

Synthesize Efficient Safety Certificates for Learning-Based Safe Control using Magnitude Regularization

Haotian Zheng¹, Haitong Ma², Sifa Zheng¹, Shengbo Eben Li¹, Jianqiang Wang^{1*}

Abstract—Safety certificates based on energy functions can provide demonstrable safety for complex robotic systems. However, all recent studies on learning-based energy function synthesis only consider the feasibility of the control policy, which might cause over-conservativeness and even fail to achieve the control goal. To solve the problem of over-conservative controllers, we proposed the magnitude regularization technique to improve the controller performance of safe controllers by reducing the conservativeness inside the energy function, while keeping the promising provable safety guarantees. Specifically, we quantify the conservativeness by the magnitude of the energy function, and we reduce the conservativeness by adding a magnitude regularization term to the synthesis loss. We propose an algorithm using reinforcement learning (RL) for synthesis to unify the learning process of safe controllers and energy functions. We conducted simulation experiments on Safety Gym and real-robot experiments using small quadrotors. Simulation results show that the proposed algorithm does reduce the conservativeness of the energy function and outperforms baselines in terms of controller performance while maintaining safety. Real-robot experiments have shown that the proposed algorithm indeed reduce conservativeness on the small quadrotors.

I. INTRODUCTION

Safety is one of the most important factors in the real-world application of robots, where the robots should always obey state constraints. For example, autonomous vehicles should not collide with pedestrians, and industrial robots should not hit the collaborating human workers. However, safety always has conflict with the controller performance. Simply emphasizing safety might cause over-conservativeness. For instance, the autonomous vehicle might be too conservative so that it only stays still, industrial robotic arms can stay still for extended periods due to an overly conservative strategy, impacting overall productivity.

A major branch of robot safety, or safe control studies is the energy-function-based safety certificates. Existing studies includes safety index [1]–[3], barrier certificates or control barrier functions (CBF) [4]–[6], and reachability analysis [7]–[9]. Intuitively, the energy functions mean that the dangerous states should be assigned high energy, and the safe states have low energy. Then the safe control policies are designed to dissipate the system energy [10]. The most promising point of energy-function-based safety certificates is that they can provide *provable safety guarantees* by ensuring forward invariance in the safe sets [3]. Forward invariance indicates that the system will never leave the safety set if there always exist actions to dissipate the system

energy. The provable safety guarantees are favorable in both algorithmic design and real-world robotic applications.

However, it is extremely difficult to synthesize a feasible energy function by hand [3] (Provable safety guarantees are valid only with the feasibility). The difficulties have stimulated many recent studies using learning-based techniques to synthesize energy functions [5], [11]–[20]. RL has gained increasing attention since it learns from environment interactions and does not need prior controllers or dynamics. Some recent studies have shown that RL can synthesize the safety certificates while learning the safe control policies [19], [20].

Nevertheless, safety usually conflicts with the controller performance. For example, if the energy function enforces that the robot arm always stays still, it will not encounter any danger but will not accomplish any tasks. We define an efficient energy function as it generates an efficient safe control policy. Intuitively, an ideal energy function is both feasible and efficient in terms of the controller performance. Unfortunately, few learning-based studies have discussed the controller performance when synthesizing the energy function.

Therefore, in this paper, we proposed the *magnitude regularization* method for learning-based energy function synthesis. We quantify the conservativeness of the energy function by the magnitude of the energy function. We add a regularization term in the synthesis loss, the energy function should not be unnecessarily high for dangerous states. The magnitude regularization term is applicable to any learning-based synthesis methods. Then we proposed SafeMR based on RL to synthesize the safe controller. We conduct experiments on SafetyGym, a commonly used safe RL benchmark. Results show that the *magnitude regularization* method effectively improves policy efficiency while guaranteeing safety. Due to page limit, interested readers are referred to an extended manuscript [21] to explore about the application of this method in the field of transportation.

II. RELATED WORKS

A. Energy-Function-Based Safety Certificates

Representative energy-function-based safety certificates include control barrier functions (CBF) [1], barrier certificates [22], and the safety set algorithm (SSA) [3]. Safety certificates and safe control policies are closely related and both play a significant role in guaranteeing the safety of dynamic systems. Recent research on learning-based approaches can be categorized into three main categories based on their learning objectives: (1) learning to synthesize energy functions with known dynamic models or controllers [11], [14], [23]–[25]; (2) learning safe control policies using

¹H. Zheng, S. Zheng, S. E. Li and J. Wang are with School of Vehicle and Mobility, Tsinghua University, Beijing, China, 100084. ²H. Ma is with Harvard John A. Paulson School of Engineering and Applied Sciences, Cambridge, MA, United States, 02138. The work was done during H. Ma's master study at Tsinghua University.*All correspondences go to J. Wang.

known feasible energy functions [15], [17], [18]; and (3) joint synthesis of safe control policies and energy functions [5], [19].

However, in all of these three branches of related studies, only feasibility is considered in the learning objectives or the prior knowledge. Few studies have discussed the efficiency in synthesizing the energy function. To the best of our knowledge, the only related studies suggest that some reachability analyses have explored the idea that states situated on the boundary of safe sets exhibit the most conservative actions. [20], [26], [27]. However, the reachability-based techniques only focus on the purely safe policy without explicitly learning an efficient and safe policy.

B. Safety in Reinforcement Learning

Safety has always been a significant concern in decision-making problems, particularly in reinforcement learning (RL), which involves learning through interactions with the environment. Various branches of safety-related RL studies have emerged, including risk-sensitive RL [28], [29], constrained Markov decision process (CMDP) [30]–[35], post-processing of RL policy output [17], [36], [37], and safe exploration in MDP [38], [39]. However, previous studies have encountered difficulties in handling state-dependent constraints, as the constraints for actions to dissipate are state-dependent [40]. While post-processing methods can handle state-dependent constraints to some extent, they require additional known information. To address this issue, a Lagrangian-based method with state-dependent multipliers [20], [41] was proposed to explicitly handle state-dependent constraints in RL. The proposed method in this paper also builds upon this approach.

III. PRELIMINARIES

In this section, we introduce the preliminaries about the problem formulation and energy-function-based safety certificates.

A. Safety Specifications and Problem Formulation

In this paper, safety means that the system state s should be bounded in a connected closed set \mathcal{S}_s , which is called the safe set. \mathcal{S}_s can also be represented by a zero-sublevel set of a safety exponential function $\phi_0(\cdot)$, $\mathcal{S}_s = \{s | \phi_0(s) \leq 0\}$. We use the Markov Decision Process (MDP) with deterministic dynamics (a reasonable assumption when dealing with robot safety control problems) defined by the tuple $(\mathcal{S}, \mathcal{A}, \mathcal{F}, r, c, \gamma)$, where \mathcal{S}, \mathcal{A} is the state and action space, and $\mathcal{F} : \mathcal{S} \times \mathcal{A} \rightarrow \mathcal{S}$ is the unknown system dynamics, $r, c : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow \mathbb{R}$ is the reward and cost functions, γ is the discount factor.

B. Energy-Function-Based Safety Certificates

We define the energy function by $\phi : \mathcal{S} \rightarrow \mathbb{R}$. Intuitively, the system should be assigned high energy if the system state is dangerous, for example, the robot arm is close to the human operator. If the system state is safe, then the system should be assigned low energy. For the provable safety guarantee, two conditions should be satisfied: (1) the system should keep at low energy, ($\phi \leq 0$), and (2) the system should rapidly dissipate energy when the system is at

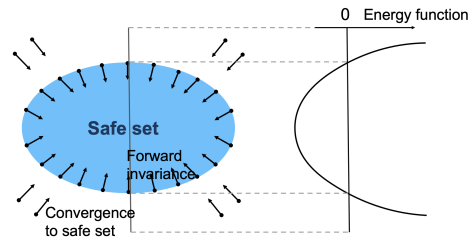


Fig. 1. Demonstration of energy function and the relationship with the safe set.

high energy ($\phi > 0$). Figure 1 demonstrates the relationship between energy function and safe set, and the two conditions. Therefore, we can get the safety constraint for

$$\phi(s') < \max\{\phi(s) - \eta_D, 0\} \quad (1)$$

where η_D is a slack variable controlling the descent rate of energy function. For simplicity, we use s' to represent the next state. If there always exists an action $a \in \mathcal{A}$ satisfying (1) at s , or the safe action set $\mathcal{U}_s(s) = \{a | \phi(s') < \max\{\phi(s) - \eta_D, 0\}\}$ is always nonempty, we say the energy function ϕ is *feasible*. Only satisfying the constraint (1) of a feasible certificate can guarantee safety. Otherwise, there might be no action to guarantee safety at some specific states, then there is no safety guarantee. Recent energy function synthesis studies all focus only on the feasibility of the energy function [13], [14], [19].

Indeed, feasibility is important, and synthesizing a feasible energy function is already challenging. However, focusing solely on feasibility is not enough. We must also consider efficiency when dealing with real-world robotics applications. For example, an autonomous vehicle should not come to a complete stop on a narrow road where both sides indicate danger (as shown in Figure 2), and robot arms should aim to improve their efficiency while ensuring safety. The design of the energy function ϕ undoubtedly influences the efficiency and policy performance.

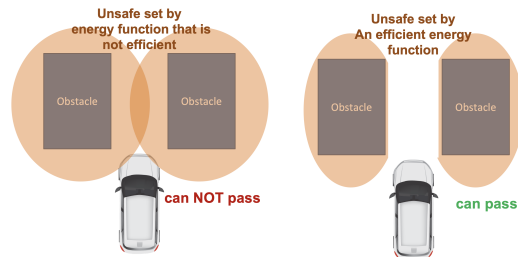


Fig. 2. Comparison between energy functions that are not efficient and efficient. energy function must be efficient for cases like passing a narrow road.

IV. ENERGY FUNCTION SYNTHESIS AND MAGNITUDE REGULARIZATION

A. Energy Function Synthesis using Constrained Reinforcement Learning

We present a formulation for the Constrained Reinforcement Learning (CRL) problem, aimed at synthesizing feasible and efficient energy functions. RL is chosen as the learning framework because it allows us to learn both the safe control policy and energy functions without any prior knowledge. Unlike supervised learning techniques, which would require a known safe control policy or dynamics model, RL can handle complex real-world tasks where obtaining such knowledge is infeasible.

The CRL problem is structured to maximize the expected return while adhering to the energy constraints (1):

$$\begin{aligned} \max_{\pi} \mathbb{E}_{\tau \sim \pi} \left\{ \sum_{t=0}^{\infty} \gamma^t r_t \right\} &= \mathbb{E}_s \{ V^{\pi}(s) \} \\ \text{s.t. } \phi(s') - \max\{\phi(s) - \eta_D, 0\} &< 0, \forall s \in \mathcal{S} \end{aligned} \quad (2)$$

where $V^{\pi}(s)$ is the state-value function of s . Notably, the constraints (1) are posed on every states, rather than only the safe states.

To address this specific problem formulation, we adopt a Lagrangian-based constraint RL algorithm, as proposed in [41]. This algorithm leverages a Lagrange multiplier network $\lambda(s)$ to handle the state-dependent constraints. Following their approach, we define the Lagrange function. In the subsequent section, we will elaborate on incorporating the magnitude regularization into this Lagrange function.

$$\mathcal{L}(\pi, \lambda) = \mathbb{E}_s \left\{ -V^{\pi}(s) + \lambda(s) (\phi(s') - \max\{\phi(s) - \eta_D, 0\}) \right\} \quad (3)$$

As the Lagrange function is the loss function of RL, or the policy learning, we name it the *original loss function*. The original loss function is able to jointly learn the safe control policy and energy functions [19]¹. However, the original loss function only considers feasibility and does not consider efficiency.

B. Efficiency Consideration by Magnitude Regularization

We will introduce how to implement the magnitude regularization based on the original loss function. Recall that in Section I, we briefly introduce the difficulties of designing loss function for efficiency. In the reinforcement learning algorithm 2, one may question that the expected return is a straightforward choice of efficiency measure and there is no need to introduce others. However, (2) is a constrained RL problem. The expected return is only in the *objective function*, and the energy function will only change the *constraints*. What we want to optimize is a *better constraint*, where the objective function has no gradient w.r.t. the constraints².

¹Normally, a safe RL algorithm can only learn safe control policies. However, an outer iteration cycle was added in [19] to learn the energy functions.

²Notably, we have two major learning objectives here, the policy and the energy function. The objective function has gradients w.r.t. policy but is irrelevant to the energy function.

To deal with this problem, we create a method called magnitude regularization. The motivation behind our method is that, the higher the energy function is, the more conservative the energy constraint (1) becomes. The motivation directly originates from the basic motivation of the energy function, the lower, the safer. In other words, we want the energy function to be necessarily high (to guarantee feasibility) but not too high (to improve performance). It is easy to understand we can add a regularization term to the energy function synthesis loss (3) so that the energy function will not increase too much. We name it the regularization term the *magnitude regularization*.

To further explain how to implement the magnitude regularization, we take an example of the energy functions tuned for the collision avoidance tasks in [25]. Collision avoidance is a commonly seen safety requirement, and the following methods could be easily migrated to other safety requirements.

$$\phi(s) = (\sigma + d_{\min})^n - d^n - k\dot{d} \quad (4)$$

where d is the distance between the robots and the obstacles to avoid, d_{\min} is the minimum safe distance to the obstacle. The \dot{d} is the derivative of distance with respect to time, $\xi = [\sigma, k, n]$ are the tunable parameters we desire to optimize in the synthesis algorithm. They should all be positive real numbers.

Given a specific energy function formulation, we can analyze how the change of parameters will affect the magnitude of the energy function. Since d_{\min} is a constant, it is clear that if σ and n increase, the ϕ will increase. The situation is a little bit tricky for the parameter k since the correlation between energy function and k depends on \dot{d} . We can only consider the case that the robot is approaching obstacles that are dangerous. In these cases, ϕ always increases with positive k .

Based on the analysis above, it is clear that the tunable parameters $\xi = [\sigma, k, n]$ affects the loss function indirectly by changing the size of ϕ . This makes it difficult to adjust the loss function directly, so we thought of taking some of the tunable parameters out of phi and adding them directly to the original loss function $\mathcal{L}(\pi, \lambda)$ to make the effect of the tunable parameters on the loss function more direct.

Furthermore, \dot{d} holds significant physical significance, representing the relative velocity of the agent concerning obstacles (positive when the agent is moving away). Considering that the agent's velocity does not directly impact safety as distance does, as a means to reduce the excessively high loss function, \dot{d} is extracted as an independent parameter. By multiplying it by a coefficient and adding it to the energy function, it serves as a method to avoid the energy function from becoming too high.

After a lot of experimentation, we found that by adding two parameters from $\phi(s)$, $(\sigma + d_{\min})^n$ and \dot{d} , to the loss function $\mathcal{L}(\pi, \lambda)$, and then introducing two new adjustable parameters a and b to set their weights, we can adjust the size of the safety set and increase the efficiency of the robot while ensuring safety.

Overall, we conclude that the magnitude of ϕ increase with σ, n, k . Therefore, the specific magnitude regularization term

is designed as

$$a(\sigma + d_{\min})^n + b\dot{d} \quad (5)$$

where a and b are two positive parameters. We will analyze the sensitiveness of a, b in the experimental results section.

Eventually, adding the magnitude regularization term to the original loss function (3), the final loss function is

$$\begin{aligned} \mathcal{L}'(\pi, \lambda, \phi) = & \mathbb{E}_s \{ -V^\pi(s) + \lambda(s)(\phi(s') - \\ & \max\{\phi(s) - \eta_D, 0\}) \} \\ & + a(\sigma + d_{\min})^n + b\dot{d} \end{aligned} \quad (6)$$

V. ENERGY FUNCTION SYNTHESIS USING CONSTRAINED REINFORCEMENT LEARNING

In this section, we introduce the practical algorithm we used for synthesizing the energy function, including the practical algorithm, gradient computation and discussion about the convergence.

A. Details of Algorithm

We construct our CRL algorithm based on the actor-critic framework [42]. Generally speaking, our algorithm is designed to be a multi-timescale learning process. The multi-timescale means that we simultaneously train different parameters, and some converge faster than others. In our specific algorithm, the fastest timescale is the value function (or Q-function) learning, the second fast one is the policy. These two timescales are also what the actor-critic algorithm did. After that, the multiplier network is updated. The multiplier network is proposed to handle the state-dependent constraints in CRL problems [41]. Finally, the energy function converges the most slowly.

We name the algorithm SafeMR since we focus on safe control problems with magnitude regularization (MR). This algorithm denotes the parameters of the policy network, multiplier network, and energy function as θ, ξ, ζ , and the gradients to update policy, multiplier, and certificate by $G_\#, \# \in \{\theta, \xi, \zeta\}$. In addition, the assigns multiple delayed updates, $m_\pi < m_\lambda < m_\phi$, for stable multi-timescale optimization. Due to the space limitation, we only presented the policy improvement part of SafeMR in Algorithm 1.

Algorithm 1: Policy Improvement in SafeMR

Require: Buffer \mathcal{D} with sampled data, policy parameters θ , multiplier parameters ξ , energy function parameters ζ .
Update the policy
if gradient steps mod $m_\pi = 0$ **then** $\theta \leftarrow \theta - \overline{\beta}_\pi G_\theta$
Update the multipliers
if gradient steps mod $m_\lambda = 0$ **then** $\xi \leftarrow \xi + \overline{\beta}_\lambda G_\xi$
Update the energy functions
if gradient steps mod $m_\phi = 0$ **then** $\zeta \leftarrow \zeta - \overline{\beta}_\zeta G_\zeta$
Ensure: w_1, w_2, θ, ξ .

Notably, the loss function to update the policy and multiplier is the same which follows the nature of dual ascent algorithm [43]:

$$\begin{aligned} J_\pi(\theta) = & \mathbb{E}_{s_t \sim \mathcal{D}} \left\{ \mathbb{E}_{a_t \sim \pi_\theta} \left\{ \alpha \log(\pi_\theta(a_t | s_t)) \right. \right. \\ & \left. \left. - Q_w(s_t, a_t) + \lambda_\xi(s_t) Q_\phi(s_t, a_t) \right\} \right\} \end{aligned} \quad (7)$$

We omit the detailed gradient computation due to space limits. Similar computation could be found in [41]. The objective function for synthesizing the energy function parameter ζ is the loss function (6), the gradient of energy function parameters ζ is

$$G_\zeta = \nabla_\zeta \mathcal{L}'(\pi, \lambda, \phi) |_{\pi=\pi^*(\lambda, \phi), \lambda=\lambda^*(\phi)} \quad (8)$$

The subscripts means that ϕ, λ has already reached the locally optimal solutions w.r.t. current energy function ϕ since they converge faster than ϕ .

VI. SIMULATION EXPERIMENTS

In the experiments, we mainly focus on solving the following problems.

- 1) Does the proposed method reduce conservativeness of the energy function design?
- 2) If the answer to the first question is yes, does the conservativeness reduction result in efficiency improvement?
- 3) How does the proposed algorithm compare with other constraint RL algorithms? Can it achieve a safe policy with zero constraint violation?

We chose two experimental environments. One is pure collision avoidance or aircrafts [8] to verify the conservativeness reduction. The other is Safety Gym [32], a commonly used safety RL benchmark environment with different tasks and obstacles. Here we select four environments with different tasks and different obstacles for verifying the performance of the robot after CRL training.

A. Conservativeness Reduction

In this experiment, an airplane (the blue one in Figure 3) is controlled to avoid another airplane. Control input is the angular velocity and the state is the position and heading angle differences between two airplanes. Figure 3 shows the boundaries of learned *unsafe* sets. Notably, states *outside* the boundaries are safe. We compare SafeMR with JointSIS [19] and handcrafted energy function in [25]. The reward is designed to be the L2 norm of actions. Results show that the unsafe set learned by SafeMR is smaller than the JointSIS and also covers the real safe set (red boundary, numerical solution). The handcrafted unsafe set does not cover all the real unsafe sets, which means that there might exist no safe action in that uncovered area, which will result in danger. Therefore, the experimental results show that SafeMR indeed reduces conservativeness while guaranteeing safety, answering the first question proposed at the beginning of this section.

B. Safety Gym: Baseline Algorithms and Experiment Setup

In this section, three types of baseline algorithms were compared with the proposed algorithm: (1) joint synthesis method of safe control policy and safety certificate without

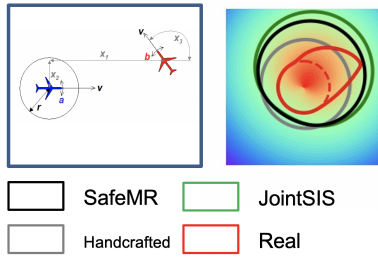


Fig. 3. Demonstration for the aircraft task and learned unsafe sets for the airplane task with different algorithms. SafeMR learns the smallest safe set covering the real safe set. Handcrafted energy function scales a smaller unsafe set but there are unsafe states in the safe set, so the unsafe set is not valid.

efficiency consideration [19], we name it JointSIS here for clearance; (2) constrained RL baselines. CRL baselines include PPO-Lagrangian, TRPO-Lagrangian and CPO [31], [32]. Notably, the safety specification in this paper, zero constraint violation, is different from those in the original CRL paper. Therefore, we set the cost threshold to be zero to make the CRL baseline head to solid safe policies (However, they will not learn the zero-constraint-violation policies as the following results show). (3) FAC with original energy function ϕ_0 and handcrafted energy function ϕ_h , where $\phi_0 = d_{min} - d$ and $\phi_h = (0.3 + d_{min})^2 - d^2 - kd$, named as *FAC with ϕ_0* and *FAC with ϕ_h* . The choice of ϕ_h is based on empirical knowledge.

The four experimental environments are shown in Figure 4 and named by Obstacles-Size-Tasks. The red robot's policy is to get to the green target area while avoiding overlap or collision with the blue obstacles. In two of the environments, there are no physical obstacles, only virtual obstacles. The other two environments have pillars as real physical obstacles. In addition, the size of the obstacles is different in the two environments with virtual obstacles and the two environments with real obstacles.

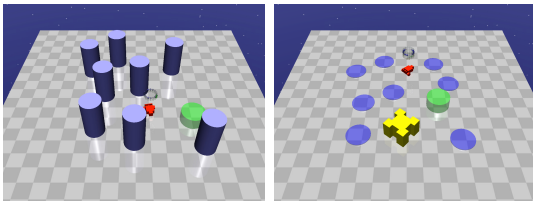


Fig. 4. Demonstration of experimental environments. Each environment also have two variants with different obstacle numbers and sizes.

C. Simulation Experimental Results

The results of the simulation experiment are shown in the figure 5. Table I compares the expected return between SafeMR and the baseline with highest expected return and zero violations. We present the expected return and the expected cost as the metric to evaluate the algorithm performance. Specifically, expected cost is counted by summing up the constraint violations (also called costs in some previous CRL paper) in a single episode and averaging them across the multiple runs.

Simulation experimental results show that SafeMR achieves the highest expected return among all safe algorithms. The baseline algorithms could be divided into two categories: (1) achieve higher expected return but are not safe, including all CMDP-based algorithms (TRPO-L, PPO-L, and CPO), and FAC- ϕ_0 ; (2) guarantee safety but have lower expected return, including the JointSIS (FAC- ϕ_h in some of the environments). According to the original Safety Gym paper [32], the metric to evaluate policy performance needs to consider both safety and performance. However, for the zero-violation safe control problem, the policy is meaningless if it can not guarantee safety. Therefore, we can conclude that the SafeMR improves policy efficiency while guaranteeing safety, compared to all baseline algorithms.

D. Microscopic and Sensitive Analysis

We give some microscopic and sensitive analysis for the hyper-parameters in the magnitude regularization, i.e., $[a, b]$. We first present the learned energy function parameters σ, k, n with different a, b in Table II. We only show the result in the Pillar-0.15-Goal environment due to space limitations. The feasibility of energy function parameters could be verified by Equation (3) in [25], and all three sets of learned energy functions are feasible. The training curves are shown in Figure 6. It shows that the expected return is related to the choices of $[a, b]$, but they all outperform the JointSIS which has no efficiency consideration. For safety, the expected costs all converge to zero with different hyper-parameters. Experimental results show that, for SafeMR, all the hyper-parameter choices lead to performance improvement while guaranteeing safety.

VII. REAL-ROBOT EXPERIMENTS

We conduct the experiments on designing the collision avoidance controller of Crazyflie quadrotors.

We conduct experiments on a collision avoidance task, where two quadrotors are heading towards each other and need to avoid each other. We compare our controller with the collision avoidance baseline JointSIS [19]. We analyze the controller performance by the mean squared error and the maximal absolute error between the quadrotor and reference trajectories. The goal is to track the straight line towards the target position, which is the initial position of the other quadrotor. For the controller implementation details, we use the idea of residual RL to integrate the learning-based controller and original on-board tracking controller based on PID. Residual RL means that the RL controller does not directly control the quadrotor. Instead, the control is the sum of a nominal controller output and the RL policy output. The NN of RL controller is a MLP with two hidden-layers with 256 neurons each and the activation function is RELU. We implement it with the onboard STM32F405 controller. We use the CrazySwarm [44] to control multiple quadrotors simultaneously.

Table III shows the experimental results. More results could be seen in the accompany video.

VIII. CONCLUSION

This paper proposed the magnitude regularization technique to synthesize efficient energy functions while guaran-

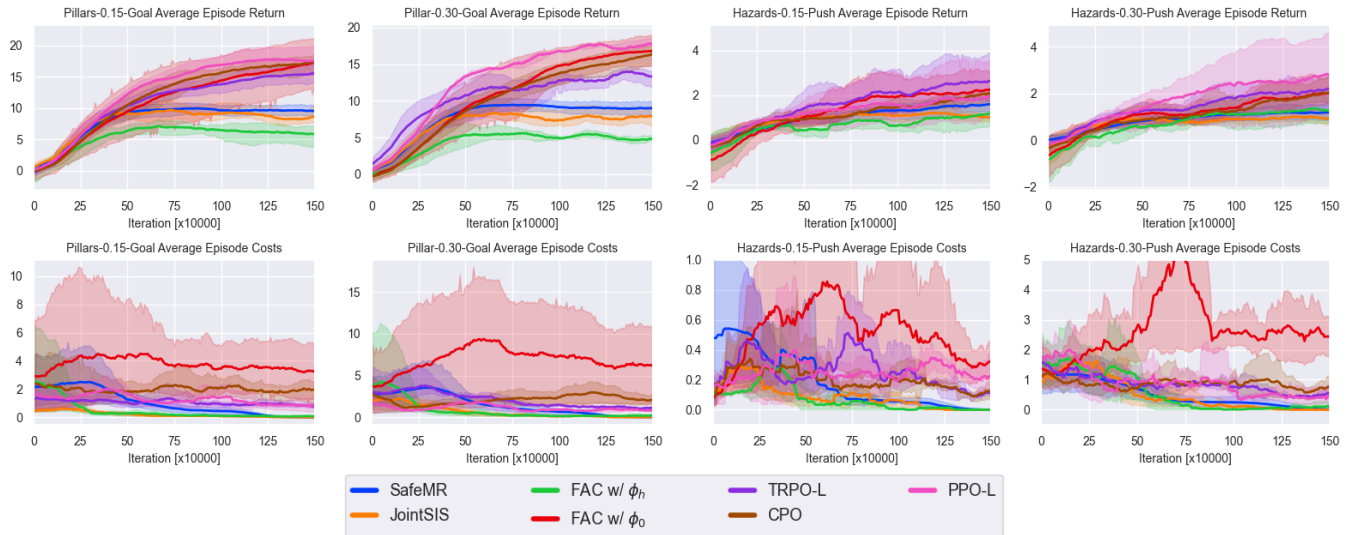


Fig. 5. Training curves of SafeMR and baseline methods on 4 different Safety Gym environments over five random seeds. Shaded regions are the 95 confidential intervals.

TABLE I
EFFICIENCY IMPROVEMENT OF SAFEMR COMPARED TO ALL SAFE BASELINES.

Expected Return	Pillars-0.15-Goal	Pillars-0.30-Goal	Hazards-0.15-Push	Hazards-0.30-Push
SafeMR	10.034	8.802	1.639	1.203
Highest among safe algorithms	9.286	6.921	1.020	0.871
Improvement	10.2%	27.2%	60.7%	38.1%

TABLE II
LEARNED ENERGY FUNCTION PARAMETERS WITH DIFFERENT $[a, b]$

Hyper-parameter $[a, b]$	Energy Function parameter
$[0.35, 0.15]$	$[0.201, 0.835, 2.084]$
$[0.45, 0.15]$	$[0.185, 0.753, 2.194]$
$[0.35, 0.25]$	$[0.223, 0.882, 1.984]$

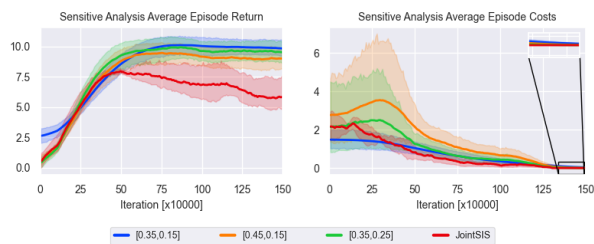


Fig. 6. Sensitive analysis of SafeMR. Policy efficiency is improved while safety is guaranteed for all hyper-parameters.

teering safety in robotic safe control tasks. We quantify the conservativeness by the magnitude of the energy function and construct a magnitude regularization term to control the magnitude growing during synthesis. An algorithm called SafeMR is proposed to combine magnitude regularization and RL and synthesize feasible and efficient energy functions. Experimental results on various tasks show that the proposed algorithm can reduce the conservativeness of the energy function and then improve the efficiency of the safe

TABLE III
MEAN SQUARED ERROR AND MAXIMAL ABSOLUTE ERROR IN REAL-ROBOT EXPERIMENTS.

	Mean squared error ($\times 10^{-2}$) (m)
JointSIS	1.349
SafeMR	0.834
Maximal absolute error	
JointSIS	0.303
SafeMR	0.231

control policies. Meanwhile, the algorithm solidly guarantees safety and is robust to hyper-parameter choices.

In future work, we will generalize the magnitude regularization to more complex energy function models, like the neural networks (NN). NN has potential to further remove all the conservativeness of energy function in Figure 3 but is also much more difficult to analyze.

IX. ACKNOWLEDGMENT

This study is supported by National Key R&D Program of China with 2020YFB1600202 and National Natural Science Foundation of China, the Key Project (52131201). This study is also supported by Tsinghua-Toyota Joint Research Fund.

REFERENCES

- [1] P. Wieland and F. Allgöwer, "Constructive safety using control barrier functions," *IFAC Proceedings Volumes*, vol. 40, no. 12, pp. 462–467, 2007.

- [2] A. D. Ames, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs with application to adaptive cruise control," in *53rd IEEE Conference on Decision and Control*. IEEE, 2014, pp. 6271–6278.
- [3] C. Liu and M. Tomizuka, "Control in a safe set: Addressing safety in human-robot interactions," in *Dynamic Systems and Control Conference*, vol. 46209. American Society of Mechanical Engineers, 2014, p. V003T42A003.
- [4] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs for safety critical systems," *IEEE Transactions on Automatic Control*, vol. 62, no. 8, pp. 3861–3876, 2016.
- [5] H. Ma, J. Chen, S. E. Li, Z. Lin, and S. Zheng, "Model-based constrained reinforcement learning using generalized control barrier function," *arXiv preprint arXiv:2103.01556*, 2021.
- [6] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada, "Control barrier functions: Theory and applications," in *2019 18th European Control Conference (ECC)*. Bochum, Germany: IEEE, 2019, pp. 3420–3431.
- [7] K.-C. Hsu, V. Rubies-Royo, C. J. Tomlin, and J. F. Fisac, "Safety and liveness guarantees through reach-avoid reinforcement learning," *arXiv preprint arXiv:2112.12288*, 2021.
- [8] I. Mitchell, A. Bayen, and C. Tomlin, "A time-dependent hamilton-jacobi formulation of reachable sets for continuous dynamic games," *IEEE Transactions on Automatic Control*, vol. 50, no. 7, pp. 947–957, 2005.
- [9] S. Bansal, M. Chen, S. Herbert, and C. J. Tomlin, "Hamilton-jacobi reachability: A brief overview and recent advances," in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*. IEEE, 2017, pp. 2242–2253.
- [10] T. Wei and C. Liu, "Safe control algorithms using energy functions: A unified framework, benchmark, and new directions," in *2019 IEEE 58th Conference on Decision and Control (CDC)*. IEEE, 2019, pp. 238–243.
- [11] Y.-C. Chang, N. Roohi, and S. Gao, "Neural lyapunov control," *arXiv preprint arXiv:2005.00611*, 2020.
- [12] M. Saveriano and D. Lee, "Learning barrier functions for constrained motion planning with dynamical systems," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 112–119.
- [13] M. Srinivasan, A. Dabholkar, S. Coogan, and P. A. Vela, "Synthesis of control barrier functions using a supervised machine learning approach," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 7139–7145.
- [14] Z. Qin, K. Zhang, Y. Chen, J. Chen, and C. Fan, "Learning safe multi-agent control with decentralized neural barrier certificates," *arXiv preprint arXiv:2101.05436*, 2021.
- [15] L. Wang, A. D. Ames, and M. Egerstedt, "Safety barrier certificates for collisions-free multirobot systems," *IEEE Transactions on Robotics*, vol. 33, no. 3, pp. 661–674, 2017.
- [16] A. Agrawal and K. Sreenath, "Discrete control barrier functions for safety-critical control of discrete systems with application to bipedal robot navigation," in *Robotics: Science and Systems*, 2017.
- [17] R. Cheng, G. Orosz, R. M. Murray, and J. W. Burdick, "End-to-end safe reinforcement learning through barrier functions for safety-critical continuous control tasks," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 3387–3395.
- [18] A. Taylor, A. Singletary, Y. Yue, and A. Ames, "Learning for safety-critical control with control barrier functions," in *Learning for Dynamics and Control*. PMLR, 2020, pp. 708–717.
- [19] H. Ma, C. Liu, S. E. Li, S. Zheng, and J. Chen, "Joint synthesis of safety certificate and safe control policy using constrained reinforcement learning," in *Proceedings of The 4th Annual Learning for Dynamics and Control Conference*, ser. Proceedings of Machine Learning Research, R. Firoozi, N. Mehr, E. Yel, R. Antonova, J. Bohg, M. Schwager, and M. Kochenderfer, Eds., vol. 168. PMLR, 23–24 Jun 2022, pp. 97–109.
- [20] D. Yu, H. Ma, S. Li, and J. Chen, "Reachability constrained reinforcement learning," in *Proceedings of the 39th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvari, G. Niu, and S. Sabato, Eds., vol. 162. PMLR, 17–23 Jul 2022, pp. 25 636–25 655. [Online]. Available: <https://proceedings.mlr.press/v162/yu22d.html>
- [21] H. Zheng, C. Chen, S. Li, S. Zheng, S. E. Li, Q. Xu, and J. Wang, "Learning-based safe control for robot and autonomous vehicle using efficient safety certificate," *IEEE Open Journal of Intelligent Transportation Systems*, vol. 4, pp. 419–430, 2023.
- [22] S. Prajna, A. Jadbabaie, and G. J. Pappas, "A framework for worst-case and stochastic safety verification using barrier certificates," *IEEE Transactions on Automatic Control*, vol. 52, no. 8, pp. 1415–1428, 2007.
- [23] Y. Luo and T. Ma, "Learning barrier certificates: Towards safe reinforcement learning with zero training-time violations," *arXiv preprint arXiv:2108.01846*, 2021.
- [24] W. Jin, Z. Wang, Z. Yang, and S. Mou, "Neural certificates for safe control policies," *arXiv preprint arXiv:2006.08465*, 2020.
- [25] W. Zhao, T. He, and C. Liu, "Model-free safe control for zero-violation reinforcement learning," in *5th Annual Conference on Robot Learning*, 2021. [Online]. Available: <https://openreview.net/forum?id=UGp6FDaxB0f>
- [26] J. F. Fisac, A. K. Akametalu, M. N. Zeilinger, S. Kaynama, J. Gullula, and C. J. Tomlin, "A general safety framework for learning-based control in uncertain robotic systems," *IEEE Transactions on Automatic Control*, vol. 64, no. 7, pp. 2737–2752, 2018.
- [27] J. F. Fisac, N. F. Lugovoy, V. Rubies-Royo, S. Ghosh, and C. J. Tomlin, "Bridging hamilton-jacobi safety analysis and reinforcement learning," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 8550–8556.
- [28] Y. Chow, M. Ghavamzadeh, L. Janson, and M. Pavone, "Risk-constrained reinforcement learning with percentile risk criteria," *The Journal of Machine Learning Research*, vol. 18, no. 1, pp. 6070–6120, 2017.
- [29] P. Geibel and F. Wyszotzki, "Risk-sensitive reinforcement learning applied to control under constraints," *Journal of Artificial Intelligence Research*, vol. 24, pp. 81–108, 2005.
- [30] E. Altman, *Constrained Markov decision processes*. CRC Press, 1999, vol. 7.
- [31] J. Achiam, D. Held, A. Tamar, and P. Abbeel, "Constrained policy optimization," in *International Conference on Machine Learning*. Sydney, Australia: PMLR, 2017, pp. 22–31.
- [32] A. Ray, J. Achiam, and D. Amodei, "Benchmarking safe exploration in deep reinforcement learning," *arXiv preprint arXiv:1910.01708*, 2019.
- [33] Y. Zhang, Q. Vuong, and K. Ross, "First order constrained optimization in policy space," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, Eds., vol. 33, Online, 2020, pp. 15 338–15 349.
- [34] T.-Y. Yang, J. Rosca, K. Narasimhan, and P. J. Ramadge, "Projection-based constrained policy optimization," *arXiv preprint arXiv:2010.03152*, 2020.
- [35] C. Tessler, D. J. Mankowitz, and S. Mannor, "Reward constrained policy optimization," in *International Conference on Learning Representations*, 2019. [Online]. Available: <https://openreview.net/forum?id=SkfrvsA9FX>
- [36] T.-H. Pham, G. De Magistris, and R. Tachibana, "Optlayer-practical constrained optimization for deep reinforcement learning in the real world," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. Brisbane, Australia: IEEE, 2018, pp. 6236–6243.
- [37] G. Dalal, K. Dvijotham, M. Večerik, T. Hester, C. Paduraru, and Y. Tassa, "Safe exploration in continuous action spaces," *arXiv preprint arXiv:1801.08757*, 2018.
- [38] T. M. Moldovan and P. Abbeel, "Safe exploration in markov decision processes," *arXiv preprint arXiv:1205.4810*, 2012.
- [39] M. Turchetta, F. Berkenkamp, and A. Krause, "Safe exploration in finite markov decision processes with gaussian processes," *Advances in Neural Information Processing Systems*, vol. 29, 2016.
- [40] H. Ma, C. Liu, S. E. Li, S. Zheng, W. Sun, and J. Chen, "Learn zero-constraint-violation policy in model-free constrained reinforcement learning," *arXiv preprint arXiv:2111.12953*, 2021.
- [41] H. Ma, Y. Guan, S. E. Li, X. Zhang, S. Zheng, and J. Chen, "Feasible actor-critic: Constrained reinforcement learning for ensuring statewise safety," *arXiv preprint arXiv:2105.10682*, 2021.
- [42] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *International Conference on Machine Learning*. Stockholm, Sweden: PMLR, 2018, pp. 1861–1870.
- [43] D. P. Bertsekas, "Nonlinear programming," *Journal of the Operational Research Society*, vol. 48, no. 3, pp. 334–334, 1997.
- [44] J. A. Preiss, W. Honig, G. S. Sukhatme, and N. Ayanian, "Crazyswarm: A large nano-quadcopter swarm," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 3299–3304.