Minimally Invasive Morphology Adaptation via Parameter Efficient Fine-Tuning

Michael Przystupa[1*] Hongyao Tang[2] Mariano Phielipp[3]
Santiago Miret[3] Martin Jagersand[1] Glen Berseth[2,4]

***Abstract*— Learning reinforcement learning policies to control individual robots is often computationally non-economical because minor variations in robot morphology (e.g. dynamics or number of limbs) can negatively impact policy performance. This limitation has motivated morphology agnostic policy learning, in which a monolithic deep learning policy learns to generalize between robotic morphologies. Unfortunately, these policies still have sub-optimal zero-shot performance compared to end-to-end finetuning on target morphologies. This limitation has ramifications in practical robotic applications, as online finetuning large neural networks can require immense computation. In this work, we investigate *parameter efficient finetuning* techniques to specialize morphology-agnostic policies to a target robot that minimizes the number of learnable parameters adapted during online learning. We compare direct finetuning, which update subsets of the base model parameters, and input-learnable approaches, which add additional parameters to manipulate inputs passed to the base model. Our analysis concludes that tuning relatively few parameters (0.01% of the base model) can measurably improve policy performance over zero shot. These results serve a prescriptive purpose for future research for which scenarios certain PEFT approaches are best suited for adapting policy's to new robotic morphologies.**

I. Introduction

Applying deep reinforcement learning (DRL) to robotics is often challenging because of it's brittleness to small variations in the task. Even on the same class of robot, small deviations in dynamics and kinematics can affect policy performance [1], [2]. Data re-use also becomes difficult because policies trained on specific robot do not easily transfer to other robots [3]. This paper aims to investigate effective means of re-using previously trained policies that can adapt to robot variations using parameter efficient learning techniques. Our work differs from other generalization research in reinforcement learning, such as goal-conditioning [4], [5], [6] or meta-learning [7], [8], [9], because we focus on the generalizing across different robots to a single task as opposed to a single robot that perform many tasks.

One solution to these problems is learning policies on invariant spaces that generalize across robot designs without needing explicit design information. Cartesian control on robotic manipulators, for example, has enabled large-scale data collection efforts for imitation learning [10], [11], [12], as well as enabled the generalization of reinforcement learning policies between multiple robot arms [13], [14]. This control approach relies on the robot's internal inverse kinematic's controllers finding solutions for the desired joint configuration to reach target poses [15]. Unfortunately, Cartesian control has limits due to local minima regions (e.g. singularities [15]) and does not transfer easily to controlling other robotic morphologies, such as quadrupeds.

Fortunately, particularly for locomotion tasks, an alternative research direction is learning morphology agnostic policies that directly control the robot limbs. These approaches utilize graph representations of morphology to allow the policy generalization over distributions of robot's by processing the policies either with Graph Neural Networks [16] or Transformers [17]. Author's have investigated a variety factors in morphology agnostic policies including the efficacy of morphology generalization [3], [18], [19], tokenizing graph representations [20], [21], evaluting simulated robotic designs [22], [23], as well more effectively defining inductive biases directly in the neural network [24], [25], [26], [27]. These methods enable more control to the reinforcement learning policy by directly sending commands to the limbs.

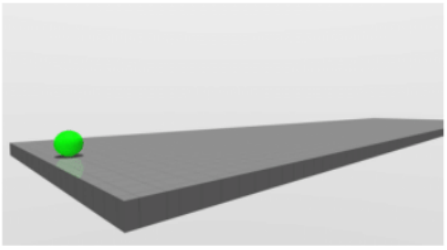
However, existing morphology agnostic policy learning research has largely ignored the computational constraints when using robotics in the real world. Often, robot's can have limited onboard computers meaning it's necessary to have efficient control algorithms on the hardware [28], [29]. Hardware constraints makes incorporating deep learning difficult, especially as many neural network models employ Foundation models (tens of millions to billions of parameters) trained on internet-scale text or image data sets [30], [31], [32]. Even in these cases, it is still necessary to adapt the policy for the target robotic task as otherwise the policies often under perform at deployment [33], [34], [13], [35]. These computation and performance requirements presents an interesting challenge as large neural networks help policies transfer, but become difficult to continuously update on a target robot.

We propose using parameter efficient fine-tuning (PEFT) algorithms offer solutions to address both these challenges. PEFT algorithms use subsets of a model's parameters to finetune a pre-trained neural network or otherwise introduce a small set of new learning parameters that specialize for a target task [36], [37]. The latter approach comes with more flexibility because approaches can be input-learnable parameters that do not require direct access to the pre-trained model [38]. Researcher's have shown that PEFT methods work well on large networks in natural language [39], and in computer vision problems [40] while reducing computation costs to update the PEFT parameters. Closely related to our work is the work of Liu et. al [35] who investigates PEFT

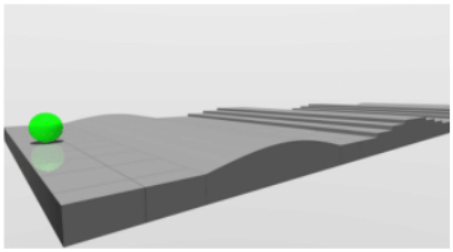
[1]University of Alberta, *przystup@ualberta.ca
[2]Mila, Université de Montréal [3] Intel Labs, [4] CIFAR Canada AI Chair

method's in robotics for continual imitation learning. Our work is different as we deal with *morphology transfer* and evaluate PEFT methods with deep reinforcement learning which presents different challenges from supervised learning.

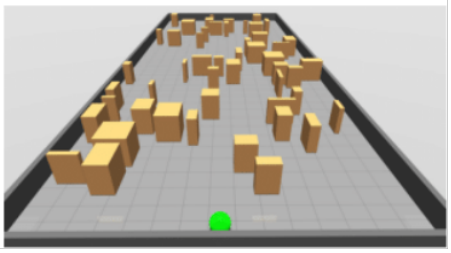
In summary, the primary contribution of our work is the analysis of a number of PEFT techniques for morphology transfer. Our results demonstrate that it is generally achievable to substantially reduce the total parameter used and achieve statistically measurable improvement over zero-shot performance, even with strong initial zero-shot performance. Using even 1% total learnable parameters relative to the base model's total parameter count is beneficial. As part of our work, we propose two modifications to input-learnable PEFT algorithms that address preserving zero-shot performance in online reinforcement learning. This research is highly relevant for the field of robotics learning, as our results provide a guidelines that researchers can use to determine which PEFT techniques are appropriate for their computation and model usage settings.



(a) Flat Terrain



(b) Variable Terrain



(c) Obstacle Avoidance

Fig. 1: Locomotion environments considered in experiments. Diagrams are reproduced from Gupta et. al [19]

.

II. Background

A. Contextual Markov Decision Process

Morphology agnostic policy learning can be understood as a form of contextual Markov Decision Process (CMDP) [41]. A CMDP is characterized by a distribution $\mathcal{C}$ where for $c \sim \mathcal{P}(C)$ we have an induced tuple $M(c) = (\mathcal{S}^c, \mathcal{A}^c, \mathcal{P}^c, r, p^c(s_0))$. For each c, $\mathcal{S}^c$ is a finite set of states, $p^c(s_0)$ represents the initial state distribution and $\mathcal{A}^c$ is a finite set of actions. The state transition probability function, $\mathcal{P}^c(s'|s,a) = \Pr(s_{t+1} = s' \mid s_t = s, a_t = a; c)$, defines the probability of transitioning from state s to state s' when action a occurs. The reward function, $r^c(s,a,s')$, represents the immediate value of transitioning from s to s' due to a. A policy $\pi : \mathcal{S} \times \mathcal{C} \to \mathcal{P}(\mathcal{A})$ is a mapping from states and contexts to a probability distribution over actions, where π samples actions $a \sim \pi(s,c)$ to transition following $\mathcal{P}^c(s'|s,a)$. The goal of a CMDP is to maximize the expected sum of rewards over the distribution of contexts,

$$\pi^\star(s,c) = \arg\max_{\pi \in \Pi} \mathbb{E}_{p(c)}[G_c],$$

where $G_c = \mathbb{E}_{p^c(\tau)}[\sum_{t=0}^{T} \gamma^t r(s_t, a_t)]$ is the expected cumulative reward for a given context with discount factor $\gamma \in [0,1]$, time horizon T, and $p^c(\tau) = p^c(s_0) \prod_{t=0}^{T} \pi(s_t, c) \mathcal{P}^c(s_{t+1}|s_t, a_t)$ is the distribution over trajectories in the environment.

In our work, c refers to morphology information about the agent. The morphology variables affect the dimension of the state and action spaces, $a \in \mathbb{R}^{2n(c)}$ or $s \in \mathbb{R}^{n(c) \times d}$ where $n(c)$ are the number of limbs in the morphology, and the actions in our experiments include desired joint angles and velocity. Our context variables use robot morphology information per limb as described in Appendix A.1, Gupta et al. [19]. We further note that our experiments' reward functions r^c are *independent* of morphology.

B. Transformers

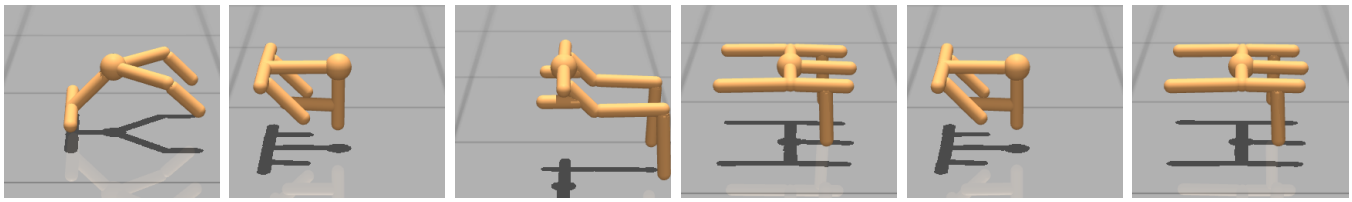
An essential component of the morphology agnostic policies used in this work are Transformer models [17]. We assume that the observation sequence $o \in \mathbb{R}^{n \times d}$ is projected by some linear function to an embedding space $\bar{o} = oW^{embed} + W^{position}$, where $W^{embed}, W^{position} \in \mathbb{R}^{n \times h}$. The major components of transformers are the *self-attention* mechanism and LayerNorm function (LN) [42]. The self-attention mechanism generate a weighted combination of the sequence for each embedding $\bar{o}_i$,

$$f^i(o) = \text{softmax}(\epsilon Q K^T) V,$$

where we call $Q = \bar{o}W^{Qi}$, $V = \bar{o}W^{Vi}$, $K = \bar{o}W^{Ki}$ the query, key and value respectively, $\bar{o} = \text{LN}(o)$, and $\epsilon = 1/\sqrt{h}$. The parameter set $W^{attn} = \{W^{Qi}, W^{Vi}, W^{Ki}\} \in \{\mathbb{R}^{d \times h}, \mathbb{R}^{d \times h}, \mathbb{R}^{d \times h}\}$ are learned linear projections. In practice, a multi-headed variation of self-attention $f(o) = [f^1(o); f^2(o); ...; f^n]W^{out}$ is used, each with their own weights W^{attn_i} and the outputs are aggregated against linear transform $W^{out} \in \mathbb{R}^{(nh) \times h}$.

After the attention layers, a residual connection between $f(o)$ and o is passed to a nonlinear model $g(x, f(o)) =$

$$W^{out}\sigma(W^{in}(\text{LN}(x + f(x))) + \text{LN}(f(x)) + x,$$



Morphology 1 Morphology 2 Morphology 3 Morphology 4 Morphology 5 Morphology 6

Fig. 2: The six testing morphologies used in our evaluation. For morphologies with similar visual embodiments, they had different dynamic and kinematic values. Morphology numbers correspond to those shown in relevant results.

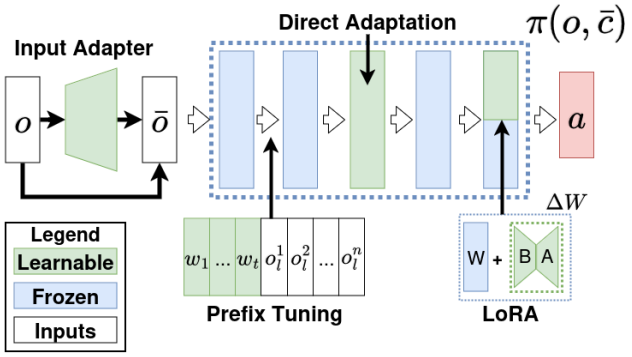


Fig. 3: Different PEFT techniques used in this analysis. We consider both directly modifying the networks as well as learnable input approaches for adaptation.

where $W^{out}, W^{in} \in \mathbb{R}^{h \times h}$. We exclude bias terms and note that σ is ReLU in our experiments. We refer to a Transformer layer as $T_i(o) = g_i(o, f_i(o))$.

III. Minimally Invasive Morphology Adaptation

This section discusses our work investigating the efficacy of PEFT algorithms for morphology-agnostic online reinforcement learning. We assume access to a trained policy $\pi_{\theta^\star}(s, c)$ with optimal parameter set $\theta^\star$ over a morphology distribution. For a new morphology $\bar{c} \sim \mathcal{P}(C)$, we learn an optimal set of parameters $\phi^\star$,

$$\phi^* = \arg\max_{\phi} \mathbb{E}_{\pi_{\theta^\star \cup \phi}(s,\bar{c})}[G_{\bar{c}}(s)].$$

We hypothesize that learning a small set ϕ will perform measurably better than the base policy's zero-shot performance, $E_{\pi_{\theta^\star \cup \phi^\star}(s,\bar{c})}[G_{\bar{c}}(s)] > E_{\pi_{\theta^\star}(s,\bar{c})}[G_{\bar{c}}(s)]$ where $|\phi| << |\theta|$. DRL policies require immense computation to learn and physical resources to collect data. Learnable policies agnostic over robot morphology are thus practical because they enable data usage between robots for learning and maximize the policy's applications for real-world deployment.

Unfortunately, a generalist policy may not elicit the optimal performance of a target robot due to these generalization capabilities. For real robotic applications, it is likely necessary that base model components continue to learn to maximize task performance. Reducing the total necessary learnable parameters is thus significant to achieving this result because, at deployment, it may not be feasible to access sufficient computation resources to perform learning updates. These limitations motivate the potential of PEFT solutions, which are applicable in varying resource limitations when deploying these policies. The rest of this section discusses the framework used to learn the base policy and the PEFT algorithms we investigate, which are visualized in Figure 3.

TABLE I: Layer tuning parameters and experiment identifiers

Layer Tuned	Parameters ϕ	Exp. Identifier
End-to-end	θ^*	E2E
Transformer Layers	$\{T_i; i \in [1, L]\}$	Layer 5
Attention layers	$\{W_i^{\text{attn}}; i \in [1, L]\}$	Lora
Nonlinear transformers	$\{W_i^{\text{in}}, W_i^{\text{out}}; i \in [1, L]\}$	Lora
Input Embedding	$\{W^{\text{embed}}, W^{\text{position}}\}$	Embedding
Decoder	$\{W_i^{\text{decoder}}; i \in [1, L^{\text{dec}}]\}$	Decoder

A. Pre-trained Models

We conduct experiments using policies trained with the *Metamorph* framework [19]. Morphologies are represented as graphs but treated as sequences $o = [o_1, o_2, o_3, ..., o_n]$ where $o_i \in \mathbb{R}^d$ contains local joint information per limb. Transformer models (Section II) process the sequences into latent representations. A multi-layer perceptron, the decoder, then predicts actions per limb $d_\theta(T(o)) = a$.

The policy $\pi_\theta(s)$ is optimized over an empirical distribution of morphologies $\mathcal{P}(\hat{C})$ using Proximal Policy Optimization (PPO) [43], with the loss function: $L^{CLIP}(\theta) =$

$$-\mathbb{E}_{\mathcal{P}(\hat{C})p^c(\tau)}[\min(r_t(c,\theta)\hat{A}_t, \text{clip}(r_t(c,\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t)],$$

where $r_t(c,\theta) = \frac{\pi_\theta(a_t|s_t,c)}{\pi_{\theta_{old}}(a_t|s_t,c)}$ is the ratio of new to old policy probabilities, $\hat{A}_t$ is the estimated advantage, and ϵ is the clipping hyperparameter.

B. Parameter Efficient Finetuning Across Morphologies

We group PEFT approaches as either direct or input-level adaptation techniques. *Direct adaptive* PEFT approaches modify some subset of the weights $\phi \subseteq \theta^\star$ or else a add learnable Delta weights $\hat{W} = W + \Delta W$. *Input adaptive* PEFT approaches perform some transformation of the inputs to elicit the optimal performance in the model.

In our evaluations, we consider tuning subsets of $\theta^\star$ for direct adaptive PEFT learning, which we itemize in Table I to summarize the configurations we consider and their identifier in experimental results. *Layer 5* represents directly tuning the final Transformer layer to compliment observations for prefix tuning results. For Attention and Nonlinear transform layers we used Low-Rank Adaptation (LoRA) [44], to learn $\Delta W \in \mathbb{R}^{h^1 \times h^2} = AB$, where $A \in \mathbb{R}^{h^1 \times r}$ and $B \in \mathbb{R}^{r \times h^2}$ are low-rank matrices of rank r to reduce learnable weights. In our experiments, we initialize A to zero and B to small Gaussian noise $b_{ij} \sim N(0, 1e^{-4})$ to preserve the initial model performance. When discussing aggregated results, we group these two under the *LoRA* approach and discuss differences in our ablation experiments.

For input adaptive PEFT approaches, we consider both prefix fine-tuning and learning an extra input adapter layer. We consider an input adapter layer that modifies the policy observation as $h : \mathbb{R}^d \to \mathbb{R}^d$, so that policy uses modified inputs $a \sim \pi_{\theta^\star}(h(o))$. We consider two variations of the function h where one is a direct nonlinear transform $h(o) = H^{out}\sigma(H^{in}o)$ or else a nonlinear transformation with a residual connection $h(o) = o + H^{out}\sigma(H^{in}o)$, with learnable weights $\phi = \{H^{in}, H^{out}\}$. We use a hidden layer size of 256 units. The input adapter transforms observations to elicit better performance from a frozen model.

Prefix-tuning is a PEFT approach where a set of learnable tokens are pre-pended to the input sequence to elicit desired outputs from the model [39]. These prefixes are a sequence $\phi = [w_1, w_2, ..., w_m]$ of m tokens, where $w_i \in \mathbb{R}^h$ is a vector. These tokens are then pre-pended to the observations $o^{prefix} = [\phi; o^1, o^2, ..., o^d]$ and normally processed by Transformer layers. Tokens optionally can be pre-pended deeper in the model (e.g. $o_l^{prefix} = [\phi : T^l(o^{l-1})]$ for layer $l > 1$) or multiples prefix sets can be used (e.g. $\phi = \{\phi_1, \phi_2, ..., \phi_l\}$ would be learnable prefixes for each layer). We consider three major factors for effective prefix usage: (1) the number of tokens, (2) the injection layer, and (3) comparing token initialization approaches. Each factor represents hyperparameters in other PEFT research to impact performance substantially [45], [39]. For (3), we propose a second pre-training stage to learn morphology agnostic tokens. This second stage repeats the Metamorph training, but keep the base model frozen while learning the tokens.

IV. Experiments

This research aims to evaluate the efficacy of PEFT approaches for online learning on target morphologies. These experiments strive to address the following research questions: *(1) How effectively do each PEFT learning approach compare between each other and end-to-end finetuning? (2) What are the relevant factors for using prefix tuning and LoRA in online reinforcement learning? (3) What is relationship between total learnable parameters and performance for adapting to target morphologies?* Our results contributes to understanding the efficacy of these approaches in online learning, and can help guide future research developing PEFT algorithms for this setting.

We report representative experimental findings on the efficacy of different forms of parameter-efficient finetuning in morphological transfer. We use three locomotion tasks that differ in the terrain types shown in Figure 1; these include a flat surface, randomized variable terrain, and rectangular obstacles. Each task's reward function is to run as fast as possible to the right. To evaluate the PEFT techniques, we randomly sampled six morphologies from the Metamorph test dataset [19]. We evaluate PEFT techniques on eighteen environment-morphology combinations.

As mentioned in Section III, we generate our pre-trained models using the Metamorph framework [19]. We train five base models using one hundred training morphologies for ten million time steps for each environment. The variable terrain and obstacle avoidance tasks use external sensor data to estimate the locations of objects in the scene, which the decoder takes as an additional input. We then apply each PEFT approach with the pre-trained models on the six test morphologies for five million timesteps each. We repeat experiments for five random seeds for every set of PEFT hyperparameters we report. For each seed, we use one of the pre-trained models without replacement. We use the same learning hyperparameters for the pre-training phase, except we *do not use Dropout* in the Transformer embedding. Previous research shows that Dropout is critical for Metamorph pre-training [24]. In preliminary evaluations, we found Dropout interfered with prefix methods, and we elected to exclude its use in all targeted morphology learning.

A. Best Performances Across Methods

In our first set of experimental evaluations, we report performances of the best configuration for different PEFT approaches in Table II. We calculate the statistical significance of performance improvements with a t-test between each PEFT algorithm and zero-shot results. Except for E2E, all other approaches have fewer learning parameters than the size of the pre-trained model for the target morphology. These results confirm our hypothesis in Section III because the mean performance is higher for all PEFT approaches than zero-shot for each morphology-environment combination. In cases without statistical significance, the zero-shot performance was relatively high compared to other morphologies.

We further observed several trends in our results when looking between PEFT algorithms for each morphology. Across morphologies, results suggest that the best input-learnable configurations behave similarly to directly tuning the input Embedding and Decoder, suggesting some equivalence between the two approaches for the model sizes used in our experiments. Interestingly, we observed substantial performance improvements tuning just the fifth Transformer block, suggesting that if direct model access is possible and a more generous computation budget is available, this layer substantially influences the policy.

In order to contextualize these results to the proportion of learning parameters, we plot the percentage of learnable parameters with respect to total base model parameters against normalized cumulative reward in Figure 4. We divide PEFT

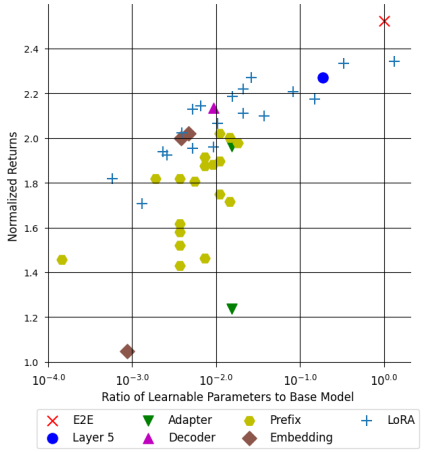


Fig. 4: Percentage of trainable ratios to total base model parameters vs achieved normalized results on Obstacle Avoidance. Results suggest total learnable parameters are a notable contributing factor.

cumulative rewards by zero-shot performance to normalize and plot the average results across morphologies. Each dot represents a hyperparameter configuration considered in our analysis. This plot reveals that across almost all configurations, even as few as less than 1% of parameters elicit improvements over zero-shot performance. These findings suggest that increasing the total learnable parameters whenever possible can lead to substantial performance improvements, offering practical insights for future research and application.

B. Ablation of LoRA and Prefix Tuning

In this section, we report results comparing different hyperparameter choices for LoRA and Prefix approaches. These methods are particularly interesting given their success in Foundation models [45], [39], [44], [46]. Reported results represent consistent behaviours we observed between evaluations in each environment.

Figure 6 shows the results of using LoRA in either the nonlinear transformations (MLP) or attention layer (Attn.) of the fifth transformer layer. The results show that across morphologies for single layer's full rank matrices are necessary and that nonlinear transformation is preferable for adaption to elicit optimal performance. These results suggest that directly tuning a single layer is better as it does not introduce additional learnable parameters like LoRA.

Prefixing tuning results have more nuanced conclusions. We generally observe that more learnable parameters are beneficial, such as by increasing the number of tokens used (see Figure 5), which agrees with other findings. In our experiments, a complication with prefix tuning is that introducing *un-trained tokens can negatively impact policy zero-shot performance*. When the base model is not trained jointly with the prefix, it introduces noise initially, which impacts zero-shot performance. This problem is largely missed in supervised learning applications because performance is

TABLE II: Best performance of different PEFT approaches for each terrain and morphology. We report mean cumulative reward across five seeds and † show statistically signifiant over zero shot results by a t-test with p-value < 0.01.

Morphology	1	2	3	4	5	6
Flat Terrain						
E2E	4281.46†	4552.77†	1635.82†	5545.44†	5019.71†	5558.61†
Layer 4	3761.11†	4121.84†	1491.06†	5183.88†	4666.95†	5192.58†
Lora	3798.90†	4208.41†	1639.69†	5223.47†	4761.58†	5223.47†
Decoder	2732.26†	3112.46†	1398.42†	4868.54	3404.67†	4858.76
Embeding	3308.43†	3684.66†	1554.28†	4986.16	4062.05†	4997.82
Adapter	3231.84†	3529.41†	1510.46†	4927.72	3946.59†	4963.53
Prefix	3332.33†	3750.54†	1604.92†	5064.15	4199.89†	5066.47
Zero Shot	1867.58	1703.19	253.70	4392.08	1849.41	4431.93
Variable Terrain						
E2E	2253.96†	1983.81†	2001.18†	3560.43†	2047.49†	3595.38†
Layer 4	2093.75†	1871.09†	1879.22†	3254.06†	1912.17†	3279.03
Lora	2141.39†	1848.53†	1786.88†	3230.13†	1878.93†	3234.25
Decoder	1969.63†	1623.70†	1299.89†	3164.72†	1672.47†	3180.90
Embeding	1836.54†	1529.38†	1441.65†	2872.51	1549.29†	2887.67
Adapter	1820.01†	1521.18†	1338.57†	2869.53	1512.25†	2895.01
Prefix	1902.95†	1643.33†	1406.55†	2930.13	1601.95†	2918.47
Zero Shot	1259.92	591.83	136.82	2452.59	685.54	2476.77
Obstacle Avoidance						
E2E	2652.41†	3101.42†	1705.64†	3577.09†	3219.75†	3558.26†
Layer 4	2246.88†	2684.70†	1592.29†	3276.76†	2888.34†	3194.19
Lora	2137.75†	2585.71†	1672.75†	3191.40†	2851.48†	3189.01
Decoder	2263.74†	2531.13†	1456.02†	3061.26	2672.06†	3132.18
Embeding	1863.25†	2189.46†	1556.72†	2882.24	2398.29†	2877.35
Adapter	1839.40†	2159.73†	1458.49†	2929.91	2367.39†	2833.04
Prefix	1841.45†	2334.01†	1514.42†	2877.43	2538.31†	2935.63
Zero Shot	1300.21	1184.87	332.64	2467.45	1295.92	2488.64

evaluated *after training*. In contrast, we care for performance *during training* especially because it's preferable policies have strong initial performance for real robotic systems avoid consequences of poor performaing policies (e.g. damage to the hardware). We conducted experiments adding 50 prefix tokens as input before different transformer blocks to investigate their impact on learning performance. We compared different token initializations, including zero vectors, small Gaussian noise ($N(0, 1e-4)$), or pretraining tokens, as described in our two-stage pretraining phase in Section III. We show learning curves in Figure 7.

Generally, we observed that the initial zero-shot performance is often negatively affected by zero or random initialization approaches, especially when introducing prefix tokens to the earlier layers of the Transformer. This result suggests that deep layers are generally less sensitive to the base models' perturbations and seem to better steer feature representations for target morphologies. Interestingly, pre-

trained prompting embeddings significantly improved policy performance during learning compared to other initialization approaches, especially on Morphology #3, which we found most PEFT approaches struggled to learn. This demonstrates that prefix initialization can mitigate loss in zero-shot performance during finetuning in online learning.

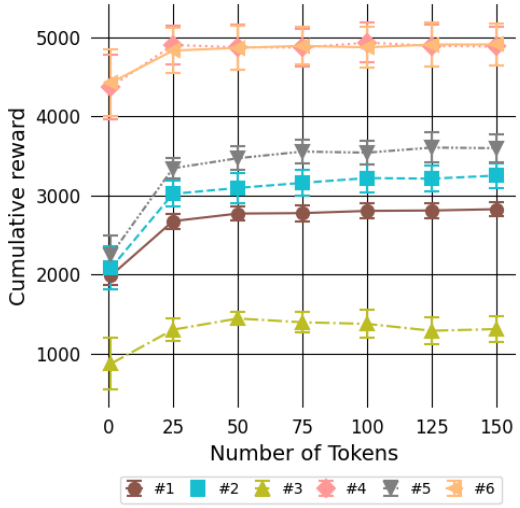


Fig. 5: Affect of number of randomly initialized prefix tokens used on Flat terrain. Each lines is is a morphology.

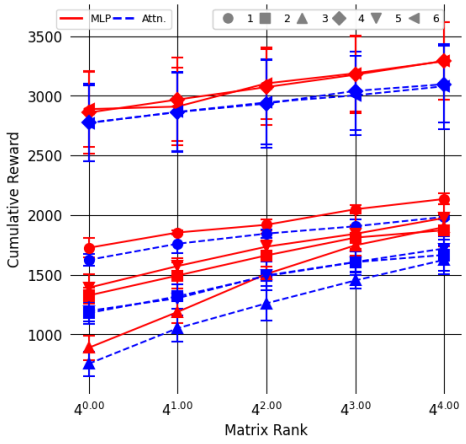


Fig. 6: LoRA's affect on Attention (Attn.) or Nonlinear Transformation Layers (MLP) in fifth transformer layer on Obstacle Avoidance. Markers indicate morphology.

V. Conclusions and Future Work

In this paper, we have investigated the impact of PEFT approaches that directly or indirectly can influence the behaviour of morphology agnostic policies. We demonstrate that in most cases, one should learn as many parameters online as possible to elicit the best performances of a pretrained policy, at least for locomotion tasks. Our analysis reveals that many PEFT approaches provide substantial benefits in deeper layers, so tuning the final transformer block

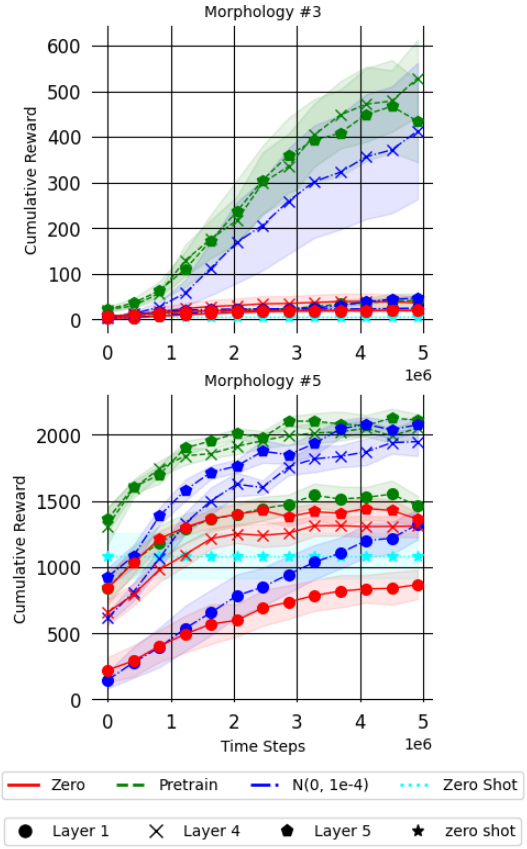


Fig. 7: Choice of initialization and injection layers of prefix tuning in obstacle avoidance. Initial zero-shot results of E2E learning are plotted to compare affect of prefixes.

is likely effective for policy finetuning. In scenarios where directly finetuning the base model is difficult, learnable inputs perform similarly to tuning either the input embeddings or decoder layers of the transformer-based policy.

There are several promising future research directions to extend our findings. One crucial factor, particularly for prefix tuning approaches, is the scale of the model. Many reported successes of PEFT approaches are on Foundation models with tens of millions to billions of parameters [39], where in this work, we use relatively small models (∼3.5 million parameters at most between policy and value function in PPO). We also focused on Transformer architectures used in Metamorph, but other variations have also been considered for morphology agnostic policies [24], [21]. Whether directly changing the model or scaling the number of parameters, we see much future promise in developing PEFT solutions targeted for online learning, which stands to impact the applications of deep reinforcement learning to robotics.

REFERENCES

[1] T. Chen, A. Murali, and A. Gupta, "Hardware conditioned policies for multi-robot transfer learning," in *Advances in Neural Information Processing Systems*, S. Bengio, H. M. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., 2018, pp. 9355–9366. [Online]. Available: https://proceedings.neurips.cc/paper/2018/hash/b8cfbf77a3d250a4523ba67a65a7d031-Abstract.html

[2] C. B. Schaff, D. Yunis, A. Chakrabarti, and M. R. Walter, "Jointly learning to construct and control agents using deep reinforcement learning," in *International Conference on Robotics and Automation*. IEEE, 2019, pp. 9798–9805. [Online]. Available: https://doi.org/10.1109/ICRA.2019.8793537

[3] W. Huang, I. Mordatch, and D. Pathak, "One policy to control them all: Shared modular policies for agent-agnostic control," in *Proceedings of the 37th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 119. PMLR, 2020, pp. 4455–4464. [Online]. Available: http://proceedings.mlr.press/v119/huang20d.html

[4] X. Pan, T. Zhang, B. Ichter, A. Faust, J. Tan, and S. Ha, "Zero-shot imitation learning from demonstrations for legged robot visual navigation," in *2020 IEEE International Conference on Robotics and Automation, ICRA*. IEEE, 2020, pp. 679–685. [Online]. Available: https://doi.org/10.1109/ICRA40945.2020.9196602

[5] A. Nair, V. Pong, M. Dalal, S. Bahl, S. Lin, and S. Levine, "Visual reinforcement learning with imagined goals," in *Advances in Neural Information Processing Systems*, 2018, pp. 9209–9220. [Online]. Available: https://proceedings.neurips.cc/paper/2018/hash/7ec69dd44416c46745f6edd947b470cd-Abstract.html

[6] V. Pong, M. Dalal, S. Lin, A. Nair, S. Bahl, and S. Levine, "Skew-fit: State-covering self-supervised reinforcement learning," in *Proceedings of the 37th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 119. PMLR, 2020, pp. 7783–7792. [Online]. Available: http://proceedings.mlr.press/v119/pong20a.html

[7] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proceedings of the 34th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 70. PMLR, 2017, pp. 1126–1135. [Online]. Available: http://proceedings.mlr.press/v70/finn17a.html

[8] S. James, M. Bloesch, and A. J. Davison, "Task-embedded control networks for few-shot imitation learning," in *2nd Annual Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, vol. 87. PMLR, 2018, pp. 783–795. [Online]. Available: http://proceedings.mlr.press/v87/james18a.html

[9] A. Nagabandi, I. Clavera, S. Liu, R. S. Fearing, P. Abbeel, S. Levine, and C. Finn, "Learning to adapt in dynamic, real-world environments through meta-reinforcement learning," in *7th International Conference on Learning Representations*, 2019. [Online]. Available: https://openreview.net/forum?id=HyztsoC5Y7

[10] N. M. M. Shafiullah, A. Rai, H. Etukuru, Y. Liu, I. Misra, S. Chintala, and L. Pinto, "On bringing robots home," *arXiv preprint arXiv:2311.16098*, 2023.

[11] A. O'Neill, A. Rehman, A. Maddukuri, A. Gupta, A. Padalkar, A. Lee, A. Pooley, A. Gupta, A. Mandlekar, A. Jain, A. Tung, A. Bewley, A. Herzog, A. Irpan, A. Khazatsky, A. Rai, A. Gupta, A. Wang, A. Singh, A. Garg, A. Kembhavi, A. Xie, A. Brohan, A. Raffin, A. Sharma, A. Yavary, A. Jain, A. Balakrishna, A. Wahid, B. Burgess-Limerick, B. Kim, B. Schölkopf, B. Wulfe, B. Ichter, C. Lu, C. Xu, C. Le, C. Finn, C. Wang, C. Xu, C. Chi, C. Huang, C. Chan, C. Agia, C. Pan, C. Fu, C. Devin, D. Xu, D. Morton, D. Driess, D. Chen, D. Pathak, D. Shah, D. Büchler, D. Jayaraman, D. Kalashnikov, D. Sadigh, E. Johns, E. P. Foster, F. Liu, F. Ceola, F. Xia, F. Zhao, F. Stulp, G. Zhou, G. S. Sukhatme, G. Salhotra, G. Yan, G. Feng, G. Schiavi, G. Berseth, G. Kahn, G. Wang, H. Su, H. Fang, H. Shi, H. Bao, H. B. Amor, H. I. Christensen, H. Furuta, H. Walke, H. Fang, H. Ha, I. Mordatch, I. Radosavovic, I. Leal, J. Liang, J. Abou-Chakra, J. Kim, J. Drake, J. Peters, J. Schneider, J. Hsu, J. Bohg, J. Bingham, J. Wu, J. Gao, J. Hu, J. Wu, J. Wu, J. Sun, J. Luo, J. Gu, J. Tan, J. Oh, J. Wu, J. Lu, J. Yang, J. Malik, J. Silvério, J. Hejna, J. Booher, J. Tompson, J. Yang, J. Salvador, J. J. Lim, J. Han, K. Wang, K. Rao, K. Pertsch, K. Hausman, K. Go, K. Gopalakrishnan, K. Goldberg, K. Byrne, K. Oslund, K. Kawaharazuka, K. Black, K. Lin, K. Zhang, K. Ehsani, K. Lekkala, K. Ellis, K. Rana, K. Srinivasan, K. Fang, K. P. Singh, K. Zeng, K. Hatch, K. Hsu, L. Itti, L. Y. Chen, L. Pinto, L. Fei-Fei, L. Tan, L. J. Fan, L. Ott, L. Lee, L. Weihs, M. Chen, M. Lepert, M. Memmel, M. Tomizuka, M. Itkina, M. G. Castro, M. Spero, M. Du, M. Ahn, M. C. Yip, M. Zhang, M. Ding, M. Heo, M. K. Srirama, M. Sharma, M. J. Kim, N. Kanazawa, N. Hansen, N. Heess, N. J. Joshi, N. Sünderhauf, N. Liu, N. D. Palo, N. M. M. Shafiullah, O. Mees, O. Kroemer, O. Bastani, P. R. Sanketi, P. T. Miller, P. Yin, P. Wohlhart, P. Xu, P. D. Fagan, P. Mitrano, P. Sermanet, P. Abbeel, P. Sundaresan, Q. Chen, Q. Vuong, R. Rafailov, R. Tian, R. Doshi, R. Martín-Martín, R. Baijal, R. Scalise, R. Hendrix, R. Lin, R. Qian, R. Zhang, R. Mendonca, R. Shah, R. Hoque, R. Julian, S. Bustamante, S. Kirmani, S. Levine, S. Lin, S. Moore, S. Bahl, S. Dass, S. D. Sonawani, S. Song, S. Xu, S. Haldar, S. Karamcheti, S. Adebola, S. Guist, S. Nasiriany, S. Schaal, S. Welker, S. Tian, S. Ramamoorthy, S. Dasari, S. Belkhale, S. Park, S. Nair, S. Mirchandani, T. Osa, T. Gupta, T. Harada, T. Matsushima, T. Xiao, T. Kollar, T. Yu, T. Ding, T. Davchev, T. Z. Zhao, T. Armstrong, T. Darrell, T. Chung, V. Jain, V. Vanhoucke, W. Zhan, W. Zhou, W. Burgard, X. Chen, X. Wang, X. Zhu, X. Geng, X. Liu, L. Xu, X. Li, Y. Lu, Y. J. Ma, Y. Kim, Y. Chebotar, Y. Zhou, Y. Zhu, Y. Wu, Y. Xu, Y. Wang, Y. Bisk, Y. Cho, Y. Lee, Y. Cui, Y. Cao, Y. Wu, Y. Tang, Y. Zhu, Y. Zhang, Y. Jiang, Y. Li, Y. Li, Y. Iwasawa, Y. Matsuo, Z. Ma, Z. Xu, Z. J. Cui, Z. Zhang, and Z. Lin, "Open x-embodiment: Robotic learning datasets and RT-X models : Open x-embodiment collaboration," in *IEEE International Conference on Robotics and Automation*. IEEE, 2024, pp. 6892–6903. [Online]. Available: https://doi.org/10.1109/ICRA57147.2024.10611477

[12] M. J. Kim, K. Pertsch, S. Karamcheti, T. Xiao, A. Balakrishna, S. Nair, R. Rafailov, E. P. Foster, G. Lam, P. Sanketi, Q. Vuong, T. Kollar, B. Burchfiel, R. Tedrake, D. Sadigh, S. Levine, P. Liang, and C. Finn, "Openvla: An open-source vision-language-action model," *arXiv preprint*, vol. arXiv:2406.09246, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2406.09246

[13] X. Lin, J. So, S. Mahalingam, F. Liu, and P. Abbeel, "Spawnnet: Learning generalizable visuomotor skills from pre-trained network," in *IEEE International Conference on Robotics and Automation*. IEEE, 2024, pp. 4781–4787. [Online]. Available: https://doi.org/10.1109/ICRA57147.2024.10610356

[14] K. Lu, K. Ly, W. Hebberd, K. Zhou, I. Havoutis, and A. Markham, "Learning generalizable manipulation policy with adapter-based parameter fine-tuning," 05 2024.

[15] J. J. Craig, *Introduction to Robotics: Mechanics and Control*, ser. Addison-Wesley series in electrical and computer engineering: Control engineering. Pearson/Prentice Hall, 2005. [Online]. Available: https://books.google.ca/books?id=ZJkOSgAACAAJ

[16] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, "The graph neural network model," *IEEE Transactions on Neural Networks*, vol. 20, no. 1, pp. 61–80, 2009. [Online]. Available: https://doi.org/10.1109/TNN.2008.2005605

[17] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, 2017, pp. 5998–6008.

[18] T. Wang, R. Liao, J. Ba, and S. Fidler, "Nervenet: Learning structured policy with graph neural networks," in *International Conference on Learning Representations*, 2018. [Online]. Available: https://openreview.net/forum?id=S1sqHMZCb

[19] A. Gupta, L. Fan, S. Ganguli, and L. Fei-Fei, "Metamorph: Learning universal controllers with transformers," in *International Conference on Learning Representations*, 2022. [Online]. Available: https://openreview.net/forum?id=Opmqtk_GvYL

[20] S. Hong, D. Yoon, and K. Kim, "Structure-aware transformer policy for inhomogeneous multi-task reinforcement learning," in *The Tenth International Conference on Learning Representations*. OpenReview.net, 2022. [Online]. Available: https://openreview.net/forum?id=fy_XRVHqly

[21] B. Trabucco, M. Phielipp, and G. Berseth, "AnyMorph: Learning transferable polices by inferring agent morphology," in *Proceedings of the 39th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvari, G. Niu, and S. Sabato, Eds., vol. 162. PMLR, 17–23 Jul 2022, pp. 21 677–21 691. [Online]. Available: https://proceedings.mlr.press/v162/trabucco22b.html

[22] D. Pathak, C. Lu, T. Darrell, P. Isola, and A. A. Efros, "Learning to control self-assembling morphologies: A study of generalization via modularity," in *Advances in*

Neural Information Processing Systems, 2019, pp. 2292–2302. [Online]. Available: https://proceedings.neurips.cc/paper/2019/hash/c26820b8a4c1b3c2aa868d6d57e14a79-Abstract.html

[23] Y. Yuan, Y. Song, Z. Luo, W. Sun, and K. M. Kitani, "Transform2act: Learning a transform-and-control policy for efficient agent design," in *The Tenth International Conference on Learning Representations*. OpenReview.net, 2022. [Online]. Available: https://openreview.net/forum?id=UcDUxjPYWSr

[24] Z. Xiong, J. Beck, and S. Whiteson, "Universal morphology control via contextual modulation," in *Proceedings of the 40th International Conference on Machine Learning*, ser. ICML'23. JMLR.org, 2023.

[25] C. Sferrazza, D.-M. Huang, F. Liu, J. Lee, and P. Abbeel, "Body transformer: Leveraging robot embodiment for policy learning," in *Workshop on Embodiment-Aware Robot Learning*, 2024. [Online]. Available: https://openreview.net/forum?id=IbXqRpANPD

[26] B. Li, H. Li, Y. Zhu, and D. Zhao, "MAT: morphological adaptive transformer for universal morphology policy learning," *IEEE Transactions on Cognitive and Developmental Systems*, vol. 16, no. 4, pp. 1611–1621, 2024. [Online]. Available: https://doi.org/10.1109/TCDS.2024.3383158

[27] Y. Hao, Y. Yang, J. Song, W. Peng, W. Zhou, T. Jiang, and W. Yao, "Heteromorpheus: Universal control based on morphological heterogeneity modeling," *arXiv preprint*, vol. arXiv:2408.01230, 2024. [Online]. Available: http://arxiv.org/abs/2408.01230

[28] S. M. Neuman, B. Plancher, B. P. Duisterhof, S. Krishnan, C. Banbury, M. Mazumder, S. Prakash, J. Jabbour, A. Faust, G. C. de Croon, and V. J. Reddi, "Tiny robot learning: Challenges and directions for machine learning in resource-constrained robots," in *2022 IEEE 4th International Conference on Artificial Intelligence Circuits and Systems (AICAS)*, 2022, pp. 296–299.

[29] Z. Huai, B. Ding, H. Wang, M. Geng, and L. Zhang, "Towards deep learning on resource-constrained robots: A crowdsourcing approach with model partition," in *2019 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*, 2019, pp. 989–994.

[30] S. Nair, A. Rajeswaran, V. Kumar, C. Finn, and A. Gupta, "R3M: A universal visual representation for robot manipulation," in *Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, vol. 205. PMLR, 2022, pp. 892–909. [Online]. Available: https://proceedings.mlr.press/v205/nair23a.html

[31] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," in *Proceedings of the 38th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 139. PMLR, 2021, pp. 8748–8763. [Online]. Available: http://proceedings.mlr.press/v139/radford21a.html

[32] I. Radosavovic, T. Xiao, S. James, P. Abbeel, J. Malik, and T. Darrell, "Real-world robot learning with masked visual pre-training," in *Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, vol. 205. PMLR, 2022, pp. 416–426. [Online]. Available: https://proceedings.mlr.press/v205/radosavovic23a.html

[33] R. Julian, B. Swanson, G. S. Sukhatme, S. Levine, C. Finn, and K. Hausman, "Never stop learning: The effectiveness of fine-tuning in robotic reinforcement learning," in *4th Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, J. Kober, F. Ramos, and C. J. Tomlin, Eds., vol. 155. PMLR, 2020, pp. 2120–2136. [Online]. Available: https://proceedings.mlr.press/v155/julian21a.html

[34] M. Sharma, C. Fantacci, Y. Zhou, S. Koppula, N. Heess, J. Scholz, and Y. Aytar, "Lossless adaptation of pretrained vision models for robotic manipulation," in *The Eleventh International Conference on Learning Representations*. OpenReview.net, 2023. [Online]. Available: https://openreview.net/forum?id=5IND3TXJRb-

[35] Z. Liu, J. Zhang, K. Asadi, Y. Liu, D. Zhao, S. Sabach, and R. Fakoor, "TAIL: task-specific adapters for imitation learning with large pretrained models," in *The Twelfth International Conference on Learning Representations*. OpenReview.net, 2024. [Online]. Available: https://openreview.net/forum?id=RRayv1ZPN3

[36] Q. Dong, L. Li, D. Dai, C. Zheng, Z. Wu, B. Chang, X. Sun, J. Xu, L. Li, and Z. Sui, "A survey for in-context learning," *arXiv preprint*, vol. arXiv:2301.00234, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2301.00234

[37] R. Kirk, A. Zhang, E. Grefenstette, and T. Rocktäschel, "A survey of zero-shot generalisation in deep reinforcement learning," *J. Artif. Int. Res.*, vol. 76, may 2023. [Online]. Available: https://doi.org/10.1613/jair.1.14174

[38] Y.-Y. Tsai, P.-Y. Chen, and T.-Y. Ho, "Transfer learning without knowing: Reprogramming black-box machine learning models with scarce data and limited resources," in *Proceedings of the 37th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, H. D. III and A. Singh, Eds., vol. 119. PMLR, 13–18 Jul 2020, pp. 9614–9624. [Online]. Available: https://proceedings.mlr.press/v119/tsai20a.html

[39] X. L. Li and P. Liang, "Prefix-tuning: Optimizing continuous prompts for generation," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2021, pp. 4582–4597. [Online]. Available: https://doi.org/10.18653/v1/2021.acl-long.353

[40] Y. Lee, A. S. Chen, F. Tajwar, A. Kumar, H. Yao, P. Liang, and C. Finn, "Surgical fine-tuning improves adaptation to distribution shifts," *International Conference on Learning Representations*, 2023.

[41] A. Hallak, D. D. Castro, and S. Mannor, "Contextual markov decision processes," 2015. [Online]. Available: https://arxiv.org/abs/1502.02259

[42] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," 2016. [Online]. Available: https://arxiv.org/abs/1607.06450

[43] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *CoRR*, vol. abs/1707.06347, 2017. [Online]. Available: http://arxiv.org/abs/1707.06347

[44] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "Lora: Low-rank adaptation of large language models," in *The Tenth International Conference on Learning Representations*, 2022. [Online]. Available: https://openreview.net/forum?id=nZeVKeeFYf9

[45] N. Ding, Y. Qin, G. Yang, F. Wei, Z. Yang, Y. Su, S. Hu, Y. Chen, C.-M. Chan, W. Chen, J. Yi, W. Zhao, X. Wang, Z. Liu, H.-T. Zheng, J. Chen, Y. Liu, J. Tang, J. Li, and M. Sun, "Parameter-efficient fine-tuning of large-scale pre-trained language models," *Nature Machine Intelligence*, vol. 5, no. 3, pp. 220–235, Mar 2023. [Online]. Available: https://doi.org/10.1038/s42256-023-00626-4

[46] Y. Hao, Y. Cao, and L. Mou, "Flora: Low-rank adapters are secretly gradient compressors," in *Forty-first International Conference on Machine Learning*, 2024. [Online]. Available: https://arxiv.org/abs/2402.03293

References are important to the reader; therefore, each citation must be complete and correct. If at all possible, references should be commonly available publications.