OXFORD JOURNAL
OF STUDENT
SCHOLARSHIP

Why Do Quadruped Locomotion Policies Fail When Transferred from Simulation to Real Hardware? A Systematic Analysis of Failure Modes Across MPC, RL, and IL

Amr Ahmed
amrwaelashraf2009@gmail.com

ABSTRACT

Sim-to-real transfer remains one of the main challenges in deploying quadruped controllers, as discrepancies between simulated and real-world dynamics frequently cause policies that succeed in simulation to fail on hardware. This paper presents a structured, systematic review comparing how three dominant control paradigms: Model Predictive Control, Reinforcement Learning, and Imitation Learning handle sim-to-real transfer in quadruped locomotion, alongside hybrid approaches that combine elements of all three. Reviewed studies were organized using a four-dimensional taxonomy: control paradigm, gap source, failure manifestation, and mitigation strategy. Results show that MPC offers strong physical interpretability and constraint handling but is vulnerable to inaccurate contact and inertia models. RL demonstrates robust terrain adaptation but is prone to overfitting to simulation physics. IL produces structured, natural-looking behaviours but struggles when simulation observations do not match real-world sensing constraints. Hybrid architectures, which embed learned components into model-based frameworks or combine multiple paradigms, showed the most reliable transfer by compensating for the weaknesses of each paradigm. The study concludes that standardized evaluation benchmarks and unified sim-to-real assessment pipelines are needed to enable rigorous cross-paradigm comparison in future work.

INTRODUCTION

Locomotion is one of the many tasks a robot can perform. There is an argument to be made that it is the most important, as it dictates the motion of the robot and how it behaves when navigating terrain. Quadruped Locomotion is the movement of an organism or machine from one place to another using four limbs, with distinct patterns of limb coordination and body mechanics [1].

Quadruped robots have many uses; they are helpful due to their biomimetic design and superior adaptability, allowing them to navigate complex and uneven terrains better than wheeled and tracked counterparts. [2] Making them a valuable solution to tasks that were previously impossible to complete

using conventional wheeled robots. Another aspect that makes quadruped robots particularly helpful is their efficient dynamic performance and adaptation to environments.[3]

One downside to quadruped robots is that they can be hard to train and control. This is due to many factors, including the need for fast dynamic locomotion and precise movements; any small errors can compound and will ultimately lead to the failure of the robot. Another is sim-to-real transfer, which is the focus of this study. These machines are often trained on data using simulation software. After the training process ends, the trained model is then transferred to the hardware. This process is often the root cause of most issues; simulations cannot always take into account all the different variables that are present in the real world.[4]

Many algorithms are used to train quadruped robots; the focus of this study will be on 3 different paradigms: MPC (Model Predictive Control), IL (Imitation Learning), and RL (Reinforcement Learning). MPC is a control methodology that operates by repeatedly solving a short-horizon optimization problem, applying only the first control action, then re-solving at the next timestep with updated state information. In quadruped locomotion, that means that the controller predicts how the robot will move over the next few steps, chooses foot placements and adjustments that satisfy dynamics and contact constraints, and then keeps updating as the robot moves. Reinforcement learning works by letting a robot learn through trial and error: it tries actions, receives a reward for behaviour that is deemed to be positive, and updates its policy to receive higher rewards for behaviour that should be reinforced.[5]

Imitation learning works by having the robot learn from demonstrations instead of learning purely through trial and error. In quadruped locomotion, this usually means that the policy is trained to copy expert motions coming from motion capture, teleoperation, or trajectories generated by a stronger controller such as MPC.

The purpose of this study is to compare the three paradigms using a multi-variable taxonomy. These variables are: paradigm type, sim-to-real failure, failure manifestation, and gap source. The study aims to gain a better understanding of how sim-to-real failure manifests and how it can be avoided across the different paradigms.

LITERATURE REVIEW

Quadruped locomotion and control paradigms

Quadruped locomotion in robotics explains the cyclic motion of four-legged robots that traverse multiple types of terrain. This capability is paramount for applications such as search and rescue, inspection, and logistics, where wheeled robots struggle with obstacles such as stairs, rubble, and uneven terrain [4]. Recent years have seen a shift from manually tuned controllers to learning-based and model-based schemes that can adapt to changing conditions. The most prominent control paradigms are model predictive control (MPC), reinforcement learning (RL), and imitation learning; they are often used in

hybrid combinations. Each paradigm interacts differently with the sim-to-real gap; comparing their failure modes and mitigation strategies provides a framework for understanding modern quadruped research [4].

Sim-to-real as a failure taxonomy

The sim-to-real gap in quadruped locomotion stems from mismatches in dynamics, contact behaviour, actuator response, and sensor noise that arise between simulation and hardware. Across the three algorithms (MPC, RL, and IL), these mismatches produce similar failure modes: unstable gaits, slipping, tripping, and collapses when a controller trained or tuned in simulation is naively deployed on real robots [4]. Instead of treating each paper in isolation, this paper frames the problem as a failure taxonomy, organizing approaches along four dimensions: the control paradigm (e.g., MPC vs RL vs IL), the gap source (e.g., inertia, friction, latency), the failure manifestation (e.g., falls, task failure), and the mitigation strategy (e.g., domain randomization, MPC reformulation, IL-based policy distillation). This taxonomy provides researchers with a comprehensive basis for comparison.

Model Predictive Control in sim-to-real quadrupeds

Model predictive control provides an optimization-based framework for whole-body locomotion, utilizing dynamics and contact constraints to plan stable trajectories and torque profiles. In MPC-based quadrupeds, the sim-to-real gap stems from inaccurate contact and inertia models, which cause the planner to generate contact forces that are too aggressive or insufficient for the platform [30]. Common failures include slipping, loss of balance, and an inability to complete locomotion or manipulation tasks, even when the same controller performs well in an idealized simulator. Mitigation strategies emphasize contact phase scheduling, conservative torque and force bounds, and online parameter tuning. This is often combined with system identification steps that refine the simulator's dynamics using real-world data [35]. These interventions reduce the sensitivity of MPC to model errors and enable more reliable transfer to hardware.

Reinforcement Learning and Imitative Adaptation

Reinforcement learning has become a dominant method for sim-to-real quadruped locomotion, as it features policies that adapt to stairs and rough terrain [4]. However, RL policies are highly sensitive to the reality gap that comes with sim-to-real transfer, because they easily overfit to idealized simulation physics, sensor streams, and fixed environments. When deployed, they often show unstable gaits. The workaround for this issue is explicit regularization [32],[34]. To avoid this, researchers have proposed compact, noise-resistant observation spaces, domain randomization tailored to the most sensitive parameters, and curriculum-style training that increases the difficulty of simulated environments. [24] On the other hand, Imitation learning methods exploit expert demonstrations or teacher-student architectures and latent space dynamics models to decouple perception from control. This leads to improved generalization across terrain and sensor modalities.

Hybrid control

A growing trend is the integration of MPC, RL, and IL into hybrid control architectures that blend the strengths of each paradigm. For example, RL-augmented MPC combines a model-based planner for stance-foot control with a learned RL module for swing-foot adaptation, enabling robust stair-climbing and obstacle negotiation. [25],[29] Similarly, MPC-guided IL uses trajectory-optimization outputs as privileged references for an imitation policy that operates in the real world. These hybrid schemes typically reduce the direct impact of the sim-to-real gap by constraining the learning component to outputs that are already physically plausible and stable. Emerging frameworks such as PACE and DrEureka further refine this idea by treating sim-to-real transfer as a multi-stage pipeline, where simulator adaptation, controller redesign, and reward shaping are systematically applied to all three paradigms. As a result, the literature increasingly converges on a cross-paradigm view: MPC, RL, and IL are not isolated boxes, but interlocking pieces of a broader failure-mitigation strategy for robust quadruped locomotion. [39]

Current trends, limitations, and future directions

Recent work shows a clear shift from traditional, manually tuned controllers toward hybrid MPC-RL-IL schemes that emphasize robustness, adaptability, and explicit consideration of the reality gap [29]. Strengths include improved terrain handling, faster adaptation to changing conditions, and the ability to deploy policies with minimal manual tuning. At the same time, key limitations remain: high computational cost, the need for costly and accurate simulators, and a lack of standardized benchmarks that quantify sim-to-real failure modes across paradigms. Many studies still rely on closed-loop sim-to-sim evaluations or limited real-world testing, making direct comparison difficult. Future work should focus on bridging the sim-to-real gap through unified evaluation suites, energy-efficient control architectures, and tools for sim-to-real predictability that can be uniformly applied to MPC, RL, and IL [15]. Such advances will help transition quadruped locomotion from a research-bench curiosity to a robust, deployable technology for real-world tasks.

METHODOLOGY

Review design:

This study adopted a structured systematic review design to examine sim-to-real transfer failure in quadruped locomotion. This design was chosen because the literature includes multiple control paradigms, including Model Predictive Control (MPC), Reinforcement Learning (RL), Imitation Learning (IL), and hybrid approaches. The studies differ in objectives, robot platforms, and evaluation settings. Rather than quantitatively combining results, this review synthesizes findings thematically to identify common failure modes, their underlying causes, and the mitigation strategies proposed across the literature.

The review organizes the literature using a four-dimensional taxonomy consisting of control paradigm, gap source, failure manifestation, and mitigation strategy. This review design also supports transparency and reproducibility by using explicit search terms, inclusion criteria, and a consistent coding framework for extracting the information from each study.

Search strategy:

The literature search was conducted between March/2026- July 2026 using IEEE Xplore, Google Scholar, arXiv, ACM Digital Library, and ScienceDirect. These databases were selected because they collectively provide broad coverage of robotics, machine learning, and control systems research.

The search strategy used targeted keywords across major robotics databases and research repositories to identify studies on sim-to-real in quadruped locomotion. Searches were conducted using a Boolean combination of terms. Representative search strings included:

- ("quadruped robot" OR "quadruped locomotion") AND ("sim-to-real" OR "simulation-to-reality")
- ("quadruped locomotion") AND ("reinforcement learning" OR "model predictive control" OR "imitation learning")
- ("quadruped locomotion") AND (“MPC, reinforcement learning hybrid” OR "reinforcement learning, imitation learning hybrid”)
- ("quadruped robot") AND ("hybrid control") AND ("sim-to-real transfer")

Minor variations of these search terms were used where appropriate to accommodate differences in database search syntax.

Retrieved results were screened in three stages: title screening, abstract screening, and full-paper screening. The search initially identified 80 records. After removing duplicate entries. 68 records were screened by title and abstract, of which 50 papers underwent full-text assessment; 42 studies satisfied the inclusion criteria. These papers were included in the final review. In comparison, 8 papers were excluded for not meeting the eligibility requirements **Figure (1)**

Figure 1. PRISMA flow diagram of study identification, screening, and inclusion

Records identified through database searching (n = 80)	
↓	**Duplicate records removed** (n = 12)
Records screened — title and abstract	

(n = 68)	
↓	**Records excluded at screening** (n = 18)
Full-text articles assessed for eligibility (n = 50)	
↓	**Full-text articles excluded** (n = 8)
Studies included in the review (n = 42)	

Eligibility criteria

- Studies were included if they:
- Investigated quadruped robot locomotion.
- Mentioned sim-to-real transfer or analyzed the reality gap.
- Examined one or more of the following control paradigms: Model Predictive Control, Reinforcement Learning, Imitation Learning, or hybrid approaches.
- Reported failure modes, mitigation strategies, or hardware validation related to sim-to-real transfer.

Studies were excluded if:

- Did not mention quadruped robots.
- Did not address sim-to-real transfer.
- Focused exclusively on hardware or mechanical design without discussing locomotion control.
- Were editorials, abstracts, tutorials, or non-technical articles lacking sufficient methodological detail.

Data extraction:

Data from each included study were extracted using a standardized coding framework. For every paper, the following information was recorded:

- Paper title
- Control paradigm
- Source of the sim-to-real gap
- Failure manifestation
- Mitigation strategy

Each study was then classified according to the four-dimensional taxonomy developed for this review. Where a study did not explicitly report one of the variables, the corresponding field was recorded as "Not specified". This standardized coding procedure enabled systematic comparison across studies despite differences in robot conditions.

RESULTS

MPC

As shown from the results in Table (2), MPC's performance is highly dependent on several factors, one of the most important being the accuracy of the model used to make the predictions.[6],[7],[42] Across the extracted papers, the main source of sim-to-real difficulty is the mismatch between the simplified dynamics assumed by the controller and the complex behaviour of the real robot. This mismatch appears in several forms, including inaccurate inertial parameters, reduced-order body models, unmodeled payload uncertainty in friction, imperfect contact assumptions, and disturbances from external factors that are not represented well in simulation.[6],[9],[40]

When these factors are present, the controller may generate force commands or foot trajectories that look feasible in simulation but prove to be unstable on hardware.[28],[41]

The results also suggest that successful MPC transfer depends heavily on implementation details. Papers that report better real-world behaviour typically use conservative or explicitly constrained ground-reaction forces (e.g., friction cones, chance constraints, robust cones), well-structured or event-based contact timing, and explicit attention to foot placement and body stabilization through tailored costs or hierarchical whole-body tracking.[8] Rather than relying on optimization quality, they make the controller more robust by limiting the actions to physically realistic regions. [8],[42]

Overall, the trade-off that MPC offers shows that it provides strong interpretability, constraint handling, and stability-oriented planning, but it is vulnerable to modelling errors and contact uncertainty. The

reviewed studies indicate that the most effective MPC systems are those that account for uncertainty using adaptive formulations.[9] This makes MPC a strong paradigm for quadruped locomotion, especially when precise physical reasoning is needed, but also highlights why hybrid approaches are becoming increasingly attractive. [26]

Reinforcement learning:

The reinforcement learning literature shows that RL is highly effective for quadruped locomotion, but only when the training setup is carefully designed to handle the sim-to-real gap.[12],[16],[31] Across the reviewed studies, the main challenge is that policies trained in simulation often fail to transfer directly to hardware because the simulator does not fully capture actuator dynamics, contact behaviour, latency variation, or disturbance effects, a pattern demonstrated empirically through controlled ablation studies on legged platforms, where removing dynamics randomization, perturbation training, or well-chosen state variables measurably degraded real-world transfer even when simulated performance remained high [11], [18].

This discrepancy is repeatedly identified as a major source of performance degradation, particularly when the learned policy encounters conditions that were not represented during training. [18]A comparable pattern is reported in bipedal sim-to-real research, where the same categories of dynamics, contact, and sensing mismatches are analyzed using quadruped-derived techniques such as domain randomization on ANYmal [10], suggesting that these sources of gaps are not morphology-specific. [11]

A common failure pattern across the RL papers is poor transfer from simulation to hardware or from one terrain to another. In practice, this shows up as unstable gaits, loss of balance, poor performance on unseen terrains, or failure when the robot encounters rough ground, stairs, obstacles, or faults that were not represented during training.[13],[19],[32] Additionally, reinforcement learning can be computationally costly and complicated, especially when using it with a dynamic system such as a quadruped in motion [12][31]. Some papers also show that RL controllers can be especially vulnerable to joint failures, payload changes, and other disturbances if those conditions were not included in the training setup. The overall consensus is that RL policies generalize best when the training environment deliberately includes the kinds of uncertainty expected at deployment [4].

To address these issues, the reviewed studies use several mitigation strategies. One major strategy is domain randomization, where physical parameters, terrain conditions, sensor noise, or disturbance effects are varied during training so that the policy learns to code with uncertainty. [13],[16],[18] Another approach is to improve simulator fidelity through system identification, actuator modelling, and latency modelling so that the training environment better matches the real robot.[31],[36] Some papers combine both ideas: refining the simulator first, then training the policy under randomized conditions, which

improves the hardware transfer. These strategies appear repeatedly and are among the most common ways to reduce the simulation-to-real gap.

The RL studies also show that observation and reward design substantially affect performance [12],[13],[14],[33]. Several papers use proprioceptive-only inputs to reduce dependence on vision and improve portability across terrains [14],[32], while others incorporate terrain information or multimodal representations when simple proprioception is insufficient [15]. This suggests that RL transfer depends not only on the algorithm itself but also on what the policy observes and what it is rewarded for.[15],[32],[33] Reward shaping can encourage stable, efficient, and symmetric gait; but is the reward is too narrow, the learned policy may only work well in the training environment [4]. The papers repeatedly show that careful policy design is needed to avoid brittle behaviour when deployed outside simulation.

Another major trend is hybrid or staged training pipelines. Several papers do not rely on plain end-to-end RL alone but instead combine RL with imitation learning (IL), teacher-student training, expert guidance, or multi-stage training.[17] In fault-tolerant locomotion specifically, RL is used to detect joint failures or recover from leg impairments by switching skills or adapting gaits. In other words, RL is combined with terrain information or teacher-driven training to improve generalization and efficiency. These results suggest that many of the strongest RL systems are structured policies borrowing from other control ideas to improve transfer.

Several papers also show successful zero-shot sim-to-real transfer when the training setup is carefully matched to the robot and task. These studies report robust locomotion, especially when disturbance randomization, fault modelling, or task-specific policy design are included in training [10],[18]. Some focus on rough terrain, as mentioned previously; others focus on joint failures [19], and others on gravity jumping or stair climbing [20], but all show that RL can transfer successfully when the simulator and pipeline match deployment conditions.

Imitation learning:

The imitation learning studies in the reviewed set show that successful transfer from simulation to hardware depends on how closely the training setup matches the real robot's sensing and actuation constraints.[21] Studies show that policies trained under observation spaces and dynamics aligned to the physical platform transfer more reliably than those relying on privileged simulator signals or idealized actuators [22].

Across the studies, the main source of sim-to-real difficulty is the mismatch between the richer state and perfect measurements available in simulation and the more limited, noisy information available on hardware, together with differences in actuator dynamics, contact behaviours, and terrain geometry; these gaps motivate domain randomization over friction, torque limits, mass, perception noise, and contact properties, as well as careful system modelling, to bridge the reality gap [23]. As a result, policies that reproduce expert or optimized demonstrations well in simulation may still fail to execute reliably on the

physical robot when confronted with unmodeled variations in terrain or dynamics, even though some works demonstrate that, with sufficient modelling and regularization, zero-shot transfer of complex behaviours such as hops, flips, or terrain-adaptive gaits is feasible [24]. A recurring result is that direct imitation of reference trajectories is often insufficient when the robot encounters conditions outside the training distribution, and this manifests as poor transfer when the real robot has fewer observable variables than the simulator, or when the terrain configurations differ from the reference motions used during training, leading to failures when stair dimensions or slopes deviate from those implicitly encoded in motion data.[22] This indicates that imitation learning is highly sensitive to both observation mismatch and dynamics mismatch, such that policies imitating animal motion, human motion, or trajectory-optimized references must be coupled with appropriate regularization, randomization, or adaptation to remain robust in the real world.[21] The reviewed studies addressed these limitations through several mitigation strategies: restricting the policy to deployable observations and avoiding privileged simulator-only state when designing inputs; combining imitation learning with reinforcement learning or latent-space domain adaptation so the policy can be fine-tuned after simulation pretraining; improving transfer by learning or using more accurate actuator and contact models. Including torque space parameterizations and motor or PD-controller models grounded in the real robot, and introducing terrain adaptation modules. Height map-based exteroceptive inputs, or randomized terrain parameters, so that the policy can generalize beyond a single reference motion or fixed environment.[24] Together, these methods show that imitation learning becomes more transferable when expert demonstrations, such as those from optimized trajectories, are treated as a structured prior and starting point for adaptation rather than a fixed target to be copied exactly, enabling the synthesis of controllable, statistically coherent, and hardware-executable locomotion primitives that can be reused across tasks.[22] Overall, the results suggest that imitation learning is effective for producing structured natural looking locomotion behaviours and diverse repertoires of skills, but that direct transfer is limited unless the learning pipeline is explicitly designed around the real robot's sensing, actuation, and terrain constants, with the strongest outcomes obtained from methods that combine imitation with adaptation, regularization, domain randomization, or staged training instead of relying on behaviour cloning alone.

Hybrid:

The hybrid control literature shows that combining model-based control with learning is effective for quadruped locomotion, but only when the interaction between the learned and model-based components is carefully designed to handle model mismatch and the sim-to-real gap. Across the reviewed hybrid studies, the main challenge is that the nominal MPC or reduced-order models do not fully capture contact dynamics, unmodeled forces, terrain variability, or changing robot parameters, which lead to degraded performance when moving from idealized assumptions to deployment on real robots or new platforms.[28],[40] This discrepancy is repeatedly identified as a key source of instability in the nominal model or initial controller design.[41],[42]

A common failure pattern in the hybrid papers is poor robustness when classical MPC or fixed parameter controllers are used alone. In practice, this appears as loss of stability or reduced tracking performance on

rough, slippery, or conveyor terrains, and as sensitivity to modelling errors such as inaccurate dynamic parameters, leg mass, or payload changes [28],[40],[42]. Some works also emphasize that reduced-order template models introduce a substantial gap to the full-order robot, so trajectories that are optimal in the simplified model may not yield robust locomotion on the hardware.[28] Overall, the consensus is that model-based controllers generalize best when they are augmented with data-driven components that explicitly learn to adapt to the residual dynamics and uncertainties encountered during operations. [25],[26],28],[40]

To address these issues, the hybrid studies adopt several mitigation strategies. One major strategy is to embed learned residual models inside MPC, where a neural network or online learner predicts unmodeled dynamics or decision parameters that the nominal model cannot capture.[26],[27],[28] For example, deep reinforcement learning is used to learn the unmodeled dynamics term in a robust MPC formulation, which plans single rigid-body trajectories and ground reaction forces that are then mapped to the full-order robot, enabling robust blind locomotion on different terrains.[28] Online residual learning with random Fourier features is employed to estimate modelling errors and external disturbances, which are then fed back to the MPC to maintain tracking performance under large unknown forces, slopes, and payload variations.[27] Learning-based MPC parameter tuning similarly trains a policy to choose MPC parameters automatically, improving velocity tracking and motion smoothness compared to fixed-parameter MPC.[25] On slippery ground, a learning-based MPC approach estimates friction factors via a heuristic expectation-maximization algorithm, overcoming the sensitivity of standard EM and adapting MPC to different surfaces.[28]

Another major trend is imitation or teacher-guided hybrid pipelines. Some works do not rely on pure RL or pure MPC but instead stage training to leverage expert behaviour and then refine it. [24],[29] For example, the IFM framework first constructs a conventional MPC controller as an expert, then uses imitation learning to clone it, and finally applies deep RL with limited exploration to fine-tune the policy on challenging terrains [29]. This staged design reduces the burden of reward shaping, constrains the learned policy to a region of the action space that is feasible for the robot, and yields gaits that are more symmetric, periodic, and energy-efficient than vanilla RL, while outperforming the original MPC on rough, slippery, and conveyor terrains.[29] Similarly, hybrid Kino dynamic MPC uses an RL policy trained via motion imitation to generate dynamically plausible trajectories and contact schedules, which then seed an MPC that stabilizes and robustifies the motion on a different robot. This structure enables zero-shot retargeting of trotting and pacing skills.

The hybrid literature also highlights that reinforcement learning components can help address computational and design challenges in model-based control. For instance, a predictive RL tail cost implemented as a Q-function allows MPC to operate with shorter horizons while maintaining stable gaits, reducing the exponential growth in computational complexity associated with long-horizon MPC.[30] Other works show that learning MPC parameters or decision variables can improve performance without requiring massive RL sampling, thereby avoiding the full complexity of end-to-end RL while still gaining adaptability.[25] Together, these results indicate that many of the strongest hybrid systems are structured

controllers that combine MPC, RL, and IL to explicitly compensate for modelling errors, handle terrain uncertainty, and improve transfer across robots and environments.

Table (1): Cross-Paradigm Summary of Sim-to-Real Failure Sources, Modes, Mitigations, Strengths, and Limitations

Control Paradigm	Dominant Sim-to-Real Gap Source	Characteristic Failure Mode	Common Mitigation Strategies	Strengths Reported	Limitations Reported
Model Predictive Control (MPC)	Model inaccuracies, simplified dynamics, contact uncertainty	Force commands or foot trajectories that appear feasible in simulation but prove unstable when deployed on hardware	Conservative/constrained ground-reaction force formulations, structured contact-timing schedules, tailored costs for foot placement and body stabilization, system identification, adaptive uncertainty-aware formulations	Strong stability, interpretable control, effective constraint handling	Performance depends heavily on accurate system models and is sensitive to modelling errors
Reinforcement Learning (RL)	Simulation-to-reality gap caused by inaccurate physics, actuator modelling, and terrain differences	Poor transfer to hardware, instability, or degraded performance on unseen conditions	Domain randomization, system identification, curriculum learning, robust observation design, high-fidelity simulation	Strong adaptability and robust locomotion on diverse terrains	Successful transfer depends on simulator fidelity, domain randomization, and reward design
Imitation Learning (IL)	Differences between simulated and real observations, dynamics, and actuator behaviour	Reduced transferability and limited generalization beyond demonstrated	Domain adaptation, terrain adaptation, realistic actuator models, fine-tuning after imitation learning	Produces structured and natural locomotion behaviours	Direct transfer is limited unless adaptation mechanisms are incorporated

Control Paradigm	Dominant Sim-to-Real Gap Source	Characteristic Failure Mode	Common Mitigation Strategies	Strengths Reported	Limitations Reported
		motions or environments			
Hybrid Approaches	Model mismatch, residual dynamics, terrain variability, and changing robot parameters	Reduced robustness when nominal models cannot capture real-world dynamics	Learned residual models, adaptive parameter tuning, staged MPC–IL–RL training, online learning	Reviewed studies indicate improved robustness and transferability by combining complementary control methods	Performance depends on careful integration of learning and model-based components

LIMITATIONS

This review is subject to potential publication bias. Studies that report successful sim-to-real transfer or novel mitigation strategies may be more likely to be published and indexed than studies reporting negative or null results. As a result, the prevalence of certain failure modes or the effectiveness of certain mitigation strategies may be overrepresented in the reviewed literature relative to the true distribution of outcomes across all research conducted in this area.

The search was restricted to English-language publications. Relevant work published in other languages was not captured by the search strategy, which may have excluded studies particularly from non-English-speaking robotics research groups that would otherwise meet the eligibility criteria.

Several included studies exhibited inconsistent reporting of key variables, particularly failure sources and mitigation strategies. Where a study did not explicitly report a variable, it was coded as "not specified" rather than inferred, which may understate the true prevalence of certain failure modes or mitigation approaches in the underlying literature.

Sim-to-real transfer for quadruped locomotion is a rapidly evolving research area. New control paradigms, simulation platforms, and mitigation strategies continue to emerge, meaning findings from studies published even a short time before the search period may not reflect the current state of the art.

The review's synthesis reflects the literature available as of the search window (March–July 2026) and may not capture more recent developments.

This review did not conduct a quantitative or meta-analytic comparison across studies, due to inconsistency in how authors reported results. Included studies used different metrics to characterize sim-to-real performance (e.g., success rate, tracking error, number of hardware trials, simulation-to-real performance drop), measured under different experimental conditions, robot platforms, and evaluation protocols, without a shared baseline or standardized unit of comparison. This heterogeneity made it infeasible to aggregate results numerically or to statistically compare the relative effectiveness of different control paradigms or mitigation strategies. As a result, the conclusions of this review are necessarily qualitative and thematic rather than quantitative, and claims about which approaches are "more effective" should be interpreted as patterns observed across the literature rather than statistically validated findings.

CONCLUSION

The goal of this study was to compare how different quadruped locomotion paradigms, Model predictive control, Reinforcement learning, Imitation learning, and hybrid approaches handle sim-to-real transfer challenges, and to identify the major sources of failure, how they manifest, and the mitigation strategies used across these control methods.

The reviewed literature shows that each paradigm offers distinct strengths and weaknesses when deployed in real-world environments, as shown in **Table (1)**. MPC provides strong stability and constraint handling while being interpretable and generally grounded, but it is highly sensitive to modelling inaccuracies and contact uncertainty. Reinforcement learning demonstrates strong adaptability and robust locomotion capabilities, especially on rough or unpredictable terrain, though successful transfer depends heavily on simulator fidelity, domain randomization, and reward design. Imitation learning is effective for generating natural and structured locomotion behaviours, but it struggles when there is a mismatch between simulation observations and real-world sensing or dynamics. The reviewed hybrid studies suggest that combining model-based control with learning can effectively address modelling errors, terrain uncertainty, and the sim-to-real gap. Common mitigation strategies include learned residual models, adaptive parameter tuning, and staged training that combines Model Predictive Control, Reinforcement Learning, and Imitation Learning. Overall, the literature indicates that hybrid approaches offer a promising direction for improving the robustness and transferability of quadruped locomotion across diverse real-world environments.

This study contributes to the field primarily by organizing quadruped locomotion sim-to-real transfer research into a structured taxonomy based on control paradigm, gap source, failure manifestation, and mitigation strategy. This framework allows clearer comparison between different approaches, including how simulation relates to real failure in each, how it manifests, and how it is reduced.

However, several limitations remain both in the literature and within this review. Many studies rely on limited hardware validation, simulator-specific experiments, or non-standardized evaluation methods.

This makes direct comparison between papers difficult. Additionally, differences in robot platforms, terrain setups, and performance metrics reduce consistency across studies. Some papers also lacked complete reporting of failure conditions or implementation details.

Future research should focus on developing standardized evaluation benchmarks for sim-to-real transfer, improving simulator realism and adaptive modelling techniques, and designing more efficient control systems, especially hybrid systems, to better integrate the complementary strengths of multiple control paradigms.

References:

[1]. Karaca, S., Tan, M., & Tan, Ü. (2013). Humans Walking on All Four Extremities With Mental Retardation and Dysarthric or no Speech: A Dynamical. *Developmental Disabilities: Molecules Involved, Diagnosis, and Clinical Care*, 81.

[2]. Jiang, Z. (2025). A Review of Recent Developments in Quadruped Robots. Theoretical and Natural Science, 79(1), 28–32. https://doi.org/10.54254/2753-8818/2025.19933

[3]. Abdulwahab, A. H., Mazlan, A. Z. A., Hawary, A. F., & Hadi, N. H. (2023). Quadruped robots mechanism, structural design, energy, gait, stability, and actuators: a review study. International Journal of Mechanical Engineering and Robotics Research, 12(6), 385-395.

[4]. Tan, J., Zhang, T., Coumans, E., Iscen, A., Bai, Y., Hafner, D., Bohez, S., Vanhoucke, V., Brain, G., & Deepmind, G. (2018). Sim-to-Real: Learning Agile Locomotion For Quadruped Robots. https://www.roboticsproceedings.org/rss14/p10.pdf

[5]. Zhang, H.; He, L.; Wang, D. Deep reinforcement learning for real-world quadrupedal locomotion: a comprehensive review. Intell. Robot. 2022, 2, 275-97. http://dx.doi.org/10.20517/ir.2022.20

[6]. Kim, D., & Park, J. (2024). Reduced Model Predictive Control Toward Highly Dynamic Quadruped Locomotion. IEEE Access, 12, 20003-20018. https://doi.org/10.1109/access.2024.3360479.

[7]. Pandala, A., Fawcett, R., Rosolia, U., Ames, A., & Hamed, K. (2022). Robust Predictive Control for Quadrupedal Locomotion: Learning to Close the Gap Between Reduced- and Full-Order Models. IEEE Robotics and Automation Letters, 7, 6622-6629. https://doi.org/10.1109/lra.2022.3176105.

[8]. Trivedi, A., Prajapati, S., Zolotas, M., Everett, M., & Padır, T. (2024). Chance-Constrained Convex MPC for Robust Quadruped Locomotion Under Parametric and Additive Uncertainties. IEEE Robotics and Automation Letters, 10, 8388-8395. https://doi.org/10.1109/lra.2025.3585315.

[9]. Elobaid, M., Turrisi, G., Rapetti, L., Romualdi, G., Dafarra, S., Kawakami, T., Chaki, T., Yoshiike, T., Semini, C., & Pucci, D. (2025). Adaptive Non-Linear Centroidal MPC With Stability Guarantees for Robust Locomotion of Legged Robots. IEEE Robotics and Automation Letters, 10, 2806-2813. https://doi.org/10.1109/lra.2025.3536296.

[10]. Bao, L., Peng, T., & Zhou, C. (2025). Sim-to-Real Transfer in Deep Reinforcement Learning for Bipedal Locomotion. ArXiv, abs/2511.06465. https://doi.org/10.48550/arxiv.2511.06465.

[11]. Kim, D., Lee, H., Cha, J., & Park, J. (2025). Bridging the Reality Gap: Analyzing Sim-to-Real Transfer Techniques for Reinforcement Learning in Humanoid Bipedal Locomotion. IEEE Robotics & Automation Magazine, 32, 49-58. https://doi.org/10.1109/mra.2024.3505784.

[12]. Hwangbo, J., Dosovitskiy, A., Bellicoso, D., Tsounis, V., Koltun, V., & Hutter, M. (2019). Learning agile and dynamic motor skills for legged robots. Science Robotics, 4. https://doi.org/10.1126/scirobotics.aau5872.

[13]. Hwangbo, J., Wellhausen, L., Koltun, V., & Hutter, M. (2020). Learning quadrupedal locomotion over challenging terrain. Science Robotics, 5. https://doi.org/10.1126/scirobotics.abc5986.

[14]. Bellegarda, G., & Ijspeert, A. (2022). CPG-RL: Learning Central Pattern Generators for Quadruped Locomotion. IEEE Robotics and Automation Letters, 7, 12547-12554. https://doi.org/10.1109/lra.2022.3218167.

[15]. Lai, H., Zhang, W., He, X., Yu, C., Tian, Z., Yu, Y., & Wang, J. (2022). Sim-to-Real Transfer for Quadrupedal Locomotion via Terrain Transformer. 2023 IEEE International Conference on Robotics and Automation (ICRA), 5141-5147. https://doi.org/10.1109/icra48891.2023.10160497.

[16]. Gangapurwala, S., Geisert, M., Orsolino, R., Fallon, M., & Havoutis, I. (2020). RLOC: Terrain-Aware Legged Locomotion Using Reinforcement Learning and Optimal Control. IEEE Transactions on Robotics, 38, 2908-2927. https://doi.org/10.1109/tro.2022.3172469.

[17]. Wang, H., Luo, H., Zhang, W., & Chen, H. (2024). CTS: Concurrent Teacher-Student Reinforcement Learning for Legged Locomotion. IEEE Robotics and Automation Letters, 9, 9191-9198. https://doi.org/10.1109/lra.2024.3457379.

[18]. Dowdy, J., & Vaz, J. (2025). Isaac Sim-to-Real: Reinforcement Learning based Locomotion for Quadrupeds. 2025 IEEE 21st International Conference on Automation Science and Engineering (CASE), 2194-2199. https://doi.org/10.1109/case58245.2025.11163761.

[19]. Liu, D., Zhang, T., Yin, J., & See, S. (2022). Saving the Limping: Fault-tolerant Quadruped Locomotion via Reinforcement Learning. ArXiv, abs/2210.00474. https://doi.org/10.48550/arxiv.2210.00474.

[20]. Rudin, N., Kolvenbach, H., Tsounis, V., & Hutter, M. (2021). Cat-Like Jumping and Landing of Legged Robots in Low Gravity Using Deep Reinforcement Learning. IEEE Transactions on Robotics, 38, 317-328. https://doi.org/10.1109/tro.2021.3084374.

[21]. Peng, X., Coumans, E., Zhang, T., Lee, T., Tan, J., & Levine, S. (2020). Learning Agile Robotic Locomotion Skills by Imitating Animals. ArXiv, abs/2004.00784. https://doi.org/10.15607/rss.2020.xvi.064.

[22]. Bohez, S., Tunyasuvunakool, S., Brakel, P., Sadeghi, F., Hasenclever, L., Tassa, Y., Parisotto, E., Humplik, J., Haarnoja, T., Hafner, R., Wulfmeier, M., Neunert, M., Moran, B., Siegel, N., Huber, A., Romano, F., Batchelor, N., Casarini, F., Merel, J., Hadsell, R., & Heess, N. (2022). Imitate and Repurpose: Learning Reusable Robot Movement Skills From Human and Animal Behaviors. ArXiv, abs/2203.17138. https://doi.org/10.48550/arxiv.2203.17138.

[23]. Li, T., Zhang, Y., Zhang, C., Zhu, Q., Sheng, J., Chi, W., Zhou, C., & Han, L. (2023). Learning Terrain-Adaptive Locomotion with Agile Behaviors by Imitating Animals. 2023 IEEE/RSJ International

Conference on Intelligent Robots and Systems (IROS), 339-345. https://doi.org/10.1109/iros55552.2023.10342271.

[24]. Fuchioka, Y., Xie, Z., & Panne, M. (2022). OPT-Mimic: Imitation of Optimized Trajectories for Dynamic Quadruped Behaviors. 2023 IEEE International Conference on Robotics and Automation (ICRA), 5092-5098. https://doi.org/10.1109/icra48891.2023.10160562.

[25]. Yazdi, M., Pourghavam, M., & Bidari, R. (2024). Hybrid Control of Advanced Quadruped Locomotion: Integrating Model Predictive Control with Deep Reinforcement Learning. 2024 12th RSI International Conference on Robotics and Mechatronics (ICRoM), 201-208. https://doi.org/10.1109/icrom64545.2024.10903592.

[26]. Zhang, Z., Chang, X., Ma, H., An, H., & Lang, L. (2023). Model Predictive Control of Quadruped Robot Based on Reinforcement Learning. *Applied Sciences (2076-3417)*, *13*(1).

[27]. Zhou, H., Zhang, X., & Tzoumas, V. (2025). Adaptive Legged Locomotion via Online Learning for Model Predictive Control. IEEE Robotics and Automation Letters, 11, 1778-1785. https://doi.org/10.1109/lra.2025.3644161.

[28]. Zhang, Z., An, H., Wei, Q., & Ma, H. (2022, December). Learning-based model predictive control for quadruped locomotion on slippery ground. In *2022 4th International Conference on Control and Robotics (ICCR)* (pp. 47-52). IEEE.

[29]. Youm, D., Jung, H., Kim, H., Hwangbo, J., Park, H., & Ha, S. (2023). Imitating and Finetuning Model Predictive Control for Robust and Symmetric Quadrupedal Locomotion. IEEE Robotics and Automation Letters, 8, 7799-7806. https://doi.org/10.1109/lra.2023.3320827.

[30]. Kovalev, V., Shkromada, A., Ouerdane, H., & Osinenko, P. (2023). Combining model-predictive control and predictive reinforcement learning for stable quadrupedal robot locomotion. ArXiv, abs/2307.07752. https://doi.org/10.48550/arxiv.2307.07752.

[31]. Bagajo, J., Schwarke, C., Klemm, V., Georgiev, I., Sleiman, J. P., Tordesillas, J., ... & Hutter, M. (2024). Diffsim2real: Deploying quadrupedal locomotion policies purely trained in differentiable simulation. arXiv preprint arXiv:2411.02189.

[32]. Bellegarda, G., Chen, Y., Liu, Z., & Nguyen, Q. (2022, October). Robust high-speed running for quadruped robots via deep reinforcement learning. In 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (pp. 10364-10370). IEEE.

[33]. Yang, R., Zhang, M., Hansen, N., Xu, H., & Wang, X. (2021). Learning vision-guided quadrupedal locomotion end-to-end with cross-modal transformers. arXiv preprint arXiv:2107.03996

[34]. Jin, Y., Liu, X., Shao, Y., Wang, H., & Yang, W. (2022). High-speed quadrupedal locomotion by imitation-relaxation reinforcement learning. Nature Machine Intelligence, 4(12), 1198-1208.

[35]. Xie, Z., Da, X., Van de Panne, M., Babich, B., & Garg, A. (2021, May). Dynamics randomization revisited: A case study for quadrupedal locomotion. In 2021 IEEE International Conference on Robotics and Automation (ICRA) (pp. 4955-4961). IEEE.

[36]. Lee, M., Kim, H., Kim, J. T., Park, S., Lee, J. H., Cho, J., & Hwangbo, J. (2025). Learning Quadrupedal Locomotion for a Heavy Hydraulic Robot Using an Actuator Model. IEEE Robotics and Automation Letters.

[37]. Zhang, H., Yu, H., Zhao, L., Choi, A., Bai, Q., Yang, Y., & Xu, W. (2025). Learning Multi-Stage Pick-and-Place With a Legged Mobile Manipulator. IEEE Robotics and Automation Letters.

[38]. Zhang, H., Yu, H., Zhao, L., Choi, A., Bai, Q., Yang, B., & Xu, W. (2025). Slim: Sim-to-real legged instructive manipulation via long-horizon visuomotor learning. arXiv preprint arXiv:2501.09905.

[39]. Ma, S., others, & DrEureka Team. (2024). DrEureka: Language model guided sim-to-real transfer [Preprint]. arXiv. https://arxiv.org/abs/2406.01967

[40]. Zeng, X., Zhang, H., Yue, L., Song, Z., Zhang, L., & Liu, Y. (2024). Adaptive Model Predictive Control with Data-driven Error Model for Quadrupedal Locomotion. *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 5731-5737. https://doi.org/10.1109/icra57147.2024.10611302

[41]. Amanzadeh, L., Chunawala, T., Fawcett, R. T., Leonessa, A., & Hamed, K. (2024). Predictive Control With Indirect Adaptive Laws for Payload Transportation by Quadrupedal Robots. *IEEE Robotics and Automation Letters, 9*, 10359-10366. https://doi.org/10.1109/lra.2024.3474550

[42]. Xu, S., Zhu, L., Zhang, H.-T., & Ho, C. (2023). Robust Convex Model Predictive Control for Quadruped Locomotion Under Uncertainties. IEEE Transactions on Robotics, 39, 4837-4854. https://doi.org/10.1109/tro.2023.3299527

Appendix A

Full Study Extraction Table

This appendix presents the complete per-study extraction underlying the thematic synthesis reported in the Results section, grouped by control paradigm. Table 1 in the main text summarizes these findings at the paradigm level.

Table (2): Full taxonomy table for MPC,RL,IL, and Hybrid approaches

Model Predictive Control (MPC) Studies

Paper Name	Failure Source (Sim-to-Real)	Failure Manifestation	Mitigation Strategy
Adaptive Legged Locomotion via Online Learning for Model Predictive Control [27]	Modeling errors and unknown real-world uncertainties (unknown payloads, varying friction, external forces) cause mismatch between nominal dynamics and real robot	Potential tracking degradation under large unknown external forces (up to 12g), slopes, rough terrain, time-varying payloads (up to 8 kg), and changing friction	Combine MPC with online learned residual dynamics (random Fourier features + least-squares updates during control) to adapt model and maintain trajectory tracking under these real-world disturbances
Adaptive Model Predictive Control with Data-driven Error Model for Quadrupedal Locomotion [40]	Model mismatch caused by simplified and linearized SRB dynamics; inaccurate model parameters (especially mass); unmodeled payloads; neglected leg dynamics; dynamic uncertainties between the reduced-order MPC model and the real robot	Large body-height vibration; reduced state-tracking accuracy; incorrect ground reaction force (GRF) predictions; instability during trotting; inability of baseline MPC to maintain desired body height under unknown payloads, leading to robot collapse in simulation and	Integrates a data-driven ARMAV error model with MPC to predict future state errors from sensor data and compensate for model inaccuracies; multi-step prediction corrects MPC state estimates while one-step prediction compensates Whole-Body Control

Paper Name	Failure Source (Sim-to-Real)	Failure Manifestation	Mitigation Strategy
		degraded hardware performance	(WBC); validated on hardware with up to 20 kg unmodeled payload
Predictive Control with Indirect Adaptive Laws for Payload Transportation by Quadrupedal Robots [41]	Unknown and varying payloads; parametric uncertainty (particularly robot mass); model mismatch between the reduced-order SRB model and the real robot; rough terrain disturbances	Nominal MPC cannot maintain stable locomotion under heavy payloads, exhibiting degraded trajectory tracking, reduced body-height regulation, lower success rates on rough terrain, and complete failure with large unknown payloads	Integrates an indirect adaptive estimation law with a convex MPC to estimate unknown mass online; embeds stability conditions within the MPC optimization; combines the adaptive MPC planner with a low-level nonlinear Whole-Body Controller (WBC) for trajectory tracking
Robust Predictive Control for Quadrupedal Locomotion: Learning to Close the Gap between Reduced- and Full-Order Models [42]	Reduced-order model abstraction; unmodeled dynamics; mismatch between reduced- and full-order models; payload uncertainty; terrain variability (slippery, compliant, gravel, slopes); contact force prediction errors	Reduced tracking accuracy; inaccurate ground reaction forces (GRFs); instability on rough/slippery terrain; failure of nominal MPC on uneven terrain; degraded locomotion under payloads and environmental uncertainty	Hierarchical Robust MPC (RMPC) with RL-trained neural network (PPO) to learn uncertainty sets; learned uncertainty compensation inside MPC; low-level QP whole-body controller; robust optimization; online GRF planning; experimental validation across diverse terrains and payloads
Quadrupedal Locomotion via Event-Based Predictive Control and QP-Based Virtual Constraints [43]	Terrain uncertainty; unpredictable foothold timing; model mismatch arising from varying contact events and ground height changes; inaccuracies in event	Reduced gait stability on uneven terrain; inaccurate foot placement; degraded locomotion performance when contact events deviate from the nominal model; sensitivity to disturbances	Introduces event-based predictive control that replans only at contact events rather than fixed time intervals; combines QP-based virtual constraints with predictive control to

Paper Name	Failure Source (Sim-to-Real)	Failure Manifestation	Mitigation Strategy
	timing during dynamic locomotion	during dynamic gait transitions	stabilize gait generation; whole-body optimization enforces dynamic consistency while adapting to changing contact conditions

Reinforcement Learning (RL) Studies

Paper Name	Failure Source (Sim-to-Real)	Failure Manifestation	Mitigation Strategy
Sim-to-Real: Learning Agile Locomotion For Quadruped Robots [4]	Simulator ≠ real robot: inaccurate dynamics, actuator behavior, latency	Policies trained in sim often fail when deployed on hardware	Improve simulator via system ID, accurate actuator model, latency simulation; learn robust policies with domain randomization, perturbations, compact observation space
CPG-RL: Learning Central Pattern Generators for Quadruped Locomotion [14]	Potential modeling mismatch; no explicit domain randomization	Disturbances and added payload (115% mass) could destabilize naive policies	RL modulates CPG parameters; careful proprioceptive observation design; shows robust sim-to-real without domain randomization
Robust High-Speed Running for Quadruped Robots via Deep RL [32]	Model uncertainty, rough terrain differences between sim and real	High-speed running and loads can cause failures/instability if controller not robust	Train in PyBullet with environmental noise (model uncertainty, rough terrain) → successful sim-to-sim and sim-to-real transfer to Unitree A1

Paper Name	Failure Source (Sim-to-Real)	Failure Manifestation	Mitigation Strategy
Isaac Sim-to-Real: RL based Locomotion for Quadrupeds [18]	Sim-to-real gap for whole-body RL controllers	Policies that work in sim fail or degrade on Unitree Go1	Use high-fidelity Isaac Sim/Isaac Lab, train robust RL policy achieving zero-shot sim-to-real; improved disturbance recovery vs built-in controller
Towards Fault-tolerant Quadruped Locomotion with Reinforcement Learning [19]	Critical hardware joint failures rarely handled in standard RL locomotion	Immediate falling threat during joint failures	Teacher-student RL with tailored training strategy → fault-tolerant controller with zero-shot deployment, no fine-tuning
Sim-to-Real Transfer for Quadrupedal Locomotion via Terrain Transformer (TERT) [15]	Reality gap from changing terrain dynamics and limited capacity of conventional networks	Baseline policies cannot traverse challenging terrains like sandpit, stair down on hardware	High-capacity Transformer policy + 2-stage training (offline pretrain, online correction with privileged info); domain randomization/system ID compatible; succeeds on 9 terrains
Learning Quadrupedal Locomotion over Challenging Terrain [13]	Unmodeled factors (mud, snow, vegetation, rubble, water), contact estimation errors	Classical controllers fail in such natural terrains; naive RL policies overfit sim	Proprioceptive-only model-free RL with large-scale training + domain randomization, achieving zero-shot generalization to many outdoor terrains
Learning Vision-Guided Quadrupedal Locomotion End-to-End with Cross-Modal Transformers (LocoTransformer) [33]	Blind RL with only proprioception + domain randomization insufficient for complex obstacles	Poor generalization and failures on unseen obstacles/terrains when transferred	End-to-end RL with Transformer combining vision + proprioception; better sim-to-real generalization indoors and in the wild

Paper Name	Failure Source (Sim-to-Real)	Failure Manifestation	Mitigation Strategy
Dynamics Randomization Revisited [35]	Mismatch between sim dynamics and real robot; uncertainty about role of randomization and design choices	Potential transfer failure or degraded performance (studied via ablations)	Careful selection of design elements (e.g., dynamics randomization, state history, noise) to improve robustness; show direct transfer can work without heavy randomization for Laikago
Learning Quadrupedal Locomotion for a Heavy Hydraulic Robot Using an Actuator Model [36]	Complex hydraulic dynamics; slow control response make detailed sim unsuitable for RL	Standard simulators cannot accurately capture 12 hydraulic actuators, policies would not transfer	Fast analytical actuator model driven by hydraulic dynamics; predicts joint torques accurately enough for RL, enabling stable, robust transfer on 300-kg robot
Cat-Like Jumping and Landing of Legged Robots in Low Gravity [20]	Differences between simulated and real low-gravity dynamics for long flight phases	Risk of unstable attitude/landing when deploying on testbed	Domain randomization and RL training over tasks with varied conditions; validated on microgravity testbed to ensure stable jumping/landing
DiffSim2Real [31]	Differentiable sims usually have poor physical accuracy; inaccurate contact models	Policies from differentiable sim typically fail on real robot	Smooth but physically accurate contact model; domain randomization; PD controller with system-ID parameters and velocity-based torque saturation instead of complex learned actuator model

Paper Name	Failure Source (Sim-to-Real)	Failure Manifestation	Mitigation Strategy
Learning Multi-Stage Pick-and-Place With a Legged Mobile Manipulator [37]	Dynamics gap: mismatched object/motor properties, unmodeled friction/backlash	Failures in precise tasks: e.g., kicking cube away, shaking so cube is out of reach, early/late dropping, basket kicked away, visual occlusion of object	PID arm control; noise on arm joint targets; random perturbation of arm mount pose; object perturbations during training to enforce reactive, robust policies
SLIM: Sim-to-Real Legged Instructive Manipulation [38]	Dynamics gap (objects, motors, friction, backlash) causing sim-trained policies to fail in real, especially for accurate motor control like grasping small objects	Degraded real performance vs sim on long-horizon manipulation/locomotion	Suite of sim-to-real techniques: dynamics randomization, visual augmentation, teacher-student with privileged decomposition, etc., rated by importance for closing both dynamics and vision gaps

Imitation Learning (IL) Studies

Paper Name	Failure Source (Sim-to-Real)	Failure Manifestation	Mitigation Strategy
Learning Agile Robotic Locomotion Skills by Imitating Animals [20]	Need for domain adaptation from sim to real due to differences between simulated and real robot/environment dynamics	Policies trained in sim may not directly transfer to hardware without adaptation	Use imitation learning from animal motion plus sample-efficient domain adaptation techniques so policies trained in simulation can be quickly adapted to the real robot
Imitate and Repurpose: Learning Reusable Robot Movement Skills From Human and Animal Behaviours [22]	Need accurate actuator modeling for ANYmal's series elastic actuators; basic rigid-body sim is insufficient for realistic joint behaviour and efficiency	If actuators are mis-modeled, locomotion skills learned by imitation in sim would not match real ANYmal torques/currents; could hurt transfer and energy usage	Train a dedicated actuator network (analytic PID + learned low-level torque model) using real data to match SEA behaviour, including current draw; use this higher-fidelity actuator model during IL and reuse

Paper Name	Failure Source (Sim-to-Real)	Failure Manifestation	Mitigation Strategy
			to improve sim-to-real transfer and efficiency
Learning Terrain-Adaptive Locomotion with Agile Behaviors by Imitating Animals [23]	IL step trains only on limited MoCap terrains (fixed-size slopes/stairs); when real terrain configuration differs, naive tracking fails	Example: stair-climbing reference motion with 3 steps will likely hit the stairs when more steps exist in reality; IL policy cannot generalize and cannot be user-controlled	Add a terrain adaptation module and second training step: learn to generalize over randomized terrain parameters and commands; compare against 'without IL step' and 'without terrain adaptation' to show improved robustness across unseen terrains

Hybrid Approaches(RL-IL-MPC)

Paper Name	Failure Source (Sim-to-Real)	Failure Manifestation	Mitigation Strategy
Imitating and Finetuning Model Predictive Control for Robust and Symmetric Quadrupedal Locomotion [29]	Conventional MPC alone degrades on rough, slippery, conveyor terrains due to model/terrain mismatch and limited robustness	Poor robustness and gait quality (less symmetric, periodic, energy-efficient) on challenging real and simulated terrains	IFM framework: (i) design MPC expert, (ii) clone via imitation learning, (iii) finetune with deep RL (limited exploration) on harder terrains to improve robustness, symmetry, and efficiency over vanilla RL and plain MPC
Robust Predictive Control for Quadrupedal Locomotion: Learning to Close the Gap Between Reduced- and Full-Order Models [28]	Abstraction/unmodeled dynamics in reduced-order (single rigid body) MPC model create a gap to full-order robot dynamics, harming transfer to hardware	Reduced-order plans that are not robust when executed on full-order robot, especially on rough terrain	Robust MPC + DRL: learn unmodeled dynamics with deep RL and inject them into RMPC; map RMPC ground-reaction-force plans to full-order via nonlinear controller, validated in sim and

Paper Name	Failure Source (Sim-to-Real)	Failure Manifestation	Mitigation Strategy
			experiments for robust blind locomotion on different terrains
High-Speed Quadrupedal Locomotion by Imitation-Relaxation Reinforcement Learning [34]	Gap between simulation and reality when optimizing multiple objectives at high speed; risk of period-doubling bifurcation and chaos in real dynamics	Loss of periodic gait and possible chaotic/unstable running behavior at high speed (captured via entropy metric)	Introduce stochastic stability analysis using state-space entropy decreasing rate to detect bifurcations; use staged IRRL training plus this analysis to obtain stable 5.0 m/s running on hardware
OPT-Mimic: Imitation of Optimized Trajectories for Dynamic Quadruped Behaviors [24]	Limited observations on the real robot vs richer simulator state; need for policies that transfer without external motion capture; overfitting to sim if rewards don't regularize for transfer	Potential mismatch between sim-trained policy (using full state) and real robot (limited sensors), which would break execution after transfer	Use imitation-based RL with residual joint-angle actions on top of optimized reference motions; restrict observations to quantities available on hardware.

Note. MPC = Model Predictive Control; RL = Reinforcement Learning; IL = Imitation Learning.