An Agent-Based Approach for Air Traffic Conflict Resolution in a Flow-Centric Airspace

基于代理的流量中心空域空中交通冲突解决方法

Yash Guleria, Duc-Thinh Pham and Sameer Alam

Yash Guleria, Duc-Thinh Pham 和 Sameer Alam

Abstract- The air traffic control paradigm is shifting from sector-based operations to flow-centric approaches to address the scalability limitations of geographically-bound air traffic sectors and to meet the growing demands of air traffic. These future concepts of operations differ from traditional air traffic operations, especially in maintaining safe separation between flights. Flow-centric operations are characterized by maintaining safe separation between traffic flows (both interflow as well as intra-flow), in contrast to current standards of maintaining separation between pairs of aircraft. This paper proposes a novel approach for resolving air traffic conflicts in flow-centric en-route airspace by employing a combination of a model-free Deep Reinforcement Learning policy and a self-stabilizing graph structure. The problem is formulated as a sequential decision-making task in a large action space, requiring a series of decisions to be made over time to resolve potential conflicts at both the inter-flow and intra-flow levels, while adhering to the flow plans and subsequently reaching the destination. Model performance is evaluated by measuring the frequency of safe separations achieved and the efficiency of the maneuvers (deviation from the flow plans). Despite the intra-flow and inter-flow speed uncertainties and the dynamic behavior of the shape of the flows due to variations in the number of aircraft in each flow in every scenario, the proposed approach achieves safe separations for 100% of the scenarios evaluated. The results also demonstrate that despite the induced delay due to conflict resolution maneuvers, the flows closely adhere to their original flow plan. This approach can be used to develop intelligent conflict resolution advisory tools in a flow-centric airspace paradigm.

摘要- 空中交通管制范式正在从基于扇区的操作转向流量中心方法，以解决地理限制的空中交通扇区的可扩展性限制，并满足空中交通日益增长的需求。这些未来的操作概念与传统的空中交通操作有所不同，特别是在保持航班间安全间隔方面。流量中心操作的特点是保持交通流 (包括流间以及流内) 之间的安全间隔，与目前保持飞机对之间间隔的标准相比。本文提出了一种新颖的方法，通过结合无模型深度强化学习策略和自稳定图结构，解决流量中心航路空域中的空中交通冲突。问题被构建为一个序列决策任务，在一个大的动作空间中，需要随时间做出一系列决策来解决潜在的冲突，同时遵守流量计划并最终到达目的地。模型性能通过衡量实现安全间隔的频率和机动效率 (偏离流量计划) 来评估。尽管流内和流间速度存在不确定性，且由于每种情况下每个流量中的飞机数量变化导致流量形状的动态行为，所提出的方法在 100% 的评估场景中实现了安全间隔。结果还表明，尽管由于冲突解决机动产生了延迟，流量仍然紧密遵循其原始流量计划。这种方法可以用于在流量中心空域范式中开发智能冲突解决咨询工具。

I. INTRODUCTION

I. 引言

The existing sector-based air traffic control (ATC) system presents inherent scalability constraints that hinder meeting future demands and ensuring sustainable growth [1]. In day-to-day operations, the airspace is dynamically reconfigured according to the controllers' workload [2]. Under-loaded sectors are collapsed to form larger sectors, and overloaded sectors are split into several smaller sectors, which are then operated by different air traffic controllers (ATCOs). This sectorization, especially splitting the sectors, becomes inefficient beyond a certain threshold, rendering sector sizes impractical for operational purposes.

现有的基于扇区的空中交通管制 (ATC) 系统存在固有的可扩展性限制，这些限制阻碍了满足未来需求并确保可持续发展 [1]。在日常工作过程中，根据管制员的任务负荷，空域会动态重新配置 [2]。负荷不足的扇区会被合并形成更大的扇区，而负荷过重的扇区会被分割成几个较小的扇区，然后由不同的空中交通管制员 (ATCOs) 操作。这种扇区划分，尤其是分割扇区，在超过一定阈值后变得低效，使得扇区大小对于运营目的来说不切实际。

Therefore, a paradigm shift in the current operational concepts is necessary to effectively plan for and accommodate future air traffic. Alternative concepts, such as flight-centric and flow-centric operations (FCOs), are being explored. While flight-centric operations are associated with a free route airspace, FCOs are proposed as a hybrid of flight-centric and structured airspace operations. In this concept, a traffic flow refers to a group of aircraft operating on multiple airways with distinct geographical characteristics, including flight trajectory orientation, proximity to their current geographic area, flights originating from and heading towards the same region, and flights proceeding to destinations within the same area/region. [3]. FCOs involve managing air traffic from an aggregated perspective based on the formation and evolution of major traffic flows, instead of individual flights. This aggregated traffic management implies that FCOs can be considered as a next stage of trajectory-based operations (TBOs), which are focused on flight-centric operations to reduce capacity demand imbalances. FCOs aim to optimize traffic at a regional level, improve the workload distribution across under and over-loaded ATC units, and overcome the scalability constraints of current geographic sectors. This approach shifts the ATCOs' responsibility from managing all traffic within a given sector to overseeing a specified number of aircraft throughout their flight segment within an airspace. Air traffic conflict resolution is a crucial aspect of air traffic control, wherein ATCOs bear the responsibility of ensuring the safety and efficiency of flight operations. In the current operations, each sector is controlled by a separate ATCO, who is in charge of ensuring safe and efficient flight operations in that sector. ATCOs possess knowledge of individual aircraft's movements and intentions, maintain situational awareness regarding potential safety violations, and are trained to respond effectively to such scenarios. Communication between ATCOs from different sectors typically occurs during aircraft handover/takeover or in situations where aircraft deviate from their designated reporting waypoints due to weather or other factors. In FCOs, the responsibilities of ATCOs remain unchanged, but the characteristics significantly differ from existing structured airspace operations. ATCOs handling a flow must ensure safe separations between the flows as well as the aircraft of the same flow. This makes it a two-level conflict resolution problem. Thus, greater situational awareness is required to monitor and ensure safety at both levels. Furthermore, a collaborative approach between the ATCOs managing different flows is required at all times. For conflict resolution, factors such as the type of maneuvers, the maneuver frequency, and information sharing from both, intra-flow and inter-flow aspects also become crucial.

因此，当前运营理念的范式转变对于有效规划和适应未来空中交通是必要的。正在探索替代性概念，如以飞行为中心和以流量为中心的运营 (FCOs)。以飞行为中心的运营与自由航线空域相关，而 FCOs 被提议为以飞行为中心和结构化空域运营的混合体。在这个概念中，流量指的是在具有不同地理特征的多个航路上运行的飞机群，包括飞行轨迹方向、接近其当前地理区域、从同一地区出发并飞往同一地区的航班，以及飞往同一区域/地区目的地的航班。[3]。FCOs 涉及从流量形成和演化的聚合视角管理空中交通，而不是单个航班。这种聚合交通管理意味着 FCOs 可以被视为基于轨迹的运营 (TBOs) 的下一阶段，后者专注于以飞行为中心的运营以减少容量需求不平衡。FCOs 旨在优化区域级别的交通，改善负载不足和超负荷空中交通管制单位的工作量分配，并克服当前地理扇区的可扩展性限制。这种方法将空中交通管制员 (ATCOs) 的责任从管理给定扇区内的所有交通转变为在整个空域内监督指定数量的飞机的飞行段。空中交通冲突解决是空中交通管制的一个关键方面，ATCOs 承担确保飞行运营安全和效率的责任。在当前运营中，每个扇区由一个单独的 ATCO 控制，他们负责确保该扇区内的飞行运营安全高效。ATCOs 了解每架飞机的移动和意图，保持对潜在安全违规情况的态势感知，并接受有效应对此类情景的训练。不同扇区的 ATCO 之间的通信通常在飞机交接/接管过程中或在飞机因天气或其他因素偏离指定报告点时发生。在 FCOs 中，ATCOs 的责任保持不变，但与现有的结构化空域运营的特点显著不同。处理流量的 ATCOs 必须确保流量之间以及同一流量内的飞机之间的安全间隔。这使得它成为一个双层次的冲突解决问题。因此，需要更高的态势感知来监控并确保两个级别的安全。此外，不同流量管理 ATCOs 之间的协作方法始终是必要的。对于冲突解决，机动类型、机动频率以及来自流量内部和流量之间的信息共享等因素也变得至关重要。

With the fundamental differences in sector-based and flow-centric air traffic operations and the increasing air traffic demand, the existing air traffic conflict resolution advisory tools based on structured airspace are not applicable in a flow-centric setting. To address this challenge, we propose a two-level model-free and learning-based approach to resolve air traffic conflicts using deep reinforcement learning (DRL). The proposed approach formulates the conflict resolution task as a decision-making problem within a large and complex action space. It employs a DRL-based policy to ensure safe separation in inter-flow conflicts and a self-stabilizing graph structure to ensure safe separation in intra-flow conflicts. By combining speed and heading maneuvers, our approach ensures safe separations between aircraft and facilitates conflict resolution in FCOs.

随着基于扇区的空中交通运行和基于流的空中交通运行的根本差异以及空中交通需求的增加，现有基于结构化空域的空中交通冲突解决咨询工具在基于流的设置中不适用。为了应对这一挑战，我们提出了一种无模型、基于学习的双层方法，使用深度强化学习 (DRL) 解决空中交通冲突。所提出的方法将冲突解决任务公式化为在一个大而复杂的动作空间内的决策问题。它采用基于 DRL 的策略来确保流间冲突中的安全间隔，并采用自稳定图结构来确保流内冲突中的安全间隔。通过结合速度和航向机动，我们的方法确保了飞机之间的安全间隔，并促进了流量集中操作 (FCOs) 中的冲突解决。

II. BACKGROUND

II. 背景

A.Air traffic conflict resolution in flow-centric airspace

A. 基于流的空域中的空中交通冲突解决

According to the existing definitions, flow-centric operations (FCOs) involve groups of aircraft operating on multiple airways, guided by their geographical characteristics [3]. Based on the insights and reasoning derived from the limited literature on FCOs, we formulate the FCOs as illustrated in Figure 1. The flow routes are determined based on future demands, and the flow size (width and aircraft number) may vary depending on the traffic volume. Figure 1 illustrates the flow routes within a specific airspace region, where the width of each flow route represents the level of traffic demand on that particular route. Furthermore, in order to accommodate increased airspace requirements, a flow route may be extended to additional flight levels. To demonstrate the potential conflicts that may arise, Figure 1(b) presents an enlarged view of two flows, with arrows indicating the direction of flow for each route.

根据现有的定义，基于流的运行 (FCOs) 涉及在多条航路上运行的飞机群体，由它们的地理特征引导 [3]。基于对 FCOs 有限文献的见解和推理，我们制定了如图 1 所示的 FCOs。流量路线根据未来需求确定，流量大小 (宽度和飞机数量) 可能根据交通量而变化。图 1 说明了特定空域区域内的流量路线，其中每条流量路线的宽度代表该特定路线上的交通需求水平。此外，为了满足增加的空域需求，流量路线可以扩展到额外的飞行高度。为了展示可能出现的潜在冲突，图 1(b) 展示了两条流的放大视图，箭头指示每条路线的流向。

With multi-level dimensions of each flow, the most preferred resolution would involve either flow speed change or heading change, or a combination of both. Furthermore, from the operational aspect, there are significant challenges that must be addressed. For instance, the cruise speed of different aircraft in a flow may vary, resulting in dynamic flow structures. In other words, the flow sizes may increase or decrease depending on the aircraft speeds. This may lead to situations involving loss of separation within the flows. Thus, along with monitoring inter-flow separations, a mechanism to further ensure safe separation within the flows is also required. Decisions should also be made whether, in case of a detected conflict, the maneuver should be implemented by the entire flow/both flows or just the affected aircraft. This will govern the interactions between the ATCOs and the aircraft, necessitating mechanisms for splitting and merging flow for conflict resolution. It also involves detecting and resolving potential secondary conflicts. Overall, at first glance, FCOs may appear similar to conventional structured airspace operations. However, upon deeper analysis, the underlying complexities of FCOs become evident.

每个流量具有多级维度，最理想的解决方案将涉及流量速度变化或航向变化，或者两者的结合。此外，从运营角度来看，必须解决一些重大挑战。例如，流量中不同飞机的巡航速度可能会有所不同，从而导致动态的流量结构。换句话说，流量大小可能会根据飞机速度的增加或减少而增大或减小。这可能导致流量内失去间隔的情况。因此，除了监控流量间的间隔外，还需要一种机制来进一步确保流量内的安全间隔。在检测到冲突的情况下，还应决定是否整个流量/两个流量或仅受影响的飞机实施机动。这将决定空中交通管制员 (ATCOs) 与飞机之间的交互，需要分割和合并流量的机制来解决冲突。这也涉及检测和解决潜在的二次冲突。总的来说，乍一看，流量管理 (FCOs) 可能与传统的结构化空域运营相似。然而，在深入分析后，FCOs 的内在复杂性变得明显。

B.Air traffic conflict resolution in structured airspace

B. 结构化空域中的空中交通冲突解决

ATCOs are responsible for ensuring safe and efficient flight operations, and they must always maintain a safe separation distance between any two aircraft. In an en-route flight phase, a conflict or loss of separation happens when the distance between two aircraft is less than the separation standard, such as 5 nautical miles (NM) laterally and 1000 feet vertically. When a potential of a loss of separation for

Yash Guleria, Duc-Thinh Pham and Sameer Alam are with the School of Mechanical and Aerospace Engineering, Nanyang Technological University Singapore. yash009@e.ntu.edu.sg, dtpham@ntu.edu.sg, sameeralam@ntu.edu.sg

Yash Guleria、Duc-Thinh Pham 和 Sameer Alam 均任职于新加坡南洋理工大学机械与航空工程学院。电子邮件:yash009@e.ntu.edu.sg, dtpham@ntu.edu.sg, sameeralam@ntu.edu.sg

a look-ahead window is detected, ATCOs issue resolution advisories to one or both aircraft, which may involve instructions for speed changes, heading changes, flight level changes, or a combination thereof.

空中交通管制员 (ATCOs) 负责确保安全高效的飞行操作，并且他们必须始终在任意两架飞机之间保持安全的间隔距离。在航路飞行阶段，当两架飞机之间的距离小于间隔标准，例如横向 5 海里 (NM) 和垂直 1000 英尺时，就会发生冲突或失去间隔。当检测到前瞻窗口内可能失去间隔时，ATCOs 会向一架或两架飞机发出解决建议，这可能包括速度变化、航向变化、飞行高度变化或这些变化的组合。

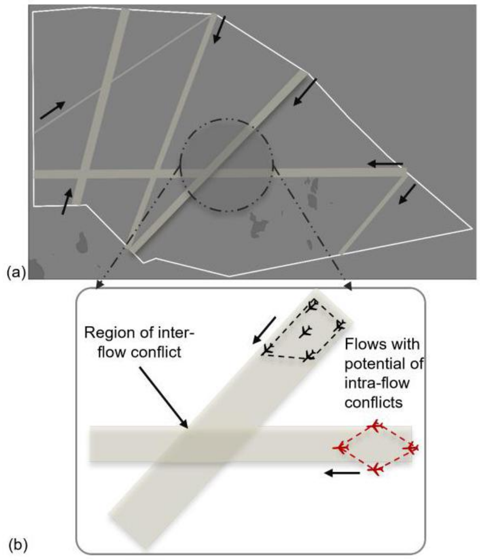

Fig. 1: An illustration of flow-centric operations (FCOs). (a) Multiple flow routes with directions in a given airspace: the width of each flow depicts the corresponding traffic demand passing through it. (b) Magnified view of two crossing flows with the potential of inter and intra-flow conflicts.

图 1: 流中心操作 (FCOs) 的示意图。(a) 在给定空域中的多个流向及方向: 每条流的宽度表示通过其的相应交通需求。(b) 两个交叉流的放大视图，具有潜在的流间和流内冲突。

To address the growing demands of air transportation, various approaches have been proposed in the literature for developing automation tools to support air traffic control operations. These approaches encompass mathematical models as well as learning-based methods. Heuristic methods, such as those explored by Durand et al. [4], Chiang et al. [5], and Durand and Eliot [6], have been investigated for air traffic conflict resolution involving multiple aircraft. A comprehensive review of these methods is provided by Kuchar and Yang [7]. Additionally, Liu et al. [8] and Allignol et al. [9] have proposed large-scale conflict resolution models. Liu et al. [8] employ an aircraft location network and restrict resolution maneuvers to velocity adjustments, while Allignol et al. [9] allow for 3D conflict resolution with limited uncertainty. Stollenwerk et al. [10] propose the use of quantum heuristics to solve a simplified version of the Air Traffic Management conflict-resolution problem, focusing on wind-optimal trajectories and minimizing trajectory adjustments. However, these mathematical models for conflict resolution suffer from notable limitations. Firstly, they require complete knowledge of the mapping between conflict scenarios and appropriate maneuvers, making them highly complex and less effective in situations with high uncertainty where comprehensive knowledge of environmental factors cannot be achieved. Secondly, these models heavily rely on standardized input scenarios and do not possess the ability to adapt and learn when confronted with novel or non-standard situations. We attempt to address these drawbacks by using a DRL approach where the learning model does not require prior knowledge of how to efficiently resolve a conflict and can self-evolve when exposed to dynamic, unseen scenarios.

为了应对航空运输需求的不断增长，文献中提出了各种方法来开发自动化工具以支持空中交通管制操作。这些方法包括数学模型以及基于学习的方法。启发式方法，如 Durand 等人 [4]、Chiang 等人 [5] 以及 Durand 和 Eliot[6] 所研究的方法，已被用于涉及多架飞机的空中交通冲突解决。Kuchar 和 Yang[7] 提供了这些方法的全面回顾。此外，Liu 等人 [8] 和 Allignol 等人 [9] 提出了大规模冲突解决模型。Liu 等人 [8] 使用飞机位置网络并将解决机动限制在速度调整上，而 Allignol 等人 [9] 允许进行具有有限不确定性的三维冲突解决。Stollenwerk 等人 [10] 提出使用量子启发式方法来解决空中交通管理冲突解决问题的简化版本，重点放在风最优轨迹和最小化轨迹调整上。然而，这些用于冲突解决的数学模型存在明显的局限性。首先，它们需要完整了解冲突场景与适当机动之间的映射，这使得它们在无法获得环境因素全面知识的高不确定性情况下变得高度复杂且效果不佳。其次，这些模型严重依赖标准化的输入场景，

且没有能力适应和学习 novel 或非标准情况。我们试图通过使用深度强化学习 (DRL) 方法来解决这些缺点，其中学习模型不需要预先了解如何有效地解决冲突，且在面临动态、未见过的场景时可以自我进化。

Conventional techniques experience difficulty with large, continuous and discrete state and action spaces in decision-making situations such as air traffic conflict resolution. Deep Reinforcement Learning (DRL) combines deep learning and reinforcement learning to tackle decision-making problems with high-dimensional state and action spaces that were previously difficult to handle. Researchers such as Brittain and Wei [11] have proposed deep-distributed multi-agent reinforcement learning frameworks for resolving conflicts in the en-route phase by adjusting aircraft speeds. These frameworks can detect and resolve conflicts in high-density, stochastic, and dynamic en-route sectors that involve multiple intersections. Mollinga and van Hoof [12] have introduced an RL algorithm capable of steering an arbitrary number of aircraft through unstructured three-dimensional airspace every 5 seconds while avoiding conflicts and collisions. In a recent development, Pham et al. [13] have presented an RL approach for conflict resolution that can adapt to varying levels of surrounding traffic and uncertainty.

传统技术在处理具有大范围、连续和离散状态及动作空间的决策情境 (如空中交通冲突解决) 时遇到困难。深度强化学习 (DRL) 结合了深度学习和强化学习，用以解决具有高维状态和动作空间的决策问题，这些问题之前很难处理。诸如 Brittain 和 Wei [11] 等研究者提出了深度分布式多代理强化学习框架，用于通过调整飞机速度解决航路阶段的冲突。这些框架能够检测并解决涉及多个交叉点的高密度、随机和动态航路区域中的冲突。Mollinga 和 van Hoof [12] 引入了一种 RL 算法，能够每 5 秒引导任意数量的飞机通过无结构的立体空域，同时避免冲突和碰撞。近期，Pham 等人 [13] 提出了一种冲突解决的 RL 方法，能够适应周围交通和不确定性的不同水平。

The background highlights that the future concept of operations in flow-centric airspace introduces significant differences that render existing approaches, including mathematical and learning-based methods, unsuitable for addressing air traffic conflict resolution in such a setting. While the current DRL-based approaches show promising results in dense and stochastic structured environments, none of the existing approaches, are suitable for addressing the air traffic conflict resolution problem in the future concept of operations, i.e. FCOs. To the best of our knowledge, this work presents the first attempt to conceptualize and propose a novel two-level conflict resolution approach specifically designed for flow-centric air traffic conflict resolution. The proposed approach takes into account the uncertainties associated with flow size and speed, as well as the dynamic evolution of flow structures, to ensure safe separation both at the inter-flow and intra-flow levels.

背景部分突显了以流量为中心的空域未来运行概念引入了重大差异，使得现有的方法，包括数学和基于学习的方法，不适合解决此类环境下的空中交通冲突解决。虽然当前的基于 DRL 的方法在密集和随机的结构化环境中显示出有希望的结果，但现有的方法均不适合解决未来运行概念中的空中交通冲突解决问题，即流量为中心的运行概念 (FCOs)。据我们所知，这项工作是首次尝试概念化并提出了专门为流量为中心的空中交通冲突解决设计的创新双层冲突解决方法。该方法考虑了流量大小和速度的不确定性，以及流量结构的动态演变，以确保在流间和流内层面均保持安全间隔。

III. OVERVIEW: PROBLEM DESCRIPTION AND SCOPE

III. 概述: 问题描述与范围

This study introduces a novel approach for resolving air traffic conflicts in flow-centric airspace, employing a combination of a model-free Deep Reinforcement Learning (DRL) policy and a self-stabilizing graph structure. The problem of air traffic conflict resolution is formulated as a sequential decision-making task, requiring a series of decisions to be made over time to resolve potential conflicts at both the inter-flow and intra-flow levels, while adhering to the flow plans and subsequently reaching the destination. This is a centralized coordination scheme where an agent takes actions to ensure safe separation between the flows.

本研究介绍了一种在以流量为中心的空域中解决空中交通冲突的新方法，该方法结合了无需模型的深度强化学习 (DRL) 策略和自稳定图结构。空中交通冲突解决问题被构建为一个顺序决策任务，需要在一段时间内连续做出决策，以解决流量之间以及流量内部的潜在冲突，同时遵循流量计划，最终到达目的地。这是一个集中协调方案，其中一个代理采取行动以确保流量之间的安全间隔。

In this context, a flow plan is defined similarly to a flight plan, where each flow (representing a group of aircraft) is expected to reach the subsequent reporting waypoint at a specified time, ensuring minimal chances of secondary flow conflicts in the airspace. The formal framework of Markov Decision Processes (MDPs) is utilized to model the sequential decision-making problem, incorporating states, actions, and rewards. The selected actions not only impact immediate rewards but also influence subsequent states and, consequently, future rewards. In the proposed approach, the system's state, subsequent actions, and rewards are influenced not only by the learned policy of the agent but also by the output of the self-stabilizing graph structure, which guarantees intra-flow safe separation. This integration ensures the effective resolution of air traffic conflicts in FCOs. Figure 2 shows the overall concept diagram for the proposed approach.

在此背景下，流量计划被定义得类似于飞行计划，其中每个流量 (代表一组飞机) 都应在指定时间到达后续报告航点，以确保空域中二次流量冲突的最小化机会。使用了马尔可夫决策过程 (MDPs) 的正式框架来建模顺序决策问题，包括状态、动作和奖励。所选动作不仅影响即时奖励，还影响后续状态，进而影响未来奖励。在提出的方法中，系统的状态、后续动作和奖励不仅受到代理学习到的策略的影响，还受到自稳定图结构输出的影响，这保证了流量内部的安全间隔。这种整合确保了在流量控制区 (FCO) 中有效解决空中交通冲突。图 2 展示了所提方法的整体概念图。

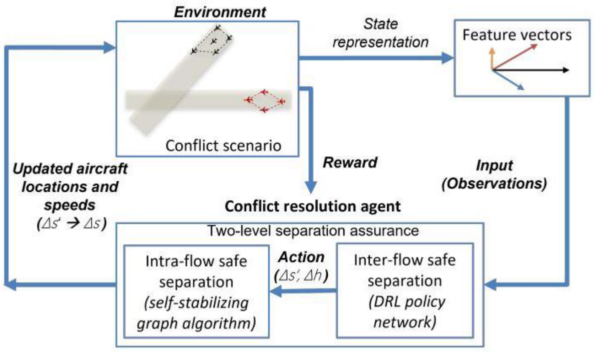

Fig. 2: A concept diagram for the interaction between the agent and the learning environment. Scenarios involving interflow and intra-flow conflict are generated and the vector representation of the extracted features is used by the agent to propose an action based on the learned policy, thereby reaching a new state and receiving a certain reward. The updated actions pass into a self-stabilizing algorithm which ensures intra-flow safe separation and outputs the updated location and speed of the aircraft in both flows.

图 2: 代理与学习环境交互的概念图。生成涉及流量之间和流量内部冲突的场景，提取特征的向量表示被代理用来根据学习到的策略提出一个动作，从而达到新的状态并获得一定的奖励。更新的动作传递给自稳定算法，确保流量内部的安全间隔，并输出两个流量中飞机更新后的位置和速度。

The present study focuses on the specific scope of addressing conflict resolution in FCOs involving two crossing flows with varying numbers of aircraft and speeds within each flow. The dynamic nature of these flows, which evolve over time steps due to different aircraft cruise speeds, is taken into account. The action space includes options such as heading change, speed change, or a combination of both, applicable to one or both flows. These actions are based on the nominal aircraft performance parameters in the cruise flight phase. Furthermore, since the research focuses on en-route flight phase with sufficient inter-flow and intra-flow separation, effect of factors such as aircraft wake turbulence is not considered. In this approach, the selected action by the agent is implemented on the entire flow, affecting each aircraft within the flow. The allowed range of aircraft speeds is determined based on Automatic Dependent Surveillance-Broadcast (ADS-B) data (Figure 3). By training a policy that maximizes the cumulative reward over time, the proposed approach aims to effectively resolve air traffic conflicts while minimizing deviations from the flow plan. The model-free nature of the approach enables it to handle the added uncertainties, and achieve the global objectives of safety, efficiency, and adherence to flow plans.

当前研究专注于解决涉及两个交叉流量中飞机数量和每个流量中速度不同的飞行冲突解决的具体范围。考虑到这些流量随时间步骤变化，因为不同飞机的巡航速度不同，具有动态性质。动作空间包括改变航向、改变速度或两者组合的选项，适用于一个或两个流量。这些动作基于巡航飞行阶段飞机的名义性能参数。此外，由于研究聚焦于航路飞行阶段，且流量间和流量内分离充足，因此不考虑诸如飞机尾流湍流等因素的影响。在这种方法中，智能体选择的动作应用于整个流量，影响流量中的每架飞机。允许的飞机速度范围是根据自动依赖监视广播 (ADS-B) 数据确定的 (见图 3)。通过训练一个最大化随时间累积奖励的策略，所提出的方法旨在有效解决空中交通冲突，同时最小化偏离流量计划。该方法的无模型特性使其能够处理增加的不确定性，并实现安全性、效率和遵守流量计划的全局目标。

IV. METHODOLOGY

IV. 方法论

A. Conflict scenario generation

A. 冲突场景生成

In this study, we use one day's ADS-B data of flights in the South East Asian region to obtain the inter-flow and intra-flow speeds for flow movements. The cruise speed distribution obtained from the data is depicted in Figure 3. For each conflict scenario, inter-flow velocities are randomly selected from this distribution. Thus, the inter-flow velocities range between 360 knots (kts) and 540 kts. Additionally, the velocity of each aircraft within a flow, known as the intra-flow velocity, is sampled from a normal distribution. The mean of this distribution is set to the flow speed, while the standard deviation is set to 30kts . This approach ensures that there is variation in aircraft velocities within a flow, reflecting real-world operational conditions. In accordance with the definition provided by the International Civil Aviation Organization (ICAO), a crossing conflict is characterized by conflicts occurring within the range of 45° to 135° on both, the left and the right of the reference aircraft [14]. Therefore, the scenarios considered in this study involve crossing conflicts at three specific angles: $45°, 90°$, and 135° , in an abstract airspace. The time interval preceding the occurrence of a conflict is approximately 19-23 minutes, allowing for a sufficient look-ahead time. To introduce variability, the starting points of each flow are randomized by applying certain offsets, ensuring that the loss of separation occurs at different locations in each scenario. Additionally, the number of aircraft within each flow ranges from 3 to 10 in each scenario. This range of is selected to ensure sufficient scenario variability while maintaining a reasonable flow size. These aircraft are randomly generated within a square region of side 20NM at the start of the scenario. These aforementioned variations are included to enhance the generalization capabilities of the proposed model. For the remainder of this work, the terms 'conflict scenario' and 'episode' are used interchangeably. Figure 4 shows a representative conflict scenario with two flows. Here, $D_{\min}$ is the minimum distance between the flows, and h_{i1} and h_{i2} are the initial headings of the flows. These flows must reach their destinations (reporting points) based on the flow plans.

在本研究中，我们使用东南亚地区一天内的 ADS-B 航班数据来获取流量运动中的流间速度和流内速度。从数据中获得的巡航速度分布如图 3 所示。对于每种冲突场景，流间速度从该分布中随机选取。因此，流间速度范围在 360 节 (kts) 至 540 kts 之间。此外，每个流量中的飞机速度，即流内速度，是从正态分布中采样的。该分布的平均值设定为流量速度，而标准差设定为 30kts 。这种方法确保了流量内飞机速度的差异性，反映了现实世界的运行条件。根据国际民用航空组织 (ICAO) 提供的定义，交叉冲突是指冲突发生在参考飞机左右两侧的 45° 至 135° 范围内 [14]。因此，本研究考虑的场景涉及在抽象空域中三个特定角度的交叉冲突：$45°, 90°$ 和 135° 。冲突发生前的间隔时间约为 19-23 分钟，允许足够的前瞻时间。为了引入变化，每个流起点通过应用特定偏移量随机化，确保每个场景中失去间隔的位置不同。此外，每个场景中每个流量内的飞机数量范围从 3 到 10 架。选择这个范围是为了确保场景的充分变化，同时保持合理的流量大小。这些飞机在场景开始时随机生成在边长为 20NM 的正方形区域内。上述变化包括在内是为了提高所提出模型的泛化能力。在本研究的其余部分，术语“冲突场景”和“情节”可互换使用。图 4 展示了具有两个流量的代表性冲突场景。在此，$D_{\min}$ 是流量之间的最小距离，h_{i1} 和 h_{i2} 是流量的初始航向。这些流量必须根据流量计划到达其目的地 (报告点)。

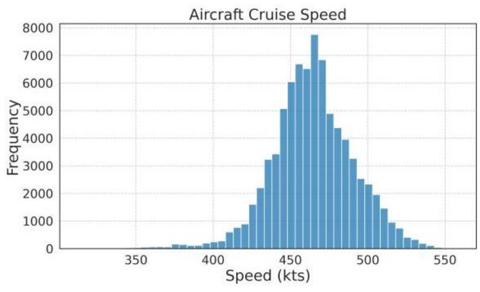

Fig. 3: Distribution of aircraft cruise speed from the ADS-B data. For each scenario, inter-flow and intra-flow speeds are sampled from this distribution.

图 3: 从 ADS-B 数据中得到的飞机巡航速度分布。对于每种情况，从该分布中抽取流量之间的速度和流量内部的速度。

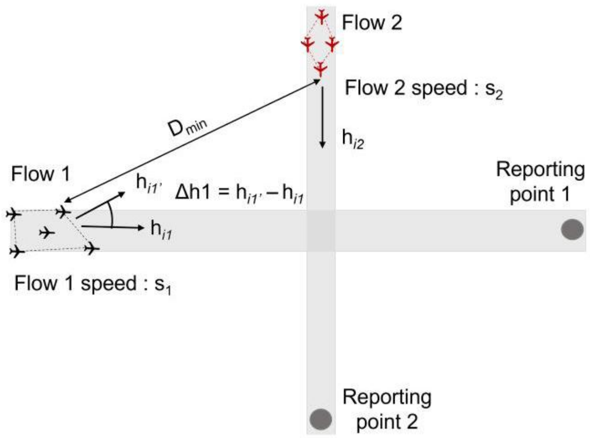

Fig. 4: An illustration of the conflict scenarios with two flows in a 90° crossing conflict. △h1 represents the difference between the original heading of the flow (h_{i1}) and the current heading ($h_{i1'}$) based on the selected action.

图 4: 两个流量在 90° 交叉冲突情况下的冲突场景示意图。△h1 表示流量 (h_{i1}) 的原始航向与基于选定动作的当前航向 ($h_{i1'}$) 之间的差异。

B. Learning Mechanism

B. 学习机制

The learning mechanism has several key components, which include providing the agent with sufficient information to support decision-making, evaluating the agent's actions, and providing feedback to the agent in the form of a reward. To facilitate the learning process for resolving conflicts, we have developed a scenario generator that produces conflict scenarios (discussed in the previous subsection) and represents them in a way that can be perceived by the agent. The agent's actions are defined and the mapping from these actions to the maneuvers taken by the agent is established. A reward function has been designed to assess the effectiveness of the maneuvers suggested by the agent. These components are discussed in detail as follows:

学习机制有几个关键组成部分，包括为代理提供足够的信息以支持决策，评估代理的行为，并以奖励的形式向代理提供反馈。为了促进解决冲突的学习过程，我们开发了一个场景生成器，该生成器产生冲突场景 (在前一个子节中讨论) 并以代理能够感知的方式表示这些场景。定义了代理的行为，并建立了这些行为与代理采取的机动动作之间的映射。已经设计了一个奖励函数来评估代理提出的机动动作的有效性。以下详细讨论这些组成部分:

1) Observation space: The observation space contains all information that the agent can receive at each step in order to decide the next action. In this work, the observation space includes information like the distance between the two flows, the number of aircraft in each flow, the initial (at simulation start) and current headings and speeds of the two flows, the cross-track deviations of the flows (i.e the distance of each flow from its original path) and the distance of the flows from the target locations. Furthermore, to indicate the current direction of the flow with respect to the original flow path, the change in the heading (positive or negative) is also added to the observation space. Overall, this is a vector of 17 elements.

1) 观察空间: 观察空间包含代理在每一步可以接收的所有信息，以便决定下一个动作。在这项工作中，观察空间包括诸如两个流量之间的距离、每个流量中的飞机数量、两个流量的初始 (模拟开始时) 和当前航向与速度、流量的偏航偏差 (即每个流量偏离原始路径的距离) 以及流量距离目标位置的距离等信息。此外，为了指示流量相对于原始流量路径的当前方向，航向的变化 (正值或负值) 也被添加到观察空间中。总体而言，这是一个包含 17 个元素的向量。

2) Action space: in the current work, the agent takes two actions at each time step i.e. one for each flow. One time step implies one minute in the scenarios. The actions include speed change, heading

change, or a combination of the two. Given the initial flow speed, s, and the initial heading of the flow, α , and the intra-flow speeds, the agent has the following available action space for each flow:

2) 动作空间: 在当前工作中，代理在每个时间步 i.e. 每个流中采取两个动作。一个时间步意味着场景中的一分钟。动作包括速度变化、航向变化，或两者的组合。给定初始流速 s 和初始流向航向 α ，以及流内速度，代理对于每个流有以下可用的动作空间:

$$A_S = [\delta s_-, 0, \delta s_+]$$

$$A_h = [\delta \alpha_-, 0, \delta \alpha_+]$$

At each step, a speed change of ±30kts or 0kts , a heading change of ±30° or 0° , or a combination of these can be made. Thus, a total of 9 actions are available for each flow, scaling the total number of available actions for both flows to 81 . The action space is encoded in Table I. Due to such a large action space, DRL is a suitable approach for conflict resolution.

在每一步，可以进行 ±30kts 或 0kts 的速度变化，±30° 或 0° 的航向变化，或者这些动作的组合。因此，每个流有 9 个可用的动作，将两个流的总可用动作数扩展到 81。动作空间在表 I 中编码。由于动作空间如此之大，深度强化学习 (DRL) 是解决冲突的合适方法。

TABLE I: Encoding of the agent's action space.

表 I: 代理动作空间的编码。

Encoding	Action(δs, δh)	Encoding	Action(δs, δh)
0	-30 kts, -30°	5	0 kts, 30°
1	-30 kts, 0°	6	30 kts, -30°
2	-30 kts, 30°	7	30 kts, 0°
3	0 kts, -30°	8	30 kts, 30°
4	0 kts, 0°		

编码	动作（ δs, δh ）	编码	动作（ δs, δh ）
0	-30 节，-30°	5	0 节，30°
1	-30 节，0°	6	30 节，-30°
2	-30 节，30°	7	30 节，0°
3	0 节，-30°	8	30 节，30°
4	0 节，0°		

3) Reward mechanism: The reward mechanism is based on two primary criteria: safety and efficiency. Thus, a maneuver that successfully separates the two flows is given a positive reward and any maneuver that causes loss of separation is given a heavy negative reward. On similar lines, maneuvers that reduce the induced delay and allow the flows to successfully reach the destination are promoted with positive rewards. An inter-flow loss of separation in the simulations is a situation where the separation between flows is less than 5 nautical miles (NM). Efficiency is measured in the context of the deviation from the flow plan, the crosstrack deviations and closest point of approach between the flows. The agent is also penalized for taking excessive actions while resolving the conflicts. At each step, the agent receives a reward $R_s(s', a)$, which is based on the action a and the resultant state vector s' . This reward is based on the number of actions ($n_{\text{ actions }}$) and the deviation from the flow plan $\left(dev_{(\text{ flowplan })}\right)$ which is estimated by the change in the number of steps required by both flows to reach the destination, as compared to the initial number of steps (based on the flow velocities at the simulation start).

3) 奖励机制: 奖励机制基于两个主要标准: 安全性和效率。因此，成功分离两个流的操作会被给予正奖励，而任何导致分离损失的操作会被给予重罚负奖励。类似地，减少诱导延迟并允许流成功到达目的地的操作会得到正奖励。在模拟中，流之间的分离损失是指流之间的距离小于 5 海里 (NM) 的情况。效率是在偏离流计划、横向偏差和流之间的最近点接近的背景下进行衡量的。在解决冲突时，如果代理采取过度的行动，也会受到惩罚。在每一步，代理都会收到一个基于动作 $R_s(s', a)$ 和结果状态向量 a 的奖励 s' 。这个奖励基于动作的数量 ($n_{\text{ actions }}$) 和偏离流计划的程度 $\left(dev_{(\text{ flowplan })}\right)$ ，后者是通过比较两个流到达目的地所需的步数的变化 (基于模拟开始时流的速度) 来估算的。

$$R_s(s', a) = \begin{cases} -\alpha * n_{\text{ actions }} \\ -\beta * \text{dev}_{(\text{ flowplan })} \end{cases} \tag{1}$$

The agent receives rewards at episode termination based on the criteria in equation 2. Here, $d_{\min}$ refers to the distance between the two flows and $d_{\text{ sep }}$ is the minimum safety separation. Invalid maneuvers involve aircraft speeds and headings outside the specified range. $d_{(f1+f2)}$ is the sum of the distances of the flows from their respective destination. An episode finished when both the flows reach their destinations. The total reward after each episode is represented by equation 3, where N is the total steps in the episode.

代理在剧集结束时根据方程 2 中的标准接收奖励。在这里，$d_{\min}$ 指两个流之间的距离，d_{sep} 是最小安全间隔。无效操作涉及超出规定范围的飞机速度和航向。$d_{(f1+f2)}$ 是两个流各自目的地距离的总和。当两个流都到达目的地时，剧集结束。每个剧集之后的总奖励由方程 3 表示，其中 N 是剧集中的总步数。

$$R_t\left(s', a\right)=\begin{cases}-\gamma & \text{if } d_{\min}<d_{\text{ sep}} \\ -\delta & \text{if } a \text{ is an invalid maneuver} \\ \lambda & \text{if both flows reach the destination} \\ \frac{\Gamma}{d_{(f1+f2)} \text{ otherwise}} & \end{cases} \tag{2}$$

$$R_T=\sum_{n=1}^{N} R_s+R_t \tag{3}$$

Here, the values of the parameters $\alpha, \beta, \gamma, \delta, \lambda$ and Γ are 0.002,0.0002,10,5,10, and 5, respectively. These values were obtained after multiple iterations and model performance evaluations.

在这里，参数 $\alpha, \beta, \gamma, \delta, \lambda$ 和 Γ 的值分别是 0.002、0.0002、10、5、10 和 5。这些值是在经过多次迭代和模型性能评估后获得的。

C. Learning algorithm

C. 学习算法

FCOs require inter-flow and intra-flow level air traffic conflict resolution. Therefore, in this work, the learning algorithm also consists of two stages: inter-flow conflict resolution using a reinforcement learning algorithm and intra-flow safety separation using a self-stabilizing graph algorithm. Proximal Policy Optimization (PPO) [15] has been used to train the agent for inter-flow air traffic conflict resolution, due to its faster convergence as compared to off-policy methods, for our specific research problem. PPO is a deep reinforcement learning algorithm that is widely used in various domains of intelligent transportation systems, has achieved state-of-the-art performance in several benchmarks. It is a model-free policy optimization algorithm that operates by updating the policy with a clipped surrogate objective function to ensure stable training and avoid large policy updates. PPO has several advantages, including the ability to handle large, continuous, or discrete action spaces, providing guarantee on monotonic improvements in the objective, and good sample efficiency.

FCOs 需要解决流间和流内的空中交通冲突。因此，在这项工作中，学习算法也分为两个阶段: 使用强化学习算法解决流间冲突，以及使用自稳定图算法实现流内安全间隔。由于在特定研究问题中，与离策略方法相比，Proximal Policy Optimization (PPO) [15] 具有更快的收敛速度，因此已被用于训练解决流间空中交通冲突的智能体。PPO 是一种深度强化学习算法，在智能交通系统的多个领域得到广泛应用，并在多个基准测试中取得了最先进的表现。它是一种无需模型的策略优化算法，通过更新带有剪辑替代目标函数的策略来确保稳定训练并避免策略更新过大。PPO 具有多个优点，包括能够处理大型的、连续的或离散的动作空间，为目标函数提供单调改进的保证，以及良好的样本效率。

To ensure intra-flow safe separation, we represent each flow as a self-stabilizing graph structure. Self-stabilizing algorithms are a fundamental branch of fault-tolerant computing and were first introduced by Dijkstra [16]. Based on pre-defined criteria, two states for the system can be defined, which are (i) the legitimate state, and (ii) the illegitimate state. Self-stabilizing algorithms are resilient to transient faults i.e. if a perturbation brings the system to an illegitimate state, then the system must be able to again reach a legitimate state after a finite number of moves without any external intervention. These have been used to solve synchronization problems [17], constructing breadth-first trees [18] and graph explorations [19]. From the context of air traffic conflict resolution, an illegitimate state implies a situation where aircraft witness a loss of safe separation. A state where all the aircraft in a flow are safely separated is considered a legitimate state.

为了确保流内安全隔离，我们将每个流表示为一个自稳定的图结构。自稳定算法是容错计算的基本分支，最初由 Dijkstra [16] 提出。基于预定义的标准，可以定义系统的两种状态，分别是 (i) 合法状态，和 (ii) 非法状态。自稳定算法能够抵御短暂故障，即如果扰动使系统进入非法状态，那么系统必须在没有任何外部干预的情况下，经过有限次数的移动再次达到合法状态。这些算法已被用于解决同步问题 [17]、构造广度优先树 [18] 和图探索 [19]。在航空冲突解决的背景下，非法状态意味着飞机目睹了安全间隔的丧失。一个流程中所有飞机都被安全隔离的状态被认为是合法状态。

Algorithm 1 Intra-flow Separation Algorithm

算法 1 流内隔离算法

procedure INTRA-FLOW SEPARATION $(G,(x,y))$

过程 INTRA-FLOW SEPARATION $(G,(x,y))$

sorted_nodes $\leftarrow$ sortNodes $(G.V,(x,y))$

sorted_nodes $\leftarrow$ sortNodes $(G.V,(x,y))$

for i in sorted_nodes do

对于 i 在 sorted_nodes 中的每一个元素执行

basenode $\leftarrow i$

basenode $\leftarrow i$

for j in sorted_nodes do

对于 j 在 sorted_nodes 中的每一个元素执行

if index $j >$ index i then

如果 index $j >$ index i 那么

dist $\leftarrow$ calculateDistance(basenode, j)

dist $\leftarrow$ calculateDistance(basenode, j)

while dist $<$ safetybuffer do

当 dist $<$ safetybuffer 时

delta $\leftarrow$ safetybuffer - distance

delta $\leftarrow$ safetybuffer - distance

$j' \leftarrow$ moveNode $(j,$ delta $, \alpha)$

dist $\leftarrow$ calculateDistance (basenode, j')

dist $\leftarrow$ calculateDistance (basenode, j')

end while

结束 while 循环

end if

结束 if 条件

sorted_nodes $\leftarrow$ sortNodes $(G.V,(x,y))$

sorted_nodes $\leftarrow$ sortNodes $(G.V,(x,y))$

end for

结束 for 循环

end for

结束 for 循环

end procedure

结束过程

The graph spacing algorithm to ensure intra-flow separation is depicted in Algorithm 1. Let $G = (V, E)$ represent a uni-directed, complete, weighted graph for each flow, where V represents the vertices/nodes (the aircraft) and E represents the edges, with the distance between the nodes implying the weights. At inter-flow level, an action is taken to change the state of the environment each step of the episode based on the current observations received by the agent. In other words, this action updates the locations of the aircraft in the flow. At intra-flow level, since the aircraft's speeds differ, this may cause loss of separation if the trailing aircraft in a flow are flying faster, as the episode progresses. Thus, before the state is updated after each step, the self-stabilization algorithm checks if the distance between any two aircraft in a flow is below a specified threshold. If yes, the algorithm provides updated speed changes and consequently, the updated locations of the aircraft, which are then used to transition to a new state. For this, the farthest aircraft with respect to the destination is taken as reference for sorting the nodes (Algorithm 1). Each episode always starts with a legitimate state. The threshold is currently set to 7 NM. This accommodates the operational uncertainties associated with aircraft location and speed.

图间距算法用于确保流内分离，如图 1 所示。令 $G = (V, E)$ 表示每个流的单向、完整、加权图，其中 V 表示顶点/节点 (飞机)，E 表示边，节点间的距离表示权重。在流间层面，智能体在每个步骤根据接收到的当前观察改变环境状态。换句话说，这个动作更新了流中飞机的位置。在流内层面，由于飞机速度不同，随着情节的进行，可能会因为后续飞机飞得更快而失去分离。因此，在每一步更新状态之前，自稳定算法会检查流中任意两架飞机之间的距离是否低于指定的阈值。如果是，算法会提供更新的速度变化，进而更新飞机的位置，然后用于过渡到新状态。为此，以相对于目的地的最远飞机为参考对节点进行排序 (算法 1)。每个情节始终从一个合法状态开始。阈值目前设置为 7 海里。这考虑了与飞机位置和速度相关的操作不确定性。

V. EXPERIMENTS AND RESULTS

V. 实验与结果

A. Experimental setting

A. 实验设置

The PPO algorithm used in this work is adapted from the Stable-Baselines 3 [20]. The parameters for the PPO algorithm are in Table II. The training process consists of 3,000,000 scenarios. Each scenario lasts for a maximum of 60 time steps. Training is performed on the Intel(R) Core(TM) i9-9900X CPU which takes about 4 hours.

本工作中使用的 PPO 算法是从 Stable-Baselines 3 [20] 改编而来。PPO 算法的参数在表 II 中。训练过程包括 3,000,000 个场景。每个场景最多持续 60 个时间步。训练在 Intel(R) Core(TM) i9-9900X CPU 上进行，大约需要 4 小时。

Figure 5 shows the model's convergence after 3,000,000 scenarios. During training, the model's performance is measured by the average reward achieved and the number of steps taken to complete each episode. One episode should take approximately 45 steps to finish. Initially, the agent takes over 50 steps to reach the destination but as the iterations increase, it identifies optimal paths for both flows to reach their destinations. The number of steps for the episodes to terminate eventually stabilizes at around 45. This is because additional steps imply a deviation from the flow plan, incurring a negative reward for every additional step. On similar lines, in an ideal case with no actions and no conflicts, the theoretical maximum reward is 10 . Since actions and deviations from the flow plan incur negative rewards, the rewards stabilize around 8.5. Even though the model can achieve high average rewards after 1,000,000 iterations, its performance is unstable. Therefore, we run the simulations to continue training till the model converges.

图 5 显示了模型在 3,000,000 个场景后的收敛情况。在训练过程中，模型的性能是通过实现的平均奖励和完成每个回合所需的步数来衡量的。一个回合应该大约需要 45 步来完成。最初，代理需要超过 50 步才能到达目的地，但随着迭代次数的增加，它确定了两个流到达目的地的最优路径。回合结束所需的步数最终稳定在约 45 步左右。这是因为额外的步数意味着偏离了流计划，每多出一步都会导致负奖励。同理，在理想情况下，没有动作和冲突，理论上的最大奖励是 10。由于动作和偏离流计划都会导致负奖励，奖励最终稳定在 8.5 左右。尽管模型在经过 1,000,000 次迭代后能够实现高平均奖励，但其性能不稳定。因此，我们运行模拟以继续训练，直到模型收敛。

TABLE II: Parameters for PPO training.

表 II:PPO 训练的参数。

Parameters	Value
Training episodes	3e+6
Learning rate	3e-4
Discount factor	0.99
Clipping coefficient	0.2
ANN architecture	Multi-layer Perceptron
Optimizer	Adam
Hidden layers	64 X 64
Activation function	Tanh
N_epochs	10

参数	值
训练回合	3e+6
学习率	3e-4
折扣因子	0.99
剪切系数	0.2
神经网络架构	多层感知器
优化器	Adam
隐藏层	64 X 64
激活函数	双曲正切函数 (Tanh)
训练轮数 (N_epochs)	10

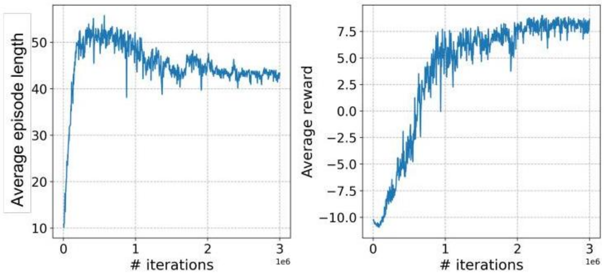

Fig. 5: Model convergence: episode length (left) and the average reward (right).
图 5: 模型收敛: 回合长度 (左) 和平均奖励 (右)。

To evaluate the trained model's performance, results are documented for 1000 test set scenarios in the following subsections. In the discussions, Flow 1 refers to the flow moving in the horizontal direction. Flow 2 refers to the second flow which intersects the path of Flow 1.

为了评估训练后模型的性能，以下小节中记录了 1000 个测试集场景的结果。在讨论中，流 1 指的是沿水平方向移动的流。流 2 指的是与流 1 路径交叉的第二流。

B. Safe separations achieved

B. 实现的安全分离

1) Inter-flow safe separation: The Air transportation system is a safety-critical system. Thus, model performance in terms of safe separations achieved is important. The model achieves a safe separation for 100% of scenarios. This implies that the policy learned by the agent is successful in resolving the conflicts despite the associated uncertainties in the speed of the flows, and their dynamic shape and size.

1) 流间安全分离: 空中运输系统是一个安全关键的系统。因此，模型在实现安全分离方面的性能很重要。模型在 100% 的场景中实现了安全分离。这意味着代理学习到的策略能够在流的速率、动态形状和大小相关的不确定性下成功解决冲突。

2) Intra-flow safe separation: Figure 6 shows the number of actions (speed adjustments) taken by the intra-flow safe separation algorithm at each time step. The figure shows the results for 10 aircraft in each flow. Due to the varying speed of the aircraft in a flow, there are instances where the intra-flow separation reduces below the specified threshold as the scenario evolves. To ensure safe separation, the algorithm updates the aircraft speed (and hence the location) before the state of the environment is changed. Furthermore, Figure 10, right, shows the minimum distance between the aircraft in Flow 1, the lowest value of which is 7 NM. Thus, the intra-flow safe separation algorithm successfully ensures that the aircraft within a flow are separated by the defined threshold at each time step.

2) 流内安全分离: 图 6 显示了流内安全分离算法在每个时间步长采取的动作数量 (速度调整)。该图显示了每个流中 10 架飞机的结果。由于流中飞机速度的不同，随着情景的发展，有时流内间隔会降至规定的阈值以下。为确保安全间隔，算法会在环境状态改变之前更新飞机速度 (从而更新位置)。此外，图 10 右侧显示了流 1 中飞机之间的最小距离，其最低值为 7 海里。因此，流内安全分离算法成功确保了在每个时间步长，流内的飞机都保持了定义的阈值间隔。

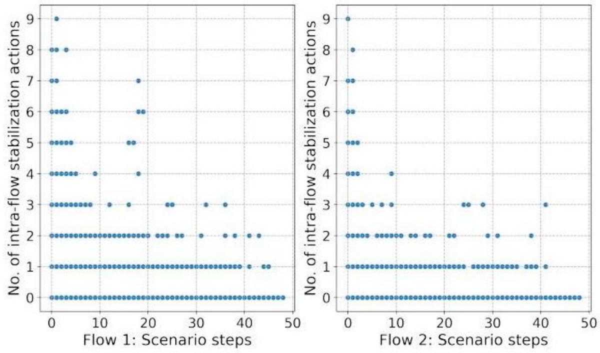

Fig. 6: Number of actions taken by the intra-flow safe separation algorithm at each time step before the state of the environment is updated.
图 6: 在环境状态更新之前，流内安全分离算法在每个时间步长采取的动作数量。

C. Maneuver efficiency

C. 操作效率

We analyze maneuver efficiency in terms of adhering to the flow plans, the cross-track deviations (CTD) during each episode, and the closest points of approach (CPA) between the two flows. Figure 7 highlights the delay distribution of the two flows. The negative values imply reaching the destination early, which is not preferred since this might also influence the events (secondary conflicts) further down the line. In terms of the absolute delays, the average values for Flow 1 and Flow 2 are 2.53 minutes and 9.49 minutes.

我们从遵循流计划、每个阶段的横向偏差 (CTD) 以及两个流之间的最近接近点 (CPA) 三个方面分析操作效率。图 7 突出了两个流的延迟分布。负值意味着提前到达目的地，这并不理想，因为这可能会影响后续的事件 (次生冲突)。就绝对延迟而言，流 1 和流 2 的平均值分别为 2.53 分钟和 9.49 分钟。

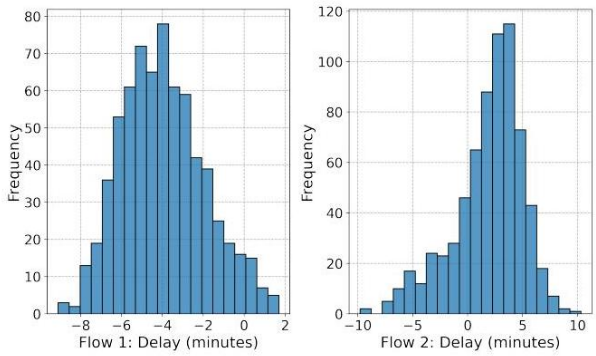

Fig. 7: Delay distribution in terms of the difference between the original flow plan and the actual flow plan.
图 7: 延迟分布，表示原始流计划与实际流计划之间的差异。

The maximum CTDs of the two flows from their original paths are shown in Figure 8. There are two potential reasons for the higher CTDs of Flow 1 in some scenarios. First, with the increase in the number of aircraft, the flow structure itself increases, given the intra-flow separation must be 7 NM. Second, the shape and size of the flows are dynamic. As each conflict scenario evolves, these attributes of the flows change depending on the velocities of the aircraft. Thus, the size of a flow changes based on the slowest and the fastest aircraft as its member.

两个流从其原始路径的最大 CTD(横截距离) 显示在图 8 中。在某些情况下，流 1 的 CTD 较高可能有以下两个潜在原因。首先，随着飞机数量的增加，考虑到流内分离必须为 7 海里，流结构本身也会增加。其次，流的形状和大小是动态的。随着每个冲突场景的发展，这些属性会根据飞机的速度发生变化。因此，流的大小会根据其中最慢和最快的飞机而改变。

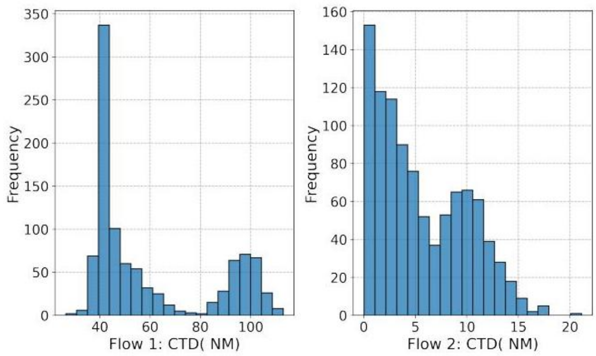

Fig. 8: The maximum cross-track deviation distribution (NM) for Flow 1 and Flow 2.

图 8: 流 1 和流 2 的最大横截偏差分布 (海里)。

Similarly, the CPA between the two flows is relatively higher. This is measured as the smallest distance between any two aircraft in the two flows (Figure 9).

同样，两个流之间的最近点距离 (CPA) 相对较高。这是通过测量两个流中任意两架飞机之间的最小距离来确定的 (见图 9)。

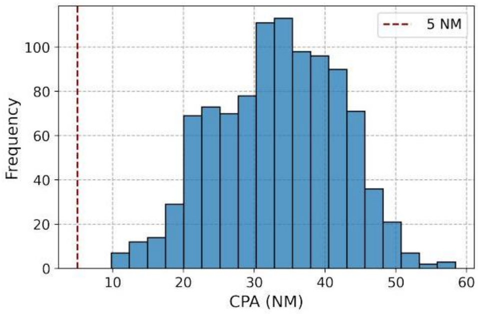

Fig. 9: CPA distribution between Flow 1 and Flow 2. The CPA values are always above 5NM which indicates 100% safe separation assurance.

图 9: 流 1 和流 2 之间的 CPA 分布。CPA 值总是高于 5NM ，这表明 100% 安全分离得到保证。

As discussed, the higher values of the efficiency metrics in some cases are due to the dynamic nature of the flows. Figure 10 shows the change in the flow size for 100 conflict scenarios with the evolution of each scenario (time steps) for Flow 1. The size is measured in terms of the maximum distance between an aircraft pair in the flow and the minimum distance between an aircraft pair in that flow.

如前所述，效率指标在某些情况下较高是由于流的动态性质。图 10 显示了流 1 在 100 个冲突场景发展过程中 (时间步) 流大小的变化。流的大小是通过测量流中飞机对之间的最大距离和最小距离来衡量的。

It can be seen that the maximum distance between aircraft (Figure 10 left), varies from approximately 12 NM to 39 NM depending on the number of aircraft in the flow. Further, this size also varies as the scenarios evolve, due to different speeds of the aircraft. With such dynamics associated with both flows, it is significantly complex to achieve better efficiency metrics in terms of lower CTD, CPA, and delay. In terms of the conflict scenarios and the current action space, the following have been observed. Since the acute conflict scenarios (45°) with larger flow sizes witness a loss of separation sooner, they require higher deviations to ensure safe separations. Thus, a larger look-ahead time would allow for smaller and relatively more efficient maneuvers. Furthermore, the current action space allows for a change of 30kts in one minute, which is very gradual. Higher magnitudes of speed change may also allow for conflict to be resolved by speed change only, leading to significantly lower CTD and improving other metrics as well.

可以看到，飞机间的最大距离 (图 10 左) 根据流量中飞机的数量，从大约 12 海里变化到 39 海里。此外，随着场景的发展，由于飞机速度的不同，这个距离也会有所变化。由于这些动态因素与两个流量相关，因此在降低 CTD、CPA 和延迟方面实现更高的效率指标是非常复杂的。在冲突场景和当前动作空间方面，有以下观察结果。由于较大流量的大角度冲突场景 (45°) 更快地失去间隔，它们需要更大的偏航来确保安全间隔。因此，更长的前瞻时间将允许进行更小且相对更有效的机动。此外，当前的行动空间允许在一分钟内改变 30kts ，这是非常渐进的。速度变化的高幅度也可能仅通过速度变化来解决问题，从而显著降低 CTD 并改善其他指标。

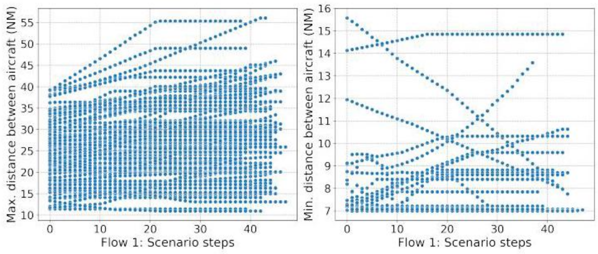

Fig. 10: Flow1 size and its variation with scenario evolution, for 100 conflict scenarios. Variation of the maximum distance (left) and the minimum distance between aircraft (right) with respect to the episode length.

图 10: 流量 1 的大小及其随场景演化的变化，针对 100 个冲突场景。最大距离 (左) 和飞机间最小距离 (右) 随剧集长度的变化。

VI. CONCLUSIONS

VI. 结论

The flow centric concepts of operations differ significantly from the current sector based operations, especially in maintaining safe separation between flights. The existing conflict resolution approaches (mathematical and learning based) are not suitable for applications in a flow-centric setting. Thus, in this work, we have proposed a novel approach for air traffic conflict resolution for FCOs which involves a policy-based learning for inter-flow conflicts and a self-stabilizing graph approach for intra-flow conflicts. The methodology ensures 100% safe separations despite the uncertainties present in the learning environment. The absolute values of delays for the flow are 2.53 and 9.49 minutes respectively, which are within acceptable limits, given the uncertainties and dynamics associated with the flows' size, speed and evolution over time. Results and discussion are also presented regarding the performance of the learned policy in terms of maneuver efficiency and potential improvements in the action space and scenario design to improve the model performance. Nonetheless, further experiments and analysis are still required to improve the model performance in terms of efficiency metrics.

流中心概念的操作与当前基于行业的操作存在显著差异，特别是在保持航班间安全间隔方面。现有的冲突解决方法 (基于数学和基于学习的方法) 不适合在流中心环境中应用。因此，在这项工作中，我们提出了一种针对流量中心操作 (FCO) 的空中交通冲突解决新方法，该方法包括基于策略的学习来解决流间冲突，以及用于流内冲突的自稳定图方法。该方法确保了 100% 在学习环境中的不确定性下安全间隔。该流量延迟的绝对值分别为 2.53 和 9.49 分钟，考虑到流量大小、速度和随时间演变的不确定性和动态性，这些延迟值在可接受范围内。还讨论了关于学习策略在机动效率方面的性能以及为了提高模型性能而在动作空间和场景设计上可能的改进。尽管如此，仍需进行更多实验和分析以提高模型在效率指标方面的性能。

Future work in this direction involves the development of algorithms to generate stable topological graph structures to represent the flows, increasing the traffic complexity and extensive simulations and analysis of the challenges such as splitting and merging of flow, that the FCOs bring to the air traffic management systems.

未来在这方面的工作包括开发算法以生成稳定的拓扑图结构来表示流量，增加交通复杂性，以及对流量分割和合并等挑战进行广泛的仿真和分析，这些挑战是流量中心操作给空中交通管理系统带来的。

ACKNOWLEDGMENT

致谢

This research is supported by the National Research Foundation, Singapore, and the Civil Aviation Authority of Singapore, under the Aviation Transformation Programme. Any opinions, findings, and conclusions or recommendations expressed are of the authors and do not reflect the views of the National Research Foundation, Singapore, and the Civil Aviation Authority of Singapore.

本研究得到了新加坡国家研究基金会和新加坡民航局的支持，属于航空转型计划的一部分。任何观点、发现、结论或建议均属于作者，并不反映新加坡国家研究基金会和新加坡民航局的观点。

REFERENCES

参考文献

[1] E. Zeki, P. L. de Matos, K. Purves, and M. Gibellini, "Business models for flight-centric air traffic control," Competition and Regulation in Network Industries, vol. 20, no. 4, pp. 319-332, 2019.

[2] D. Gianazza, "Learning air traffic controller workload from past sector operations," in ATM Seminar, 12th USA/Europe Air Traffic Management R&D Seminar, 2017.

[3] SESAR, ”A proposal for the future architecture of the european airspace,” The Single European Sky ATM Research (SESAR), 2019.

[4] N. Durand, J.-M. Alliot, and O. Chansou, ”Optimal resolution of en route conflicts,” in Procceedings of the Irst USA/Europe Seminar, vol. 1, no. 5, 1997, p. 2.

[5] Y.-J. Chiang, J. T. Klosowski, C. Lee, and J. S. Mitchell, ”Geometric algorithms for conflict detection/resolution in air traffic management,” in Proceedings of the 36th IEEE Conference on Decision and Control, vol. 2. IEEE, 1997, pp. 1835-1840.

[6] N. Durand and J.-M. Alliot, ”Ant colony optimization for air traffic conflict resolution,” in Atm seminar 2009, 8th usa/europe air traffic management research and developpment seminar, 2009.

[7] J. K. Kuchar and L. C. Yang, ”A review of conflict detection and resolution modeling methods,” IEEE Transactions on intelligent transportation systems, vol. 1, no. 4, pp. 179-189, 2000.

[8] Z. Liu, K. Cai, X. Zhu, and Y. Tang, ”Large scale aircraft conflict resolution based on location network,” in 2017 IEEE/AIAA 36th Digital Avionics Systems Conference (DASC). IEEE, 2017, pp. 1-8.

[9] C. Allignol, N. Barnier, N. Durand, A. Gondran, and R. Wang, ”Large scale 3d en-route conflict resolution,” in ATM Seminar, 12th USA/Europe Air Traffic Management R&D Seminar, 2017.

[10] T. Stollenwerk, B. O’Gorman, D. Venturelli, S. Mandra, O. Rodi-onova, H. Ng, B. Sridhar, E. G. Rieffel, and R. Biswas, ”Quantum annealing applied to de-conflicting optimal trajectories for air traffic management,” IEEE transactions on intelligent transportation systems, vol. 21, no. 1, pp. 285-297, 2019.

[11] M. Brittain and P. Wei, ”Autonomous separation assurance in an high-density en route sector: A deep multi-agent reinforcement learning approach,” in 2019 IEEE Intelligent Transportation Systems Conference (ITSC). IEEE, 2019, pp. 3256-3262.

[12] J. Mollinga and H. van Hoof, ”An autonomous free airspace en-route controller using deep reinforcement learning techniques,” 9th International Conference on Research in Air Transportation, ICRAT, 2020.

[13] D.-T. Pham, P. N. Tran, S. Alam, V. Duong, and D. Delahaye, ”Deep reinforcement learning based path stretch vector resolution in dense traffic with uncertainties,” Transportation research part C : emerging technologies, vol. 135, p. 103463, 2022.

[14] I. Doc, ”4444-procedures for air navigation services-air traffic management,” Montreal, QC, Canada: The International Civil Aviation Organization (ICAO), 2016.

[15] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, ”Proximal policy optimization algorithms,” arXiv preprint arXiv:1707.06347, 2017.

[16] E. W. Dijkstra, ”Self-stabilizing systems in spite of distributed control,” Communications of the ACM, vol. 17, no. 11, pp. 643-644, 1974.

[17] C. Boulinier, F. Petit, and V. Villain, ”When graph theory helps self-stabilization,” in Proceedings of the twenty-third annual ACM symposium on Principles of distributed computing, 2004, pp. 150- 159.

[18] N.-S. Chen, H.-P. Yu, and S.-T. Huang, ”A self-stabilizing algorithm for constructing spanning trees,” Information Processing Letters, vol. 39, no. 3, pp. 147-151, 1991.

[19] Y. Sudo, F. Ooshita, and S. Kamei, ”Self-stabilizing graph exploration by a single agent,” arXiv preprint arXiv:2010.08929, 2020.

[20] A. Raffin, A. Hill, A. Gleave, A. Kanervisto, M. Ernestus, and N. Dormann, ”Stable-baselines3: Reliable reinforcement learning implementations,” Journal of Machine Learning Research, vol. 22, no. 268, pp. 1-8, 2021. [Online]. Available: http://jmlr.org/papers/v22/20- 1364.html