Proceedings of Machine Learning Research vol vvv:1–14, 2025

Bridging Adaptivity and Safety: Agile Legged Locomotion Across Varied Physics

Yichao Zhong YICHAOZ@ANDREW.CMU.EDU
Carnegie Mellon University

Chong Zhang CHOZHANG@ETHZ.CH
ETH Zurich

Tairan He TAIRANH@ANDREW.CMU.EDU
Carnegie Mellon University

Guanya Shi GUANYAS@ANDREW.CMU.EDU
Carnegie Mellon University

Abstract

Real-world legged locomotion systems often need to reconcile agility and safety for different scenarios. Moreover, these systems face challenges as the real-world dynamics are varied and changing (e.g., payload, friction). In this paper, we introduce BAS (**B**ridging **A**daptivity and **S**afety), which builds upon the pipeline of prior work ABS(Agile But Safe) (He et al., 2024) and is designed to ensure adaptive safety even in demanding environments while maintaining agility. To this end, we make the whole system adaptive to varying physics by conditioning modules on physical parameters, and train a physics estimator concurrently with the controller. To mitigate the distribution shift of physical parameter estimation during deployment, we further introduce on-policy fine-tuning on the estimator to enhance the estimation accuracy. The simulation results show that BAS achieves 50% better safety in demanding environments while maintaining a higher speed on average. In real-world experiments, BAS shows its capability to handle complex environments (e.g. slippery floor, carrying up to 8kg payload), while ABS and other baselines lack adaptivity, leading to collisions or compromised in agility.

Keywords: Reinforcement Learning, Adaptive Safe Control, Legged Locomotion

1. INTRODUCTION

Legged robot locomotion in cluttered and dynamic environments requires adaptivity to varying physics and environmental changes while simultaneously ensuring agility (for efficient navigation) and safety (for reliable deployment). This adaptivity is crucial for real-world tasks such as disaster response in forests (Sun et al., 2020), evacuation in fire-prone areas (Panahi et al., 2023), and rescue operations (Arabboev et al., 2021). Despite recent progress in legged locomotion (Brunke et al., 2022; Hwangbo et al., 2019; Kumar et al., 2021; Lee et al., 2020; Li et al., 2024), there remains a significant gap in methodologies that effectively integrate adaptivity, safety, and agility. In this work, we enable the robot to jointly achieve agility and safety with adaptivity simultaneously, maintaining strong performance in challenging environments.

Striking a good balance of adaptivity, safety, and agility in legged locomotion remains a significant challenge, as focusing on one aspect often comes at the expense of the others. Recent advances in legged/wheeled locomotion work use reinforcement learning (RL) (Levine et al., 2020; Silver et al., 2017), prioritizing adaptability in agility to handle environmental changes (Kumar et al., 2021; Lee et al., 2020; Wang et al., 2024; Long et al., 2024; Zhang et al., 2024; Yang et al., 2023; Luo et al., 2024). However, these approaches somehow neglect safety considerations. ABS (He et al.,

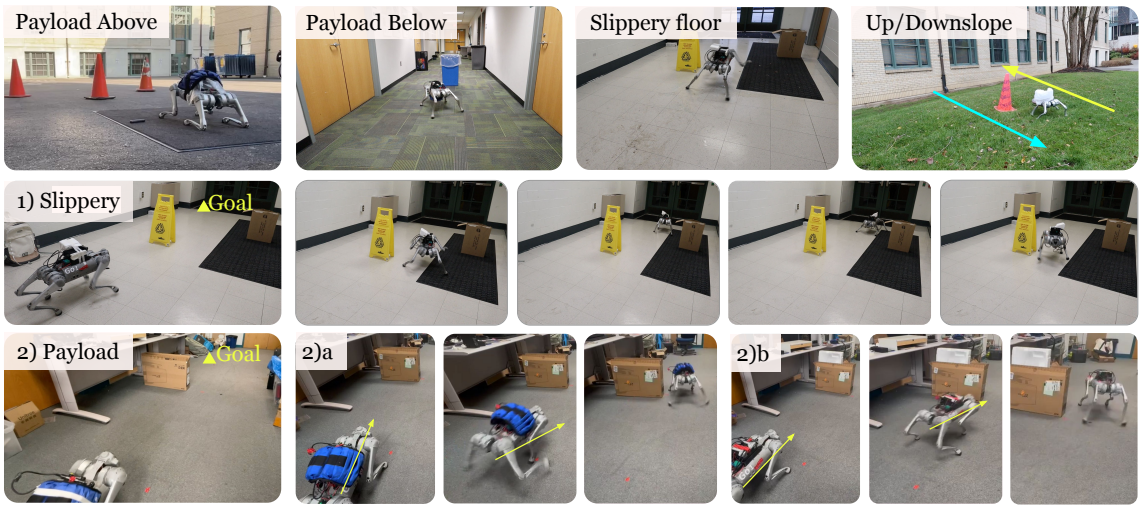


Figure 1: 1) The robot can handle collision-free locomotion in even super slippery terrain condition (soap water on both floor and robot feet), and also can adapt to rough terrain (dry carpet) suddenly. 2) Adaptive recovery triggering of the robot in different circumstances as early recovery with 8kg payload in a) and late recovery in b) with no payload.

2024) has pushed the joint limit of being agile and safe in nominal environments but is not adaptive to varying physics, and agility and safety performances can drop severely when the environment changes, as shown in our experimental results.

Other studies (Xiao et al., 2024; Chiu et al., 2022; Yun et al., 2024; Borquez et al., 2023; Gao et al., 2024) focus on adaptivity and safety, but sacrifice agility or need precise dynamics in the real world. For example, Yun et al. (2024) solves safe legged locomotion from a model-based control perspective but sacrifices much on speed. On the theory side, Borquez et al. (2023) proves that it's possible to achieve adaptive safety by parameter-conditioned reachability analysis, but the ground truth physical parameters are not accessible in the real world, making it challenging to be deployed in real world.

In addition to being adaptive, another way to handle changing environments is to improve the robustness by making the system more conservative (Buchanan et al., 2021; Kim et al., 2020). However, being conservative can be insufficient in certain scenarios (e.g., search and rescue tasks). Moreover, although the safety-related literature is rich (Achiam et al., 2017; Bansal et al., 2017; Liu et al., 2022; Xu et al., 2021; Margellos and Lygeros, 2011), most of them are not tested in the real world, and Hsu et al. (2023) shows the gap between mathematical theory and real-world uncertainty can be significant. In summary, there is a missing space for adaptive, safe, and agile locomotion for the needs of real-world applications.

To address this, we propose BAS, which builds on ABS (He et al., 2024) by introducing adaptivity to achieve a comprehensive balance of agility, safety, and environmental adaptability. ABS involves an agile policy to avoid obstacles rapidly and a recovery policy to prevent failures, and a learned control-theoretic reach-avoid value network which governs the policy switch and guides the recovery policy as an objective function and safeguards the robot in a closed loop. Yet, different from ABS, BAS employs an explicit physics-parameter estimator learned from proprioceptive

history during policy training and feeds forward the estimated parameters to the controller and the RA(Reach-Avoid) network to enhance the adaptivity. To mitigate the distribution shift caused by switching between agile and recovery policies, we further introduce an end-to-end on-policy fine-tuning strategy, improving the accuracy of the estimator during inference. Extensive evaluations demonstrate that BAS significantly outperforms ABS and other adaptive-and-safe baselines in both safety and agility metrics. In real-world experiments, BAS achieves a 19.8% increase in speed and is 2.36× lower in collision rate than ABS among diverse and challenging environments. We also have impressive videos of how BAS works in real on our website: https://adaptive-safe-locomotion.github.io/

Briefly, we identify our contributions as follows:

1. We propose an adaptive safety framework that incorporates physics parameter estimation.
2. We propose on-policy fine-tuning to enhance the robustness of the explicit estimator in dynamic environments.
3. We validate the adaptivity, safety, and agility of BAS through extensive evaluations in both simulation and real-world scenarios.
4. Theoretically, we extend the reach-avoid analysis in Section 2, providing provable insights that support our approach and its practical applicability.

2. Preliminaries and Problem Formulation

Dynamics The dynamics is defined by state $s \in \mathcal{S} \in \mathbb{R}^{|s|}$ and action $a \in \mathcal{A} \in \mathbb{R}^{|a|}$ and environmental physical parameters $e \in \mathcal{E} \in \mathbb{R}^{|e|}$:

$$s_{t+1} = s_t + f(s_t, a_t, e_t) \tag{1}$$

For simplicity, in this paper, we discuss the case where the dynamics can be affected by the privileged environmental factors e, i.e., the mass of payload, the friction coefficient, the CoM shift, etc., which is assumed quasi-static within a trajectory in training sessions. The observations are from proprioceptive and exteroceptive sensors, denoted as $o = h(s)$.

Goal Settings Given local position and goals $G \in \Gamma$, we learn a goal-reaching policy $\pi : \mathcal{O} \times \Gamma \to \mathcal{A}$ to maximize the expected return:$J(\pi) = \mathbb{E}_{\pi,G}\left[\sum_{t=0}^{\infty} \gamma_{RL}^t r(s_t, e_t, a_t, G)\right]$, where $r(\cdot)$ is the reward at time t and γ_{RL} is the discount factor.

Safety Settings First, we denote the system trajectory starting from state $s' = (s, e)$ while using control inputs from the policy π as $\xi_{s,e}^{\pi}(\cdot) : \mathbb{R} \to \mathcal{S} \times \mathcal{E}$. As in Bansal et al. (2017), we define several basic sets: The target set $\mathcal{T} \in \mathcal{S}$ which represents the area of the goal, the constraint set $\mathcal{K} \in \mathcal{S}$ which refers to the traversable areas for robots. and the failure set $\mathcal{F} = \mathcal{K}^C$ which is the complement of the constraint set and represents hazardous areas like obstacles.

Based on the basic sets, we can define the following sets in the context of the reachability theory. The safe set is defined as the set of states from which the robot can start and has a positive probability of rolling out a trajectory without failure, expressed as: $\omega^{\pi}(\mathcal{F}) := \{(s, e) \in \mathcal{S}' \mid \forall \tau \geq 0, \xi_{s,e}^{\pi}(\tau) \notin \mathcal{F}\}$. The backward reachable set is the collection of states from which the robot has a positive probability of reaching the target: $\mathcal{R}^{\pi}(\mathcal{T}) := \{(s, e) \in \mathcal{S}' \mid \exists \tau \geq 0, \xi_{s,e}^{\pi}(\tau) \in \mathcal{T}\}$. And the reach-avoid set combines the safe set and the backward reachable set: $\mathcal{RA}^{\pi}(\mathcal{T}, \mathcal{F}) := \{(s, e) \in \mathcal{S}' \mid \exists \tau \geq 0, \xi_{s,e}^{\pi}(\tau) \in \mathcal{T} \wedge \forall \tau \geq 0, \xi_{s,e}^{\pi}(\tau) \notin \mathcal{F}\}$, which represents states from which the robot can reach the target while avoiding failure.

Reach-Avoid Value and time-Discounted Reach-Avoid Bellman Equation (DRABE) Identical to the vanilla reach-avoid analysis (Bansal et al., 2017), we define two Lipschitz-continuous functions $l(\cdot), \zeta(\cdot) : \mathcal{S} \to \mathcal{R}$ which satisfy $\begin{cases} l(s) \leq 0 \iff s \in \mathcal{T} \\ \zeta(s) > 0 \iff s \in \mathcal{F} \end{cases}$ to illustrate if it has reached without failure. Note that this function is only dependent on the state s and is environment-agnostic. Then we define the reach-avoid value function which satisfies $V_{RA}^{\pi}(s,e) \leq 0 \iff s \in \mathcal{RA}^{\pi}(\mathcal{T},\mathcal{F})$:

$$V_{RA}^{\pi}(s,e) = \min_{\tau \in \{0,1,\dots\}} \max\left\{l(\xi_{s,e}^{\pi}(\tau), \max_{\kappa \in \{0,1,\dots,\tau\}} \zeta(\xi_{s,e}^{\pi}(\kappa)))\right\} \tag{2}$$

And as introduced in Hsu* et al. (2021), Discounted Reach-Avoid Bellman Equation (DRABE) could make the value function iteration a contraction mapping, which guarantees the convergence of the iteration to optimal:

$$B_{\gamma}[V_{RA_{\gamma}}^{\pi}](s,e) = (1-\gamma)\max\left\{l(s),\zeta(s)\right\} + \gamma \max\{\min\{V_{RA_{\gamma}}^{\pi}(s_{+},e_{+}), l(s)\}, \zeta(s)\} \tag{3}$$

where $s_{+} = s + f(s,\pi(s),e), e_{+} = e$ with subscript '+' denoting 'next step'. For e we use the quasi-static assumption here.

Convergence Guarantees for parameter-conditioned DRABE For Theorem 1, we prove that it guarantees that a value function conditioned on physical parameters, denoted as $V_{\gamma}^{\pi}(s,e)$, converges to optimal through eq. (3) iterations. A comprehensive proof of Theorem 1 can be found in the supplementary materials at our website.

Theorem 1 *Contraction Mapping of Parameter-conditioned DRABE. With the DRABE operator $B_{\gamma}[\cdot]$ defined in Equation (3), for any state and quasi-static physical parameter e, we have $\|B_{\gamma}[V_{\gamma}^{\pi}(s,e)] - V^{*}(s,e)]\| < \|V_{\gamma}^{\pi}(s,e) - V^{*}(s,e)\|$, i.e. DRABE is a contraction mapping.*

3. METHODOLOGIES

In this section, we present our proposed framework as shown in Figure 2 which has four training phases:(Section 3.1) training parameter estimator for adaptation; (Section 3.2) training RA network; (Section 3.3) on-policy fine-tuning estimator to address the history distribution shift.

3.1. Phase 1: Joint-train Agile Policy and Physical Parameter Estimator

Policy-conditioned Physical Parameter Estimator Since the environmental factors are often inaccessible in the real world, we tackle this challenge by learning an estimator of physical parameters conditioned on agile policy $\phi^{\pi_{agile}}(o_{t:t-49})$ from robot proprioception history. During the training pipeline, the estimations are fed back to agile policy, making it a concurrent estimator. However, training a general state estimator with high accuracy is super challenging, so we first opt to train a policy-conditioned estimator to lower the challenges, and also propose the fusion interpolation in the joint-train pipeline to further boost the accuracy.

The estimator $\phi^{\pi_{agile}}$ explicitly estimates the mass of payload m, position shift of CoM $\Delta x_c, \Delta y_c, \Delta z_c$, and friction coefficient μ, which are critical to daily usage of autonomous robots.

Additionally, note that physical parameters are policy-invariant variables. So compared to predicting dynamics which tangles with policies, predicting physical parameters is more suitable for cases where multiple policies are used together like He et al. (2024); Hoeller et al. (2023). What's

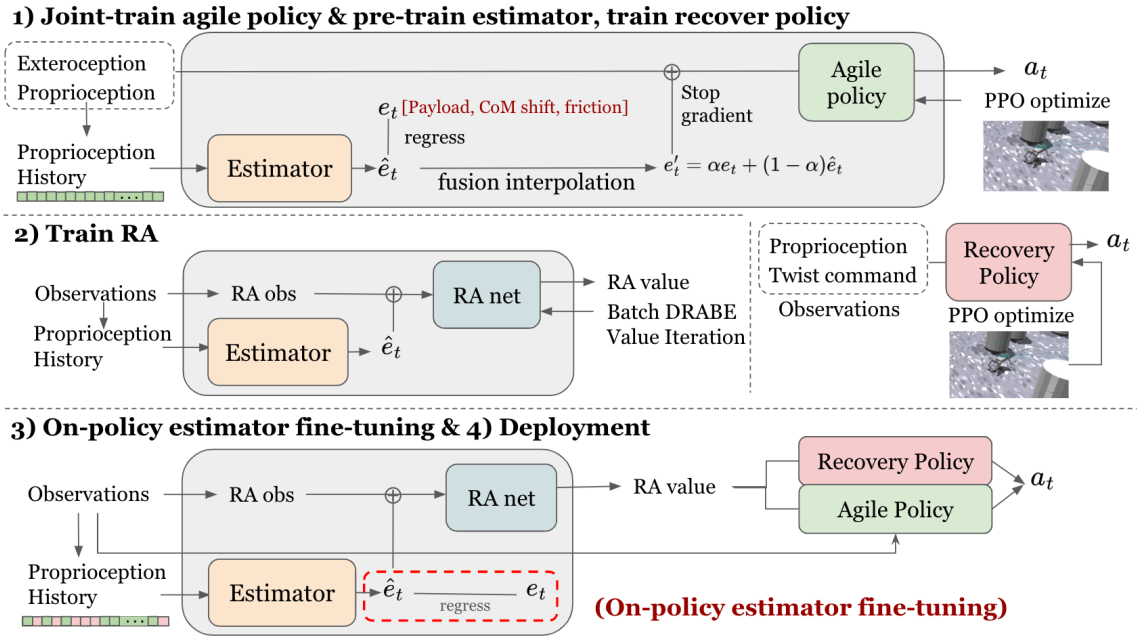


Figure 2: BAS Pipeline Overview.

more, estimating policy-invariant physical parameters partly lessens the potential issue of the history pattern misalignment when changing policies. We also do on-policy fine-tuning as described in Section 3.3 to diminish this misalignment further.

Policy training Following ABS, we keep the two-policy-switching structure: agile and recovery policy. The agile policy is a goal-reaching policy and takes control the most of time, and we retain the recovery policy designs from ABS, which tracks a given twist command. To make the agile policy adaptive, we add the estimated physical parameters to its observation spaces: $a(t) = \pi^{Agile}(o^{Agile}(t), \hat{e}(t))$ where $\hat{e}(t)$ is the estimated physical parameters at timestep t.

Training pipeline To ensure that the policy and the estimator co-work well, we propose a fusion on parameters co-train scheme as described in Figure 2, where the policy and the estimator is concurrently trained as inspired by Ji et al. (2022). In this fusion mechanism, for the beginning steps, we want the policy to converge fast with the aid of ground truth privileged observations, and when the policy converges, we want the policy and estimator to co-adapt to each other's distribution. Furthermore, to train a more accurate estimator, we employ an MSE loss with L2 regularization. We also did an ablation analysis in Section 4.2 to validate this joint-training pipeline and fusion interpolation.

3.2. Phase 2: Learning adaptive Reach-Avoid Network

As in ABS, the RA network learns a Reach-avoid value function as the DRABE described in Section 2. To boost the RA network's adaptivity, we extend the RA observation space with the estimated physical parameters to make it environment-agnostic. We learn a policy-conditioned, normalized, and adaptive-to-environment RA value function $V^{\pi}(s, \hat{e})$ as a safety guard. Guarding is triggered when $V^{\pi}(s, \hat{e}) > 0$ and then the system calls recovery policy to take control.

3.3. Phase 3: On-policy Estimator Finetuning

To ensure the estimator performs well during inference and to cope with the distribution shift of the history buffer introduced by switching policies, we fine-tune it with supervision in deployment where the agile policy, recovery policy, and RA network work together. This is different from the training session in Section 3.1 in that we generate the history rollouts only with the agile policy in phase 1, but with both policies taking effects in turn in this phase.

4. SIMULATION EXPERIMENTS

In this section, we present a series of simulation experiments in IsaacGym (Rudin et al., 2022; Makoviychuk et al., 2021) to investigate the following research questions:

Q1: What are the most effective methodologies for achieving a balance between adaptive safety and agility in robotic systems?

Q2: What is the recipe for training a the best estimation module in BAS?

Q3: How can we quantitatively assess the adaptivity and robustness of the BAS framework through in-depth analytical and experimental evaluations?

Q4: How well does BAS performs in real-world unseen scenarios, and how accurate can BAS's parameter estimator be in real-world?

We followed the simulation setup and rewards settings in ABS (He et al., 2024), and the domain randomization settings are as shown in Table 3.

Policy	Collision Rate(%) ↓	Reach Rate(%) ↑	Timeout Rate(%)	$\bar{v}_{peak}$ of success (m/s) ↑
a) Adaptivity-wise				
BAS	**1.11**	**93.84**	5.06	**2.70**
BAS w/o explicit estimator	5.64	90.50	3.86	2.65
ABS	14.84	63.83	21.33	2.65
RMA-RA	12.51	80.12	7.37	**2.70**
Action-Distillation	15.72	68.99	15.29	2.63
b) Safety-wise				
BAS	**1.11**	**93.84**	5.06	2.70
BAS-Lagrangian	3.20	90.40	6.40	2.51
RMA-Lagrangian	13.69	76.33	9.98	2.48
BAS-π_{agile}	10.35	89.00	0.65	2.75
c) For adptivity-robustness analysis				
BAS	1.11	93.84	5.06	2.70
BAS-random	100.00	0.00	0.00	/
RMA-RA	12.51	80.12	7.37	2.70
RMA-RA-random	19.37	74.59	6.04	2.49

Table 1: Simulation experimental results. *ABS values may differ from He et al. (2024) because these experiment are done under larger domain randomizations, as shown in Table 3.

4.1. Safety and Agility Performance Analysis

To answer Q1 (*What are the most effective methodologies for achieving a balance between adaptive safety and agility?*), we compare the safe rates and average speed in simulation of BAS and other adaptive-and/or-safe locomotion baselines. To show BAS's adaptivity, we introduce the following baselines: **1) ABS**, which has non-adaptive agile policy and RA network; **2) BAS w/o explicit estimator**, which adopts long-short term history structures, and learns an encoder which maps history to latent space with end-to-end RL training (Li et al., 2024); **3) RMA-RA**: we incorporate RMA (Kumar et al., 2021) and RA network with the latent environmental representation z_t as the inputs. **4)**

Action-Distillation, which is similar to RMA and inspired from Lee et al. (2020), where a student policy is distilled from an adaptive teacher policy by minimizing the difference between their actions. **5) BAS-π_{Agile}**: which only uses agile policy π_{agile}; **6) BAS-Lagrangian**, which learns the agile policy with PPO-Lagrangian (Alex Ray and Amodei, 2019) with explicit estimation without RA network; **7) RMA-Lagrangian**, which learns a teacher PPO-Lagrangian policy with RMA and then distills it into a student policy. As shown in Table 1 (a), BAS outperform original ABS by by 50% in reach rates in varied physics, and distinctively stands out with the lowest collision rate and the highest reach rate throughout all of the adaptive methods. And in Table 1 (b), the RA safeguard structure also outperforms policies trained by PPO-Lagrangian especially in agility. Moreover, the validation of the effect of the safetyguard to the whole system is obeserved through the comparison between BAS and BAS-π_{agile}, which indicates adopting RA guard would transfer most of the failure cases to success cases or safe cases.

4.2. Estimation Analysis

To answer Q2 *(How to train the best estimation module in BAS?)*, we investigate our proposed methodologies to train the estimator: joint-train pipeline with fusion and on-policy fine-tuning.

Estimator training pipeline For ablation purposes, we tested **w/o fusion** (arbitrarily setting α in Figure 2 to 0 or 1) and **split-trained** estimator(first learn a privileged policy then learn estimation from rollout data and use estimation as privileged observation at inference) as in Table 2(a), which shows that fusion interpolation offers a better accuracy on estimation and better overall agility-safety performance. Also, we demonstrated the tracking of mass as in Figure 4, where the jointly-trained estimator is much more accurate.

Entry	estimation loss	Collision Rate(%) ↓	Reach Rate(%) ↑	Timeout Rate(%)	$\bar{v}_{peak}$ of success (m/s)↑
a) Ablation: on training pipelines (before finetuning)					
BAS	**0.570**	**3.10**	**92.48**	4.42	**2.69**
BAS w/o fusion($\alpha \equiv 1$)	1.955	3.71	91.10	5.19	2.66
BAS w/o fusion($\alpha \equiv 0$)	5.008	16.31	52.30	31.39	2.63
BAS split-train	1.511	6.21	88.20	1.89	2.69
b) Ablation: on-policy finetuning					
BAS w/o finetuning	0.570	3.10	92.48	4.42	**2.69**
BAS	**0.323**	**1.11**	**93.84**	5.06	2.68

Table 2: Comparisons on estimators w/ and w/o fusion or joint-train and on-policy finetuning.

On-policy finetuning Note that the estimator is only trained under the agile policy's rollout, which may not have seen trajectories contributed by the agile and recovery policies. To this end, we implemented on-policy post-finetuning on the estimator to diminish this distribution shift in an end-to-end scheme. As can be seen in Table 2, on-policy fine-tuning improves both the accuracy of the estimator and the agility-safety performance.

4.3. Adaptivity-Robustness Analysis

To Answer Q3 (*Can we identify adaptivity of BAS with deeper analysis?*), we visualize the RA values under various physics conditions (see Figure 3). The trend in RA values aligns with common sense: heavier payloads correlate with increased danger.

Moreover, for further adaptivity-robustness analysis, we try to compare BAS to the classic adaptive baseline, RMA. As robustness can be identified by 'robustness' to noise on adaptation modules,

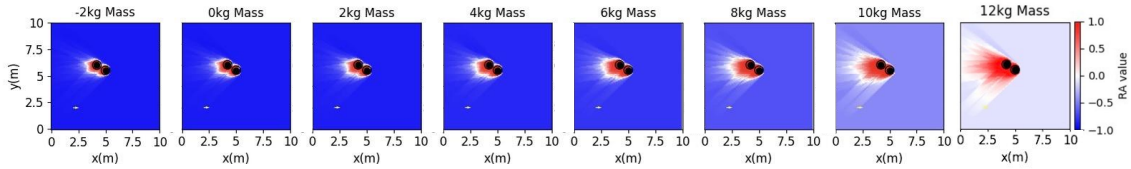
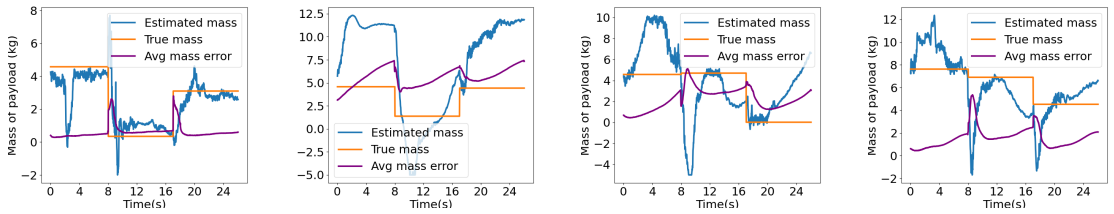


Figure 3: RA Value Heatmap under different mass of payload at the state of 3m/s base linear velocity right forward. The more reddish, the higher RA value the more dangerous; the more bluish, the lower RA value the safer.



(*a*) BAS (*b*) BAS w/o fusion ($\alpha \equiv 0$) (*c*) BAS w/o fusion ($\alpha \equiv 1$) (*d*) BAS w/o co-train

Figure 4: Mass estimation tracking of BAS, BAS w/o fusion and BAS w/o co-train pipeline.

we tested **BAS-random** and **RMA-RA-random**, where the adaptation module output (explicit estimation in BAS, latent z_t in RMA) is replaced with random numbers within the same numerical ranges.

As Table 1(c) shows, BAS deviates when the predicted mass is masked with random numbers, while masking RMA's latent vector has a minor loss on performance, which means BAS is less conservative and more adaptive than RMA.

Explicit estimation also enhances the interpretability of the system by providing a clear understanding of the underlying physical significance of the estimations. Conversely, if environments are encoded to latent space, their meaning may remain obscure.

Hyperparameter Name	Value
Mass of Payload range(kg)	-2.0,12.0
Friction range	0.25,1.5
CoM shift-x(m) range	-0.05,0.05
CoM shift-y(m) range	-0.05,0.05
CoM shift-z(m) range	-0.05,0.15
External Force-x range	-15N,15N
External Force-y range	-15N,15N

Table 3: Domain Randomization Setting

To sum up, BAS outperforms other baselines in agility and safety across our testing environments, demonstrating that its adaptivity, safety, and agility are all bridged together.

4.4. Real-world Experiments

4.4.1. EXPERIMENT SETUP

To answer Q4 *(How well does BAS performs in real-world unseen scenarios, and how accurate can BAS's parameter estimator be in real-world?)*, we deploy our modules to a Unitree Go1 with onboard computations on NVIDIA Orin NX. We test three entries here: BAS, ABS and RMA+Lagrangian, as described in Section 4.1, among which ABS is our prior work and RMA+Lagrangian is also an adaptive-and-safe baseline which is worth comparing.

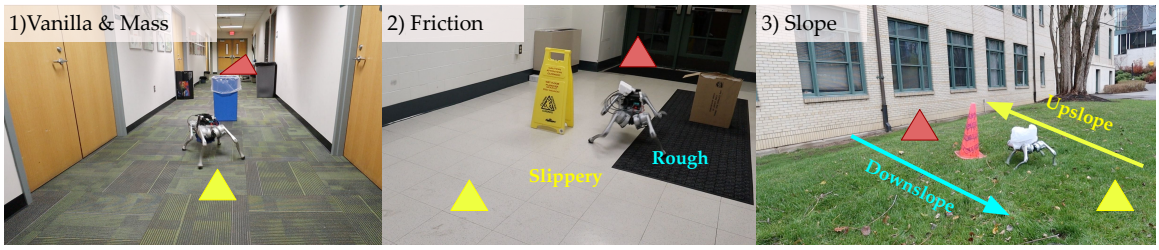


Figure 5: Real Experiment for Adaptive Safety test Settings, where yellow triangle notes the starting point and red triangle notes the goal. Once the robot reaches the goal, we switch the goal and starting point. A trajectory from the start to the goal and then get back to start without collision, is counted as success. **0) Vanilla test:** Same as Mass test but without payloads. **1) Mass test**: carry a 5kg payload in a corridor and avoid boxes. **2) Friction test**: avoid box and slip sign on very slippery floor and a fry mattress. **3) Slope test**: avoid cone on grass slope after rain, which is also very slippery.

In our experiment, the agility tests measures the agility of a policy under conditions as in Figure 5 but without obstacles, and the safety tests quantify the safety by statistics on success rates in different environments Figure 5.

As shown in Figure 5, we have different environmental settings for each physical factor that should be adapted, along with the vanilla test. Note that CoM shift is very hard to identify in real, so as an alternative, we build the overall test which is the slope test on grass.

4.4.2. REAL WORLD SAFETY-AGILITY PERFORMANCES

As shown in Table 4, BAS outperforms ABS and RMA+lagrangian in both safety and agility across different physics and settings. We also find that ABS totally cannot turn with 5kg payload or maintain safety on slippery terrains during experiments. During experiments, we also come across some failures for BAS. And we analyzed some causes of failure: 1) Restricted by limited visual angle. 2) The robot only got mild collision with obstacles. 3) Indoor environment is highly obstacle-dense and the ray-prediction network deviates due to out-of-distribution rays.

Policy	**Adaptive Agility test(s)↓**					**Adaptive Safety test↑**				
	Vanilla	Mass	Slope	Friction	Avg.	Vanilla	Mass	Slope	Friction	Avg.
BAS	**1.39**	**1.67**	**1.50**	**1.09**	**1.41**	**8/8**	**7/8**	**5/8**	**6/8**	**81.25%**
ABS	1.52	2.37	1.67	1.40	1.74	7/8	1/8	3/8	0/8	34.38%
RMA-Lag	1.76	1.92	1.85	2.02	1.89	6/8	5/8	0/8	2/8	40.63%

Table 4: Test results in real world. For pure agility tests we compare the average time consumed to run 2.4m from stance in 3 trials. For safety-related tests, we compare the average success rate of 8 trials.

4.4.3. REAL WORLD ADAPTATION ANALYSIS AND RUN-TIME ESTIMATION

Figure 6 shows that BAS maintains adaptive safety even under online sudden environmental changes, such as extracting the 8kg payload or the sudden change of terrain properties like friction, while ABS fails with insufficient adaptivity to maintain safety in the case. Moreover, as shown in the esti-

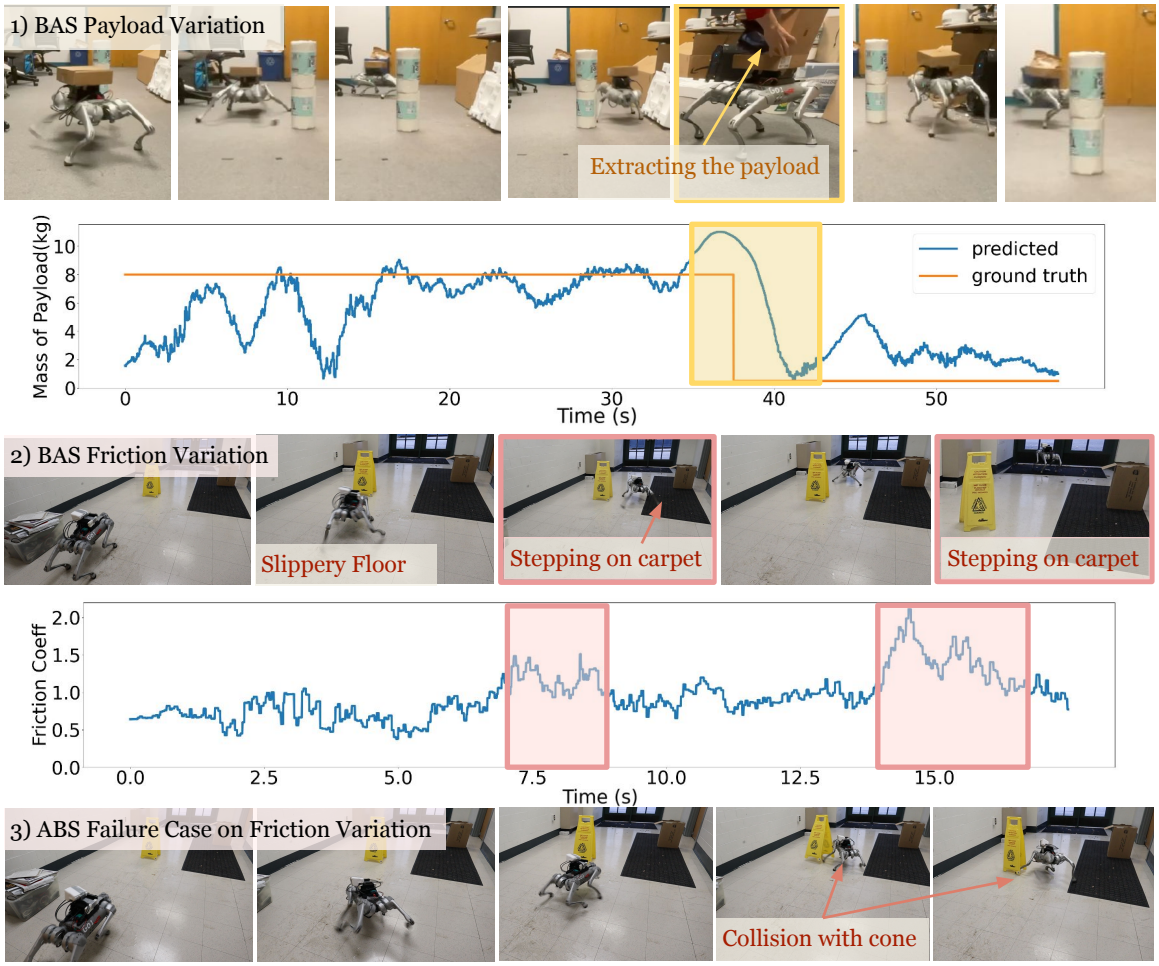


Figure 6: Adaptation analysis in real with online changes in environment. 1) BAS Accomplishing collision avoidance while carrying a 8kg payload at first and then no payload in one trajectory. 2) BAS Accomplishing collision avoidance in terrains with different frictions(liquid soap and water on floor and dry mattress), while 3) ABS fails due to the lack of adaptivity. BAS estimator functions well in both cases with a correct trend.

mation plots in Figure 6, the estimation remains accurate after the changes, and BAS accomplishes avoiding obstacles under all the environmental conditions, which validates its adaptivity.

5. CONCLUSIONS AND FUTURE PROSPECTS

In this paper, we propose BAS, which bridges adaptivity and safety together. We came up with two interesting research topics based on BAS: 1) Tackle 3D reach-avoid tasks with semantic information; 2) Since BAS assumes all the future physics as identical to the current frame, it makes sense to combine planning with safety adaptation.

Acknowledgments

We gratefully acknowledge the dedication and contribution of Guanqi He who helps us repair the hardware. And we appreciate Haotian Lin for his assistance in real-world experiments.

References

Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization, 2017. URL https://arxiv.org/abs/1705.10528.

Joshua Achiam Alex Ray and Dario Amodei. Bench-marking safe exploration in deep reinforcement learning. Technical report, 2019. URL https://openai.com/research/benchmarking-safe-exploration-in-deep-reinforcement-learning.

Mukhriddin Arabboev, Shohruh Begmatov, Khabibullo Nosirov, Alisher Shakhobiddinov, Jean Chamberlain Chedjou, and Kyandoghere Kyamakya. Development of a prototype of a search and rescue robot equipped with multiple cameras. In *2021 International Conference on Information Science and Communications Technologies (ICISCT)*, pages 1–5, 2021. doi: 10.1109/ICISCT52966.2021.9670087.

Somil Bansal, Mo Chen, Sylvia Herbert, and Claire J. Tomlin. Hamilton-jacobi reachability: A brief overview and recent advances, 2017. URL https://arxiv.org/abs/1709.07523.

Javier Borquez, Kensuke Nakamura, and Somil Bansal. Parameter-conditioned reachable sets for updating safety assurances online. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, page 10553–10559. IEEE, May 2023. doi: 10.1109/icra48891.2023.10160554. URL http://dx.doi.org/10.1109/ICRA48891.2023.10160554.

Lukas Brunke, Melissa Greeff, Adam W. Hall, Zhaocong Yuan, Siqi Zhou, Jacopo Panerati, and Angela P. Schoellig. Safe learning in robotics: From learning-based control to safe reinforcement learning. *Annual Review of Control, Robotics, and Autonomous Systems*, 5(Volume 5, 2022):411–444, 2022. ISSN 2573-5144. doi: https://doi.org/10.1146/annurev-control-042920-020211. URL https://www.annualreviews.org/content/journals/10.1146/annurev-control-042920-020211.

Russell Buchanan, Lorenz Wellhausen, Marko Bjelonic, Tirthankar Bandyopadhyay, Navinda Kottege, and Marco Hutter. Perceptive whole-body planning for multilegged robots in confined spaces. *Journal of Field Robotics*, 38(1):68–84, 2021. doi: https://doi.org/10.1002/rob.21974. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/rob.21974.

Jia-Ruei Chiu, Jean-Pierre Sleiman, Mayank Mittal, Farbod Farshidian, and Marco Hutter. A collision-free mpc for whole-body dynamic locomotion and manipulation, 2022. URL https://arxiv.org/abs/2202.12385.

Jiawei Gao, Ziqin Wang, Zeqi Xiao, Jingbo Wang, Tai Wang, Jinkun Cao, Xiaolin Hu, Si Liu, Jifeng Dai, and Jiangmiao Pang. Coohoi: Learning cooperative human-object interaction with manipulated object dynamics. *arXiv preprint arXiv:2406.14558*, 2024.

Tairan He, Chong Zhang, Wenli Xiao, Guanqi He, Changliu Liu, and Guanya Shi. Agile but safe: Learning collision-free high-speed legged locomotion, 2024. URL https://arxiv.org/abs/2401.17583.

David Hoeller, Nikita Rudin, Dhionis Sako, and Marco Hutter. Anymal parkour: Learning agile navigation for quadrupedal robots, 2023. URL https://arxiv.org/abs/2306.14874.

Kai-Chieh Hsu*, Vicenç Rubies-Royo*, Claire Tomlin, and Jaime Fisac. Safety and liveness guarantees through reach-avoid reinforcement learning. In *Robotics: Science and Systems XVII*, RSS2021. Robotics: Science and Systems Foundation, July 2021. doi: 10.15607/rss.2021.xvii.077. URL http://dx.doi.org/10.15607/RSS.2021.XVII.077.

Kai-Chieh Hsu, Allen Z. Ren, Duy P. Nguyen, Anirudha Majumdar, and Jaime F. Fisac. Sim-to-lab-to-real: Safe reinforcement learning with shielding and generalization guarantees. *Artificial Intelligence*, 314:103811, January 2023. ISSN 0004-3702. doi: 10.1016/j.artint.2022.103811. URL http://dx.doi.org/10.1016/j.artint.2022.103811.

Jemin Hwangbo, Joonho Lee, Alexey Dosovitskiy, Dario Bellicoso, Vassilios Tsounis, Vladlen Koltun, and Marco Hutter. Learning agile and dynamic motor skills for legged robots. *Science Robotics*, 4(26), January 2019. ISSN 2470-9476. doi: 10.1126/scirobotics.aau5872. URL http://dx.doi.org/10.1126/scirobotics.aau5872.

Gwanghyeon Ji, Juhyeok Mun, Hyeongjun Kim, and Jemin Hwangbo. 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, April 2022. ISSN 2377-3774. doi: 10.1109/lra.2022.3151396. URL http://dx.doi.org/10.1109/LRA.2022.3151396.

D. Kim, D. Carballo, J. Di Carlo, B. Katz, G. Bledt, B. Lim, and S. Kim. Vision aided dynamic exploration of unstructured terrain with a small-scale quadruped robot. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2464–2470, 2020. doi: 10.1109/ICRA40945.2020.9196777.

Ashish Kumar, Zipeng Fu, Deepak Pathak, and Jitendra Malik. Rma: Rapid motor adaptation for legged robots, 2021. URL https://arxiv.org/abs/2107.04034.

Joonho Lee, Jemin Hwangbo, Lorenz Wellhausen, Vladlen Koltun, and Marco Hutter. Learning quadrupedal locomotion over challenging terrain. *Science Robotics*, 5(47), October 2020. ISSN 2470-9476. doi: 10.1126/scirobotics.abc5986. URL http://dx.doi.org/10.1126/scirobotics.abc5986.

Sergey Levine, Aviral Kumar, George Tucker, and Justin Fu. Offline reinforcement learning: Tutorial, review, and perspectives on open problems, 2020. URL https://arxiv.org/abs/2005.01643.

Zhongyu Li, Xue Bin Peng, Pieter Abbeel, Sergey Levine, Glen Berseth, and Koushil Sreenath. Reinforcement learning for versatile, dynamic, and robust bipedal locomotion control, 2024. URL https://arxiv.org/abs/2401.16889.

Zuxin Liu, Zhepeng Cen, Vladislav Isenbaev, Wei Liu, Zhiwei Steven Wu, Bo Li, and Ding Zhao. Constrained variational policy optimization for safe reinforcement learning, 2022. URL https://arxiv.org/abs/2201.11927.

Junfeng Long, ZiRui Wang, Quanyi Li, Liu Cao, Jiawei Gao, and Jiangmiao Pang. Hybrid internal model: Learning agile legged locomotion with simulated robot response. In *The Twelfth International Conference on Learning Representations*, 2024.

Shixin Luo, Songbo Li, Ruiqi Yu, Zhicheng Wang, Jun Wu, and Qiuguo Zhu. Pie: Parkour with implicit-explicit learning framework for legged robots, 2024. URL https://arxiv.org/abs/2408.13740.

Viktor Makoviychuk, Lukasz Wawrzyniak, Yunrong Guo, Michelle Lu, Kier Storey, Miles Macklin, David Hoeller, Nikita Rudin, Arthur Allshire, Ankur Handa, and Gavriel State. Isaac gym: High performance gpu-based physics simulation for robot learning, 2021. URL https://arxiv.org/abs/2108.10470.

Kostas Margellos and John Lygeros. Hamilton–jacobi formulation for reach–avoid differential games. *IEEE Transactions on Automatic Control*, 56(8):1849–1861, 2011. doi: 10.1109/TAC.2011.2105730.

Farzad H. Panahi, Fereidoun H. Panahi, and Tomoaki Ohtsuki. An intelligent path planning mechanism for firefighting in wireless sensor and actor networks. *IEEE Internet of Things Journal*, 10 (11):9646–9661, 2023. doi: 10.1109/JIOT.2023.3235998.

Nikita Rudin, David Hoeller, Philipp Reist, and Marco Hutter. Learning to walk in minutes using massively parallel deep reinforcement learning, 2022. URL https://arxiv.org/abs/2109.11978.

David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, Yutian Chen, Timothy Lillicrap, Fan Hui, Laurent Sifre, George van den Driessche, Thore Graepel, and Demis Hassabis. Mastering the game of Go without human knowledge. *Nature*, 550(7676):354–359, October 2017. ISSN 1476-4687. doi: 10.1038/nature24270. URL https://doi.org/10.1038/nature24270.

Shang-jie Sun, Shu-hai Jiang, Song-he Cui, Yue Kang, and Yu-tang Chen. Path planning of forest fire-fighting robots based on deep learning. 36:51–57, 2020. ISSN 1006-8023.

Zhicheng Wang, Wandi Wei, Ruiqi Yu, Jun Wu, and Qiuguo Zhu. Toward understanding key estimation in learning robust humanoid locomotion, 2024. URL https://arxiv.org/abs/2403.05868.

Wenli Xiao, Tairan He, John Dolan, and Guanya Shi. Safe deep policy adaptation, 2024. URL https://arxiv.org/abs/2310.08602.

Tengyu Xu, Yingbin Liang, and Guanghui Lan. Crpo: A new approach for safe reinforcement learning with convergence guarantee, 2021. URL https://arxiv.org/abs/2011.05869.

Yuxiang Yang, Guanya Shi, Xiangyun Meng, Wenhao Yu, Tingnan Zhang, Jie Tan, and Byron Boots. Cajun: Continuous adaptive jumping using a learned centroidal controller, 2023. URL https://arxiv.org/abs/2306.09557.

Kai S. Yun, Rui Chen, Chase Dunaway, John M. Dolan, and Changliu Liu. Safe control of quadruped in varying dynamics via safety index adaptation, 2024. URL https://arxiv.org/abs/2409.09882.

Yuanhang Zhang, Tianhai Liang, Zhenyang Chen, Yanjie Ze, and Huazhe Xu. Catch it! learning to catch in flight with mobile dexterous hands. *arXiv preprint arXiv:2409.10319*, 2024.