Interactive Learning on Safe Navigation in Cluster Dynamic Environments

HONGYI CHEN, Georgia Institute of Technology, USA
SHENGKANG CHEN, Georgia Institute of Technology, USA
YUE YANG, Georgia Institute of Technology, USA

Safe navigation is essential for mobile autonomous systems to deploy in real-world environments. In this paper, we want to investigate different safe reinforcement learning (RL) approaches for a robot to navigate safely in dynamic cluster environments. Based on the same learning framework where safe action is used, if the RL agent provides an unsafe action, we developed two different approaches: one uses an optimization-based safety controller to produce safe actions, the other uses human interventions as safe actions. Our experiment results indicate that the optimization-based safety controller can safeguard the robot from collision, but the approach using human interventions achieves very similar performance as regular RL.

ACM Reference Format:
Hongyi Chen, Shengkang Chen, and Yue Yang. 2022. Interactive Learning on Safe Navigation in Cluster Dynamic Environments. 1, 1 (November 2022), 5 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Autonomous navigation is a critical part of mobile robots. In order for mobile robots to deploy in real-world scenarios, it is important for mobile robots to navigate in a cluster environment with dynamic obstacles safely. Learning approaches using reinforcement learning [1] and interactive learning [2] have shown promising results in robot navigation.

The goal of this project is to investigate learning-based approaches with control-based hard safety constraints and learning-based approaches with human interventions for safe navigation in a dynamic cluster environment. We use a learning framework for two different safe-RL models: one using a safety controller (Safe RL-SC) and one using human intervention (Safe RL-HI). We want to compare the performances of these models. As a result, we test these models and analyze the simulation results in various environments.

2 RELATED WORKS

Autonomous navigation has been a popular research area due to its wide range of applications from self-driving cars to vacuum robots. Various learning approaches have been proposed, including reinforcement learning [1], interactive learning [2] and inverse reinforcement learning [3]. These approaches have made substantial progress. [4] uses an imitation learning technique to allow UAVs to navigate in cluttered natural environments without collisions. We want to extend it to a dynamic cluster environment, which can include both dynamic and static obstacles.

Since safety is a crucial part of robot real-world deployment to prevent damage to the robots and the environment, different approaches have been developed for RL-based controllers. One type of approaches is using soft safety constrains like the Lagrangian method [5] or constrained policy optimization [6] for robots to learn safe behaviors, but these methods can not provide derived safety guarantees.

Another type of approach is to use human interventions, which only have safety guarantees in simple scenarios [7]. [2] combines both human demonstrations and interventions to achieve safe training of a UAV. Wang et al. leverage human intervention to improve both the performance and safety of reinforcement learning in navigation [8]. Furthermore, since human interventions can be labor-intensive during training, researchers proposed to train a supervised learner to imitate human [9]. However, it has only been tested in small simulation environments. None of these methods can provide safety guarantees, either.

Researchers also developed methods that aim to guarantee zero safety constraint violation including [10] using a barrier function method [11] and a state-based action correction execution [12]. However, only systems in a static environment have been tested using these methods. Our safe-RL model with safety controller is closely related to [13], where [13] uses reactive synthesis and MDP abstraction to generate safe actions, while our safe-RL model generates safe actions using the safe set algorithm (SSA) [14].

All of these methods have their own strengths and weakness, we want to study and compare the safe RL with human intervention and the safe RL with hard safety constraints in dynamic cluster environments.

3 METHODS

We used two different safe action approaches: one is a reinforcement learning approach with a safety controller, and the other one is an interactive learning approach using human interventions. To analyze the two approaches, we build two RL models on the same framework for fair comparison, where safe action is used when it detects an unsafe action. We also created two baselines to evaluate the efficacy of the safe learning approaches.

For environments, we create 2D environments with both dynamic obstacles and static obstacles. Dynamic obstacles are located randomly and move in random directions to mimic real-world environments. The goal of the robot is to navigate from the start area to the goal area without any safety violations (collisions).

Authors' addresses: Hongyi Chen, Georgia Institute of Technology, Atlanta, USA, hchen657@gatech.edu; Shengkang Chen, Georgia Institute of Technology, Atlanta, USA, schen754@gatech.edu; Yue Yang, Georgia Institute of Technology, Atlanta, USA, yueyang9923@gatech.edu.

XXXX-XXXX/2022/11-ART $15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

For the robot and dynamic obstacles in the environment, we use double integrator dynamics. s_R presents the robot state and s_E^c represents the closest obstacle state. Both s_R and s_E^c contain the positions and velocities along the x-axis and y-axis. The control input of robot a are the accelerations along the x-axis and y-axis. Given both robot state s_R and the closest obstacle state s_E^c, the robot needs to output action $a \in A$ for navigation in the environment. The robot dynamics are defined as:

$$\dot{s}_R = f(s_R) + g(s_R)a \tag{1}$$

For all RL polices, we formulate the problem as a sparse reward task for exploration of the environment using the reward function r. The robot will receive a positive reward if it successfully reaches the goal area or a negative reward if it causes a collision.

$$r = \begin{cases} 2000, & \text{if reach goal.} \\ -500, & \text{if collide.} \end{cases} \tag{2}$$

3.1 RL with a Safety Controller

For the safety controller part, we use the safe set algorithm (SSA) [14] to output a safe action a^{safe} when we detect an unsafe action. The central port of SSA is the valid safety index ϕ. Given an adjustable $\eta \in [0, 1]$, we need to define ϕ such that $\dot{\phi} \leq -\eta\phi$ when $\phi \geq 0$ to ensure a feasible safe control input $a^{safe} \in A$ exits. The safe action a^{safe} needs to keep the state of robot within the safe set $s_R \in S^{(}_{R}safe)$ or converge it to the set $S^{(}_{R}safe)$ in finite time. As a result, the safety index ϕ is defined as:

$$\phi = d_{min}^2 - d^2 - k \cdot \dot{d}. \tag{3}$$

We use d and $\dot{d}$ to represent the distance and relative velocity from the robot to the closest obstacle respectively. Moreover, d_{min} is the user-defined safety distance and k is a constant factor. Given the robot dynamics in 1, we can express $\dot{\phi}$ as

$$\dot{\phi} = \frac{\partial\phi}{\partial x} f + \frac{\partial\phi}{\partial x} g \; a^{safe} = L_f\phi + L_g\phi \; a^{safe}. \tag{4}$$

When an unsafe action is detected ($\phi > 0$), the safety controller will compute ϕ_i for every obstacle and add the corresponding safety constraint within its sensing range. Given all safety constraints and the feasible robot control set A (actions within velocity and acceleration limits), SSA will output safe action a^{safe} using quadratic programming (QP), for simplicity, we use a to represent a^{safe} in the following equation:

$$\begin{gathered} \min_{a\in A} ||a - a^r||^2 = \min_{\in A} a^{\mathrm{T}} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} a - 2a^{\mathrm{T}} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} a^r \\ s.t. L_f\phi_i + L_g\phi_i \; a \leq -\eta \; \phi_i, i = 1, 2...m. \end{gathered} \tag{5}$$

Using SSA, the safety controller can produce safe actions a^{safe} to safeguard the robot from collision in dynamic cluster environment navigation. The action a^{safe} and its corresponding state will be stored in the Safe Action Database. The RL agent will use the database to learn safe actions during training.

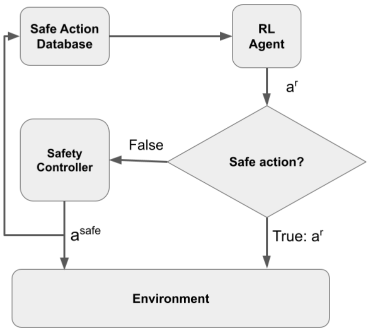


Fig. 1. The proposed reinforcement learning approach with an SSA-based safety controller (safe RL-SC)

3.2 RL with Human Intervention

Interactive learning approach has a very similar learning framework as shown in Figure 2. When a RL agent outputs an unsafe action, the simulation will pause and the robot will ask for human intervention. The human expert will provide a safe action for the RL agent to interact with the environment by giving the acceleration direction and scale. These human interventions a^h will also be stored in the Safe Action Database, so the RL agent will learn the provided human intervention to create a safe action next time. However, the human user has a much different observation space than the robot. Robots can only observe nearby obstacles, but the human user can observe the whole environment including all obstacles and the goal region.

In actual training, we take a relatively aggressive strategy to give demonstrations. In most cases, we give directions that could help the agent reach the goal as quickly as possible. And we oftentimes give the maximum scale of acceleration to enable enough encouragement for the agent.

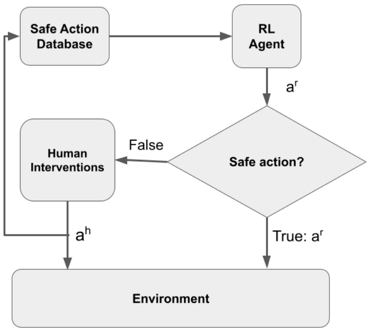


Fig. 2. The interactive learning approach using human interventions (safe RL-HI)

3.3 Baseline

The baseline (BL) we want to compare with is a basic reinforcement learning approach without any safe components. We want to see if using regular reinforcement learning can allow an agent to learn safe actions to avoid collisions.

4 EXPERIMENT

4.1 Setup

There are two environments in the environment: the default environment (Fig.3a) where the robot needs to go from the south side of the environment to the north side of the environment without any collisions with the 50 dynamic obstacles; cross environment (Fig.3b) where the robot needs to go from the south side to the west side of the environment without any collisions with the 30 dynamic obstacles and the 4 large static obstacles at all corners. The default environment is designed to simulate a busy parking area and the cross environment is designed to simulate an intersection. At the beginning of each episode, dynamic obstacles will appear randomly in the environment and start moving toward random directions. These environments are rendered using the *Pyglet* library based on [15]. In the environments, the robot can observe 3 closest obstacles for safe navigation, but it does not know the goal region, so it needs to explore the environment first to find the goal region.

For human intervention, we will collect data ourselves for human intervention, since it requires human supervision during the training process. During training, the human user will input the direction and the magnitude of the velocity when an intervention is needed. We collection ten sets of human interventions for each environment. One set of human interventions is a set of human inputs that allows a robot to reach the goal region from the start region without any collisions once.

4.2 Implementation Details

The RL algorithm we used is the Delayed Deep Deterministic Policy Gradients (TD3) [16] to train our RL agent. When training the policy, we used a fixed 40% : 60% ratio between the RL agent actions and safe actions to combine the training data. In our experiments, we trained the three models with the same hyperparameters in the two environments for 200 episodes. We trained each model 3 times independently to get our experimental data. For evaluation, we ran every model 10 times every 40 episodes.

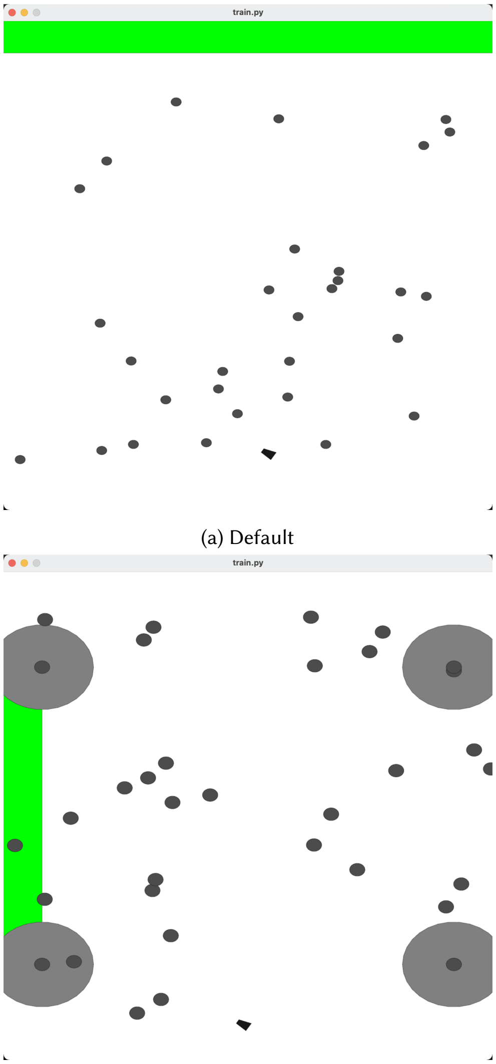
4.3 Hypothesis

We proposed three hypotheses:

H1: Safe RL approach with a safety controller (Safe RL-SC) achieves the best safety performance in all three metrics.
H2: Safe RL approach with human interventions (Safe RL-HI) has better safety performance than regular RL.
H3: Safe RL approach with human interventions (Safe RL-HI) will be more aggressive than Safe RL approach with a safety controller (Safe RL-SC) in terms of number of steps to reach the goal region.

4.4 Performance Metrics

- **Success rate**: the percentage of robots that reach the goal region without collision.
- **Collision rate**: the percentage of robots that cause collisions.
- **Halting rate**: the percentage of robots that are not making any forward progress toward the goal area (staying at the start area).



(a) Default

(b) Cross

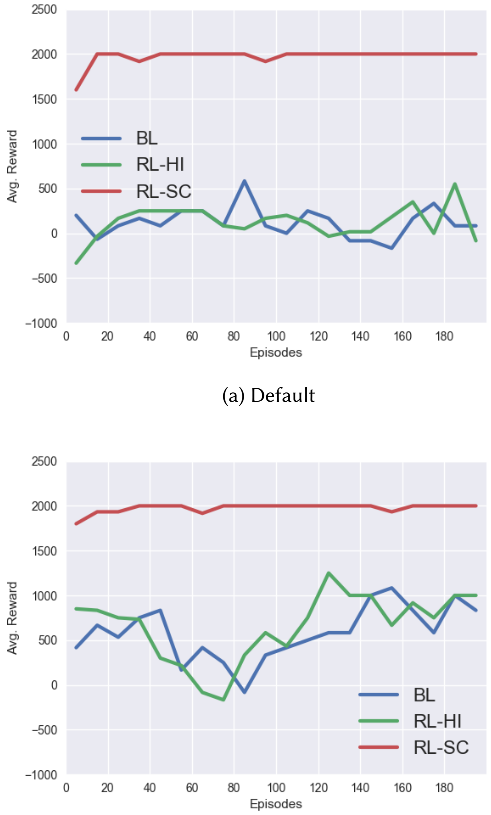
Fig. 3. The two cluster dynamic environments.

4.5 Experimental Results and Discussion

Environment	Default	Cross
Regular RL (BL)	(0.1667/0.5/0.3333)	(0.5667/0.4333/0.0)
Safe RL-HI	(0.1/0.2667/0.6333)	(0.4666/0.5333/0.0)
Safe RL-SC	(1.0/0.0/0.0)	(0.9667/0.0/0.033)

Table 1. The mean performance of 30 runs of each approach after 200 episodes of training (Success rate/Collision rate/Halting rate).

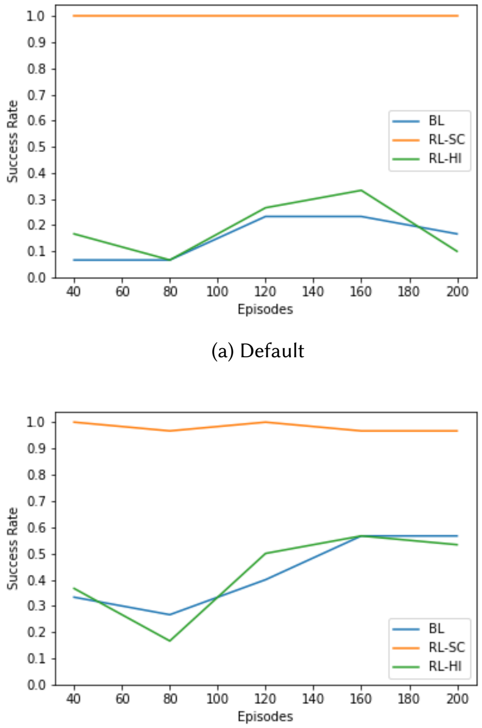
‘ Based on the experimental results (Table.1, Fig. 5), we can see the Safe RL approach with an SSA-based safety controller (Safe RL-SC) has the best safety performance. Its success rate is closed to 1 and is much higher than the other two methods. As shown in the Fig.



(a) Default

(b) Cross

Fig. 4. Average reward over the training process in different environments.



(a) Default

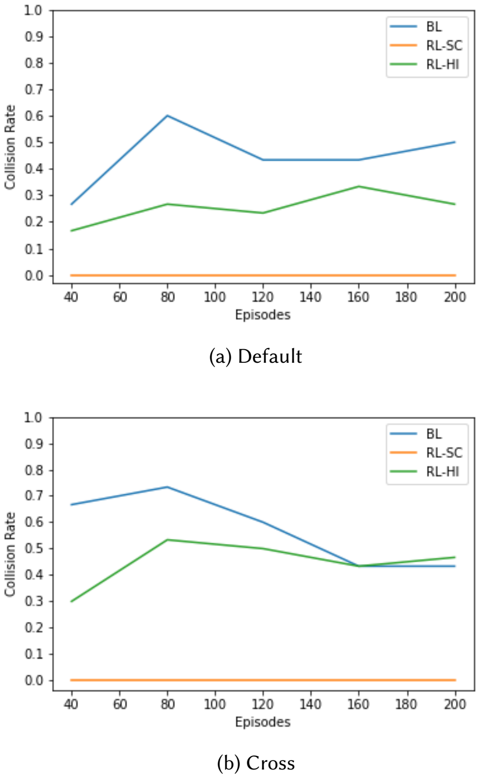
(b) Cross

Fig. 5. Success rate over the course of training in different environments.

4, safe RL-SC can reach the max reward around 20 episodes, which means the SSA-based safety controller can effectively safeguard the robot from collision to allow it to find the goal region quickly. We successfully validate hypothesis H1 using these results. It shows that the SSA-based safety controller can be a valid approach if the dynamics of the robot is known and it can access accurate sensor data.

For Safe RL approach with human interventions (safe RL-HI), we expected Safe RL-HI can allow the robot to learn safe actions from human interventions and outperform regular RL in terms of higher success rate and lower collision rate. However, based on the experimental results (Table.1, Fig. 5 and Fig. 6), we failed to find any statistical significance between Safe RL-HI and regular RL. We look at the movements of these trained robots. Both robots moved straight toward the goal regions and no safe action was performed when they are closed to obstacles. In the default environment, we noticed the robot using Safe RL-HI stayed at the start region more often than regular RL so Safe RL-HI has a much higher halting rate than regular RL. As a result, we fail to validate hypothesis H2. For hypothesis H3, since the provided human interventions ((actions to guide the robots toward the goal region without causing collisions) are more aggressive than the safe actions generated by the safety controller. However, since we did not observe any safe action from the safe RL-HI but only a straight movement toward the goal region, we conclude that we fail to validate hypothesis H3 as well.

To further investigate the issues related to safe RL-HI, we conducted a smaller scale of experiments. We changed the observation space from observing the three closest obstacles to a single closest obstacle. We trained 200 episodes for each approach while keeping all other hyperparameters unchanged. From Table.2, we can see safe RL-HI has higher success rate than regular RL. We can also observe a safer movement pattern (avoiding obstacles while moving toward the goal region) provided by the safe RL-HI. Due to time and computation constraints, we cannot find the causes of the issues, but we suspect three possible causes: one can be due to the randomness in training, another one can be the heterogeneity of human interventions, the third one can be the discrepancy of observation between the human user and the robot. For the randomness in training, we notice the performance variance between models using the same approach. For the heterogeneity of human interventions and the discrepancy of observation, since the human user can observe the whole environment and the robot can only observe the nearby robots, human interventions can be dramatic different even though the observations of the robots are very similar. Further investigation on safe RL-HI is needed.



(a) Default

(b) Cross

Fig. 6. Collision rate over the course of training in different environments.

Environment	Default
Regular RL (BL)	(0.3/0.7/0.0)
Safe RL-HI	(0.5/0.5/0.0)
Safe RL-SC	(1.0/0.0/0.0)

Table 2. The mean performance of 10 runs of each approach after 200 episodes of training where the robot can only observe the closet obstacle (Success rate/Collision rate/Halting rate).

5 CONCLUSION

In this paper, we investigated different safe RL approaches for safe navigation in cluster dynamic environments. We developed two different approaches using the same learning framework: a safe reinforcement learning approach using an SSA-based safety controller, which is a hard safety constraint approach; and a safe reinforcement learning approach using human interventions. Our experimental results show that the SSA-based safety controller is a valid method if it has access to the robot dynamics and accurate sensor data. However, the learning approach using human interventions failed to perform safe navigation. We have several suspected causes of failure, but further investigation and experiments are needed.

REFERENCES

[1] Yuke Zhu, Roozbeh Mottaghi, Eric Kolve, Joseph J. Lim, Abhinav Gupta, Li Fei-Fei, and Ali Farhadi. Target-driven visual navigation in indoor scenes using deep reinforcement learning. *Proceedings - IEEE International Conference on Robotics and Automation*, pages 3357–3364, 9 2016.

[2] Vinicius G. Goecks, Gregory M. Gremillion, Vernon J. Lawhern, John Valasek, and Nicholas R. Waytowich. Efficiently combining human demonstrations and interventions for safe training of autonomous systems in real-time. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33:2462–2470, 7 2019.

[3] David Silver, J. Andrew Bagnell, and Anthony Stentz. Learning from demonstration for autonomous navigation in complex unstructured terrain:. *http://dx.doi.org/10.1177/0278364910369715*, 29:1565–1592, 6 2010.

[4] Stephane Ross, Narek Melik-Barkhudarov, Kumar Shaurya Shankar, Andreas Wendel, Debadeepta Dey, J. Andrew Bagnell, and Martial Hebert. Learning monocular reactive uav control in cluttered natural environments. *Proceedings - IEEE International Conference on Robotics and Automation*, pages 1765–1772, 11 2012.

[5] Adam Stooke, Joshua Achiam, and Pieter Abbeel. Responsive safety in reinforcement learning by pid lagrangian methods. pages 9133–9143. PMLR, 11 2020.

[6] Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization. In *Proc. Int. Conf. Machine Learning*, page 22–31, 2017.

[7] Acm Reference Format: William Saunders, Girish Sastry, Andreas Stuhlmüller, and Owain Evans. Trial without error: Towards safe reinforcement learning via human intervention. 2018.

[8] Fan Wang, Bo Zhou, Ke Chen, Tingxiang Fan, Xi Zhang, Jiangyong Li, Hao Tian, and Jia Pan. Intervention aided reinforcement learning for safe and practical policy optimization in navigation. In Aude Billard, Anca Dragan, Jan Peters, and Jun Morimoto, editors, *Proceedings of The 2nd Conference on Robot Learning*, volume 87 of *Proceedings of Machine Learning Research*, pages 410–421. PMLR, 29–31 Oct 2018.

[9] Bharat Prakash, Mohit Khatwani, Nicholas Waytowich, and Tinoosh Mohsenin. Improving safety in reinforcement learning using model-based architectures and human intervention. *The Thirty-Second International Flairs Conference*, 5 2019.

[10] Felix Berkenkamp, Matteo Turchetta, Angela P. Schoellig, and Andreas Krause. Safe model-based reinforcement learning with stability guarantees. *Advances in Neural Information Processing Systems*, 2017-December:909–919, 5 2017.

[11] Richard Cheng, Gábor Orosz, Richard M. Murray, and Joel W. Burdick. End-to-end safe reinforcement learning through barrier functions for safety-critical continuous control tasks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33:3387–3395, 7 2019.

[12] Gal Dalal, Krishnamurthy Dvijotham, Matej Vecerik, Todd Hester, Cosmin Paduraru, and Yuval Tassa. Safe exploration in continuous action spaces. 1 2018.

[13] Mohammed Alshiekh, Roderick Bloem, R ¨ Udiger Ehlers, Bettina Könighofer, Scott Niekum, and Ufuk Topcu. Safe reinforcement learning via shielding. *Thirty-Second AAAI Conference on Artificial Intelligence*, 4 2018.

[14] Changliu Liu and Masayoshi Tomizuka. Control in a safe set: Addressing safety in human-robot interactions. *ASME 2014 Dynamic Systems and Control Conference, DSCC 2014*, 3, 2014.

[15] Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, Pieter Abbeel, and Igor Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. *Advances in Neural Information Processing Systems*, 2017-December:6380–6391, 6 2017.

[16] Scott Fujimoto, Herke van Hoof, and David Meger. Addressing function approximation error in actor-critic methods. In *Proc. Int. Conf. Machine Learning*, pages 1582–1591, 2018.