Continual Reinforcement Learning with Group Symmetries

Shiqi Liu*, Mengdi Xu*, Peide Huang, Xilun Zhang, Yongkang Liu†, Kentaro Oguchi† and Ding Zhao
Department of Mechanical Engineering, Carnegie Mellon University
†Toyota Motor North America R&D

***Abstract*—Continual reinforcement learning aims to sequentially learn a variety of tasks, retaining the ability to perform previously encountered tasks while simultaneously developing new policies for novel tasks. However, current continual RL approaches overlook the fact that certain tasks are identical under basic group operations like rotations or translations, especially with visual inputs. As a result, they unnecessarily create and train a new policy for each similar task, leading to poor sample efficiency and weak generalization capability. To address this, we introduce a unique Continual Reinforcement Learning method that recognizes Group Symmetries, cultivating a policy for each group of equivalent tasks rather than individual tasks. We introduce a PPO-based RL algorithm with an invariant feature extractor and a novel task grouping mechanism that relies on invariant features. We evaluate our algorithm on sequences of robotic manipulation tasks, where each group is associated with different objects. We show that our algorithm assigns tasks to different groups with high accuracy and outperforms baselines in terms of generalization capability by a large margin. Furthermore, we transitioned the policy, initially trained in a simulated environment, to a real-world robotic arm without fine-tuning. The results indicate that the agent is capable of adeptly solving different tasks.**

I. Introduction

Quickly adapting to unseen tasks has been a key desideratum in realistic reinforcement learning (RL) settings [8, 15, 14]. RL algorithms are generally trained in simulated environments and then deployed in the real world. However, pre-trained RL agents will likely encounter new tasks during their deployment. Blindly reusing policies obtained from the training time could cause significant performance drops and even lead to catastrophic failures in safety-critical applications [31]. Motivated by human's strong learning capability, in this work, we aim to enhance RL algorithms' quick adaptation capability in the continual learning setting.

Continual RL (also known as lifelong RL) focuses on learning a stream of tasks sequentially while generating task-specific policies for the present task and retaining the capability to solve seen tasks [14, 12, 17, 26]. Continual RL assumes non-stationary environments with changing transitions or reward functions, which result in changing tasks. The task delineations are provided by the environment or automatically detected via unsupervised learning algorithms [27]. However, the task taxonomies in existing works ignore the geometric information embedded in task representations, which naturally emerges in practical applications.

Consider an example in the field of robotic manipulation, as shown in Figure 1. Here, a robotic arm is tasked to press a button located in 2 different locations. The robot utilizes bird's-eye view images as inputs. In this scenario, it is possible that the controller might receive reflection-symmetric bird's-eye view images as inputs.

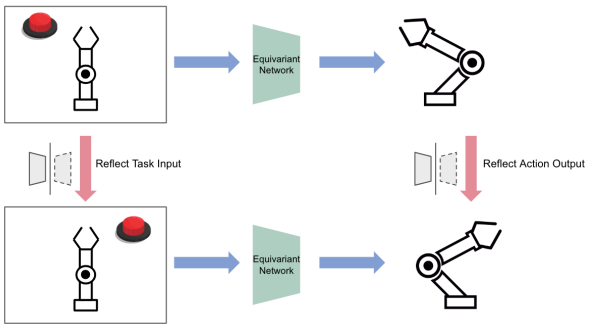

Fig. 1: This example illustrates how group symmetric information enhances adaptability. The robotic arm is instructed to press a button situated in two distinct locations, using bird's-eye view images as inputs. In this scenario, considering the symmetry of the button's location around the robot's position, the optimal control policies should be equivalent but mirrored.

However, with the bird-eye view images as inputs, prior continual RL works would treat each rotated configuration as a new task and learn the task from scratch. Such a learning process impedes positive interference among tasks and limits the agent's adaptivity. Hence, we aim to leverage the geometric similarity among tasks in the continual RL setting to adapt to unseen but equivalent tasks quickly.

In this work, we propose a continual RL algorithm that encodes group symmetries in the state space to achieve strong generalization capability and solves equivalent tasks in a zero-shot manner. We assume that there exist tasks that are equivalent under group operations, and all equivalent tasks under the same group operation construct a *task group*. Furthermore, we adopt a realistic setting where the task and group delineations are unknown ahead of time. We state our main contributions as follows:

1) We propose the *Continual Reinforcement Learning with Group Symmetries* framework, which aims to improve continual RL's adaptivity via modeling task similarity based on group symmetries.
2) We introduce a novel PPO-based algorithm with a

rotation-invariant feature extractor. Such a flexible model architecture facilitates strong generalization capability to tasks with equivalent state space.

3) We propose a new unsupervised task grouping mechanism, which automatically detects group boundaries based on invariant features. It automatically grows policy when detecting a new group to handle streaming groups in the continual RL setting.
4) We test our proposed algorithm in robotic manipulation tasks with environments built on Meta-World benchmark[29]. We show that (a) the group symmetric information from the equivariant feature extractor promotes the algorithm's adaptivity by maximizing the positive interference within each task, and (b) the task grouping mechanism recovers the ground truth group indexes, which helps minimize the negative interference among different tasks.

II. Related Work

Geometric deep learning. Geometric deep learning aims to transfer the high performance of traditional deep learning to irregular input data such as graphs and meshes [2, 3, 16, 13, 6, 1]. For mesh inputs, spline-based Convolutional Neural Networks (CNNs) deal with geometric inputs by aggregating features purely in the spatial domain [7]. For image inputs, E(2)-equivariant convolution [25] introduces CNNs that are equivariant to rotation and reflection for planar images, which we use as the feature extractor in our neural network architecture design. Recent improvements on the Transformer architecture also focus on the equivariance to Lie group transformations [9, 10].

MDP homomorphism. MDP homomorphic networks [22] are neural networks that are equivariant under symmetries in the state-action space of an MDP. By exploiting the prior knowledge of symmetries, MDP homomorphic networks impose an equivariance constraint on the policy and value network. As a result, the solution space is reduced and the RL agent is able to achieve higher sample efficiency. This single-agent MDP homomorphic network is then extended to the multi-agent RL domain by factorizing the global symmetries into local symmetries in the presence of multiple cooperative agents [23]. SO(2)-Equivariant RL [24] extends the discrete symmetry group to the group of continuous planar rotations, SO(2), to boost the performance in a series of robotic manipulation tasks. In our work, we also exploit the symmetric properties of the states to improve learning efficiency and adaptability.

Continual RL in non-stationary environments. Continual RL, a.k.a. lifelong RL, aims to train RL agents that are adaptive to non-stationary environments with varying transition dynamics or reward functions [21, 20, 5, 28]. In hierarchical model-based RL, [18] infers a relationship between tasks automatically from the collected data and maintains a hierarchical latent variable model with Gaussian Process to model dynamics. [27] proposes a Dirichlet-Process-Gaussian-Process model to maintain a hierarchical structure of different dynamic models. Similarly, [17] uses the Hierarchical Dirichlet Process to model the abrupt changes in dynamics during an episode. Instead of keeping an explicit structure like the aforementioned works, [14] uses model-agnostic meta-learning to train a dynamics model prior such that it can adapt quickly to different environments. In model-free RL, Lifelong Latent Actor-Critic [26] leverages latent representation learning of environments using data from the replay buffer, which is then used in off-policy learning. Online System Identification (OIS) [30] is used to predict the dynamics model parameters, which are then fed into the controller along with system states to adapt to new environments. Different from the aforementioned works focusing on algorithms, Benna-Fusi-RL [11] equips the RL agent with a synaptic model that reduces catastrophic forgetting. In our work, instead of modeling dynamics or latent representation, we develop an ensemble model of policy networks to handle non-stationary and unseen driving scenarios.

III. Preliminary

Markov decision process. We consider a Markov decision process (MDP) as a 5-tuple $(\mathcal{S}, \mathcal{A}, T, R, \gamma)$, where $\mathcal{S}$ is a set of the states, $\mathcal{A}$ is a set of the actions, $T : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ is the transition probability, $R : \mathcal{S} \times \mathcal{A} \to R$ is the reward function, and γ is the discount factor.

The goal of RL is to find an optimal policy $\pi_\theta : \mathcal{S} \times \mathcal{A} \to R$ parameterized by θ that maximizes the expected return $\mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{H-1} \gamma^t r\left(s_t, a_t\right)\right]$, where H is the episode length.

Invariance and equivariance. Let G be a mathematical group. $f : \mathcal{X} \to \mathcal{Y}$ is a mapping function. For a transformation $L_g : \mathcal{X} \to \mathcal{X}$ that satisfies

$$f(x) = f(L_g[x]), \forall g \in G, x \in \mathcal{X}, \tag{1}$$

then we say f is invariant or symmetric to L_g. Equivariance is closely related to invariance. If we can find a second transformation $K_g : \mathcal{Y} \to \mathcal{Y}$ that fulfills

$$K_g[f(x)] = f(L_g[x]), \forall g \in G, x \in \mathcal{X}, \tag{2}$$

then we say f is equivariant to transformation L_g. It's worth noticing that invariance is a special case of equivariance.

MDP with group symmetries. Symmetries can be identified between MDPs. For the example in Figure 1, rotational symmetries can be identified between two MDPs. In MDP with symmetries, we can identify at least one mathematical group G of transformations $L_g : \mathcal{S} \to \mathcal{S}$ and a state-dependent action transformation $K_g^s : \mathcal{A} \to \mathcal{A}$, such that Equation (3) and (4) hold for all $g \in G, s, s' \in \mathcal{S}, a \in \mathcal{A}$. Formally,

$$R(s, a) = R\left(L_g[s], K_g^s[a]\right), \tag{3}$$

$$T\left(s, a, s'\right) = T\left(L_g[s], K_g^s[a], L_g\left[s'\right]\right). \tag{4}$$

Group-structured MDP Homomorphisms. For MDP with group symmetries, every state-action pair (s, a) can be mapped to an equivalent class $(\sigma(s), \alpha_s(a))$. This means all transformations of state-action pair (s, a) can be mapped to the same

abstract state-action pair. Formally, for all $g \in G, s \in \mathcal{S}, a \in \mathcal{A}$.

$$(\sigma(s), \alpha_s(a)) = \left(\sigma\left(L_g[s]\right), \alpha_{L_g[s]}\left(K_g^s[a]\right)\right). \quad (5)$$

The state-action pairs that are mapped to the same abstracted state share the same optimal Q-value and optimal value function. Thus we can optimize the policy $\bar{\pi}(\bar{a} \mid \sigma(s))$ under the abstracted MDP and then map the policy to the original MDP using a procedure called *lifting* [23].

$$\pi^{\uparrow}(a \mid s) \triangleq \frac{\bar{\pi}(\bar{a} \mid \sigma(s))}{\left|\left\{a \in \alpha_s^{-1}(\bar{a})\right\}\right|}, \quad \forall s \in \mathcal{S}, a \in \alpha_s^{-1}(\bar{a}). \quad (6)$$

Rotation-equivariant convolutional layer. Let G be a rotation group. An rotation-equivariant convolutional layer is a function $f : \rho_{\text{in}} \to \rho_{\text{out}}$ that is equivariant to rotation transformation in G. Here we use the most general form of an equivariant convolutional layer that preserves rotational equivariance [25]. The layer consists of G-steerable kernels $k : \mathbb{R}^2 \to \mathbb{R}^{c_{\text{in}} \times c_{\text{out}}}$ that satisfied Equation (7),

$$k(gx) = \rho_{\text{out}}(g) k(x) \rho_{\text{in}}\left(g^{-1}\right) \quad \forall g \in G, x \in \mathbb{R}^2. \quad (7)$$

A rotation-equivariant convolutional neural network can be constructed by stacking multiple rotational equivariant convolutional layers since the composition of equivariant maps is also equivariant.

IV. Continual RL with Group Symmetries

In this section, we proposed an online continual RL method with group symmetries. We follow the continual RL setting, assuming that a new task group is possible to emerge at every timestep. We further adopt a realistic setting where the total number of different groups is unknown. In the following subsections, we first introduce our proposed algorithm in Section IV-A. We then introduce two key components in our proposed algorithm: the equivariant policy network architecture in Section IV-B and the dynamic policy assignment mechanism in Section IV-D.

A. Algorithm Overview

Our algorithm learns a collection of policies Π where each policy $\pi \in \Pi$ handles one group of tasks independently. The algorithm begins by rolling out n episodes from the environment under the current policy. Based on the roll-out data, the algorithm proceeds to either (a) create a new policy for an unseen group and add it to the policy collection, or (b) recall an existing policy from the collection if the group has been previously encountered. To accomplish this, we select k observations at the beginning of each episode, putting them into a set denoted as $\mathcal{O}$. Observations of these frames are chosen since the are being less impacted by the policy, therefore more valuable for group identification.

For the ith policy, denoted as π_i, within the policy collection Π, it is associated with a state buffer $\mathcal{B}_i$ which stores the initial frames of the group it governs. The policy is also capable of extracting invariant latent features from the observations using its feature extractor h_i. Hence, we can obtain the latent features of the roll-out data, denoted as $q_{i,\mathcal{O}} = h_i(\mathcal{O})$, and the features of its observation buffer, denoted as $q_{i,\mathcal{B}_i} = h_i(\mathcal{B}_i)$. Subsequently, we can compute the Wasserstein distance between these two distributions, represented as $d_i = W(q_{i,S}, q_{i,S_i})$. A smaller d_i value suggests that the observations in $\mathcal{O}$ are more likely associated with policy π_i. By applying the same process to each policy in Π, we can compile a list of distances $d_1, d_2, \ldots, d_n$. If all distances surpass a predefined threshold d_ϵ, a new policy π_{n+1} will be created and appended to the policy collection Π. Alternatively, we will designate the current policy π as the one corresponding to the smallest d. Finally, we update the observation buffer $\mathcal{B}$ of the current policy $\mathcal{B} \leftarrow \mathcal{B} \cup \mathcal{O}$.

Algorithm 1 Continual RL with Group Symmetries

Input: Distance threshold d_ϵ, initial frame number k
Output: collection of policies Π
Initialization: $\Pi \leftarrow \{\emptyset\}$

```
while task not finish do
    O ← ∅
    for i = 1, 2, ..., N do
        j ← 0
        while episode not finish do
            a_t ~ π(s_t|a_t)
            s_{t+1} ~ p(s_{t+1}|s_t, a_t)
            if j < k then
                O ← O ∪ {s(t)}
            end if
            j ← j + 1
        end while
    end for
    D ← ∅
    for all π_i ∈ Π do
        q_{i,O} = f_i(O)
        q_{i,B_i} = f_i(B_i)
        d_i = W(q_{i,O}, q_{i,B_i})
        D ← D ∪ {d_i}
    end for
    j ← argmin D
    if d_j > d_ε then
        Initialize a new policy π_θ with observation buffer B
        π ← π_θ
        Π ← Π ∪ {π_θ}
    else
        π ← π_j
        B ← B_i
    end if
    Update π to maximize expected return
    B ← B ∪ O
end while
```

B. Policy Network Architecture

We adapted Proximal Policy Optimization (PPO) [19] as the RL algorithm for each policy in our collection. PPO comprises

two networks: a policy network that produces actions, and a value network that approximates values.

For MDP with group symmetries, the optimal value function is group invariant, and the optimal policy is group equivariant [24]. To achieve this, the policy and the value networks employ a shared equivariant feature extractor to extract equivariant features from observations. As illustrated in Figure 2, in the value network, these extracted features are transformed into invariant features through a group pooling layer, and then fed forward into a fully connected network to generate the output values. On the other hand, in the policy network, these features are input into an equivariant network composed of multiple equivariant MLP layers.

C. Equivariant feature extractor.

The equivariant feature extractor is comprised of 2 parts: an equivariant convolutional network that accepts images as inputs and an equivariant fully connected network that accepts vectors. The outputs from these 2 networks are concatenated, resulting in the final equivariant features.

The equivariant convolutional network is constituted of 5 equivariant convolutional layers [25], which convert an image from a trivial representation into a regular representation. The equivariant fully connected network, on the other hand, consists of a single equivariant fully connected layer [4] that transforms a vector, which combines different representations, into a regular representation.

D. Unsupervised Dynamic Policy Assignment

Detecting task boundaries and developing skills to solve each task are still open problems. Existing methods detect task boundaries based on model prediction errors in model-based RL settings [27], task performance drops in model-free RL settings, or policy reconstruction errors in imitation learning. In this work, we propose to detect different groups of tasks instead to facilitate knowledge transfer between tasks in each group and discourage negative policy interferences across different groups.

Assignments based on invariant features. In contrast to prior works that calculate the distance in state space, our method determines policy assignments based on distance in the invariant feature space. Specifically, we compute the Wasserstein distance between the invariant features derived from $\mathcal{O}$ and buffer $\mathcal{B}$ of each policy in the collection. The invariant features are obtained from the equivariant feature extractor via a group pooling operation.

Let $\mathbf{X}$ be a matrix constructed by invariant features extracted from the state buffer $\mathcal{B}$ of size n.

$$\begin{aligned} \mathbf{X} &= (X_1, X_2, ..., X_n)^{\mathrm{T}}, \\ X_i &= h(s_i), i \in [n], s_i \in \mathcal{B}. \end{aligned} \tag{8}$$

We use the Earth Movers distance (EMD) to measure the distance between two empirical distributions $\mathbf{X}$ and $\mathbf{Y}$ with n and m features, respectively. More specifically,

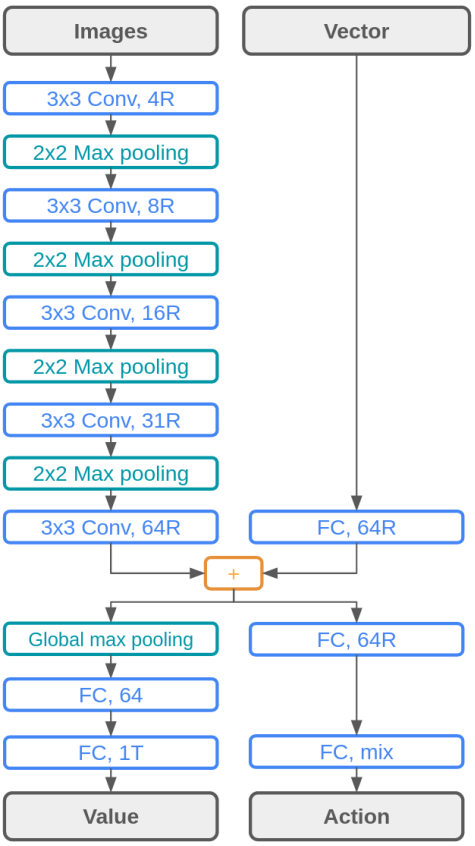

Fig. 2: Equivariant policy network architecture, ReLU non-linearity is omitted in the figure. A layer with a suffix of R indicates the layer output is in the regular representation; a layer with a suffix of T indicates the layer output is in the trivial representation; a layer with a siffix of 'mix' meaning the layer output combines different representations.

$$\begin{aligned} EMD(\mathbf{X},\mathbf{Y}) = \min_{\gamma} \quad & \langle \gamma, \mathbf{M} \rangle_F \\ \text{s.t. } \gamma \mathbf{1} = \mathbf{a}, \gamma^T \mathbf{1} &= \mathbf{b}, \gamma \geq 0 \end{aligned} \tag{9}$$

where $\mathbf{M}$ is the metric cost matrix and $\mathbf{M}_{i,j} = \|X_i - Y_j\|_2$, $\mathbf{a} = [1/n, 1/n, \ldots, 1/n]$, $\mathbf{b} = [1/m, 1/m, \ldots, 1/m]$.

V. Experimental Setup

In this section, we tested the proposed method in Meta-World benchmark [29] and evaluate its performance. We want to investigate whether our method can (i) assign or expand correct task group indices, (ii) recall policy when facing an old environment and automatically initialize a new policy when encountering an unseen environment (iii) achieve equivalent or better performance compared to baseline methods.

A. Baselines

We compare our method termed as *M-Equi* with 3 baselines detailed as follows:

1) **Multiple policies with ground truth policy indexes (M-Equi-gt)**: M-Equi-gt uses ground truth group labels

to assign policies to different groups. With this additional group index information, M-Equi-gt is expected to have the oracle performance. M-Equi-gt helps ablate the performance of our proposed policy assignment mechanism.

2) **Multiple policies with a CNN-based feature extractor (M-CNN)**: M-CNN uses the same continual learning mechanism but with a vanilla CNN block as the feature extractor. The vanilla CNN shares a similar network structure compared with our proposed invariant feature extractor in Figure 2. The only difference is switching the rotation-equivariant convolutional layer to the vanilla convolutional layer. M-CNN helps reveal the benefit of using invariant feature extractors.
3) **A single policy with a CNN-based feature extractor (S-CNN)**: S-CNN is similar to S-Equi but uses a CNN-based feature extractor.

B. Environment

Simulation Setup. Our robot manipulation scenarios consist of 4 group of tasks, and each group contains 4 tasks, rotations and reflections symmetries can be found between any 2 tasks within the same group. The robot is required to complete a sequence of groups, adhering to the standard RL setting. Here we utilized the Meta-World benchmark [29] as our simulation platform. Meta-World features a variety of table-top manipulation tasks that require interaction with diverse objects using a Sawyer robot. We chose 4 group of tasks as shown in Figure 6 from the Meta-World benchmark for our study: 'Reach', 'Button Press', 'Drawer Close', and 'Plate Slide'.

1) **Reaching.** In the 'Reaching' group, the robot is required to move its gripper towards a selected target above the table. The goal location is symmetrically arranged around the center of the table.
2) **Button Press.** In the 'Button Press' group, the robot is required to press a button located on the table. The button location is symmetrically arranged around the center of the table.
3) **Drawer Close.** In the 'Drawer Close' group, the robot is required to close a drawer using its gripper. The location of the drawer is symmetrically arranged around the center of the table.
4) **Plate Slide.** In the 'Plate Slide' group, the robot is required to move a plate to the target location using its gripper. The goal location is symmetrically arranged around the center of the table.

Real World Setup. Our real-world experiment setup utilized a Kinova GEN3 robotic arm with is equipped with a Robotiq 2F-85 gripper. The top-down RGB image is captured with an Intel RealSense D345f. Information on the gripper's coordinates and opening angle were obtained through the robot's internal sensors. The real robot setups are demonstrated in Figure 7.

C. States and Actions

The agent receives two types of observations: an RGB image captured from a top-down camera centered over the table, and the gripper's 3D coordinates and opening angle. Notably, the target location is only revealed to the agent during the 'Reaching' task. To close the gap between the simulation and real-world scenarios (Sim2Real), we prepossessed the RGB images before inputting them into the network. A comparison of the original and processed images can be found in Figure 8.

The action is a four-dimensional vector that controls the gripper's movement along three axes and its opening angle. This particular type of action has been chosen to ensure the transferability of our algorithm between different robotic models, as we utilized two distinct robots in the simulation and real-world scenarios. This choice also makes our approach more independent of the specific robot arm model and broadly applicable to various scenarios.

VI. Results and Ablations

A. Results

In this section, we analyze the performance of our method against other baselines during training and evaluate their converged performance in each group.

1) Training performance: As shown in Figure 5, when the environment switches to a new group, our method quickly detects changes and initialize different policy for each group. Our method also recalls the corresponding policy from the collection when facing the same group again. In general, our method's dynamic policy assignment result matches the ground truth group labels. We noticed that some assignment results did not match the ground truth. This may be due to the fact that the feature extractor of each sub-policy is not able to detect representative features of each group at the early stage. The misclassification rate drops significantly as the training episode grows.

In Figure 4, our method's training curve is similar to M-Equi-gt which assigns policies according to the ground truth group index. The training curve shows that the performance drop due to misclassification is minor and acceptable. Comparing the training curve of baselines using vanilla CNN, the equivariance networks dramatically improve their sample efficiency. This is because the equivariance network only needs to optimize the policy for the abstracted MDP in the group so that it does not need to handle each task in the group independently. Additionally, the training curve does not differ much between our method and S-Equi at the early stage of training. But later, the training curve of S-Equi could not keep up with our method since it suffers from the forgetting problem due to not retaining parameters of previously seen groups.

2) Converged performance: Table I and Table II show the converged performance of our method compared with other baseline methods. The performance of our method is similar to M-Equi-gt, which uses ground truth group policy indexes as extra prior knowledge. Besides M-Equi-gt, our method achieves a much higher episode reward and a much higher

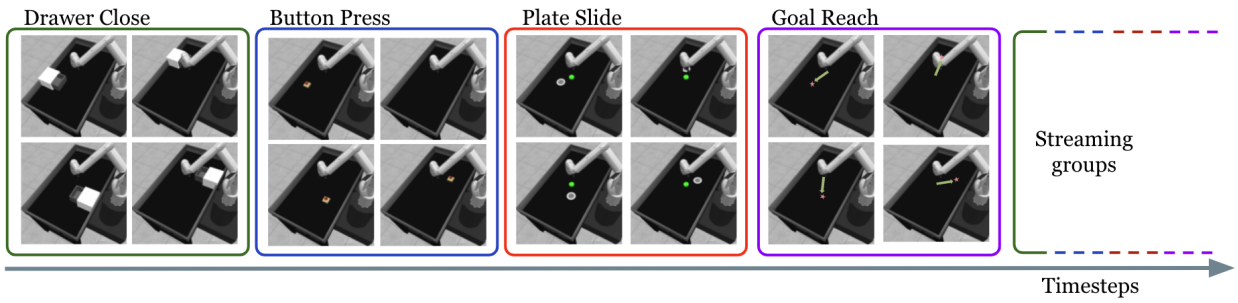

Fig. 3: The continual learning environment where there are four groups of tasks based on task configurations, including the drawer-close, button-press, plate-slide, and goal-reach. Different groups streamingly come in.

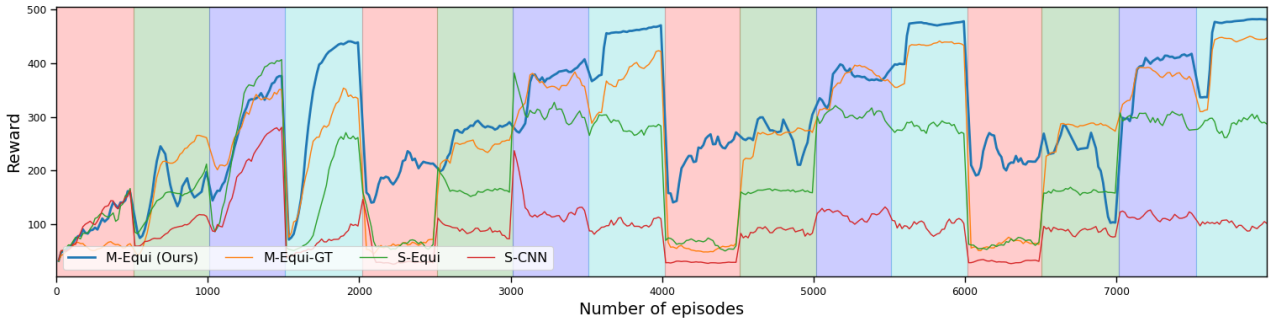

Fig. 4: Training curves for our proposed **M-Equi** approach and the baseline models are illustrated, with each background color corresponding to one group of tasks. The results demonstrate that **M-Equi** shows similar performance when compared to M-Equi-gt, which utilizes ground truth group indices, and substantially outperforms other baseline models.

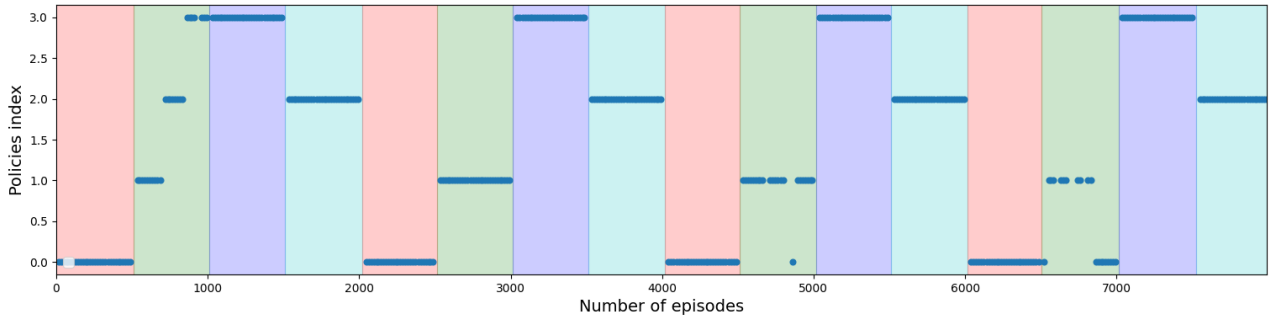

Fig. 5: The selection of policies at each episode for our proposed **M-Equi** method and the baseline models is represented, with each background color denoting a distinct group of tasks. Generally, following the initial stage, the policy indices allocated by **M-Equi** remain in alignment with the ground truth group indices.

success rate compared to other baselines. The results show that our method is able to achieve high performance across all groups.

B. Ablation Study

Here we analyze the effect of multiple policies and invariant feature extractors respectively through ablation studies.

1) The effect of group symmetric information: As shown in Table I, methods without the invariant feature extractor, such as S-CNN, have lower episodic rewards, as well as lower success rate, compared with methods using the invariant feature extractors (M-Equi, M-Equi-gt, S-Equi). Hence, we can conclude that group symmetric information extracted by the invariant feature extractor dramatically improves the performance. The invariant feature extractor maps all tasks in the same group into an abstracted task. Thus the RL agent can optimize policies under the abstracted MDP.

2) The effect of the dynamic policy assignment module: By comparing the performances between M-Equi, M-Equi-gt, and S-Equi, we can conclude that using multiple policies in the continuous learning environment helps improve performance. As shown in Figure 4, S-Equi's performance does not differ much from M-Equi and M-Equi-gt at the early stage of training. However, when the environment switches to different groups, S-Equi's performance drops quickly due to the forgetting problem since it can not recall parameters from seen groups.

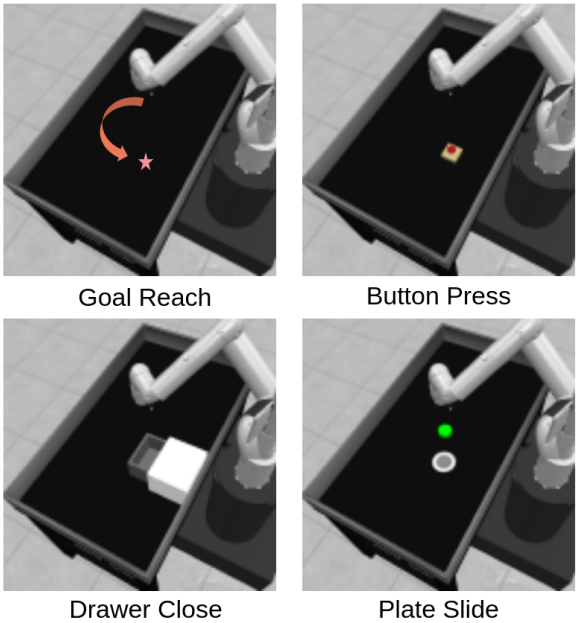

Fig. 6: The simulated Sawyer setup consists of four different tasks, including goal-reach, button-press, drawer-close, and plate-slide. The goal point marked in the figure is only disclosed to the agent in goal-reach-related tasks.

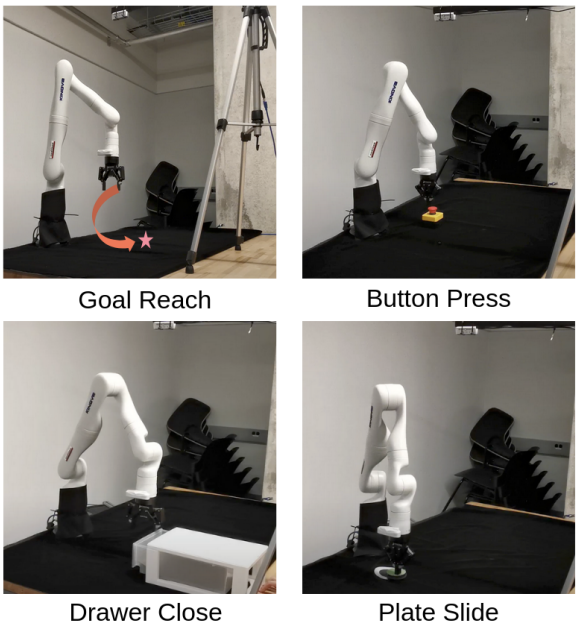

Fig. 7: The real Kinova GEN3 setup consists of four different tasks, including goal-reach, button-press, drawer-close, and plate slide. The goal point marked in the figure is only disclosed to the agent in goal-reach-related tasks.

VII. Conclusion

We propose a novel continual RL framework that leverages group symmetries to facilitate quick generalization to unseen but equivalent tasks under group operations. Our proposed algorithm unsupervisedly detects group boundaries based on invariant state features and grows policies for each group of equivalent tasks instead of a single task. By implementing our algorithm on robotic manipulation scenarios, we show that our algorithm assigns tasks to different groups with high accuracy and has a strong generalization capability outperforming baselines by a large margin. Finally we transformed the trained algorithm from simulator to a real-world robotic arm without fine-tuning. The results indicate that the agent is capable of adeptly solving different tasks.

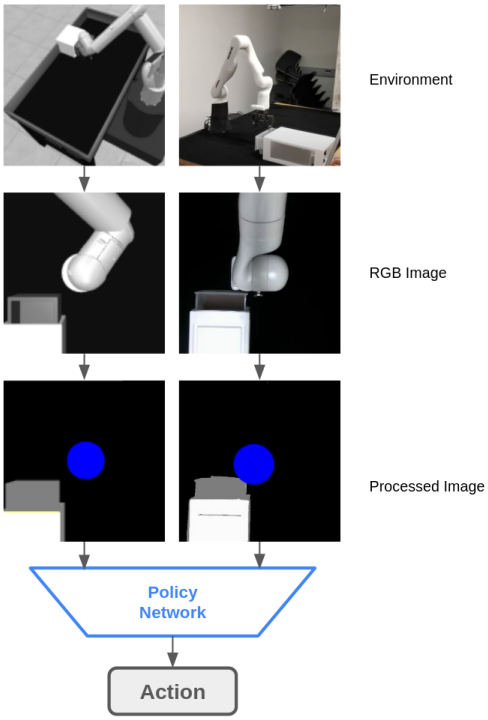

Fig. 8: To address the sim-to-real gap, we incorporate preprocessing steps for both real and simulated observation images. These steps are performed before feeding the images into the network, as illustrated in the figure.

TABLE I: Quantitative evaluation of episodic reward of policies at convergence.

Methods	Plate-slide	Button-press	Drawer-close	Goal-reach
M-Equi (Ours)	**263.24**	271.26	**411.83**	**478.56**
M-Equi-GT	72.81	**284.47**	389.46	443.26
S-Equi	58.78	163.92	304.61	292.39
S-CNN	25.60	85.35	121.23	102.21

TABLE II: Quantitative evaluation of episodic success rate of policies at convergence.

Methods	Plate-slide	Button-press	Drawer-close	Goal-reach
M-Equi (Ours)	**0.83**	**0.89**	0.96	**0.99**
M-Equi-GT	0.12	0.31	**0.97**	**0.99**
S-Equi	0.17	0.29	0.69	0.70
S-CNN	0.02	0.15	0.13	0.12

Acknowledgment

The authors gratefully acknowledge the support from the unrestricted research grant from Toyota Motor North America. The ideas, opinions, and conclusions presented in this paper are solely those of the authors.

References

[1] Davide Boscaini, Jonathan Masci, Simone Melzi, Michael M Bronstein, Umberto Castellani, and Pierre Vandergheynst. Learning class-specific descriptors for deformable shapes using localized spectral convolutional networks. In *Computer graphics forum*, volume 34, pages 13–23. Wiley Online Library, 2015.

[2] Michael M Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst. Geometric deep learning: going beyond euclidean data. *IEEE Signal Processing Magazine*, 34(4):18–42, 2017.

[3] Michael M Bronstein, Joan Bruna, Taco Cohen, and Petar Veličković. Geometric deep learning: Grids, groups, graphs, geodesics, and gauges. *arXiv preprint arXiv:2104.13478*, 2021.

[4] Gabriele Cesa, Leon Lang, and Maurice Weiler. A program to build e(n)-equivariant steerable CNNs. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=WE4qe9xlnQw.

[5] Zhiyuan Chen and Bing Liu. Lifelong machine learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 12(3):1–207, 2018.

[6] Joan Bruna Estrach, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. Spectral networks and deep locally connected networks on graphs. In *2nd International conference on learning representations, ICLR*, volume 2014, 2014.

[7] Matthias Fey, Jan Eric Lenssen, Frank Weichert, and Heinrich Müller. Splinecnn: Fast geometric deep learning with continuous b-spline kernels. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 869–877, 2018.

[8] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*, pages 1126–1135. PMLR, 2017.

[9] Fabian Fuchs, Daniel Worrall, Volker Fischer, and Max Welling. Se (3)-transformers: 3d roto-translation equivariant attention networks. *Advances in Neural Information Processing Systems*, 33:1970–1981, 2020.

[10] Michael J Hutchinson, Charline Le Lan, Sheheryar Zaidi, Emilien Dupont, Yee Whye Teh, and Hyunjik Kim. Lietransformer: Equivariant self-attention for lie groups. In *International Conference on Machine Learning*, pages 4533–4543. PMLR, 2021.

[11] Christos Kaplanis, Murray Shanahan, and Claudia Clopath. Continual reinforcement learning with complex synapses. In *International Conference on Machine Learning*, pages 2497–2506. PMLR, 2018.

[12] Khimya Khetarpal, Matthew Riemer, Irina Rish, and Doina Precup. Towards continual reinforcement learning: A review and perspectives. *arXiv preprint arXiv:2012.13490*, 2020.

[13] Jonathan Masci, Davide Boscaini, Michael Bronstein, and Pierre Vandergheynst. Geodesic convolutional neural networks on riemannian manifolds. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 37–45, 2015.

[14] Anusha Nagabandi, Ignasi Clavera, Simin Liu, Ronald S Fearing, Pieter Abbeel, Sergey Levine, and Chelsea Finn. Learning to adapt in dynamic, real-world environments through meta-reinforcement learning. *arXiv preprint arXiv:1803.11347*, 2018.

[15] Anusha Nagabandi, Chelsea Finn, and Sergey Levine. Deep online learning via meta-learning: Continual adaptation for model-based rl. *arXiv preprint arXiv:1812.07671*, 2018.

[16] Adrien Poulenard and Maks Ovsjanikov. Multi-directional geodesic neural networks via equivariant convolution. *ACM Transactions on Graphics (TOG)*, 37(6): 1–14, 2018.

[17] Hang Ren, Aivar Sootla, Taher Jafferjee, Junxiao Shen, Jun Wang, and Haitham Bou-Ammar. Reinforcement learning in presence of discrete markovian context evolution. *arXiv preprint arXiv:2202.06557*, 2022.

[18] Steindór Sæmundsson, Katja Hofmann, and Marc Peter Deisenroth. Meta reinforcement learning with latent variable gaussian processes. *arXiv preprint arXiv:1803.07551*, 2018.

[19] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[20] Fumihide Tanaka and Masayuki Yamamura. An approach to lifelong reinforcement learning through multiple environments. In *6th European Workshop on Learning Robots*, pages 93–99, 1997.

[21] Sebastian Thrun and Tom M Mitchell. Lifelong robot learning. *Robotics and autonomous systems*, 15(1-2):25–46, 1995.

[22] Elise van der Pol, Daniel Worrall, Herke van Hoof, Frans Oliehoek, and Max Welling. Mdp homomorphic networks: Group symmetries in reinforcement learning. *Advances in Neural Information Processing Systems*, 33: 4199–4210, 2020.

[23] Elise van der Pol, Herke van Hoof, Frans A Oliehoek, and Max Welling. Multi-agent mdp homomorphic networks. *arXiv preprint arXiv:2110.04495*, 2021.

[24] Dian Wang, Robin Walters, and Robert Platt. So (2) equivariant reinforcement learning. In *International conference on learning representations (ICLR)*, 2022.

[25] Maurice Weiler and Gabriele Cesa. General e (2)-equivariant steerable cnns. *Advances in Neural Information Processing Systems*, 32, 2019.

[26] Annie Xie, James Harrison, and Chelsea Finn. Deep reinforcement learning amidst continual structured non-stationarity. In *International Conference on Machine Learning*, pages 11393–11403. PMLR, 2021.

[27] Mengdi Xu, Wenhao Ding, Jiacheng Zhu, Zuxin Liu, Baiming Chen, and Ding Zhao. Task-agnostic online reinforcement learning with an infinite mixture of gaussian processes. *Advances in Neural Information Processing*

Systems, 33:6429–6440, 2020.

[28] Mengdi Xu, Zuxin Liu, Peide Huang, Wenhao Ding, Zhepeng Cen, Bo Li, and Ding Zhao. Trustworthy reinforcement learning against intrinsic vulnerabilities: Robustness, safety, and generalizability. *arXiv preprint arXiv:2209.08025*, 2022.

[29] Tianhe Yu, Deirdre Quillen, Zhanpeng He, Ryan Julian, Karol Hausman, Chelsea Finn, and Sergey Levine. Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning. In *Conference on Robot Learning*, pages 1094–1100. PMLR, 2020.

[30] Wenhao Yu, Jie Tan, C Karen Liu, and Greg Turk. Preparing for the unknown: Learning a universal policy with online system identification. *arXiv preprint arXiv:1702.02453*, 2017.

[31] Wenshuai Zhao, Jorge Peña Queralta, and Tomi Westerlund. Sim-to-real transfer in deep reinforcement learning for robotics: a survey. In *2020 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 737–744. IEEE, 2020.