# Model Free Safe Control for Reinforcement Learning in a Clustered Dynamic Environment

Guiliang Zheng*
The Robot Institute
Carnegie Mellon University
Pittsburgh, PA, USA
guilianz@andrew.cmu.edu

Minhao Yang
Faculty of Engineering
The University of Hongkong
Hongkong, China
u3609684@connect.hku.hk

Yuxuan Wu
Department of Mechanical, Materials and Manufacturing Engineering
University of Nottingham
Nottingham,UK
sqyyw3@nottingham.ac.uk

***Abstract*—**While reinforcement learning (RL) has been wildly used in continuous control tasks with impressive performance, there are two main challenges in RL development: continuously satisfying the safety constraints in a clustered dynamic environment and lacking the explicit analytical models of dynamic systems for typical safeguard algorithms in RL training. This paper proposed a model-free safe control strategy to safeguard the RL agent in a clustered dynamic environment. By extending and applying the adaptive momentum boundary approximating (AdamBA) method to monitor and modify the RL nominal controls in a clustered dynamic environment, experimental results have shown the better safeguard performance of the proposed algorithm than other safe RL methods. The proposed algorithm could be easily extended to other RL algorithm with black-box implicit analytical model of dynamic systems.

***Index Terms*—**Safe Control, Clustered Environment, Reinforcement Learning, Robotics

## I. INTRODUCTION

Recently, intelligent robot and self-driving car technologies have become a prevailing field of research and technological innovation, which have potential to profoundly change our daily life. Safety in terms of persistently satisfying the hard state constraints has always been a major challenge in aforementioned two fields. For example, when intelligent robotic arm collaboratively works with a worker in a factory, it is important to guarantee that the robotic arm do not accidentally hit or injure the worker; in the self-driving car scenario, vehicles should not crush into any of the surrounding obstacles. More safety concerns have been raised when the robot trying to find an optimal control policy using RL, where the ego robot needs to randomly explore the surrounding environment.

Safe control methods which can simultaneously protect the ego robot from potential hazard and effectively explore the environment have been extensively studied. The most widely used methods are energy function based methods proposed in [1, 2, 3, 4], in which an energy function was designed such that safe states are with low energy and then find the safe control set to make the system dissipate energy. Once the system deviates from the safe set, the control flow will pull states back to the safe states, making it as the region of attraction (ROA). By Lyapunov's theorem, the system will asymptotically converge to the safe states, and once the system enters the safe set, it will never leave. However, those methods require explicit knowledge of the robot dynamics, which are challenging to obtain in the majorities of the RL scenarios.

To ensure the zero-safety violation in a spares obstacle environment during the RL training, [5] has proposed an Implicit Safe Set Algorithm (ISSA), which synthesizes the safety index and subsequent safe control law only by querying a black-box dynamic function of the ego robot. The ISSA can be viewed as the combination of the vanilla Safe Set Algorithm (SSA) mentioned in [1] with certain parameterized constraints and a sample-efficient approximation algorithm called AdamBA. The method in [5] possesses the advantage of guaranteeing forward invariant to the safe set with only black-box non-analytical model about the robot dynamics, but it only has been proven to be successful theoretically and experimentally in a sparse obstacle environment while not in a clustered dynamics environment. In [6], an adaptive safe control algorithm has been proposed, which modifies the objective function of the SSA to adapt the algorithm in a clustered dynamic environment. However, in this algorithm, it is necessary to query knowledge about the robot dynamics and with the increasing random dynamic obstacles in the environment, the algorithm provides limited safeguard to the robot with 68.6 % success rate [6], which means collisions still happen frequently.

This paper extended the AdamBA proposed in [5] to a clustered dynamic environment and tested its capability and limitation in such environment. The improvement of AdamBA was made by changing the constraints of the safe control algorithm to adapt the modified AdamBA in the clustered dynamic environment. With the modified AdamBA, the algorithm could provide strong safety shield to the ego vehicle in a clustered dynamic obstacle environment where no white-box analytical vehicle model is available. The modified AdamBA algorithm can find safe control set in a restricted control space for a RL agent in a different RL setting. In this paper, it has been successfully tested in the RL testing environment in [6], where it achieved zero collision with 50 random moving obstacles in a confined environment.

The key contributions of this paper are summarized below:

- The vanilla AdamBA proposed in [5] was extended to a clustered dynamic environment, which proved that the vanilla AdamBA algorithm could provide good safety shield in a sparse obstacle dynamic environment but demonstrate poor performance on finding safe control set in a clustered obstacle dynamic environment.
- In order to achieve safe control in the clustered dynamic environment, this paper has proved that the change of the safe control constraints is more effective than the change of the safe control objective function.
- Based on the modified AdamBA algorithm, zero violation safe control for a mobile robot in a clustered dynamic environment with the black-box non-analytical robot dynamics model was achieved.

## II. PROBLEM DESCRIPTION

*Environment Dynamics:* The 2D environment contains multiple random moving obstacles, which evolves as $\dot{x}_E = f_E\ (x_E\ , u_E)$, where the function $f_E$ represents double integrator dynamic, state $x_E \subset \mathbb{R}^4$ involves the position and velocity of the obstacles and control $u_E \subset \mathbb{R}^2$ represents its acceleration, which is distributed on the predefined interval.

*Robot Dynamics:* Let $x_t \in X \subset \mathbb{R}^{n_x}$ be the robot state at time step $t$, where $n_x$ is the dimension of the state space $X$; $u_t \in U \subset \mathbb{R}^{n_u}$ be the control input to the robot at time step $t$, where $n_u$ is the dimension of the state space $U$. The system dynamics are defined as: $x_{t+1} = f(x_t, u_t)$, where $f\colon X_t \times U_t \rightarrow X_{t+1}$ is a function that maps the current robot state and control to the robot state in the next time step. It is assumed that the algorithm can only access an implicit black-box form of $f$. The proposed method could be easily extended to other robot dynamics and provide safe controls regardless of what the robot dynamic is. For the testing environment in this paper, the robot dynamic was assumed to be double integrator, where robot states $X$ containing positions and velocities in $x, y$ axis; control input $U$ is the accelerations in $x, y$ axis.

*Safety Specification:* The safety specification requires that the system state should be constrained in a closed subset in the state space, called the safe set $X_S$. The safe set can be represented by the zero-sub-level set of a continuous and piece-wise smooth function $\phi_0 : \mathbb{R}^{n_x} \rightarrow \mathbb{R}$, i.e.,$X_S = x | \phi_0(x) \leq 0$ . The typical safety design rule $\phi_0$ is $\sigma + d_{min}^n - d_i^n\ - k\dot{d}_i$ , where $\sigma$ , $n$, $k$ are parameters could be designed to extend the available safe set, which allow the system to adapt in different obstacle dynamic environment. Those three safety index parameters could be numerically optimized using a Covariance Matrix Adaptation Evolution Strategy (CMA-ES) method for a better design of $\phi_0$ , which is left for future work. In this paper, $\sigma$ , $n$, $k$ are designed to be 0, 2, 1, respectively, and $\phi_0$ is defined as $d_{min}^2 - d^2 - \dot{d}_i$ , where $d_{min}$ is the user defined safety distance, $d$ is the distance from robot to the closest obstacle and $\dot{d}_i$ is the relative velocity of the vehicle to the concerning obstacle. For safety metric, the model will be trained for 20 episodes, which is typically long enough for the model to converge, and the percentage in which it collides with the obstacles in those 20 episodes will be evaluated.

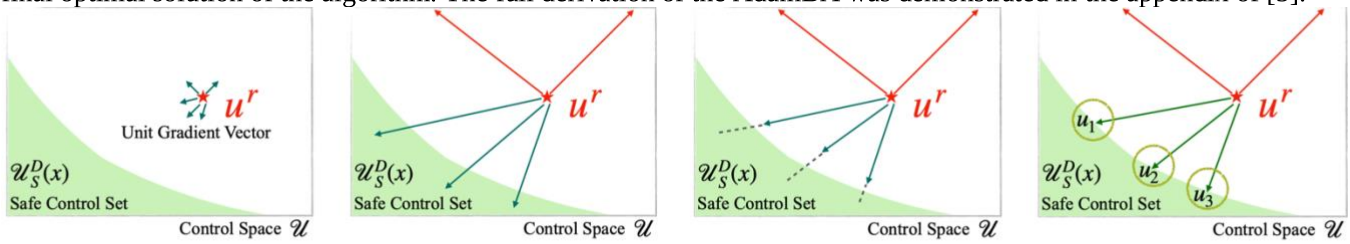
*Reward and Nominal Control:* The Twin Delayed Deep Deterministic Policy Gradients (TD3) was used as the baseline RL model. To improve the efficiency of the exploration strategy used by TD3, a Random Network Distillation (RND) strategy was used to modify the reward function and encourage the agent to visit novel states. Safeguard algorithm alters the nominal control generated by the robot learning controller to protect the robot from collision with the obstacles. The learning controller aims to maximize rewards in Markov Decision Process (MDP). In the MDP ($X$, $U$, $\sigma$, $r$, $p$), $X$ and $U$ are the robot states and robot controls, respectively. The discounting factor $\sigma$ is set to 0.99. The reward function $r$ provides positive reward of 2000 if reaching the goal state $X^*$ , negative penalty of -500 if collide, and zero otherwise. The transition function $p : X \times U \times X \rightarrow [0, 1]$ is defined as $p(x_{t+1} | x_t, u_t) = 1$ when $x_{t+1}\ = x_t + h(x_t, u_t)$, and zero otherwise. The same training experiment was conducted 5 times with different seeds and the average safety performance was evaluated.

*Problem:* The core problem of this paper is to modify the AdamBA algorithm mentioned in [5] and prove that this method could provide safeguard for the learning agent with black-box robot dynamic model in a clustered dynamic environment. The currently proposed algorithm in [1] can always find a local optima solution of the optimization rule with finite iteration given that the safety index guarantees a non-empty set of safe control. However, in a clustered dynamic environment, the current safety index design rule might not be able to find any set of safe control due to the complexity of the obstacles-vehicle interaction scenario. In such scenarios, the experimental results have shown the modified AdamBA was able to provide good safety shield by carefully choosing an action to make the most dangerous obstacle in the next step safest resulting a smallest corresponding safety index. The purposed algorithm cleverly solves the problem without finding concrete safe set with promising safeguard performance in a clustered dynamic testing environment. The redesign of the safety index synthesis rule is necessary to guarantee the SSA to find forward invariant safe set in a clustered dynamic environment, which is left for the future work.

## III. RELATED WORK

Safe Control: In the RL environment, there are typically two steps on safe tracking task for ego vehicle: I. The algorithm learning controller first generates a nominal control action $u_r$; II. A safe control algorithm projects the nominal control to a safe control set. There are four typical safe control methods based on energy function based methods, which are the safe set algorithm (SSA), potential field methods, sliding mode algorithm, and control barrier functions (CBF). For a system with known white-box analytical dynamic models, the safe control set is a half space intersecting with the nominal control U. Therefore, the second step in finding the safe control action is essentially solving a quadratic programming (QP) problem. For the SSA, an energy function was designed offline to ensures the safe control to be at the low energy state and the system was designed to dissipate energy by taking the next safe control action. The general form of safety index in the SSA method was expressed as $\phi_n = \phi_0^* + k_1\dot{\phi}_0 + \cdots + k_n\phi_0^n$, where $k_1$, ..., $k_n$ are real-valued coefficients. Based on Theorem 1 in [1], when the control is unbounded, $\phi_n$ defines an invariant set in $X_s$ if following conditions were satisfied. I. $\phi_0^*$ has the same set as $\phi_0$ to maintain nonlinear gradient $\dot{\phi}$ at the boundary of the safe set; II. The roots of $1 + k_1 s + k_2 s^2 + \cdots + k_n s^n = 0$ are all on the negative real line to maintain the initial safe constraints; III. The relative degree of $\phi_0^n$ to the control input is one to avoid singularity. The SSA define a safe control set that satisfy $\dot{\phi} \leq -\eta\phi$ when $\phi \geq 0$, and it guarantees finite time convergence of system states to the forward invariant safe set $X_s$

Sample Efficient Black-Box Contained Optimization: While the analytical control affined vehicle dynamics is unknown in the system, the safe control set could not be obtained by solving a QP problem. A sample efficient black-box optimization algorithm needs to be utilized to project the nominal control $u_r$ to the safe control set to find best safe control action and providing safety shield to the vehicle. In a sparse obstacle environment, [5] has proposed AdamBA algorithm to efficiently perform the black-box optimization in terms of finding safe control set. The core idea of the AdamBA is to find the boundary points of the safe control set which can be calculated based on the nominal control, similar to the adaptive line search [7] , where four main procedures are followed. I. AdamBA first initializes a set of unit gradient vectors in random orientation based on the reference control to be the sampling direction as show in fig.1 (a); II. The AdamBA sampling directions are expanded with an exponential increase until they reach the boundary of the safe control set $U_S^D(x)$ as shown in fig.1 (b). The sampling directions (red vectors) that exceeding the control space are discarded; III. AdamBA applies binary search in the opposite direction of the vector expansion to find the points on the boundary of the safe set as shown in fig.1 (c); IV. Finally, a set of boundary points will be returned after AdamBA converges as shown in fig.1 (d). AdamBA choses the point on the safe control set boundary that is closest to the reference control to be the final optimal solution of the algorithm. The full derivation of the AdamBA was demonstrated in the appendix of [5].



(a) Initialize gradient vectors (b) Exponential outreach. (c) Exponential decay. (d) Boundary points found.

Fig. 1. Illustration of the procedure of the AdamBA algorithm [1]

## IV. METHOD

In this paper, the specific safe constraint that we are interested is the collision avoidance for a mobile robot in clustered dynamic environment on 2D plane. When crossing a clustered obstacle field, the robot will first detect the positions and velocities of a set of unsafe obstacles nearby. The unsafe obstacles were determined by calculating the safety index ($\phi$) for each obstacle within a given circular range around the robot. The safe control algorithm will be triggered when positive $\phi$ was detected. We choose the most dangerous obstacle with the largest $\phi$ at the current time step as our object of concern for applying AdamBA to synthesize the optimal safe control.

The vanilla AdamBA find a set of boundary points on the safe control set through an algorithm similar to the line search methods and it proceeds to find the optimal solution by choosing the control with minimum deviation from the reference control as the final output. In comparison, for the modified AdamBA proposed in this paper, after all the possible safe controls were found, the vehicle state of the next time step will be simulated by applying all the safe controls found by modified AdamBA to the current vehicle state. Once the vehicle state for the next time step were simulated, the safety index $\phi_{t+1}$ was calculated again for all obstacles with the newly simulated vehicle state, which correspond to safe actions it took in the previous time step. The maximum safety index $\phi_{t+1}$ indicates the most dangerous obstacle in the next time step after the safe control action were taken in the current time step. Therefore, to make vehicle in the safest state in the next time step with respect to the most dangerous obstacle, the safe control action that yields the smallest maximum safety index $\phi_{t+1}$ was chosen to be the final output solution of the algorithm. In other words, after the optimal safe control action was taken, the vehicle will be safest for the most dangerous obstacle in the next time step compared to other safe control actions found in the algorithm. The modified AdamBA algorithm essentially takes a step prediction of the vehicle and obstacle states to determine the best current safe control action instead of choosing the safe control action that is closest to the nominal control as used by vanilla AdamBA.

The performance of modified AdamBA could be further optimized from two aspects: I. Change the three parameters($\sigma, n, k$) of the general safety index design rule ($\phi_0 = \sigma + d_{min}^n - d_i^n - k\dot{d}_i$) to make robot more adaptable in clustered environment by extending the safe control space in the nominal control space; II. Increase the resolution of sampling direction of the AdamBA to make sure majorities of the available safe controls in the control space are detected to allow the algorithm to choose the best possible safe action.

In this paper, the three parameters of safe index were tuned by hand, which were set to be ($\sigma = 0,\ n = 2,\ k = 1$). However, an evolutionary algorithm for any non-linear and non-convex black-box optimization called covariance matrix adaptation evolution strategy (CMA-ES) could be used to generate better results, which is left for future work. It is worth noting that even though increasing the search direction of the AdamBA significantly improves the performance of the algorithm in terms of the collision avoidance, it adds stress to the computational efforts.

## V. EVALUATION METHODS, PLATFORMS, METRICS

Evaluation methods and platforms: the proposed algorithm provided in [5] is evaluated in a clustered dynamic environment. The goal is to move the vehicle from the bottom to the green area on the top while avoiding the random moving obstacles in the field. It is assumed that the vehicle can sense the position and velocity of the obstacle, but not the acceleration. The obstacles will be randomly initialized with bounded velocity and acceleration at each episode. To test the capability and limitation of the algorithm, tests with increasing numbers of obstacles from 10, 20, 30 to 100 in the field were carried out.

The Twin Delayed Deterministic Policy Gradients (TD3) RL model, alone with the sample efficient exploration strategies Random Network Distillation (RND) methods mentioned in [6] will be used as the base line algorithm to generate nominal control policies. Four safeguard algorithms, vanilla SSA, Adaptive SSA in [6], vanilla AdamBA, modified AdamBA, will all be tested with the baseline RL model.

To demonstrate the advantage of the proposed modified AdamBA compared with the adaptive SSA proposed in [6], that the modified AdamBA only requires black-box vehicle dynamics information, the double integrator vehicle dynamics model will be replaced by a unicycle 4 model in the testing environment. To evaluate the effectiveness of each algorithm for providing safety shield on the nominal control policy, we train the models for 50 episodes and evaluate the collision rates. Each training scenarios will be repeated 10 times with different seeds and calculate the average performance. To test limitations of each algorithm, increasing number of obstacles will be initialized in the testing environment from 10, 20, 30, 50 up to 100. To demonstrate the effectiveness of increasing the number of searching directions of AdamBA on the performance of the algorithm, increasing numbers of the random unit gradient vectors of the vanilla and modified AdamBA from 100, 200 to 300 were tested with 100 obstacles in the environment. To demonstrate a RL model with modified AdamBA as its safeguard algorithm has better convergence rate in terms of safety metric than the RL model with any other safeguard algorithm discussed in this paper, all the algorithms are tested in a challenging 100 dynamic obstacle environment, where no algorithms could provide perfect safety shield at the beginning of the RL training. A numbers of interactions vs rewards plot for each algorithm could be generated to demonstrate the effectiveness of each safeguard algorithm on the convergence rate in terms of the safety metric.

Hypothesis: The proposed algorithm will be verified by the following four hypotheses:

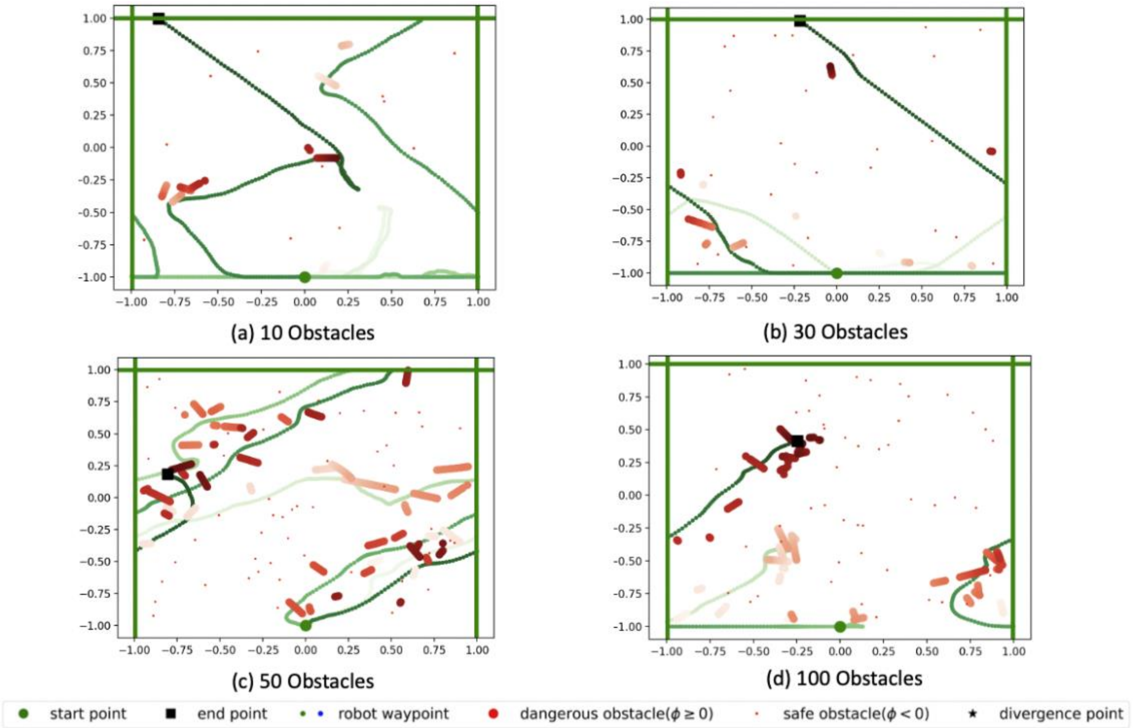
- H1: The pure AdamBA + RND + RL(TD3) could provide safety shield to the vehicle in a sparse obstacle environment but failed with the increasing number of obstacles in the environment.
- H2: The modified AdamBA + RND + RL(TD3) could provide safety shield to the vehicle in a clustered dynamic environment with minimum collision rate compared with another algorithm.
- H3: The adaptive SSA algorithm proposed in [6] could only provide good safety shield with white-box robot dynamics, for the collision rate will dramatically increase with the change of the robot dynamics. On the other hand, the performance of AdamBA will not be affected by the change of the robot dynamics.
- H4: The random unit vector searching resolution of the AdamBA affects the performance of the algorithm dramatically. With increasing number of the searching directions, the algorithm will have a higher possibility to find better safe control action, which improves the success rate in the experiment.
- H5: In all the safeguard algorithms integrated with the baseline RL, the RL with the modified AdamBA as its safeguard algorithm for the agent shows a better convergence rate in terms of safety metrics.

## VI. RESULTS

H1: The vanilla AdamBA was added as the safety shield algorithm on top of the base line algorithm (RND+RL) and was tested in an increasing number of random dynamic obstacles environment. As shown in the Table I, with the increasing numbers of obstacles in the field from 10, 30, 50 to 100 obstacles, the collision avoidance success rate drops dramatically from 82% to 22%. Corresponding trajectories of the robot crossing a dynamic obstacle field was shown in Fig.2. When the robot was tested in a relatively sparse obstacle environments with 10 and 30 obstacles, Fig.2 (a) and Fig.2 (b) shows successful trajectories in such environment. In the clustered dynamic obstacle environment, the vanilla AdamBA barely provides any safety shield. The 22 % and 8 % success rate demonstrated by vanilla AdamBA in such environments might be because of the better nominal control generated by the RL learning controller with performance improvement during the RL training process. Fig.2(c) and Fig.2(d) show failed trajectories in a clustered dynamic with 50 and100 obstacles. The vanilla AdamBA has been tested in various testing environment from the sparse obstacle environment to clustered obstacle environment, and the results have shown that insufficient safeguard effect was observed of vanilla AdamBA in a clustered environment and the further improvement of the algorithm will be needed.

TABLE I SAFETY COMPARISON BETWEEN MODELS WITH INCREASING NUMBER OF OBSTACLES IN THE FIELD

| | Obstacle Numbers | Success | Collision | Failure |
|---|---|---|---|---|
| Vanilla AdamBA (RND + RL) | 10 | 82% | 16% | 2% |
| | 30 | 68% | 24% | 8% |
| | 50 | 22% | 56% | 22% |
| | 100 | 8% | 88% | 4% |
| Modified AdamBA (RND + RL) | 10 | 100% | 0 % | 0% |
| | 30 | 100% | 0% | 0% |
| | 50 | 98% | 2% | 0% |
| | 100 | 59% | 34% | 7% |
| | 200 | 13% | 81% | 6% |



(a) 10 Obstacles

(b) 30 Obstacles

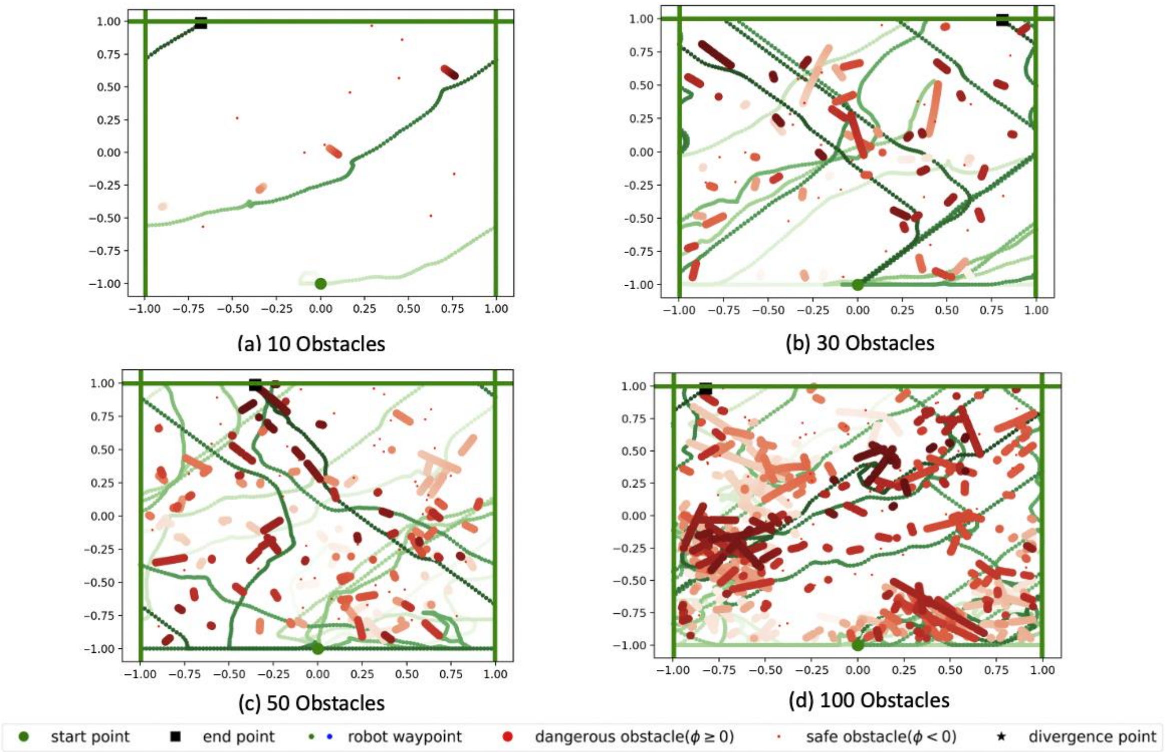
(c) 50 Obstacles

(d) 100 Obstacles

Fig. 2. Illustration of the trajectories of vanilla AdamBA with different numbers of obstacles in the field

H2: The modified AdamBA was added as the safeguard algorithm on top of the base line algorithm (RND+RL) proposed in [6] and was tested in an increasing number of random dynamic obstacles environment. As shown in Table I, with the increasing number of obstacles in the field, with sufficient searching resolutions, the proposed algorithm provides safeguard with nearly 100 % success rate for less than 50 obstacles in the field, which is significantly better than the vanilla AdamBA, the vanilla SSA, and the adaptive SSA methods. It is worth to mention that another advantage of the modified AdamBA is that it does not require white-box explicit knowledge about the system dynamics, while the adaptive SSA in [6] does. However, when the numbers of obstacles in the environment was increased from 100 to 200, the success rate of the algorithm drops from 59% to 13%, respectively. With too many obstacles in the field, the situations exist in which no theoretical or experimental safe control solutions available within the given control boundaries. The trajectories of robot using modified AdamBA as safeguard algorithm with different numbers of obstacles in the field was illustrated in Fig.3. The trajectories have shown the modified AdamBA is effective on providing safety shield to the RL training agent.

H3: In our testing environment with 50 obstacles in the field, vanilla SSA and adaptive SSA algorithm proposed in [6] provide moderately good safeguard with success rate of 50% for adaptive SSA and 69% for vanilla SSA. A comparison of the success and collision rate in such testing environment of all six models have been shown in Table II. It is easy to draw the conclusion that under the same testing environment with 50 obstacles in the field, the modified AdamBA provides the best safeguard to the randomly exploring RL agent. There are two major hurdles to apply SSA and adaptive SSA directly to a clustered dynamic environment. Firstly, due to the safety constraint of vanilla SSA, the algorithm may push the robot to the safest direction at the current time step, but risky in the future. It could also have potential to make the robot stuck in a local optimum at the bottom of the field, as shown in Fig 4a. Secondly, both vanilla SSA and adaptive SSA require the system to query information about the explicit analytical vehicle dynamics model to effectively solve the QP problem to find the safe control set. However, in most of the RL model, explicit vehicle dynamics are typically unknown. When the unicycle four vehicle model was replaced by the double integrator vehicle model in the system, the vanilla SSA and adaptive SSA barely provide any safety shield to the RL training with 0 success rate.



(a) 10 Obstacles

(b) 30 Obstacles

(c) 50 Obstacles

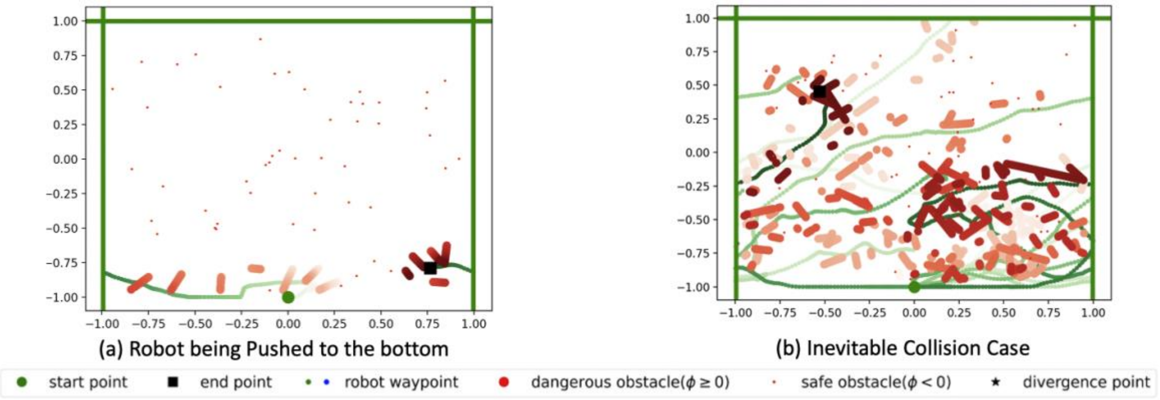
(d) 100 Obstacles

Fig. 3. Illustration of the trajectories of modified AdamBA with different numbers of obstacles in the field

TABLE II SAFETY COMPARISON BETWEEN MODELS [2]

| | Model | Success | Collision | Failure |
|---|---|---|---|---|
| Baseline Models | RL | 2% | 32% | 66% |
| | RND + RL | 5% | 48% | 47% |
| Interested Models (RND + RL) | Vanilla SSA | 50% | 1% | 49% |
| | Adapted SSA | 69% | 1% | 30% |
| | Vanilla AdamBA | 22% | 56% | 22% |
| | Adapted AdamBA | 100% | 0% | 0% |

H4: With 50 obstacles in the environment, the vanilla AdamBA and the modified AdamBA were both tested with increasing number of gradient vectors searching directions from 100, 200 to 300. The success rate of the vanilla AdamBA in such environment remains low regardless of how many searching directions being imposed on the algorithm, while the success rate of the adapted AdamBA improved significantly with the increasing number gradient vectors as shown in the Table III. A robot is called to be in an inevitable collision state when the theoretical safe control set does not exit at the current time step, as shown in Fig.4b. In vanilla AdamBA, there might exist large number of robot-obstacles interaction scenarios that a theoretical optimal safe control action does not exist, while theoretical safe control set exists in most of cases for the modified AdamBA. This could be the reason why increasing the searching directions of random unit vectors demonstrate more dominant effect on the performance of the modified AdamBA algorithm.



(a) Robot being Pushed to the bottom

(b) Inevitable Collision Case

Fig. 4. Illustration of the failed trajectories

TABLE III SAFETY COMPARISON WITH DIFFERENT SAMPLE RESOLUTIONS OF THE VANILLA AND MODIFIED ADAMBA

| | Sample Directions | Success | Collision | Failure |
|---|---|---|---|---|
| vanilla | 100 | 4% | 93% | 3% |
| AdamBA | 200 | 6% | 86% | 8% |
| (50 obstacles) | 300 | 14% | 83% | 3% |
| modified | 100 | 14% | 83% | 3% |
| AdamBA | 200 | 50% | 45% | 5% |
| (50 obstacles) | 300 | 98% | 1% | 1% |

**H5:** The average performances of the four safeguard algorithms with the baseline RL are shown in Fig. 5. The modified AdamBA shows the best convergence rate and safeguard performance to the agent during the training as expected. The pure AdamBA shows similar limited safeguard performance with vanilla SSA in a clustered dynamic environment. While the adapted SSA shows better performance than the vanilla SSA and pure AdamBA, the modified AdamBA converges with almost no collisions by around 40000 steps of interactions. In a complicated clustered dynamic environment with 100 obstacles, the RL results coincide with all the four hypothesis discussed previously, which further demonstrated the effectiveness of the modified AdamBA on safeguard performance to a randomly exploring RL agent during the training.

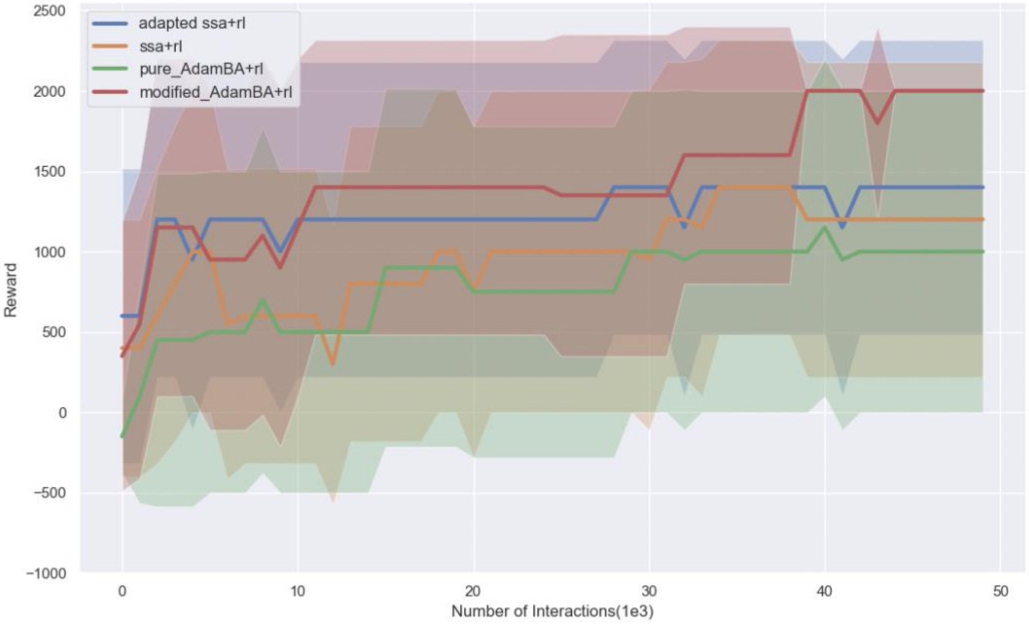


Fig. 5. Average performance of baseline RL + four safeguard algorithms over 50000 steps of interactions

## VII. DISCUSSION

The Implicit Safe Set Algorithm (ISSA) proposed in [5] is a safety guarantee algorithm for collision avoidance in a sparse obstacle environment. Although it only requires a black-box implicit vehicle dynamics model to achieve zero-safety violation during the training, its safeguard performance on the safe tracking vehicle remains unknown in a clustered dynamic environment. The adaptive SSA proposed in [6] provides safety shield to the RL in a clustered dynamic environment with the success rate of 69%, but it requires explicit knowledge about the robot dynamics.
The modified AdamBA proposed in this paper provides reliable safety shield to a moving robot in a clustered obstacle dynamics environment with only black-box non-analytical robot dynamics information. This algorithm has been tested up to 50 obstacles in the environment with the success rate of 100%. This links the gap between [5] and [6], which provide a solution for zero safety violation in a clustered environment with black-box vehicle dynamics model. However, further increase of the number of obstacles in the field leads to dramatic raise of the failure rate. The reason to observe the drop of success rate might because with too many obstacles in the environment, in lots of robot-obstacle interaction scenarios, theoretical safe control solution does not exist, in which case the collision is unavoidable with a confined control boundary.

In the situation that the explicit robot dynamic is known, the performance of the adaptive SSA was compared with the modified AdamBA in terms of collision avoidance. With 50 obstacles in the field, the success rates of the adaptive SSA and modified AdamBA for the first 20 RL(TD3) training episodes are 69% and 100%, respectively. Those two methods both involve designing a safety $\phi$ offline, but those two methods modify the original SSA in two different ways: the adaptive SSA changed the objective function of original SSA, while the modified AdamBA changed the safe control constraints. With the explicit vehicle dynamics model, adaptive SSA redefines the objective function to be minimizing the total distance between robot and all the unsafe obstacles for next time step, and then solves a Quadratic Programming (QP) problem to find the optimal control solution. With the implicit vehicle dynamics model, after finding all the available safe controls using method similar to line regression method, modified AdamBA find the optimal safe control by change the constraints for the optimal solution from among all the safe controls found, choosing the safe control that is closest to the reference control, to choosing the safe control that gives the smallest maximum phi for the next time step. From the result, the modified AdamBA method is much more effective. Thus, it is safe to conclude that to achieve safe control in a clustered dynamic environment, it is more effective to change the safe control constraints than to change the safe control objective function.

There are multiple tasks could be explored to further improve the algorithm in the future. The three parameters ($\sigma$, $n$, $k$) of the general safety index design rule $\phi_0 = \sigma + d_{min}^n - d_i^n - k\dot{d}_i$ could be optimized using CMA-ES algorithm to make safety index ($\phi$) more adaptable in a clustered environment. The modified AdamBA algorithm could be tested in different obstacle dynamics environments, where the obstacles could be designed to be adversarial to the robot. A visual representation of the safe control subspace of the nominal control space could be added for better knowledge of where the safe control set stands in the control space and what control actions have been chosen by the modified AdamBA algorithm for each time step. Better safe constrain options could be explored on the modified AdamBA to extend the probability to find the optimal safe control for the given time step. A system state prediction algorithm could be added to assist the modified AdamBA to make the best safe control action at the current time step based on the future prediction. The modified AdamBA has only being tested on a macro level with the results being the numbers of the collisions for first 20 episodes in each RL training, while the experiments on a micro level could be conducted to analyze how each collision happened, what could be the best theoretical safe control action for the agent to take and how to adjust the algorithm to find that best action.

## VIII. CONCLUSION

In this paper, we have tested the AdamBA algorithm purposed in [5] in a clustered dynamic environment and proved that vanilla AdamBA algorithm can provide moderately good safety shield up to certain numbers of obstacles in the field. We proposed a modified AdamBA algorithm to make the vanilla AdamBA algorithm more adaptable to clustered dynamic environment, and the modified AdamBA algorithm has proved to provide safety shield with 100 % success rate for the collision avoidance task with 50 obstacles in the testing environment. We also proved that to achieve safe control in a clustered dynamic environment, it is more effective to change the safe control constraints than to change the safe control objective function. The proposed algorithm, modified AdamBA, could provide safe controls with high success rate to the randomly exploring robot without any implicit knowledge about the robot dynamics in a clustered dynamic environment.

## REFERENCES

[1] C. Liu and M. Tomizuka. Control in a safe set: Addressing safety in human-robot interactions. In ASME 2014 Dynamic Systems and Control Conference. American Society of Mechanical Engineers Digital Collection, 2014.

[2] O. Khatib. Real-time obstacle avoidance for manipulators and mobile robots. In Autonomous robot vehicles, pages 396-404. Springer, 1986

[3] A.D. Ames, J.W.Grizzle, and P.Tabuada. Control barrier function based quadratic programs with application to adaptive cruise contro. In 53rd IEEE Conference on Decision and Control, pages 6271- 6278.IEEE,2014.

[4] L.Gracia, F.Garelli, and A.Sala. Reactive sliding-mode algorithm for collision avoidance in robotic systems. IEEE Transactions on Control System Technology, 21(6):2391-2399,2013.

[5] W. Zhao, T. He, and C. Liu, “Model-free safe control for zero-violation reinforcement learning,”in Proc. 5th Annu. Conf. Robot Learn., 2021.

[6] H. Chen, C. Liu,” Safe and sample-efficient reinforcement learning for clustered Dynamic Environments,”in IEEE Control System Letters VOL. 6, 2022.

[7] L. Armijo. Minimization of functions having lipschitz continuous first partial derivatives. Pacific Journal of mathematics, 16(1):1-3, 1966.

[8] J. Achiam, D. Held, A. Tamar, and P. Abbeel, “Constrained policy optimization,”in Proc. Int. Conf. Mach. Learn., 2017, pp. 22–31.

[9] A. Ray, J. Achiam, and D. Amodei. Benchmarking safe exploration in deep reinforcement learning. CoRR, abs/1910.01708, 2019.