Article

The International Journal of Robotics Research
2025, Vol. 44(8) 1396–1427

DOI: 10.1177/02783649241312698
journals.sagepub.com/home/ijr


Learning-based legged locomotion: State of the art and future perspectives

Sehoon Ha[1], Joonho Lee[2], Michiel van de Panne[3], Zhaoming Xie[4], Wenhao Yu[5] and Majid Khadiv[6]

Abstract

Legged locomotion holds the premise of universal mobility, a critical capability for many real-world robotic applications. Both model-based and learning-based approaches have advanced the field of legged locomotion in the past three decades. In recent years, however, a number of factors have dramatically accelerated progress in learning-based methods, including the rise of deep learning, rapid progress in simulating robotic systems, and the availability of high-performance and affordable hardware. This article aims to give a brief history of the field, to summarize recent efforts in learning locomotion skills for quadrupeds, and to provide researchers new to the area with an understanding of the key issues involved. With the recent proliferation of humanoid robots, we further outline the rapid rise of analogous methods for bipedal locomotion. We conclude with a discussion of open problems as well as related societal impact.

Keywords

Learning locomotion skills, quadrupedal locomotion, reinforcement learning for locomotion

Received 3 June 2024; Revised 10 October 2024; Accepted 21 November 2024
Senior Editor: Ioannis Poulakakis
Associate Editor: Koushil Sreenath

1. Introduction

Legged robots are complex systems with highly nonlinear, hybrid, and inherently unstable dynamics. Early demonstrations in the 1980s showed that simple feedback mechanisms could achieve robust dynamic locomotion for systems with one, two, and four legs (Raibert, 1984, 1986). This set the stage for more than forty years of research on legged locomotion. In particular, we have witnessed explosive progress in quadruped locomotion over the past few years (Figure 1). There are multiple reasons for this rapid progress, including reliable and affordable hardware, significant improvements in simulation environments, and scalable learning algorithms for high-dimensional continuous control problems. There has been further recent success with similar approaches for bipedal and humanoid robots. We believe this is therefore an opportune time to broadly summarize the efforts to date as well as to reflect on future research directions. This article aims to provide current and new researchers with a big-picture view of how the field has evolved to date. This review article may also provide a useful foundation for courses covering *learning-based legged locomotion*. We note that while many learning-based approaches can be applied more generically to arbitrary robot tasks, for example, including manipulation, we focus specifically on the challenges related to learning legged locomotion.

To place the remainder of the survey in a relevant context, we now provide a historical perspective of the key elements that have enabled the rapid advances seen for learning-based legged locomotion, with a focus on quadrupeds. This includes the evolution of the hardware, the physics-based simulators required for

[1]School of Interactive Computing, Georgia Institute of Technology, Atlanta, USA
[2]Neuromeka Co., Ltd, Seoul, Korea
[3]Department of Computer Science, University of British Columbia, Vancouver, Canada
[4]The AI Institute, Cambridge, USA
[5]Google DeepMind, California, USA
[6]Department of Computer Engineering, Technical University of Munich (TUM), Munich, Germany

Corresponding author:
Majid Khadiv, Munich Institute of Robotics and Machine Intelligence (MIRMI), Technical University of Munich (TUM), Georg-Brauchle-Ring 60-62, Munich 80992, Germany.
Email: majid.khadiv@tum.de

learning, and the most common methods and algorithms applied to learning-based control. We also briefly situate this survey itself with respect to other recent surveys.

1.1. Hardware

Hardware for quadrupedal locomotion has evolved considerably over the past few decades in terms of both capabilities and cost (Figure 2). Torque-controlled quadruped robots traditionally use one of three actuation mechanisms: hydraulic, electric motors with torque sensors, and series elastic actuators. The legged robots from the Leg Lab (Raibert, 1986) and later the success of the Bigdog project (Raibert et al., 2008) introduced hydraulics as an attractive actuation mechanism for quadrupedal locomotion. In particular, the large power-to-weight ratio enables hydraulically-actuated quadruped robots to perform highly dynamic motions and carry large payloads (Doi et al., 2006; Semini, 2010; Rong et al., 2012; Junyao et al., 2013; Semini et al., 2016). However, hydraulic systems are expensive and require specialized expertise for design, maintenance, and repair.

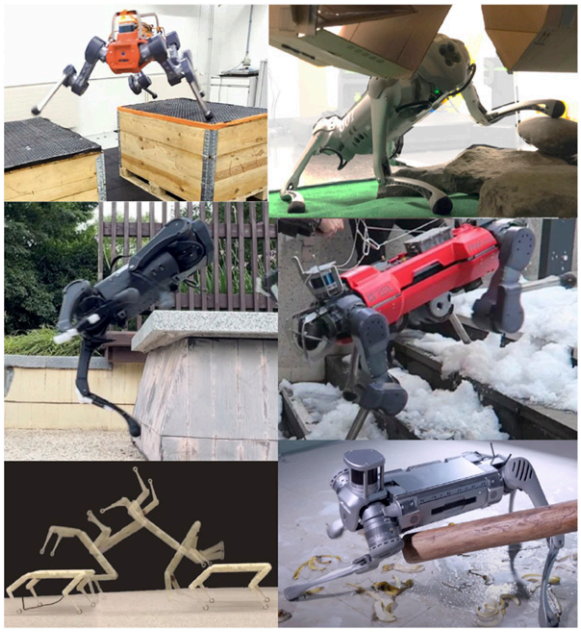

Figure 1. Examples of notable learned behaviors in the real world (Hoeller et al., 2023; Xu et al., 2024; Zhuang et al., 2023; Miki et al., 2022a; Li et al., 2023b; Unitree, 2021b).

Electric motors are the most common choice for actuation mechanism of quadruped robots (Buehler et al., 1998, 1999). To provide sufficient torque at the output of the joints with an electric motor, a gearing mechanism with a high ratio (such as harmonic drive) was traditionally required. However, the large static friction introduced by the gearbox renders the joints non-backdrivable which precludes output torque control by simply controlling the motor current. Classically, two modifications were introduced to enable joint torque control, namely using springs (Hutter et al., 2012, 2016) or joint torque/force sensors (BDI, 2015) after the gearbox. However, the former design limits the control bandwidth of the joint, and the latter is sensitive to large impact forces that could damage the gearbox or saturate (and damage) the force-torque sensor.

Using custom-made high torque-density actuators, the next generation of quadrupeds could control their interaction through direct joint current control with a low gear ratio and without a need for springs and torque sensors in the robot structure (Seok et al., 2014). This ability, sometimes called proprioceptive actuation (Wensing et al., 2017), revolutionized the field of quadrupedal locomotion. High torque density, high-bandwidth force control, and the ability to mitigate impacts through backdrivability are some of the most interesting features of these actuators. Thanks to these features, proprioceptive actuators have become the dominant paradigm in the design of legged robots, and a wide variety of quadrupeds have been developed based on this

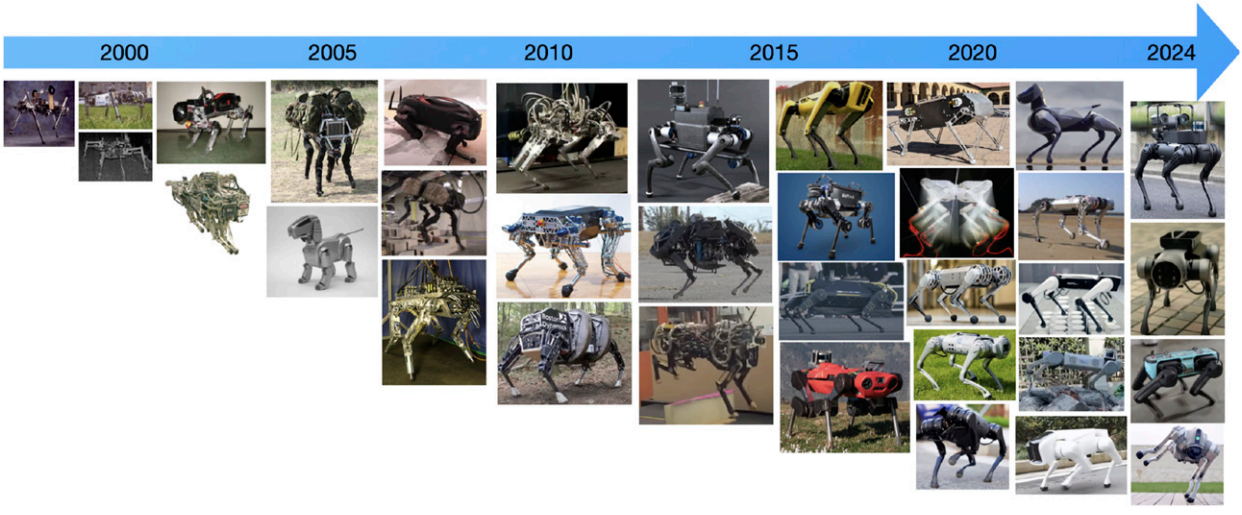

Figure 2. Evolution of quadruped hardware over time.

concept (Katz et al., 2019; Grimminger et al., 2020; Shin et al., 2022). In particular, the open-source initiatives from academia (Kau et al., 2019; Grimminger et al., 2020) as well as inexpensive hardware from industry (Deeprobotics, 2021; Unitree, 2021a; CyberDog, 2021) have dramatically accelerated experimental progress in quadrupedal locomotion. Among the many different robots, the Unitree A1 and Go1/Go2 are among the most popular choices for showcasing learned quadrupedal behaviors in the real world, due to their price and performance.

Bipedal robots and humanoids are currently seeing resurgent interest, given that much of our world is designed around the human form factor. The hardware is following a similar evolutionary path as that seen for quadrupedal robots. Early torque-controlled humanoids often used either hydraulic actuators (Smith, 2016; Schaal, 2018; Nelson et al., 2018), electrical motors with torque sensors (Englsberger et al., 2014), or series elastic actuators (Hubicki et al., 2016; Abate, 2018; Ahn et al., 2019). A more complete summary of humanoid hardware and software can be found in Goswami and Vadakkepat (2018). Recently, we are observing the same trend toward humanoids with proprioceptive actuators (Chignoli et al., 2021; Zhu, 2023). As with quadrupeds, these systems will become more affordable and capable. Toward the end of this survey, we revisit the key differences and opportunities of learned bipedal locomotion as compared to that for quadrupeds.

1.2. Simulators

The availability of highly efficient numerical algorithms for computing forward and inverse dynamics of articulated rigid body systems gave the robotics community with the necessary tools to simulate high-dimensional systems (Featherstone, 1987). These algorithms can solve forward and inverse dynamics problems for a high-dimensional articulated system such as a full humanoid robot in a fraction of a millisecond (Carpentier et al., 2019), if contacts do not exist. However, the bigger challenge for simulating legged locomotion is to include contact interactions.

Early simulation environments sometimes modeled contact using springs and dampers, also known as penalty-based methods (Marhefka and Orin, 1996; Yamane and Nakamura, 2006; Schaal, 2009). While being able to simulate compliant contacts, these models by default result in large penetrations that do not accurately reflect reality. To address this requires increasing the stiffness of the system and therefore using small simulation time steps that result in slow simulations. To avoid this *stiff* numerical behavior, more recent simulators rely on rigid contact models, which can be either elastic or inelastic. In this approach, contact is handled via a complementarity condition together with friction cone constraints (Smith, 2005; Todorov et al., 2012; Coumans and Bai, 2016; Lee et al., 2018b; Hwangbo et al., 2018; Makoviychuk et al., 2021). Access to fast simulation environments has been a key enabler for the success of deep reinforcement learning (DRL) for locomotion. GPU-friendly algorithms, for example, Makoviychuk et al. (2021), have enabled fast training of complex robotics tasks on a single GPU (Kim et al., 2021). Importantly, it is equally relevant to be able to simulate the sensors of a legged robot, including robot-mounted cameras, and thus simulators have also seen advances in this regard.

1.3. Control and learning algorithms

Early methods to control quadruped robots commonly relied on the use of simplified template models (Blickhan, 1989; Papadopoulos and Buehler, 2000; Kajita et al., 2001; Geyer et al., 2002) and simple-yet-effective heuristic-based strategies (Raibert, 1986; Pratt et al., 1997). With the introduction of biologically-inspired approaches such as the central pattern generators (CPG) (Ijspeert, 2001), a large body of work employed CPGs for generating oscillatory and cyclic behaviors for quadrupeds (Buchli et al., 2006; Buchli and Ijspeert, 2008; Ijspeert, 2008; Spröwitz et al., 2013; Barasuol et al., 2013). CPG-like parameterizations are still adopted in several learning algorithms to expedite the training process (Miki et al., 2022a; Bellegarda and Ijspeert, 2022; Ruppert and Badri-Spröwitz, 2022; Zhang et al., 2024) or to explain the motion of biological systems using robots (Shafiee et al., 2023). However, the enforced structure in these approaches limits the range of locomotion behaviors they can express, which can lead to sub-optimal performance.

Currently, the two dominant approaches to controlling legged robots in multi-contact scenarios are optimal control (OC) and reinforcement learning (RL) methods. OC uses a forward model of the system dynamics to help solve for a locally optimal control policy by minimizing a performance cost, typically over a finite future horizon (Raković and Levine, 2018). In contrast, RL solves for a state-indexed optimal policy by maximizing the expected reward based on collected samples from rolling out a control policy (Sutton and Barto, 2018). The common ground between the two approaches is that they ultimately estimate an optimal policy for a desired task, as described via a cost or reward function. In recent years, both OC and RL have shown great success in the control of quadrupeds.

Early work on the use of OC for quadrupedal locomotion cast the problem as a convex optimization problem, most typically as a quadratic program, either through using linear dynamics over a finite future time horizon (Kalakrishnan et al., 2011; Park et al., 2015; Di Carlo et al., 2018; Bledt and Kim, 2020) or for the current instant in time, acting as an inverse dynamics controller (Buchli et al., 2009; Mistry et al., 2010; Righetti et al., 2013; Hutter et al., 2014; Focchi et al., 2017). To enable planning and control for multi-contact behaviors through a holistic optimal control framework with contact, several formulations have been proposed, which are based on differential dynamic programming (DDP) with relaxed contact (Tassa et al., 2012), contact-invariant optimization (Mordatch et al., 2012),

contact-implicit optimization (Posa et al., 2014), mixed integer convex optimization (Deits and Tedrake, 2014; Aceituno-Cabezas et al., 2017), and phase-based parameterization of the end-effector trajectories (Winkler et al., 2018). While all of these approaches have shown impressive results in simulation, and some variants have become real-time capable for model predictive control (MPC) (Neunert et al., 2016; Farshidian et al., 2017; Neunert et al., 2018), their application in the real world has been limited. Most recent efforts to implement MPC on legged robots have focused on separating contact planning and whole-body motion generation and can achieve impressive behaviors on real hardware (Bledt and Kim, 2020; Li et al., 2021a; Mastalli et al., 2023; Grandia et al., 2023; Meduri et al., 2023).

While very successful, the use of optimal control and optimization-based control algorithms (Wensing et al., 2023) comes with multiple challenges. Solving a high-dimensional non-convex optimization problem (Wensing et al., 2023) in real-time is computationally expensive (Sleiman et al., 2021; Ponton et al., 2021; Mastalli et al., 2023; Grandia et al., 2023; Meduri et al., 2023). Dealing with uncertainties, especially in contact interaction is difficult (Tassa and Todorov, 2011; Drnach and Zhao, 2021; Hammoud et al., 2021; Gazar et al., 2023). The estimation and control problems are separated through the certainty equivalence principle, and the direct inclusion of different sensor modalities, such as vision, into control policy is prohibitively difficult. These issues motivated efforts toward alternative learning-based approaches for the control of legged robots.

The use of reinforcement learning for generating stable locomotion patterns has a history of more than two decades (Hornby et al., 2000; Kohl and Stone, 2004). Early works commonly use hand-crafted policies with a few free parameters that are then tuned using RL on the hardware. Subsequent work employs the notion of the Poincare map to ensure the cyclic stability of the gaits (Tedrake et al., 2005; Morimoto et al., 2005). These approaches have enabled a humanoid robot to walk (Morimoto and Atkeson, 2009), but their underlying function approximation has limited expressivity, which limits their application. Later reinforcement learning for quadrupeds mostly use the framework of stochastic optimal control, for example, path integral policy improvement (PI^2) (Theodorou et al., 2010; Fankhauser et al., 2013).

The DARPA Learning Locomotion program led to a variety of learning-enabled solutions. All the teams were provided with the Boston Dynamics LittleDog (Murphy et al., 2011) such that they could focus on the software development. At the end of the program in 2009, four of six teams completed the challenge with satisfactory performance (Pippine et al., 2011). To tackle a wide range of locomotion tasks on rough terrain, Neuhaus et al. (2011) and Shkolnik et al. (2011) resorted to traditional planning and control algorithms. Other teams relied heavily on optimization with limited usage of learning techniques (Zucker et al., 2011; Zico Kolter and Ng, 2011; Kalakrishnan et al., 2011). Notably, Zico Kolter and Ng (2011) developed a hierarchical apprenticeship learning framework (Kolter et al., 2007) to approximate a cost map for path planning. Zucker et al. (2011) used inverse optimal cost-preference learning to reduce the laborious cost-tuning procedure for footstep planning. The most extensive use of machine learning was demonstrated by Kalakrishnan et al. (2011) where the authors developed a learning-from-demonstration framework to optimally select foothold choices using terrain templates. Overall, despite the name of the program, learning techniques only constituted a small portion of the control software.

Thanks to the success of deep learning in the past decade (LeCun et al., 2015), much of the focus of reinforcement learning has shifted to DRL. Since DRL approaches commonly require a large number of samples (trial-and-error experience with the system being controlled) to learn control policies, it is common practice to train the policies in simulation and then transfer them to the real world (Tan et al., 2018; Hwangbo et al., 2019). In contrast with MPC, DRL approaches make extensive use of offline simulation of the world to learn a policy. Once learned, the online computation for the deployed policy consists of a simple forward pass through the neural network (often referred to as "inference") at each control time step, which is fast and efficient to compute. Furthermore, while training policies using DRL, robustness to many types of uncertainty can be achieved by adding noise during training to the relevant robot parameters and environments (Lee et al., 2020; Bogdanovic et al., 2022). Finally, it is simple to directly incorporate arbitrary sensory data as input to the policy, including high-dimensional modalities such as vision. This largely eliminates the need for a separate state estimation module (Miki et al., 2022a; Agarwal et al., 2023), as state estimation becomes an implicit part of the control policy. These unique features have led to the success of DRL in controlling quadruped robots on various terrains and under a wide range of conditions.

1.4. Related survey papers

Several existing survey papers have some overlap with the scope of our survey. The most relevant is Zhang et al. (2022) which briefly surveys the use of DRL for quadrupedal locomotion. Another recent survey paper (Bao et al., 2024) reviews the progress of DRL for bipedal locomotion. In comparison, this paper aims to provide a more comprehensive overview of the methods and complexities in this area. We cover not only DRL but also other types of learning for quadrupedal locomotion, such as imitation-based learning methods. We provide a historical perspective and further describe how learning-based approaches complement the rich literature on model-based control methods for legged locomotion. We further discuss ongoing advances and open directions. Complementary to our survey, Wensing et al. (2023) focus on optimal control methods for

locomotion, while Darvish et al. (2023) review works on the teleoperation of humanoid robots. The survey provided in Ibarz et al. (2021) has a focus on DRL for manipulation. While the focus in this survey is on quadrupeds and bipeds, the bulk of the foundations and algorithms are also relevant for robots with an arbitrary number of legs, from one-legged (Bogdanovic et al., 2020; Bussola et al., 2024) to six (Schilling et al., 2021) and eight-legged robots (Li et al., 2020, 2021b).

Advances in simulators and simulation environments have been key to enabling deep learning for robotics. Numerous high-performance open-source simulators are now available and are accelerating progress (e.g., Coumans and Bai, 2016; Makoviychuk et al., 2021; MJX, 2023). Multiple recent works have extensively reviewed the state of the art in this area and provide detailed comparisons (e.g., Collins et al., 2021; Kim et al., 2021; Liu and Negrut, 2021).

2. Theoretical background

2.1. Markov decision process and reinforcement learning

For many decades, machine learning (ML) has demonstrated promise for solving complex problems across a wide range of domains including computer vision, natural language processing, finance, environmental science, and many more. In robotics, RL (Sutton and Barto, 2018) stands out as a popular choice because the control of robots can naturally be viewed as a sequential decision-making problem. RL is one of the ML paradigms that seeks to develop an intelligent agent in dynamic environments to maximize cumulative rewards.

Typically, RL models the control problem using the Markov decision processes (MDP), which is defined as a tuple $(\mathcal{S}, \mathcal{A}, p, r, \gamma)$ of the state space $\mathcal{S}$, action space $\mathcal{A}$, deterministic $\mathbf{s}_{t+1} = f(\mathbf{s}_t, \mathbf{a}_t)$ or stochastic transition function $p(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t)$, reward function $r(\mathbf{s}_t, \mathbf{a}_t, \mathbf{s}_{t+1})$, and a discount factor γ. By executing a policy $\pi(\mathbf{a}_t|\mathbf{s}_t)$ in a stochastic environment, we can generate a trajectory of state and actions $\tau = (\mathbf{s}_0, \mathbf{a}_0, \mathbf{s}_1, \mathbf{a}_1, \ldots)$. We denote the trajectory distribution induced by π as $\rho_\pi(\tau) = p(\mathbf{s}_0)\prod_t \pi_t(\mathbf{a}_t|\mathbf{s}_t)p(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t)$. Our goal is to find the optimal policy π^* that maximizes the sum of expected returns:

$$J(\pi) = \mathbb{E}_{\tau\sim\rho_\pi}\left[\sum_{t=0}^{T} r(\mathbf{s}_t, \mathbf{a}_t)\right]. \quad (1)$$

In basic movement scenarios that involve navigation over a flat ground plane, the MDP state is given by the state of the robot alone, that is, the information needed to describe the positions and velocities of all the links of the robot. In more interesting and complex scenarios, the MDP state should encapsulate both the state of the robot and the state of the surrounding environment. In practice, many variations of MDPs are commonly used in order to accommodate diverse scenarios. For example, partially-observable MDPs (PoMDPs) capture the need to make control decisions using incomplete observations instead of the full MDP state, as defined by the observation space $\mathcal{O}$ and the observation function $o(\mathbf{o}_t|\mathbf{s}_t)$. The PoMDP formulation is common in robotics, where a robot camera can only observe part of the world at any given time, for example. The available observations are sometimes directly referred as the state, $\mathbf{s}_t$, which is a slight abuse of notation that allows policies to always be described as performing a state-to-action mapping, even when the actual mapping being used is from observations-to-actions.

Researchers have developed various RL algorithms to address MDP problems (Figure 3). Early algorithms can be broadly categorized into value-function and policy iteration methods. Value-function methods, exemplified by Q-learning (Watkins and Dayan, 1992), SARSA (Rummery and Niranjan, 1994), or (Fitted) Value Iteration (Bellman, 1966; Boyan and Moore, 1994), aim to estimate the expected values of states or state-action pairs, which implicitly define the optimal policy. Conversely, policy iteration methods, whether gradient-based (Howard, 1960) or gradient-free (Hansen et al., 2003), frame the given MDP as

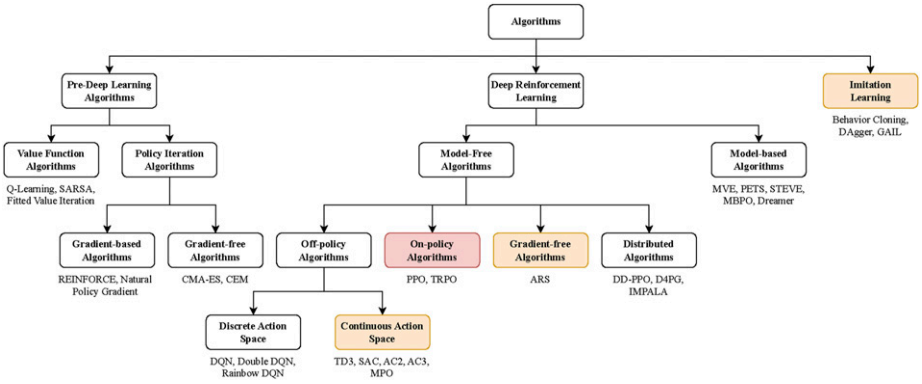

Figure 3. Taxonomy of reinforcement learning algorithms. Popular algorithms are highlighted in color. On-policy algorithms (red), such as TRPO or PPO, have been the most frequent choice for legged locomotion. Other off-policy, gradient-free, or imitation learning algorithms (orange) are also selected for sample efficiency or to reproduce styles from example data. For further details, please refer to the background section.

an optimization problem and seek to find the best parameters from a policy standpoint. These algorithms have found successful applications in diverse robotics problems, including legged locomotion.

Despite promising results, these early algorithms encounter challenges in solving complex, large-scale control problems, mainly due to scalability and convergence issues. Nonetheless, the established theoretical advancements demonstrate significant performance enhancements when combined with deep learning, which opens a new era of research.

2.2. Deep reinforcement learning

Deep learning (LeCun et al., 2015; Goodfellow et al., 2016) stands out as one of the most significant breakthroughs in ML. Artificial neural networks, composed of layers of neurons, are recognized as universal function approximators that can learn any multi-variate functions (Hornik, 1991; Cybenko, 1989). Various researchers have demonstrated that neural networks can effectively solve complex regression problems by processing vast amounts of data if they are provided with appropriate network architectures and learning frameworks.

The pioneering work of Mnih et al. (2015) demonstrated that RL can also be combined with deep learning to achieve human-level performance for Atari games. This achievement was made possible by introducing replay buffers and target networks to stabilize learning. Following Deep Q Network (DQN), several innovations have emerged to improve convergence and sample efficiency (Wang et al., 2016; Van Hasselt et al., 2016; Hessel et al., 2018). While many algorithms addressed simple discrete actions, robotic control usually requires continuous action spaces to generate motor commands. Deep deterministic policy gradient (DDPG) (Lillicrap et al., 2015) was one of the first algorithms supporting continuous actions, followed by numerous other RL algorithms and variations (e.g., Fujimoto et al., 2018; Haarnoja et al., 2018b; Mnih et al., 2016; Abdolmaleki et al., 2018; Schulman et al., 2015, 2017). Model-based RL holds the premise of achieving the best sample efficiency by utilizing the learned world model of the given MDP either in state space (Nagabandi et al., 2018; Feinberg et al., 2018; Buckman et al., 2018; Chua et al., 2018; Janner et al., 2019) or image space (Ha and Schmidhuber, 2018; Hafner et al., 2019b, a). Nevertheless, recent model-free approaches (Chen et al., 2021; Hiraoka et al., 2021) have also shown competitive sample efficiency. Besides sample efficiency, there have been numerous efforts to improve the scalability of RL algorithms for both gradient-based methods (Espeholt et al., 2018; Barth-Maron et al., 2018; Wijmans et al., 2019) and gradient-free approaches (Mania et al., 2018).

In the domain of learning-based locomotion, on-policy model-free algorithms like trust region policy optimization (TRPO) (Schulman et al., 2015) and proximal policy optimization (PPO) (Schulman et al., 2017) are often preferred choices. This preference arises from a pursuit of optimal-and-robust performance while being less concerned with sample efficiency. PPO has been a particularly popular choice in the legged locomotion community due to its excellent convergence and adaptability to diverse policy architectures. Table 1 provides a detailed taxonomy of the most common RL algorithms, and identifies those most commonly used for learning legged locomotion.

It is important here to clarify one common misconception when the term *model-free* is used in the context of DRL in robotics. While none of the model-free DRL algorithms need a model of physics of the world per se, in practice these policies are trained in simulation environments that are physics models. This means that application of model-free DRL methods relies heavily on models (simulation environments) that are developed using first principles. Hence, the term *model-free* in the context of DRL in robotics should be taken with a grain of salt.

2.3. Behavior cloning and imitation learning

As mentioned, the main goal of RL is to find control policies that maximize the cumulative reward, in a model-based or

Table 1. Common algorithms and use-cases for legged locomotion. A popular option has been on-policy RL, such as TRPO or PPO, due to their convergence to the best performing policies. However, other algorithms have been used for other reasons, such as better explotation, sample efficiency, or scalability.

			Example papers
Reinforcement learning	Off-policy Learning	DDPG	Bogdanovic et al. (2020); Huang et al. (2022a)
		SAC	Haarnoja et al. (2019); Ha et al. (2020); Smith et al. (2023)
	On-policy Learning	TRPO	Hwangbo et al. (2019); Lee et al. (2020); Yang et al. (2022b)
		PPO	Iscen et al. (2018); Tan et al. (2018); Peng et al. (2020); Kumar et al. (2021); Margolis et al. (2021); Rudin et al. (2022b); Xie et al. (2022); Miki et al. (2022a); Zhuang et al. (2023); Agarwal et al. (2023); Liu et al. (2024)
	Model-based	Dreamer	Wu et al. (2023b)
	Gradient-free	ARS	Yu et al. (2020, 2021)
Imitation learning	Behavior cloning		Lee et al. (2020); Kumar et al. (2021); Liu et al. (2024); Reske et al. (2021)
	GAIL		Escontrela et al. (2022)

model-free fashion. Either way, we assume that the reward function is given to us and the algorithm collects data to learn a model (model-based), or a value/policy function (model-free). While this has shown to be very promising in practice, it requires an immense amount of reward engineering, in particular for problems with sparse rewards.

Learning to imitate actions from an expert human demonstration or an existing expert policy (Pomerleau, 1988) can be seen as a remedy to this problem. However, naively mimicking the actions of the expert often fails due to the compounding nature of errors in control problems and the resulting distribution mismatches between the states visited during training and those seen at runtime. To mitigate this issue, researchers have explored alternative approaches for developing robust policies, such as dataset aggregation (DAgger) (Ross et al., 2011) or generative adversarial imitation learning (GAIL) (Ho and Ermon, 2016). One notable usage of IL in learning-based locomotion is privileged learning (Chen et al., 2020), which first learns a capable teacher policy with ground-truth information and copies the behaviors into a student policy with a realistic sensor configuration. Given its importance, we will revisit this particular strategy later in Section 4.4.

3. Components of MDP for locomotion

As mentioned in Section 2, RL algorithms typically frame sequential decision-making problems as MDPs. In this section, we will provide a brief summary of common practices in MDP formulation for legged robot control problems, including dynamics, state/observation, reward, and action space.

3.1. Dynamics

A legged robot can be described as a (typically deterministic) dynamical system, with the equations of motion given by $\dot{\mathbf{s}} = f(\mathbf{s}, \mathbf{a})$, where $\mathbf{s}$ is the state of the robot and the environment, and $\mathbf{a}$ is the action taken by the robot. To formulate it as an MDP, we discretize the dynamics with respect to time, $\mathbf{s}_{t+1} = \mathbf{s}_t + f(\mathbf{s}_t, \mathbf{a}_t)dt$, with dt being the discretization step. In MDP notation, this is modeled using a stochastic transition function $p(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t)$ having the probability mass lumped at $\mathbf{s}_{t+1}$. The state-transition data tuples are generated from the dynamics, $(\mathbf{s}_t, \mathbf{a}_t, \mathbf{r}_t, \mathbf{s}_{t+1})$, and used for the RL algorithms.

3.1.1. Simulators. Since DRL requires a large number of samples to train a policy, it is common practice to learn policies in simulation and then directly deploy them in the real world. As noted in Section 1.2, fast and accurate simulators have played a critical role in recent advances in learning locomotion behaviors.

State-of-the-art simulators used in robotics commonly rely on a rigid contact (elastic or inelastic) assumption. They model contact using a complementarity condition together with friction cone constraints (linear or nonlinear) (Todorov et al., 2012; Coumans and Bai, 2016; Hwangbo et al., 2018; Makoviychuk et al., 2021). Also, there has been a tremendous effort toward making these simulation environments differentiable (Geilinger et al., 2020; Le Lidec et al., 2021; Howell et al., 2022) which would make it possible to perform efficient system identification through contact or to directly update the parameters of a policy via back-propagation through time. Multiple recent works have extensively compared the simulation environments and we refer the readers to those works for a complete survey (Collins et al., 2021; Kim et al., 2021; Liu and Negrut, 2021) and to (Lidec et al., 2023) for an extensive survey of the multitude of methods used by simulators to solve for (rigid) contact in robotics. The GPU-friendly nature of Isaac Sim (Makoviychuk et al., 2021) allows for collecting state-transition data at a rate of nearly one million per second, making it especially interesting for DRL (Kim et al., 2021). Relatedly, the new version of MuJoCo includes MuJoCo XLA (MJX) which makes it GPU-compatible via the JAX framework (MJX, 2023). The trend of recent simulator usage for learning-based locomotion is illustrated in Figure 4, which captures early-developed communities for PyBullet and MuJoCo and also highlights the recent rise of Isaac Sim.

Relying on a rigid contact model that disallows any compliance or penetration is a limitation when dealing with tasks where interacting with a compliant environment is essential. Recently, researchers in both fields of locomotion and in-hand manipulation have looked into more advanced contact models to better capture the complexity of compliant interactions (Khadiv et al., 2019; Masterjohn et al., 2022; Choi et al., 2023). For instance, Khadiv et al. (2019) propose a nonlinear compliant contact model that can handle both rigid and soft interaction scenarios. In Masterjohn et al. (2022), the idea of pressure field contact patches is re-visited to foster their use with available velocity-level time steppers which enable simulation of compliant interactions with real-time rates. In Choi et al. (2023), on the other hand, an analytical nonlinear point contact model is used. To match the behavior of the model with the real-world data, the authors randomized the

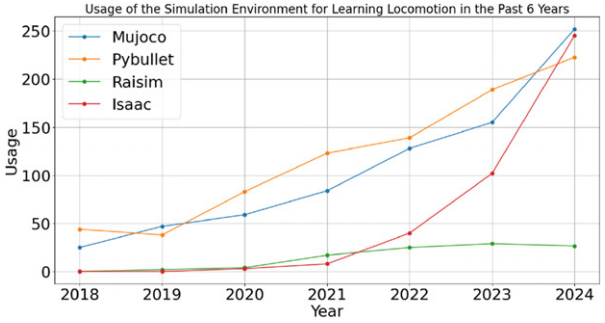

Figure 4. A rough estimation of the usage of various simulators for locomotion learning. The numbers are collected using a Google Scholar search with the following keywords "simulator name" "legged" "robot" "hardware" and "learning".

stiffness and damping ratio of the terrain such that the distribution of the simulation results matched the experimental distributions. To enable interaction with a wide range of environments and objects, a crucial component for learning-based algorithms will be reliable and fast simulation environments that can model a wide range of physical interactions.

3.1.2. Real robot. Another possible choice for generating motion data is to directly use the real robot to record the reward (Saggar et al., 2007) or the resulting transitions $\{\mathbf{s}_t, \mathbf{a}_t, \mathbf{s}_{t+1}\}$ (Haarnoja et al., 2019; Choi and Kim, 2019; Yang et al., 2020). This approach can potentially guarantee performance in the real-world environment by avoiding the need to build a model of the system and ensuring the fidelity of the data being used to train our ML model. However, collecting such data can be costly, which greatly limits its scalability. In addition, safety becomes a more important issue, especially for legged robots, because their underactuated base makes them vulnerable to falls and reliable mechanisms do not exist for resetting to a known initial state to begin every new episode. Therefore, additional mechanisms are required to mitigate the safety issues, such as safety-aware RL (Ha et al., 2020; Yang et al., 2022b), automatic resetting (Luck et al., 2017; Ha et al., 2018), failure recovery (Smith et al., 2022), or simulation-based pre-training (Smith et al., 2023).

3.2. Observation

One crucial aspect of MDPs is the observations and how to infer the states from them. The observation space is comprised of noisy sensory measurements that give information about the state of the robot and the environment. In the following, we categorize the body of works in the field of locomotion based on the types of observations they use for learning policies.

3.2.1. Proprioception. The set of sensors that provide information about the internal state of the robot is called proprioception, like IMU, joint encoders, or contact sensors. In legged robotics, raw measurements from these sensors are typically not directly used due to potential noise and inaccuracies. Instead, a state estimator is typically used to estimate crucial states of the robot such as base pose, base twist, joint position, and velocity. Such states are often referred to as *proprioceptive states* in DRL works, including Lee et al. (2020), Nahrendra et al. (2023), Yang et al. (2021), and Yu et al. (2023b).

Recently, Ji et al. (2022a) and Lee et al. (2024) have used sequences of IMU measurements and joint encoder data directly, instead of relying on pre-estimated velocities and poses. Specifically, Ji et al. (2022a) implemented an RNN-based state estimator, diverging from traditional model-based estimators, to directly compute the base velocity and contact states. On the other hand, Lee et al. (2024) employed a privileged training strategy, effectively utilizing the raw sensor data sequences to enhance the robustness.

The emerging research in legged robotics highlights the pivotal roles of joint positions, base pose (often represented by the gravity vector), and base linear and angular velocities in developing robust locomotion skills (Yu et al., 2023b). A significant body of work, including Lee et al. (2020), Ji et al. (2022a), and Haarnoja et al. (2019), emphasizes the importance of leveraging a history of proprioceptive states and joint command information. This approach addresses the non-Markovian characteristics of robotic systems that stem from hardware latencies and partial observations. Using only the current state is typically insufficient due to state estimation errors and partial observability issues in the real world; therefore, a short history buffer is almost always employed. In particular, historical data of proprioceptive states provide critical insights into foot-terrain interactions and external disturbances, as demonstrated by Lee et al. (2020), Ji et al. (2022a), and Li et al. (2024b). Also, Peng et al. (2018b) examine the benefits of including history inputs, demonstrating that under dynamics randomization, a policy with memory outperforms the ones without.

3.2.2. Exteroception. The sensory layer of the robot that provides information about the surrounding environments is normally referred to as exteroception. This information is in particular vital when the robot moves on non-flat environments. Conventional approaches in legged robotics often rely on explicit mapping techniques to pre-process the measurements before they are used by the controllers. For example, methods like elevation mapping (Miki et al., 2022b) or voxel mapping (Oleynikova et al., 2017; Besselmann et al., 2021) are commonly employed. Early works on perceptive locomotion (Miki et al., 2022a; Xie et al., 2022) adopted elevation mapping, where height values around the robot were sampled and used as observation to the policy.

Recent works removed such explicit mapping and instead used raw sensory readings directly as input to the policy, such as depth images or point clouds. This shift is driven by the necessity to effectively handle highly dynamic situations such as parkour (Zhuang et al., 2023) or obstacle avoidance (Yang et al., 2021), and tasks requiring high-resolution perception, such as stepping stone scenarios (Duan et al., 2022; Zhang et al., 2023). This approach not only facilitates more responsive and accurate locomotion in complex environments, but also reduces the likelihood of errors and inaccuracies that often arise from mapping failures.

RGB data can also be used for a more evolved perception of complex scenes that go beyond mere geometric information, enabling sidewalk navigation and obstacle avoidance (Sorokin et al., 2022). This modality includes a richer context, recognizing and responding to elements that cannot be captured through geometry alone, such as textures and colors. Semantics of the scenes can also be learned for more efficient navigation. For instance, Yang et al. (2023c) utilize semantic information from images to adapt the locomotion gaits and speed of a quadruped to handle different terrains.

Margolis et al. (2023) further augment RGB data with proprioceptive data for better traversability estimation.

Instead of feeding the sensory values directly into the policy, a compressed representation of them can be learned. For example, Hoeller et al. (2021) and Yang et al. (2023b) use unsupervised learning to obtain a latent space that can compress and reconstruct the original camera images, enabling learning policies that can navigate complex terrains automatically.

3.2.3. Task-related inputs (goals). Apart from proprioception and exteroception, other information specific to the robot's task, including command inputs like velocity and pose, or more complex data representations such as learned task embeddings can be included as input to the policy. Note that these inputs can be seen more as goals than observations, but to not overload the sections in the paper, we cover them under observation.

Velocity commands are frequently used to steer locomotion. Pose commands, specifying particular position or orientation targets, are also employed (Rudin et al., 2022a). In addition to direct command inputs, some approaches involve the use of learned task embeddings, which could be a latent representation of a specific reference motion (Peng et al., 2022) or any latent space that can direct the low-level behavior (Haarnoja et al., 2018a). For robotic systems employing structured action spaces, such as those based on CPGs or trajectory parameters, task-related information can also include specific details about the desired phase or trajectory patterns (Iscen et al., 2018; Lee et al., 2020).

Some methods consider future reference trajectories as input to the policies. For example, Ma et al. (2022) use planned end-effector trajectories of a manipulator as additional input for a quadrupedal robot with an arm, allowing the policy to adjust whole-body motions in anticipation of the arm's movements. Gangapurwala et al. (2022) and Jenelten et al. (2024) provide planned footholds and trajectories as references to train a tracking controller. This approach, where model-based controllers supply motion trajectories as references to the policy, enables more versatile solutions to complex tasks like navigating stepping stones. However, it also increases complexity and engineering effort due to the need for additional planning and coordination between controllers. Consequently, many recent works on dynamic locomotion prefer to learn a single independent control policy without any reference trajectory (Hoeller et al., 2023; Zhuang et al., 2023; Cheng et al., 2023c).

3.3. Reward

The reward function plays a crucial role in delivering the desired behavior by the RL algorithm, as it defines the objective of the agent. A common approach is to formulate the reward function as a linear combination of various reward and penalty terms, denoted as $r(\mathbf{s}_t, \mathbf{s}_{t+1}, \mathbf{s}_t) = \sum_i c_i r_i(\mathbf{s}_t, \mathbf{s}_{t+1}, \mathbf{a}_t)$, while it is also possible to use multiplicative combination (Kim et al., 2022). There are various methods to define each reward component.

3.3.1. Manual reward shaping. Manual reward shaping involves defining each reward component r_i and tuning each weight c_i by the engineer (hence a.k.a. reward engineering). Some examples of these components are velocity tracking, pose tracking, and other regularization terms. Also, normally the physical constraints of the system are added as a cost to this function, for example, limiting the magnitude of joint velocity, acceleration, and base pose during locomotion.

There is no general set of rules one can follow to define the components of the reward function. One common practice is to use bounded functions for stable training. This is often done by simply clipping or applying exponential kernels. For instance, exponential functions such as $exp(-c\|e\|^2)$ or $exp(-c\|e\|)$, with e representing the error term and c being the shaping coefficient, are frequently employed for tracking tasks (Lee et al., 2020; Rudin et al., 2022b; Duan et al., 2022). A set of commonly-used physical quantities used to define reward components are summarized in Table 2. To make the shaping of the reward terms easier, recent works have looked into alternatives such as potential-based rewards (Jeon et al., 2023b).

3.3.2. Imitation reward. We usually have a rough idea of how a legged robot should move due to the biologically inspired nature of these robots. These behaviors can be obtained from motion capture data of humans and animals (or online videos). Reward signals can be designed using this information, which greatly reduces the engineering effort required.

There exist many works (Peng et al., 2018a, 2020; Han et al., 2023; Yang et al., 2023a) that leverage dog motion

Table 2. Common physical quantities used to define rewards for training quadruped locomotion. Various norms and kernel functions are then applied to reward or regularize these terms. An asterisk ($*$) indicates a target quantity. e_z^b represents the Z-axis of the world in the base frame, q denotes the joint angle, τ represents the joint torque, and a_t is the action. Reward functions can be similarly defined using different norms or element-wise operations.

Horizontal velocity error	$v_{xy}^* - v_{xy}$
Yaw rate error	$\omega_z^* - \omega_z$
Base vertical velocity	v_z
Roll and pitch rate	ω_{xy}
z-axis deviation	$\|[0,0,1]^T - e_z^b\|$
Joint velocities	$\sum_{i\in\text{joints}}\|\dot{q}_i\|$
Joint accelerations	$\sum_{i\in\text{joints}}\|\ddot{q}_i\|$
Joint torques	$\sum_{i\in\text{joints}}\|\tau_i\|$
Joint mechanical power	$\sum_{i\in\text{joints}}\tau_i * \dot{q}_i$
Action rate	$\|\mathbf{a}_t - \mathbf{a}_{t-1}\|$
Action smoothness	$\|\mathbf{a}_t - 2\mathbf{a}_{t-1} + \mathbf{a}_{t-2}\|$

capture data to enable quadrupedal robots to learn animal-like motions, which is often approached by traditional model-based control approaches (Kang et al., 2021; Li et al., 2023d). In this motion imitation problem, the reward is designed to simply imitate the given reference trajectory, which can be obtained by either retargeting motion capture data of a real canine (Peng et al., 2020; Klipfel et al., 2023), model-based controllers (Reske et al., 2021; Fuchioka et al., 2023), or solving trajectory optimization (Bogdanovic et al., 2022; Kim et al., 2023c).

Instead of simply tracking the given reference motion, a set of motion clips can be used to define the style of legged robot movements, such as gait frequency and base motions. Generative adversarial imitation learning (GAIL) (Ho and Ermon, 2016) simultaneously trains a discriminator along with a policy, which distinguishes whether the motion is from the existing database or generated by a policy. Then, the learned discriminator can serve as a generic motion prior to realistic or stylistic movements, which can be co-optimized with downstream task rewards. The concept of adversarial motion priors (AMP) as a substitute for complex reward terms has also been used in multiple recent works (Escontrela et al., 2022; Wu et al., 2023a; Vollenweider et al., 2023; Li et al., 2023b,c). For periodic or quasi-periodic motions, one can also enforce more structure to extract spatio-temporal relationships in demonstrations (Li et al., 2024a).

3.4. Action space

Action space plays an important role in the performance of learned controllers for robotics, especially in the case of RL where exploration happens in the action space, the appropriate choice of action space can greatly improve exploration efficiency.

3.4.1. Low-level joint commands. Most works in quadrupedal learning use joint target position as the action space (PD policy). For each joint of the robot and a given action value $a(s)$ for that joint, the motor will attempt to generate a torque of $\tau = k_p(a(s) - \theta) - k_d\dot{\theta}$ to move the joint (k_p and k_d are the gains, while θ and $\dot{\theta}$ are the measured joint angles and velocities). In addition to its practicality, another reason this is the default choice may be attributed to Peng and Van De Panne (2017), where they show using joint position target as action space performs the best for physics-based character animation tasks, compared to other choices such as torque and velocity. Also, Bogdanovic et al. (2020) systematically showed on a one-legged hopping task that varying the gains of the controller as a function of states can improve the performance, when contact is highly uncertain.

More recently, Chen et al. (2023) and Kim et al. (2023a) also demonstrate successful learned quadrupedal and bipedal behaviors without imposing any structure in the action space and directly outputting torque (torque policy). The main benefit of this approach is that the policy function is not limited by the choice of the imposed structure as in the PD policy case. However, in this case, one needs to increase the frequency of policy evaluation at run-time (e.g., 1 KHz), whereas the PD policy can be updated much slower (e.g., 50 Hz) with torque still being produced at 1 KHz. Furthermore, it is argued in Bogdanovic et al. (2020) that the sim-to-real transfer for the PD policy is easier. It is important to note that, as it is also explained in Hwangbo et al. (2019), the PD policy is different from position control traditionally used in robotics. For position control, the controller tracks a *desired time-indexed trajectory* with large gains. However, no desired trajectory is tracked by the PD policy (note that the desired velocity is zero) and the target position is never achieved during the motion. In fact, both the torque policy and PD policy are used for torque control on the real robot.

3.4.2. Structured action spaces. Instead of directly controlling the robot via joint space control, many works also explore controlling the robot in the task space (feet for instance) (e.g., Krishna et al., 2022; Duan et al., 2021; Castillo et al., 2023). This allows for boosting learning efficiency and simpler control architecture. However, direct control in joint space still dominates the literature, mainly because it is more general and avoids singularities when inverse problems are solved.

Prior knowledge of how the robot should move can also be embedded in the action space via the use of residual RL (Johannink et al., 2019). In this approach, an open-loop reference control signal $\widehat{\mathbf{a}}_t$ is provided, and policies are learned to generate feedback signals, which are then added to the reference. For example, in Iscen et al. (2018), reference signals $\widehat{\mathbf{a}}_t$ is a sinusoidal wave over time that encourages the policy to generate desired gaits such as trotting and bounding, and the sum $\mathbf{a}_t = \widehat{\mathbf{a}}_t + \pi(\mathbf{s}_t)$ are used as the joint target for the low-level PD policy.

One can also learn to modulate parameters of high-level motion primitives. For example, Bellegarda and Ijspeert (2022) learn to modulate the intrinsic oscillator amplitude and phases of a central pattern generator. Xie et al. (2022) and Margolis et al. (2021) learn to output the desired center of mass accelerations, which are then used to generate motor commands using the single rigid body model. We will further detail these approaches in Section 6.

4. Learning frameworks

In the previous section, we discussed the common formulation of MDPs for legged locomotion. Solving the MDP is a highly challenging nonlinear optimization problem and thus there has been extensive research to efficiently solve this problem. In this section, we discuss popular choices of learning frameworks researchers have explored to solve the MDP for locomotion policies (Figure 5).

4.1. End-to-end learning

The most straightforward approach would be to treat the given MDP as a monolithic formulation and solve everything end-to-end using DRL algorithms. Among diverse

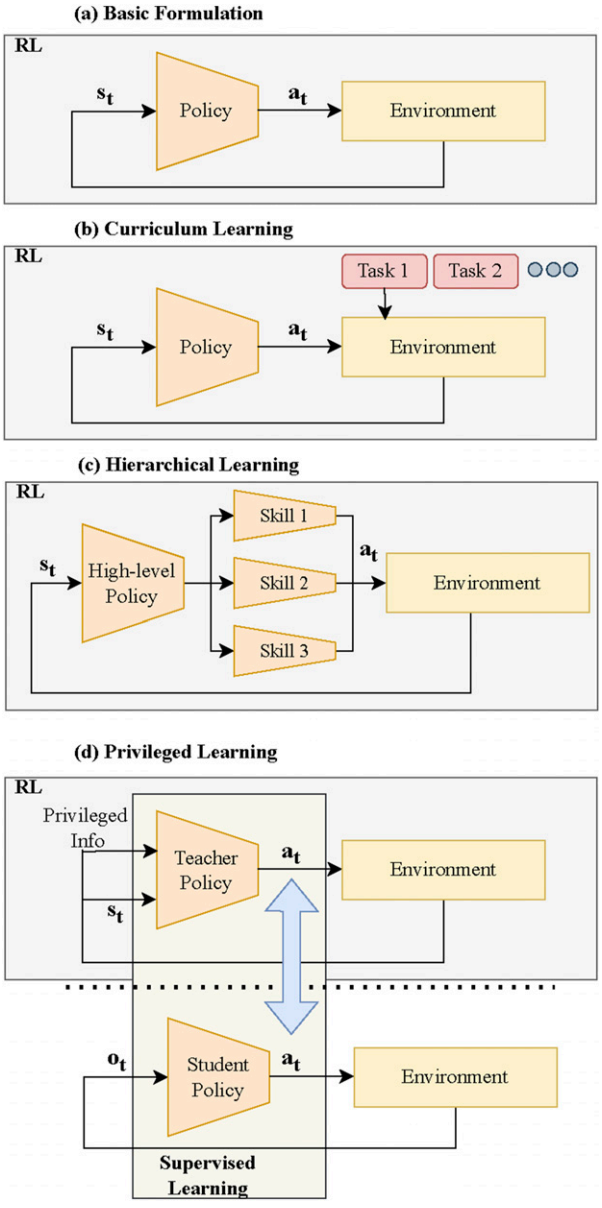

Figure 5. Illustration of popular learning frameworks: (a) basic learning, (b) curriculum learning, (c) hierarchical learning, and (d) privileged learning.

choices, the most popular DRL algorithm for solving legged locomotion is an on-policy algorithm, such as TRPO (Schulman et al., 2015) and PPO (Schulman et al., 2017). These algorithms are designed to take conservative update steps by constraining the parameter changes within the *trust region* defined in the vicinity of the current policy. As a result, these on-policy algorithms provide a robust learning framework that can reliably find high-performing policies at the end, which makes them suitable for challenging locomotion problems. However, off-policy algorithms, such as DDPG (Lillicrap et al., 2015) or SAC (Haarnoja et al., 2018b), have also been adopted by many researchers, particularly when the sample efficiency becomes a bottleneck of the problem.

End-to-end learning works well when the initial policy can effectively explore states that provide learning signals. However, for more challenging tasks and without any structure in the action space, the initial policy may not be able to obtain any useful reward signal and thus make it difficult for end-to-end methods to work well. In such cases, adding more problem-specific structures to the problem can be helpful, for example, curriculum and hierarchical learning.

4.2. Curriculum learning

Similar to the way schools create progressively challenging learning milestones to enhance educational outcomes, researchers have also explored a similar idea known as curriculum Learning (CL) for handling more difficult problems in robot learning. CL can be applied to different aspects of the learning framework. For example, Heess et al. (2017), Xie et al. (2020b), Rudin et al. (2022b), and Margolis et al. (2024) trained virtual and real agents in increasingly challenging environments and obtained policies that can handle environments that basic learning cannot achieve. CL has also been applied to improve the robustness of the policies by imposing an increasing amount of disturbances and randomness throughout the training (Akkaya et al., 2019). Constraints can also be enforced in a CL setting by starting with a soft constraint and gradually tightening it (Zhuang et al., 2023).

An important aspect of CL settings is to decide when to progress to the next stage of training and what the next stage should be. For locomotion problems on uneven terrains, the parameters used for terrain generation can parameterize the stages of the curriculum, for example, the steepness of a slope or the height of a stair (Rudin et al., 2022b). The stages of a curriculum can then be manually defined and the decision to progress to the next stage is usually determined by the capability of the policy, as measured by the sum of rewards it gets over one episode (Xie et al., 2020b) or the progress it can make over the current stage (Rudin et al., 2022b). Instead of manually defining the next stage of the curriculum, adaptive curriculum methods are also proposed to update the sampling distribution of the curriculum parameter. For example, in Xie et al. (2020b), the authors propose an adaptive curriculum strategy to update the sampling distribution by estimating the capability of the policies over a range of terrain parameters as measured by the value function. To enable high-speed locomotion skills for a mini-cheetah robot, Margolis et al. (2024) propose to adapt the sampling distribution of the velocity commands using the rewards it gets.

4.3. Hierarchical learning

Another common strategy to handle more challenging learning problems, especially ones with long horizons like navigation, is to decompose the problem into hierarchies. Typically, the problem is decomposed into high-level tasks

and low-level skills, where the action space of the high-level task will be part of the input of the low-level skills, and each level is learned separately.

Hierarchical learning is adopted in several learning-based control works in the computer graphics literature. The decomposition of the problem can be based on human intuition, for example, in Peng et al. (2017), a high-level policy will prescribe the desired footsteps to accomplish tasks such as navigation and soccer dribbling, while the low-level policy takes the desired footsteps as input to generate joint level control to follow the footsteps. The output of the high-level policy can also be a learned latent space. For example, Peng et al. (2022), Won et al. (2022), and Zhu et al. (2023) present frameworks where the low-level policies are trained to reproduce diverse motion capture data, driven by a learned latent representation; high-level policies are then trained to manipulate these latent spaces to accomplish high-level tasks.

Similar frameworks are adopted in several quadrupedal locomotion works. Han et al. (2023) learn low-level policies using the vector quantized variational encoder (VQ-VAE) (Van Den Oord et al., 2017). This allows a quadruped to reproduce a variety of motion capture data of animals and a high-level policy to manipulate the latent space of the VQ-VAE to realize navigation and multi-agent chase tag game. Ji et al. (2022b) divide a soccer shooting task into an end-effector motion tracking task and an end-effector trajectory planning task, where the output of the trajectory planning is used as input for the motion tracking policy. The two tasks are learned separately to accomplish quadruped soccer shooting. Ji et al. (2023) similarly learn task and recovery policies for soccer dribbling.

There are also hierarchical architectures combining a model-based control approach and a learning-based approach, where a control policy can either be learned or manually designed. We leave the discussion to Section 6.

4.4. Privileged learning

Robotic tasks in the real world are inherently partially observable, leading to challenges in achieving high-performance locomotion. For instance, accurately measuring or estimating factors like payload weight or friction coefficient is nearly impossible, but crucial for the policy to be robust or adaptive.

To address these challenges, existing approaches often involve constructing a belief state from a history of available measurements. In DRL, this is typically accomplished by stacking a sequence of previous observations (Haarnoja et al., 2018b; Hwangbo et al., 2019) or using models with memory such as RNN or TCN. However, training a complex neural network policy with such a large input space using RL from scratch can be time-consuming and challenging.

Chen et al. (2020) propose a strategy called *learning by cheating*, which was renamed to *privileged learning* in the context of quadrupedal locomotion (Lee et al., 2020). This approach leverages fully observable MDP in simulation to train a *teacher* policy, which is then distilled into a *student* policy based on sequence models like RNN or TCN. Specifically, the teacher has access to privileged information, such as the noiseless, rich simulated states of the environment (e.g., friction coefficient). In this context, Chen et al. (2020) collected human demonstrations in simulation, while Lee et al. (2020) trained a teacher policy for quadrupedal locomotion using noiseless robot states and terrain information as privileged information. At test time, the student policy has no access to such information, instead, it processes a history of measurements available from the robot.

Lee et al. (2020) first demonstrated the effectiveness of this approach for quadrupedal locomotion, enabling rough terrain traversal that outperformed conventional optimization-based approaches. By imitating the privileged teacher policy, the student policy constructs an internal representation of the world that conveys locomotion-related information. This idea of combining RL and imitation learning has been adopted by subsequent works to enhance robustness or utilize more complex input modalities such as raw images or voxel maps (Miki et al., 2022a; Haarnoja et al., 2024; Agarwal et al., 2023; Miki et al., 2024; Lee et al., 2024).

4.5. Starting point to train your Robot

Up to this point, we have provided all the information required to train a legged robot in simulation. To further help those new to the field, we give some further recommendations on where to start. In terms of algorithms, there are simple and intuitive repositories that explain (mainly RL) algorithms with code (e.g., Abel, 2019; Huang et al., 2022b). In terms of simulation, as suggested by Figure 4, Isaac and Mujoco are the best choices to start with. In particular, Mujoco open-sourced the simulation code which can help with understanding the components within a simulator. A wide range of URDFs for robots can be found in Caron (2022).

Starting from a robot model, one can build an environment for reinforcement learning. An initial set of observations can include base height and orientation, base linear and angular velocity, and joint angles and velocities. The action space can be set to be the target joint angle for a joint PD controller and the reward can be designed around tracking a desired body velocity. PPO (Schulman et al., 2017) is now the most popular algorithm for training locomotion policies for legged robots and there are many open-source implementations (e.g., Rudin, 2021b; Sidor, 2021). Several open-source codebases for learning quadruped locomotion are available. Some representative choices are listed here for the readers' reference: the implementation of Peng et al. (2020) can be found in Coumans (2020) in the Pybullet simulator (Coumans and Bai, 2016). It also contains an MPC-based controller similar to Bledt et al. (2018). Rudin (2021a) contains the implementation of Rudin et al. (2022b) in Isaac Sim (Makoviychuk et al., 2021). Most recent works

on legged robot learning are built on top of this, thanks to the efficiency of Isaac Sim.

5. Sim-to-real transfer

The promise of computer simulations to efficiently generate large amounts of training data creates a great synergy with modern deep learning algorithms that are capable of absorbing large-scale datasets and often require abundant data sources to achieve effective learning. However, a major challenge for policies that are trained using simulation data is whether they will perform well on real hardware due to the discrepancies between computer simulation and real-world robot dynamics, also known as the sim-to-real gap. It is not practical to construct a simulation model to accurately represent every aspect of the real world due to its complexity. For example, motors can heat up over time, showing a different behavior given the same command input, non-deterministic delays exist due to communication of different parts of the hardware, and the feet of the robots are usually made of soft materials instead of the rigid body model commonly used in most simulators. Furthermore, a policy trained entirely in simulation can take advantage of some aspects of the simulation that are not present in reality, resulting in direct failure upon direct deployment. In this section, we review some of the techniques commonly used to combat the sim-to-real gap.

5.1. Good system design

A naive implementation of the environment for locomotion tasks often leads to undesirable behaviors that take advantage of the peculiarities of the simulators. For example, a reward that encourages forward walking motion for the commonly used MuJoCo humanoid often generates control policies that resemble bang–bang control, with the joints oscillating at high frequency, resulting in behaviors that glide forward. Such a bang–bang control strategy is not suitable for hardware and sim-to-real is doomed to fail. A good design of the overall system helps constrain the policies to operate in the same distribution as the simulator and to combat the sim-to-real gap.

5.1.1. Reward design. The design of the reward function can directly affect the sim-to-real performance. In Section 3.3, we described the most common reward terms, but we reiterate some important elements here. To avoid jittery motions, joint acceleration is often penalized; a reward for foot air time is often used to avoid foot-dragging behavior; foot impact penalty is used to avoid stomping behaviors (Makoviychuk et al., 2021). Maintaining a good balance between task rewards such as velocity tracking and regularization rewards makes the reward shaping problem a laborious task.

5.1.2. Observation and action space design. The choice of observation and action space also plays an important role. Xie et al. (2021) identify several key factors that enable direct sim-to-real transfer for the Laikago robot, without randomization or adaptation techniques commonly used in other works. In particular, a low proportional gain for the joint PD controller that allows for compliant behavior greatly reduces sim-to-real gap, and the use of a state estimator that allows for using body velocity as part of the observation allows for rejecting velocity perturbation.

5.1.3. Domain knowledge. Domain knowledge can also be used to improve the system design. For example, in Xie et al. (2020a), symmetry constraints are used to promote the left-right symmetry of bipedal walking policies, greatly improving the motion quality and as a result greatly improving sim-to-real performance. Domain knowledge can also be used in the reward function to promote certain behaviors, for example, the use of motion capture data in the reward function can promote the style of the motions (Peng et al., 2020; Han et al., 2023) or using CPGs as a structure in the policy (Lee et al., 2020; Bellegarda and Ijspeert, 2022; Iscen et al., 2018) can lead to more desirable gait patterns.

With good system design, a few works demonstrated sim-to-real success for legged robot locomotion without dynamics randomization. For instance, Smith et al. (2023) demonstrate successful sim-to-real transfer of highly dynamic motions (such as jumping and walking on hind legs) on the A1 quadruped. Xie et al. (2020a) and Dao et al. (2020) successfully show sim-to-real for the bipedal robot Cassie. These examples demonstrate that a good design can sometimes help bridge the sim-to-real gap, without a need for any further strategies.

5.2. System identification

Even with good system design, inaccurate modeling or unmodeled dynamics in the physical system can cause direct sim-to-real to fail. To improve the fidelity of the simulator, system identification can be performed. A dominant source of modeling errors comes from the actuator dynamics. In simulation, motors can apply arbitrary torque profiles that a policy commands, while a physical actuator often produces less accurate torque profiles due to the limited bandwidth as well as limited tracking accuracy of the underlying motor controller. One of the earliest successful sim-to-real transfer for quadrupedal locomotion used an analytic actuator model (Tan et al., 2018) for a miniature quadruped. Later Hwangbo et al. (2019) used a fully black-box model (neural network) to learn an actuator model, which was then used in the simulation to train locomotion policies.

Another important source of the sim-to-real gap is the discrepancy between the contact model in simulation and the real-world interactions. Most of the existing simulators use rigid contact, while in reality there always exist some deformations in the interaction. However, it is quite challenging to simulate deformable terrains or those with complicated shapes, which often involves prohibitively

expensive finite-element methods. One notable success in using a more complicated contact model is Choi et al. (2023), which develops a compliant contact model to simulate diverse terrains, ranging from very soft beach sand to hard asphalt. Training with this model boosts the performance of the policy handling similar terrains in the real world compared to training with a rigid contact model.

5.3. Domain randomization

In addition to better system design and more accurate simulation, another important strategy to mitigate sim-to-real gap is to improve the generalization capabilities of the trained policies. One important approach to improve policy transfer is domain randomization, where the system parameters are randomized during training. This is similar to robust control, where a controller is optimized to perform robustly under a bounded set of system parameters and uncertainties (Dorato, 1987). A fundamental assumption behind domain randomization is that if we generate control policies that can handle sufficiently diverse training environments, it is more likely that this policy also work in the real world, even though it has never seen it during training. Another way to interpret this approach is that if we have a control policy that works for an ensemble of different environments, the real world will hopefully fall into one of these environments, hence the policy will work in the real world.

Early works in using domain randomization for robot learning applied the idea to the problem of manipulation to enable more robust perception modules (Tobin et al., 2017). In the legged locomotion field, Tan et al. (2018) were among the first to adopt the idea of domain randomization. Specifically, they randomized key dynamic parameters that are important for obtaining robust policies such as robot mass, friction coefficient, motor strength, and latency. This approach has since been widely adopted in follow-up works and combined with feed-forward trajectories (Iscen et al., 2018; Lee et al., 2020; Miki et al., 2022a), motion imitation (Peng et al., 2020), task-space control (Bellegarda et al., 2022), and policy distillation (Caluwaerts et al., 2023; Kumar et al., 2021) to further improve legged locomotion learning in different scenarios.

The aforementioned approach in domain randomization largely focused on bridging the physics gap between the simulation and the real world, where the resulting policies are often blind to their environments. As these techniques mature, recent works in legged locomotion explored randomizing also the simulated visual perception parameters such as camera intrinsics and extrinsics, and noise models to achieve reliable vision-based legged locomotion on highly unstructured terrains (Yu et al., 2021; Miki et al., 2022a; Margolis and Agrawal, 2023; Zhuang et al., 2023; Agarwal et al., 2023; Cheng et al., 2023c). Notably, Truong et al. (2023) reported that higher visual fidelity does not always lead to better performance due to slow simulation speed.

5.4. Domain adaptation

Like domain randomization, domain adaptation aims to develop generalizable policies that can cover real-world environments, with the main difference being that domain adaptation explicitly identifies the current scenario that the policy is operating within and adjusts the policy behavior to be optimal for the current scenario (similar to adaptive control). As a result, when successfully trained, domain adaptation-based policies can often achieve better performance and cover a wider range of scenarios than domain randomization techniques. Figure 6 demonstrates common components in domain adaptation algorithms.

One class of domain adaptation algorithms is to explicitly identify the parameters of the environment, which are used as input to the control policy or during policy training (Yu et al., 2017; Chebotar et al., 2019; Yu et al., 2019; Muratore et al., 2021; Lee et al., 2018a). To identify the environment parameters, Yu et al. (2017) learned models, Chebotar et al. (2019) optimized trajectory matching loss, and Yu et al. (2019) directly optimized task performance. Although explicitly identifying the environmental parameters enables an intuitive interpretation of the adaptation process, it is difficult to extend this idea to cases where a moderate number of environment parameters need to be adapted.

To mitigate this issue, researchers have opted for implicitly representing the environment parameters during the adaptation (Yu et al., 2020; Kumar et al., 2021; Peng et al., 2020; Lee et al., 2020), where the high-dimensional environment parameters are compressed into a low-dimensional latent representation. There is a variety of approaches for obtaining a good latent representation. For example, Peng et al. (2020) propose an optimization-based approach to find the latent environment representation for an estimated advantage value during policy adaptation. Kumar et al. (2021) and Lee et al. (2020), on the other hand, directly learn the latent environment representation during policy training, and later on train a separate system identification module to predict the latent representation during inference. Lee et al. (2022) leverage ideas from the representation learning field and use predictive information to acquire a state representation that captures key environment dynamic information. Due to the ability to handle larger amounts of environment variations, this class of domain adaptation

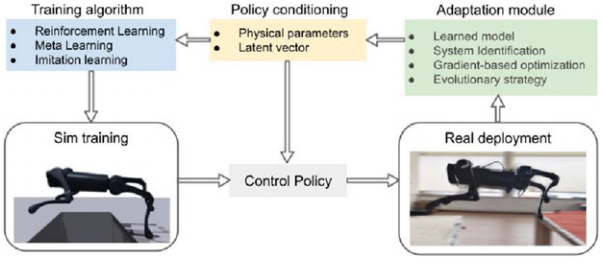

Figure 6. Common components in domain adaptation algorithms (Zhuang et al., 2023).

approaches has been more widely applied to real legged systems and has become one of the most popular classes of methods in recent years.

There are also works that learn to improve the simulation accuracy by using real-world data. The transition function in a simulation can be augmented in order to better match the real world. Policies trained with the augmented simulation are shown to demonstrate better sim-to-real performance (e.g., Golemo et al., 2018; Jeong et al., 2019; Jiang et al., 2021).

6. Combining control and learning

While both RL and OC aim at finding the optimal policy for a desired task, they have complementary features that attract researchers to devise frameworks that benefit from the best of both worlds. For instance, as model-based approaches use the knowledge of dynamics and constraints in the policy design, they transfer our knowledge of the world among different tasks. Furthermore, they can explicitly take constraints into account which makes them suitable for safety-critical cases. On the other hand, learning-based approaches can randomize the parts of the model with uncertainties and deliver a robust policy without much overhead in the policy design. Furthermore, the online computation is at a minimum, as training is done purely in simulation and one only needs to perform inference during execution.

This line of work can be split into four general categories: 1) learning some parameters of model-based controllers, 2) learning high-level policies to set commands for a model-based controller, 3) leveraging learning to improve the generalization and reduce online computation of model-based controllers, and 4) using model-based control to make learning more sample-efficient.

6.1. Learning control parameters

There are certain parameters in model-based control policies that directly affect the performance and robustness of the controller. Normally, tuning these parameters is done manually on the real system by the control designer. However, it is always desired to automate this process. To do that, Ponton et al. (2014) and Heijmink et al. (2017) use PI^2 to learn optimal gains for tracking a desired trajectory. Pandala et al. (2022) learn the boundaries of the uncertainty set that is then used to design a robust MPC controller.

To efficiently learn the control parameters, some recent approaches rely on Bayesian optimization (BO). Yeganegi et al. (2019, 2021) learn the cost weights of the trajectory optimization module and an MPC controller for humanoid locomotion, respectively. Marco et al. (2021) learn the joint impedance to track trajectories from a trajectory optimization module, while simultaneously learning the failure constraints. Yang et al. (2022a) use BO to learn the control parameters of a hybrid-zero-dynamics-based controller. As BO can only handle a small number of decision variables, Yuan et al. (2019) and Sarmadi et al. (2023) first perform dimensionality reduction and then apply BO to the low-dimensional latent space. To make the exploration in the real world safe, Widmer et al. (2023) apply a safe BO framework to tune both the gait and feedback control parameters in the real world.

6.2. Learning a high-level policy

A more sophisticated version of simply learning a few parameters of a model-based controller is to learn a policy that works in concert with a model-based controller. For instance, given a general map of the environment, finding foothold locations for locomotion is a highly challenging task for model-based controllers. Furthermore, model-based controllers are sensitive to the precision of their underlying model, hence a residual policy can correct for this model discrepancy.

Villarreal et al. (2020), Byun et al. (2021), Xu et al. (2022), and Omar et al. (2023) learn a contact planner that, given the terrain map, finds safe foothold locations and passes them to an MPC module to optimize ground reaction forces. Gangapurwala et al. (2022) train a policy that generates desired foot locations which are tracked using a model-based whole-body controller. Taouil et al. (2023) develop a footstep location planner for a given low-level controller that is conditioned on a desired velocity. Dhédin et al. (2024) leverage the power of diffusion models in handling multi-modal datasets to learn in a supervised fashion a policy that outputs contact plans for MPC. Gangapurwala et al. (2021) leverage DRL to learn a feedback trajectory corrector to update the output of trajectory optimization that is then stabilized using a whole-body controller. Yang et al. (2022c) learn a high-level policy to transition between different gaits. Viereck and Righetti (2021) and Xie et al. (2022) learn a centroidal trajectory planner that generates desired forces and body trajectories that are fed to an inverse dynamics controller for tracking.

6.3. Learning for efficient model-based control

While considering the model in the structure of the policy can result in a better generalization, it comes with the cost of more online computational burden. For instance, a model-based controller needs to plan online for a walking motion in real-time. However, it seems much more efficient to simply cache the solutions in an offline phase, for example, using a function approximator, to reduce the online computation. In this vein, Carius et al. (2020) and Reske et al. (2021) learn the control Hamiltonian function, whereas Wang et al. (2022), Viereck et al. (2022), and Kovalev et al. (2023) learn the value function of the optimal control problem. Lembono et al. (2020) and Dantec et al. (2021) use a large dataset from trajectory optimization to learn warm-starts for quick resolutions of the nonlinear program in the MPC. Yang et al. (2020), Bechtle et al. (2021), Anne et al. (2021), Li et al. (2021b), and Wu et al. (2023b) learn a forward model that is then used within a model-based

module to control the robot. Kwon et al. (2020) learn the solution map of the trajectory optimization problem to enable reactive planning in real-time.

6.4. Model-based control for efficient learning

The fourth category leverages tools from model-based control to efficiently learn locomotion skills. For instance, one can use trajectory optimization to guide the RL through high-reward parts of state space, for environments with sparse rewards. The final policy in these approaches is a neural network that sends actions to the joins.

Within this line of research, Gangapurwala et al. (2020) use optimal control to guide and constrain the exploration of the DRL algorithm. Tsounis et al. (2020) leverage trajectory optimization to evaluate gait transitions in the high-level controller in a hierarchical DRL setting. Green et al. (2021) leverage the power of reduced-order models for planning and learn a residual policy to compensate for the discrepancy between the reduced and full robot models. Bellegarda et al. (2020), Bogdanovic et al. (2022), Brakel et al. (2022), and Kang et al. (2023) use optimal control to generate demonstrations to guide DRL. Similarly, Jenelten et al. (2024) interactively query a trajectory optimization module to provide demonstrations for the DRL policy in different initial and final configurations. Fuchioka et al. (2023) investigate the use of feed-forward inputs from trajectory optimization to improve the RL learning efficiency and sim-to-real transfer. Khadiv et al. (2023) take a behavioral cloning approach to learn optimal policy from MPC directly in sensor space, while Pua and Khadiv (2024) introduce a more sample-efficient framework compared to behavioral cloning. Youm et al. (2023) use a similar approach, but improves the cloned policy using another DRL phase like Bogdanovic et al. (2022).

7. From quadrupeds to bipeds

Works using RL for bipedal locomotion dates back to as early as 2005, when policy gradient (Tedrake et al., 2005) was used to train a small biped to walk in under 20 minutes. Schuitema et al. (2010) use RL to train a 2D biped to walk in 2010. Hester et al. (2010) applied model-based RL to learn soccer penalty kicks for a NAO humanoid. These early works focused on learning directly on the physical robots and therefore were only applied to simplified bipeds, for example, a small biped with large feet or a biped with motion restricted to a 2D plane.

From early 2000 to the DARPA Robotics Challenge (2015) (DRC) in 2015, the dominant approach to control humanoid robots was OC or heuristic methods with simplified dynamics models. Early works on the use of OC for humanoids resorted to the simplified linear models of the dynamics to enable linear MPC (Kajita et al., 2003; Wieber, 2006). However, for multi-contact behaviors (making contact with the environment using both feet and arm), later works focused on solving the holistic problem using the robot full dynamics with contact (Tassa et al., 2012; Mordatch et al., 2012; Lengagne et al., 2013; Posa et al., 2014). These holistic approaches showed little success through a model predictive control fashion in real-world humanoid control (Koenemann et al., 2015). In DRC 2015, most teams used a combination of simplified models planning together with full-body inverse dynamics/kinematics (Feng et al., 2015; Kuindersma et al., 2016; Johnson et al., 2017). This DRC demonstrated that humanoid technology is by no measure close to being deployed for real-world problems and a paradigm shift is required (Murphy, 2015; Atkeson et al., 2018).

As DRL achieved success in physics-based character animation (Peng et al., 2018a) and quadrupedal locomotion (Tan et al., 2018; Hwangbo et al., 2019), researchers started to use DRL algorithms in physics simulations to train walking policies for the bipedal robot Cassie and successfully transferred them to the hardware (Xie et al., 2020a). Techniques such as domain randomization, more modern neural network architectures, and smarter reward design enabled dynamic and versatile behaviors that are robust to real-world perturbation (e.g., Siekmann et al., 2021; Radosavovic et al., 2024; Li et al., 2024b). More recent works explored how to incorporate perception to enable challenging terrain navigation, such as stepping stones (Duan et al., 2022) and uneven terrains (Duan et al., 2023). In addition, more expressive learning-based motions are recently shown on humanoid robots (Cheng et al., 2024).

In the years 2023 and 2024, we witness an explosion of new companies starting to develop humanoid robots, most of which have bipedal forms. The highlight of the exhibition section of the 2024 International Conference in Robotics and Automation (ICRA) was an immense increase in the number of electrically-actuated humanoids. In particular, Unitree has unveiled its new humanoid G1 with an incredibly low price, on par with manipulator arms. This marks the starting point of a new era where many research laboratories will have access to humanoid robots. We have seen this trend in the past decade that enabled exponential progress in the fields of drones and quadrupedal locomotion. Furthermore, due to the simplicity and ease of use of modern DRL algorithms and the availability of fast and parallelizable simulators, the barriers to generating and implementing new motions on highly complex humanoid robots are being removed. This has resulted in a new wave of pioneering demos from the industry, for example, backflip motion from Unitree H1 (Unitree, 2021c) and highly stylish motions of the Disney bipedal creature (Disney, 2023). We believe this trend will continue and we will see more and more of these impressive behaviors on real robots from different industrial players.

Parallel to the humanoid efforts in industry, researchers have developed open-source (Daneshmand et al., 2021) and affordable (Liu et al., 2022a) miniature bipedal platforms. These miniature robots are much easier to handle than full humanoid robots and could potentially impact the rate of progress of research on learning-based bipedal locomotion.

For instance, Haarnoja et al. (2024) develop a hierarchical system to enable learning of soccer plays on the miniature Robotis OP3 humanoid. Recently, the miniature robot from LimX has shown impressively robust bipedal locomotion behaviors in the wild (Dynamics, 2024). With affordable human-size humanoids becoming widely available, one would expect lessons learned from quadrupedal learning works to be transferred to humanoid robots, while also additional techniques are to be developed to handle the difficulties of balancing bipeds compared to quadrupeds.

8. Unsolved problems and research frontiers

As surveyed in the previous sections, there have been major developments in the field since the early work on the use of DRL for legged robots (Tan et al., 2018; Hwangbo et al., 2019). The significant progress enabled by learning-based approaches opens the door to new challenges. One major issue of current DRL algorithms is their need for heavy reward shaping for each skill and the lack of transfer of knowledge between skills. Unsupervised skill discovery is an interesting avenue for more research in this regard (Section 8.1). Another issue is the sample inefficiency of DRL algorithms, which begs for the use of higher-order gradients in policy optimization. Section 8.2 discusses how differentiable simulators can be leveraged to improve sample efficiency.

While DRL has enabled many robust behaviors in the wild, they struggle in safety-critical situations such as stepping stones and confined environments (Section 8.3). This will require handling safety constraints which is an interesting area for future research (Section 8.4). Furthermore, the deployment of legged robots in the real world may need more efficient mobility. Exploring hybrid locomotion modalities is another interesting aspect that needs more research (Section 8.5).

As legs are becoming the modality of choice for traversing various environments, locomotion needs to be combined with manipulation to perform tasks in the real world. Adding the complexity of object manipulation to the locomotion problem opens up new challenges that require a substantial amount of research (Section 8.6). Finally, the rise of foundation models has sparked interest among roboticists. In particular, the fact that these models have shown some capability for common-sense reasoning, points to them as being potentially useful tools to handle the combinatorial nature of the loco-manipulation problem in the real world (Section 8.7).

8.1. Unsupervised skill discovery

DRL often requires labor-intensive reward engineering of domain experts to obtain desirable behaviors. Instead, unsupervised skill discovery can potentially reduce the burden of repetitive reward engineering by learning a set of reusable skills based on intrinsic motivation. Typically, unsupervised skill discovery aims to learn a skill-conditioned policy that executes diverse and discernable skills, as discussed in Eysenbach et al. (2018).

Researchers have applied the idea of unsupervised skill discovery to learn diverse locomotion skills (Sharma et al., 2020; Cheng et al., 2023a). For example, Schwarke et al. (2023) successfully used this approach to learn loco-manipulation skills for a real quadrupedal robot. Once the skills are obtained, they can be leveraged for downstream tasks via model-predictive control (Sharma et al., 2020) or hierarchical RL (Eysenbach et al., 2018). A similar idea has been also applied to learn skills from unstructured motions from expert policies (Li et al., 2023a) or even from human motion clips (Li et al., 2023c).

8.2. Differentiable simulators

To improve the sample efficiency of learning-based approaches for legged locomotion, there has been an increasing interest in the use of differentiable simulators (Schwarke et al., 2024). The main idea in these approaches is to better exploit the structure available in the simulation environments, for instance through the use of gradients obtained either analytically (Werling et al., 2021; Howell et al., 2022) or using automatic differentiation (Degrave et al., 2019; Hu et al., 2019; Freeman et al., 2021) or even finite difference (Todorov et al., 2012; MJX, 2023). However, the main problem with this approach is poor local minima due to contact. One interesting approach to overcome this issue is the use of randomized smoothing techniques (Suh et al., 2022; Lidec et al., 2022; Pang et al., 2023). As more and more differentiable simulators become available, this under-explored area in the field can attract more researchers.

8.3. Traversing challenging environments

After the success of DRL in generating highly robust behaviors (Hwangbo et al., 2019; Lee et al., 2020; Bogdanovic et al., 2022), recently there has been an increasing interest in learning locomotion skills in challenging environments, for example, performing parkour (Zhuang et al., 2023; Cheng et al., 2023c; Hoeller et al., 2023), traversing stepping stones (Duan et al., 2022; Zhang et al., 2023; Dhédin et al., 2024), moving in confined environments (Xu et al., 2024; Chane-Sane et al., 2024) (see Figure 1 for a few examples). This is an interesting direction that can greatly benefit from the power of deep learning to include perception in the control loop. While it has been argued that DRL-based controllers may struggle with traversing highly constrained environments (Grandia et al., 2023), recent efforts have shown the opposite (Zhang et al., 2023). However, it remains an interesting question to be answered; to what extent DRL approaches can handle safety-critical situations?

8.4. Safety

With continuous advancements in robot locomotion, they will start to be deployed in more diverse locations such as homes, offices, plants, or in the wild. As such, safety

becomes a critical topic as we want to make sure the robots do not damage any human nearby or the environment nor do they damage themeselves. One possible way to tackle safety in robot locomotion is to formulate it as a constrained optimization problem. Indeed, researchers in safe reinforcement learning have developed a variety of safe RL algorithms that aim to train control policies without violating safety constraints (Thananjeyan et al., 2021; Hsu et al., 2023; Achiam et al., 2017; Liu et al., 2022b), where they focus on theoretical properties of the algorithms and are applied to synthetic tasks. More recently, researchers have started to apply these methods to legged robots and demonstrated safe learning in the real world (Yang et al., 2022b). Constrained policy optimization approaches have also been used to achieve safer and more precise behaviors for locomotion in complex environments (Kim et al., 2023d; Lee et al., 2023; Xu et al., 2024). It is anticipated that this direction of research will continue to thrive with the more advanced legged robot capabilities and hardware.

Another way to ensure safety, specifically in deployment, is to use safety filters, which have been extensively studied in the control community (Wabersich et al., 2023). These filters can be designed using Hamilton-Jacobi (HJ) reachability, control barrier functions (CBF), and predictive methods for uncertain systems. Variations of algorithms with and without access to the robot dynamics have been developed. However, their application to locomotion has been limited (Kim et al., 2023b; He et al., 2024). More research on this under-explored area in the future can facilitate certification of this technology for the real-world safety-critical applications.

8.5. Hybrid wheeled-legged locomotion

A wheeled-legged robot is a variation of legged robots designed to enhance efficiency and endurance. Unlike traditional point-foot legged robots, the actuated wheels allow for an additional mode of locomotion, enabling them to drive without stepping in the longitudinal direction.

Achieving both efficiency and stability requires a locomotion controller capable of transitioning between walking and driving locomotion modes. In conventional model-based approaches, this mode switching often relies on operator commands or heuristics (Bjelonic et al., 2019, 2021). However, gait generation for wheeled-legged robots is not straightforward, because there is no such system in the nature. This makes heuristic approaches with periodic patterns such as CPG impractical.

While this research field is still very young, recent studies have explored training locomotion policies in an end-to-end manner using DRL. Lee et al. (2024) applied an approach similar to Miki et al. (2022a) without CPGs to a wheeled-legged system, and demonstrated adaptive gait transitions and locomotion over rough terrains. Similarly, Chamorro et al. (2024) and Cui et al. (2021) demonstrated wheeled-legged bipeds traversing uneven terrains, albeit without gait transitions. This combined modality has a lot to offer and it seems to be an interesting direction for more research.

8.6. Loco-manipulation

When manipulating objects by a legged robot, it becomes essential to take the object affordances, dynamics, and constraints into account in the planning and control phases. Given the affordances of the objects and the environment (Gibson, 1977), the robot needs to simultaneously decide on both locomotion and manipulation aspects. There have been recent efforts on the use of DRL for loco-manipulation tasks in the real world. In general, there are two ways to think about loco-manipulation using quadrupeds. The first one is to use the robot's body and feet to perform some pushing and pressing tasks. The second way is to add an arm to the robot and use the gripper to grasp objects and perform more sophisticated loco-manipulation behaviors.

Within the first category, Ji et al. (2022b), Huang et al. (2023), Jeon et al. (2023a), and Arm et al. (2024) propose a hierarchical framework to use the robot body and feet to generate interesting loco-manipulation behaviors. Kim et al. (2022) use teleoperation of human motion and employ DRL to re-target the motions to realizable quadrupedal loco-manipulation behaviors. Cheng et al. (2023b) use CL to learn different skills in simulation and then build a behavior tree to sequence the skills.

Using only feet and body for manipulation is limited to simple loco-manipulation skills, as the robot is not able to grip/grasp objects/environments. To perform more sophisticated object manipulation, it is now common to mount a manipulator on quadrupeds. To perform loco-manipulation using quadruped with an arm, Fu et al. (2023) use a privileged learning framework to train a policy that simultaneously deals with locomotion and manipulation. Liu et al. (2024) develop a hierarchical framework, using a high-level policy that takes visual information as input and provides end-effector commands for a learned low-level controller. However, these approaches are limited to simple manipulation tasks with a quadruped and cannot reason about the complex loco-manipulation tasks. Recently, Kumar et al. (2023) proposed to use a library of loco-manipulation skill primitives and to compose them and generate a long sequence of loco-manipulation system. However, the manipulation part of the framework is still basic (such as opening a door). Using learning-based algorithms to perform a long sequence of complex loco-manipulation tasks is an interesting future frontier.

More recent work also explores how to learn loco-manipulation behaviors for bipedal robots (e.g., Fu et al., 2024; Dao et al., 2024; Seo et al., 2023). One advantage of using a humanoid form is the vast source of loco-manipulation data from human demonstration, either via motion capture, teleoperation, or video. However, bipedal loco-manipulation is also more challenging to achieve as compared to quadruped due to quadruped loco-manipulation

due to the latter being more stable. Loco-manipulation for humanoid robots is also an active area of research area and more research is to be done in the future.

8.7. Foundation models

Learning locomotion skills and empowering robots with mobility is a key milestone toward building general-purpose robots that are truly useful. However, a notable gap still exists in terms of enabling robots to generalize their behaviors to arbitrary tasks, environments, and interactions with humans. The recent progress in training large-capacity pre-trained models with web-scale data (foundation models) has demonstrated impressive common-sense reasoning, learning, and perception capabilities without specialized training (Achiam et al., 2023; Touvron et al., 2023; Team et al., 2023). Such capabilities are critical for creating general-purpose robots and have thus led to a plethora of works that combine robotics and foundation models in planning, reasoning, navigation, and simulation of robotic systems (Kira, 2022). We refer readers to several existing survey papers summarizing these efforts (Hu et al., 2023; Zhou et al., 2023; Firoozi et al., 2023; Xiao et al., 2023) and focus on discussing the ones applied to robot locomotion here.

A common approach in combining foundation models with legged robots applies it for high-level planning (Lykov et al., 2024; Xu et al., 2023). In these methods, a foundation model is used to interpret the environment and task and to plan a sequence of skills that the robot should perform to achieve a longer horizon task. However, these methods usually do not have fine control of the robots' low-level behavior such as locomotion gaits. This is due to the fact that pre-trained foundation models are usually not trained with robotic data and cannot directly generate low-level robot actions. To leverage foundation models that also work with low-level locomotion control, researchers have designed different interfaces including foot contact patterns (Tang et al., 2023), and reward functions (Yu et al., 2023a; Liang et al., 2024; Ma et al., 2024) that allow foundation models to directly interact with robot's low-level controllers to perform tasks such as hopping, or high-five with a person. Alternatively, one may also fine-tune a large foundation model to directly output the low-level robot action using a large amount of robot data. This has been explored in the manipulation domain and shown promising generalization capabilities to novel tasks (Brohan et al., 2023). Applying this idea to robot locomotion is a promising idea to obtain generalist robot locomotion controllers.

9. Societal impact

Ever-more capable legged robots bring significant potential for both good and bad. This includes the potential to be deployed in many humanitarian applications, tackling dull-and-repetitive tasks in warehouse and agriculture operations, factory inspection, and performing jobs that are dangerous for humans, for example, search and rescue and firefighting. At the same time, these robots have the potential to be used in multiple destructive applications. We believe that our community should play an active role in the discussions related to the regulations at this stage of development toward general-purpose robots (which are mostly legged systems) and remain alerted about the potential risks of this technology.

Here, we aim to only list some immediate issues related to the use of general-purpose robots in the real world. While being preliminary, we believe initiatives from the robotics research community into these topics can have several positive impacts. In particular, we urge the audience of this paper to read the excellent article on the ethics of artificial intelligence and robotics Müller (2020). As well said in the article: *there is a tendency for businesses, the military, and some public administrations to* "*just talk*" *and do some* "*ethics washing*" *in order to preserve a good public image and continue as before. Actual policy is not just an implementation of ethical theory, but subject to societal power structures—and the agents that do have the power will push against anything that restricts them. There is thus a significant risk that regulation will remain toothless in the face of economical and political power.* Within such a power structure, academia and researchers can constitute a powerful entity that raises awareness about the new technologies and their implications, and makes sure that the right policies are implemented.

One dangerous application is to use general-purpose robots as lethal autonomous weapon systems (Righetti et al., 2018). Unfortunately, we have seen such activities toward weaponizing quadrupeds in recent years. Such systems would raise multiple important ethical and societal issues. From an ethical point of view, this question arises: should an autonomous system that *does not suffer* or *has no feelings* itself have the right to decide to kill a person? On the legal side, who would be responsible (and punished) for the potential mistakes of an autonomous weapon system? Also, from the human rights perspective, autonomous weapons would not be aligned with international humanitarian law. Human Rights Watch and Harvard Law School's International Human Rights Clinic (IHRC) notes (Docherty, 2012): *...such revolutionary weapons would not be consistent with international humanitarian law and would increase the risk of death or injury to civilians during armed conflict. A preemptive prohibition on their development and use is needed.* A recent open letter (letter, 2022) signed by multiple companies (Agility Robotics, ANYbotics, Clearpath Robotics, Open Robotics, Unitree, Boston Dynamics) aligns with this view and condemns any use of weaponized mobile robots. Activities of this kind, together with practical measures, can help to safe guard the technology of legged robots in the future.

Another impact of developing general-purpose robots in the near future is the possible human jobs lost due to new automation abilities (Pham et al., 2018). Replacing humans in health-threatening jobs, for example, firefighting, search

and rescue in the wild, dangerous industrial work, and caregiving in pandemics (Shen et al., 2020), can be of significant benefit. However, the problem arises with the potential to replace human workers by robots for many other job categories, and the impact on the lives of displaced workers. An alternative that is regularly advocated is to use robots to take over the physical work while humans use their cognitive capabilities. Human-robot collaboration is another scenario that is advocated over replacing humans. While human history has seen significant shifts due to mechanization before, as with agriculture, for example, past outcomes and adaptations may be a poor predictor of future outcomes on these issues. A recent study spanning different cultures and industries suggests that increased exposure to robots can lead to increased job insecurity (Yam et al., 2023). With rapidly advancing technology, it will be important for policy makers to consider the impact of robotics on the society and economy of the future.

Environmental concerns currently see minimal discussion in the robotics community. As deep learning is increasingly used with models of increasing size, it is important to discuss the carbon footprint of training and using these massive models. This issue is already discussed in the machine learning community, and it is important that the robotics community also thinks about how to mitigate the rapid and unsustainable growth of compute requirements. Another more nuanced environmental issue is that of the life-cycle of the robot itself: What happens at the end of a robot's operational life? Currently legged robots and humanoids are not contributing significantly in this area and the question is more immediate concern for industrial robots and autonomous cars. However, the adoption of quadrupeds and humanoids on a larger scale means that we may eventually see large numbers of these robots that require disposal or recycling when they are no longer functional. It is therefore incumbent on us to think about this issue sooner rather than later.

We have only touched on some of the immediate risks related to general-purpose robotic systems. This list is by no means complete. It does not take other aspects of the problem into account that are related to the software of an embodied AI, including enabling unwanted surveillance, biases in decision making, possible amplification of existing inequities, and more.

10. Conclusions

Learning-based methods are at the heart of so many of the advances seen in legged locomotion control. In this article, we summarized the core issues and methods for quadrupedal locomotion learning, as well as touching on recent related methods for bipedal locomotion. While it is impossible to be fully comprehensive, we hope that this survey provides a valuable framework for understanding recent progress along with ample points of reference. We have further outlined unsolved problems and future directions that, because of their difficulty and broad nature, will provide significant challenges for years to come. However, it is reasonable to expect the rapid pace of innovation to continue, yielding ever-more capable legged locomotion for quadrupeds and bipeds.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported in part by Korea Evaluation Institute of Industrial Technology (KEIT) funded by the Korea Government (MOTIE) under Grant No.20018216, Development of mobile intelligence SW for autonomous navigation of legged robots in dynamic and atypical environments for real application.

ORCID iD

Majid Khadiv https://orcid.org/0000-0001-9889-6543

References

Abate AM (2018) *Mechanical Design for Robot Locomotion*. Graduate Thesis.

Abdolmaleki A, Springenberg JT, Tassa Y, et al. (2018) Maximum a posteriori policy optimisation. arXiv preprint arXiv: 1806.06920.

Abel D (2019) simple_rl: reproducible reinforcement learning in python. In: *RML@ ICLR*. New Orleans, LA.

Aceituno-Cabezas B, Mastalli C, Dai H, et al. (2017) Simultaneous contact, gait, and motion planning for robust multilegged locomotion via mixed-integer convex optimization. *IEEE Robotics and Automation Letters* 3(3): 2531–2538.

Achiam J, Held D, Tamar A, et al. (2017) Constrained policy optimization. In: *International Conference on Machine Learning*. PMLR, 22–31.

Achiam J, Adler S, Agarwal S, et al. (2023) Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Agarwal A, Kumar A, Malik J, et al. (2023) Legged locomotion in challenging terrains using egocentric vision. In: *Conference on Robot Learning*. PMLR, 403–415.

Ahn J, Kim D, Bang S, et al. (2019) Control of a high performance bipedal robot using viscoelastic liquid cooled actuators. In: *2019 IEEE-RAS 19th International Conference on Humanoid Robots (Humanoids)*. IEEE, 146–153.

Akkaya I, Andrychowicz M, Chociej M, et al. (2019) Solving rubik's cube with a robot hand. arXiv preprint arXiv:1910.07113.

Anne T, Wilkinson J and Li Z (2021) Meta-learning for fast adaptive locomotion with uncertainties in environments and robot dynamics. In: *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 4568–4575.

Arm P, Mittal M, Kolvenbach H, et al. (2024) Pedipulate: enabling manipulation skills using a quadruped robot's leg. arXiv preprint arXiv:2402.10837.

Atkeson CG, Benzun PB, Banerjee N, et al. (2018) What happened at the darpa robotics challenge finals. In: *The DARPA*

Robotics Challenge Finals: Humanoid Robots to the Rescue, Springer Tracts in Advanced Robotics 667–684.

Atkeson JCG and Atkeson CG (2009) Nonparametric representation of an approximated poincaré map for learning biped locomotion. *Autonomous Robots* 27(2): 131–144.

Bao L, Humphreys J, Peng T, et al. (2024) Deep reinforcement learning for bipedal locomotion: a brief survey. arXiv preprint arXiv:2401.16889.

Barasuol V, Buchli J, Semini C, et al. (2013) A reactive controller framework for quadrupedal locomotion on challenging terrain. In: *2013 IEEE International Conference on Robotics and Automation*. IEEE, 2554–2561.

Barth-Maron G, Hoffman MW, Budden D, et al. (2018) Distributed distributional deterministic policy gradients. arXiv preprint arXiv:1804.08617.

BDI (2015) *Spot Specifications*. URL. https://support.bostondynamics.com/s/article/Robot-specifications.

Bechtle S, Hammoud B, Rai A, et al. (2021) Leveraging forward model prediction error for learning control. In: *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 4445–4451.

Bellegarda G and Ijspeert A (2022) Cpg-rl: learning central pattern generators for quadruped locomotion. *IEEE Robotics and Automation Letters* 7(4): 12547–12554.

Bellegarda G, Nguyen C and Nguyen Q (2020) Robust quadruped jumping via deep reinforcement learning. arXiv preprint arXiv:2011.07089.

Bellegarda G, Chen Y, Liu Z, et al. (2022) Robust high-speed running for quadruped robots via deep reinforcement learning. In: *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 10364–10370.

Bellman R (1966) Dynamic programming. *Science* 153(3731): 34–37.

Besselmann MG, Puck L, Steffen L, et al. (2021) Vdb-mapping: a high resolution and real-time capable 3d mapping framework for versatile mobile robots. In: *2021 IEEE 17th International Conference on Automation Science and Engineering (CASE)*. IEEE, 448–454.

Bjelonic M, Bellicoso CD, de Viragh Y, et al. (2019) Keep rollin'-whole-body motion control and planning for wheeled quadrupedal robots. *IEEE Robotics and Automation Letters* 4(2): 2116–2123.

Bjelonic M, Grandia R, Harley O, et al. (2021) Whole-body mpc and online gait sequence generation for wheeled-legged robots. In: *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 8388–8395.

Bledt G, Powell MJ, Katz B, et al. (2018) Mit cheetah 3: design and control of a robust, dynamic quadruped robot. In: *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2245–2252.

Blickhan R (1989) The spring-mass model for running and hopping. *Journal of Biomechanics* 22(11-12): 1217–1227.

Bogdanovic M, Khadiv M and Righetti L (2020) Learning variable impedance control for contact sensitive tasks. *IEEE Robotics and Automation Letters* 5(4): 6129–6136.

Bogdanovic M, Khadiv M and Righetti L (2022) Model-free reinforcement learning for robust locomotion using demonstrations from trajectory optimization. *Frontiers in Robotics and AI* 9: 6.

Boyan J and Moore A (1994) Generalization in reinforcement learning: safely approximating the value function. *Advances in Neural Information Processing Systems* 7: 7.

Brakel P, Bohez S, Hasenclever L, et al. (2022) Learning coordinated terrain-adaptive locomotion by imitating a centroidal dynamics planner. In: *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 10335–10342.

Brohan A, Brown N, Carbajal J, et al. (2023) Rt-2: vision-language-action models transfer web knowledge to robotic control. arXiv preprint arXiv:2307.15818.

Buchli J and Ijspeert AJ (2008) Self-organized adaptive legged locomotion in a compliant quadruped robot. *Autonomous Robots* 25: 331–347.

Buchli J, Iida F and Ijspeert AJ (2006) Finding resonance: adaptive frequency oscillators for dynamic legged locomotion. In: *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 3903–3909.

Buchli J, Kalakrishnan M, Mistry M, et al. (2009) Compliant quadruped locomotion over rough terrain. In: *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 814–820.

Buckman J, Hafner D, Tucker G, et al. (2018) Sample-efficient reinforcement learning with stochastic ensemble value expansion. *Advances in Neural Information Processing Systems* 31: 1.

Buehler M, Battaglia R, Cocosco A, et al. (1998) A simple quadruped that walks, climbs, and runs. In: *Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No. 98CH36146)*. IEEE, Vol. 2, 1707–1712.

Buehler M, Cocosco A, Yamazaki K, et al. (1999) Stable open loop walking in quadruped robots with stick legs. In: *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No. 99CH36288C)*. IEEE, Vol. 3, 2348–2353.

Bussola R, Focchi M, Del Prete A, et al. (2024) Efficient reinforcement learning for 3d jumping monopods. *Sensors* 24(15): 4981.

Byun JW, Youm D, Jeon S, et al. (2021) Learning footstep planning for the quadrupedal locomotion with model predictive control. In: *International Conference on Robot Intelligence Technology and Applications*. Springer, 35–43.

Caluwaerts K, Iscen A, Kew JC, et al. (2023) Barkour: benchmarking animal-level agility with quadruped robots. arXiv preprint arXiv:2305.14654.

Carius J, Farshidian F and Hutter M (2020) Mpc-net: a first principles guided policy search. *IEEE Robotics and Automation Letters* 5(2): 2897–2904.

Caron S (2022) Awesome robot description. URL. https://github.com/robot-descriptions/awesome-robot-descriptions?tab=readme-ov-file.

Carpentier J, Saurel G, Buondonno G, et al. (2019) *2019 IEEE/SICE International Symposium on System Integration (SII)*. IEEE, 614–619.

Castillo GA, Weng B, Yang S, et al. (2023) Template model inspired task space learning for robust bipedal locomotion. In:

2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 8582–8589.

Chamorro S, Klemm V, Valls I, et al. (2024) Reinforcement learning for blind stair climbing with legged and wheeled-legged robots. arXiv preprint arXiv:2402.06143.

Chane-Sane E, Leziart P-A, Flayols T, et al. (2024) Cat: constraints as terminations for legged locomotion reinforcement learning. arXiv preprint arXiv:2403.18765.

Chebotar Y, Handa A, Makoviychuk V, et al. (2019) Closing the sim-to-real loop: adapting simulation randomization with real world experience. In: *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 8973–8979.

Chen D, Zhou B, Koltun V, et al. (2020) Learning by cheating. In: *Conference on Robot Learning*. PMLR, 66–75.

Chen X, Wang C, Zhou Z, et al. (2021) Randomized ensembled double q-learning: learning fast without a model. arXiv preprint arXiv:2101.05982.

Chen S, Zhang B, Mueller MW, et al. (2023) Learning torque control for quadrupedal locomotion. In: *2023 IEEE-RAS 22nd International Conference on Humanoid Robots (Humanoids)*. IEEE, 1–8.

Cheng J, Vlastelica M, Kolev P, et al. (2023a) Learning diverse skills for local navigation under multi-constraint optimality. arXiv preprint arXiv:2310.02440.

Cheng X, Kumar A and Pathak D. (2023b) Legs as manipulator: pushing quadrupedal agility beyond locomotion. arXiv preprint arXiv:2303.11330.

Cheng X, Shi K, Agarwal A, et al. (2023c) Extreme parkour with legged robots. arXiv preprint arXiv:2309.14341.

Cheng X, Ji Y, Chen J, et al. (2024) Expressive whole-body control for humanoid robots. arXiv preprint arXiv:2402.16796.

Chignoli M, Kim D, Stanger-Jones E, et al. (2021) The mit humanoid robot: Design, motion planning, and control for acrobatic behaviors. In: *2020 IEEE-RAS 20th International Conference on Humanoid Robots (Humanoids)*. IEEE, 1–8.

Choi S and Kim J (2019) Trajectory-based probabilistic policy gradient for learning locomotion behaviors. In: *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 1–7.

Choi S, Ji G, Park J, et al. (2023) Learning quadrupedal locomotion on deformable terrain. *Science Robotics* 8(74): eade2256.

Chua K, Calandra R, McAllister R, et al. (2018) Deep reinforcement learning in a handful of trials using probabilistic dynamics models. *Advances in Neural Information Processing Systems* 31: 63.

Collins J, Chand S, Vanderkop A, et al. (2021) A review of physics simulators for robotic applications. *IEEE Access* 9: 51416–51431.

Coumans E (2020) *Motion Imitation*. URL. https://github.com/erwincoumans/motion_imitation.

Coumans E and Bai Y (2016) Pybullet, a python module for physics simulation for games. *Robotics and machine learning*.

Cui L, Wang S, Zhang J, et al. (2021) Learning-based balance control of wheel-legged robots. *IEEE Robotics and Automation Letters* 6(4): 7667–7674.

Cybenko G (1989) Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals, and Systems* 2(4): 303–314.

CyberDog (2021) *Xiaomi Launches Cyberdog – an Open Source Quadruped Robot Companion*. URL. https://www.mi.com/global/discover/article?id=2069.

Daneshmand E, Khadiv M, Grimminger F, et al. (2021) Variable horizon mpc with swing foot dynamics for bipedal walking control. *IEEE Robotics and Automation Letters* 6(2): 2349–2356.

Dantec E, Budhiraja R, Roig A, et al. (2021) Whole body model predictive control with a memory of motion: experiments on a torque-controlled talos. In: *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 8202–8208.

Dao J, Duan H, Green K, et al. (2020) Learning to walk without dynamics randomization. In: *2nd Workshop on Closing the Reality Gap in Sim2Real Transfer for Robotics*, p. 16.

Dao J, Duan H and Fern A (2024) Sim-to-real learning for humanoid box loco-manipulation. In: *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 16930–16936.

DARPA Robotics Challenge D (2015) The 2015 darpa robotics challenge finals. URL. https://www.youtube.com/watch?v=8P9geWwi9e0.

Darvish K, Penco L, Ramos J, et al. (2023) Teleoperation of humanoid robots: a survey. *IEEE Transactions on Robotics* 39(3): 1706–1727.

Deeprobotics (2021) Quadruped series from deep robotics. URL. https://www.deeprobotics.cn/en/index/product1.html.

Degrave J, Hermans M, Dambre J, et al. (2019) A differentiable physics engine for deep learning in robotics. *Frontiers in Neurorobotics* 13: 6.

Deits R and Tedrake R (2014) Footstep planning on uneven terrain with mixed-integer convex optimization. In: *2014 IEEE-RAS International Conference on Humanoid Robots*. IEEE, 279–286.

Dhédin V, Chinnakkonda Ravi AK, Jordana A, et al. Diffusion-based learning of contact plans for agile locomotion. In: *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, arXiv–2403, 2024. URL. https://arxiv.org/abs/2403.03639.

Di Carlo J, Wensing PM, Katz B, et al. (2018) Dynamic locomotion in the mit cheetah 3 through convex model-predictive control. In: *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 1–9.

Disney (2023) A new approach to Disney's robotic character pipeline. URL. https://www.youtube.com/watch?v=-cfIm06tcfA.

Docherty BL (2012) *Losing Humanity: The Case against Killer Robots*. Human Rights Watch.

Doi T, Hodoshima R, Hodoshima Y, et al. (2006) Development of quadruped walking robot TITAN XI for steep slopes - slope map generation and map information application -. *Journal of Robotics and mechatronics* 18(3): 318–324.

Dorato P (1987) A historical review of robust control. *IEEE Control Systems Magazine* 7(2): 44–47.

Drnach L and Zhao Y (2021) Robust trajectory optimization over uncertain terrain with stochastic complementarity. *IEEE Robotics and Automation Letters* 6(2): 1168–1175.

Duan H, Dao J, Green K, et al. (2021) Learning task space actions for bipedal locomotion. In: *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 1276–1282.

Duan H, Malik A, Gadde MS, et al. (2022) Learning dynamic bipedal walking across stepping stones. In: *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 6746–6752.

Duan H, Pandit B, Gadde MS, et al. (2023) Learning vision-based bipedal locomotion for challenging terrain. arXiv preprint arXiv:2309.14594.

Dynamics L (2024) Limx dynamics' biped robot p1 conquers the wild based on reinforcement learning. URL. https://www.youtube.com/watch?v=UpNid_rWDnI.

Englsberger J, Werner A, Ott C, et al. (2014) Overview of the torque-controlled humanoid robot toro. In: *2014 IEEE-RAS International Conference on Humanoid Robots*. IEEE, 916–923.

Escontrela A, Peng XB, Yu W, et al. (2022) Adversarial motion priors make good substitutes for complex reward functions. In: *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 25–32.

Espeholt L, Soyer H, Munos R, et al. (2018) Impala: scalable distributed deep-rl with importance weighted actor-learner architectures. In: *International Conference on Machine Learning*. PMLR, 1407–1416.

Eysenbach B, Gupta A, Ibarz J, et al. (2018) Diversity is all you need: learning skills without a reward function. arXiv preprint arXiv:1802.06070.

Fankhauser P, Hutter M, Gehring C, et al. (2013) Reinforcement learning of single legged locomotion. In: *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 188–193.

Farshidian F, Jelavic E, Satapathy A, et al. (2017) Real-time motion planning of legged robots: a model predictive control approach. In: *2017 IEEE-Ras 17th International Conference on Humanoid Robotics (Humanoids)*.

Featherstone R (1987) *Robot Dynamics Algorithms*. The Springer International Series in Engineering and Computer Science, 1–211.

Feinberg V, Wan A, Stoica I, et al. (2018) Model-based value estimation for efficient model-free reinforcement learning. arXiv preprint arXiv:1803.00101.

Feng S, Whitman E, Xinjilefu X, et al. (2015) Optimization-based full body control for the DARPA robotics challenge. *Journal of Field Robotics* 32(2): 293–312.

Firoozi R, Tucker J, Tian S, et al. (2023) Foundation models in robotics: applications, challenges, and the future. arXiv preprint arXiv:2312.07843.

Focchi M, del Prete A, Havoutis I, et al. (2017) High-slope terrain locomotion for torque-controlled quadruped robots. *Autonomous Robots* 41: 259–272.

Freeman CD, Frey E, Raichuk A, et al. (2021) Brax–a differentiable physics engine for large scale rigid body simulation. arXiv preprint arXiv:2106.13281.

Fu Z, Cheng X and Pathak D (2023) Deep whole-body control: learning a unified policy for manipulation and locomotion. In: *Conference on Robot Learning*. PMLR, 138–149.

Fu Z, Zhao Q, Wu Q, et al. (2024) Humanoid shadowing and imitation from humans. *arXiv preprint arXiv:2406.10454*.

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

Fujimoto S, Hoof H and Meger D (2018) Addressing function approximation error in actor-critic methods. In: *International Conference on Machine Learning*. PMLR, 1587–1596.

Gangapurwala S, Mitchell A and Havoutis I (2020) Guided constrained policy optimization for dynamic quadrupedal robot locomotion. *IEEE Robotics and Automation Letters* 5(2): 3642–3649.

Gangapurwala S, Geisert M, Orsolino R, et al. (2021) Real-time trajectory adaptation for quadrupedal locomotion using deep reinforcement learning. In: *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 5973–5979.

Gangapurwala S, Geisert M, Orsolino R, et al. (2022) Terrain-aware legged locomotion using reinforcement learning and optimal control. *IEEE Transactions on Robotics* 38(5): 2908–2927.

Gazar A, Khadiv M, Del Prete A, et al. (2023) Multi-contact stochastic predictive control for legged robots with contact locations uncertainty. arXiv preprint arXiv:2309.04469.

Geilinger M, Hahn D, Zehnder J, et al. (2020) Add: analytically differentiable dynamics for multi-body systems with frictional contact. *ACM Transactions on Graphics* 39(6): 1–15.

Geyer H, Blickhan R and Seyfarth A (2002) Natural dynamics of spring-like running: emergence of selfstability. In: *5th International Conference on Climbing and Walking Robots*. England: Professional Engineering Publishing Ltd Suffolk, Vol. 92.

Gibson JJ (1977) The theory of affordances. *Hilldale, USA* 1(2): 67–82.

Golemo F, Taiga AA, Courville A, et al. (2018) Sim-to-real transfer with neural-augmented robot simulation. In: *Conference on Robot Learning*. PMLR, 817–828.

Goodfellow I, Bengio Y and Courville A (2016) *Deep Learning*. MIT press.

Goswami A and Vadakkepat P (2018) *Humanoid Robotics: A Reference*. Incorporated: Springer Publishing Company.

Grandia R, Jenelten F, Yang S, et al. (2023) Perceptive locomotion through nonlinear model predictive control. *IEEE Transactions on Robotics*.

Green K, Godse Y, Dao J, et al. (2021) Learning spring mass locomotion: guiding policies with a reduced-order model. *IEEE Robotics and Automation Letters* 6(2): 3926–3932.

Grimminger F, Meduri A, Khadiv M, et al. (2020) An open torque-controlled modular robot architecture for legged locomotion research. *IEEE Robotics and Automation Letters* 5(2): 3650–3657.

Ha D and Schmidhuber J. (2018) World models. arXiv preprint arXiv:1803.10122.

Ha S, Kim J and Yamane K (2018) Automated deep reinforcement learning environment for hardware of a modular legged robot. In: *2018 15th International Conference on Ubiquitous Robots (UR)*. IEEE, 348–354.

Ha S, Xu P, Tan Z, et al. (2020) Learning to walk in the real world with minimal human effort. arXiv preprint arXiv: 2002.08550.

Haarnoja T, Hartikainen K, Abbeel P, et al. (2018a) Latent space policies for hierarchical reinforcement learning. In: *International Conference on Machine Learning*. PMLR, 1851–1860.

Haarnoja T, Zhou A, Hartikainen K, et al. (2018b) Soft actor-critic algorithms and applications. arXiv preprint arXiv: 1812.05905.

Haarnoja T, Ha S, Zhou A, et al. (2019) Learning to walk via deep reinforcement learning. *Robotics: Science and Systems*.

Haarnoja T, Moran B, Lever G, et al. (2024) Learning agile soccer skills for a bipedal robot with deep reinforcement learning. *Science Robotics* 9(89): eadi8022.

Hafner D, Lillicrap T, Ba J, et al. (2019a) Dream to control: learning behaviors by latent imagination. *arXiv preprint arXiv:1912.01603*.

Hafner D, Lillicrap T, Fischer I, et al. (2019b) Learning latent dynamics for planning from pixels. In: *International Conference on Machine Learning*. PMLR, 2555–2565.

Hammoud B, Khadiv M and Righetti L (2021) Impedance optimization for uncertain contact interactions through risk sensitive optimal control. *IEEE Robotics and Automation Letters* 6(3): 4766–4773.

Han L, Zhu Q, Sheng J, et al. (2023) Lifelike agility and play on quadrupedal robots using reinforcement learning and generative pre-trained models. arXiv preprint arXiv:2308.15143.

Hansen N, Müller SD and Koumoutsakos P (2003) Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (cma-es). *Evolutionary Computation* 11(1): 1–18.

He T, Zhang C, Xiao W, et al. (2024) Agile but safe: learning collision-free high-speed legged locomotion. arXiv preprint arXiv:2401.17583.

Heess N, Tb D, Sriram S, et al. (2017) Emergence of locomotion behaviours in rich environments. arXiv preprint arXiv: 1707.02286.

Heijmink E, Radulescu A, Ponton B, et al. (2017) Learning optimal gait parameters and impedance profiles for legged locomotion. In: *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)*. IEEE, 339–346.

Hessel M, Modayil J, Van Hasselt H, et al. (2018) Combining improvements in deep reinforcement learning. In: *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 32.

Hester T, Quinlan M and Stone P (2010) Generalized model learning for reinforcement learning on a humanoid robot. In: *2010 IEEE International Conference on Robotics and Automation*. IEEE, 2369–2374.

Hiraoka T, Imagawa T, Hashimoto T, et al. (2021) Dropout q-functions for doubly efficient reinforcement learning. arXiv preprint arXiv:2110.02034.

Ho J and Ermon S (2016) Generative adversarial imitation learning. *Advances in Neural Information Processing Systems* 29: 1.

Hoeller D, Wellhausen L, Farshidian F, et al. (2021) Learning a state representation and navigation in cluttered and dynamic environments. *IEEE Robotics and Automation Letters* 6(3): 5081–5088.

Hoeller D, Rudin N, Sako D, et al. (2023) Anymal parkour: learning agile navigation for quadrupedal robots. arXiv preprint arXiv:2306.14874.

Hornby GS, Takamura S, Yokono J, et al. (2000) Evolving robust gaits with aibo. In: *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*. IEEE, Vol. 3, 3040–3045.

Hornik K (1991) Approximation capabilities of multilayer feedforward networks. *Neural Networks* 4(2): 251–257.

Howard RA (1960) *Dynamic Programming and Markov Processes*. Technology Press of Massachusetts Institute of Technology.

Howell TA, Cleac'h SL, Kolter JZ, et al. (2022) Dojo: a differentiable simulator for robotics. arXiv preprint arXiv: 2203.00806.

Hsu K-C, Hu H and Fisac JF (2023) The safety filter: a unified view of safety-critical control in autonomous systems. *Annual Review of Control, Robotics, and Autonomous Systems* 7: 1.

Hu Y, Anderson L, Li T-M, et al. (2019) Difftaichi: differentiable programming for physical simulation. arXiv preprint arXiv: 1910.00935.

Hu Y, Xie Q, Jain V, et al. Toward general-purpose robots via foundation models: a survey and meta-analysis. arXiv preprint arXiv:2312.08782.

Huang C, Wang G, Zhou Z, et al. (2022a) Reward-adaptive reinforcement learning: dynamic policy gradient optimization for bipedal locomotion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Huang S, Dossa RFJ, Ye C, et al. (2022b) Cleanrl: high-quality single-file implementations of deep reinforcement learning algorithms. *Journal of Machine Learning Research* 23(274): 1–18, URL. https://jmlr.org/papers/v23/21-1342.html.

Huang X, Li Z, Xiang Y, et al. (2023) Creating a dynamic quadrupedal robotic goalkeeper with reinforcement learning. In: *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2715–2722.

Hubicki C, Grimes J, Jones M, et al. (2016) ATRIAS: design and validation of a tether-free 3D-capable spring-mass bipedal robot. *The International Journal of Robotics Research* 35(12): 1497–1521.

Hutter M, Gehring C, Bloesch M, et al. (2012) Starleth: a compliant quadrupedal robot for fast, efficient, and versatile locomotion. In: *Adaptive Mobile Robotics*. World Scientific, 483–490.

Hutter M, Sommer H, Gehring C, et al. (2014) Quadrupedal locomotion using hierarchical operational space control. *The International Journal of Robotics Research* 33(8): 1047–1062.

Hutter M, Gehring C, Jud D, et al. (2016) Anymal-a highly mobile and dynamic quadrupedal robot. In: *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 38–44.

Hwangbo J, Lee J and Hutter M (2018) Per-contact iteration method for solving contact dynamics. *IEEE Robotics and Automation Letters* 3(2): 895–902.

Hwangbo J, Lee J, Dosovitskiy A, et al. (2019) Learning agile and dynamic motor skills for legged robots. *Science Robotics* 4(26): eaau5872.

Ibarz J, Tan J, Finn C, et al. (2021) How to train your robot with deep reinforcement learning: lessons we have learned. *The International Journal of Robotics Research* 40(4-5): 698–721.

Ijspeert AJ (2001) A connectionist central pattern generator for the aquatic and terrestrial gaits of a simulated salamander. *Biological Cybernetics* 84(5): 331–348.

Ijspeert AJ (2008) Central pattern generators for locomotion control in animals and robots: a review. *Neural Networks* 21(4): 642–653.

Iscen A, Caluwaerts K, Tan J, et al. (2018) Policies modulating trajectory generators. In: *Conference on Robot Learning*. PMLR, 916–926.

Janner M, Fu J, Zhang M, et al. (2019) When to trust your model: model-based policy optimization. *Advances in Neural Information Processing Systems* 32: 1.

Jenelten F, He J, Farshidian F, et al. (2024) Dtc: deep tracking control. *Science Robotics* 86: eadh5401.

Jeon S, Jung M, Choi S, et al. (2023a) Learning whole-body manipulation for quadrupedal robot. arXiv preprint arXiv: 2308.16820.

Jeon SH, Heim S, Khazoom C, et al. (2023b) Benchmarking potential based rewards for learning humanoid locomotion. In: *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 9204–9210.

Jeong R, Kay J, Romano F, et al. (2019) Modelling generalized forces with reinforcement learning for sim-to-real transfer. arXiv preprint arXiv:1910.09471.

Ji G, Mun J, Kim H, et al. (2022a) Concurrent training of a control policy and a state estimator for dynamic and robust legged locomotion. *IEEE Robotics and Automation Letters* 7(2): 4630–4637.

Ji Y, Li Z, Sun Y, et al. (2022b) Hierarchical reinforcement learning for precise soccer shooting skills using a quadrupedal robot. In: *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 1479–1486.

Ji Y, Margolis GB and Agrawal P (2023) Dribblebot: dynamic legged manipulation in the wild. In: *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 5155–5162.

Jiang Y, Zhang T, Ho D, et al. (2021) Simgan: hybrid simulator identification for domain adaptation via adversarial reinforcement learning. In: *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2884–2890.

Johannink T, Bahl S, Nair A, et al. (2019) Residual reinforcement learning for robot control. In: *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 6023–6029.

Johnson M, Shrewsbury B, Bertrand S, et al. (2017) Team ihmc's lessons learned from the darpa robotics challenge: finding data in the rubble. *Journal of Field Robotics* 34(2): 241–261.

Junyao G, Xingguang D, Qiang H, et al. (2013) The research of hydraulic quadruped bionic robot design. In: *2013 ICME International Conference on Complex Medical Engineering*. IEEE, 620–625.

Kajita S, Kanehiro F, Kaneko K, et al. (2001) The 3d linear inverted pendulum mode: a simple modeling for a biped walking pattern generation. In: *Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the the Next Millennium (Cat. No. 01CH37180)*. IEEE, Vol. 1, 239–246.

Kajita S, Kanehiro F, Kaneko K, et al. (2003) Biped walking pattern generation by using preview control of zero-moment point. In: *2003 IEEE international conference on robotics and automation (Cat. No. 03CH37422)*, IEEE, Vol. 2, 1620–1626.

Kalakrishnan M, Buchli J, Pastor P, et al. (2011) Learning, planning, and control for quadruped locomotion over challenging terrain. *The International Journal of Robotics Research* 30(2): 236–258.

Kang D, Zimmermann S and Coros S (2021) Animal gaits on quadrupedal robots using motion matching and model-based control. In: *2021 IEEERSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE.

Kang D, Cheng J, Zamora M, et al. (2023) Rl+ model-based control: using on-demand optimal control to learn versatile legged locomotion. *IEEE Robotics and Automation Letters*.

Katz B, Di Carlo J and Kim S (2019) Mini cheetah: a platform for pushing the limits of dynamic quadruped control. In: *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 6295–6301.

Kau N, Schultz A, Ferrante N, et al. (2019) Stanford doggo: an open-source, quasi-direct-drive quadruped. In: *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 6309–6315.

Khadiv M, Moosavian SAA, Yousefi-Koma A, et al. (2019) Rigid vs compliant contact: an experimental study on biped walking. *Multibody System Dynamics* 45: 379–401.

Khadiv M, Meduri A, Zhu H, et al. (2023) Learning locomotion skills from mpc in sensor space. In: *Learning for Dynamics and Control Conference*. PMLR.

Kim GS and Kim S (2020) Extracting legged locomotion heuristics with regularized predictive control. In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 406–412.

Kim T, Jang M and Kim J (2021) A survey on simulation environments for reinforcement learning. In: *2021 18th International Conference on Ubiquitous Robots (UR)*. IEEE, 63–67.

Kim S, Sorokin M, Lee J, et al. (2022) Humanconquad: human motion control of quadrupedal robots using deep reinforcement learning. In: *SIGGRAPH Asia 2022 Emerging Technologies*, 1–2.

Kim D, Berseth G, Schwartz M, et al. (2023a) Torque-based deep reinforcement learning for task-and-robot agnostic learning on bipedal robots using sim-to-real transfer. *IEEE Robotics and Automation Letters*.

Kim J, Lee J and Ames AD (2023b) Safety-critical coordination for cooperative legged locomotion via control barrier functions. In: *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2368–2375.

Kim J, Li T and Ha S (2023c) Armp: autoregressive motion planning for quadruped locomotion and navigation in complex indoor environments. In: *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2731–2737.

Kim Y, Oh H, Lee J, et al. (2023d) Not only rewards but also constraints: applications on legged robot locomotion. arXiv preprint arXiv:2308.12517.

Kira Z (2022) Awesome-llm-robotics. URL. https://github.com/GT-RIPL/Awesome-LLM-Robotics.

Klipfel A, Sontakke N, Liu R, et al. (2023) Learning a single policy for diverse behaviors on a quadrupedal robot using scalable motion imitation. In: *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2768–2775.

Koenemann J, Del Prete A, Tassa Y, et al. (2015) Whole-body model-predictive control applied to the hrp-2 humanoid. In: *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 3346–3351.

Kohl N and Stone P (2004) Policy gradient reinforcement learning for fast quadrupedal locomotion. In: *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA'04. 2004*. IEEE, Vol. 3, 2619–2624.

Kolter J, Abbeel P and Ng A (2007) Hierarchical apprenticeship learning with application to quadruped locomotion. *Advances in Neural Information Processing Systems* 20: 1.

Kovalev V, Shkromada A, Ouerdane H, et al. ()Combining model-predictive control and predictive reinforcement learning for stable quadrupedal robot locomotion. arXiv preprint arXiv: 2307.07752.

Krishna L, Castillo GA, Mishra UA, et al. (2022) Linear policies are sufficient to realize robust bipedal walking on challenging terrains. *IEEE Robotics and Automation Letters* 7(2): 2047–2054.

Kuindersma S, Deits R, Fallon M, et al. (2016) Optimization-based locomotion planning, estimation, and control design for the atlas humanoid robot. *Autonomous Robots* 40: 429–455.

Kumar A, Fu Z, Pathak D, et al. (2021) Rma: rapid motor adaptation for legged robots. arXiv preprint arXiv:2107.04034.

Kumar KN, Essa I and Ha S (2023) Cascaded compositional residual learning for complex interactive behaviors. *IEEE Robotics and Automation Letters*.

Kwon T, Lee Y and Van De Panne M (2020) Fast and flexible multilegged locomotion using learned centroidal dynamics. *ACM Transactions on Graphics* 39(4): 46.

Le Lidec Q, Kalevatykh I, Laptev I, et al. (2021) Differentiable simulation for physical system identification. *IEEE Robotics and Automation Letters* 6(2): 3413–3420.

LeCun Y, Bengio Y and Hinton G (2015) Deep learning. *Nature* 521(7553): 436–444.

Lee G, Hou B, Mandalika A, et al. (2018a) Bayesian policy optimization for model uncertainty. arXiv preprint arXiv:1810.01014.

Lee J, Grey MX, Ha S, et al. (2018b) DART: dynamic animation and robotics toolkit. *Journal of Open Source Software* 3(22): 500.

Lee J, Hwangbo J, Wellhausen L, et al. (2020) Learning quadrupedal locomotion over challenging terrain. *Science Robotics* 5(47): eabc5986.

Lee K-H, Nachum O, Zhang T, et al. (2022) Pi-ars: Accelerating Evolution-Learned Visual-Locomotion with Predictive Information Representations. In: *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 1447–1454.

Lee J, Schroth L, Klemm V, et al. (2023) Evaluation of constrained reinforcement learning algorithms for legged locomotion. *arXiv preprint arXiv:2309.15430*.

Lee J, Bjelonic M, Reske A, et al. (2024) Learning robust autonomous navigation and locomotion for wheeled-legged robots. *Science Robotics* 9(89): eadi9641. DOI: 10.1126/scirobotics.adi9641.

Lembono TS, Mastalli C, Fernbach P, et al. (2020) Learning how to walk: warm-starting optimal control solver with memory of motion. In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 1357–1363.

Lengagne S, Vaillant J, Yoshida E, et al. (2013) Generation of whole-body optimal dynamic multi-contact motions. *The International Journal of Robotics Research* 32(9-10): 1104–1119.

letter (2022) General purpose robots should not be weaponized: an open letter to the robotics industry and our communities. URL. https://shorturl.at/f9LhG.

Li T, Lambert N, Calandra R, et al. (2020) Learning generalizable locomotion skills with hierarchical reinforcement learning. In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 413–419.

Li H, Frei RJ and Wensing PM (2021a) Model hierarchy predictive control of robotic systems. *IEEE Robotics and Automation Letters* 6(2): 3373–3380.

Li T, Calandra R, Pathak D, et al. (2021b) Planning in learned latent action spaces for generalizable legged locomotion. *IEEE Robotics and Automation Letters* 6(2): 2682–2689.

Li C, Blaes S, Kolev P, et al. (2023a) Versatile skill control via self-supervised adversarial imitation of unlabeled mixed motions. In: *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2944–2950.

Li C, Vlastelica M, Blaes S, et al. (2023b) Learning agile skills via adversarial imitation of rough partial demonstrations. In: *Conference on Robot Learning*. PMLR, 342–352.

Li T, Jung H, Gombolay M, et al. (2023c) Crossloco: human motion driven control of legged robots via guided unsupervised reinforcement learning. arXiv preprint arXiv:2309.17046.

Li T, Won J, Cho J, et al. (2023d) Fastmimic: model-based motion imitation for agile, diverse and generalizable quadrupedal locomotion. *Robotics* 12(3): 90.

Li C, Stanger-Jones E, Heim S, et al. (2024a) Fld: fourier latent dynamics for structured motion representation and learning. arXiv preprint arXiv:2402.13820.

Li Z, Peng XB, Abbeel P, et al. (2024b) Reinforcement learning for versatile, dynamic, and robust bipedal locomotion control. arXiv preprint arXiv:2401.16889.

Liang J, Xia F, Yu W, et al. (2024) Learning to learn faster from human feedback with language model predictive control. arXiv preprint arXiv:2402.11450.

Lidec QL, Montaut L, Schmid C, et al. (2022) Augmenting differentiable physics with randomized smoothing. arXiv preprint arXiv:2206.11884.

Lidec QL, Jallet W, Montaut L, et al. (2023) Contact models in robotics: a comparative analysis. arXiv preprint arXiv: 2304.06372.

Lillicrap TP, Hunt JJ, Pritzel A, et al. (2015) Continuous control with deep reinforcement learning. arXiv preprint arXiv: 1509.02971.

Liu Y, Shen J, Zhang J, et al. (2022a) Design and control of a miniature bipedal robot with proprioceptive actuation for dynamic behaviors. In: *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 8547–8553.

Liu Z, Cen Z, Isenbaev V, et al. (2022b) Constrained variational policy optimization for safe reinforcement learning. In: *International Conference on Machine Learning*. PMLR, 13644–13668.

Liu M, Chen Z, Cheng X, et al. (2024) Visual whole-body control for legged loco-manipulation. arXiv preprint arXiv:2403.16967.

Luck KS, Campbell J, Jansen MA, et al. (2017) From the lab to the desert: fast prototyping and learning of robot locomotion. arXiv preprint arXiv:1706.01977.

Lykov A, Litvinov M, Konenkov M, et al. (2024) Cognitivedog: large multimodal model based system to translate vision and language into action of quadruped robot. arXiv preprint arXiv:2401.09388.

Ma Y, Farshidian F, Miki T, et al. (2022) Combining learning-based locomotion policy with model-based manipulation for legged mobile manipulators. *IEEE Robotics and Automation Letters* 7(2): 2377–2384.

Ma YJ, Liang W, Wang H, et al. (2024) Dreureka: language model guided sim-to-real transfer. In *Proceedings of Robotics: Science and Systems*.

Makoviychuk V, Wawrzyniak L, Guo Y, et al. (2021) Isaac gym: high performance gpu-based physics simulation for robot learning. arXiv preprint arXiv:2108.10470.

Mania H, Guy A and Recht B (2018) Simple random search of static linear policies is competitive for reinforcement learning. *Advances in Neural Information Processing Systems* 31: 1.

Marco A, Baumann D, Khadiv M, et al. (2021) Robot learning with crash constraints. *IEEE Robotics and Automation Letters* 6(2): 1439–1446.

Margolis GB and Agrawal P (2023) Walk these ways: tuning robot control for generalization with multiplicity of behavior. In: *Conference on Robot Learning*. PMLR, 22–31.

Margolis GB, Chen T, Paigwar K, et al. (2021) Learning to jump from pixels. arXiv preprint arXiv:2110.15344.

Margolis GB, Fu X, Ji Y, et al. (2023) Learning to see physical properties with active sensing motor policies. arXiv preprint arXiv:2311.01405.

Margolis GB, Yang G, Paigwar K, et al. (2024) Rapid locomotion via reinforcement learning. *The International Journal of Robotics Research* 43(4): 572–587.

Marhefka DW and Orin DE (1996) Simulation of contact using a nonlinear damping model. In: *Proceedings of IEEE International Conference on Robotics and Automation*. IEEE, Vol. 2, 1662–1668.

Mastalli C, Merkt W, Xin G, et al. (2023) Agile maneuvers in legged robots: a predictive control approach. *IEEE Transactions on Robotics*.

Masterjohn J, Guoy D, Shepherd J, et al. (2022) Velocity level approximation of pressure field contact patches. *IEEE Robotics and Automation Letters* 7(4): 11593–11600.

Meduri A, Shah P, Viereck J, et al. (2023) A nonlinear model predictive control framework for whole body motion planning. *IEEE Transactions on Robotics*.

Miki T, Lee J, Hwangbo J, et al. (2022a) Learning robust perceptive locomotion for quadrupedal robots in the wild. *Science Robotics* 7(62): eabk2822.

Miki T, Wellhausen L, Grandia R, et al. (2022b) Elevation mapping for locomotion and navigation using gpu. In: *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2273–2280.

Miki T, Lee J, Wellhausen L, et al. (2024) Learning to walk in confined spaces using 3d representation. arXiv preprint arXiv:2403.00187.

Mistry M, Buchli J and Schaal S (2010) Inverse dynamics control of floating base systems using orthogonal decomposition. In: *2010 IEEE International Conference on Robotics and Automation*. IEEE, 3406–3412.

MJX (2023) Mujoco3. URL. https://mujoco.readthedocs.io/en/stable/mjx.html.

Mnih V, Kavukcuoglu K, Silver D, et al. (2015) Human-level control through deep reinforcement learning. *Nature* 518(7540): 529–533.

Mnih V, Badia AP, Mirza M, et al. (2016) Asynchronous methods for deep reinforcement learning. In: *International Conference on Machine Learning*. PMLR, 1928–1937.

Mordatch I, Todorov E and Popović Z (2012) Discovery of complex behaviors through contact-invariant optimization. *ACM Transactions on Graphics* 31(4): 1–8.

Morimoto J, Nakanishi J, Endo G, et al. (2005) Poincare-map-based reinforcement learning for biped walking. In: *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*. IEEE, 2381–2386.

Müller VC (2020) *Ethics of Artificial Intelligence and Robotics*. Stanford Encyclopedia of Philosophy.

Muratore F, Eilers C, Gienger M, et al. (2021) Data-efficient domain randomization with bayesian optimization. *IEEE Robotics and Automation Letters* 6(2): 911–918.

Murphy RR (2015) Meta-analysis of autonomy at the DARPA robotics challenge trials. *Journal of Field Robotics* 32(2): 189–191.

Murphy MP, Saunders A, Moreira C, et al. (2011) The littledog robot. *The International Journal of Robotics Research* 30(2): 145–149.

Nagabandi A, Kahn G, Fearing RS, et al. (2018) Neural network dynamics for model-based deep reinforcement learning with model-free fine-tuning. In: *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 7559–7566.

Nahrendra IMA, Yu B and Myung H (2023) Dreamwaq: learning robust quadrupedal locomotion with implicit terrain imagination via deep reinforcement learning. In: *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 5078–5084.

Negrut CKD and Negrut D (2021) The role of physics-based simulators in robotics. *Annual Review of Control, Robotics, and Autonomous Systems* 4: 35–58.

Nelson G, Saunders A and Playter R (2018) *The PETMAN and Atlas Robots at Boston Dynamics*. Springer Netherlands, Dordrecht, 1–18. DOI: 10.1007/978-94-007-7194-9_15-1.

Neuhaus PD, Pratt JE and Johnson MJ (2011) Comprehensive summary of the institute for human and machine cognition's experience with LittleDog. *The International Journal of Robotics Research* 30(2): 216–235.

Neunert M, de Crousaz C, Furrer F, et al. (2016) Fast nonlinear model predictive control for unified trajectory optimization and tracking. In: *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 1398–1404.

Neunert M, Stauble M, Giftthaler M, et al. (2018) Whole-body nonlinear model predictive control through contacts for quadrupeds. *IEEE Robotics and Automation Letters* 3(3): 1458–1465.

Oleynikova H, Taylor Z, Fehr M, et al. (2017) Voxblox: incremental 3d euclidean signed distance fields for on-board mav planning. In: *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 1366–1373.

Omar S, Amatucci L, Barasuol V, et al. (2023) Safesteps: learning safer footstep planning policies for legged robots via model-based priors. In: *2023 IEEE-RAS 22nd International Conference on Humanoid Robots (Humanoids)*. IEEE, 1–8.

Pandala A, Fawcett RT, Rosolia U, et al. (2022) Robust predictive control for quadrupedal locomotion: learning to close the gap between reduced- and full-order models. *IEEE Robotics and Automation Letters* 7(3): 6622–6629.

Pang T, Suh HT, Yang L, et al. (2023) Global planning for contact-rich manipulation via local smoothing of quasi-dynamic contact models. *IEEE Transactions on Robotics*.

Papadopoulos D and Buehler M (2000) Stable running in a quadruped robot with compliant legs. In: *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*. IEEE, Vol. 1, 444–449.

Park H-W, Wensing PM, Kim S, et al. (2015) Online planning for autonomous running jumps over obstacles in high-speed quadrupeds. In: *Proceedings of Robotics: Science and Systems*.

Peng XB and Van De Panne M (2017) Learning locomotion skills using deeprl: does the choice of action space matter? In: *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, 1–13.

Peng XB, Berseth G, Yin K, et al. (2017) Deeploco: dynamic locomotion skills using hierarchical deep reinforcement learning. *ACM Transactions on Graphics* 36(4): 1–13.

Peng XB, Abbeel P, Levine S, et al. (2018a) Deepmimic: example-guided deep reinforcement learning of physics-based character skills. *ACM Transactions on Graphics* 37(4): 1–14.

Peng XB, Andrychowicz M, Zaremba W, et al. (2018b) Sim-to-real transfer of robotic control with dynamics randomization. In: *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 3803–3810.

Peng XB, Coumans E, Zhang T, et al. (2020) Learning agile robotic locomotion skills by imitating animals. arXiv preprint arXiv:2004.00784.

Peng XB, Guo Y, Halper L, et al. (2022) Ase: large-scale reusable adversarial skill embeddings for physically simulated characters. *ACM Transactions On Graphics (TOG)ACM Transactions on Graphics* 41(4): 1–17.

Pham Q, Madhavan R, Righetti L, et al. (2018) The impact of robotics and automation on working conditions and employment [ethical, legal, and societal issues]. *IEEE Robotics and Automation Magazine* 25(2): 126–128.

Pippine J, Hackett D and Watson A (2011) An overview of the defense advanced research projects agency's learning locomotion program. *The International Journal of Robotics Research* 30(2): 141–144.

Pomerleau DA (1988) Alvinn: an autonomous land vehicle in a neural network. *Advances in Neural Information Processing Systems* 1: 1.

Ponton B, Farshidian F and Buchli J (2014) Learning compliant locomotion on a quadruped robot. In *IROS Workshop (Ed.), Compliant Manipulation: Challenges in Learning and Control*. Citeseer.

Ponton B, Khadiv M, Meduri A, et al. (2021) Efficient multicontact pattern generation with sequential convex approximations of the centroidal dynamics. *IEEE Transactions on Robotics* 37(5): 1661–1679.

Posa M, Cantu C and Tedrake R (2014) A direct method for trajectory optimization of rigid bodies through contact. *The International Journal of Robotics Research* 33(1): 69–81.

Pratt J, Dilworth P and Pratt G (1997) Virtual model control of a bipedal walking robot. In: *Proceedings of international conference on robotics and automation*. IEEE, Vol. 1, 193–198.

Pua X and Khadiv M (2024) Safe learning of locomotion skills from mpc. In: *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*. URL. https://arxiv.org/pdf/2407.11673.

Radosavovic I, Xiao T, Zhang B, et al. (2024) Real-world humanoid locomotion with reinforcement learning. *Science Robotics* 9(89): eadi9579.

Raibert MH (1984) Hopping in legged systems—modeling and simulation for the two-dimensional one-legged case. *IEEE Transactions on Systems, Man, and Cybernetics*: 451–463.

Raibert MH (1986) *Legged Robots that Balance*. MIT press.

Raibert MH, Blankespoor K, Nelson GM, et al. (2008) Bigdog, the rough-terrain quadruped robot. *IFAC Proceedings Volumes* 41: 10822–10825, URL. https://api.semanticscholar.org/CorpusID:8414062.

Raković SV and Levine WS (2018) *Handbook of Model Predictive Control*. Springer.

Reske A, Carius J, Ma Y, et al. (2021) Imitation learning from mpc for quadrupedal multi-gait control. In: *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 5014–5020.

Righetti L, Buchli J, Mistry M, et al. (2013) Optimal distribution of contact forces with inverse-dynamics control. *The International Journal of Robotics Research* 32(3): 280–298.

Righetti L, Pham Q-C, Madhavan R, et al. (2018) Lethal autonomous weapon systems [ethical, legal, and societal issues]. *IEEE Robotics and Automation Magazine* 25(1): 123–126.

Rong X, Li Y, Ruan J, et al. (2012) Design and simulation for a hydraulic actuated quadruped robot. *Journal of Mechanical Science and Technology* 26: 1171–1177.

Ross S, Gordon G and Bagnell D (2011) A reduction of imitation learning and structured prediction to no-regret online learning. In: *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*. JMLR Workshop and Conference Proceedings, 627–635.

Rudin N (2021a) Isaac gym environments for legged robots. URL. https://github.com/leggedrobotics/legged_gym.

Rudin N (2021b) *Rsl rl*. URL. https://github.com/leggedrobotics/rsl_rl.git.

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

Rudin N, Hoeller D, Reist P, et al. (2022b) Learning to walk in minutes using massively parallel deep reinforcement learning. In: *Conference on Robot Learning*. PMLR, 91–100.

Rummery GA and Niranjan M (1994) *On-line Q-Learning Using Connectionist Systems*. Cambridge, UK: University of Cambridge, Department of Engineering, Vol. 37.

Ruppert F and Badri-Spröwitz A (2022) Learning plastic matching of robot dynamics in closed-loop central pattern generators. *Nature Machine Intelligence* 4(7): 652–660.

Saggar M, D'Silva T, Kohl N, et al. (2007) Autonomous learning of stable quadruped locomotion. In: *RoboCup 2006: Robot Soccer World Cup X 10*. Springer, 98–109.

Sarmadi A, Krishnamurthy P and Khorrami F (2023) High-dimensional controller tuning through latent representations. arXiv preprint arXiv:2309.12487.

Schaal S (2009) The sl simulation and real-time control software package. Citeseer. Technical report.

Schaal S (2018) Historical perspective of humanoid robot research in the Americas. In: *Humanoid Robotics: A Reference*. Netherlands, Dordrecht: Springer, 1–9. DOI: 10.1007/978-94-007-7194-9_143-1.

Schilling M, Paskarbeit J, Ritter H, et al. (2021) From adaptive locomotion to predictive action selection–cognitive control for a six-legged walker. *IEEE Transactions on Robotics* 38(2): 666–682.

Schuitema E, Wisse M, Ramakers T, et al. (2010) The design of leo: a 2d bipedal walking robot for online autonomous reinforcement learning. In: *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 3238–3243.

Schulman J, Levine S, Abbeel P, et al. (2015) Trust region policy optimization. In: *International Conference on Machine Learning*. PMLR, 1889–1897.

Schulman J, Wolski F, Dhariwal P, et al. (2017) Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Schwarke C, Klemm V, Van der Boon M, et al. (2023) Curiosity-driven learning of joint locomotion and manipulation tasks. In *Proceedings of the 7th Conference on Robot Learning*. PMLR, Vol. 229, 2594–2610.

Schwarke C, Klemm V, Tordesillas J, et al. (2024) Learning quadrupedal locomotion via differentiable simulation. arXiv preprint arXiv:2404.02887.

Semini C (2010) Hyq-design and development of a hydraulically actuated quadruped robot. In: *Doctor of Philosophy (Ph. D.)*. University of Genoa, Italy.

Semini C, Barasuol V, Goldsmith J, et al. (2016) Design of the hydraulically actuated, torque-controlled quadruped robot hyq2max. *IEEE/Asme Transactions on Mechatronics* 22(2): 635–646.

Seo M, Han S, Sim K, et al. (2023) Deep imitation learning for humanoid loco-manipulation through human teleoperation. In: *2023 IEEE-RAS 22nd International Conference on Humanoid Robots (Humanoids)*. IEEE, 1–8.

Seok S, Wang A, Chuah MY, et al. (2014) Design principles for energy-efficient legged locomotion and implementation on the mit cheetah robot. *Ieee/asme transactions on mechatronics* 20(3): 1117–1129.

Shafiee M, Bellegarda G and Ijspeert A (2023) Puppeteer and marionette: learning anticipatory quadrupedal locomotion based on interactions of a central pattern generator and supraspinal drive. In: *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 1112–1119.

Sharma A, Ahn M, Levine S, et al. (2020) Emergent real-world robotic skills via unsupervised off-policy reinforcement learning. arXiv preprint arXiv:2004.12974.

Shen Y, Guo D, Long F, et al. (2020) Robots under covid-19 pandemic: a comprehensive survey. *IEEE Access* 9: 1590–1615.

Shin Y-H, Hong S, Woo S, et al. (2022) Design of kaist hound, a quadruped robot platform for fast and efficient locomotion with mixed-integer nonlinear optimization of a gear train. In: *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 6614–6620.

Shkolnik A, Levashov M, Manchester IR, et al. (2011) Bounding on rough terrain with the littledog robot. *The International Journal of Robotics Research* 30(2): 192–215.

Sidor z. (2021) Stable baselines. URL. https://github.com/hill-a/stable-baselines.

Siekmann J, Godse Y, Fern A, et al. (2021) Sim-to-real learning of all common bipedal gaits via periodic reward composition. In: *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 7309–7315.

Sleiman J-P, Farshidian F, Minniti MV, et al. (2021) A unified mpc framework for whole-body dynamic locomotion and manipulation. *IEEE Robotics and Automation Letters* 6(3): 4688–4695.

Smith F (2016) *Sarcos*. Netherlands, Dordrecht: Springer, 1–11. DOI: 10.1007/978-94-007-7194-9_22-1.

Smith R, (2005) Open dynamics engine. URL. https://www.ode.org/.

Smith L, Kostrikov I and Levine S. (2022) A walk in the park: learning to walk in 20 minutes with model-free reinforcement learning. arXiv preprint arXiv:2208.07860.

Smith L, Kew JC, Li T, et al. (2023) Learning and adapting agile locomotion skills by transferring experience. *arXiv preprint arXiv:2304.09834*.

Sorokin M, Tan J, Liu CK, et al. (2022) Learning to navigate sidewalks in outdoor environments. *IEEE Robotics and Automation Letters* 7(2): 3906–3913.

Spröwitz A, Tuleu A, Vespignani M, et al. (2013) Towards dynamic trot gait locomotion: design, control, and experiments with cheetah-cub, a compliant quadruped robot. *The International Journal of Robotics Research* 32(8): 932–950.

Suh HJT, Pang T and Tedrake R (2022) Bundled gradients through contact via randomized smoothing. *IEEE Robotics and Automation Letters* 7(2): 4000–4007.

Sutton RS and Barto AG (2018) *Reinforcement Learning: An Introduction*. MIT press.

Tan J, Zhang T, Coumans E, et al. (2018) Sim-to-real: learning agile locomotion for quadruped robots. arXiv preprint arXiv: 1804.10332.

Tang Y, Yu W, Tan J, et al. (2023) Language to quadrupedal locomotion. arXiv preprint arXiv:2306.07580.

Taouil I, Turrisi G, Schleich D, et al. (2023) Quadrupedal footstep planning using learned motion models of a black-box controller. In: *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 800–806.

Tassa Y and Todorov E (2011) Stochastic complementarity for local control of discontinuous dynamics. In: *Proceedings of Robotics: Science and Systems*.

Tassa Y, Erez T and Todorov E (2012) Synthesis and stabilization of complex behaviors through online trajectory optimization. In: *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 4906–4913.

Team G, Anil R, Borgeaud S, et al. (2023) Gemini: a family of highly capable multimodal models. arXiv preprint arXiv: 2312.11805.

Tedrake R, Zhang TW, Seung HS, et al. (2005) Learning to walk in 20 minutes. In: *Proceedings of the Fourteenth Yale Workshop on Adaptive and Learning Systems*. Beijing, Vol. 95585, p. 1939.

Thananjeyan B, Balakrishna A, Nair S, et al. (2021) Recovery rl: safe reinforcement learning with learned recovery zones. *IEEE Robotics and Automation Letters* 6(3): 4915–4922.

Theodorou E, Buchli J and Schaal S (2010) Reinforcement learning of motor skills in high dimensions: a path integral approach. In: *2010 IEEE International Conference on Robotics and Automation*. IEEE, 2397–2403.

Tobin J, Fong R, Ray A, et al. (2017) Domain randomization for transferring deep neural networks from simulation to the real world. In: *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 23–30.

Todorov E, Erez T and Tassa Y (2012) Mujoco: a physics engine for model-based control. In: *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 5026–5033.

Touvron H, Lavril T, Izacard G, et al. (2023) Llama: open and efficient foundation language models. arXiv preprint arXiv: 2302.13971.

Truong J, Rudolph M, Yokoyama NH, et al. (2023) Rethinking sim2real: lower fidelity simulation leads to higher sim2real transfer in navigation. In: *Conference on Robot Learning*. PMLR, 859–870.

Tsounis V, Alge M, Lee J, et al. (2020) Deepgait: planning and control of quadrupedal gaits using deep reinforcement learning. *IEEE Robotics and Automation Letters* 5(2): 3699–3706.

Unitree (2021a) Unitree legged robots. URL. https://www.unitree.com.

Unitree (2021b) Unitree b2, go beyond the limits. URL. https://www.unitree.com/b2/.

Unitree (2021c) Unitree h1 the world's first full-size motor drive humanoid robot flips on ground. URL. https://www.youtube.com/watch?v=V1LyWsiTgms.

Van Den Oord A, Vinyals O, kavukcuoglu K, et al. (2017) Neural discrete representation learning. *Advances in Neural Information Processing Systems* 30: 11.

Van Hasselt H, Guez A and Silver D (2016) Deep reinforcement learning with double q-learning. In: *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 30.

Viereck J and Righetti L (2021) Learning a centroidal motion planner for legged locomotion. In: *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 4905–4911.

Viereck J, Meduri A and Valuenetqp LR (2022) Learned one-step optimal control for legged locomotion. In: *Learning for Dynamics and Control Conference*. PMLR, 931–942.

Villarreal O, Barasuol V, Wensing PM, et al. (2020) Mpc-based controller with terrain insight for dynamic legged locomotion. In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2436–2442.

Vollenweider E, Bjelonic M, Klemm V, et al. (2023) Advanced skills through multiple adversarial motion priors in reinforcement learning. In: *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 5120–5126.

Wabersich KP, Taylor AJ, Choi JJ, et al. (2023) Data-driven safety filters: Hamilton-Jacobi reachability, control barrier functions, and predictive methods for uncertain systems. *IEEE Control Systems Magazine* 43(5): 137–177.

Wang Z, Schaul T, Hessel M, et al. (2016) Dueling network architectures for deep reinforcement learning. In: *International Conference on Machine Learning*. PMLR, 1995–2003.

Wang J, Lembono TS, Kim S, et al. (2022) Learning to guide online multi-contact receding horizon planning. In: *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 12942–12949.

Watkins CJ and Dayan P (1992) Q-learning. *Machine Learning* 8: 279–292.

Wensing PM, Wang A, Seok S, et al. (2017) Proprioceptive actuator design in the mit cheetah: Impact mitigation and high-bandwidth physical interaction for dynamic legged robots. *Ieee transactions on robotics* 33(3): 509–522.

Wensing PM, Posa M, Hu Y, et al. (2023) Optimization-based control for dynamic legged robots. *IEEE Transactions on Robotics*.

Werling K, Omens D, Lee J, et al. (2021) Fast and feature-complete differentiable physics for articulated rigid bodies with contact. arXiv preprint arXiv:2103.16021.

Widmer D, Kang D, Sukhija B, et al. (2023) Tuning legged locomotion controllers via safe bayesian optimization. In: *Conference on Robot Learning*. PMLR, 2444–2464.

Wieber P-B (2006) Trajectory free linear model predictive control for stable walking in the presence of strong perturbations. In: *2006 6th IEEE-RAS International Conference on Humanoid Robots*. IEEE, 137–142.

Wijmans E, Kadian A, Morcos A, et al. (2019) Dd-ppo: learning near-perfect pointgoal navigators from 2.5 billion frames. arXiv preprint arXiv:1911.00357.

Winkler AW, Bellicoso CD, Hutter M, et al. (2018) Gait and trajectory optimization for legged systems through phase-based end-effector parameterization. *IEEE Robotics and Automation Letters* 3(3): 1560–1567.

Won J, Gopinath D and Hodgins J (2022) Physics-based character controllers using conditional vaes. *ACM Transactions on Graphics* 41(4): 1–12.

Wu J, Xin G, Qi C, et al. (2023a) Learning robust and agile legged locomotion using adversarial motion priors. *IEEE Robotics and Automation Letters*.

Wu P, Escontrela A, Hafner D, et al. (2023b) Daydreamer: world models for physical robot learning. In: *Conference on Robot Learning*. PMLR, 2226–2240.

Xiao X, Liu J, Wang Z, et al. (2023) Robot learning in the era of foundation models: a survey. arXiv preprint arXiv: 2311.14379.

Xie Z, Clary P, Dao J, et al. (2020a) Learning locomotion skills for cassie: iterative design and sim-to-real *Conference on Robot Learning*. PMLR, 317–329.

Xie Z, Ling HY, Kim NH, et al. (2020b) ALLSTEPS: curriculum-driven learning of stepping stone skills. In: *Computer Graphics Forum*. Wiley Online Library, Vol. 39, 213–224.

Xie Z, Da X, Van de Panne M, et al. (2021) Dynamics randomization revisited: a case study for quadrupedal locomotion. In: *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 4955–4961.

Xie Z, Da X, Babich B, et al. (2022) Glide: generalizable quadrupedal locomotion in diverse environments with a centroidal model. In: *International Workshop on the Algorithmic Foundations of Robotics*. Springer, 523–539.

Xu S, Zhu L and Ho CP (2022) Learning efficient and robust multimodal quadruped locomotion: a hierarchical approach. In: *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 4649–4655.

Xu M, Huang P, Yu W, et al. (2023) Creative robot tool use with large language models. *arXiv preprint arXiv:2310.13065*.

Xu Z, Raj AH, Xiao X, et al. (2024) Dexterous legged locomotion in confined 3d spaces with reinforcement learning. arXiv preprint arXiv:2403.03848.

Yam KC, Tang PM, Jackson JC, et al. (2023) The rise of robots increases job insecurity and maladaptive workplace behaviors: multimethod evidence. *Journal of Applied Psychology* 108(5): 850–870.

Yamane K and Nakamura Y (2006) Stable penalty-based model of frictional contacts. In: *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006*. IEEE, 1904–1909.

Yang Y, Caluwaerts K, Iscen A, et al. (2020) Data efficient reinforcement learning for legged robots. In: *Conference on Robot Learning*. PMLR, 1–10.

Yang R, Zhang M, Hansen N, et al. (2021) Learning vision-guided quadrupedal locomotion end-to-end with cross-modal transformers. arXiv preprint arXiv:2107.03996.

Yang L, Li Z, Zeng J, et al. (2022a) Bayesian optimization meets hybrid zero dynamics: safe parameter learning for bipedal locomotion control. In: *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 10456–10462.

Yang T-Y, Zhang T, Luu L, et al. (2022b) Safe reinforcement learning for legged locomotion. In: *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2454–2461.

Yang Y, Zhang T, Coumans E, et al. (2022c) Fast and efficient locomotion via learned gait transitions. In: *Conference on Robot Learning*. PMLR, 773–783.

Yang R, Chen Z, Ma J, et al. (2023a.) Generalized animal imitator: agile locomotion with versatile motion prior. arXiv preprint arXiv:2310.01408.

Yang R, Yang G and Wang X (2023b) Neural volumetric memory for visual locomotion control. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1430–1440.

Yang Y, Meng X, Yu W, et al. (2023c) Learning semantics-aware locomotion skills from human demonstration. In: *Conference on Robot Learning*. PMLR, 2205–2214.

Yeganegi MH, Khadiv M, Moosavian SAA, et al. (2019) Robust humanoid locomotion using trajectory optimization and sample-efficient learning. In: *2019 IEEE-RAS 19th International Conference on Humanoid Robots (Humanoids)*. IEEE, 170–177.

Yeganegi MH, Khadiv M, Del Prete A, et al. (2021) Robust walking based on mpc with viability guarantees. *IEEE Transactions on Robotics*.

Youm D, Jung H, Kim H, et al. (2023) Imitating and finetuning model predictive control for robust and symmetric quadrupedal locomotion. *IEEE Robotics and Automation Letters*.

Yu W, Tan J, Liu CK, et al. (2017) Preparing for the unknown: learning a universal policy with online system identification. In: *Proceedings of Robotics: Science and Systems*. Cambridge, Massachusetts. DOI: 10.15607/RSS.2017.XIII.048.

Yu W, Kumar VC, Turk G, et al. (2019) Sim-to-real transfer for biped locomotion. In: *2019 IEEE/rsj International Conference on Intelligent Robots and Systems (Iros)*. IEEE, 3503–3510.

Yu W, Tan J, Bai Y, et al. (2020) Learning fast adaptation with meta strategy optimization. *IEEE Robotics and Automation Letters* 5(2): 2950–2957.

Yu W, Jain D, Escontrela A, et al. (2021) Visual-locomotion: learning to walk on complex terrains with vision. In: *5th Annual Conference on Robot Learning*.

Yu W, Gileadi N, Fu C, et al. (2023a) Language to rewards for robotic skill synthesis. In: *Conference on Robot Learning*. PMLR, 374–404.

Yu W, Yang C, McGreavy C, et al. (2023b) Identifying important sensory feedback for learning locomotion skills. *Nature Machine Intelligence* 5(8): 919–932.

Yuan K, Chatzinikolaidis I and Li Z (2019) Bayesian optimization for whole-body control of high-degree-of-freedom robots

through reduction of dimensionality. *IEEE Robotics and Automation Letters* 4(3): 2268–2275.

Zhang H, He L and Wang D (2022) Deep reinforcement learning for real-world quadrupedal locomotion: a comprehensive review. *Intelligence and Robotics* 2(3): 11.

Zhang C, Rudin N, Hoeller D, et al. (2023) Learning agile locomotion on risky terrains. arXiv preprint arXiv:2311.10484.

Zhang J, Heim S, Jeon SH, et al. (2024) Learning emergent gaits with decentralized phase oscillators: on the role of observations, rewards, and feedback. arXiv preprint arXiv:2402.08662.

Zhou H, Yao X, Meng Y, et al. (2023) Language-conditioned learning for robotic manipulation: a survey. arXiv preprint arXiv:2312.10807.

Zhu T (2023) *Design of a Highly Dynamic Humanoid Robot*. Los Angeles: University of California.

Zhu Q, Zhang H, Lan M, et al. (2023) Neural categorical priors for physics-based character control. *ACM Transactions on Graphics* 42(6): 1–16.

Zhuang Z, Fu Z, Wang J, et al. (2023) Robot parkour learning. arXiv preprint arXiv:2309.05665.

Zico Kolter J and Ng AY (2011) The stanford littledog: a learning and rapid replanning approach to quadruped locomotion. *The International Journal of Robotics Research* 30(2): 150–174.

Zucker M, Ratliff N, Stolle M, et al. (2011) Optimization and learning for rough terrain legged locomotion. *The International Journal of Robotics Research* 30(2): 175–191.