Physics Informed Deep Reinforcement Learning for Aircraft Conflict Resolution

物理信息深度强化学习在航空冲突解决中的应用

Peng Zhao and Yongming Liu

彭赵和刘永明

Abstract-A novel method for aircraft conflict resolution in air traffic management (ATM) using physics informed deep reinforcement learning (RL) is proposed. The motivation is to integrate prior physics understanding and model in the learning algorithm to facilitate the optimal policy searching and to present human-explainable results for display and decision-making. First, the information of intruders' quantity, speeds, heading angles, and positions are integrated into an image using the solution space diagram (SSD), which is used in the ATM for conflict detection and mitigation. The SSD serves as the prior physics knowledge from the ATM domain which is the input features for learning. A convolution neural network is used with the SSD images for the deep reinforcement learning. Next, an actor-critic network is constructed to learn conflict resolution policy. Several numerical examples are used to illustrate the proposed methodology. Both discrete and continuous RL are explored using the proposed concept of physics informed learning. A detailed comparison and discussion of the proposed algorithm and classical RL-based conflict resolution is given. The proposed approach is able to handle arbitrary number of intruders and also shows faster convergence behavior due to the encoded prior physics understanding. In addition, the learned optimal policy is also beneficial for proper display to support decision-making. Several major conclusions and future work are presented based on the current investigation.

摘要- 本文提出了一种新颖的航空冲突解决方法，该方法应用于空中交通管理 (ATM) 中，使用物理信息深度强化学习 (RL)。其动机是将先验的物理理解和模型整合到学习算法中，以促进最优策略的搜索，并为显示和决策提供人类可解释的结果。首先，将入侵者的数量、速度、航向角和位置信息整合到解决方案空间图 (SSD) 中，该图在 ATM 中用于冲突检测和缓解。SSD 作为来自 ATM 领域的先验物理知识，是学习的输入特征。使用卷积神经网络与 SSD 图像进行深度强化学习。接下来，构建了一个演员-评论家网络来学习冲突解决策略。本文使用了几个数值示例来说明所提出的方法。探讨了使用物理信息学习的概念来研究离散和连续的 RL。本文详细比较和讨论了所提出算法与基于经典 RL 的冲突解决方法。提出的方法能够处理任意数量的入侵者，并且由于编码了先验物理理解，显示出更快的收敛行为。此外，学到的最优策略也有利于为决策提供适当的显示支持。基于当前研究，提出了几个主要结论和未来工作。

Index Terms- Conflict resolution, deep reinforcement learning, air traffic management.

关键词- 冲突解决，深度强化学习，空中交通管理。

I. INTRODUCTION

I. 引言

TIHE increased airspace density is putting a greater burden on pilots, controllers, and conflict resolution systems such as Center TRACON Automation System (CTAS) [1]. In addition, the recent trends for integration of unmanned aerial systems (UAS) into the national airspace system (NAS) will cause additional complexity for safety. Thus, efficient and scalable conflict resolution method is in critical need to accommodate this challenge. First, conflict resolution in a dense airspace requires the algorithm to be efficient, which makes real-time/in-time decision for safety assurance. Next, the scalability refers to that the algorithm should be robust to the number of air vehicles. This is especially true for UAS which usually includes a large number of unmanned aerial vehicles. In addition, the algorithm should have good resilience to adapt for changing environments and non-cooperative air vehicles. Most existing reinforcement learning methods have to learn from the environment with fixed number of intruders in which the dimension is known as a prior and cannot be changed. Retraining is generally required for scenarios with different number of intruders. As the number of intruders increases, the computational complexity increases quickly for training. To resolve this problem, an image-based deep reinforcement learning method is proposed for conflict resolution. The image-based deep learning largely improve the scalability issue as the algorithm can handle arbitrary number of aircraft as images rather than aircraft states are used. Another major difference is that the proposed model uses both the true observation data, including air vehicles' position, speed and heading angle (e.g., in classical reinforcement learning), and the imaginary data, which is informed by the prior physics knowledge for the conflict detection and resolution. The added data is not a simple manipulation of the real observations, but includes the physics modeling of the prediction and conflict detection using a specified safety bound model. The observations and the physics-informed knowledge are encoded in the images to facilitate the final policy searching. Thus, We name this approach as physics-informed deep reinforcement learning for aircraft conflict resolution hereafter.

空域密度的增加正在给飞行员、管制员以及冲突解析系统 (如中心终端雷达 Approach 控制自动化系统 (CTAS)[1]) 带来更大的压力。此外，近期将无人机系统 (UAS) 融入国家空域系统 (NAS) 的趋势将给安全带来额外的复杂性。因此，迫切需要一种高效且可扩展的冲突解析方法来应对这一挑战。首先，在密集空域中的冲突解析要求算法必须是高效的，以便为安全保证做出实时/及时决策。其次，可扩展性指的是算法应该能够适应空中交通工具数量的变化。这对于通常包含大量无人航空器的 UAS 尤其如此。此外，算法还应具有良好的恢复力，以适应环境变化和非合作航空器。大多数现有的强化学习方法必须从具有固定入侵者数量的环境中学习，其中的维度是已知的先验，并且不能改变。对于具有不同数量入侵者的场景，通常需要重新训练。随着入侵者数量的增加，训练的计算复杂性会迅速增加。为了解决这个问题，提出了一种基于图像的深度强化学习方法用于冲突解析。基于图像的深度学习在很大程度上改善了可扩展性问题，因为算法可以处理任意数量的飞机，使用的是图像而非飞机状态。另一个主要区别是，所提出的模型同时使用真实观测数据 (包括航空器的位置、速度和航向角等，例如在经典强化学习中) 和想象数据，后者是由冲突检测和解析的先验物理知识得出的。添加的数据不仅仅是真实观测的简单操作，还包括使用特定安全边界模型对预测和冲突检测的物理建模。观测数据和物理知识信息编码在图像中，以促进最终策略的搜索。因此，我们在此将这种方法命名为基于物理知识的深度强化学习，用于飞机冲突解析。

A. Related Work

A. 相关工作

Many studies have been done for aircraft conflict resolution and a review can be found in [2]. One group of the most widely used methods is based on the advanced autoresolver investigated by NASA Ames Research Center. A logic of iterating the generated maneuvers to resolve the conflict is presented in [3]. A path-stretch algorithm is proposed in [4], [5] to avoid a conflict in its present course with a turn back to a downstream waypoint. An algorithm for computing horizontal resolution trajectories with a constraint on the bank angle is described in [6], where a set of maneuvers are generated to achieve or exceed the specified minimum separation. The Tactical Separation-Assisted Flight Environment (TSAFE) is designed [7] to alert air traffic controllers of imminent conflicts. An updated version [8] is presented to unify the resolution to three types of separation assurance problems that includes separation conflicts, arrival sequencing, and weather-cell avoidance. A similar encounter-based simulation architecture is presented in [9] for conflict detect-and-avoid modelling.

针对飞机冲突解决已进行了许多研究，相关综述可以在 [2] 中找到。最广泛使用的方法之一是基于美国宇航局艾姆斯研究中心研究的先进自动解决器。在 [3] 中提出了一个迭代生成机动来解决冲突的逻辑。在 [4]、[5] 中提出了一个路径拉伸算法，通过转向下游航点来避免当前航向上的冲突。在 [6] 中描述了计算水平解决轨迹的算法，该算法对银行角有限制，生成一组机动以满足或超过指定的最小间隔。设计了战术分离辅助飞行环境 (TSAFE)[7]，用以提醒空中交通管制员即将发生的冲突。在 [8] 中提出了一个更新版本，用以统一解决三种分离保证问题，包括分离冲突、到达序列和避开气象细胞。在 [9] 中提出了一个类似的遭遇型仿真架构，用于冲突检测与避免建模。

Another major category of methods is based on the force field (also known as voltage potential)

Manuscript received 13 May 2020; revised 11 October 2020, 22 January 2021, and 13 April 2021; accepted 27 April 2021. Date of publication 13 May 2021; date of current version 8 July 2022. This work was supported by the National Aeronautics and Space Administration (NASA) University Leadership Initiative Program (Project Officer: Dr. Anupa Bajwa, Principal Investigator: Dr. Yongming Liu) under Contract NNX17AJ86A. The Associate Editor for this article was D. Sun. (Corresponding author: Yongming Liu.)

手稿于 2020 年 5 月 13 日收到；于 2020 年 10 月 11 日、2021 年 1 月 22 日和 2021 年 4 月 13 日进行了修订；于 2021 年 4 月 27 日接受。发表日期为 2021 年 5 月 13 日；当前版本日期为 2022 年 7 月 8 日。本研究得到了美国国家航空航天局 (NASA) 大学领导力倡议计划 (项目官员:Anupa Bajwa 博士，首席研究员:Yongming Liu 博士) 的支持，合同号为 NNX17AJ86A。本文的副编辑是 D. Sun。(通讯作者:Yongming Liu。)

The authors are with the School for Engineering of Matter, Transport & Energy, Arizona State University, Tempe, AZ 85281 USA (e-mail: pzhao28@asu.edu; yongming.liu@asu.edu).

作者们隶属于美国亚利桑那州立大学物质、运输与能源工程学院，Tempe, AZ 85281 USA(电子邮件:pzhao28@asu.edu; yongming.liu@asu.edu)。

Digital Object Identifier 10.1109/TITS.2021.3077572

数字对象标识符 10.1109/TITS.2021.3077572

methods. By analogy with the positively charged particles pushing away from each other, the authors [10] proposed an approach to calculate the paths of aircraft for conflict resolution. Based on closest point of approach, an improved force field method is proposed to increase the minimum distance between aircraft [11]. The force field method resolves multi-aircraft conflicts by calculating resolution maneuvers for each individual conflict pair separately [12]. To resolve the coordination issues in the unknown behavior in multi-aircraft conflicts, an optimized method was proposed in [13] from the display perspective to assist the operator for conflict resolution, which is subsequently known as solution space diagram (SSD) method and is further extended in [14], [15]. These methods can provide a visualized way for operators to read the conflict resolution method from display and assist them to perform maneuvers. However, lacking of dynamic behavior can cause more conflicts, large path deviation, and long conflict duration [12]. The image produced by the SSD method is implemented to model and predict controllers' deconflict decisions using a supervised learning algorithm [16]. The model trained by this method is individual and scenario sensitive, which is difficult to extend to a general situation. An Optimal Reciprocal Collision Avoidance (ORCA) algorithm [17] was proposed using geometric approach to resolve conflict in a decentralized way. This algorithm guarantees a conflict free solution for the agents with no speed constraint (e.g., robots in horizontal). The algorithm was modified to apply to speed constrained aircraft [18] without guaranteeing conflicts being resolved.

另一个主要的方法类别是基于力场 (也称为电压势) 的方法。通过类比带正电的粒子相互推开，文献 [10] 的作者提出了一种计算飞机冲突解决路径的方法。基于最近点接近法，提出了一种改进的力场方法，以增加飞机之间的最小距离 [11]。力场方法通过分别为每个单独的冲突对计算解决机动来处理多架飞机的冲突 [12]。为了解决多架飞机冲突中未知行为协调问题，文献 [13] 从显示角度提出了一种优化方法，以帮助操作员进行冲突解决，该方法随后被称为解决方案空间图 (SSD) 方法，并在 [14]、[15] 中进一步扩展。这些方法可以为操作员提供一种从显示屏上读取冲突解决方法的可视化方式，并帮助他们执行机动。然而，缺乏动态行为可能导致更多冲突，路径偏移大，冲突持续时间长 [12]。SSD 方法产生的图像被用于使用监督学习算法建模和预测控制员的冲突解决决策 [16]。由这种方法训练的模型是针对个体和场景敏感的，难以扩展到一般情况。提出了一种最优互斥碰撞避免 (ORCA) 算法 [17]，使用几何方法以分布式方式解决冲突。该算法保证了在没有速度约束的代理 (例如，水平面上的机器人) 之间的无冲突解决方案。该算法被修改以应用于有速度约束的飞机 [18]，但不保证冲突能够得到解决。

Reinforcement learning based methods for air traffic management have been investigated recently due to the rapid development in artificial intelligence. Many studies implemented reinforcement learning to avoid air traffic congestion [19]-[22]. The others focus on collision avoidance or conflict resolution. The collision avoidance problem is formulated as Markov Decision Process (MDP) problem in [23], where the generic MDP solvers are used to generate avoidance strategies. Then a dynamic programming approach is proposed in [24] to update the TCAS system that is broadly used in the NextGen (next generation air traffic management system). These methods only tackle the conflict problem between aircraft pair due to the dimensionality problem in multi-aircraft scenarios. To solve this problem, two approaches are proposed in [25] to compress the lookup table. Another method involves decomposing a large multi-agent Markov decision process and fusing their solutions [26]. However, these methods use decentralized training and centralized execution schemes, which still train the policy with single intruder. Thus, the trained policy cannot account for the risk from other intruders. These methods also cannot be extended to continuous action due to the space for heuristic search could be extremely large. To improve the responding to multi-threats, a method is proposed in [27] to improve methods developed in [26] by training corrections for the pair-wise policy. Many methods were also proposed where multiple intruders are directly used to train the policy [28]-[30]. In these methods, fixed number of the nearest intruders are observed to formulate the state for training. Ignorance of the other intruders may lead to risk condition especially in dense airspace. In addition, most of these methods are assumed that the intruders are flying at a fixed velocity and only the own aircraft takes actions to resolve conflicts, which is not practical. Multi-agent reinforcement learning that uses a centralized learning and decentralized execution scheme is proposed in [31]. The learned conflict resolution policies are not in conformance with the preference of air traffic controller in practice. To solve this problem, an interactive conflict solver that combine the controller's demonstrations and AI agent is proposed in [32]. An hierarchical deep reinforcement learning method is also implemented in [33] to solve the conflict resolution problem while performing the air traffic sequencing. However, most of these existing methods train policies with respect to specific number of intruders, which limit the scalibility for different scenarios in practice.

最近，由于人工智能的快速发展，基于强化学习的空中交通管理方法受到了研究。许多研究实现了强化学习以避免空中交通拥堵 [19]-[22]。其他研究则专注于碰撞避免或冲突解决。在文献 [23] 中，碰撞避免问题被构建为马尔可夫决策过程 (MDP) 问题，其中使用了通用的 MDP 求解器来生成规避策略。然后在文献 [24] 中提出了一种动态规划方法来更新 TCAS 系统，该系统在下一代 (NextGen) 空中交通管理

系统中被广泛使用。这些方法仅解决了多机场景中由于维度问题导致的飞机对之间的冲突问题。为了解决这个问题，文献 [25] 中提出了两种压缩查找表的方法。另一种方法涉及分解大型多代理马尔可夫决策过程并融合它们的解决方案 [26]。然而，这些方法使用去中心化训练和集中执行方案，仍然用单个入侵者训练策略。因此，训练出的策略无法考虑到其他入侵者的风险。这些方法也无法扩展到连续动作，因为启发式搜索的空间可能非常大。为了提高对多威胁的反应，文献 [27] 提出了一种改进文献 [26] 的方法，通过为成对策略训练修正来提高方法。还有许多方法直接使用多个入侵者来训练策略 [28]-[30]。在这些方法中，观察到的最近邻入侵者的固定数量被用来构建训练状态。忽视其他入侵者可能导致密集空域中的风险条件。此外，这些方法中的大多数假设入侵者以固定速度飞行，只有自己的飞机采取行动来解决冲突，这在实际中并不实用。文献 [31] 提出了使用集中学习去中心执行方案的多代理强化学习。学到的冲突解决策略与实际中空中交通管制员的偏好不符。为了解决这个问题，文献 [32] 提出了一种结合管制员演示和 AI 代理的交互式冲突求解器。文献 [33] 还实现了一种分层深度强化学习方法来解决冲突解决问题，同时执行空中交通排序。然而，这些现有的大多数方法都是针对特定数量的入侵者来训练策略，这限制了实际中不同场景的可扩展性。

In recent years, many approaches have been developed for reinforcement learning with neural network function approximation. The leading contenders are deep Q-learning [34], ”vanilla” policy gradient methods [35], trust region policy optimization (TRPO) [36], and proximal policy optimization (PPO) method [37]. The PPO method is developed on the basis of TRPO method and outperforms the other methods on efficiency and robustness. Therefore, we use PPO in this proposed method to train the deconflict policy.

近年来，许多方法已经被开发出来用于带有神经网络函数近似的强化学习。主要的竞争者包括深度 Q 学习 [34]，” 香草” 策略梯度方法 [35]，信任域策略优化 (TRPO)[36]，以及近似策略优化 (PPO) 方法 [37]。PPO 方法是在 TRPO 方法的基础上开发的，并在效率和鲁棒性上超过了其他方法。因此，我们在本提议的方法中使用 PPO 来训练解脱策略。

B. Proposed Method

B. 提议方法

A novel method is proposed which integrates the observations and the physics-informed knowledge in images to learn aircraft conflict resolution policy. This is also beneficial for future visual decision-making support from the pilot’s or controller’s screen. Pixels of the screen carry the intruders’ information including number of aircraft, positions, speeds and heading angles that is integrated by the solution space diagram (SSD) method. Most importantly, SSD provides prior physics understanding in identifying the potencial conflict and guidance for de-conflict action determination. This is intrinsically different than existing learning-based methods where direct physical observations are used. A convolution neural network (CNN) is used to extract hidden information from the SSD-based image for deep reinforcement learning to learn the resolution policy. The deep reinforcement learning is based on the Poximal Policy Optimization (PPO) algorithm [37].

提出了一种新颖的方法，该方法将观测数据和图像中的物理知识信息整合起来，以学习飞行器冲突解决策略。这对于飞行员或管制员屏幕上的未来视觉决策支持也是有益的。屏幕上的像素携带入侵者的信息，包括飞机数量、位置、速度和航向角，这些信息通过解决方案空间图 (SSD) 方法进行整合。最重要的是，SSD 为识别潜在冲突和解脱行动确定的指导提供了先验的物理理解。这与现有基于学习的方法本质上是不同的，后者直接使用物理观测。使用卷积神经网络 (CNN) 从基于 SSD 的图像中提取隐藏信息，用于深度强化学习来学习解决策略。深度强化学习基于近似策略优化 (PPO) 算法 [37]。

Most existing reinforcement learning methods for conflict resolution assume the fixed state dimension (such as ACAS X), which have to train the agent in an environment with certain number of intruders. The proposed method does not suffer from this problem as the SSD images can present arbitrary number of intruders. The proposed method provides a new physics-informed learning scheme for the aircraft to learn resolution from prior knowledge and current environment, which fuses different sources of information to assist the decision-making.

大多数现有的用于冲突解决的强化学习方法假设固定状态维度 (例如 ACAS X)，这些方法必须在一个具有特定数量入侵者的环境中训练代理。提议的方法不存在这个问题，因为 SSD 图像可以显示任意数量的入侵者。提议的方法为飞行器提供了一种新的基于物理知识的学习方案，从先验知识和当前环境中学习解决策略，该方案融合了不同来源的信息来辅助决策。

C. Limitation

C. 局限性

No uncertainties are included in the current formulation and further study for probabilistic failure probability (i.e., risk) constraints needs to be included. Only horizontal deconflict is considered and combined vertical-horizontal deconflict needs further investigation.

当前公式中没有包含不确定性，并且需要进一步研究将概率故障概率 (即风险) 约束包含在内。只考虑了水平冲突消除，水平和垂直联合冲突消除需要进一步研究。

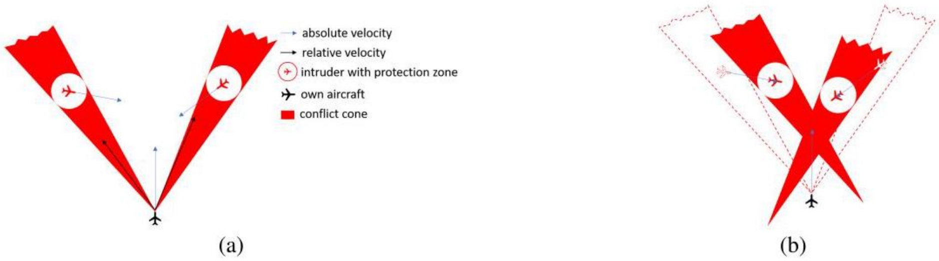


Fig. 1. Illustration of SSD method for conflict detection. (a) Conflicts are detected if the relative velocities within the conflict cone. (b) Conflicts are detected if the absolute velocity vector's end is in the conflict cones that are translated from (a) along their own corresponding intruder's velocity vector.

图 1. SSD 方法用于冲突检测的说明。(a) 如果相对速度在冲突锥内，则检测到冲突。(b) 如果绝对速度向量的端点位于从 (a) 沿着各自对应入侵者的速度向量平移的冲突锥内，则检测到冲突。

D. Summary of Contributions

D. 贡献总结

- A physics-informed learning method is proposed to integrate prior domain knowledge and advanced data analytic method for conflict resolution. The prior knowledge encoded in SSD as images to allow flexible training with reduced samples and faster convergence.

- 提出了一个基于物理信息的学习方法，以整合先验领域知识和先进的数据分析方法用于冲突解决。将编码在 SSD 中的先验知识作为图像，以允许使用更少的样本进行灵活训练并加快收敛速度。

- The proposed learning-based method improves the limitation of classical SSD method in ATM domain, in which only static SSD is used for de-conflict and did not consider the dynamic environment changes.

- 所提出的学习方法改进了经典 SSD 方法在空管领域的局限性，在空管领域中，仅使用静态 SSD 进行冲突消除，并未考虑动态环境变化。

- A meta-control logic is used to add the flight intent information to ensure the small deviation from the original flight plan while performing the conflict resolution.

- 使用元控制逻辑添加飞行意图信息，以确保在执行冲突解决时，偏离原始飞行计划的偏差很小。

II. RESOLUTION POLICY LEARNING FOR SINGLE AIRCRAFT

II. 单架飞机的解决策略学习

A. Prior Domain Knowledge

A. 先验领域知识

Solution space diagram (SSD) method was first proposed as an alternative metric to predict workload for an air traffic controller [38], and further extended as a visual aid for possible airborne separation task on navigational display in the cockpit [39].

解决空间图 (SSD) 方法最初被提出来作为替代指标，用于预测空中交通管制员的工作量 [38]，并进一步扩展为在驾驶舱导航显示上可能的空中分离任务的视觉辅助工具 [39]。

The SSD method as a conflict resolution method is briefly illustrated in Fig. 1a and Fig. 1b. As shown in Fig. 1a, the circle around an intruder represents the protection zone whose radius is the minimum separation between aircrafts. Loss of separation leads to a conflict. The conflict cone is constructed by two lines that is from the own aircraft and is tangent to the protection zone. It is easy to demonstrate that if the vector of relative velocity of the own aircraft is within the conflict cone, conflict will occur if the corresponding intruder keeps its current velocity. To resolve this potential conflict, the relative velocity vector should be moved out of the conflict zone. However, in the multi-intruders scenario, it will be difficult to make an easy decision. This problem can be resolved by translating the relative velocity to the absolute velocity of the own aircraft, which is obtained by summing the relative velocity and the intruder's absolute velocity. The conflict cone is also translated by adding the corresponding intruder's velocity vector to the coordinates of each point, as illustrated in Fig. 1b. This translation moves the end of the relative velocity vector to the end point of the own aircraft's absolute velocity vector. Therefore, it indicates a potential conflict if the end point of the absolute velocity is within the conflict cone area. Then, the multi-aircraft conflicts resolution needs to move the end point of the own aircraft's absolute velocity vector out of the conflict cone. Different predefined rules leads to different deconflict velocity vector. For example, the conflict can be resolved by taking the shortest way out or by only changing heading angle [12].

SSD 方法作为一种冲突解决方法在图 1a 和图 1b 中简要说明。如图 1a 所示，围绕入侵者的圆圈代表保护区，其半径是飞机间的最小间隔。间隔损失会导致冲突。冲突锥由两条从本机出发并与保护区相切的直线构成。可以容易地证明，如果本机的相对速度向量在冲突锥内，如果相应的入侵者保持当前速度，则将发生冲突。为了解决这种潜在的冲突，相对速度向量应移出冲突区。然而，在多入侵者场景中，做出决策将会很困难。这个问题可以通过将相对速度转换为本机的绝对速度来解决，即通过将相对速度与入侵者的绝对速度相加得到。冲突锥也通过将相应的入侵者速度向量加到每个点的坐标上来转换，如图 1b 所示。这种转换将相对速度向量的末端移动到本机绝对速度向量的末端。因此，如果绝对速度的末端在冲突锥区域内，则表示存在潜在的冲突。接着，多飞机冲突解决需要将本机绝对速度向量的末端移出冲突锥。不同的预设规则导致不同的解冲突速度向量。例如，可以通过选择最短路径出冲突区或仅改变航向角来解决问题 [12]。

In this paper, the SSD method is employed as an image generation model to encode the information of intruders into an image, rather than a conflict resolution method that directly resolves conflict. Generation of SSD image is not computational demanding. Thus, there is negligible impact on computational complexity of training process or operation.

在本文中，SSD 方法被用作图像生成模型，将入侵者的信息编码到图像中，而不是直接解决冲突的冲突解决方法。生成 SSD 图像对计算资源的要求不高。因此，对训练过程或操作的计算复杂度影响可以忽略不计。

B. Deep Reinforcement Learning

B. 深度强化学习

The deep reinforcement learning methods are categorized into two groups, value-based methods and policy-based methods. With value-based methods, the agent learns from the environment to maintain an estimate of the optimal action-value function, from which the optimal policy is obtained. Policy-based methods directly learn the optimal policy without having to maintain a separate value function estimate. The policy-based methods estimate the policy gradient as [37], [40]:

深度强化学习方法可以分为两类: 基于价值的方法和基于策略的方法。基于价值的方法中，智能体从环境中学习以维持最优动作价值函数的估计，从而获得最优策略。基于策略的方法直接学习最优策略，而无需维持单独的价值函数估计。基于策略的方法估计策略梯度如下 [37]，[40]:

$$\widehat{g}=\widehat{\mathbb{E}}_t\left[\nabla_\theta \log \pi_\theta\left(a_t \mid s_t\right) \widehat{A}_t\right] \tag{1}$$

where π_θ is a stochastic policy and $\widehat{A}_t$ is an estimator of the advantage function at time t . The expectation $\widehat{\mathbb{E}}_t$ indicates the empirical average over a finite batch of samples. The gradient is rewritten as the following after importance sampling,

其中 π_θ 是一个随机策略，$\widehat{A}_t$ 是时间 t 处优势函数的估计器。期望 $\widehat{\mathbb{E}}_t$ 表示在有限批次的样本上的经验平均值。在重要性抽样之后，梯度重写为以下形式，

$$\widehat{g}=\nabla_\theta \widehat{\mathbb{E}}_t\left[\frac{\pi_\theta\left(a_t \mid s_t\right)}{\pi_{\theta old}\left(a_t \mid s_t\right)} \widehat{A}_t\right] \tag{2}$$

where the θ_{old} is the vector of policy parameters before update. The estimator $\widehat{g}$ is obtained by differentiating the objective:

其中 θ_{old} 是更新前的策略参数向量。估计器 $\widehat{g}$ 通过对目标函数求导获得:

$$L^{CPI}(\theta)=\widehat{\mathbb{E}}_t\left[\frac{\pi_\theta\left(a_t \mid s_t\right)}{\pi_{\theta old}\left(a_t \mid s_t\right)} \widehat{A}_t\right] \tag{3}$$

where the superscript refers to conservative policy iteration [41]. To keep the policy from an excessively large policy update, PPO method uses the following objective function:

其中上标指的是保守策略迭代 [41]。为了防止策略更新过大，PPO 方法使用了以下目标函数:

$$L^{CLIP}(\theta)=\widehat{\mathbb{E}}_t\left[\min \left(k_t(\theta) \widehat{A}_t, \operatorname{clip}\left(k_t(\theta), 1-\epsilon, 1+\epsilon\right) \widehat{A}_t\right)\right]$$

(4)

where $k_t(\theta)$ denotes the probability ratio $k_t(\theta)=\frac{\pi_\theta(a_t|s_t)}{\pi_{\theta old}(a_t|s_t)}$, ϵ is a hyperparameter.

其中 $k_t(\theta)$ 表示概率比 $k_t(\theta)=\frac{\pi_\theta(a_t|s_t)}{\pi_{\theta old}(a_t|s_t)}$ ，ϵ 是一个超参数。

This objective is further augmented by adding an entropy bonus and value function error terms, which is maximized in each iteration:

这个目标函数通过添加熵奖励和价值函数误差项进一步增加，每次迭代中最大化:

$$L_t^{CLIP+VF+S}(\theta)=\widehat{\mathbb{E}}_t\left[L_t^{CLIP}(\theta)-c_1 L_t^{VF}(\theta)+c_2 S\left[\pi_\theta\right]\left(s_t\right)\right]$$

(5)

where c_1, c_2 are coefficients, and S denotes an entropy bonus, and L_t^{VF} is a squared-error loss $\left(V_\theta\left(s_t\right)-V_t^{\text{targ}}\right)^2$

其中 c_1, c_2 是系数，S 表示熵奖励，L_t^{VF} 是平方误差损失 $\left(V_\theta\left(s_t\right)-V_t^{\text{targ}}\right)^2$ 。

C. Action Space

C. 动作空间

Main maneuvers for conflict resolution in horizontal decon-flict includes heading angle change and speed control. Altitude changing is also an efficient way to resolve conflict, but is not considered in this study. In practice, combination of the heading angle change and speed control is not implemented by the pilot and controller due to high workload. A fully automated system may use both these maneuvers simultaneously to make the resolution more efficient. In this work, we first investigate the heading angle maneuvers and speed control separately (i.e., related to the current ATM practice), and then combine them together (i.e., related to a fully automated system in the future).

水平冲突解决的主要机动包括改变航向角和速度控制。改变高度也是解决冲突的有效方式，但在本研究中不予考虑。在实际操作中，由于工作负荷较高，飞行员和管制员并未实施航向角改变和速度控制的组合。完全自动化的系统可能会同时使用这两种机动，以提高解决效率。在这项工作中，我们首先分别

研究航向角机动和速度控制 (即与当前空中交通管理实践相关)，然后将它们结合起来 (即与未来的完全自动化系统相关)。

1) Discrete Action Space: Discrete action space refers to a set $\mathbb{D}$ that contains finite number of fixed actions. For example, $\mathbb{D}=[-10^{\circ}, 0^{\circ}, 10^{\circ}]$ represents using heading angle change of $\pm 10^{\circ}$ or 0° to resolve conflicts. Note that the action space is not confined to heading angle change, speed change, and simultaneous speed change and heading change are also applicable as long as the actions are finite and fixed.

1) 离散动作空间: 离散动作空间指的是包含有限个固定动作的集合 $\mathbb{D}$ 。例如，$\mathbb{D}=[-10^{\circ}, 0^{\circ}, 10^{\circ}]$ 表示使用 $\pm 10^{\circ}$ 或 0° 的航向角改变来解决冲突。注意，动作空间不仅限于航向角改变，速度改变以及同时的速度改变和航向改变也适用，只要动作是有限和固定的。

2) Continuous Action Space: Continuous action space refers to a continuous space from which the agent chooses deconflict actions. In this work, two kinds of continuous space are considered, which are heading angle and speed. For example, the heading angle action space $(-\pi/2, \pi/2)$ allows the agent choose any heading angle change from $-\pi/2$ to $\pi/2$, the speed control action space(-50knots,50knots)implies the agent can choose any value from -50 knots to50knots to add to its current speed.

2) 连续动作空间: 连续动作空间指的是智能体选择冲突解决动作的连续空间。在这项工作中，考虑了两种连续空间，分别是航向角和速度。例如，航向角动作空间 $(-\pi/2, \pi/2)$ 允许智能体选择从 $-\pi/2$ 到 $\pi/2$ 的任何航向角改变，速度控制动作空间 (-50 海里/小时，50 海里/小时) 意味着智能体可以选择从-50 海里/小时到 50 海里/小时的任何值来增加其当前速度。

D. Reward Function

D. 奖励函数

The reward function of the proposed method is formulated to balance the objectives of safety assurance and minimized disruption. Safety is the most critical consideration in the proposed method. The reward with respect to safety is expressed as:

所提出方法的奖励函数被制定为平衡安全保证和最小化干扰的目标。安全是所提出方法中最关键的考虑因素。关于安全的奖励表示为:

$$R_c=\begin{cases}-1 & \text{if conflict} \\ 0 & \text{otherwise}\end{cases} \tag{6}$$

where conflict is defined as the condition that the distance between two aircraft is less than a specific value (5 nautical miles as specified by International Civil Aviation Organization (ICAO)).

冲突定义为两架飞机之间的距离小于特定值 (国际民用航空组织 (ICAO) 规定的 5 海里) 的条件。

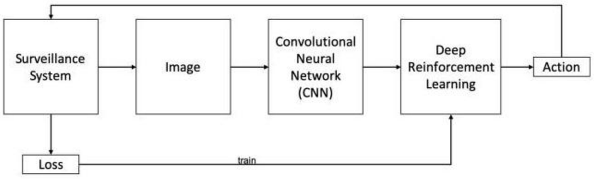

Surveillance System
Image
Convolutional Neural Network (CNN)
Deep Reinforcement Learning
Action
Loss
train

Fig. 2. Flow chart of the proposed conflict resolution algorithm.

图 2. 提出的冲突解决算法流程图。

The heading-change action will cause the aircraft to deviate from its original flight plan. A term of the reward function to minimize disruption is defined as:

航向改变动作将导致飞机偏离其原始飞行计划。奖励函数中定义了一个用于最小化中断的项为:

$$R_h=0.001 \cos (\theta) \tag{7}$$

where θ is the angle between the action direction and the intention direction. This term rewards the action following the intention and penalizes the action of deviation.

其中 θ 是动作方向与意图方向之间的角度。这个项奖励遵循意图的动作，并对偏离的动作进行惩罚。

The speed change action makes no spatial deviation from the original flight plan, but deviating from the cruise speed is fuel consuming. A term of the reward function is defined to reward moderate speed change:

速度改变动作不会使飞机在原始飞行计划上产生空间偏移，但偏离巡航速度会消耗燃料。奖励函数中定义了一个用于奖励适度速度改变的项:

$$R_s = 0.001e^{-\left(\frac{v-v_0}{v_u-v_l}\right)^2} \tag{8}$$

where v is the current speed, v_0 is the original speed, v_l and v_u are the lower bound and upper bound of the speed respectively.

其中 v 是当前速度，v_0 是原始速度，v_l 和 v_u 分别是速度的下限和上限。

The total reward is expressed as:

总奖励表示为:

$$R = \begin{cases} R_c + R_h & \text{heading change} \\ R_c + R_s & \text{speed change} \\ R_c + R_d + R_s & \text{simultaneous heading and speed} \end{cases} \tag{9}$$

E. Network Architecture

E. 网络架构

The network architecture for learning is illustrated in Fig. 2. Information from the surveillance systems is processed first to generate images based on SSD method. Next, convolutional neural network is used to extract features and output the state vector for learning. The deep reinforcement learning process is performed to select an action and interact with the environment. Then a loss calculated by the PPO method is used to train the learning network. This process is iterated until the result converges to a predefined level.

学习用的网络架构在图 2 中有所说明。首先，来自监视系统的信息通过 SSD 方法处理以生成图像。然后，使用卷积神经网络提取特征并输出学习用的状态向量。深度强化学习过程用于选择一个动作并与环境交互。接着，使用 PPO 方法计算出的损失用于训练学习网络。这个过程一直迭代，直到结果收敛到预定义的水平。

1) Architecture of Convolutional Neural Network (CNN): The SSD-based image contains information of number, speeds, headings of the intruders, and indications for potential conflict identification criteria, which are critical for conflict resolution. These information should be extracted for the deep reinforcement learning algorithm to learn the resolution policy. Two hidden layers using convolutional neural network are used for this purpose in this work, which is illustrated in Fig. 3. The main parameters are listed in Table I. As shown in Fig. 3, the input image is 80 × 80 pixels, which are produced by the SSD method. The first layer output $4 \times 38 \times 38$ pixels, followed by the second layer that outputs $16 \times 9 \times 9$ pixels. The final output is a 1×1296 vector as the input for the following neural network for deep reinforcement learning. Through this process, the vector contains all the intruders' information extracted by the neural network.

1) 卷积神经网络 (CNN) 架构: 基于 SSD 的图像包含入侵者的数量、速度、航向以及潜在的冲突识别标准的信息，这些信息对于冲突解决至关重要。这些信息应当被提取出来，以供深度强化学习算法学习解决策略。在本工作中，为此目的使用了两个使用卷积神经网络的隐藏层，这在图 3 中有所说明。主要参数列在表 I 中。如图 3 所示，输入图像是 80×80 像素，由 SSD 方法生成。第一层输出 $4 \times 38 \times 38$ 像素，随后是第二层，输出 $16 \times 9 \times 9$ 像素。最终输出是一个 1×1296 向量，作为后续用于深度强化学习的神经网络的输入。通过这个过程，向量包含了由神经网络提取的所有入侵者的信息。

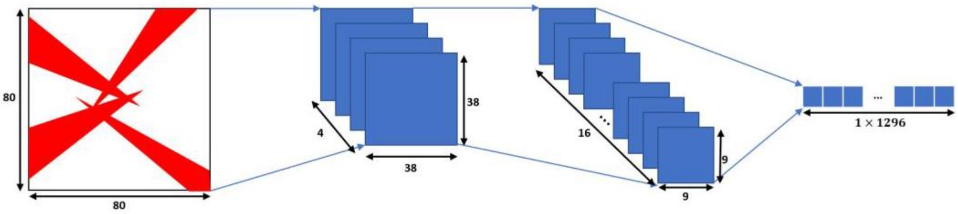


Fig. 3. Illustration of two layers convolution neural network for features extraction. The output of this neural network is an vector which is the state of the following reinforcement learning.

图 3. 两个层次卷积神经网络的特征提取说明。该神经网络的输出是一个向量，它是后续强化学习的状态。

TABLE I
HYPERPARAMETERS FOR CNN
CNN 的超参数

	Layer 1	Layer 2
Input Depth	1	4
Output Depth	4	16
Kernel Size	6	6
Stride	2	4
Padding	0	0
Bias	False	True

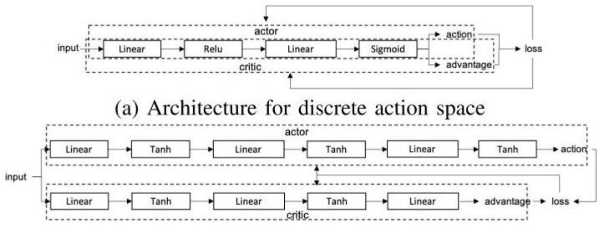
	第一层	第二层
输入深度	1	4
输出深度	4	16
卷积核大小	6	6
步长	2	4
填充	0	0
偏置	假	真

2) Architecture of Deep Reinforcement Learning Neural Network: The output of CNN is the input of the deep reinforcement learning neural network that consists of several layers and activations. We denote the output of CNN at time t as s_t that is the state for reinforcement learning. The deep reinforcement learning neural network takes the state as input and output the policy $\pi\left(a_t \mid s_t ; \theta_{\pi_{old}}\right)$. The action is taken following the policy and the environment will return a reward r_t and the next-state s_{t+1} . A T time steps trajectory is collected in this manner, which is denoted as $\left\{\left(s_t, r_t\right),\left(s_{t+1}, r_{t+1}\right), \cdots,\left(s_T, r_T\right)\right\}$. Fig. 4 presents two architectures of neural network used to train policies of discrete action and continuous action, respectively. Fig. 4a shows the neural network for discrete action, where a hidden layer is connected and followed by the output layer that is the probabilities of the actions. Each node of the output layer corresponds to an action, associated with probability of this action being selected. The probability distribution of the actions is discrete, and the action is selected by sampling from the probability distribution. The selected action is used for the agent to interact with the environment and a loss is calculated using the PPO method described previously. Note that even though there are not two separate networks used to respectively estimate the value function and policy in the network for discrete action space, it also belongs to actor-critic style (see [40]).

2) 深度强化学习神经网络的架构: 卷积神经网络 (CNN) 的输出是深度强化学习神经网络的输入，该网络由数层及激活函数组成。我们表示时间 t 的 CNN 输出为 s_t ，这是强化学习的状态。深度强化学习神经网络将状态作为输入，输出策略 $\pi\left(a_t \mid s_t ; \theta_{\pi_{old}}\right)$ 。根据策略采取行动，环境将返回一个奖励 r_t 和下一个状态 s_{t+1} 。以这种方式收集了 T 时间步长的轨迹，表示为 $\left\{\left(s_t, r_t\right),\left(s_{t+1}, r_{t+1}\right), \cdots,\left(s_T, r_T\right)\right\}$ 。图 4 展示了两种用于训练离散动作和连续动作策略的神经网络架构。图 4a 显示了用于离散动作的神经网络，其中一个隐藏层连接后跟输出层，输出层是动作的概率。输出层的每个节点对应一个动作，与选择该动作的概率相关联。动作的概率分布是离散的，通过从概率分布中抽样来选择动作。选择的动作用于智能体与环境的交互，并使用前面描述的 PPO 方法计算损失。注意，尽管在离散动作空间的网络中没有分别使用两个独立的网络来估计价值函数和策略，它也属于演员-评论家风格 (参见 [40])。

The training network architecture for continuous action space is shown in Fig. 4b, where the output of the CNN network is connected to an actor-critic network. The actor-critic network is widely used for continuous action space, where the critic network is trained to approximate the value function while the actor network is trained to approximate the policy. The activation (tanh) in the output layer of the actor network is used to generate an continuous action. The actor network is trained using Monte-Carlo method that has no bias but suffers from large variance. The critic network is trained using boot-strap (also named as ”temporal difference”) method that has small variance but introduces bias [42]. Combining these two networks makes the training process to be more efficient and robust.

连续动作空间的训练网络架构如图 4b 所示，其中 CNN 网络的输出连接到一个演员-评论家网络。演员-评论家网络在连续动作空间中被广泛使用，其中评论家网络被训练来近似值函数，而演员网络被训练来近似策略。演员网络输出层的激活函数 (tanh) 用于生成连续动作。演员网络使用蒙特卡洛方法进行训练，该方法无偏但方差较大。评论家网络使用自举方法 (也称为“时间差分”) 进行训练，该方法方差小但引入了偏差 [42]。结合这两个网络使得训练过程更加高效和健壮。



(a) Architecture for discrete action space

(b) Architecture for continuous action space
(b) 连续动作空间的架构
Fig. 4. Architectures of deep reinforcement learning neural network.
图 4. 深度强化学习神经网络的架构。
The advantage function used for both the two architectures is expressed as [40]:
用于这两种架构的优势函数表示为 [40]:

$$\widehat{A}_t = \sum_{i=t}^{T-1} \gamma^{i-t} r_i + \gamma^{T-t} V\left(s_T\right) - V\left(s_t\right) \tag{10}$$

where $V(s)$ is the state value.
其中 $V(s)$ 是状态值。

F. Meta Control Logic for Returning to Intention

F. 返回意图的元控制逻辑

In this work, we apply a meta control logic to let the aircraft return to its own waypoint after deconflict. The logic is performed on the basis of conflict detection. As we have mentioned previously, it indicates a potential conflict if the end point of the absolute velocity of the own aircraft is in the conflict cone area. In this situation, the aircraft must perform the deconflict behavior to avoid the risk. Therefore, the meta control logic is designed as: aircraft keeps heading to its intention until the end point of its velocity vector inserts the conflict cone, then a resolution velocity is selected. At each time step of resolution, the intention velocity is monitored. The intention velocity refers to a vector pointing to the next waypoint from the current position. The magnitude of the intention velocity would be the same with the current speed. Once the end point of the intention velocity moves out of the conflict cone, the aircraft will choose the intention velocity to return to its original flight plan. This process is illustrated by the following pseudo-code and Fig. 5, where the green arrow represents the intention velocity and the black arrow denotes the resolution velocity.

在这项工作中，我们应用了一种元控制逻辑，让飞机在冲突解除后返回到自己的航点。该逻辑是基于冲突检测来执行的。如我们之前提到的，如果自身飞机的绝对速度终点位于冲突锥区域内，则表明存在潜在的冲突。在这种情况下，飞机必须执行冲突解除行为以避免风险。因此，元控制逻辑被设计为: 飞机保持朝向其意图方向飞行，直到其速度向量的终点进入冲突锥，然后选择一个解决速度。在解决的每个时间步，都会监控意图速度。意图速度指的是从当前位置指向下一个航点的向量。意图速度的大小将与当前速度相同。一旦意图速度的终点移出冲突锥，飞机将选择意图速度以返回到其原始飞行计划。这个过程由以下伪代码和图 5 说明，其中绿色箭头代表意图速度，黑色箭头表示解决速度。

Algorithm 1 Meta Controller
算法 1 元控制器

```
Result: $v$ : velocity of next step
while run do
   obs. ← observe airspace
   states ← SSDProcesse(obs.)
   velocity for resolution: $v_r$ ← Policy(states)
   velocity for return: $v_i$ ← intention velocity
   conflict detection for $v_i$
   if conflict then
```

```
        $v \leftarrow v_r$ (black arrow in Fig. 5)
    else
        $v \leftarrow v_i$ (green arrow in Fig. 5)
    end
end
```

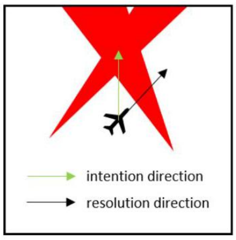


Fig. 5. Illustration of meta control logic. The resolution action is selected if a potential conflict is detected (the end of the green vector is within the red area).

图 5. 元控制逻辑的说明。如果检测到潜在的冲突 (绿色向量的末端在红色区域内)，则选择解决动作。

The above process is initiated when the en-route phase starts. The algorithm will perform in the entire en-route phase to detect and resolve potential conflict. SSD image can be used to identify whether a potential conflict exists in the current direction. Once detected, the conflict resolution process will be triggered. If the conflict is resolved, the aircraft will head forward its next waypoint following the above returning process. This horizontal conflict resolution terminates when the en-route phase ends. If the algorithm fails and a conflict is happening in a short time (e.g., 2 minutes or less), the TCAS system or other conflict/collision avoidance system performing in vertical dimension will take charge of decision making and the algorithm will be suspended.

上述过程在航路阶段开始时启动。算法将在整个航路阶段执行，以检测并解决潜在的冲突。SSD 图像可用于识别当前方向是否存在潜在冲突。一旦检测到冲突，将触发冲突解决过程。如果冲突得到解决，飞机将按照上述返回过程前往下一个航点。这种水平冲突解决在航路阶段结束时终止。如果算法失败，并且在短时间内 (例如 2 分钟或更少) 发生冲突，TCAS 系统或其他在垂直维度执行的冲突/碰撞避免系统将负责决策制定，算法将暂停。

In the process of resolution, the controller will not know in advance when the resolution maneuvers will be completed. The conflict resolution using reinforcement learning (rather than optimization) is a one-step method that gives a strategy with respect to the current state. The benefit is that it can cope with uncertainty of the environment. Therefore, the method will not predict when the resolution maneuvers will be completed. The aircraft will inform the controller when no conflict is detected after resolution.

在解决过程中，控制器无法预先知道解决机动何时完成。使用强化学习 (而不是优化) 的冲突解决是一步方法，它给出了关于当前状态的策略。其优点在于能够应对环境的不确定性。因此，该方法不会预测解决机动何时完成。飞机将在解决后未检测到冲突时通知控制器。

III. MULTI-AGENT COORDINATION

III. 多代理协调

Cooperatively resolving conflicts by all affected aircraft is safer and more practical. This cooperation can be arranged by a computation center (ground center or one of the effected aircraft). When conflicts are detected, all affected aircraft need to communicate with the center to upload their conflict information. The center will determine a minimum-pilot-disturbance cooperative strategy. This strategy can be beneficial as it reduces the usage of limited communication resource, reduces the pilot disturbance, and reduces the risk due to the uncertainty of cooperative operations.

所有受影响飞机合作解决冲突更为安全且实用。这种合作可以通过计算中心 (地面中心或受影响飞机之一) 来安排。当检测到冲突时，所有受影响的飞机需要与中心通信，上传它们的冲突信息。中心将确定一种最小飞行员干扰的合作策略。这种策略可能是有益的，因为它减少了有限通信资源的使用，减少了飞行员的干扰，并降低了由于合作操作不确定性带来的风险。

Aircraft uses SSD method to detect potential conflict intuitively by identifying whether the end point of its absolute velocity is within the other aircraft's conflict cone. Once a conflict is detected, all the aircraft that involve in the conflict and the associated conflict should be coupled together, for example, aircraft a_1 detects a potential conflict with a_2 , while a_3 also detects a conflict with a_2 , then all the three aircraft should be requested to cooperate. Once all affected aircraft have been identified, the computation center will determine the aircraft and the order to perform conflict resolution according to the following criteria.

飞机使用 SSD 方法通过直观地识别其绝对速度的终点是否在另一架飞机的冲突锥内来检测潜在的冲突。一旦检测到冲突，所有涉及冲突的飞机以及相关的冲突应该被耦合在一起，例如，飞机 a_1 检测到与 a_2 的潜在冲突，而 a_3 也检测到与 a_2 的冲突，那么这三架飞机都应该被要求进行合作。一旦确定了所有受影响的飞机，计算中心将根据以下标准确定飞机及其执行冲突解决的顺序。

1) : Aircraft may not respond to the cooperation request due to communication failures or human errors. In this situation, the other aircraft is responsible to avoid conflict with the non-cooperative one. Therefore, the aircraft that has conflict with the non-cooperative aircraft must take actions.

1) 飞机可能由于通信故障或人为错误而不响应合作请求。在这种情况下，其他飞机负责避免与非合作的飞机发生冲突。因此，与非合作飞机有冲突的飞机必须采取行动。

2): If the conflict is detected between an aircraft pair, which is the most common case, the one that has a greater q-value should take actions. In reinforcement learning, the q-value is the estimated reward of the future in the episode if the agent takes a specific action at the current state. In the studied scenario, according to the definition of reward function in the previous section, the q-value here reflects the possibility of successful conflict resolution. Therefore, it is safer to require the aircraft with greater q-value to take actions.

2) 如果在飞机对之间检测到冲突，这是最常见的案例，q 值较大的飞机应采取行动。在强化学习中，q 值是如果代理在当前状态采取特定行动，那么在剧集的未来中估计的奖励。在研究的场景中，根据上一节中定义的奖励函数，这里的 q 值反映了成功解决冲突的可能性。因此，要求 q 值较大的飞机采取行动更为安全。

3) : Fig. 6 gives two typical scenarios of multiple aircraft conflict. In the Fig. 6a, there exists at least one aircraft subset in which each of the aircraft has no conflict with others. The subsets can be easily identified by iterating all the affected aircraft to find their no-conflict sets. Among all the subsets, the one that contains maximum number of aircraft is used for efficiency. For example, in Fig. 6a, the maximum no-conflict set is $\{a_1, a_2\}$. Following this, aircraft in this subset will keep their original flight plans and the other takes actions for conflict resolution. If more than one aircraft are needed to take actions, they should act cooperatively. A time interval is assigned for each aircraft to take actions in rotation. The increasing order of the q-value should be used since smaller q-value indicates more risk of conflict according to the cost function, and thus the immediate actions should be taken. While the rotating aircraft is taking its action, the other aircraft need to keep their current velocity until their turns for taking actions. This method resolves conflicts with minimum pilot disturbance.

3) : 图 6 展示了多架飞机冲突的两个典型场景。在图 6a 中，至少存在一个飞机子集，该子集中的每架飞机与其他飞机均无冲突。可以通过遍历所有受影响的飞机来找到它们的无冲突集合，从而轻松识别这些子集。在所有子集中，包含飞机数量最多的子集被用于提高效率。例如，在图 6a 中，最大的无冲突集合是 $\{a_1, a_2\}$ 。接下来，该子集中的飞机将保持其原始飞行计划，而其他飞机则采取措施解决冲突。如果有多个飞机需要采取措施，它们应该合作行动。为每架飞机分配一个时间间隔，轮流采取行动。由于较小的 q 值根据成本函数表示更高的冲突风险，因此应使用 q 值的升序，立即采取行动。当一架飞机正在采取行动时，其他飞机需要保持当前速度，直到它们轮到采取行动。这种方法在最小化飞行员干扰的情况下解决冲突。

Fig. 6b presents an example where no-conflict subset does not exist, although this scenario rarely happen in practice. In this condition, only one aircraft is allowed to keep its original flight plan. The largest and/or heaviest aircraft has the priority to stay in its flight plan since high action cost and low agility. The other aircraft need to take actions in a time interval alternatively, using the same manner as we have demonstrated in the above paragraph.

图 6b 展示了一个无冲突子集不存在的例子，尽管这种情况在实际情况中很少发生。在这种情况下，只允许一架飞机保持其原始飞行计划。最大和/或最重的飞机有优先权保持在飞行计划中，因为它们的行动成本高且灵活性低。其他飞机需要轮流在时间间隔内采取行动，方式与上文所示相同。

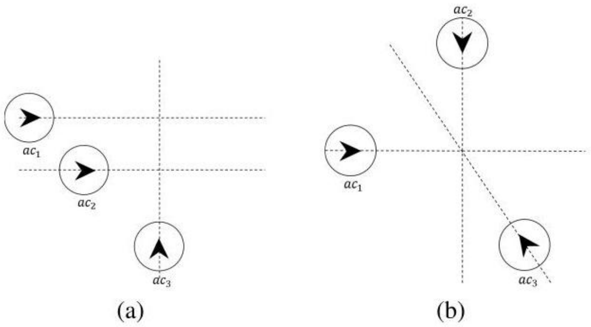


Fig. 6. Two typical scenarios of multiple aircraft conflict.
图 6. 多架飞机冲突的两个典型场景。

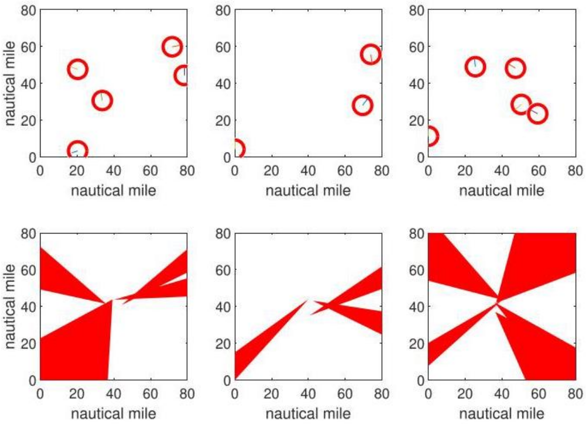


Fig. 7. Samples of environment image showing intruders' positions, protection zones and SSD representation.
图 7. 环境图像样本，显示入侵者位置、保护区和 SSD 表示。

Short interval can decreases conflict risk since each aircraft can have many opportunities to adjust its actions to avoid conflict. But the short interval also increases the pilot workload.

短时间间隔可以降低冲突风险，因为每架飞机都有多次机会调整其行动以避免冲突。但短时间间隔也会增加飞行员的负担。

IV. NUMERICAL EXAMPLES

IV. 数值示例

A. Training Data

A. 训练数据

The environment is developed using Pygame. A 80 × 80 nautical miles surveillance area is simulated where the own aircraft is located at the center. Every $0 \sim 150$ seconds, an intruder is created on a random location of the border and then flies along a line with random direction. The intruder is removed if it flies beyond the scope. The intruder's speed is randomly set as $400 \sim 500$ knots according to the typical cruise speed of commercial aircraft. A time step in the simulation corresponds to 40 seconds in practice, and an episode consists of 200 time steps. Environment reset is not needed at the beginning of each episode, due to the randomly set of intruders. The training is terminated after 500 episodes that is sufficient for convergence.

环境是使用 Pygame 开发的。模拟了一个 80 × 80 海里的监控区域，其中自己的飞行器位于中心。每 $0 \sim 150$ 秒，在边界的一个随机位置创建一个入侵者，然后沿着一个随机方向的直线飞行。如果入侵者飞

出了范围，则将其移除。入侵者的速度被随机设置为 400 ∼ 500 节，根据商用飞机的典型巡航速度。模拟中的一个时间步对应实际中的 40 秒，一个回合由 200 个时间步组成。由于入侵者的随机设置，每个回合开始时不需要重置环境。训练在经过 500 个回合后终止，这足以达到收敛。

As presented in the first row of Fig. 7, three images are sampled from the dynamic environment to illustrate the simulated airspace. The second row images are the corresponding SSD airspace representations which are the inputs of the training network.

如图 7 第一行所示，从动态环境中抽取了三张图像来展示模拟的空域。第二行的图像是对应的 SSD 空域表示，这是训练网络的输入。

Fig. 8 gives the conflict number per two flight hours if the own aircraft keeps its original flight plan in this airspace.

图 8 给出了如果自己的飞行器保持原飞行计划在这个空域中，每两个飞行小时的冲突数量。

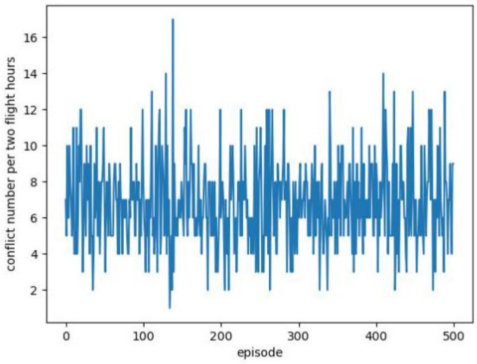


Fig. 8. Number of conflicts if no resolution is performed in the constructed environment.

图 8. 如果在构建的环境中不执行解决措施，冲突的数量。

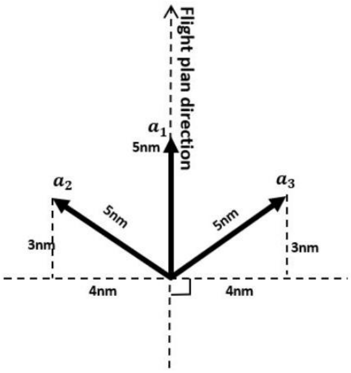


Fig. 9. Discrete action space.

图 9. 离散动作空间。

B. Action Space

B. 动作空间

Several categories of action space are studied, including discrete heading angle action space, continuous heading angle action space, speed action space, and simultaneous heading change and speed control action space, which are depicted in the following:

研究了动作空间的几个类别，包括离散航向角动作空间、连续航向角动作空间、速度动作空间以及同时航向变化和速度控制动作空间，以下进行了描述:

1) Discrete Heading Angle Action Space: It is easy for pilots or controllers to perform conflict resolution using actions selected from discrete action space. An example action space is shown in Fig. 9, which includes three actions a_1, a_2, a_3 . These actions are to move 5 nautical miles over a time interval

in different directions illustrated in Fig. 9. The three actions are used because the current deconflict system also uses few number of actions and we want to be consistent with the current practice. More actions will lead to increased action space and training time.

1) 离散航向角动作空间: 飞行员或控制器可以轻松使用从离散动作空间中选择的行为来执行冲突解决。一个示例动作空间如图 9 所示，其中包含三个动作 a_1, a_2, a_3 。这些动作是在图 9 所示的不同方向上，在一段时间内移动 5 海里的动作。选择这三个动作是因为当前的冲突解脱系统也使用少量动作，我们希望与当前实践保持一致。更多的动作将导致动作空间的增加和训练时间的延长。

2) Continuous Heading Angle Action Space: Continuous action space leads to a more smooth resolution path. Though the continuous action is not practical for the pilots or controllers to operate manually, the automated operation of manned or unmanned aircraft can implement the continuous action space to resolve conflict. In this work, we use the continuous action space where the selected direction should have an angle with the original direction no greater than $\frac{\pi}{2}$. An additional reward (Eq. 7) is added to make sure the deviation between the selected direction and the intention direction is small if not necessary. The reward decreases with the angle between the two directions increases, and should be much smaller than 1 to prevent affecting the deconflict policy learning. This reward can keep the aircraft from deviating its flight plan significantly when performing conflict resolution.

2) 连续航向角动作空间: 连续动作空间导致更平滑的解决路径。尽管连续动作对飞行员或控制器手动操作来说不切实际，但有人或无人飞机的自动操作可以实施连续动作空间来解决冲突。在这项工作中，我们使用连续动作空间，其中选定方向与原始方向的角度不应大于 $\frac{\pi}{2}$ 。为了确保选定方向与意向方向之间的偏差较小 (如果不是必要的)，增加了一个额外的奖励 (式 7)。当两个方向之间的角度增加时，奖励会减少，并且应该远小于 1，以防止影响冲突解脱策略的学习。这种奖励可以保持飞机在执行冲突解决时，不显著偏离其飞行计划。

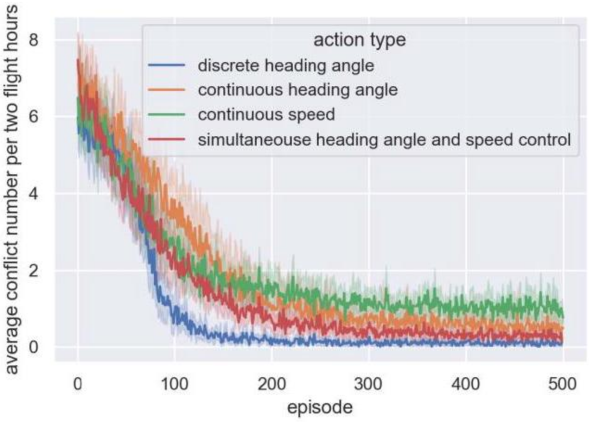


Fig. 10. Convergence speed comparison between different action types.

图 10. 不同动作类型之间的收敛速度比较。

3) Speed Control: Another conflict resolution method is to change the aircraft's speed while keeping its original direction. In this example we use continuous action space, which is 300 ∼ 600 knots (the intruders' speeds are randomly chosen from 400 ∼ 500 knots).

3) 速度控制: 另一种冲突解决方法是改变飞机的速度，同时保持其原始方向。在这个示例中，我们使用连续动作空间，即 300 ∼ 600 节 (入侵者的速度是从 400 ∼ 500 节中随机选择的)。

4) Simultaneous Heading Angle and Speed Control: Combining the heading angle changing and speed control extends the action space from 1 dimension to 2 dimensions, which provides more options for conflict resolution.

4) 同时航向角和速度控制: 结合航向角改变和速度控制，将动作空间从一维扩展到二维，为冲突解决提供了更多选项。

The results of convergence behavior for training the neural network using the above four action space categories are presented in Fig. 10. Each of the curve converges to a low level, indicating the average conflict number per two flight hours reduces in the training process. The discrete action space performs best among the four categories, but limited actions restrict feasibility of the agent. Continuous action space allows the agent performing conflict resolution in a broader range. The agent using speed control action space keeps the original direction and avoids conflicts by changing speed. However, both the continuous action space and speed control performs lower comparing with the discrete action space. The action space of simultaneous heading change and speed control shows better performance than any single control method and closes to the level of discrete action space.

使用上述四种动作空间类别训练神经网络的收敛行为结果展示在图 10 中。每条曲线都收敛到一个低水平，表明在训练过程中每两小时平均冲突数减少。在四种类别中，离散动作空间表现最佳，但有限的动作限制了智能体的可行性。连续动作空间允许智能体在更广泛的范围内进行冲突解决。使用速度控制动作空间的智能体保持原有方向，并通过改变速度避免冲突。然而，连续动作空间和速度控制的表现都低于离散动作空间。同时改变航向和速度的动作空间表现优于任何单一控制方法，并接近离散动作空间的水平。

C. Path of Deconflict and Returning

C. 冲突解脱与返回路径

Implementation in practice needs the algorithm to include the aircraft's intention. The aircraft should return to its original flight plan when conflicts are resolved. To achieve this objective, a high level control mechanism is used. This mechanism consists of three part: conflict detection, conflict resolution, and flight plan return. In practice, aircraft always performs conflict detection. Once conflicts are detected, the conflict resolution function is triggered. If no conflict is detected when resolution is completed, the flight-plan-return logic is implemented.

在实践中实施需要算法包含飞机的意图。当冲突解决后，飞机应返回其原始飞行计划。为了达到这个目标，使用了一个高级控制机制。该机制由三部分组成: 冲突检测、冲突解决和飞行计划返回。在实际中，飞机总是执行冲突检测。一旦检测到冲突，就会触发冲突解决功能。如果在完成解决后未检测到冲突，则实施飞行计划返回逻辑。

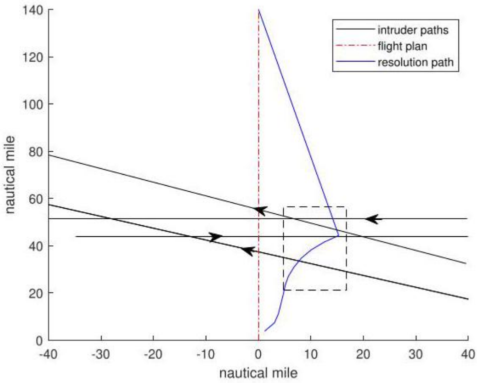


Fig. 11. Paths of the intruders and the own aircraft's resolution.

图 11. 入侵者和本架飞机解决路径。

Fig. 11 presents an example, where four intruders have potential conflicts with the own aircraft in its flight plan. The blue line presents the path of the aircraft to resolve these conflicts using continuous heading change action. The dynamic resolution processes in the dashed rectangle are shown in Fig. 12, where the blue circle area represents the intruder's protection zone, the green arrow indicates the velocity for intention, and the black arrow is the velocity for resolution. The agent selects the resolution action from time step 1 to time step 8 . At time steps 9 to 12, the potential conflicts have been resolved, thus the return action is selected. Note that the continuous changes in heading is difficult for the controllers to observe and high workload for the pilots to perform. The continuous action space may be applicable for the unmanned aircraft system traffic management (UTM) or the future fully automated systems.

图 11 展示了一个示例，其中四个入侵者可能与自己的飞机在飞行计划中存在潜在的冲突。蓝线展示了飞机使用连续航向改变动作来解决这些冲突的路径。动态解决过程在虚线矩形中显示，如图 12 所示，其中蓝色圆形区域代表入侵者的保护区，绿色箭头表示意图速度，黑色箭头是解决速度。代理从时间步 1 到时间步 8 选择解决动作。在时间步 9 到 12，潜在的冲突已经被解决，因此选择了返回动作。注意，连续的航向变化对控制器来说难以观察，对飞行员来说工作量很大。连续动作空间可能适用于无人航空器系统交通管理 (UTM) 或未来的完全自动化系统。

D. Multi-Agent Coordination

D. 多代理协调

The proposed method can be extended to coordinate multiple aircraft in conflict resolution. This section uses three multiple aircraft encounter examples illustrated in Fig. 13, Fig. 14, and Fig. 15 to verify the effectiveness of the proposed method. The 3 aircraft encounter example is also used to compare the proposed method with the existing CSORCA method and value-based learning method.

所提出的方法可以扩展到协调多个飞机在冲突解决中的动作。本节使用图 13、图 14 和图 15 中的三个多飞机遭遇示例来验证所提出方法的有效性。3 架飞机遭遇示例也用于将所提出的方法与现有的 CSORCA 方法以及基于价值的学习方法进行比较。

As mentioned in Section III, in most conditions, some of the aircraft keep their original flight plans according to the minimum pilot disturbance policy, while the others have the responsibilities for conflict resolution. In the 3-aircraft example, at least two of the three aircraft have to take actions in order to resolve conflicts. The aircraft 3 (”ac3”) is selected to keep its original flight plan according to the minimum pilot disturbance policy. The other two aircraft take discrete deconflict actions described in Section II. Aircraft 1 and aircraft 2 detect the potential conflict using the method mentioned in the previous part. Once the potential conflict has been resolved, the return action should be taken to return to its intention. In this process, the aircraft 1 takes action first while aircraft 2 keeps its current velocity. In the second time step, the aircraft 2 takes action and the aircraft 1 keeps its current velocity. This process is going on until the potential conflicts are resolved. Then the return actions are taken. Fig. 13 shows the paths generated by the deconflict and return actions. Fig. 14 shows the deconflict paths of the 4-aircraft encounter scenario, where the aircraft 3 is selected as the non-disturbance aircraft. Fig. 15 presents the result of 5-aircraft scenario in which the aircraft 3 and aircraft 5 are requested to keep their original plans according to the rule of minimum-pilot-disturbance.

如第三节所述，在大多数情况下，部分飞机根据最小飞行员干扰策略保持其原始飞行计划，而其他飞机则负责冲突解决。在 3 架飞机的示例中，至少有两架飞机必须采取行动以解决冲突。根据最小飞行员干扰策略，选择飞机 3 (”ac3”) 保持其原始飞行计划。另外两架飞机采取第二节中描述的离散冲突解决行动。飞机 1 和飞机 2 使用前一部分提到的方法检测潜在冲突。一旦潜在冲突得到解决，应采取返回行动以返回其原定意图。在此过程中，飞机 1 首先采取行动，而飞机 2 保持当前速度。在第二个时间步，飞机 2 采取行动，飞机 1 保持当前速度。此过程持续进行，直到潜在冲突得到解决，然后采取返回行动。图 13 显示了由冲突解决和返回行动产生的路径。图 14 展示了 4 架飞机相遇场景中的冲突解决路径，其中飞机 3 被选为非干扰飞机。图 15 展示了 5 架飞机场景的结果，其中根据最小飞行员干扰规则，要求飞机 3 和飞机 5 保持其原始计划。

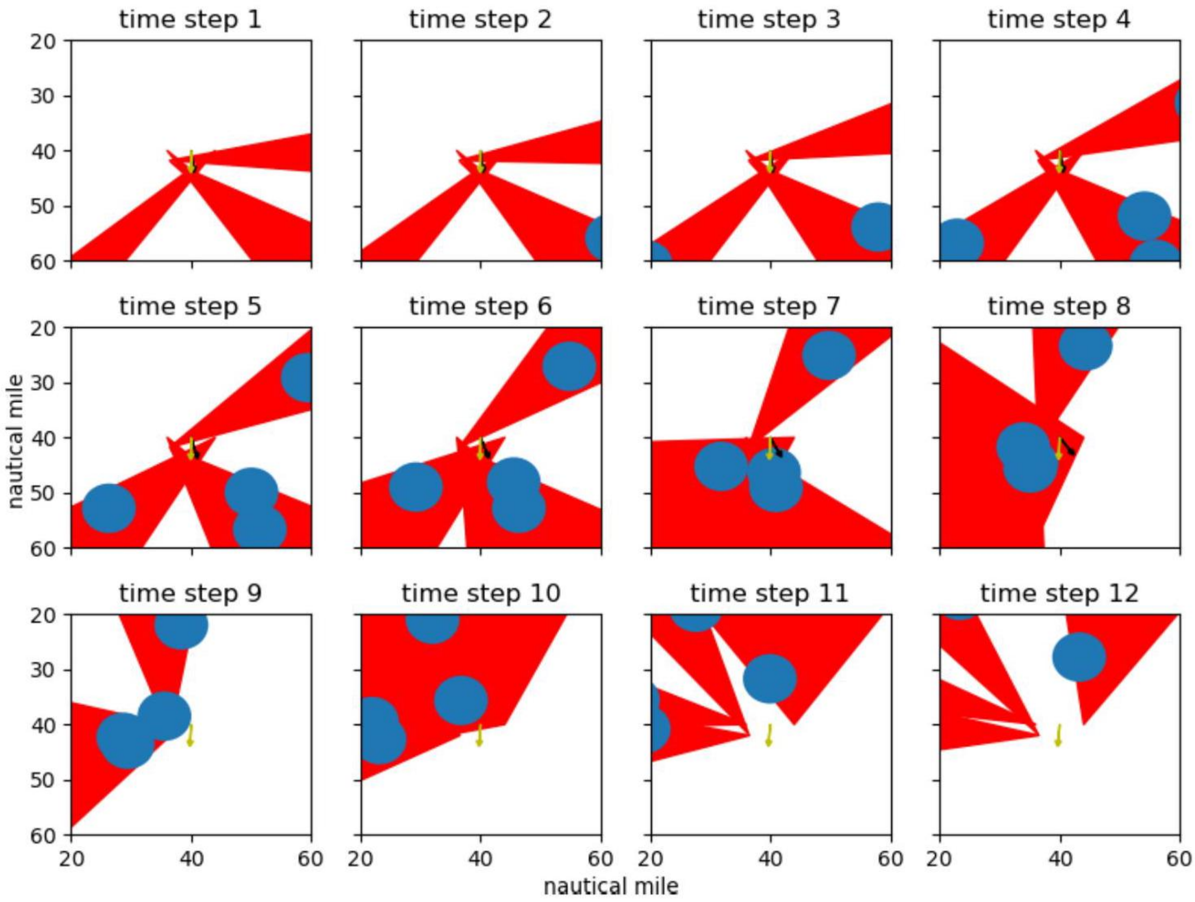


Fig. 12. An example of the conflict resolution process.
图 12. 冲突解决过程的一个示例。

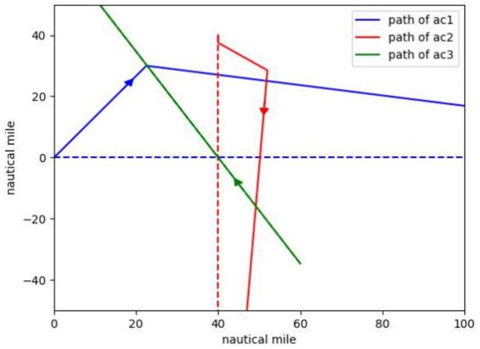


Fig. 13. Result of 3 aircraft deconflicting paths using the proposed method.
图 13. 使用提出方法得到的 3 架飞机冲突解决路径的结果。

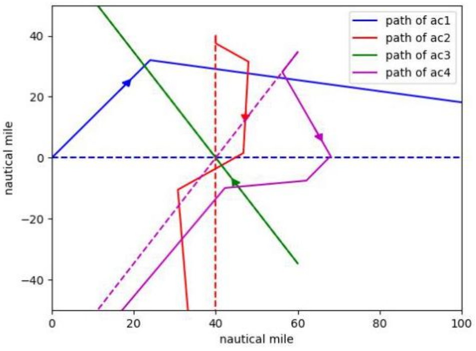


Fig. 14. Result of 4 aircraft deconflicting paths using the proposed method.

图 14. 使用提出方法得到的 4 架飞机冲突解决路径的结果。

As a comparison, the value-based method developed in [26] was tested in the same scenario with the 3-aircraft example. The result of deconflict paths are presented in Fig. 16. Although the max-sum utility function combines multiple affected aircraft to perform conflict resolution cooperatively, the deconflict policy is trained in pair-wise scenario. This leads the agent to make decisions mainly depending on the proximity distance between aircraft rather than the entire airspace awareness. This explains the aircraft 1 (blue curve) turns right first and then turns left to avoid a new conflict.

作为比较，文献 [26] 中提出的基于价值的方法在与 3 架飞机示例相同的场景中进行了测试。冲突解决路径的结果展示在图 16 中。尽管最大和效用函数将多个受影响的飞机组合起来协同执行冲突解决，但冲突解决策略是在成对场景中训练的。这导致代理在做决定时主要依赖于飞机之间的接近距离，而不是整个空域的意识。这解释了为什么飞机 1(蓝色曲线) 首先向右转，然后向左转以避免新的冲突。

Fig. 15. Result of 5 aircraft deconflicting paths using the proposed method.

图 15。使用所提出方法对 5 架飞机进行冲突解决路径的结果。

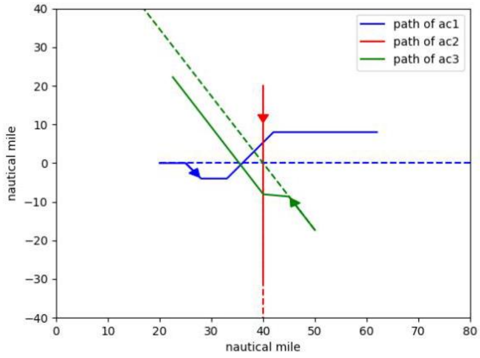


Fig. 16. Result of deconflict path of the value-based method.

图 16。基于价值方法的冲突解决路径结果。

Comparing with the value-based method, the proposed method has several other benefits. First, the intention information is accounted for. The proposed method guides the aircraft to return to the intention if no conflict is detected in the future path. Second, the proposed method is also applicable to continuous action space. The value-based method, which uses search heuristic method to find deconflict policy, cannot search the optimal policy in continuous action space. Third, the proposed method selects a subset of aircraft to perform deconflict to reduce disturbance of pilots, which is not applicable in the value-based method.

与基于价值的方法相比，所提出的方法还具有其他几个优点。首先，考虑了意图信息。所提出的方法指导飞机在未检测到未来路径中的冲突时返回到原意图。其次，所提出的方法也适用于连续动作空间。使用搜索启发式方法寻找冲突解决策略的基于价值方法，无法在连续动作空间中搜索最优策略。第三，所提出的方法选择了一组飞机执行冲突解决，以减少对飞行员的干扰，这在基于价值的方法中是不适用的。

The constant speed optimal reciprocal collision avoidance (CSORCA) method developed in [18] is also implemented using the same scenario. The deconflict path of this method is presented in Fig. 17. It can be seen that the deconflict paths do not deviate much from the original flight plans, which is beneficial for fuel saving. However, as mentioned previously, this method cannot guarantee to find a resolution, causing a relative high probability of conflict in implementation [18]. This will be discussed in the next section. Another drawback is that the strong assumption is enforced that all the affected aircraft must take the same conflict resolution strategy. If one aircraft does not follow this assumption in practice, the ORCA might lead to a dangerous situation.

在文献 [18] 中开发的恒速最优互撞避免方法 (CSORCA) 也使用相同的场景进行了实施。该方法的冲突解脱路径在图 17 中展示。可以看出，冲突解脱路径与原始飞行计划偏差不大，这对节省燃料是有益的。然而，如前所述，此方法不能保证找到解决方案，导致在实际应用中相对较高的冲突概率 [18]。这一点将在下一节中讨论。另一个缺点是，该方法强加了这样一个假设: 所有受影响的飞机必须采取相同的冲突解决策略。如果在实践中有一架飞机不遵循这一假设，ORCA 可能会导致危险情况。

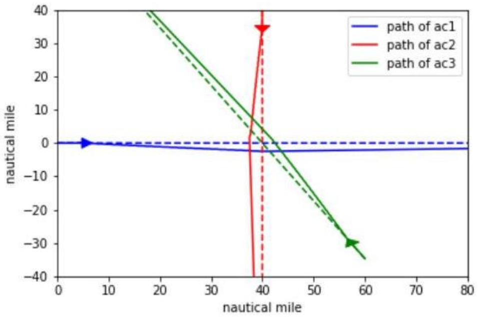


Fig. 17. Result of deconflict path of the CSORCA method.
图 17。CSORCA 方法的冲突解脱路径结果。

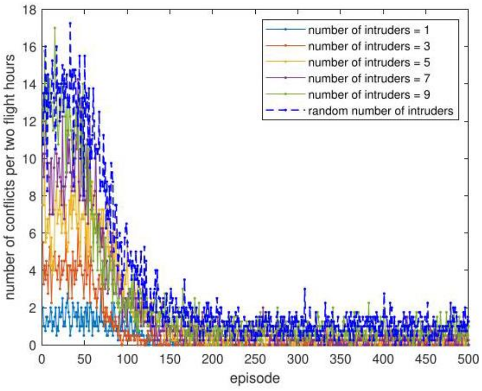


Fig. 18. Sample efficiency with different number of intruders.
图 18。不同入侵者数量的样本效率。

V. DISCUSSION

V. 讨论

A. Scalability

A. 可扩展性

The scalability of the proposed method is studied using the example of discrete actions. Fig. 18 shows the sample efficiency of the proposed method by comparing with the convergences of learning using different number of intruders. The policy with fixed number of intruders is trained by keeping a specific number of intruders in the view (80 × 80 nautical miles). The policy with random number of intruders is trained by randomly generating a number of aircraft (i.e., 1-5 aircraft) in random time steps (i.e., 1-5 time steps). This setting increases the randomness of aircraft number in the view. It is observed from Fig. 18 that the algorithm all converges around 150 episode despite different initial conflict rates. The initial conflict rates typically increases as the aircraft density increases. This demonstrates that samples for the training are efficient irrespective of the number of intruders, which is due to that the learning is based on the SSD image rather than raw state vector. The dimension of the raw state vector depends on the number of intruders, but the image keeps the same size irrespective of the number of intruders.

研究了提出方法的可扩展性，以离散动作的例子为例。图 18 通过比较使用不同入侵者数量的学习收敛性，展示了提出方法的样本效率。固定入侵者数量的策略是通过在视野中保持特定数量的入侵者 ((80 × 80 海里) 进行训练。随机入侵者数量的策略是通过在随机时间步骤中随机生成飞机数量 (即 1-5 架飞机) 进行训练 (即 1-5 个时间步骤)。这种设置增加了视野中飞机数量的随机性。从图 18 观察到，尽管初始冲突率不同，算法都在大约 150 个回合时收敛。初始冲突率通常随着飞机密度的增加而增加。这表明，用于训练的样本在入侵者数量上效率是高的，这是由于学习是基于 SSD 图像而不是原始状态向量。原始状态向量的维度取决于入侵者的数量，但图像的大小不受入侵者数量的影响。

The trained policy with random number of intruders is tested in several scenarios where the average number of intruders ranges from 1 to 20 . The average number of conflicts per two hours flight presented in Fig. 19, which indicates the scalibility of the learned policy. The policy is trained using cases having 1-5 aircrafts and it is shown that the conflict rate with the proposed physics-informed learning is almost zero (see green line in Fig. 19). The trained policy is used for testing cases where the number of intruders increases to 20 . We observe that the conflict rates are close to zero until 15 and increases slightly for 20 intruders. If the proposed physics-informed learning is not used, it is observed that the conflict rate increases significantly. The increase is almost linear, which is the theoretical increase rate with no resolution action. The value-based method and constant speed ORCA (CS-ORCA) method are compared with the proposed method and the results are shown in Fig. 19. As shown in the figure, the proposed method outperforms the value-based method and CS-ORCA method as the number of intruders increases. It should be noted that the increase of conflict rate for large number of intruders is partially due to the significantly increased airspace density and the resolution may not always be successful. Other deconflict actions, such as vertical conflict resolution or collision avoidance should be used to ensure safety. The current field of view is 80-by-80 nautical miles and 20 aircraft in this view is a very high density in the current NAS and is not likely to happen in real life.

随机入侵者数量的训练策略在多个场景中进行了测试，这些场景中入侵者的平均数量从 1 到 20 不等。图 19 所示的每两小时航班的平均冲突数，显示了学习策略的可扩展性。该策略使用 1-5 架飞机的案例进行训练，并且结果表明，采用所提出的物理信息学习方法的冲突率几乎为零 (见图 19 中的绿色线条)。训练好的策略用于测试入侵者数量增加到 20 的案例。我们观察到，在入侵者数量达到 15 之前，冲突率接近零，而对于 20 个入侵者，冲突率略有上升。如果不使用所提出的物理信息学习方法，可以观察到冲突率显著增加。增加几乎呈线性，这是没有解决行动的理论增加率。基于价值的方法和恒定速度的 ORCA(CS-ORCA) 方法与所提出的方法进行了比较，结果见图 19。如图所示，随着入侵者数量的增加，所提出的方法优于基于价值的方法和 CS-ORCA 方法。需要注意的是，大量入侵者时冲突率的增加部分是由于空域密度显著增加，且解决措施可能不一定总是成功。其他冲突解决行动，如垂直冲突解决或避撞，应被用来确保安全。当前视野为 80 英里 ×80 海里，在此视野内 20 架飞机的密度在当前的空管系统中非常高，在现实生活中不太可能发生。

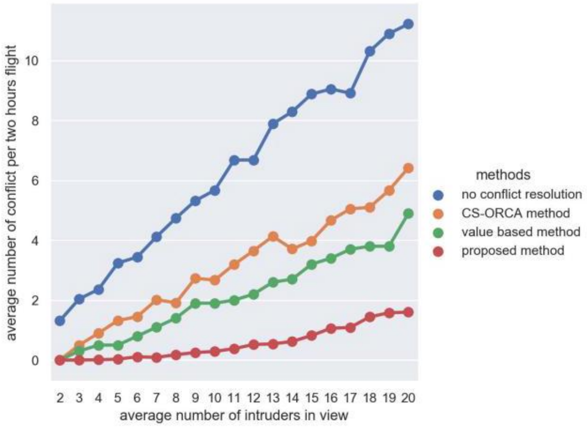


Fig. 19. Performance of resolving different number of intruders using the same policy trained by random number of intruders, in comparison with no conflict resolution, ORCA method and value based method.

图 19. 使用同一策略解决不同数量入侵者的性能，该策略通过随机入侵者数量进行训练，与无冲突解决、ORCA 方法和基于价值的方法进行比较。

B. Comparison With Inputting Adjacent Frames

B. 与输入相邻帧的比较

The proposed method uses the preprocessed frame where the information (speed, direction and position information broadcasted by ADS-B systems) of intruders are encoded by SSD-based method. These information can also be represented by multiple adjacent frames. Feeding multiple adjacent frames into the neural network so that it can detect motion is a typical process in reinforcement learning trained by images. For example, this procedure has been commonly used for reinforcement learning to play Atari Games [34], [43]. In this section, we compare the performances resulted from the two different inputs, i.e., single frame with the information encoded by SSD-based method and raw multiple adjacent frames. As illustrated in Fig. 20a, the red circles directly provide the positions and protection zones of the intruders, and their velocities are obtained by differentiating two adjacent frames. The Fig. 20b presents the corresponding SSD-based image.

提出的方法使用了预处理框架，其中入侵者的信息 (由 ADS-B 系统广播的速度、方向和位置信息) 通过基于 SSD 的方法进行编码。这些信息也可以由多个相邻帧表示。将多个相邻帧输入到神经网络中，使其能够检测运动是图像强化学习训练中的典型过程。例如，这一过程已被广泛应用于通过图像进行强化学习以玩 Atari 游戏 [34]、[43]。在本节中，我们比较了两种不同输入产生的性能，即由 SSD 方法编码信息的单帧和原始的多个相邻帧。如图 20a 所示，红色圆圈直接提供了入侵者的位置和保护区域，它们的速度通过对两个相邻帧求导获得。图 20b 展示了相应的基于 SSD 的图像。

We compare the performances of the two different inputs (i.e., multiple adjacent frames and the SSD-based single frame) using the discrete action example described previously. In the training process, an aircraft enters the surveillance area (80×80 nautical miles area around the own aircraft) at a random time step from 1 to 5, and pops out if it flies beyond the scope of the area, this ensures the policy is trained by random number of aircraft. Two adjacent frames (the previous frame and the current frame) are feed into two channels of the convolutional neural network. The CNN extracts the dynamic information from the two adjacent frames for training. The proposed method input the SSD-based single frame. The other settings and hyperparameters are all identical for the two methods for a fair comparison. Thus, the only difference between the two methods is on the inclusion of prior physics knowledge in the SSD-based image. The convergence behaviors of the two methods are presented in Fig. 21. Several experiments are performed to plot the mean and confident interval. A much better performance of convergence is observed using the proposed method, including faster convergence speed, lower mean value and smaller deviation.The difference of the number of conflicts at the end of training may appear to be very small in Fig. 21 (e.g., 0.2 vs. 0.01). However, this difference represents a more than magnitude change of probability of conflict which is significant from a safety point of view. It appears that the embeded prior physics knowledge using SSD enhances the performance in the current investigation.

我们通过之前描述的离散动作示例，比较了两种不同输入 (即多个相邻帧和基于 SSD 的单帧) 的性能。在训练过程中，一架飞机在 1 到 5 的随机时间步进入监控区域 (80×80 自身飞机周围的 nautical miles 区域)，并且如果飞出该区域范围，则会消失，这确保了策略是通过随机数量的飞机进行训练的。两个相邻帧 (前一个帧和当前帧) 被输入到卷积神经网络的两个通道中。CNN 从这两个相邻帧中提取动态信息用于训练。所提出的方法输入基于 SSD 的单帧。两种方法的其它设置和超参数均相同，以进行公平比较。因此，两种方法之间的唯一区别在于基于 SSD 的图像中是否包含先验物理知识。两种方法的收敛行为在图 21 中展示。进行了多项实验以绘制平均值和置信区间。使用所提出的方法观察到更好的收敛性能，包括更快的收敛速度、更低的平均值和更小的偏差。在图 21 中，训练结束时冲突数量的差异可能看起来非常小 (例如，0.2 对 0.01)，然而，这种差异代表了冲突概率超过一个量级的变化，从安全角度来看这是显著的。在当前研究中，使用 SSD 嵌入的先验物理知识似乎提高了性能。

Lower performance of using adjacent frames than SSD encoded image as input might due to the synchronization problem between the state and action. As has been mentioned, the velocities of intruders extracted by differentiating the previous frame and the current frame are the average velocities of the last interval. Therefore, the environment state used for making decision includes the intruders' average velocities of the last interval. Changes in the current velocities are not respected in the current decision, which might mislead the training process and increase conflict risk. Contrarily, the SSD-based single frame encodes the real-time velocity and position information (broadcasted by ADS-B system) of each intruder, ensuring the own aircraft responds to the current velocities of intruders. This explains the lag of convergence of the method using adjacent frames as input. In practice, this situation might be improved by reducing the time interval of sampling images, but noises will dominate the observations (position and velocity) if the interval is too small and thus compromise the safety. In addition, high frequency of taking actions will increase workload of pilots and controllers.

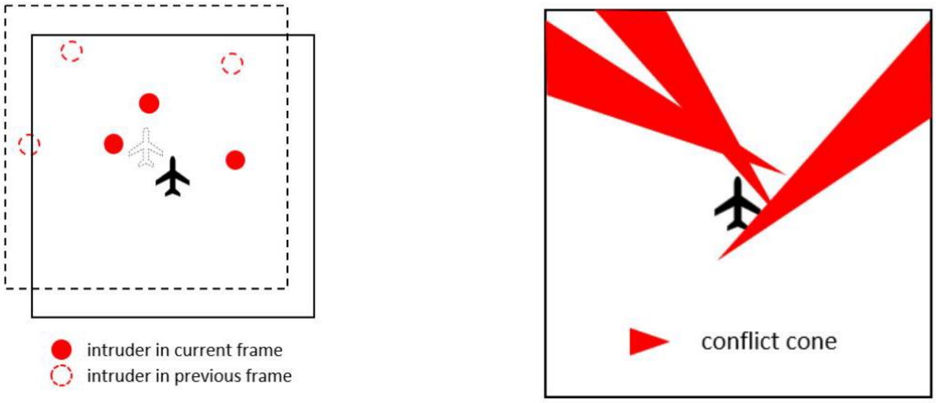
使用相邻帧作为输入的性能低于使用 SSD 编码图像可能是因为状态与动作之间的同步问题。如前所述，通过微分前一个帧和当前帧得到的入侵者的速度是上一个时间间隔的平均速度。因此，用于决策的环境状态包括入侵者上一个时间间隔的平均速度。当前决策中没有考虑到当前速度的变化，这可能会误导训练过程并增加冲突风险。相反，基于 SSD 的单帧编码了每个入侵者的实时速度和位置信息 (由 ADS-B 系统广播)，确保了自身飞机对入侵者当前速度的响应。这解释了使用相邻帧作为输入的方法的收敛滞后。在实际中，通过减小图像采样的时间间隔可能会改善这种情况，但如果间隔太小，噪声将主导观测值 (位置和速度)，从而损害安全性。此外，频繁采取动作会增加飞行员和控制器的工作量。

C. Look-Ahead Time and Action Frequency

C. 预测时间和动作频率

The proposed method focuses on short-term conflict resolution. A 80 × 80 pixels image is used in all the above experiments to represent 80 × 80 nautical miles airspace centered by the own aircraft, which is projected time of 5 minutes for the own aircraft to fly out the area if the speed is assumed to be 450 knots. The conflict detection and resolution are triggered once the intruders enter this airspace, the look-ahead time in this situation is 5 minutes. The Fig. 22 presents convergence performance using different look-ahead times of 2.5 minutes (corresponds to 40 × 40 nautical miles surveillance area), 5 minutes and 10 minutes (corresponds to 160 × 160 nautical miles surveillance area), indicating that the algorithm can learn deconflict policies for all the 3 look-ahead times. The 2.5 minutes look-ahead time converges with a very little delay than the others in the training, this might be explained by less choices of deconflict strategies existing in the case of short-term look-ahead time and thus is more difficult to learn. The 5 minutes and 10 minutes look-ahead time cases behave the same in training. This might because the strategy spaces are sufficient for the agents to learn. In practice, potential conflict with too short look-ahead time will be taken charge of by airborne TCAS system, which is beyond the scope of this research. A long look-ahead time deconflict strategy can increase the false alarm rate, which should be avoided to reduce the pilot's workload.

提出的方法专注于短期冲突解决。在上述所有实验中使用 80 × 80 像素的图像来表示以己方飞机为中心的 80 × 80 海里空域，假设速度为 450 节，则己方飞机飞出该区域的时间为 5 分钟。一旦入侵者进入这个空域，就会触发冲突检测和解决，在这种情况下，前瞻时间为 5 分钟。图 22 展示了在使用不同前瞻时间 (分别为 2.5 分钟 (对应 40 × 40 海里监控区域)、5 分钟和 10 分钟 (对应 160 × 160 海里监控区域)) 时的收敛性能，表明算法可以在所有 3 个前瞻时间下学习冲突解决策略。2.5 分钟的前瞻时间在训练中的收敛延迟比其他情况要小，这可能是因为在短期前瞻时间的情况下，冲突解决策略的选择较少，因此更难学习。5 分钟和 10 分钟前瞻时间的情况在训练中表现相同。这可能是因为策略空间足够代理学习。在实际中，过短前瞻时间的潜在冲突将由机载 TCAS 系统处理，这超出了本研究的范围。长时间前瞻的冲突解决策略会增加误报率，应当避免以减少飞行员的负担。



(a) 2-frames input

(b) SSD-based single frame input

Fig. 20. Illustration of information encoding manipulation of reinforcement learning method using multiple frames versus the SSD based single frame. The former uses at least two adjacent frames to encode dynamic information while the SSD based method only uses one frame to include the dynamic.

图 20。展示了使用多个帧与基于 SSD 的单帧进行强化学习方法的信息编码操作对比。前者至少使用两个相邻帧来编码动态信息，而基于 SSD 的方法只使用一个帧来包含动态信息。

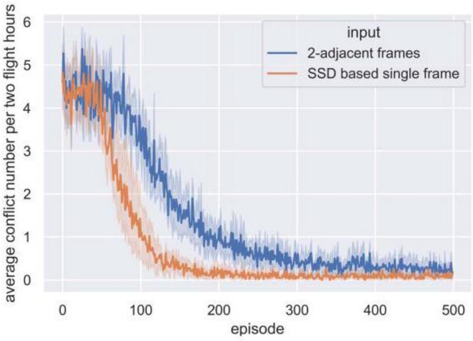


Fig. 21. Convergence speed comparison between different inputs.

图 21。不同输入之间的收敛速度比较。

The time steps of all the above experiments are assumed to be 40 seconds, i.e., the time interval between two adjacent actions is 40 seconds. Convergence behaviors of different time steps (20 seconds, 40 seconds, and 60 seconds) are studied in Fig. 23, which shows no significant difference. It should be noted that a short time-step will increase the pilot or controller's workload, and a long time-step will increase the risk due to uncertainties in each time-step. Therefore, a proper time step should be set in practice.

以上所有实验的时间步长假定为 40 秒，即两个相邻动作之间的时间间隔为 40 秒。图 23 研究了不同时间步长 (20 秒、40 秒和 60 秒) 的收敛行为，结果显示没有显著差异。需要注意的是，较短的时间步长会增加飞行员或控制员的负担，而较长的时间步长则会因为每个时间步的不确定性增加风险。因此，在实践中应设置适当的时间步长。

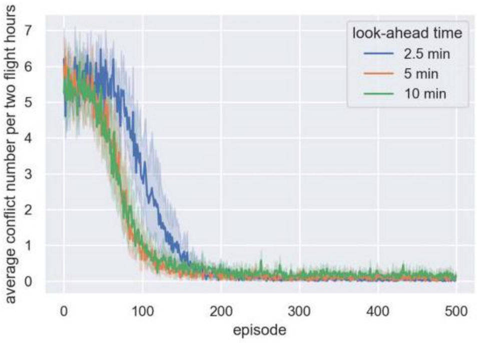


Fig. 22. Convergence speed comparison between different look-ahead times.

图 22. 不同前瞻时间下的收敛速度比较。

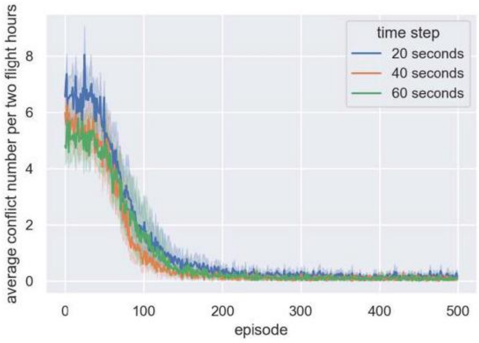


Fig. 23. Convergence speed comparison between different time intervals.
图 23. 不同时间间隔下的收敛速度比较。

D. Action Smoothness

D. 动作平滑性

The reinforcement learning based controller may exhibit non-smoothed behaviors. Oscillatory actions have been observed in many tasks [44]-[48], especially in continuous control implementation. Action smoothness is a challenge but critical work for transferring RL-based controller from simulation to real-world application. Attempts to solve this problem mainly focuse on reward-engineering to induce the desired behavior [45], [46]. In our work, we also use the reward-engineering method (see Eq.7) to induce smooth actions. It is recognized that reward-engineering for action smoothness needs careful parameters tuning and provides no guarantees for the desired behavior [44]. Therefore, action smoothness would be an import future work for the implementation of this work.

基于强化学习的控制器可能会表现出非平滑行为。在许多任务中已经观察到振荡性动作 [44]-[48]，特别是在连续控制实现中。动作平滑性是将在基于 RL 的控制器从模拟转移到实际应用中的一项挑战性但关键的工作。解决这个问题的尝试主要集中在通过奖励工程来诱导期望行为 [45]、[46]。在我们的工作中，我们还使用了奖励工程方法 (见公式 7) 来诱导平滑动作。人们认识到，为了动作平滑性进行的奖励工程需要仔细调整参数，并不能保证期望行为 [44]。因此，动作平滑性将是实现这项工作的重要未来工作。

VI. CONCLUSION AND FUTURE WORK

结论与未来工作

A novel physics-informed deep reinforcement learning for conflict resolution in air traffic management is proposed in this study. Based on the solution space diagram (SSD) method, intruders' information at each time step are integrated into an image. This image is used by an convolutional network to extract the information that is the input into the deep reinforcement learning network for learning the resolution policy. Several numerical examples are studied, including both heading angle changing and speed control. Extensive discussion for the scalability, convergence rate, single/multi agent learning, and flight intent effect are presented. Several conclusions can be drawn based on the proposed study:

在本研究中，提出了一种新颖的基于物理信息的深度强化学习算法，用于空中交通管理中的冲突解决。基于解决方案空间图 (SSD) 方法，将每个时间步的入侵者信息整合成一幅图像。该图像被卷积网络用来提取信息，作为深度强化学习网络的输入，以学习解决策略。研究了许多数值示例，包括改变航向角和速度控制。对可扩展性、收敛速度、单/多智能体学习和飞行意图影响进行了广泛讨论。基于提出的研究，可以得出以下结论:

- This work provides a mechanism for the aircraft to learn conflict resolution policy from the simulation environment. The physics-informed deep learning largely improve the scalability issue as the algorithm can handle arbitrary number of aircraft as images rather than aircraft states are used.

- 本工作为飞机从仿真环境中学习冲突解决策略提供了一种机制。物理信息深度学习在很大程度上解决了可扩展性问题，因为算法可以处理任意数量的飞机，使用图像而不是飞机状态。

- The discrete action space performs better than the continuous action space on convergence speed and robustness.

- 离散动作空间在收敛速度和鲁棒性方面表现优于连续动作空间。

- The current study shows that the heading angle changing method performs better than the speed control method on convergence speed and robustness.

- 当前研究表明，改变航向角的方法在收敛速度和鲁棒性方面优于速度控制方法。

- Conflict resolution that simultaneously using heading angle and speed control actions has better performance than any single control action.

- 同时使用航向角和速度控制动作的冲突解决方法比任何单一控制动作的表现都要好。

- The embedded physics knowledge shows significantly improvement of scalability and only needs a small number of intruders to train the policy. Traditional reinforcement learning method fails to resolve the conflict when using the policy to a larger number of intruders.

- 内嵌的物理知识显著提高了可扩展性，并且只需要少量入侵者来训练策略。当将传统强化学习方法的策略应用于更大数量的入侵者时，无法解决冲突。

The future work is listed as the following:
未来的工作如下所列:

- Action smoothness would be an import future work for the implementation of this work.

- 动作平滑性将是本工作实施的一个重要未来研究方向。

- Detailed investigation of scalable multi-agent reinforcement learning (MARL) for large number of aircraft conflict resolution would be a future work.

- 对大量飞机冲突解决的可扩展多智能体强化学习 (MARL) 进行详细研究将是未来的工作。

- Further study for probabilistic failure probability constraints needs to be conducted.

- 需要进行概率性故障概率约束的进一步研究。

3-dimensional
三维的

- deconflict strategy should be studied in the future.

- 今后应研究冲突消解策略。

ACKNOWLEDGMENT

致谢

The works related to this research were performed at the Prognostic Analysis and Reliability Assessment Laboratory, Arizona State University. The support is gratefully acknowledged.

与本研究相关的工作是在亚利桑那州立大学的预后分析与可靠性评估实验室进行的。我们对所提供的支持表示衷心的感谢。

REFERENCES

参考文献

[1] H. Erzberger, ”CTAS: Computer intelligence for air traffic control in the terminal area,” NASA Ames Res. Center, Moffett Field, CA, USA, Tech. Rep. 92N33080, 1992.

[2] J. K. Kuchar and L. C. Yang, ”A review of conflict detection and resolution modeling methods,” IEEE Trans. Intell. Transp. Syst., vol. 1, no. 4, pp. 179-189, Dec. 2000.

[3] H. Erzberger, ”Automated conflict resolution for air traffic control,” NASA Ames Res. Center, Moffett Field, CA, USA, Tech. Rep. 20050242942, 2005.

[4] R. Bach, Y.-C. Chu, and H. Erzberger, ”A path-stretch algorithm for conflict resolution,” NASA-Ames Research Center, Moffett Field, CA, USA, Tech. Rep. NASA/CR-2009-214574, 2009.

[5] R. Bach, C. Farrell, and H. Erzberger, ”An algorithm for level-aircraft conflict resolution,” NASA, Washington, DC, USA, Tech. Rep. CR-

[6] H. Erzberger and K. Heere, ”Algorithm and operational concept for resolving short-range conflicts,” Proc. Inst. Mech. Eng., G, J. Aerosp. Eng., vol. 224, no. 2, pp. 225-243, Feb. 2010.

[7] R. A. Paielli, H. Erzberger, D. Chiu, and K. R. Heere, ”Tactical conflict alerting aid for air traffic controllers,” J. Guid., Control, Dyn., vol. 32,

[8] H. Erzberger, T. A. Lauderdale, and Y.-C. Chu, ”Automated conflict Proc. 27th Int. Congr. Aeronaut. Sci., 2010, pp. 1-20.

[9] M. Refai, M. Abramson, S. Lee, and G. Wu, ”Encounter-based simulation architecture for detect-and-avoid modeling,” in Proc. AIAA Scitech Forum, Jan. 2019, p. 1476.

[10] M. S. Eby, ”A self-organizational approach for resolving air traffic conflicts,” Lincoln Lab. J., vol. 7, no. 2, p. 239-254, Sep. 1995.

[11] K. Zeghal, ”A review of different approaches based on force fields for airborne conflict resolution,” in Proc. Guid., Navigat., Control Conf. Exhibit, Boston, MA, USA, Aug. 1998, p. 4240.

[12] S. Balasooriyan, ”Multi-aircraft conflict resolution using velocity obstacles,” M.S. thesis, Delft Univ. Technol., Delft, The Netherlands, 2017. [Online]. Available: https://repository.tudelft.nl/islandora/object/uuid%3Acf361aff-a7ec-444c-940a-d711e076d108

[13] J. Ellerbroek, M. Visser, S. B. J. van Dam, M. Mulder, and M. M. van Paassen, ”Design of an airborne three-dimensional separation assistance display,” IEEE Trans. Syst., Man, Cybern., A, Syst. Humans, vol. 41, no. 5, pp. 863-875, Sep. 2011.

[14] S. M. Abdul Rahman, M. Mulder, and R. van Paassen, ”Using the solution space diagram in measuring the effect of sector complexity during merging scenarios,” in Proc. AIAA Guid., Navigat., Control Conf., Aug. 2011, p. 6693, doi: 10.2514/6.2011-6693.

[15] J. G. d’Engelbronner, C. Borst, J. Ellerbroek, M. M. van Paassen, and M. Mulder, ”Solution-space-based analysis of dynamic air traffic controller workload,” J. Aircr., vol. 52, no. 4, pp. 1146-1160, 2015, doi: 10.2514/1.C032847.

[16] S. J. van Rooijen, J. Ellerbroek, C. Borst, and E. van Kampen, ”Toward individual-sensitive automation for air traffic control using convolutional neural networks,” J. Air Transp., vol. 28, no. 3, pp. 105-113, Jul. 2020, doi: 10.2514/1.D0180.

[17] J. van den Berg, S. J. Guy, M. Lin, and D. Manocha, ”Reciprocal n-body collision avoidance,” in Robotics Research, C. Pradalier, R. Siegwart, and G. Hirzinger, Eds. Berlin, Germany: Springer, 2011, pp. 3-19.

[18] N. Durand, ”Constant speed optimal reciprocal collision avoidance,” Transp. Res. C, Emerg. Technol., vol. 96, pp. 366-379, Nov. 2018. [Online]. Available: http://www.sciencedirect.com/ science/article/pii/S0968090X18314232

[19] S. Ghosh, S. Laguna, S. H. Lim, L. Wynter, and H. Poonawala, ”A deep ensemble multi-agent reinforcement learning approach for air traffic control,” 2020, arXiv:2004.01387. [Online]. Available: https://arxiv.org/abs/2004.01387

[20] A. M. F. Crespo, L. Weigang, and A. G. De Barros, ”Reinforcement learning agents to tactical air traffic flow management,” Int. J. Aviat. Manage., vol. 1, no. 3, pp. 145-161, 2012.

[21] T. Kravaris et al., ”Resolving congestions in the air traffic management domain via multiagent reinforcement learning methods,” 2019, arXiv:1912.06860. [Online]. Available: https://arxiv.org/abs/1912.06860

[22] K. Malialis, S. Devlin, and D. Kudenko, ”Resource abstraction for reinforcement learning in multiagent congestion problems,” 2019, arXiv:1903.05431. [Online]. Available: https://arxiv.org/abs/1903.05431

[23] S. Temizer, M. Kochenderfer, L. Kaelbling, T. Lozano-Perez, and J. Kuchar, ”Collision avoidance for unmanned aircraft using Markov decision processes,” in Proc. AIAA Guid., Navigat., Control Conf., Toronto, ON, Canada, Aug. 2010, p. 8040, doi: 10.2514/6.2010-8040.

[24] M. J. K. J. P. Chryssanthacopoulos, ”Robust airborne collision avoidance through dynamic programming,” Lincoln Lab., Massachusetts Inst. Technol., Cambridge, MA, USA, Tech. Rep. ATC-371, Jan. 2011.

[25] K. D. Julian, J. Lopez, J. S. Brush, M. P. Owen, and M. J. Kochenderfer, ”Policy compression for aircraft collision avoidance systems,” in Proc. IEEE/AIAA 35th Digit. Avionics Syst. Conf. (DASC), Sep. 2016, pp. 1-10.

[26] H. Y. Ong and M. J. Kochenderfer, ”Markov decision process-based distributed conflict resolution for drone air traffic management,” J. Guid., Control, Dyn., vol. 40, no. 1, pp. 69-80, Jan. 2017, doi:

[27] S. Li, M. Egorov, and M. Kochenderfer, ”Optimizing collision avoidance in dense airspace using deep reinforcement learning,” 2019, arXiv:1912.10146. [Online]. Available: https://arxiv.org/abs/1912.10146

[28] M. Ribeiro, J. Ellerbroek, and J. Hoekstra, ”Improvement of conflict detection and resolution at high densities through reinforcement learn-

[29] H. Wen, H. Li, Z. Wang, X. Hou, and K. He, ”Application of DDPG-based collision avoidance algorithm in air traffic control,” in Proc. 12th Int. Symp. Comput. Intell. Design (ISCID), vol. 1, Dec. 2019,

[30] Z. Wang, H. Li, J. Wang, and F. Shen, ”Deep reinforcement learning Transp. Syst., vol. 13, no. 6, pp. 1041-1047, Jun. 2019. [Online]. its.2018.5357

[31] M. Brittain and P. Wei, ”Autonomous air traffic controller: A deep multi-agent reinforcement learning approach,” 2019, arXiv:1905.01303. [Online]. Available: https://arxiv.org/abs/1905.01303

[32] P. N. Tran, D.-T. Pham, S. K. Goh, S. Alam, and V. Duong, ”An interactive conflict solver for learning air traffic conflict resolutions,” J. Aerosp. Inf. Syst., vol. 17, no. 6, pp. 271-277, 2020, doi: 10.2514/1.1010807.

[33] M. Britaain and P. Wei, ”Autonomous aircraft sequencing and separation with hierarchical deep reinforcement learning,” in Proc. Int. Conf. Res. Air Transp., Catalonia, Spain, Jun. 26-29, 2018.

[34] V. Mnih et al., ”Human-level control through deep reinforcement learning,” Nature, vol. 518, no. 7540, pp. 529-533, 2015.

[35] V. Mnih et al., ”Asynchronous methods for deep reinforcement learning,” in Proc. Int. Conf. Mach. Learn., 2016, pp. 1928-1937.

[36] J. Schulman, S. Levine, P. Moritz, M. I. Jordan, and P. Abbeel, ”Trust region policy optimization,” in Proc. Int. Conf. Mach. Learn., 2015, pp. 1889-1897.

[37] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, ”Proximal policy optimization algorithms,” 2017, arXiv:1707.06347. [Online]. Available: https://arxiv.org/abs/1707.06347

[38] P. Hermes, M. Mulder, M. M. van Paassen, J. H. L. Boering, and H. Huisman, ”Solution-space-based complexity analysis of the difficulty of aircraft merging tasks,” J. Aircr., vol. 46, no. 6, pp. 1995-2015, Nov. 2009, doi: 10.2514/1.42886.

[39] J. Ellerbroek, K. C. R. Brantegem, M. M. van Paassen, and M. Mulder, ”Design of a coplanar airborne separation display,” IEEE Trans. Human-Mach. Syst., vol. 43, no. 3, pp. 277-289, May 2013.

[40] J. Schulman, P. Moritz, S. Levine, M. Jordan, and P. Abbeel, ”High-dimensional continuous control using generalized advantage estimation,” 2018, arXiv:1506.02438. [Online]. Available: https://arxiv.org/pdf/1506.02438.pdf

[41] S. Kakade and J. Langford, ”Approximately optimal approximate reinforcement learning,” in Proc. 19th Int. Conf. Mach. Learn., 2002, pp. 267-274.

[42] D. Silver, R. S. Sutton, and M. Müller, ”Temporal-difference search in computer go,” Mach. Learn., vol. 87, no. 2, pp. 183-219, May 2012, doi: 10.1007/s10994-012-5280-0.

[43] V. Mnih et al., ”Playing atari with deep reinforcement learning,” 2013, arXiv:1312.5602. [Online]. Available: https://arxiv.org/abs/1312.5602

[44] S. Mysore, B. Mabsout, R. Mancuso, and K. Saenko, ”Regularizing action policies for smooth control with reinforcement learning,” 2020, arXiv:2012.06644. [Online]. Available: https://arxiv.org/abs/2012.06644

[45] A. R. Mahmood, D. Korenkevych, G. Vasan, W. Ma, and J. Bergstra, ”Benchmarking reinforcement learning algorithms on real-world robots,” in Proc. Conf. Robot Learn., 2018, pp. 561-591.

[46] W. Koch, ”Flight controller synthesis via deep reinforcement learning,” 2019, arXiv:1909.06493. [Online]. Available:

[47] Y. Duan, X. Chen, R. Houthooft, J. Schulman, and P. Abbeel, ”Bench-Int. Conf. Mach. Learn., 2016, pp. 1329-1338.

[48] F. Sadeghi, A. Toshev, E. Jang, and S. Levine, ”Sim2real view invariant visual servoing by recurrent control,” 2017, arXiv:1712.07642. [Online]. Available: https://arxiv.org/abs/1712.07642

Peng Zhao received the Ph.D. degree from the School of Electronic and Information Engineering, Beihang University, China, working on integrity monitoring of satellite navigation systems application in civil aircraft. He is currently a Post-Doctoral Researcher with the School for Engineering of Matter, Transport & Energy, Arizona State University. His research interests include information fusion for real-time national air transportation system prognostics under uncertainty, collision avoidance and resolution methods, and unmanned air traffic management systems.

彭赵博士毕业于中国北京航空航天大学电子与信息工程学院，研究方向为卫星导航系统在民用飞机上的完整性监控。他目前是亚利桑那州立大学物质、运输与能源工程学院的博士后研究员。他的研究兴趣包括不确定性下实时国家航空运输系统的信息融合预后、碰撞避免与解决方法以及无人航空交通管理系统。

Yongming Liu is currently a Professor of aerospace and mechanical engineering with the School for Engineering of Matter, Transport & Energy, Arizona State University. He heads the Prognostic Analysis and Reliability Assessment Laboratory (PARA). His research interests include fatigue and fracture of engineering materials and structures, probabilistic computational mechanics, risk assessment and management to multi-physics damage modeling and structural durability, multi-scale uncertainty quantification and propagation, and imaging-based experimental testing, diagnostics, and prognostics.

刘永明目前是亚利桑那州立大学物质、运输与能源工程学院航空航天与机械工程教授。他领导了预后分析与可靠性评估实验室 (PARA)。他的研究兴趣包括工程材料与结构的疲劳与断裂、概率计算力学、多物理损伤建模与结构耐久性的风险评估与管理、多尺度不确定性量化与传播以及基于成像的实验测试、诊断与预后。