2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

TOLEBI: Learning Fault-Tolerant Bipedal Locomotion via Online Status Estimation and Fallibility Rewards

Hokyun Lee[1], Woo-Jeong Baek[2*], Junhyeok Cha[1], and Jaeheung Park[1,3*]

***Abstract*— With the growing employment of learning algorithms in robotic applications, research on reinforcement learning for bipedal locomotion has become a central topic for humanoid robotics. While recently published contributions achieve high success rates in locomotion tasks, scarce attention has been devoted to the development of methods that enable to handle hardware faults that may occur during the locomotion process. However, in real-world settings, environmental disturbances or sudden occurrences of hardware faults might yield severe consequences. To address these issues, this paper presents *TOLEBI: A fault-tolerant learning framework for bipedal locomotion* that handles faults on the robot during operation. Specifically, joint locking, power loss and external disturbances are injected in simulation to learn fault-tolerant locomotion strategies. In addition to transferring the learned policy to the real robot via sim-to-real transfer, an online joint status estimator incorporated. This module enables to classify joint conditions by referring to the actual observations at runtime under real-world conditions. The validation experiments conducted both in real-world and simulation with the humanoid robot TOCABI highlight the applicability of the proposed approach. To our knowledge, this work provides the first learning-based fault-tolerant framework for bipedal locomotion, thereby fostering the development of efficient learning methods in this field.**

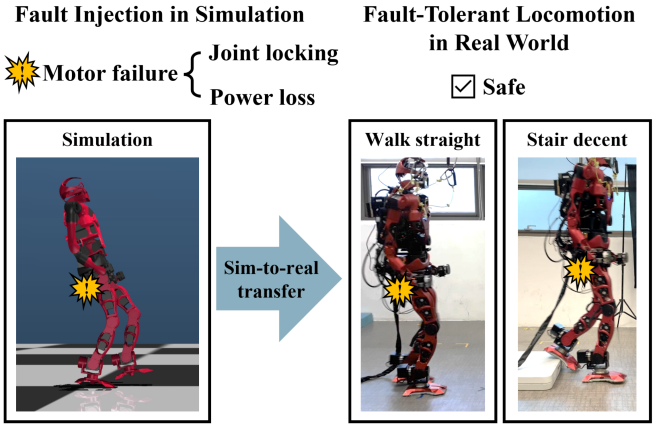

Fig. 1: **TOLEBI, A framework for learning fault-tolerant bipedal locomotion.** Motor failures are injected during training in simulation to learn a fault-tolerant locomotion policy, and the policy is transferred to the real humanoid robot.

I. Introduction

Dealing with unexpected system failures during operation is one crucial issue in real-world robotic applications. While extensively studied across robotics subdomains, the derivation of suitable techniques that enable to avoid the negative consequences of undesired faults has become increasingly challenging with the growing employment of learning algorithms. Indeed, the performance and flexibility of robot applications has been improved significantly with advances in the learning domain. For example, efforts for deriving complex robot control algorithms, like bipedal locomotion for humanoid robots or manipulation tasks can be efficiently reduced via reinforcement learning [1], [2]. However, one drawback of learning methods is their black-box character that makes the prediction regarding unseen data difficult [3], [4]. As a consequence, compromises that reduce the flexibility and retain robot safety are required to facilitate the application of according systems under real-world conditions. In the last decade, bipedal locomotion has gained increased attention [5] in the humanoid robotics domain. However, controllers that handle hardware failures in bipedal locomotion are missing in current robotics literature. In contrast to quadruped robots, reduced functionalities on one leg can significantly degrade the system performance for biped robots. Sudden occurrences of faults on one leg can result in situations where the robot loses its balance and falls. Particularly, these events can occur in an unexpected manner at runtime due to environmental disturbances or undesired force perturbations [6], [7]. Therefore, the derivation of methods that can deal with faults inherently is essential. One critical challenge here lies in retaining the benefits of reinforcement learning while ensuring that faults and their consequences can be mitigated despite the black-box character of learning algorithms. Recent works that present fault-tolerant strategies for robot locomotion have been designed for quadruped robot systems, where the fault behavior is assumed for one of four legs. While promising performances have been achieved, these approaches cannot be directly applied to bipedal locomotion. Furthermore, model-based methods for quadruped systems that handle faults by incorporating them in the control explicitly like [8] exist. Here, the faults are modeled and incorporated manually into the controller.

This work was supported by the Technology Innovation Program (RS-2025-25453780, Development of a National Humanoid AI Robot Foundation Model for Multi-Task Applications) funded By the Ministry of Trade, Industry, and Resources (MOTIR, Korea).

[1]Hokyun Lee, Junhyeok Cha and Jaehueng Park are with the Department of Intelligence and Information, Graduate School of Convergence Science and Technology, Seoul National University, Seoul 08826, Republic of Korea. [hkleetony, threeman1, park73]@snu.ac.kr

[2]Woo-Jeong Baek is with the Artificial Intelligence Institute (AIIS), Seoul National University, Republic of Korea. wjbaek@snu.ac.kr

[3]Jaeheung Park is also with Advanced Institutes of Convergence Technology (AICT), Suwon 16229, Republic of Korea and with ASRI, AIIS, Seoul National University, Seoul 08826, Republic of Korea.

*Corresponding authors: Jaeheung Park and Woo-Jeong Baek

While their performance regarding the fault detection and failure reduction has been validated, their applicability is limited to the situations considered in the models. Therefore, model-based approaches struggle to deal with unseen situations. In order to derive a locomotion framework for bipedal robot locomotion that exploits the benefits of learning and can handle faults, this paper presents **TOLEBI** (a fault**T**-t**O**lerant **L**earning fram**E**work for **B**ipedal locomot**I**on) based on reinforcement learning with the humanoid robot TOCABI [9], as shown in Fig. 1. In particular, TOLEBI leverages phase modulation actions and the proposed fallibility rewards to derive a control policy that can deal with faults while transferring the policy successfully to the real robot. To be specific, the reward is designed with respect to the foot force aiming that the contact force between the foot and the floor is reduced. Scientifically, the novelty of the TOLEBI lies in training a joint status estimator online on the basis of a curriculum of motor failure simulations. The online estimation of the current joint status is considered as the observation, which enables to achieve higher robustness. The performance of TOLEBI is evaluated on two real-world locomotion scenarios: Walking on a plain floor and descending stairs. Therefore, this manuscript presents the first work in the robotics domain that presents a learning-based fault-tolerant bipedal locomotion approach for real-world environments. The remainder of this paper is structured as follows: Section II introduces state-of-the-art literature in the field of learning for bipedal locomotion, thereby highlighting the scientific novelties of the TOLEBI. After summarizing the preliminaries in Section III, TOLEBI is derived in Section IV by specifying the reinforcement learning algorithm with its rewards. In addition, the policy learning approach and the sim-to-real transfer are described in detail. The results of the validation experiments are presented in Section V. Finally, Section VI summarizes the scientific findings and suggests directions for future research.

II. Related Work

A. Reinforcement Learning for Bipedal Locomotion

Robot locomotion control has long been a central research topic in robotics, and recent advances in deep reinforcement learning (DRL) have established it as a key methodology for developing robust controllers in complex and dynamic environments. DRL has been successfully applied to a wide range of tasks including fall recovery control for quadruped robots [10], locomotion control conditioned on parameterized gait and task inputs for bipedal robots [11], [12], and improving controller generalization through diverse behavioral training [13]. More recently, real-world deployment of DRL-based controllers has enabled successful locomotion on humanoid robots [14]. These advances have been largely facilitated by high-performance simulation environments such as Isaac Gym [15] that enable to process large-scale parallel data collection with high-fidelity physics, thereby bridging the sim-to-real gap and allowing the development of versatile, dynamic, and transferable locomotion policies like presented in the contributions [16], [17].

B. Fault-tolerant Locomotion

When robots operate in real-world environments, unexpected hardware failures such as actuator malfunctions can severely degrade performance or lead to complete loss of mobility. Fault-tolerant locomotion aims to maintain stability and accomplish mission objectives even under such fault conditions, significantly enhancing the reliability and practicality of legged robots. These can be split in model-based and learning-based approaches:

1) Model-based Methods: Early research on fault-tolerant control mainly relies on model-based methods, where accurate kinematic and dynamic models were used to design alternative gaits for failed actuators in quadruped [18], [19], and hexapod robots [20], [21]. Approaches such as posture optimization and whole-body control were introduced to handle specific fault scenarios in quadruped [8]. However, these methods often required extensive manual modeling, showed limited scalability in unstructured environments, and faced difficulties in dealing with perturbations or failures.

2) Learning-based Methods: To address these limitations, learning approaches have recently gained attention. As mentioned above, Reinforcement learning (RL) enables robots to learn robust control policies through interaction with diverse failure scenarios in simulation, eliminating the need for explicit fault modeling. In contrast, meta-learning techniques allow rapid adaptation to changing dynamics [22], while curriculum learning strategies gradually increase fault severity during training to improve robustness [23]. Recent works have proposed RL-based frameworks for quadrupedal robots that incorporate fault detection, recovery, and locomotion adaptation under degraded actuator conditions [24], [25], [26]. Methods such as random joint masking [27] and multi-task learning [28] further enhance generalization across different failure modes.

Despite these advances, the majority of existing methods focus on quadrupedal robots, where stability margins are inherently larger. In contrast, bipedal locomotion presents additional challenges due to reduced stability and higher risk of catastrophic falls under motor failures on one leg. This motivates the development of fault-tolerant learning frameworks specifically designed for humanoid robots operating under unexpected joint locking or power loss events. Therefore, the scientific contributions of this manuscript can be summarized as follows:

1) First, a fault-tolerant learning framework for bipedal locomotion **TOLEBI** is proposed that incorporates a curriculum learning approach and motor failure simulations to facilitate sim-to-real transfer.
2) Second, we integrate the online joint status estimator concurrently trained with the policy to infer joint status without additional training phases into TOLEBI.
3) Third, the fallibility reward is designed for TOLEBI under motor failure while preserving nominal locomotion.

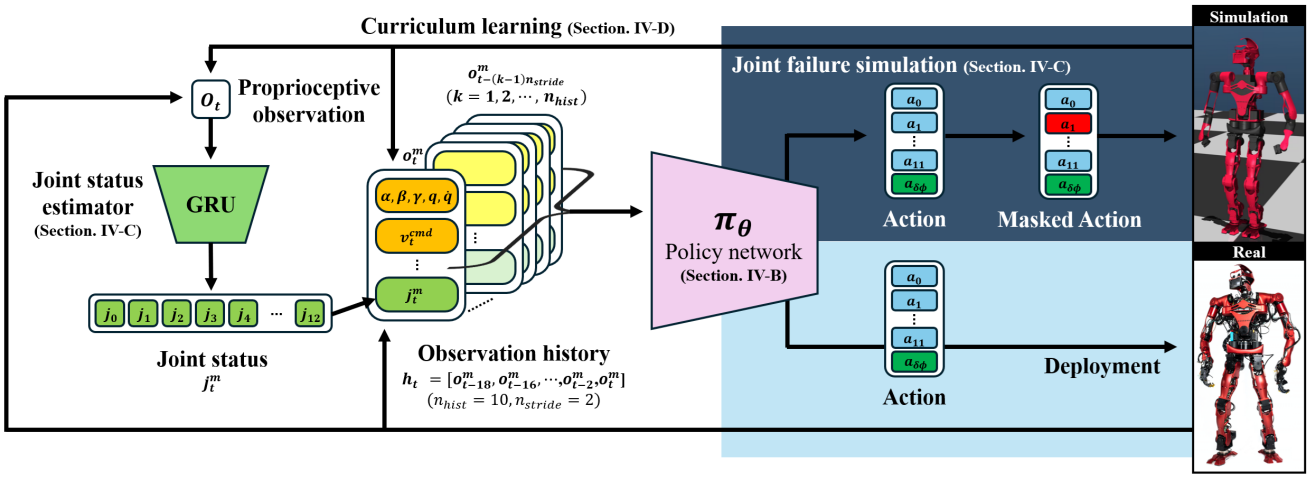

Fig. 2: **Schematic description of the framework TOLEBI (faulT-tOlerant Learning framEwork for Bipedal locomotIon).** A joint status estimator processes proprioceptive observations to infer joint status, storing the results in the observation history for policy training. During simulation, motor failure scenarios mask the corresponding actions, enabling robust fault-tolerant policy learning. The trained policy is deployed on the real humanoid robot for fault-tolerant locomotion.

III. Preliminaries

A. Reinforcement Learning

We formulate the bipedal locomotion control problem under motor failure as a Markov Decision Process (MDP). The MDP defined by a tuple $(\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma)$ that is composed of the state space $\mathcal{S}$, the action space $\mathcal{A}$, a transition probability function $\mathcal{P}$, a reward function $\mathcal{R}$, and a discount factor γ. The agent aims to learn a policy $\pi_\theta(a|s)$ by interacting with the environment to maximize the expected discounted return $J(\pi_\theta)$ defined as the cumulative discounted reward over a finite-horizon T:

$$J(\pi_\theta) = \mathbb{E}_{\tau \sim p(\tau|\pi_\theta)} \left[\sum_{t=0}^{T-1} \gamma^t r_t \right] \tag{1}$$

where τ denotes a trajectory generated under the policy π_θ.

B. Motor Failures

Motor failures in robots primarily arise from two sources. External disturbances can mechanically constrain the actuators while internal faults in electrical or control subsystems might disrupt power transmission or motor command signals. Both failure types can lead to significant performance degradation or even a complete loss of mobility during bipedal locomotion. This paper focuses on two failure scenarios: Joint locking and power loss. Here, **joint locking** occurs when actuators become stuck and prevent any motion in the affected joints. In contrast, **power loss** arises when joints remain physically free but cannot generate torque. Both failures severely compromise the robot's balance, hinder the ability to track commanded velocities, and disrupt stable locomotion.

IV. TOLEBI - A Fault-Tolerant Learning Framework for Bipedal Locomotion

This section introduces TOLEBI: A reinforcement learning framework for fault-tolerant bipedal locomotion under unexpected motor failures. The proposed approach integrates motor failure simulation with joint masking, joint status estimation, curriculum learning and an action structure that incorporates phase modulation for adaptive gait timing. The entire control strategy is trained using Proximal Policy Optimization (PPO) [29] in Issac Gym and deployed on the real robot to ensure robustness at operation.

A. Overview

Our proposed framework TOLEBI shown in Fig. 2 integrates joint status estimation, observation history modeling, and curriculum learning for fault-tolerant bipedal locomotion under motor failures. At each time step, the observation includes proprioceptive observation, command velocities, and the estimated joint status with a history buffer that stores the latest $n_{history} = 10$ entries with a stride of $n_{stride} = 2$ between them. This approach allows the policy to leverage sequential information without significant computational overhead. Motor failures are simulated by masking joint torque commands in Section IV-C enabling the policy to learn control under partial actuation. The joint status estimator appends inferred joint status to the observation history during curriculum learning in Section IV-D.

B. Reinforcement Learning for Biped Locomotion

1) State Space: The state $\mathcal{S} \in \mathbb{R}^{51}$ consists of the robot's base orientation in Euler angles $\Theta_t = (\alpha_t, \beta_t, \gamma_t) \in \mathbb{R}^3$, joint positions $q_t \in \mathbb{R}^{12}$ in the legs, joint velocities $\dot{q}_t \in \mathbb{R}^{12}$, and the walking phase information encoded as the sine and cosine of its phase $\Phi_t = (\sin(2\pi\phi_t), \cos(2\pi\phi_t)) \in \mathbb{R}^2$. The phase variable ϕ_t evolves cyclically from 0 to 1. The

TABLE I: REWARD FUNCTIONS AND SCALES AT EACH CURRICULUM LEARNING PHASE

Category	**Reward**	**Equation** (r_i)	**Scale in nominal phase** (w_i)	**Scale in fault phase** (w_i)
Task Rewards	Linear velocity tracking	$\exp(-\frac{1}{0.45^2}\|\mathbf{v}_{xy}^{cmd}-\mathbf{v}_{xy}\|_2^2)$	0.4	0.4
	Angular velocity tracking	$\exp(-\frac{1}{0.35^2}\|\mathbf{w}_{z}^{cmd}-\mathbf{w}_{z}\|^2)$	0.2	0.2
	Foot contact	$\mathbf{1}_{\text{DSP/RSSP/LSSP sync}}$	0.2	0.2
Regulation Terms	Body orientation	$\exp(-500.0(roll^2+pitch^2))$	0.3	0.3
	Joint torque	$\exp(-\frac{1}{100.0}\|\boldsymbol{\tau}\|)$	0.05	0.05
	Joint velocity	$\exp(-\frac{1}{100.0}\|\dot{\boldsymbol{q}}\|)$	0.05	0.05
	Joint acceleration	$\exp(-\frac{1}{0.05}\|\ddot{\boldsymbol{q}}\|)$	0.05	0.05
	Feet contact force	$\exp(-\frac{1}{140.0}\sum_{i\in\{L,R\}}\|\text{ReLU}(\mathbf{F}_z^i-1.4W)\|)$	0.1	0.1
	Torque difference	$\exp(-\frac{1}{1.20^2}\|\tau_t-\tau_{t-1}\|)$	0.7	0.7
	Contact force difference	$\exp(-\frac{1}{100.0}\sum_{i\in\{L,R\}}\|\mathbf{F}_{z,t}^i-\mathbf{F}_{z,t-1}^i\|)$	0.2	0.2
Fallibility Rewards	Trajectory mimicking	$\exp(-\frac{1}{0.5}\|\mathbf{q}^{ref}-\mathbf{q}\|^2)$	**0.35**	**0.35**
	Contact force tracking	$\exp(-\frac{1}{10.0}\sum_{i\in\{L,R\}}\|\mathbf{F}_z^{i,ref}-\mathbf{F}_z^i\|)$	0.0	**0.3**
	Termination penalty	$\mathbf{1}_{\text{terminate}}$	0.0	**-100.0**

command velocity $v_t^{cmd} = (v_t^x, v_t^y, \omega_t^z) \in \mathbb{R}^3$, and base velocity $v_t^{base} \in \mathbb{R}^6$. The state also incorporates a joint status vector $j_t^m \in \mathbb{R}^{13}$ inferred online by a GRU-based joint status estimator during training. Formally, the state is expressed as:

$$s_t = \{\Theta_t, q_t, \dot{q}_t, \Phi_t, v_t^{cmd}, v_t^{base}, j_t^m\} \tag{2}$$

In the joint status vector, the first dimension indicates whether the entire system is in a healthy state, while the remaining 12 dimensions represent the operational status of individual motors.

2) Action Space: The action space $\mathcal{A} \in \mathbb{R}^{13}$ comprises 12 joint torque commands and an additional action for modulating the phase ϕ_t. At each timestep, the policy outputs the mean μ_θ of a Gaussian distribution $\mathcal{N}(\mu_\theta, \sigma^2)$, from which actions are sampled. The standard deviation σ is predetermined according to the torque limits of individual joints for exploration. The action space also includes a phase modulation action $a_{\delta\phi,t}$, which plays a critical role in adapting locomotion timing under motor failures. To enable rapid adaptation when actuators malfunction, the phase modulation action adjusts the motion period and timing as follows:

$$\phi_{t+1} = \left(\phi_t + \frac{\Delta t}{T_{ref}} + a_{\delta\phi,t}\right) \bmod 1.0 \tag{3}$$

By directly influencing the gait timing, $a_{\delta\phi,t}$ allows the policy to modify the motion cycle and maintain stability under unexpected motor failures.

3) Reward Function: The reward function is designed to ensure stable and fault-tolerant bipedal locomotion while promoting energy efficiency, smooth control, and sim-to-real transfer. At each timestep, the total reward is defined as the weighted sum of multiple components as follows:

$$r_{total} = r_{task} + r_{regulation} + r_{fall} \tag{4}$$

Task Rewards. Task rewards encourage the robot to track commanded linear and angular velocities and maintain proper foot contacts. The foot contact reward r_c, is given for matching the gait cycle, which is divided into Double Support Phase (DSP), Right Single Support Phase (RSSP), and Left Single Support Phase (LSSP):

$$r_{task} = w_{v,xy} r_{v,xy} + w_{w,z} r_{w,z} + w_c r_c \tag{5}$$

Regulation Terms. We introduce the regulation terms to improve motion smoothness, energy efficiency, and physical feasibility during locomotion. These terms penalize sudden or excessive movements in body posture, joint dynamics, and interaction forces. The total regulation reward is given by

$$\begin{aligned} r_{\text{regulation}} = w_o r_o + w_\tau r_\tau + w_v r_v + w_a r_a + w_f r_f \\ + w_{\Delta\tau} r_{\Delta\tau} + w_{\Delta f} r_{\Delta f}, \end{aligned} \tag{6}$$

where each weight w balances the contribution of the respective term.

Fallibility Rewards. Finally, the proposed fallibility reward is a composite function designed to maintain stable locomotion under motor failures. The first is a trajectory mimic reward r_q, encourages tracking of nominal joint trajectory q^{ref} even under faulty conditions. Alternative strategies for quadruped locomotion—such as removing the reward [27] or adding rewards to lift the leg with the failed motor [30]—turned out to be unsuitable for bipedal locomotion in our work. Removing the reward caused the policy to adopt an overly stable, crouching gait that lost its natural walking style and was not transferable to the real robot. A foot-lifting strategy is similarly impractical, as it requires the robot to hop on one leg. The second term, a force reference reward $r_{f,ref}$ mitigates large impacts caused by early foot contacts under fault conditions by encouraging adherence to

the reference foot contact force F_z^{ref}. Finally, a termination penalty r_T imposes a strong penalty when the episode ends due to falling or self collisions:

$$r_{fall} = w_q r_q + w_{f,ref} r_{f,ref} + w_T r_T \tag{7}$$

where the definitions of the reward terms r_i and the weights w_i that balance their contributions are listed in Table I.

C. Motor Failure Simulation and Status Estimation

1) Failure Simulation: We simulate two types of motor failures: Joint locking and power loss by masking the action commands during training. For each training iteration, 90% of the environments are randomly assigned to a fault condition. In this subset, the failure type is sampled with a 50% probability for each case. Next, a joint index $j \in \{0, 1, \ldots, 11\}$ is uniformly selected to mask its corresponding action.

Joint Locking. The joint torque is computed via predefined proportional gains K_p and derivative gains K_d to hold the current joint position q_j^0 fixed at the moment of failure.

Power Loss. The commanded torque is set to zero to effectively disable actuation. Formally, the masked torque command τ_j for joint j is given by

$$\tau_j = \begin{cases} K_p(q_j^0 - q_j) - K_d \dot{q}_j, & \text{if joint locking} \\ 0, & \text{if power loss} \\ \tau_j, & \text{otherwise} \end{cases} \tag{8}$$

Since our policy directly outputs the joint torque commands τ_j as actions, this masking strategy imposes realistic failure conditions and enables the policy to learn robust locomotion behaviors under partial actuation.

2) Status Estimation: A joint status estimator based on a single-layer GRU with a hidden size of 128 and a learning rate of 10^{-4} is trained online to infer joint status from proprioceptive inputs. A joint status is classified as faulty (status = 1) when the estimator output, produced by a sigmoid activation in the range $[0, 1]$, exceeds a threshold of 0.7. The estimator does not distinguish between joint locking and power loss, considering both as fault status. The estimator is optimized using the BCE loss between the predicted probabilities and the actual masking indices, with the threshold applied only at the decision stage rather than during training. The online learning scheme updates the estimator continuously in parallel with policy training, allowing it to gradually adapt to changing joint status. The estimated joint status vector is then appended to the observation space so that the policy can adjust its control commands based on real-time estimates of motor health.

D. Curriculum Learning

Curriculum learning plays an important role in stabilizing training for complex tasks like fault-tolerant humanoid locomotion. Rather than exposing the policy to all challenging conditions from the beginning, TOLEBI gradually increases task complexity based on the agent's performance, allowing the policy to acquire fundamental locomotion skills before adapting to motor failures and external disturbances. As outlined in Algorithm 1 the training starts with nominal locomotion under ideal conditions without failures. This approach is taken because our preliminary experiments showed that premature exposure to faults led to unstable training and degraded performance. When the average episode length exceeds 20 seconds, motor failure simulation is introduced following the masking strategy in Section IV-C. The fallibility rewards are adjusted according to Table I. where premature exposure to faults led to unstable training and degraded performance. Subsequently, when the average episode length under motor failures surpasses 24 seconds, push perturbations are applied to the robot's base to improve robustness for sim-to-real transfer. This progressive curriculum provides a structured path from nominal locomotion to fault-tolerant and disturbance-resilient behaviors.

Algorithm 1 Curriculum Policy Learning

```
Initialize policy π_θ with PPO
for epoch = 1, 2, ..., N do
    Collect rollouts with current policy π_θ
    Compute average episode length L_k
    if L_k > 20 s and joint masking not enabled then
        Enable motor failure simulation (Sec. IV-C)
    end if
    if L_k > 24 s and perturbations not enabled then
        Enable push perturbations (Sec. IV-E)
    end if
    Update policy π_θ using PPO
end for
```

E. Sim-to-Real Transfer

To bridge the gap between simulation and real-world deployment, both domain randomization and dynamics randomization techniques [13], [17], [31] are employed during training, as summarized in Appendix, Table IV.

1) Domain Randomization: Environmental and sensory variations are introduced to improve the policy's robustness against real-world uncertainties. Commanded velocities are randomized, horizontal push perturbations are randomly applied, and noise is injected into base velocity measurements to account for sensing and actuation inaccuracies.

2) Dynamics Randomization: Physical parameters of the robot, including motor constants, base mass, joint damping, inertia, and actuation delays, are randomized to bridge the modeling gap between simulation and reality. These randomization strategies enhance the policy's generalization capability, enabling robust performance when deployed on the real humanoid robot.

V. Experimental Results

A. Experimental Setup

The proposed policy is trained on the PPO algorithm with both the actor and critic networks modeled as two hidden layer MLPs consisting of 256 ReLU units per layer. The training is performed in the Isaac Gym simulator with 4096

TABLE II: **Success rate across failure scenarios in simulation.** Locomotion success rates are shown for the baseline, our method with joint masking and status estimation, and the full approach with curriculum learning with the fallibility reward r_{fall}. **Bold** indicates the best performance while underlined values denote no successful trials were recorded.

	Scenarios	Baseline [16]	+ joint mask. and status est.	+ curriculum and r_{fall} (Ours)
	Healthy	**0.9893**	0.5237	0.9624
Joint locking	hip yaw	0.2378	0.7063	0.9194
	hip roll	0.0000	0.6956	0.7974
	hip pitch	0.0000	0.3406	0.7073
	knee pitch	0.1462	0.4856	0.8130
	ankle pitch	0.0000	0.2683	0.6440
	ankle roll	0.1150	0.5398	0.9951
	Average	0.0832	0.5060	**0.8127**
Power Loss	hip yaw	0.6187	0.8989	0.9753
	hip roll	0.0000	0.0000	0.0000
	hip pitch	0.0000	0.5647	0.5784
	knee pitch	0.0000	0.0000	0.0000
	ankle pitch	0.0000	0.5205	0.6189
	ankle roll	0.7173	0.7949	0.9873
	Average	0.2227	0.4632	**0.5267**

parallel environments at a simulation frequency of 500 Hz and a control rate of 250 Hz. The maximum episode is set to 32 seconds. At each training iteration, a batch of 16,384 samples is collected, and the policy parameters are updated with a mini-batch size of 128 and a learning rate that linearly decays from 10^{-5} to 3×10^{-6}.

B. Performance of Simulated Failure Scenarios

Table II reports the locomotion success rates across different motor failure scenarios in the Isaac Gym simulation. These rates are calculated by counting the number of successful episodes in 4096 parallel environments, where a successful episode is defined as one that lasts for at least 20 seconds. We compare three methods: The baseline policy [16], the policy trained with joint masking and status estimation, and our complete approach with the curriculum learning and the proposed fallibility reward. The results demonstrate the enhanced performance of our method. Under joint locking conditions, the baseline policy fails to maintain stable locomotion in most cases with multiple scenarios showing zero success rates. Incorporating joint masking and status estimation significantly improves the performance. The success rate remains limited in challenging cases like knee pitch and ankle pitch power loss failures. Our full approach achieves the highest average success rates under both joint locking (81.27%) and power loss (52.67%) conditions. The staged curriculum learning allows the policy to first acquire

TABLE III: **Ablation study results for velocity tracking. Bold** indicated the best performance, while underlined values denote the worst performance among the compared methods.

	Conditions	Lin Vel [m/s] RMSE *(MBE)*	Ang Vel [rad/s] RMSE *(MBE)*
Ablation	w/o Joint status observation	0.1795 *(-0.0512)*	0.2074 *(-0.0831)*
	w/o Fallibility rewards	0.1529 *(-0.0450)*	0.1911 *(-0.0799)*
	w/o Phase modulation	0.2190 *(-0.1585)*	0.1499 *(-0.0614)*
	w/o Curriculum learning	0.2017 *(-0.0411)*	0.1320 *(-0.0559)*
	Ours (TOLEBI)	**0.0833** ***(-0.0346)***	**0.1110** ***(-0.0016)***

nominal locomotion skills prior to progressively handling motor failures and external perturbations while the fallibility reward encourages the robust recovery during fault-tolerance actuation. These results confirm that both components are essential for fault-tolerant locomotion across diverse failure scenarios.

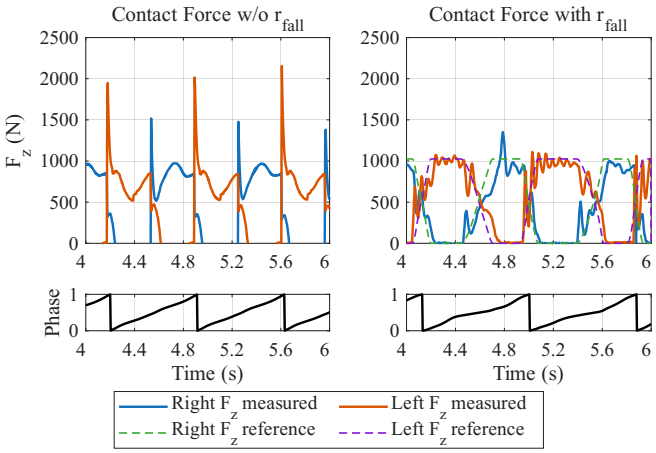

Fig. 3: **Effect of the fallibility rewards.** The reward mitigates early-contact impacts under motor failures and reduces impulsive forces that can reach up to 2000 N on the 100 kg robot TOCABI in real-world experiments.

C. Ablation Study

This section evaluates TOLEBI's key components: Joint status observation, fallibility rewards, phase modulation and curriculum learning. Table III shows the performance metrics Root Mean Square Error (RMSE) and Mean Bias Error (MBE) for both linear and angular velocity tracking averaged

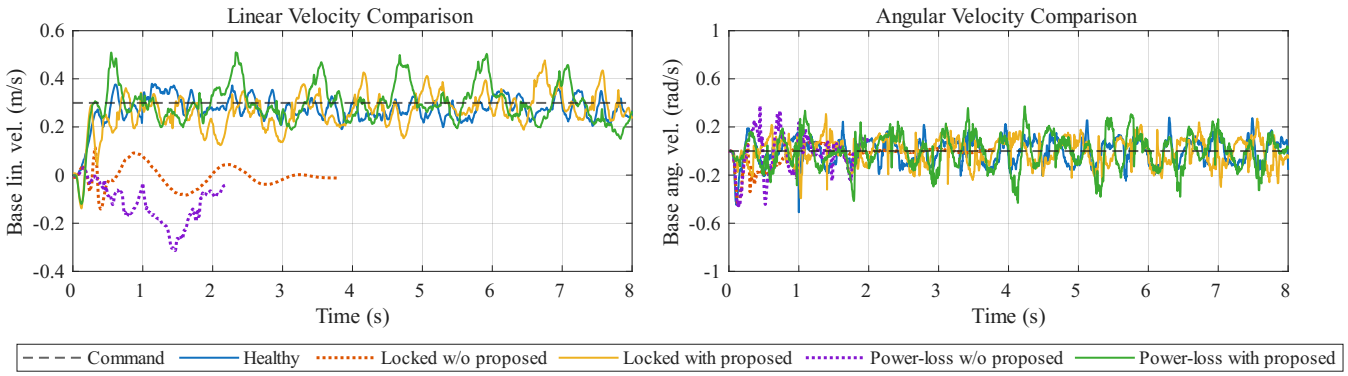

Fig. 4: **Comparison of linear and angular velocity tracking performance in real-world experiments under different motor failure scenarios.** The plots show base linear velocity (left) and base angular velocity (right) over time for the commanded velocity, healthy case, and motor failure cases (locked joint, power loss) with and without the proposed method. The results demonstrate that the proposed approach maintains stable velocity tracking even under motor failures.

over motor failure scenarios. Removing the joint status observation led to a remarkable performance degradation with particularly low performance under healthy conditions. This indicates that estimating the joint status online is essential for accurate and fault-tolerant locomotion control. Excluding the fallibility rewards also reduced robustness under motor failures as the policy failed to effectively promote safe behaviors, where the contact force reward specifically mitigates early-contact impacts that otherwise destabilize locomotion as shown in Fig. 3. Similarly, removing the phase modulation resulted in the highest MBE among all ablation settings since the absence of phase modulation failed to shorten the stance duration of the impaired leg. This eliminated the controller's ability to adapt the gait timing. Furthermore, training in the absence of curriculum learning prevented the policy from learning nominal locomotion that resulted in gaits where the robot failed to properly lift its feet even under healthy conditions. The complete model of TOLEBI with all components achieved the lowest tracking errors that highlights their benefits for robust fault-tolerant biped locomotion.

D. Sim-to-Real Validation Experiments

1) Walking Straight Experiments: To evaluate the robustness of TOLEBI in real-world settings, flat-ground walking straight experiments were conducted under both healthy and motor failure conditions. As shown in Fig. 4, the robot was commanded to walk straight with a forward velocity of $v_x = 0.3m/s, v_y = 0, w_z = 0$. The comparison between simulation and real-world trials demonstrates that TOLEBI maintains stable in linear and angular velocity tracking under joint locking and power loss scenarios. These demonstrate TOLEBI's capability to handle faults.

2) Stair Descent Experiments: In addition to flat-ground walking, stair descent tasks with 9 cm steps were conducted to further evaluate the robustness of TOLEBI under motor failures. Stair environments expose humanoid robots to frequent risks of external impacts or internal circuit faults, making motor failures more likely during descent. To assess TOLEBI's ability to handle such challenging scenarios, the humanoid robot TOCABI was tested in both MuJoCo simulation and real-world experiments. As illustrated in Fig. 5, the robot successfully descended stairs in the presence of joint locking or power loss conditions. Notably, no additional terrain-specific curriculum learning for stair descending was employed, yet the policy generalized its learned skills to this unseen task. These results demonstrate TOLEBI's robustness in maintaining balance and stable locomotion despite unexpected motor failures, confirming its practical applicability for fault-tolerant humanoid locomotion.

VI. Conclusion

This paper presented TOLEBI, a fault-tolerant reinforcement learning framework for biped locomotion. TOLEBI integrates joint masking for diverse failure scenarios, an estimator for joint status inference, and a curriculum learning strategy that gradually exposes the policy to nominal walking, motor failures, and external disturbances. Demonstrating successful sim-to-real transfer, the learned policy enabled the humanoid robot TOCABI to maintain stable locomotion under motor failures as joint locking and power loss. Moreover,

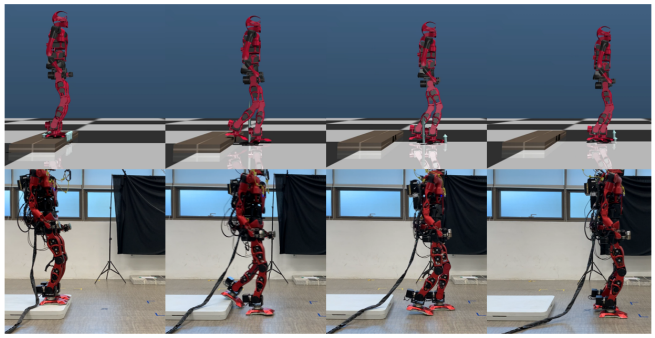

Fig. 5: **Validation of stair descent under motor failure conditions.** TOLEBI enables the humanoid robot TOCABI to successfully perform stair descent in both MuJoCo simulation and real-world experiments.

the learned policy achieved straightforward walking and stair descent without additional terrain-specific training. Future work will focus on handling multiple simultaneous failures and robust locomotion in unstructured environments toward resilient locomotion strategies.

APPENDIX

A. Randomization Parameters

TABLE IV: Randomization parameters.

	Parameter	Randomization Range
Domain Randomization	Command Lin. Velocity	$v_x \in U[-0.3, 0.6]\ m/s$, $v_y \in U[-0.3, 0.3]\ m/s$
	Command Ang. Velocity	$\omega_z \in U[-0.5, 0.5]\ rad/s$
	Push Perturbation	Force $F \in U[50, 250]\ N$, Time $T \in U[0.1, 1]\ s$
	Base Velocity Noise	Lin vel $v \in U[-0.025, 0.025]\ m/s$, Ang vel $w \in U[-0.02, 0.02]\ rad/s$
Dynamics Randomization	Link Mass	$U[0.6, 1.4]\ kg$
	Link Inertia	$U[0.6, 1.4]\ kg * m^2$
	Link Center of mass	$U[0.6, 1.4]\ m$
	Motor Constants	$U[0.9, 1.1]$
	Joint Friction	$U[0.6, 1.4]\ Nm$
	Joint Damping	$U[0.6, 1.4]\ Nm * s/rad$
	Actuation Delay	$U[0.5, 1.5]\ ms$

REFERENCES

[1] Q. Zhang, P. Cui, D. Yan, J. Sun, Y. Duan, G. Han, W. Zhao, W. Zhang, Y. Guo, A. Zhang *et al.*, "Whole-body humanoid robot locomotion with human reference," in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2024.

[2] X. Zhang, X. Wang, L. Zhang, G. Guo, X. Shen, and W. Zhang, "Achieving stable high-speed locomotion for humanoid robots with deep reinforcement learning," *arXiv preprint arXiv:2409.16611*, 2024.

[3] S. Schelter, T. Rukat, and F. Bießmann, "Learning to validate the predictions of black box classifiers on unseen data," in *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*, 2020, pp. 1289–1299.

[4] S. Redyuk, S. Schelter, T. Rukat, V. Markl, and F. Biessmann, "Learning to validate the predictions of black box machine learning models on unseen data," in *Proceedings of the Workshop on Human-In-the-Loop Data Analytics*, 2019, pp. 1–4.

[5] Y. Tong, H. Liu, and Z. Zhang, "Advancements in humanoid robots: A comprehensive review and future prospects," *IEEE/CAA Journal of Automatica Sinica*, vol. 11, no. 2, pp. 301–328, 2024.

[6] A. F. Winfield and J. Nembrini, "Safety in numbers: fault-tolerance in robot swarms," *International Journal of Modelling, Identification and Control*, vol. 1, no. 1, pp. 30–37, 2006.

[7] M. Huber, *A hybrid architecture for adaptive robot control*. University of Massachusetts Amherst, 2000.

[8] J. Cui, Z. Li, J. Qiu, and T. Li, "Fault-tolerant motion planning and generation of quadruped robots synthesised by posture optimization and whole body control," *Complex & Intelligent Systems*, vol. 8, no. 4, pp. 2991–3003, 2022.

[9] M. Schwartz, J. Sim, J. Ahn, S. Hwang, Y. Lee, and J. Park, "Design of the humanoid robot tocabi," in *2022 IEEE-RAS 21st International Conference on Humanoid Robots (Humanoids)*. IEEE, 2022, pp. 322–329.

[10] J. Lee, J. Hwangbo, and M. Hutter, "Robust recovery controller for a quadrupedal robot using deep reinforcement learning," *arXiv preprint arXiv:1901.07517*, 2019.

[11] Z. Li, X. Cheng, X. B. Peng, P. Abbeel, S. Levine, G. Berseth, and K. Sreenath, "Reinforcement learning for robust parameterized locomotion control of bipedal robots," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021.

[12] Z. Li, X. B. Peng, P. Abbeel, S. Levine, G. Berseth, and K. Sreenath, "Reinforcement learning for versatile, dynamic, and robust bipedal locomotion control," *The International Journal of Robotics Research*, vol. 44, no. 5, pp. 840–888, 2025.

[13] G. B. Margolis and P. Agrawal, "Walk these ways: Tuning robot control for generalization with multiplicity of behavior," in *Conference on Robot Learning*. PMLR, 2023, pp. 22–31.

[14] I. Radosavovic, T. Xiao, B. Zhang, T. Darrell, J. Malik, and K. Sreenath, "Real-world humanoid locomotion with reinforcement learning," *Science Robotics*, vol. 9, no. 89, p. eadi9579, 2024.

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

[16] D. Kim, G. Berseth, M. Schwartz, and J. Park, "Torque-based deep reinforcement learning for task-and-robot agnostic learning on bipedal robots using sim-to-real transfer," *IEEE Robotics and Automation Letters*, 2023.

[17] D. Kim, H. Lee, J. Cha, and J. Park, "Bridging the reality gap: Analyzing sim-to-real transfer techniques for reinforcement learning in humanoid bipedal locomotion," *IEEE Robotics & Automation Magazine*, 2024.

[18] J.-M. Yang, "Kinematic constraints on fault-tolerant gaits for a locked joint failure," *Journal of Intelligent and Robotic Systems*, 2006.

[19] C. Pana, I. Resceanu, and D. Patrascu, "Fault-tolerant gaits of quadruped robot on a constant-slope terrain," in *2008 IEEE International Conference on Automation, Quality and Testing, Robotics*, vol. 1. IEEE, 2008, pp. 222–226.

[20] U. Asif, "Improving the navigability of a hexapod robot using a fault-tolerant adaptive gait," *International Journal of Advanced Robotic Systems*, vol. 9, no. 2, p. 34, 2012.

[21] J.-M. Yang and J.-H. Kim, "Fault-tolerant locomotion of the hexapod robot," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 28, no. 1, pp. 109–116, 1998.

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

[23] W. Okamoto, H. Kera, and K. Kawamoto, "Reinforcement learning with adaptive curriculum dynamics randomization for fault-tolerant robot control," *arXiv preprint arXiv:2111.10005*, 2021.

[24] Z. Luo, E. Xiao, and P. Lu, "Ft-net: Learning failure recovery and fault-tolerant locomotion for quadruped robots," *IEEE Robotics and Automation Letters*, vol. 8, no. 12, pp. 8414–8421, 2023.

[25] X. Wu, W. Dong, H. Lai, Y. Yu, and Y. Wen, "Adaptive control strategy for quadruped robots in actuator degradation scenarios," in *Proceedings of the Fifth International Conference on Distributed Artificial Intelligence*, 2023, pp. 1–13.

[26] K. Liu, Z. Wang, B. Li, L. Zhu, and H. Ding, "Fault joint detection and adaptive fault-tolerant control of legged robots under joint partial failures," *IEEE Robotics and Automation Letters*, 2025.

[27] M. Kim, U. Shin, and J.-Y. Kim, "Learning quadrupedal locomotion with impaired joints using random joint masking," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 9751–9757.

[28] T. Hou, J. Tu, X. Gao, Z. Dong, P. Zhai, and L. Zhang, "Multi-task learning of active fault-tolerant controller for leg failures in quadruped robots," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 9758–9764.

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

[30] S. Wang, C. Zhao, L. Qian, and X. Luo, "Learning fault-tolerant quadruped locomotion with unknown motor failure using reliability reward," in *International Conference on Intelligent Robotics and Applications*. Springer, 2024, pp. 129–143.

[31] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, "Domain randomization for transferring deep neural networks from simulation to the real world," in *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2017, pp. 23–30.