# Learning Diverse Skills for Local Navigation under Multi-constraint Optimality

**Jin Cheng** **Marin Vlastelica** **Pavel Kolev** **Chenhao Li** **Georg Martius**

**Max Planck Institute for Intelligent Systems**
Max-Planck-Ring 4, 72076 Tübingen, Germany
{first.last}@tuebingen.mpg.de

## Abstract

Despite many successful applications of data-driven control in robotics, extracting meaningful diverse behaviors remains a challenge. Typically, task performance needs to be compromised in order to achieve diversity. In many scenarios, task requirements are specified as a multitude of reward terms, each requiring a different trade-off. In this work, we take a constrained optimization viewpoint on the quality-diversity trade-off and show that we can obtain diverse policies while imposing constraints on their value functions which are defined through distinct rewards. In line with previous work, further control of the diversity level can be achieved through an attract-repel reward term motivated by the Van der Waals force. We demonstrate the effectiveness of our method on a local navigation task where a quadruped robot needs to reach the target within a finite horizon. Finally, our trained policies transfer well to the real 12-DoF quadruped robot, Solo12, and exhibit diverse agile behaviors with successful obstacle traversal[1].

## 1 Introduction

Reinforcement Learning (RL) has proven itself as a valuable tool for equipping robotic platforms with a variety of capabilities. However, the ability of RL to provide a range of diverse solutions for the same task still remains a challenging frontier.

Given a set of reward functions describing a particular task, our goal is to train a variety of different skills that solve the same task proficiently. This essentially formulates a constraint optimization problem as proposed by Zahavy et al. [2022], where the objective is to maximize diversity while satisfying constraints that guarantee that each skill achieves a certain level of cumulative reward in comparison to an expert trained with only task rewards.

In this study, we introduce Diversity Optimization under Multiple Near-optimal Constraints (DOMiNiC), an adaptive extension to the DOMiNO [Zahavy et al., 2022] framework. DOMiNiC is particularly effective in environments characterized by a wide variety of constraints, including important facets such as task-based rewards, safety-oriented regularization, and discretionary auxiliary rewards – all with different requirements on how much they can be sacrificed.

## 2 Preliminaries

In the general RL setup, an agent interacts with an environment to maximize the cumulative discounted reward. From the definition of Markov decision processes (MDPs) [Puterman, 2014], an initial

[1]Project website with videos: https://sites.google.com/view/icra2024-dominic

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

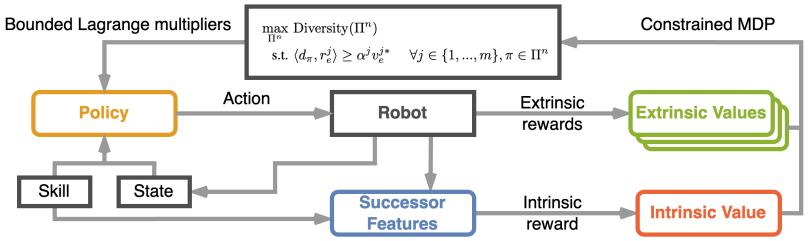


Figure 1: The DOMiNiC training scheme. We collect samples in simulation and fit the extrinsic values for updating the Lagrange multipliers and an intrinsic value function based on eq. (10) (VDW) used for measuring diversity. These are combined into an aggregate advantage term in eq. (6), ensuring that intrinsic reward is maximized only after all constraints are satisfied.

state $s_0$ is sampled from a state distribution $\rho(s_0)$, then at each time step $t$, the agent applies an action $a_t$ according to a policy $\pi(a_t|s_t)$ given a state $s_t$, and receives from the environment a reward $r_t \sim R(s_t, a_t)$ and a next state $s_{t+1} \sim P(s_{t+1}|s_t, a_t)$. The performance metric can be written as $v_\pi = (1-\gamma)\mathbb{E}[\sum_{t=0}^{\infty} \gamma^t r_t]$ and the state-action occupancy is defined as $d_\pi(s,a) = (1-\gamma)\mathbb{E}[\sum_{t=0}^{\infty} \gamma^t P_\pi(s_t = s)\pi(a|s)]$. The RL objective can be rewritten as maximizing a function of the occupancy measure $\max_{d_\pi \in \mathcal{K}} \langle d_\pi, r \rangle$, where $\langle d_\pi, r \rangle = \sum_{s,a} d_\pi(s,a) r(s,a)$ denotes the inner product and $\mathcal{K}$ is the set of admissible distributions [Zahavy et al., 2021].

Zahavy et al. [2022] studied the Constrained Markov Decision Process (CMDP) formulation, which seeks to compute a set of policies $\Pi^n = \{\pi^z\}_{z=1}^n$ that satisfy

$$\max_{\Pi^n} \text{Diversity}(\Pi^n) \text{ s.t. } \langle d_\pi, r_e \rangle \geq \alpha v_e^*, \quad \forall \pi \in \Pi^n, \tag{1}$$

where $r_e$ and $v_e^*$ correspond to the extrinsic reward and optimal extrinsic value. Intuitively, it computes a set of diverse policies while maintaining a certain level of extrinsic optimality specified by the optimality ratio $\alpha \in [0, 1]$.

As shown in previous work [Zahavy et al., 2021], convex diversity objectives can be optimized by solving a sequence of standard RL problems, each with an intrinsic reward equal to the gradient of the objective evaluated at a state-action occupancy $d_\pi$ of the current iteration:

$$r_i^z = \nabla_{d_\pi^z} \text{Diversity}(d_\pi^1, ..., d_\pi^n), \quad \forall z. \tag{2}$$

Based on a distance measure from Abbeel and Ng [2004], Zahavy et al. [2022] modeled the diversity objective as the maximization of the minimum squared $\ell_2$ distance between feature expectations of different skills, namely

$$\max_{d_\pi^1, ..., d_\pi^n} 0.5 \sum_{z=1}^{n} \min_{k \neq z} \left\| \psi^z - \psi^k \right\|_2^2. \tag{3}$$

More specifically, given a feature mapping $\phi : \mathcal{S} \to \mathbb{R}^n$, the feature expectations are defined by $\psi^z = \mathbb{E}_{d_\pi^z(s)}[\phi(s)]$. Furthermore, they also introduced a physically inspired objective based on Van der Waals (VDW) force, and considered the following optimization objective

$$\max_{d_\pi^1, ..., d_\pi^n} 0.5 \sum_{z=1}^{n} \ell_z^2 - 0.2(\ell_z^5 / \ell_0^3), \tag{4}$$

where $\ell_z = \min_{k \neq z} \left\| \psi^z - \psi^k \right\|_2$, which allows the level of diversity to be controlled by $\ell_0$. When the features are in close proximity $\ell_i < \ell_0$, the repulsive force dominates, whereas when $\ell_i > \ell_0$ the attractive force prevails.

## 3 Method

The DOMiNO Zahavy et al. [2022] framework utilizes a single scalar extrinsic value as a metric to assess the proficiency of learned skills. However, this methodology faces challenges when dealing with tasks characterized by multiple objectives, as it lacks clarity in discerning which particular objective may undergo compromise. Moreover, the spectrum of different optimality considerations cannot be adequately formulated within the single constraint MDP formulation in eq. (1).

To address these problems, we present Diversity Optimization under Multiple Near-optimal Constraints (DOMiNiC) to extend the capacity of this framework to multi-constraint optimization scenarios. More specifically, we categorize the rewards into $m$ different groups. Each constraint group $j$ has an associated reward $r_e^j$ and an optimality ratio $\alpha^j$. We consider the following formulation:

$$\begin{aligned} &\max_{\Pi^n} \text{Diversity}(\Pi^n) \\ &\text{s.t. } \langle d_\pi, r_e^j \rangle \geq \alpha^j v_e^{j*} \quad \forall j \in \{1, ..., m\}, \pi \in \Pi^n, \end{aligned} \tag{5}$$

where $r_e^j$ is the extrinsic reward and $v_e^{j*}$ is the optimal value of group $j$. This multi-constraint formulation allows fine-grained control over different constraint groups, via the parameters $\{\alpha^j\}_{j=1}^m$. An overview of our framework is shown in fig. 1.

To ensure that the intrinsic reward is maximized only after all constraint groups have been satisfied, we introduce the following aggregate advantage term

$$a^z = \Big(1 - \max_j \sigma(\mu^{j,z})\Big) a_i^z + \sum_j \sigma(\mu^{j,z}) a_e^j, \quad \forall z, \tag{6}$$

where $a_e^j$ is the extrinsic advantage for reward group $j$, $a_i^z$ is the intrinsic advantage of skill $z$, and $\sigma(\mu^{j,z})$ is the bounded Lagrange multiplier for constraint $j$ and skill $z$. Note that the aggregate advantage $a^z \to a_i^z$ when $\sigma(\mu^{j,z}) \to 0$ for all constraint groups, i.e., when all constraints are satisfied.

The Lagrange multipliers are updated according to the following loss function, which is designed to guarantee the satisfaction of all constraint groups:

$$\mathcal{L}_\mu = \sum_{z=1}^{n} \sum_{j=1}^{m} \mathop{\mathbb{E}}_{v_e^{j,z}} \left[ \mu^{j,z} (\alpha^j v_e^{j*} - v_e^{j,z}) \right]. \tag{7}$$

To compute the feature expectations $\psi^z$, we use the state-conditioned Successor Features (SFs) proposed by Barreto et al. [2017], which decouple the environment dynamics from rewards and facilitate knowledge transfer across tasks. The SFs of a policy $\pi$ evaluated at a state $s$ are given by $\psi^z(s) = \mathbb{E}_z\big[\sum_{i=0}^{\infty} \gamma^t \phi(s_t) \mid s_0 = s\big]$. Our algorithm relies on the following two properties: i) feature expectations satisfy $\psi^z = \mathbb{E}_{\rho(s_0)}[\psi^z(s_0)]$, where $\rho(s_0)$ is the initial state distribution; and ii) SFs $\psi^z(s)$ can be trained by a learning process similar to training a value function, using Temporal Difference (TD) updates [Sutton and Barto, 2018], which minimizes the loss

$$\mathcal{L}_\psi = \sum_{z=1}^{n} \mathop{\mathbb{E}}_{\pi^z} \|\phi(s) + \gamma \psi^z(s') - \psi^z(s)\|_2^2. \tag{8}$$

At each time step, the intrinsic reward $r_i$ is computed from the learned SFs $\psi(s)$ either by the repulsive force in eq. (3)

$$r_i^z(s) = \langle \phi(s), \psi^z - \psi^{z^\star} \rangle, \tag{9}$$

or from the VDW force in eq. (4)

$$r_i^z(s) = (1 - (\ell_z/\ell_0)^3) \langle \phi(s), \psi^z - \psi^{z^\star} \rangle, \tag{10}$$

where $z^\star = \arg\min_{k \neq z} \left\| \psi^z - \psi^k \right\|_2$.

Instead of concatenating the one-hot encoder of discrete skills to the input as in previous work, we use randomly initialized layer masks for all skill-conditioned neural networks, including the policy, value functions, and SFs. Before training, a binary mask of the same size as the hidden dimensions is sampled and fixed for each skill. During training, the mask activation is used to set the output of the corresponding neural units to zero. Using these masks ensures that individual skills retain distinct features, mitigating interference and promoting skill diversity within the neural network architecture. The mask sampling probability is chosen to balance the independence and overlap of neural units between skills.

In contrast to prior work, our approach employs an on-policy training paradigm, drawing inspiration from Proximal Policy Optimization (PPO) Schulman et al. [2017]. This choice leads us to utilize Generalized Advantage Estimation (GAE) Schulman et al. [2015] as our preferred method, as opposed to the V-trace technique Espeholt et al. [2018] typically associated with off-policy frameworks. We also introduce a "warm-start" phase to warm up all skills to near-expert level, by pre-training them solely with extrinsic advantages. The complete pseudocode is provided in Section 6.4.

# 4 Experiments

We evaluate our method on Solo12, an open-source quadruped platform Léziart et al. [2021] with 12 degrees of freedom tasked with local navigation and locomotion both in simulation and on real hardware. We refer the interested reader to Section 6.2 and Section 6.3 for the task definition and training details.

## 4.1 Skill Discovery in Local Navigation

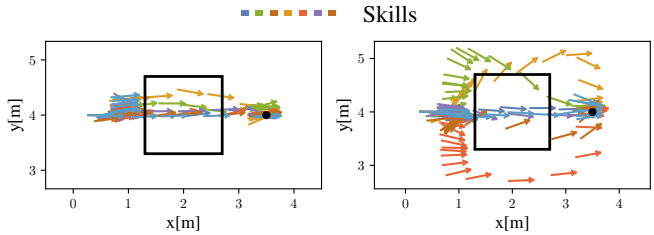


Figure 2: The top-down view of diverse trajectories of successful obstacle traversal obtained with different $\ell_0$ values (*left*: small $\ell_0$, *right*: large $\ell_0$) under the combination $[\alpha^t, \alpha^r, \alpha^s] = [0.9, 0.8, 0.7]$. Arrows indicate the yaw angle of the robot at trajectory points. The black squares are the visualization of the box obstacle and the black circles are the target positions that the robots have to reach.

In our first experiment, we show in simulation that diverse skills can be learned to successfully navigate to the target position in the presence of obstacles, while achieving a good balance between task, regularization, and style with optimality ratios $[\alpha^t, \alpha^r, \alpha^s] = [0.9, 0.8, 0.7]$. In addition, diversity can be controlled using the intrinsic objective derived from the VDW force in eq. (10).

In the top-down view of fig. 2, the learned skills exhibit different base velocity directions while moving towards the target and different strategies, including detours and jumping on the box, to overcome the obstacle. For small values of $\ell_0$, forcing low diversity, all learned skills converge to the "shortest path" solution, which is characterized by reaching the target position by jumping on the box and then jumping down to the target position on the ground.

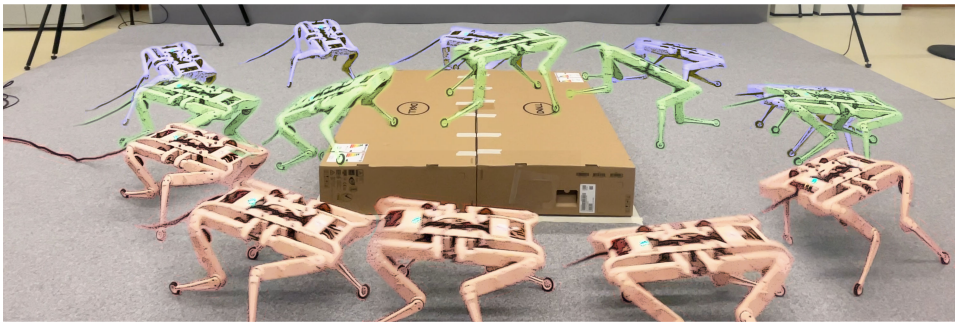

Figure 3: Obstacle experiment on hardware, we observe that the extracted skills explore different options in solving the obstacle. We have skills that go over the obstacle, to the right or to the left in different styles. The green skill is the one closest to the expert, which never takes detours around the box.

## 4.2 Quality-Diversity Balance

In the this experiment, we perform an extensive grid search on different combinations of optimality ratios $[\alpha^t, \alpha^r, \alpha^s]$ of the task, regularizer, the style, and different values of $\ell_0$ in the VDW intrinsic reward in eq. (10) to evaluate their influence on diversity. The results are shown in fig. 4.

The intrinsic reward in eq. (10), allows us to set a desired level of diversity by $\ell_0$. On the horizontal axis, the diversity is plotted by measuring the mean of the closest distance between the feature expectations $\psi^z$ evaluated on a uniform initial state distribution $\rho(s_0)$. Optimality ratios $[\alpha^t, \alpha^r, \alpha^s]$ give us the budget of how much extrinsic reward we can sacrifice for a gain in diversity. The vertical axis shows the percentage of returns achieved relative to the expert.

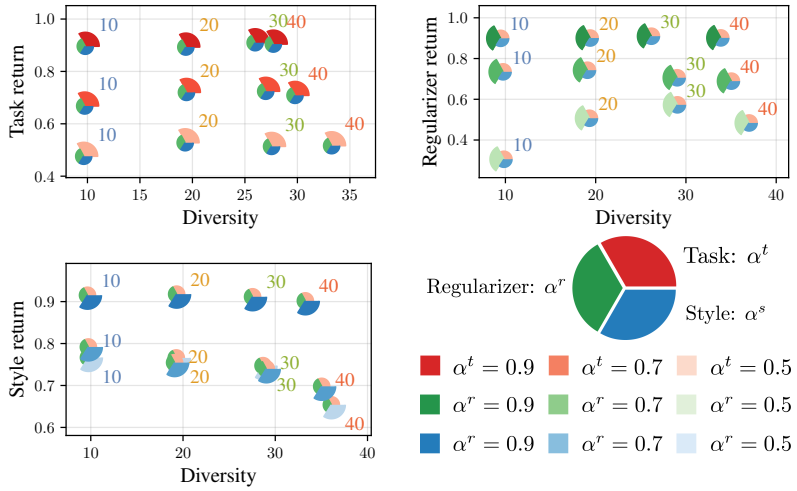


Figure 4: The controllability over quality and diversity through different values of optimality ratios $[\alpha^t, \alpha^r, \alpha^s]$ and $\ell_0$ from the VDW force objective. The results of a grid search over three different values $\{0.5, 0.7, 0.9\}$ for $[\alpha^t, \alpha^r, \alpha^s]$ and four values $\{10, 20, 30, 40\}$ for $\ell_0$ are shown as scattered pie plots. The colors on the three sectors represent different values for each optimality ratio, and the $\ell_0$ levels are shown as annotations at the points. For each figure, we fix two of the three optimality ratios and plot the return that corresponds to the varying term on the vertical axis: *top left*: $[*, 0.7, 0.9]$, *top right*: $[0.5, *, 0.7]$, *bottom left*: $[0.5, 0.7, *]$.

Overall, we find good controllability of the diversity via $\ell_0$ as well as of the quality of the behavior via the optimality ratios $[\alpha^t, \alpha^r, \alpha^s]$. It is important to emphasize that with looser constraints (smaller $\alpha$) we gain more diversity as shown in all three plots in fig. 4.

In addition to demonstrating controllability, several insights can be derived from fig. 4. First of all, we notice that the controllability of different reward groups are not entirely independent of each other. Imposing task and regularizer constraints leads to an over-satisfaction of the style constraint as shown from the bottom left plot. Second, it is more difficult to control the optimality of the regularizer group. We hypothesize that different reward terms in the multiplicative structure of the regularizer might influence each other. Managing the intricate interactions among reward groups and their sub-components remains a promising direction for future research.

### 4.3 Hardware Deployment

We deploy our trained policy on the real robot as shown in fig. 3, where the robot manages to choose diverse trajectories to reach the target behind a box obstacle with a width of 1.4 meters and height of 0.18 meters. Different skills extract diverse ways to traverse the obstacle by either jumping onto it or taking a detour around it from the left or the right. The policy deployed on hardware was trained with large optimality ratios for the task and regularizer to ensure good task performance and to fulfill the action smoothness required on a real system. For estimating the robot base state, a Vicon motion capture system is used to provide the base position and orientation at 100 Hz, and the velocity is calculated based on the finite difference. The position of the box obstacle is fixed in the global frame, so the height scan is created based on the absolute position of the robot.

## 5 Discussion

We propose DOMiNiC, a framework that effectively controls the trade-off between diversity and extrinsic rewards with multiple constraints by leveraging the CMDP formulation and incorporating the Van der Waals force as an intrinsic objective. We successfully train policies with diverse skills for Solo12, a 12-DoF quadruped robot tasked with locomotion and local navigation. The learned behaviors exhibit various successful obstacle traversal strategies in the real-world robotic system. Furthermore, the satisfaction of each constraint group contributes to the achievement of natural and diverse behaviors, emphasizing the significance of our proposed multi-constraint diversity optimization framework.

## References


P. Abbeel and A. Y. Ng. Apprenticeship learning via inverse reinforcement learning. In *Proceedings of the twenty-first international conference on Machine learning*, page 1, 2004.

E. Altman. *Constrained Markov decision processes*, volume 7. CRC Press, 1999.

A. Barreto, W. Dabney, R. Munos, J. J. Hunt, T. Schaul, H. P. van Hasselt, and D. Silver. Successor features for transfer in reinforcement learning. *Advances in neural information processing systems*, 30, 2017.

L. Espeholt, H. Soyer, R. Munos, K. Simonyan, V. Mnih, T. Ward, Y. Doron, V. Firoiu, T. Harley, I. Dunning, et al. Impala: Scalable distributed deep-rl with importance weighted actor-learner architectures. In *International conference on machine learning*, pages 1407–1416. PMLR, 2018.

B. Eysenbach, A. Gupta, J. Ibarz, and S. Levine. Diversity is all you need: Learning skills without a reward function. *arXiv preprint arXiv:1802.06070*, 2018.

Y. Fuchioka, Z. Xie, and M. Van de Panne. Opt-mimic: Imitation of optimized trajectories for dynamic quadruped behaviors. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5092–5098. IEEE, 2023.

D. Hoeller, N. Rudin, D. Sako, and M. Hutter. Anymal parkour: Learning agile navigation for quadrupedal robots. *arXiv preprint arXiv:2306.14874*, 2023.

J. Hwangbo, J. Lee, A. Dosovitskiy, D. Bellicoso, V. Tsounis, V. Koltun, and M. Hutter. Learning agile and dynamic motor skills for legged robots. *Science Robotics*, 4(26):eaau5872, 2019.

D. Kang, J. Cheng, M. Zamora, F. Zargarbashi, and S. Coros. Rl+ model-based control: Using on-demand optimal control to learn versatile legged locomotion. *arXiv preprint arXiv:2305.17842*, 2023.

A. Kumar, Z. Fu, D. Pathak, and J. Malik. RMA: Rapid motor adaptation for legged robots. In *Robotics: Science and Systems XVII (RSS)*, 2021.

S. Kumar, A. Kumar, S. Levine, and C. Finn. One solution is not all you need: Few-shot extrapolation via structured maxent rl. *Advances in Neural Information Processing Systems*, 33:8198–8210, 2020.

J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter. Learning quadrupedal locomotion over challenging terrain. *Science robotics*, 5(47):eabc5986, 2020.

P.-A. Léziart, T. Flayols, F. Grimminger, N. Mansard, and P. Souères. Implementation of a reactive walking controller for the new open-hardware quadruped solo-12. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5007–5013. IEEE, 2021.

C. Li, S. Blaes, P. Kolev, M. Vlastelica, J. Frey, and G. Martius. Versatile skill control via self-supervised adversarial imitation of unlabeled mixed motions. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2944–2950. IEEE, 2023a.

C. Li, M. Vlastelica, S. Blaes, J. Frey, F. Grimminger, and G. Martius. Learning agile skills via adversarial imitation of rough partial demonstrations. In *Conference on Robot Learning*, pages 342–352. PMLR, 2023b.

V. Makoviychuk, L. Wawrzyniak, Y. Guo, M. Lu, K. Storey, M. Macklin, D. Hoeller, N. Rudin, A. Allshire, A. Handa, et al. Isaac gym: High performance gpu-based physics simulation for robot learning. *arXiv preprint arXiv:2108.10470*, 2021.

T. Miki, J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter. Learning robust perceptive locomotion for quadrupedal robots in the wild. *Science Robotics*, 7(62):eabk2822, 2022.

S. Park, J. Choi, J. Kim, H. Lee, and G. Kim. Lipschitz-constrained unsupervised skill discovery. In *International Conference on Learning Representations*, 2021.

S. Park, K. Lee, Y. Lee, and P. Abbeel. Controllability-aware unsupervised skill discovery. *arXiv preprint arXiv:2302.05103*, 2023.

X. B. Peng, E. Coumans, T. Zhang, T.-W. Lee, J. Tan, and S. Levine. Learning agile robotic locomotion skills by imitating animals. *arXiv preprint arXiv:2004.00784*, 2020.

M. L. Puterman. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

N. Rudin, D. Hoeller, M. Bjelonic, and M. Hutter. Advanced skills by learning locomotion and local navigation end-to-end. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2497–2503. IEEE, 2022a.

N. Rudin, D. Hoeller, P. Reist, and M. Hutter. Learning to walk in minutes using massively parallel deep reinforcement learning. In *Conference on Robot Learning*, pages 91–100. PMLR, 2022b.

J. Schulman, P. Moritz, S. Levine, M. Jordan, and P. Abbeel. High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438*, 2015.

J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

A. Sharma, S. Gu, S. Levine, V. Kumar, and K. Hausman. Dynamics-aware unsupervised discovery of skills. *arXiv preprint arXiv:1907.01657*, 2019.

R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. The MIT Press, second edition, 2018.

C. Szepesvári. Constrained mdps and the reward hypothesis. *Musings about machine learning and other things (blog)*, 2020. URL https://readingsml.blogspot.com/2020/03/constrained-mdps-and-reward-hypothesis.html.

J. Tan, T. Zhang, E. Coumans, A. Iscen, Y. Bai, D. Hafner, S. Bohez, and V. Vanhoucke. Sim-to-real: Learning agile locomotion for quadruped robots. *arXiv preprint arXiv:1804.10332*, 2018.

M. Vlastelica, P. Kolev, J. Cheng, and G. Martius. Diverse offline imitation via fenchel duality. *arXiv preprint arXiv:2307.11373*, 2023.

T. Zahavy, B. O'Donoghue, G. Desjardins, and S. Singh. Reward is enough for convex mdps. In M. Ranzato, A. Beygelzimer, Y. N. Dauphin, P. Liang, and J. W. Vaughan, editors, *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 25746–25759, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/d7e4cdde82a894b8f633e6d61a01ef15-Abstract.html.

T. Zahavy, Y. Schroecker, F. Behbahani, K. Baumli, S. Flennerhag, S. Hou, and S. Singh. Discovering policies with domino: Diversity optimization maintaining near optimality. *arXiv preprint arXiv:2205.13521*, 2022.

# 6 Supplementary Material

## 6.1 Related Work

We focus on two aspects that are closely related to our work: RL-based control of quadruped robots and unsupervised skill discovery.

RL-based methods have recently shown their prominent capability in controlling legged systems [Tan et al., 2018, Hwangbo et al., 2019, Rudin et al., 2022b]. Trained in simulation, RL policies can achieve great robustness in tracking velocity commands over challenging terrains [Lee et al., 2020, Kumar et al., 2021, Miki et al., 2022]. However, velocity-commanded policies often converge to a single behavior exhibiting trotting gaits in the case of quadruped systems. Efforts have been made to use RL to achieve diverse behaviors by hierarchical control, imitation, and unsupervised skill discovery. By formulating locomotion as a position-based local navigation task, Rudin et al. [2022a] has recently shown agile behaviors emerging from RL such as climbing boxes and crossing gaps. These behaviors have been recently used as motion priors in a hierarchical control structure for long-horizon navigation tasks Hoeller et al. [2023] that preserves the diversity. Despite the impressive results, different priors require behavior-specific reward design, which needs a substantial amount of reward engineering to balance the behavior and regularization for all priors. Alternatively, combining skill-conditioned policy with imitation objective can largely reduce the extensive reward shaping. Imitating reference trajectories has shown the ability to generate agile behavior for quadruped robots such as dog-like hopping [Peng et al., 2020], backflipping [Li et al., 2023b] and walking with two feet [Fuchioka et al., 2023]. Kang et al. [2023] proposed to condition the locomotion policy on four phase variables in combination with imitating trajectories from model-based controllers to achieve different gait patterns on quadruped robots. Despite the skillful locomotion results, imitation-based methods often require prior knowledge of the robotic system and the resulting behavior is affected by the quality of the reference trajectories.

As an alternative, unsupervised skill discovery has recently gained research attention in the RL community, which is often related to maximizing the skill difference across policies that are conditioned on latent variables. The intrinsic objective can be incorporated into online training to discover diverse behaviors, and most recently Vlastelica et al. [2023] also proposed a Fenchel-duality approach for offline skill discovery. Mutual-information-based methods quantify the shared information between latent variables and the historical states by a skill-conditioned policy and maximize the mutual information between the skill and states to obtain distinct behaviors from each other [Eysenbach et al., 2018, Sharma et al., 2019], which has been shown to extract diverse behaviors successfully in combination with other rewards [Kumar et al., 2020, Li et al., 2023a]. Alternative to the mutual-information objective, diversity can also be measured by the Euclidean distance in the state or feature space [Park et al., 2021, 2023]. Li et al. [2023a] combined the imitation objective with unsupervised skill discovery to extract diverse skills from unlabeled offline demonstrations.

Despite the interesting motions discovered by these methods, acquiring meaningful and task-related behavior still remains challenging due to the need to balance quality and diversity carefully. Recently, Zahavy et al. [2022] proposed DOMiNO to combine unsupervised skill discovery with Constrained Markov Decision Processes (CMDPs) [Altman, 1999, Szepesvári, 2020] to ensure near-expert task performance as well as diversity in behaviors. CMDPs, which are a crucial part of RL with implications for safety and risk aversion, are first used to achieve quality-diversity balance.

As an extension to DOMiNO from Zahavy et al. [2022], DOMiNiC focuses on combining CMDPs to unsupervised skill discovery for real robotic systems. Specifically, we focused on a similar scenario as Rudin et al. [2022a] in training locomotion policies for quadruped robots with diverse behaviors to accomplish the local navigation task.

## 6.2 Quadruped Locomotion and Local Navigation

We demonstrate our method on the task of quadruped locomotion and local navigation in a position-based framework proposed by Rudin et al. [2022a], where the robot needs to navigate through an environment of randomly positioned boxes of different dimensions to reach a specified target position and orientation within a finite time horizon.

The observations of the policy include base linear and angular velocity, joint position and velocity, gravity vector projected in the base frame, and the height measurement sampled around the robot.

Random noise is added to these observations to simulate hardware sensor noise. In addition, the policy observes the previous actions, the three-dimensional target position in the base frame, the target yaw angle difference from the current base yaw angle, and a time indicator. The action of the policy is the target joint positions which will be taken by a PD controller and transformed into joint torques.

We group extrinsic reward terms into three categories: task $r_t$, regularizer $r_r$, and style $r_s$. The task reward is computed from the distance to the target at the end of an episode, so the robot is free to choose the trajectory and gait as long as it reaches the target.

The task reward $r_t$ consists of the rewards to track the target position and target orientation.

$$r_t = r_{\text{pos}} + r_{\text{yaw}}. \tag{11}$$

The position tracking reward is defined using the target position in the base frame $x_b^*$

$$r_{\text{pos}} = (1 + \|x_b^*\|)^{-1}. \tag{12}$$

The orientation tracking reward is defined using the target yaw angle from the current yaw angle $\theta_b^*$

$$r_{\text{yaw}} = (1 + \|\theta_b^*\|)^{-1} \cdot (\|x_b^*\| \leq 0.25), \tag{13}$$

which is non-zero when the target position is tracked well.

The regularizer reward $r_r$ includes different components to regularize the behavior as well as to ease sim-to-real transfer

$$r_r = r_{\dot{a}} \cdot r_c \cdot r_\tau \cdot r_g \cdot r_{st}, \tag{14}$$

where $r_{\dot{a}}, r_c, r_\tau$ are used to regularize action rate, non-feet contact and large torques, $r_g$ is to regularize large roll and pitch angles of the base by comparing the projected gravity vector with the global one, and $r_{st}$ is defined to encourage the robots to have a minimum velocity (not stall) when the target position is far. Each of them is mapped by an exponential function $r_x = \exp\{-\|x/\sigma_x\|^2\}$, where $x$ is the corresponding value and $\sigma_x$ is a scaling factor.

The style reward $r_s$ comprises rewards aimed at guiding robots to adopt a specific style based on prior knowledge. Notably, these rewards are not essential to task completion

$$r_s = r_{ft} \cdot r_{mt} \cdot r_q, \tag{15}$$

where $r_{ft}$ assigns a higher reward when robots face the target, $r_{mt}$ motivates robots to move towards the target, and $r_q$ keeps all joint angles close to the default ones.

By setting large optimality ratios $\alpha_t, \alpha_r$ for the task and regularizer groups, we intuitively seek skills that can both track the target and have regularized motion that can be transferred to the real system. At the same time, we can diversify the locomotion style by setting different optimality ratios $\alpha_s$ to the style group. In principle, the fixed feature mapping $\phi(s)$ can be chosen arbitrarily, but a careful selection using either human or learned expertise can lead to favorable outcomes.

### 6.3 Training Details

All robots are simultaneously trained on terrains with uniformly sampled boxes, characterized by random sizes within the intervals [0.8, 2.0] meters for length, and [0.0, 0.2] meters for height. Prior to this, a curriculum of barrier terrains is implemented to warm-start skills with the ability to track targets and to climb on and off boxes of varying heights, which is consistent with the warm-start phase discussed earlier. The two terrain types are shown in fig. 5. The curriculum involving increased heights in barrier terrains is similar to the approach of Rudin et al. [2022b].

In addition to the warm-start phase, we also randomize the mass and friction coefficients, simulate random pushes on robots, and apply a 15 ms actuator delay to bridge the gap between simulation and reality.

We train a set of policies for 2000 iterations consisting of 48 simulation steps with 4096 parallel environments in Isaac Gym [Makoviychuk et al., 2021] including 800 iterations of warm-starting, which takes about 3 hours using GeForce RTX 3080Ti GPU. Similar to the previous work of Rudin et al. [2022a], we fix the episode length to 6 seconds and give the task reward $r_t$ in the last second.

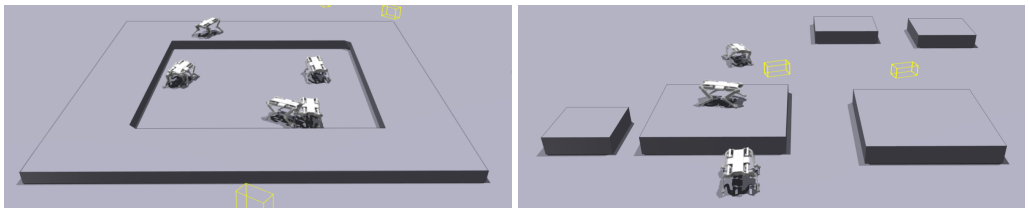

Figure 5: Terrain types used for training, *left*: barrier, *right*: box. Target positions are depicted as yellow boxes.

### 6.4 DOMiNiC

The complete pseudocode is given in algorithm 1.

**Algorithm 1** DOMiNiC

**Require:** $\pi$: Policy network, $v_i, \{v_e^j\}_{j=1}^m$: intrinsic and extrinsic value networks, $\psi(s)$: SFs network, $\mu$: Lagrange multipliers, $\psi^z$: feature expectations, $\bar{v}_e$: Moving average of extrinsic values.

Initialize networks, Lagrange multipliers, rollout buffer $\mathcal{B}$
**for** learning iterations = 1,2, ... **do**
    sample latent skill variable $z \sim p_z$
    **for** time step = 0,1,2, ... **do**
        collect transition $(s, a, s', \phi(s), \{r_e^j\}_{j=1}^m)$ with $\pi^z$
        compute $r_i$ using eq. (10)
        fill rollout buffer $\mathcal{B}$ with $(s, a, s', \phi(s), \{r_e^j\}_{j=1}^m, r_i)$
    **end for**
    compute TD targets for value update,
    estimate advantages $\{a_e^j\}_{j=1}^m, a_i^z$ for policy update
    **for** policy learning epoch = 1,2, ... **do**
        sample transition mini-batches $b \sim \mathcal{B}$
        compute aggregate advantage using $\sigma(\mu)$ in eq. (6)
        update $\pi$ and $v_i, \{v_e^j\}_{j=1}^m$ with PPO objective
        update SFs network $\psi(s)$ by the loss eq. (8)
        update feature expectations $\psi^z$
        update moving averages $\bar{v}_s$
        **if** not warm start **then**
            update $\mu$ by the loss eq. (7)
        **end if**
    **end for**
**end for**