not static but adaptive to the needs of different training stages, whose goal is to keep improving all the involved evaluation metrics. Since the intrinsic reward is varying concurrently with URL policy on the fly, we apply a multi-objective multi-armed bandits algorithm to address the exploration-exploitation trade-off. We intend to select the intrinsic reward (1) that has been rarely selected before (exploration) or (2) that results in the greatest and balanced improvement over all into account as metrics in history (exploitation). Specifically, we adopt Pareto UCB [11] to optimize the multi-objective defined by the metrics and then extend it to capture the non-stationary dynamics of best reward, i.e., which may change across learning stages. This assumption is in line with our observation that a single intrinsic reward cannot keep improving all metrics, while URL may stop exploration and ends with sub-optimal skills.

Our contributions are:

(1) To the best of our knowledge, our work is among a few pioneering studies focusing on developing evaluation metrics for URL.
(2) We theoretically justify the necessity of disentanglement metric for URL.
(3) We introduced automatic curriculum learning to Unsupervised RL.
(4) In experiments, we evaluate our curriculum URL in challenging URL environments, showing that our proposed metrics faithfully capture multiple properties of learned skills that could benefit downstream tasks, and our method consistently achieves better and more balanced results on multiple evaluation metrics than existing URL methods.
(5) We present extensive empirical analyses demonstrating the advantages of automatic curriculum and the multi-objective for optimizing the curriculum.

2 PRELIMINARIES

MDP without external rewards. In Markov Decision Process (MDP) $\mathcal{M} = (\mathcal{S}, \mathcal{A}, p)$ *without external rewards*, $\mathcal{S}$ and $\mathcal{A}$ respectively denote the state and action spaces, and $p(s_{t+1}|s_t, a_t)$ is the transition function where $s_t, s_{t+1} \in \mathcal{S}$ and $a_t \in \mathcal{A}$. Given a policy $\pi(a_t|s_t)$, a trajectory $\tau = (s_0, a_0, \ldots, s_T)$ follows the distribution $\tau \sim p(\tau) = p(s_0) \prod_{t=0}^{T-1} \pi(a_t|s_t) p(s_{t+1}|s_t, a_t)$. We formulate the problem of unsupervised skill discovery as learning a skill-conditioned policy $\pi(a_t|s_t, z)$ where $z \in \mathcal{Z}$ represents the latent skill. The latent representations of skills z can be either continuous $z \in \mathbb{R}^d$ or discrete $z \in \{z_1, z_2, ..., z_{N_z}\}$. $H(\cdot)$ and $I(\cdot;\cdot)$ denote entropy and mutual information, respectively.

3 RELATED WORKS

Unsupervised Reinforcement Learning. Intrinsic rewards are used for training URL. For exploration, intrinsic motivations can be based on curiosity and surprise of environtal dynamics [8], such as Intrinsic Curiosity Module (ICM) [38], Random Network Distillation (RND) [6], and Disagreement [39]. Another common way to explore is to maximize the state entropy. State Marginal Matching (SMM) [29] approximates the state marginal distribution, and matching it to the uniform distribution is equivalent to maximizing the state entropy. Other methods approximate state entropy by particle-based method MEPOL [34], APT [31], ProtoRL [47], APS [30]. Mutual Information-based Skill Learning (MISL) has been used for self-supervised skill discovery, such as VIC [20], DIAYN [14], VALOR [1]. VISR [21] also optimizes the same ojective, but its special approximation brought successor feature [2] into unsupervised skill learning paradigm and enables fast task inference. APS [30] combines the exploration of APT and successor feature of VISR.

Automatic Curriculum Learning. Automatic curriculum learning has been widely studied. It allows models to learn in a specific order for learning harder tasks more efficiently [3, 19]. In RL, a lot of work considers scheduling learning tasks [16–18, 32, 41]. In URL, handcrafted curriculum is used by EDL [7] and IBOL [24]. EDL first explores, then assigns the discovered states to skills, and finally learns to achieve those skills. IBOL also explores and assigns skill after a specific linearzer learning phase. Both of them are not automatic curriculum, and the number of training steps for each training phase needs to be specified before training. In addition, VALOR [1] mentioned curriculum learning, but their curriculum is just gradually increasing the number of skills.

4 BACKGROUND AND MOTIVATION

Unsupervised Reinforcement Learning needs to explore the environment and learn basic skills to prepare for downstream tasks. So URL agent naturally needs to optimize at least two objectives: one for exploration, and another for skill learning, eg., the state entropy and the mutual information between state and skill. Most prior URL algorithms try to explore and learn good skills with only a single intrinsic reward or by a simple linear combination of two intrinsic rewards. We argue that a single intrinsic reward is not enough and we need to combine the advantages of multiple intrinsic rewards to balance the multiple objectives related to exploration and skill learning.

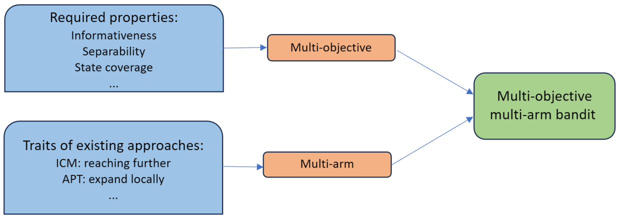


Figure 1: Multiple properties are important for exploration and skill learning of URL. Existing URL approaches have advantages and disadvantages. We can combine and take advantage of multiple intrinsic rewards to balance the URL agent for multiple properties.

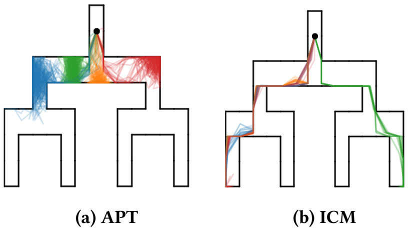

(a) APT (b) ICM

Figure 2: These are the trajectories of URL agents learned by single intrinsic rewards in a continuous tree maze environment. (a) is learned by APT and (b) is learned by ICM. Different colors represent the trajectories of different skills.

4.1 Why combine multiple intrinsic rewards?

Existing URL methods only use a single intrinsic reward to train their policies. Since the accuracy of the reward depends on the agent's modeling of the environment dynamics [38, 39] or state entropy [29, 31], whose quality heavily relies on the data collected through the agent's exploration, URL keeping using the same reward may stop exploration earlier with a sub-optimal policy.

Fig. 2 shows the trajectories of two agents learned by URL. The agent in (a) is learned with the simple linear combination of a single intrinsic reward for exploration (APT [31]) and the common skill learning objective $I(S;Z)$ [13]. The agent in (b) is also learned with the simple linear combination of one for exploration (ICM [38]) and $I(S;Z)$ for skill learning. The trajectories are in different colors representing different skills. Despite the fact that both agents are learned with a single exploration reward and a skill-learning reward, there are significant differences in the trajectories of their skills. By examining the visualized trajectories, we can see that the trajectories of APT agent are getting thicker but stay near to the starting state. It seems that APT, as a method of optimizing the state entropy $H(S)$, prefers to make the agent expand its state coverage locally. While the ICM agent reaches further areas, its trajectories are thin, possibly because its reward relies on a prediction model of environmental dynamics. When this model is accurate in a large part of the state space, the intrinsic reward might lead the agent to go only along where the model is not as accurately approximated. With this empirical example showing the advantages and disadvantages of APT and ICM, we can consider the possibility of a method combining both intrinsic rewards and taking advantage of both to make the skills both have enough coverage and reach distant areas.

4.2 Why optimize multiple objectives?

As mentioned before, as an URL agent that needs to explore and learn skills, it already has two objectives. We argue that there could be more necessary objectives for URL. For example, some previous work mentioned the distance between the starting state and the ending state could be an objective to encourage the trajectory length of skills [36]. And from this empirical example, we can see that some skills in fig. 2b overlaps and can not be separated from each other, so the disentanglement metric mentioned in [24] that measures separability between skills can also be an objective.

Besides, similar to humans that learn from basic skills to advanced skills [3, 23], at each learning stage of URL, the primary objective should also be different. For example, in the beginning, exploration should be primary, so metrics like the coverage of state space could be more important. In the later stage, the quality of learned skills should be the main concern, then it should be focused on improving the diversity and separability of skills.

5 METHODOLOGY

To combine the advantages of multiple intrinsic rewards for multiple objectives, we propose an automatic curriculum learning framework for URL

5.1 Overview of Automatic Curriculum for URL

Instead of keep using one intrinsic reward for training an URL agent, we allow the agent to choose one reward among multiple candidates in each learning stage (multiple episodes of learning) for improving multiple evaluation metrics. This generates an automatic curriculum for URL whose goal is to find a sequence of intrinsic rewards that optimizes multi-objectives each corresponding to a metric. The framework of our proposed automatic curriculum method is illustrated in fig. 3. Our curriculum adds an outer loop outside the conventional URL framework (i.e., the interaction between RL agent and environment). The curriculum has a reward selection module that selects an intrinsic reward for each learning stage based on multiple evaluation metrics computed on the replay buffer. By allocating the intrinsic reward that can result in the greatest improvement on multiple metrics in each stage, the curriculum aims to find an optimal sequence of intrinsic reward choices to keep improving the URL loop and optimize all the metrics.

Given previous work in URL, we still need to address two primary new challenges in building the curriculum: (1) how to is the intrinsic reward for each learning stage selected? and (2) what are the evaluation metrics? We propose our solutions to these two problems in Section 5.3 and 5.2, respectively. In Section 5.2, we discuss the exploration-exploitation trade-off for award selection in URL and extend a multi-objective multi-armed bandits algorithm to make non-stationary decisions on the reward used for each learning stage's URL. In Section 5.3, we propose multiple metrics to evaluate the capability of URL on exploration and skill learning. These metrics not only include existing ones but also cover other preferred properties.

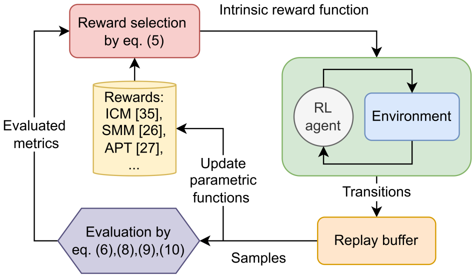

Reward selection by eq. (5)
Intrinsic reward function
Rewards: ICM [35], SMM [26], APT [27], ...
RL agent
Environment
Evaluated metrics
Update parametric functions
Transitions
Evaluation by eq. (6),(8),(9),(10)
Samples
Replay buffer

Figure 3: Block diagram of our proposed method: The inner loop is an RL agent interacting with an environment and learning from intrinsic reward selected by the reward selection module. The outer loop is a reward selection module selecting intrinsic rewards based on historical progress on the evaluation metrics.

5.2 Automatic Curriculum for URL

The curriculum for unsupervised RL should also be *unsupervised*, meaning no prior knowledge or extrinsic metric is allowed. The mechanism of choosing the next intrinsic reward for the agent should only be based on the historical information collected from environmental interactions. In order to make better choices, it also needs to try different intrinsic rewards and evaluate the improvement they bring to URL. Therefore, an exploration-exploitation trade-off process, e.g., a multi-armed bandit algorithm, is critical to curriculum development. Since our goal is an agent excelling on multiple evaluation metrics, the curriculum should take all these multiple objectives into account when selecting a reward for the next stage of training. In addition, due to the non-stationary dynamics of a curriculum, the best intrinsic reward may change across learning stages. Hence, we need an automatic curriculum that is (1) *unsupervised*, (2) *multi-objective*, and (3) able to handle *nonstationarity*.

We formulate it as a Multi-objective multi-armed bandit problem and adopt the empirical Pareto UCB [11] algorithm because of its easier implementation and tighter regret bound, and we need a non-stationary extension of Pareto UCB to capture the changes. This Empirical Pareto UCB should learn a task selection function $\mathcal{D} : \mathcal{H} \longrightarrow \mathcal{U}$ where $\mathcal{H}$ can contain any information about previous interactions and $\mathcal{U}$ is the finite candidate set of intrinsic rewards as tasks. The goal of $\mathcal{D}$ is to minimize the regret with respect to the Pareto optimal for these objectives:

$$\max_{\mathcal{D}} \mathbb{E}_{\mathcal{D}}[\sum_{t=0}^{T} P_t^1],\ \max_{\mathcal{D}} \mathbb{E}_{\mathcal{D}}[\sum_{t=0}^{T} P_t^2],\ \cdots,\ \max_{\mathcal{D}} \mathbb{E}_{\mathcal{D}}[\sum_{t=0}^{T} P_t^p], \tag{1}$$

where T is the total number of selections in the curriculum and $P \in \mathbb{R}^p$ is a vector with each entry corresponding to each evaluation metric, and it is the learning progress of the evaluation metrics $P_t = M_t - M_{t-1}$, where $M_t \in \mathbb{R}^p$ is a vector containing each entry as a considered evaluation metric. P^i is the ith entry of P. $H_t \in \mathcal{H}$ is composed of the $(P_j, u_j = \mathcal{D}(H_{t-1}))$ tuples for all $j < t$. The decision of task selection $= \mathcal{D}(H_t)$ is recurrently affected by $P_j, \forall j < t$ in the experience.

For Empirical Pareto UCB, task is uniform randomly chosen from the Pareto action set

$$\left\{u \mid \forall v \in U,\ \bar{\mu}_t(v) + c\sqrt{\frac{\ln(t\sqrt[4]{pK})}{N_t(v)}} \nsucc \bar{\mu}_t(u) + c\sqrt{\frac{\ln(t\sqrt[4]{pK})}{N_t(u)}}\right\}, \tag{2}$$

where $u_t \in \mathcal{U}$ is the intrinsic reward selected by UCB, and $\bar{\mu}(u) \in \mathbb{R}^p$ is the weighted sum of the past P by training agent in the intrinsic rewards of the task u. $N(u)$ is the number of times u has been chosen. K is a empirical number that upper bounds the Pareto optimal set of arms. $\nsucc$ means non-dominant. We say that x is non-dominated by y, $y \nsucc x$, if and only if there exists at least one dimension j for which $y^j < x^j$ [48].

This is a nonstationary multi-arm bandit, because the intrinsic reward resulting in best learning progress might change along with the agent's learning process. There are two common ways to adapt the UCB algorithm for nonstationary situations. One way to do this is with discounting and another way is to use a sliding window [28]. We find that discounting has better performance in experiments. Let $\gamma \in (0, 1)$ be the discount factor, and define

$$\bar{\mu}_t^{\gamma}(u) = \sum_{s=0}^{t-1} \gamma^{t-s} P_s \mathbb{I}\{u_s = u\}, \tag{3}$$

and

$$N_t^{\gamma}(u) = \sum_{s=0}^{t-1} \gamma^{t-s} \mathbb{I}\{u_s = u\}. \tag{4}$$

When using discounting for nonstationary Empirical Pareto UCB, the Pareto action set becomes

$$\begin{aligned}\Bigg\{u \mid \forall v \in U,\ \bar{\mu}_t^{\gamma}(v) + c\sqrt{\frac{\ln(\sum_{w \in \mathcal{U}} N_t^{\gamma}(w)\sqrt[4]{pK})}{N_t^{\gamma}(v)}} \nsucc \\ \bar{\mu}_t^{\gamma}(u) + c\sqrt{\frac{\ln(\sum_{w \in \mathcal{U}} N_t^{\gamma}(w)\sqrt[4]{pK})}{N_t^{\gamma}(u)}}\Bigg\}.\end{aligned} \tag{5}$$

Below in algorithm 1 is the algorithm outline of our proposed Unsupervised Multi-Objective Curriculum (UMOC) algorithm.

```
Algorithm 1 UMOC
  Candidate set $\mathcal{U}$ of intrinsic rewards, total number of task selections $T$, set $H_t = \emptyset$ initial Pareto action set $\mathcal{P}_a = \mathcal{U}$, number of episodes for each selected intrinsic reward $\tau$, URL agent $A$.
  $t \leftarrow 0$
  while $t \leq T$ do
      Get $u_t$ by uniformly sampling from $\mathcal{P}_a$
      Learn agent $A$ with intrinsic reward $u_t$ for $\tau$ episodes
      Evaluate agent $A$ with metrics $M_t$
      $P_t \leftarrow M_t - M_{t-1}$
      $H_t \leftarrow H_{t-1} \cup (P_t, u_t)$
      Update $\bar{\mu}_t^{\gamma}(u), N_t^{\gamma}(u)$ with $H_t$ by eqs. (3) and (4)
      Update $\mathcal{P}_a$ with $\bar{\mu}_t^{\gamma}(u), N_t^{\gamma}(u)$ by eq. (5).
  end while
```

One question could be why we consider using Pareto UCB instead of using a single objective method by linearly combining the multiple objectives. 4.7.4 of [4] showed that optimizing the linearization (weighted sum) of objectives is incapable of finding all the points in a non-convex Pareto optimal set of solutions.

5.3 Multiple Evaluation Metrics

In the following section, our goal is developing general and consistent metrics to evaluate the process of exploration and skill learning. They can be used as the objectives for the outer loop of task selection, and their evaluations on the past intrinsic reward choices can be the historical information affecting the choice of the reward selection module. Previous works rely on visualizations of trajetories to compare the performance of exploration and skill learning [7] [24] [37] [13]. However, visualisations only show two or three dimensions at most. Trying to reduce the number of state dimensions for visualizable projections requires prior knowledge about the importance of dimensions, which contradicts the concept of *unsupervised.* Next, we will define general metrics that quantitatively evaluate exploration and skill learning, which can evaluate URL online during training without any prior knowledge on downstream tasks.

State Coverage (SC). Exploration is important for RL in general. It is also prerequisite for good skill learning, only when enough states is explored can we assign them into skills. This metric evaluates how much of the state space the agent can cover. Similar to previous methods that approximate the state entropy using particle-based methods, our metric is also based on particle-based entropy. By [44] the particle-based entropy estimation should be a sum of the log of the distance between each particle and its k-th nearest neighbor, defined as

$$H_{\mathrm{PB}}(S) \propto \sum_{i=1}^{n} \log \|s_i - s_i^{(k)}\|. \tag{6}$$

For robust and stable implementation, we use the modified version from APT [31], see Appendix B. We find that with large numbers of n and k, this estimation can reflect what a good state coverage is in visualization, so we apply it for a large number of recent states in the buffer to evaluate the learning progress of state coverage.

Particle-based Mutual Information (PMI). Mutual information between state S and a latent skill Z is an essential objective for URL to learn a skill conditioned agent. Intuitively, [15] showed that, under some assumption, maximizing the objective $I(S;Z)$ initialize the agent to be optimal for certain downstream tasks. Most previous MISL approaches approximate this objective by approximating the possibility density function of the state distribution, which is not suitable for evaluation, because it could suffer from variance of neural network approximation. PMI circumvent this by proposing a non-parametric approach.

Mutual information between state and skill can be expanded as

$$I(S;Z) = H(S) - H(S|Z). \tag{7}$$

It is one entropy subtracting another, so we propose to implement particle-based entropy to approximate them and obtain the mutual information.

$$I(S;Z) \approx \hat{H}_{\mathrm{PB}}(S) - \hat{H}_{\mathrm{PB}}(S|Z). \tag{8}$$

The approximation bias of particle-based entropy in Eq. 6 depends on k and n, so with the large sample numbers of n and k for both $\hat{H}_{\mathrm{PB}}(S)$ and $\hat{H}_{\mathrm{PB}}(S|Z)$, we can get accurate approximation from this substraction.

Disentanglement. Disentanglement for URL should includes two aspects for learned skills: Informativeness and separability. Informativeness here means information shared between a skill and its inferred states. Separability in representation means that there should be no information shared between two latent dimensions, while here it means the trajectories inferred by two different skills should be separated from each other.

Disentanglement metric such as SEPIN@k and WSEPIN [9] from representation learning have been implemented for URL [24]. However, we argue that these two metrics are in fact not suitable for URL in appendix C. Instead, we proposed novel metric $I(S;\mathbf{1}_z)$ to measure the informativeness and separatability of an individual skill z, random binary variable $\mathbf{1}_z$ is the indicator of $Z = z$. Appendix D shows why it measures the properties and how it can be estimated. Instead of ignoring the ones with lower informativeness, all learned skills of an URL agent have an impact on the agent's behavior, so we care about the median and minimum of the informativeness of the skills.

We defined Median SEParability and INformativeness (MSEPIN) as

$$\mathrm{MSEPIN} = \operatorname*{med}_{z} I(S;\mathbf{1}_z), \tag{9}$$

where $\operatorname*{med}_{z}$ is the median over skills z, and Least SEParability and INformativeness (LSEPIN) as

$$\mathrm{LSEPIN} = \min_{z} I(S;\mathbf{1}_z). \tag{10}$$

Furthermore, we proposed theoretical justification for this metric, showing that it can be a complement of $I(S;Z)$ to evaluate how well the URL agent is prepared for downstream tasks in appendix F Proofs and explainations can be found in Appendix F.

Empirical analysis of our proposed metrics are in appendix H

6 EXPERIMENTS

In this section, we present experimental results with our proposed automatic curriculum framework to show its advantages. To provide an intuitive comparison between visualizations and proposed metrics, the results we present here are mainly from visualizable environments of continuous mazes [7] and Ant environment from [42]. There exists a URL benchmark [27], but it is not suitable for MISL algorithms, as discussed and analyzed in appendix J.

6.1 Setup

We compare our methods with baselines in two categories:

Conventional URL with a single intrinsic reward Minimal implementation of previous URL approaches, details in Appendix G.

Random curriculum We consider curriculums with randomly selected intrinsic rewards. By comparing our proposed method to this random curriculum, we could validate whether the bandit algorithm for selection is necessary.

The intrinsic reward candidates for the experiments are ICM [38], SMM [29], APT [31], for exploration and $\log(p(z|s))$ of DIAYN [13] for skill learning. More details are in appendix G. In our experimental setting, every exploration intrinsic reward is linearly combined with $\log p(z|s)$ for skill learning, eg., the intrinsic reward for APT is a linear combination of $H_{PB}(S)$ and $\log p(z|s)$. We call the full version of our proposed method UMOC, and the single objective version, when the metric M is a one-dimensional value, of our method USOC.

6.2 Advantages of Curriculum URL

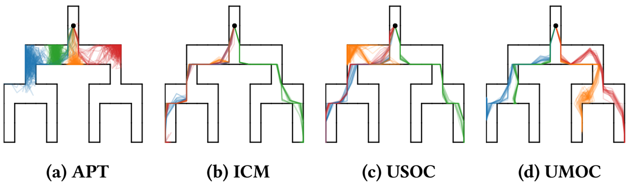

(a) APT (b) ICM (c) USOC (d) UMOC

Figure 4: Tree maze trajectories of (a) APT, (b) ICM, (c) USOC, and (d) UMOC.

Here we show how the combination of intrinsic rewards can take advantage of the candidates and complement their disadvantages. In a continuous tree maze environment, Our method uses APT and ICM as two arms. For the tree maze environment, we consider a single objective of State Coverage (SC) defined in Section 5.3 We run APT, ICM, Random curriculum and our method with the same series of 5 random seeds. The total number of reward selections is 100, one selection made every 500 episodes.

Table 1: State coverage comparison between single intrinsic rewards and combining them with our method USOC, the second value in the entries are metrics normalized by the maximum

	APT	ICM	Random	USOC
mean	1397 / 0.947	1215 / 0.824	1391 / 0.943	**1475 / 1.0**
std	54.12 / 0.18	304.28 / 1.0	114.82 / 0.38	**48.99 / 0.16**

By first looking at the trajectories learned by APT and ICM shown in Fig. 4a and 4b, the results show that APT prefers to make the agent expand its state coverage locally. The trajectories are getting thicker but stay near the starting state. ICM reaches further areas, but its trajectories are thin, possibly because its reward relies on a prediction model of environmental dynamics. When this model is accurate in a large part of the state space, the intrinsic reward might lead the agent to go only along where the model is not as accurately approximated. We found that as learning progresses, Ours (SO) prefers to choose APT more. The number of times APT is chosen in the later half of training is on average 32.6% higher than the first half. This is in agreement with an intuition that for better state coverage, the agent should first reach further and then expand its trajectories.

In table 1, it's clear that USOC dominates both in mean performance and performance variance. The baseline of the random curriculum could not be better than the single APT, so the order of intrinsic rewards in the curriculum matters and USOC is capable of finding a good curriculum.

6.3 Advantages of Multi-Objective for Curriculum

To test the advantages of multi-objective, for the same tree maze, we consider multiple objectives including SC, PMI and LSEPIN. Intrinsic reward candidates for selection are SMM, ICM and APT.

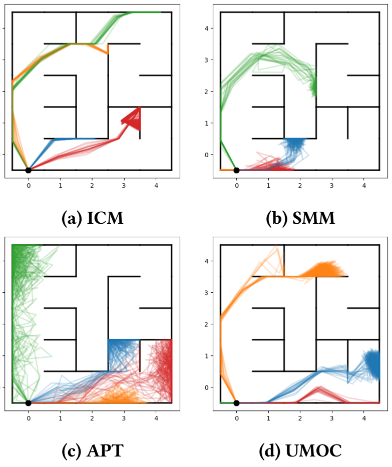
Table 2 shows the results that we compare the multi-objective Pareto UCB to UCB with only SC objective. Overall, MO works better than SO. It means that learning multiple objectives benefits the agent. Fig. 4 (c) and (d) show two agents learned by USOC with single objective of SC and UMOC learned by multi-objective of {SC, PMI, LSEPIN} respectively. The SO agent has 3 skill on the left side of the tree, while the MO agent has trajectories of all skills separated. This is an example of why UMOC has better disentanglement on average.

Table 2: Comparison between UMOC and USOC. The second value in the entries are metrics normalized by the maximum

Method	SC	PMI	LSEPIN	MSEPIN
UMOC	1394 / 0.945	**921.34 / 1.0**	**198.61 / 1.0**	**274.66 / 1.0**
USOC	**1475 / 1.0**	887.01 / 0.96	179.08 / 0.90	232.91 / 0.85

Table 3: Comparison between UMOC and baselines for 5x5 maze

Method	**SC**	**PMI**	**LSEPIN**	**MSEPIN**
UMOC	**1552.8 / 1.0**	**1076.9 / 1.0**	**251.6 / 1.0**	**345.4 / 1.0**
Random	1338.4 / 0.86	883.2 / 0.82	148.6 / 0.59	284.9 / 0.82
APT	1457.1 / 0.94	848.7 / 0.79	188.7 / 0.75	266.4 / 0.77
SMM	1279.9 / 0.82	764.5 / 0.71	137.4 / 0.55	229.7 / 0.67
ICM	1375.1 / 0.89	647.5 / 0.60	83.3 / 0.33	228.7 / 0.66

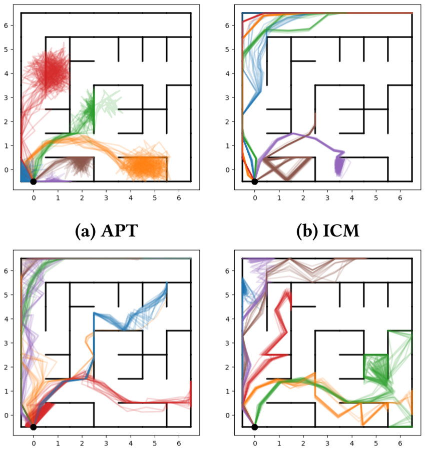


(a) ICM (b) SMM
(c) APT (d) UMOC

Figure 5: Trajectories samples of the crazy maze: (a) ICM, (b) SMM, (c) APT, and (d) UMOC.

Fig. 5 shows sample trajectories on the the crazy maze environment, which is a more complicated 2D maze environment than the tree maze. Our method works better than others. Table 3 shows that the well-rounded performance of our proposed method is consistent.

6.4 More results for 2D crazy maze



(a) APT (b) ICM

(c) SMM (d) UMOC

Figure 6: Trajectories samples of the crazy maze: Comparing single intrinsic rewards and our methods with multi-objective

We also evaluated our method on a larger crazy maze environment. This is an environment much more difficult to explore than the simple tree maze.

In Fig. 7, we compare of our method with individual intrinsic rewards and the random curriculum baseline. As we can see from Fig. 7, as expected, our proposed method achieves a well-rounded evaluation result for all 4 proposed metrics. Also, the quantitative result intuitively accords with visualizations in Fig. 6. For example, SMM has better SC metric but less disentanglement (MSEPIN and LSEPIN) than APT. In 6, SMM covered more to the upper right of the maze, but its green and purple skills seem to be entangled with each other.

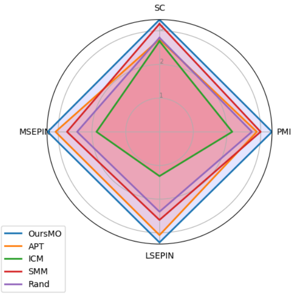
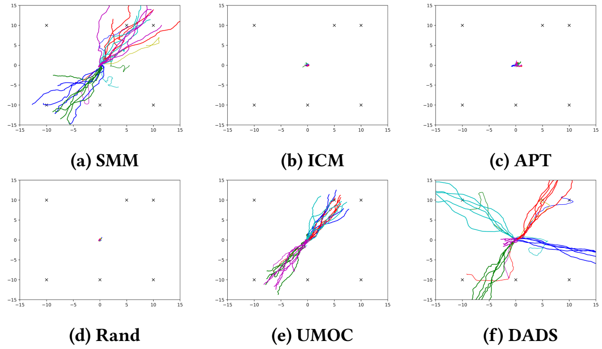


Figure 7: Multi-metrics.

6.5 Mujoco Ant environment

We have provided experimental results with complicated randomly generated mazes in section 6.4. Besides maze environments, we have also tested our multi-objective method on a high-dimensional Mujoco Ant by the setting from DADS[42], which learns navigation skills for ant and the downstream tasks are to reach certain destinations. It is a challenging environment for URL.

The reward candidates for our method are minimal implementations of SMM, ICM, and APT, their intrinsic rewards are chosen to be linearly combined with $\log p(z|s)$ for skill learning. The original DADS implementation exploited prior knowledge of state dimensions and used specific tuning and scaling as well as other tricks, so it has the best metric evaluation, visualization, and downstream task performance. The minimal implementations of ICM and APT without specific tuning have bad performance for this environment, which was also shown in [24, 36]. However, our method can combine the advantages and complement the disadvantages of the candidates, resulting in comparable evaluation metrics and downstream task performance to the original DADS implementation.



(a) SMM (b) ICM (c) APT

(d) Rand (e) UMOC (f) DADS

Figure 8: Ant trajectories samples projected on x-y axis: (a) ICM, (b) SMM, (c) APT, and (d) randomly choose rewards, (e) UMOC (f) original implementation of DADS.

Table 4: Metrics and downstream task performance

	R	SC	PMI	LSEPIN	MSEPIN
DADS (origin)	-0.340	2501.23	248.15	29.91	71.81
UMOC	**-0.467**	**2271.13**	**132.54**	**26.71**	**36.64**
Random	-0.999	308.30	21.04	≈ 0	4.31
SMM	-0.680	2416.86	122.95	18.19	31.77
ICM	-1.001	626.64	30.19	≈ 0	12.64
APT	-0.991	1081.82	28.15	2.63	27.78
Correlation efficient		0.91	0.96	**0.99**	0.88

The experimental results also validated the strong correlation between our proposed metrics and downstream task performance as shown in the last row of table 4. More results of the Ant environment are in Appendix K.

7 CONCLUSION

We proposed quantifiable and general evaluation metrics for URL and justified their necessity. Our proposed metrics can stably measure the state coverage for exploration, as well as mutual information and disentanglement for skill learning. This helps to enable evaluation of URL without specific downstream tasks. Furthermore, we proposed an automatic curriculum to select intrinsic rewards based on the agent's learning progress. This automatic curriculum does not require prior knowledge of the environment or its intrinsic reward candidates. It is a nonstationary Pareto UCB that utilizes historical evaluations for decision-making and tries to train the agent to be well-rounded in all aspects of the considered metrics. Our experimental results have demonstrated the effectiveness of our method. The proposed metrics evaluate with fully observable state samples, one future work would be developing evaluation methods for pixel-based or partially observable observations.

ACKNOWLEDGMENTS

This work is funded by the SmartTwo project of the KPN (Koninklijke PTT Nederland NV) and Tencent Robotics X.

REFERENCES

[1] Joshua Achiam, Harrison Edwards, Dario Amodei, and Pieter Abbeel. 2018. Variational option discovery algorithms. *arXiv preprint arXiv:1807.10299* (2018).

[2] André Barreto, Will Dabney, Rémi Munos, Jonathan J. Hunt, Tom Schaul, David Silver, and Hado van Hasselt. 2017. Successor Features for Transfer in Reinforcement Learning. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett (Eds.). 4055–4065. https://proceedings.neurips.cc/paper/2017/hash/350db081a661525235354dd3e19b8c05-Abstract.html

[3] 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.

[4] Stephen P. Boyd and Lieven Vandenberghe. 2014. *Convex Optimization*. Cambridge University Press. https://doi.org/10.1017/CBO9780511804441

[5] George W Brown. 1951. Iterative solution of games by fictitious play. *Act. Anal. Prod Allocation* 13, 1 (1951), 374.

[6] Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. 2019. Exploration by random network distillation. In *International Conference on Learning Representations*.

[7] Victor Campos, Alexander Trott, Caiming Xiong, Richard Socher, Xavier Giró-i-Nieto, and Jordi Torres. 2020. Explore, Discover and Learn: Unsupervised Discovery of State-Covering Skills. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event (Proceedings of Machine Learning Research, Vol. 119)*. PMLR, 1317–1327. http://proceedings.mlr.press/v119/campos20a.html

[8] Stefano I. Di Domenico and Richard M. Ryan. 2017. The Emerging Neuroscience of Intrinsic Motivation: A New Frontier in Self-Determination Research. *Frontiers in Human Neuroscience* 11 (2017). https://doi.org/10.3389/fnhum.2017.00145

[9] Kien Do and Truyen Tran. 2019. Theory and evaluation metrics for learning disentangled representations. *arXiv preprint arXiv:1908.09961* (2019).

[10] Kien Do and Truyen Tran. 2019. Theory and Evaluation Metrics for Learning Disentangled Representations. https://doi.org/10.48550/ARXIV.1908.09961

[11] Madalina M Drugan and Ann Nowe. 2013. Designing multi-objective multi-armed bandits algorithms: A study. In *The 2013 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 1–8.

[12] Ishan Durugkar, Steven Hansen, Stephen Spencer, and Volodymyr Mnih. 2021. Wasserstein Distance Maximizing Intrinsic Control. *CoRR* abs/2110.15331 (2021). arXiv:2110.15331 https://arxiv.org/abs/2110.15331

[13] Benjamin Eysenbach, Abhishek Gupta, Julian Ibarz, and Sergey Levine. 2019. Diversity is All You Need: Learning Skills without a Reward Function. In *ICLR*.

[14] Benjamin Eysenbach, Abhishek Gupta, Julian Ibarz, and Sergey Levine. 2019. Diversity is All You Need: Learning Skills without a Reward Function. In *International Conference on Learning Representations*.

[15] Benjamin Eysenbach, Ruslan Salakhutdinov, and Sergey Levine. 2022. The Information Geometry of Unsupervised Reinforcement Learning. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net. https://openreview.net/forum?id=3wU2UX0voE

[16] Meng Fang, Tianyi Zhou, Yali Du, Lei Han, and Zhengyou Zhang. 2019. Curriculum-guided hindsight experience replay. *Advances in neural information processing systems* 32 (2019).

[17] Carlos Florensa, David Held, Xinyang Geng, and Pieter Abbeel. 2018. Automatic goal generation for reinforcement learning agents. In *International conference on machine learning*. PMLR, 1515–1528.

[18] Carlos Florensa, David Held, Markus Wulfmeier, Michael Zhang, and Pieter Abbeel. 2017. Reverse curriculum generation for reinforcement learning. In *Conference on robot learning*. PMLR, 482–495.

[19] Alex Graves, Marc G Bellemare, Jacob Menick, Remi Munos, and Koray Kavukcuoglu. 2017. Automated curriculum learning for neural networks. In *international conference on machine learning*. PMLR, 1311–1320.

[20] Karol Gregor, Danilo Jimenez Rezende, and Daan Wierstra. 2017. Variational Intrinsic Control. In *International Conference on Learning Representations*.

[21] Steven Hansen, Will Dabney, André Barreto, David Warde-Farley, Tom Van de Wiele, and Volodymyr Mnih. 2020. Fast Task Inference with Variational Intrinsic Successor Features. In *International Conference on Learning Representations*.

[22] Shuncheng He, Yuhang Jiang, Hongchang Zhang, Jianzhun Shao, and Xiangyang Ji. 2022. Wasserstein unsupervised reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 6884–6892.

[23] Faisal Khan, Bilge Mutlu, and Jerry Zhu. 2011. How do humans teach: On curriculum learning and teaching dimension. *Advances in neural information processing systems* 24 (2011).

[24] Jaekyeom Kim, Seohong Park, and Gunhee Kim. 2021. Unsupervised Skill Discovery with Bottleneck Option Learning. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event (Proceedings of Machine Learning Research, Vol. 139)*, Marina Meila and Tong Zhang (Eds.). PMLR, 5572–5582. http://proceedings.mlr.press/v139/kim21j.html

[25] Bahare Kiumarsi, Kyriakos G Vamvoudakis, Hamidreza Modares, and Frank L Lewis. 2017. Optimal and autonomous control using reinforcement learning: A survey. *IEEE transactions on neural networks and learning systems* 29, 6 (2017), 2042–2062.

[26] Michael Laskin, Hao Liu, Xue Bin Peng, Denis Yarats, Aravind Rajeswaran, and Pieter Abbeel. 2022. CIC: Contrastive Intrinsic Control for Unsupervised Skill Discovery. arXiv:arXiv:2202.00161

[27] Michael Laskin, Denis Yarats, Hao Liu, Kimin Lee, Albert Zhan, Kevin Lu, Catherine Cang, Lerrel Pinto, and Pieter Abbeel. 2021. URLB: Unsupervised Reinforcement Learning Benchmark. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual*, Joaquin Vanschoren and Sai-Kit Yeung (Eds.). https://datasets-benchmarks-proceedings.neurips.cc/paper/2021/hash/091d584fced301b442654dd8c23b3fc9-Abstract-round2.html

[28] Tor Lattimore and Csaba Szepesvári. 2020. Bandit Algorithms. (2020).

[29] Lisa Lee, Benjamin Eysenbach, Emilio Parisotto, Eric P. Xing, Sergey Levine, and Ruslan Salakhutdinov. 2019. Efficient Exploration via State Marginal Matching. *CoRR* abs/1906.05274 (2019). arXiv:1906.05274

[30] Hao Liu and Pieter Abbeel. 2021. APS: Active Pretraining with Successor Features. In *International Conference on Machine Learning*.

[31] Hao Liu and Pieter Abbeel. 2021. Behavior From the Void: Unsupervised Active Pre-Training. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan (Eds.). 18459–18473. https://proceedings.neurips.cc/paper/2021/hash/99bf3d153d4bf67d640051a1af322505-Abstract.html

[32] Tambet Matiisen, Avital Oliver, Taco Cohen, and John Schulman. 2019. Teacher–student curriculum learning. *IEEE transactions on neural networks and learning systems* 31, 9 (2019), 3732–3740.

[33] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. 2013. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602* (2013).

[34] Mirco Mutti, Lorenzo Pratissoli, and Marcello Restelli. 2021. A Policy Gradient Method for Task-Agnostic Exploration. In *Conference on Artificial Intelligence*.

[35] Pierre-Yves Oudeyer and Frederic Kaplan. 2009. What is intrinsic motivation? A typology of computational approaches. *Frontiers in neurorobotics* 1 (2009), 6.

[36] Seohong Park, Jongwook Choi, Jaekyeom Kim, Honglak Lee, and Gunhee Kim. 2022. Lipschitz-constrained Unsupervised Skill Discovery. *ArXiv* abs/2202.00914 (2022).

[37] Seohong Park, Jongwook Choi, Jaekyeom Kim, Honglak Lee, and Gunhee Kim. 2022. Lipschitz-constrained Unsupervised Skill Discovery. https://doi.org/10.48550/ARXIV.2202.00914

[38] Deepak Pathak, Pulkit Agrawal, Alexei A Efros, and Trevor Darrell. 2017. Curiosity-driven exploration by self-supervised prediction. In *International Conference on Machine Learning*.

[39] Deepak Pathak, Dhiraj Gandhi, and Abhinav Gupta. 2019. Self-Supervised Exploration via Disagreement. In *International Conference on Machine Learning*.

[40] Julia Robinson. 1951. An iterative method of solving a game. *Annals of mathematics* (1951), 296–301.

[41] Jürgen Schmidhuber. 2013. Powerplay: Training an increasingly general problem solver by continually searching for the simplest still unsolvable problem. *Frontiers in psychology* 4 (2013), 313.

[42] Archit Sharma, Shixiang Gu, Sergey Levine, Vikash Kumar, and Karol Hausman. 2020. Dynamics-Aware Unsupervised Discovery of Skills. In *International Conference on Learning Representations*.

[43] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. 2016. Mastering the game of Go with deep neural networks and tree search. *nature* 529, 7587 (2016), 484–489.

[44] Harshinder Singh, Neeraj Misra, Vladimir Hnizdo, Adam Fedorowicz, and Eugene Demchuk. 2003. Nearest Neighbor Estimates of Entropy. *American Journal of Mathematical and Management Sciences* 23, 3-4 (2003), 301–321.

[45] Alexander Trott, Stephan Zheng, Caiming Xiong, and Richard Socher. 2019. Keeping your distance: Solving sparse reward tasks using self-balancing shaped rewards. *Advances in Neural Information Processing Systems* 32 (2019).

[46] Oriol Vinyals, Igor Babuschkin, Wojciech M Czarnecki, Michaël Mathieu, Andrew Dudzik, Junyoung Chung, David H Choi, Richard Powell, Timo Ewalds, Petko Georgiev, et al. 2019. Grandmaster level in StarCraft II using multi-agent reinforcement learning. *Nature* 575, 7782 (2019), 350–354.

[47] Denis Yarats, Rob Fergus, Alessandro Lazaric, and Lerrel Pinto. 2021. Reinforcement Learning with Prototypical Representations. In *International Conference on*

Machine Learning.

[48] Eckart Zitzler, Lothar Thiele, Marco Laumanns, Carlos M Fonseca, and Viviane Grunert Da Fonseca. 2003. Performance assessment of multiobjective optimizers: An analysis and review. *IEEE Transactions on evolutionary computation* 7, 2 (2003), 117–132.