Computer Science and Application 计算机科学与应用, 2023, 13(1), 104-112
Published Online January 2023 in Hans. https://www.hanspub.org/journal/csa
https://doi.org/10.12677/csa.2023.131011

Hans汉斯

基于安全强化学习的机械臂无碰路径规划

许银涛，王　涛

广东工业大学自动化学院，广东　广州

收稿日期：2022年12月15日；录用日期：2023年1月11日；发布日期：2023年1月19日

摘　要

可以通过为机器人关节添加相关约束的方式来保证规划路径的可靠性。本文研究了带有运动学约束的强化学习(Reinforcement Learning)方法，以保证规划中的安全性。通过与替代动作思想的结合，对强化学习动作空间的进行设计，可以进一步保证动作的可行性。为了评估算法性能，在船舶焊接场景中对工业机械臂进行了路径规划，使得机械臂末端成功运动到位于狭窄空间中的焊接起点。实验结果表明，该方法不仅保证了训练的收敛性，而且保证了任务的安全性和可靠性。

关键词

强化学习，工业机械臂，替代动作，运动规划

Collision-Free Path Planning of Manipulator Based on Safety Reinforcement Learning

Yintao Xu, Tao Wang

School of Automation, Guangdong University of Technology, Guangzhou Guangdong

Received: Dec. 15th, 2022; accepted: Jan. 11th, 2023; published: Jan. 19th, 2023

Abstract

The reliability of the planned path can be guaranteed by adding relevant constraints to the robot joints. In this paper, reinforcement learning (RL) with motion constraints is studied to ensure the safety in planning. With the design of the reinforcement of the learning movement space by combining with the idea of alternative movements, the feasibility of action is further guaranteed. In order to evaluate the performance of the algorithm, the path planning of the industrial robot arm was carried out in the Marine welding scene, so that the end of the robot arm successfully moved to the welding starting point located in the narrow space. Experimental results show that the proposed method not only guarantees the convergence of the training, but also ensures the secu-

文章引用: 许银涛, 王涛. 基于安全强化学习的机械臂无碰路径规划[J]. 计算机科学与应用, 2023, 13(1): 104-112.
DOI: 10.12677/csa.2023.131011

rity and reliability of the task.

Keywords

Reinforcement Learning, Industrial Manipulator, Alternative Behavior, Motion Planning

1. 引言

在工业制造场景中，机器人通常用于执行各类任务，如焊接作业。作业过程中，虽然机器人的性能，如准确性和速度，会优于人力，可机器无法对不可预见的情况做出反应，出于这个原因需要在作业前为机器人规划出一条可行的路径。焊接作业有时将焊接点设定在狭窄空间中，而在狭窄空间中进行路径运动规划往往较为困难，难以在目标点附近收敛。

常用的规划算法中，如 A* [1]之类的基于网格的算法虽然可以找到最短路径，但是随着问题复杂度的提升，时间成本以及计算机内存的占用会呈指数增长。RRT (Rapidly-exploring Random Tree) [2]算法作为基于采样的一种，也被广泛用于各类规划问题中。其着眼于解决全局性、非全局性、高维[3]和相关运动学约束下的路径规划问题。可在复杂的环境中规划时需要花费大量时间进行采样，且在狭窄空间内规划时也存在难以选择结点大的问题，使得该算法无法解决这类问题。除了传统规划算法，强化学习(Reinforcement Learning) [4]算法作为一种机器学习方法，凭借其优势也应用在各类问题中，如游戏训练、无人驾驶、机器人控制等，并且取得了巨大的成功。其通过设置合适的奖励函数，经过不断试错的方式来改善智能体的动作，实现累计奖励的最大化并最终解决相关序列决策问题。对于较为复杂的场景、连续的状态空间或动作空间的问题，则常用深度强化学习来解决。深度强化学习由强化学习与深度神经网络(Deep Neural Networks)结合得到。Deep Q-Network (DQN) [5]与 Deep Deterministic Policy Gradient (DDPG) [6]都是常用的深度强化学习算法。可是在使用这些算法的时候，智能体需要从环境中搜集经验，这使得探索过程中不可避免的会发生一些危险的情况，特别是在复杂的环境，如狭窄空间中。

当然，现有的强化学习算法，包括基于模型的(Model Based)和基于策略的(Policy Based)的强化学习算法，都或多或少存在一些问题，例如智能体训练时间过长、难以稳定收敛和容易进入危险区域等。在[7]中，作者通过为智能体添加相关约束的方法，提出了基于信任区间方法的 CPO (Constrained Policy Optimization)算法，可有效用于高纬度的任务中；高斯过程 GPs (Gaussian Process) [8]也常用于为非确定性因素建模，从而保证规划过程中的安全性。

基于现有的强化学习算法，本文在规划过程中为智能体添加了相关的运动学约束，并将替代动作[9]的思想与强化学习算法相结合。一方面，为机器人关节添加相关的运动学约束可以保证动作的安全性；另一方面，替代动作的加入，可以得到用于强化学习训练的新的动作空间，这个动作空间可以保证训练过程中所选动作的安全性。

2. 强化学习

机器学习可以大致分为三个领域：监督学习、无监督学习和强化学习(Reinforcement Learning)。其中强化学习是通过智能体与环境的交互来学习，从而得到一个策略，这个策略可以使得智能体与环境交互

之后，得到最大化的期望收益，整个过程如图 1 所示。

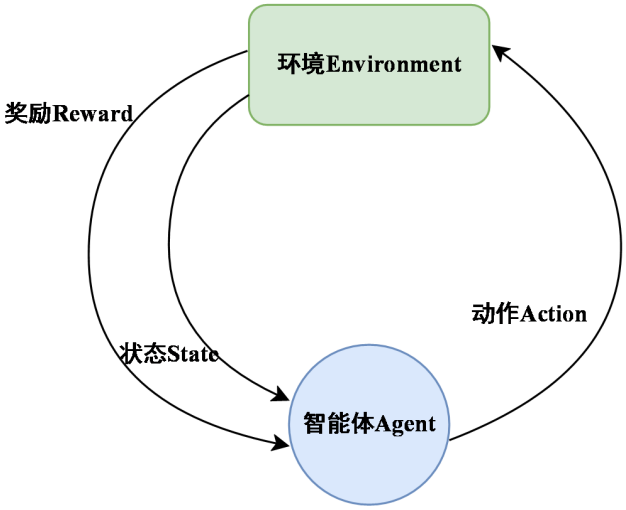

Figure 1. Reinforcement learning framework
图 1. 强化学习框架

Q-Learning 是常用的强化学习算法，其使用一个表格存放在各个状态 s 下采取相应动作 a 所获得的奖励值，通过一个贪婪值 ε 来决定下一时刻所选取的动作，若 ε 取 0.9，则表明有 90%的情况会根据表中的最优值选取动作，有 10%的情况会随机选取动作，根据所选动作来更新 Q 值。更新公式为：

$$Q(s,a) < -Q(s,a) + \alpha\left[r + \gamma \max_{a'} Q(s',a') - Q(s,a)\right]$$

其中 α 是学习率，用来决定这次的误差有多少是要被学习的，是一个小于 1 的数；γ 是对未来奖励的衰减值，可以较少之前的决策对后面决策的不必要影响。

2.1. 深度强化学习

强化学习算法常用于有限的马尔可夫决策过程(Markov Decision Process)，而生活中更多的问题的状态或者动作维度非常高，或者是连续的，故需要用其他方法来解决这类问题。深度强化学习(Deep Reinforcement Learning)就可以很好地解决这些问题。其通过结合深度函数的近似能力——高纬度输入的特征缩放，以及强化学习的泛化能力，使得智能体可以处理具有高维的离散或者连续状态/动作的复杂环境，例如控制机器人的关节[10]。DQN 是常用的深度强化学习算法。DQN 顾名思义，即是 Q-Learning 与神经网络(Neural Networks)相结合得到的算法。DQN 中有两个结构相同但参数不同的网络，一个用于预测 Q 估计(MainNet)，以一个用于预测 Q 现实(target)，MainNet 使用最新的参数，target 会使用很久之前的参数，Q 现实的 targetQ 计算为：

$$\text{targetQ} = r + \gamma \max\left(s', a', \theta\right)$$

根据 targetQ 与 Q 估计得到损失，损失函数一般采用均方差损失：

$$\text{LOSS}(\theta) = E\left[\left(\text{TargetQ} - Q(s,a;\theta)\right)^2\right]$$

其更新公式为：

$$Q(s,a) < -Q(s,a) + \alpha\left[r + \gamma \max_{a'} Q(s',a') - Q(s,a)\right]$$

初始化 MainNet 和 target 网络后，根据损失函数来更新 MainNet 参数，target 固定不变。经过多次迭代后，将 MainNet 的参数拷贝给 target 网络，再循环迭代，直到 targetQ 的值收敛，最终使得算法更新稳定。

2.2. 安全强化学习

安全强化学习指的是在强化学习的学习阶段以及部署期间，注重安全约束的方法。虽然强化学习已广泛应用于机器人，实际应用中的安全学习仍然被认为是一个突出的挑战[11]。安全强化学习的一个重要方面是安全探索问题。为了找到性能良好的策略，RL 智能体(agent)必须探索环境。在这个过程中，必须避免出现危险的情况。为了考虑学习过程中的安全约束，有人引入了约束马尔科夫决策过程 Constrained Markov Decision Process, CMDP)的概念，并开发了用于 CMDP 的强化学习算法。在[12]中，几种有约束和无约束的 RL 算法在不同的环境中进行了基准测试并取得了一定的成果。

当使用工业机器人时，相关的安全约束包括防止碰撞和遵守运动学和动力学关节极限。由于大多数神经网络都是在模拟环境下训练的，将动作部署到真正的机器人中上时，安全约束将变得十分重要。目前已经有不少人开发了各种技术来避免在真实世界进行实验时出现不安全的行为。例如对非期望行为的惩罚可以添加到无约束马尔科夫决策过程(MDP) [13] [14]的奖励函数中。虽然惩罚减少了不良行为的可能性，但是直到训练过程一直进行到收敛都不能提供严格的安全保证。此外，在训练过程开始时的不良行为并不能通过惩罚来预防。在某些情况下，可以使用特定任务的启发式方法来避免不安全的行为[15]。以能够安全地执行所有操作的方式设计动作空间是处理安全约束的一种优雅的解决方案。然而，这种方法通常具有很大的限制性，并不适用于所有类型的约束[11]。为了确保符合运动关节约束，[16]中提出了一种动作空间表示法。在不限制机器人工作空间的情况下，避免了无限时间范围内的冲突约束。

3. 结合替代动作的安全强化学习

3.1. 运动学约束

在使用强化学习算法进行训练之前，首先需要为机器人的关节添加相应的运动学约束。相关的约束包括关节位置、速度、加速度以及制动速度：

$$p_{\min} \le \theta \le p_{\max} \tag{1}$$

$$v_{\min} \le \dot{\theta} \le v_{\max} \tag{2}$$

$$a_{\min} \le \ddot{\theta} \le a_{\max} \tag{3}$$

$$j_{\min} \le \dddot{\theta} \le j_{\max} \tag{4}$$

其中 θ 是关节的位置，p、v、a、j 分别代表位置、速度、加速度以及制动速度。除了关节的约束，还需要设置机器人的连杆－连杆对以及连杆－障碍对，用以检测机器人自身的碰撞以及机器人与环境中障碍物的碰撞情况。

3.2. 替代动作

为了避免碰撞的发生，对在训练过程中所采取的动作，需要以一定频率 f_C 检测机器人与障碍物之间的最小距离，若距离小于一定安全阈值，则判定为发生碰撞。同时在每一时间间隔 t 至 $t+1$ 中，以状态 s_t 为输入，s_t 包含机器人关节的位置，速度，加速度等信息，通过一个动作预测网络进行预测，网络的隐藏层使用 SELU 为激活函数，第一层隐藏层的大小为 256，第二层大小为 128。以频率 f_N 输出各个关节

的一个动作标量 $m_t \in [-1,1]$，这个标量被用来计算下一时刻期望的动作 a_{t+1}：

$$a_{t+1} = a_{t+1_{\min}} + \frac{1+m_t}{2}\left(a_{t+1_{\max}} - a_{t+1_{\min}}\right) \tag{5}$$

$a_{t+1_{\min}}$ 和 $a_{t+1_{\max}}$ 取决于当前状态的运动学状态(p_t， v_t， a_t)以及运动学约束。对预测的动作进行安全检测，若执行动作 a_{t+1} 后机器人未发生碰撞，各关节满足安全约束，则认为预测动作是可行的，并将预测动作作为下一时刻训练采取的动作；否则下一时刻采取制动动作 a_{t+1_B}。整个流程大致如图 2 所示：

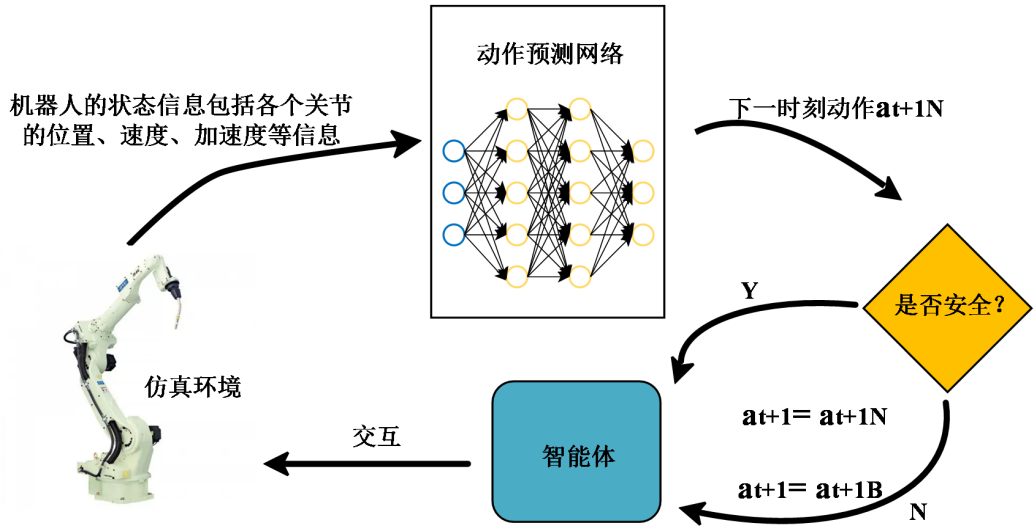

Figure 2. Alternative behavior operation process
图 2. 替代动作运作过程

计算安全动作，等同于在重新设计用于强化学习训练的动作空间。

3.3. 奖励函数

为了驱使机器人准确无误的运动到目标点，需要设置合理的为机器人的训练设置合理的奖励函数，本方法设置的奖励函数如下所示：

$$R = R_{target} - R_{action} - R_{adaptation} - R_{distance} \tag{6}$$

R_{target} 表示机械臂末端到目标点距离的奖励项，用以训练机械臂接近目标点；R_{action} 表示为动作惩罚项，用以避免动作过于接近极限值；$R_{adaptation}$ 表示制动惩罚项，当动作发生碰撞而执行制动动作时，此项为 1，否则为 0；$R_{distance}$ 表示距离惩罚项，执行替代动作后机械臂的杆件之间以及每个杆件与障碍物之间的距离小于一定阈值，则施加惩罚项 1，否则为 0。

4. 实验结果与分析

在本节中，我们在仿真环境中评估了结合了替代动作思想的安全强化学习方法的可行性。此外，还与常规的强化学习算法作对比试验，实验结果表明了此方法的优越性。

4.1. 场景搭建

本方法在 Pybullet 中进行评估。Pybullet 是一个 Python 的模块，可用于机器人、游戏、机器学习等的物理模拟。规划场景如图 3 所示。障碍物为中组立的船板。船板中包含了多种焊缝，包括垂直焊缝、

矩形焊缝等，这就产生了很多狭窄的空间，而许多作业任务都需要在狭窄空间下完成，这使得机器人的轨迹规划变得十分困难。本实验中机械臂的规划任务为从空间中任意起始点运动到目标点，目标点为预设好的焊接起点，而焊接起点位于狭窄空间中，如图 4 所示，其中机器人为六轴的华数工业机械臂。当奖励稳定收敛时，则视为规划完成，当机械臂末端平稳运动到目标点，且运动过程中未发生碰撞，则视为规划成功，如图 5 所示。

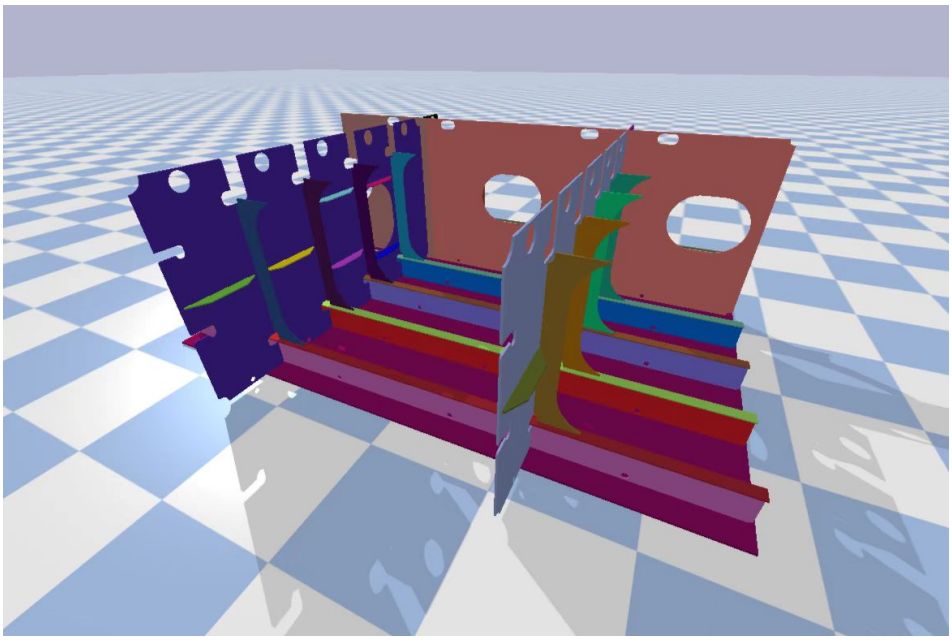

Figure 3. Ship welding scene (welded parts)
图 3. 船舶焊接场景(焊接件)

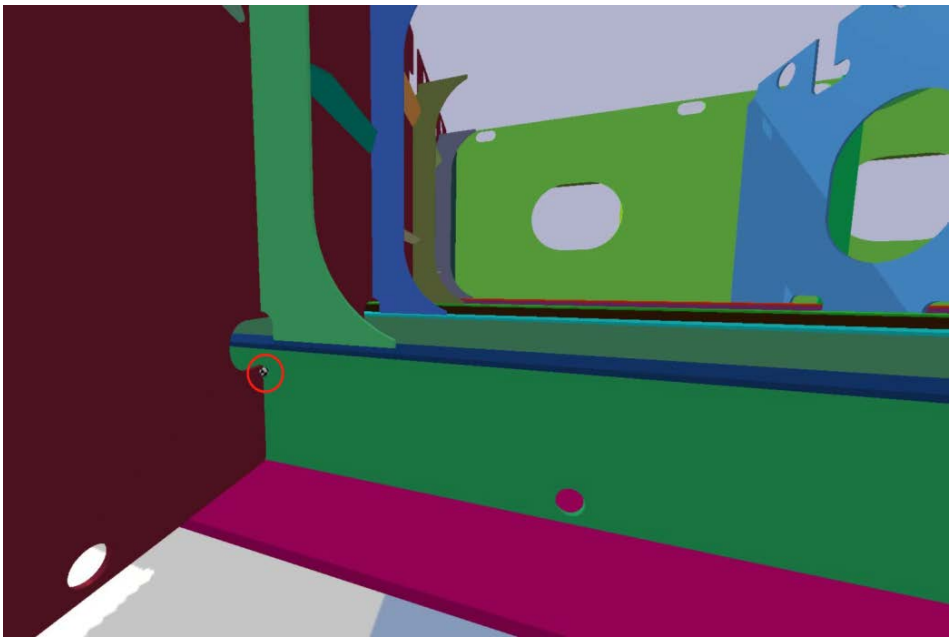

Figure 4. Welding starting point distribution
图 4. 焊接起点分布

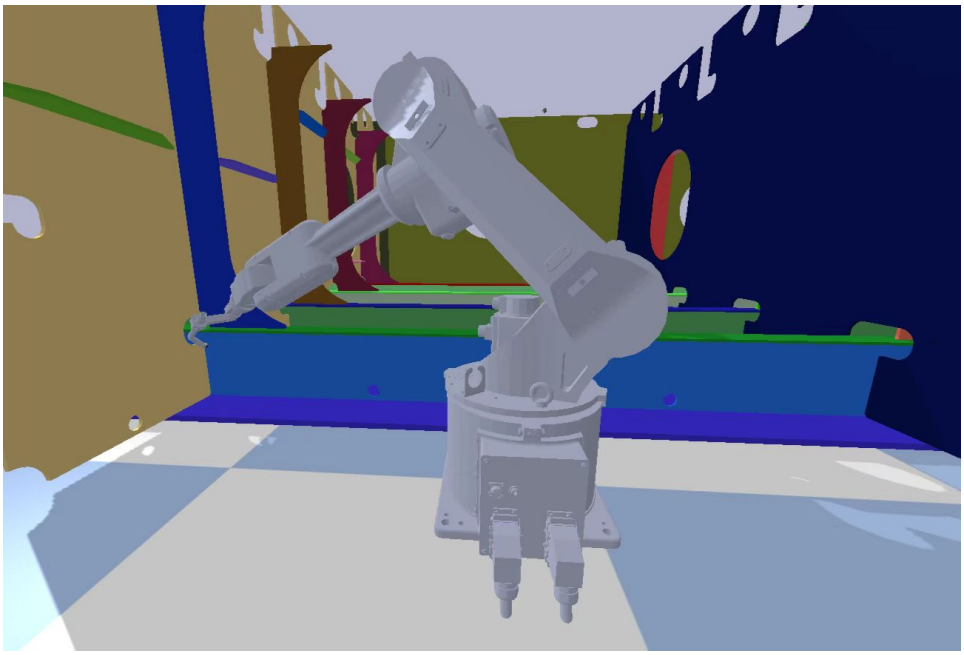

Figure 5. Planning success stories
图 5. 规划成功样例

本实验中使用的强化学习算法为 PPO 算法，为机械臂添加安全学术的同时，将 PPO 算法与替代动作方法结合，并与常规的 PPO 算法以及较具有代表性的 DDPG 算法进行对比，通过奖励的变化情况、规划过程中的碰撞率以及成功率来反映算法的性能。

4.2. 对比实验结果

由于强化学习智能体是通过试错的方法来进行学习的，对环境没有先验知识，故在训练过程中不可避免的会发生碰撞。例如在现实生活中，必须确保机器人抵达障碍物前关节的速度以及加速度的值达到0，否则将发生碰撞，因此需要计算安全的动作。本方法的目的是为了尽可能的减少训练过程中对不安全动作的采样，最终保证选择的动作的可行性。

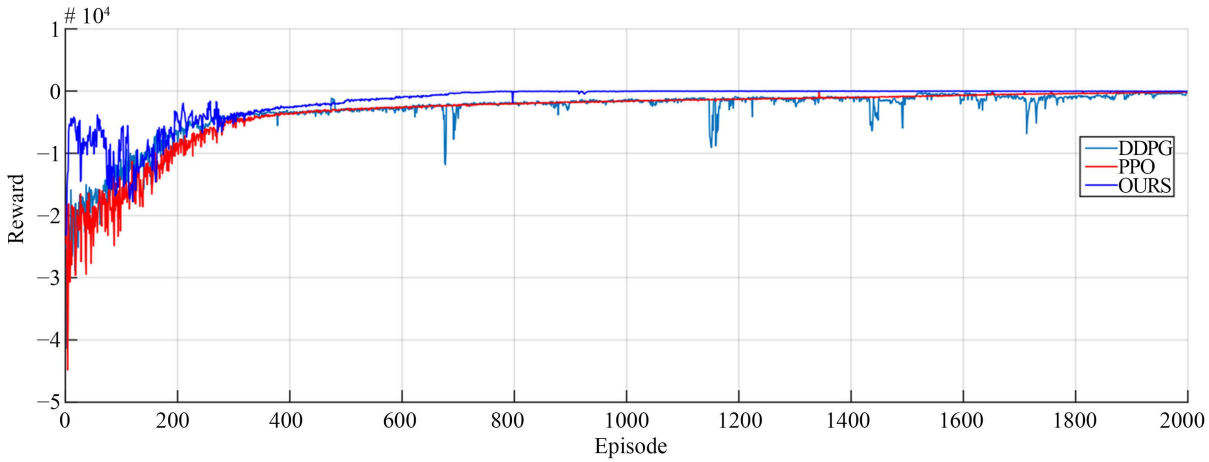

Figure 6. The learning curve corresponding to the three algorithms
图 6. 三种算法对应的学习曲线

图 6 所示的学习曲线有三种算法训练得到，其中 OURS 表示本论文提及的方法。从图中可以看出，由本方法训练得到的学习曲线无论是稳定性还是可靠性都要优于另外两种算法，并且最终能够更好地收敛。当到达目标点附近时，由 DDPG 训练的机械臂因为障碍物的关系，不断在目标点附近探索，这使得奖励难以收敛，同时还存在“失忆”的现象，使得机械臂远离目标点。

Table 1. The collision rate corresponding to the three methods and the success rate
表 1. 三种方法对应的碰撞率以及成功率

方法	碰撞率	成功率
DDPG	16.5%	91.2%
PPO	9.2%	95.3%
OURS	0%	99.5%

表 1 展示了机器人在训练过程中的碰撞率以及规划的成功率。从表中可以看出，使用常规强化学习算法训练的过程中，碰撞的发生是不可避免的，特别是当目标点位于狭窄空间内时。归功于安全动作的计算以及替代动作的结合，动作空间的可靠性以及安全性得到了进一步的保障。

5. 结论与展望

本文介绍了无碰撞且不违反运动学约束的轨迹规划方法，不仅保证了规划性能，还进一步提高了目标点附近的收敛速度以及稳定性。并在焊接场景中的狭窄空间进行规划。与现有的几个强化学习基准相比，具有优越的性能。

未来工作的潜在方向包括在多个机器人协同作业环境中测试算法，以便在更复杂的场景中进行测试，而不仅仅是在一个单一的静态环境中。

参考文献

[1] Yiu, Y.F. and Mahapatra, R. (2020) Hierarchical Evolutionary Heuristic A Search. 2020 *IEEE International Conference on Humanized Computing and Communication with Artificial Intelligence* (*HCCAI*), Irvine, 21-23 September 2020, 33-40. https://doi.org/10.1109/HCCAI49649.2020.00011

[2] Ardiyanto, I. and Miura, J. (2012) Real-Time Navigation Using Randomized Kinodynamic Planning with Arrival Time Field. *Robotics and Autonomous Systems*, **60**, 1579-1591. https://doi.org/10.1016/j.robot.2012.09.011

[3] Kuffner, J.J. and LaValle, S.M. (2000) RRT-Connect: An Efficient Approach to Single-Query Path Planning. *Proceedings* 2000 *ICRA*. *Millennium Conference*. *IEEE International Conference on Robotics and Automation*. *Symposia Proceedings* (*Cat*. *No.*00*CH*37065), San Francisco, 24-28 April 2000, 995-1001.

[4] Morales, E.F. and Zaragoza, J.H. (2012) An Introduction to Reinforcement Learning. In: Enrique, L., Morales, E. and Hoey, J., Eds., *Decision Theory Models for Applications in Artificial Intelligence*: *Concepts and Solutions*, IGI Global, Pennsylvania, 63-80. https://doi.org/10.4018/978-1-60960-165-2.ch004

[5] Mnih, V., Kavukcuoglu, K., Silver, D., *et al*. (2015) Human-Level Control through Deep Reinforcement Learning. *Nature*, **518**, 529-33. https://doi.org/10.1038/nature14236

[6] Lillicrap, T.P., *et al*. (2015) Continuous Control with Deep Reinforcement Learning. ArXiv Preprint ArXiv: 1509.02971.

[7] Achiam, J., Held, D., Tamar, A. and Abbeel, P. (2017) Constrained Policy Optimization. *Proceedings of the* 34*th International Conference on Machine Learning*, Sydney, 6-11 August 2017, 22-31.

[8] Sui, Y., Gotovos, A., Burdick, J. and Krause, A. (2015) Safe Exploration for Optimization with Gaussian Processes. *Proceedings of the* 32*nd International Conference on Machine Learning*, Lille, 6-11 July 2015, 997-1005.

[9] Rubrecht, S., Padois, V., Bidaud, P., De Broissia, M. and Da Silva Simoes, M. (2012) Motion Safety and Constraints Compatibility for Multibody Robots. *Autonomous Robots*, **32**, 333-349. https://doi.org/10.1007/s10514-011-9264-x

[10] Sutton, R.S. and Barto, V. (1998) Reinforcement Learning: An Introduction. *IEEE Transactions on Neural Networks*, **9**, 1054. https://doi.org/10.1109/TNN.1998.712192

[11] Ibarz, J., Tan, J., Finn, C., Kalakrishnan, M., Pastor, P. and Levine, S. (2021) How to Train Your Robot with Deep Reinforcement Learning: Lessons We've Learned. *The International Journal of Robotics Research*, **40**, 698-721. https://doi.org/10.1177/0278364920987859

[12] Marchesini, E., Corsi, D. and Farinelli, A. (2021) Benchmarking Safe Deep Reinforcement Learning in Aquatic Navigation. 2021 *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Prague, 27 September-1 October 2021, 5590-5595. https://doi.org/10.1109/IROS51168.2021.9635925

[13] Tan, J., *et al*. (2018) Sim-to-Real: Learning Agile Locomotion for Quadruped Robots. ArXiv: 1804.10332. https://doi.org/10.15607/RSS.2018.XIV.010

[14] Andrychowicz, M., *et al*. (2019) Learning Dexterous in-Hand Manipulation. *The International Journal of Robotics Research*, **39**, 3-20. https://doi.org/10.1177/0278364919887447

[15] Gu, S., Holly, E., Lillicrap, T. and Levine, S. (2017) Deep Reinforcement Learning for Robotic Manipulation with Asynchronous off-Policy Updates. 2017 *IEEE International Conference on Robotics and Automation*, Singapore, 29 May-3 June 2017, 3389-3396. https://doi.org/10.1109/ICRA.2017.7989385

[16] Pecka, M. and Svoboda, T. (2014) Safe Exploration Techniques for Reinforcement Learning—An Overview. In: Hodicky, J., Ed., *Modelling and Simulation for Autonomous Systems*. *MESAS* 2014. *Lecture Notes in Computer Science*, Vol. 8906, Springer, Cham, 357-375. https://doi.org/10.1007/978-3-319-13823-7_31