面向战术空中交通管制的未来能力智能体

Paul Kent、George De Ath、Martin Layton、
Allen Hart*埃克塞特大学，埃克塞特，英国

Richard Everson艾伦·图灵研究所，伦敦，英国 & 埃克塞特大学，英国

Ben Carvell†
NATS，怀特利，英国

arXiv:2601.04285v1 [cs.AI] 7 Jan 2026

空中交通需求不断增长，推动了自动化技术的应用，以支持空中交通管制员，但现有方法在安全保证和可解释性之间面临取舍。强化学习等基于优化的方法性能出色，却难以验证和解释；基于规则的系统虽然透明，却很少检验不确定条件下的安全性。本文介绍 Agent Mallard，这是一种用于系统化空域战术管制的前向规划型基于规则的智能体，它将随机数字孪生直接嵌入冲突解决循环。Mallard 在预先设定的 GPS 引导航路上运行，将连续的四维引导简化为对航路道和高度层的离散选择，并利用专家参与制定的冲突消解策略库构建分层计划。深度受限回溯搜索运用因果归因、拓扑计划拼接和单调轴约束，为所有航空器寻找完整的安全计划；在确定采用每个候选机动之前，先根据不确定执行场景（例如风况变化、飞行员响应、通信中断）对其进行验证。

与英国管制员进行的初步演练，以及在 BluebirdDT 空域数字孪生中的初步测试表明，Mallard 的行为符合专家推理，并能在简化场景中解决冲突。该架构旨在未来的结构化航路环境中，融合基于模型的安全评估、可解释的决策逻辑和可处理的计算性能。

I. 引言

全球空中交通量不断增长，亟需加以管理 [1, 2] ，这正推动空中航行服务提供者日益采用自动化工具，并探索人工智能（AI）在提升安全、效率和可扩展性方面的潜力 [3–8]。这些努力旨在支持空中交通管制员应对日益复杂的运行环境。

近期研究探索了先进的人工智能技术，尤其是多智能体强化学习，以应对空中交通管制的复杂性 [9–11]。然而，基于深度学习的方法常因其不透明的黑箱特性而受到批评，这妨碍了人们对其建立信任，也阻碍了其在安全关键环境中的实际部署 [7, 12]。此外，这些模型往往依赖理想化假设（例如，信息完备、机动能力不受限制），导致理论性能与运行现实之间存在显著差距 [13, 14]。弥合这一差距，需要对运行复杂性进行有原则的简化，使算法开发与现实的运行演变相一致。

为弥合复杂性与透明性之间的差距，本文介绍 Agent Mallard——一种基于规则、采用前向规划的 AI 智能体，专为系统化空域这一理念而设计 [15]。Mallard 旨在利用预定义 GPS 引导航路所具备的固有结构与可预测性，简化战术决策和冲突解决。这种结构化方法旨在提升透明性和可验证性——这对赢得人类信任至关重要，同时也为未来运行空域配置中的高级决策支持能力提供发展路径。

*研究员（肯特、莱顿、哈特）、讲师（德·阿斯）和教授（埃弗森），埃克塞特大学。†NATS 首席研究员。

A. 贡献

本文为自主空中交通管制研究作出以下贡献：**1) 仿真集成架构：** 我们提出一种智能体架构，在执行任何候选控制动作之前，先在不确定条件下通过前向仿真对其进行评估。该设计旨在提供基于模型的安全评估（包括对速度不确定性等随机因素的评估）和可解释性，而这些特性在现有方法中往往难以兼得。**2) 协同冲突解决算法：** 我们提出一种全空域解决算法，同时为一个扇区内的所有航空器寻找无冲突轨迹，而不是孤立地逐一解决成对冲突。该算法采用深度受限的回溯搜索，并结合专门的剪枝机制，高效管理级联交互。**3) 结构化空域设计原则：** 我们展示了预定义的 GPS制导航路结构如何将冲突消解问题从连续轨迹优化转化为离散选择，从而支持可行的实时性能，并通过数字孪生仿真实现明确的安全评估。

第五节介绍专家管制员评审和初步数字孪生测试的初步验证结果，并指出通过智能体验证框架进行全面评估仍是后续工作。

本文其余部分安排如下：第二节回顾空中交通管制运行以及自主冲突解决的相关方法。第三节介绍 Agent Mallard 架构，包括分层规划、冲突解决算法和可解释性。第四节描述所提出的验证方法，第五节则报告专家评审和初步测试的结果。第六节讨论设计假设和已知局限性，第七节指出未来发展的优先事项。

II. 背景与相关工作

A. 空中交通管制与自动化的发展

航路空中交通管制负责管理航空器在多个扇区内的爬升、巡航和下降。管制员负责保持间隔、安排交通流量顺序，并协调其负责空域扇区的移交工作。这带来了很高的认知负荷 [16–18]。随着交通密度增加，自动化监控可通过保持持续监视来增强管制员的能力，并将人类的专业技能留用于战略协调。

B. 系统化空域：未来空中交通管制的基础

实现此类自动化的一项关键要素，是空域结构本身的现代化。系统化空域标志着空中交通管理的一次重大转变：以预定义的 GPS 引导航路取代战术引导，这些航路被称为“空中高速公路” [15]。在区域导航（RNAV）等基于性能的导航（PBN）标准1 [19], 的支持下，这种结构使航空器能够沿端到端轨迹飞行，并最大限度地减少管制员干预 [20]。

这一全球性转变由多项重大现代化举措推动。在欧洲，SESAR 推动系统化的基于轨迹的运行，包括点汇聚和大规模部署基于性能导航航路 [21, 22]。在美国，美国联邦航空管理局的 NextGen计划通过其 Metroplex计划，推出了基于性能的标准仪表离场（SIDs）和进场程序（STARs） [20]。

使用人工智能智能体执行战术空中交通管制颇具挑战，部分原因在于航空器在任何时刻都可能采取大量动作。例如，航向通常按 5° 增量离散化，因此航空器可能被引导至 360/5 = 70个方向中的任意一个，也可能爬升或下降至 20,000 英尺至 40,000 英尺之间的 20 个飞行高度层中的任意一个。当涉及两架航空器时，联合动作空间的维度可能超过 100。结构化空域为人工智能提供了独特机遇。通过预定义几何结构降低复杂度，它能够实现更简单、透明度更高的算法，使其更符合人工监督和运行实践，也更适用于人机协作的安全关键环境。

基于性能的导航（PBN）大幅提升了水平导航可靠性，并持续修正风和漂移，使航空器紧贴预定地面航迹飞行。这将横向不确定性限制在有界的偏航迹误差范围内，并使大部分变化转移到沿航迹时间上。系统化空域旨在利用这一特性：通过布设几何间隔固定的航路，可以依靠几何形态而非持续的战术引导来保持横向间隔。因此，在结构适当的扇区中，冲突消解问题可以从完整的三维引导简化为管理垂直转换和纵向间隔。

为在实践中实现这些几何保障，系统化扇区必须提供一种航道结构，其间隔

沿整条航线保持不变。图2展示了这一方法在 BluebirdDT 数字孪生中的实现方式，其中定位点间航线配有相邻的冲突消解航道。该航道结构以几何方式构建：沿航线连接连续定位点的中心线两侧各 3.5 海里处设置平行线。在航线转弯处，这些偏置线相交，形成过渡节点。这种构造使 7 海里的间隔（中心线两侧各 3.5 海里）在整条航线中保持不变，包括转弯路段，从而确保无论航空器处于航线上的何处，间隔标准都得到保持。

C. 自主冲突解决方法

1. 优化与几何方法

早期的自主冲突解决方法将该问题定义为全局优化问题，利用混合整数规划或进化算法（例如遗传算法）搜索无冲突的四维轨迹[23, 24]。尽管这些方法在理论上能够找到最优解，但其计算量很大，往往难以满足高密度扇区中的实时约束。此外，虽然扇区边界等硬约束可以用数学方式表示，但要在代价函数中编码空中交通管制较为柔性的风格偏好，仍然极具挑战。管制员期待标准流量、整齐的航向以及看起来自然的机动等模式。优化方法经常会生成满足数学约束、但在人类操作员看来显得杂乱或机械的轨迹，从而降低信任，也使得使用标准无线电通话术语表达预期机动更加困难。

2. 强化学习

近期一些研究工作聚焦于强化学习（RL），让智能体在仿真环境中通过试错学习冲突消解策略 [10, 25]。尽管强化学习和多智能体强化学习（MARL）展现出了令人印象深刻的可扩展性，但其在安全关键部署方面带来了重大挑战。所学习的策略通常是不透明的“黑箱”，阻碍了验证和信任的建立 [26]。此外，连续动作强化学习智能体往往会表现出高频控制抖动（对航向或速度进行微调），这会造成过大的飞行员工作负荷，并违背空中交通管制无线电通信的离散特性。Carvell等人的一种近期技术——在线动作堆叠方法 [27], ——通过将微动作聚合为单条指令来缓解这一问题。然而，多智能体强化学习智能体的训练通常需要对环境进行非运行性简化，以促进收敛。因此，这些模型的理论性能与实际部署所需的安全保证之间仍存在显著差距。

3. 基于规则和启发式的智能体

基于规则和启发式的智能体通过遵循预定义逻辑来实施空中交通管制，通常采用由专家设计的如果—那么规则。此类早期系统试图直接编码管制员知识，后续发展则引入了更复杂的认知模型。一个值得关注的示例是美国国家航空航天局的基于CATS的智能体 [28],，它将管制员行为构建为由意图驱动任务组成的层级结构（例如监视、冲突检测和机动执行），以模拟人类决策。

美国国家航空航天局后续的研究将这些智能体扩展到 TRAC 等蒙特卡洛仿真框架中，从而能够在不同的交通流量和天气条件下大规模测试程序稳健性 [29]。然而，在这些系统中，仿真纯粹被用作离线分析工具，而非智能体实时决策逻辑的核心部分。

近年来，Bluebird 专家规则型智能体 [26] 等系统采用一组固定的反应式启发规则并结合短时域仿真，研究表明，简单的反应式规则也能以较低的计算成本取得令人鼓舞的结果。这些智能体易于理解，且与管制员的工作流程十分相似。然而，它们通常依赖程序有效性这一假设：在已知几何构型中应用标准机动将产生安全结果。若规划循环中未整合扩展时域验证，这一假设在复杂场景中可能不再成立：解决局部冲突可能会无意中在后续过程中引发新的、更严重的交互 [30]。

4. 差距：仿真集成决策

空中交通管制（ATC）智能体设计中仍存在一个关键缺口：系统必须在不确定性下验证安全性，同时对人类操作员保持可解释性。现有方法往往需要权衡取舍——基于优化的智能体（例如强化学习或进化规划器）具有鲁棒性，但缺乏透明性；基于规则的系统虽然具备可解释性，却很少通过仿真验证

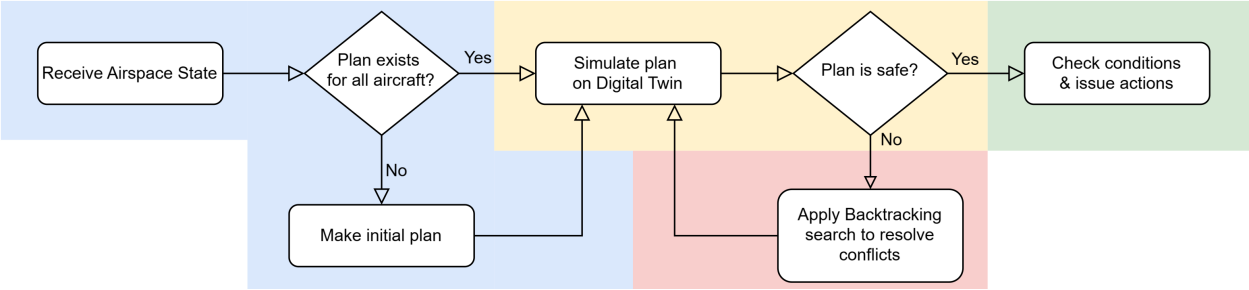

图1 Agent Mallard 的运行流程图。蓝色：初始化（接收空域状态，并在需要时生成计划）。黄色：评估（在数字孪生中仿真计划）。红色：冲突消解（若不安全，则通过回溯搜索寻找安全替代方案；参见第 III.E.4 节）。绿色：执行（若安全，则检查条件并发布动作）。若条件始终未满足，航空器将错过其协调时机，被判定为不安全，并重新规划。

结果；而 CATS/TRAC 等架构则将仿真视为一种离线工具，与实时决策脱节。Agent Mallard 旨在通过将随机前向仿真直接嵌入其机动选择循环，帮助弥合这一缺口。与 CATS 一样，它在分层决策过程中发出结构化的高层基于意图的动作，从而保持与控制器工作流程的兼容性。但不同于早期智能体，Mallard 旨在利用其数字孪生，在发出每个候选机动之前对其进行仿真——在一系列不确定执行场景中检查次生冲突和约束违反，包括风况变化和飞行员行为。

重要的是，Mallard 旨在利用系统化空域，避免承担完整四维优化所需的开销。预定义横向航路（例如 PBN航道）可缩小解空间，使智能体能够专注于垂直和时间调整。总体设计旨在结合以往研究中往往相互取舍的三项特性：可解释性、通过随机仿真评估的鲁棒性，以及在结构化航路环境中实现实时性能的潜力。

III. Agent Mallard：所提方法

Agent Mallard 是一种前向规划智能体，旨在系统化空域中提供战术空中交通管制。该智能体旨在持续监控航空器，通过前向仿真预测冲突，并生成一组管制许可，同时解决所有预测到的安全违规。

Mallard 的设计基于一个简单原则：先模拟，再执行。每个候选解在获准采用前，都会先在空域的高保真数字孪生中进行测试。模拟能够考虑现实世界中的不确定因素：风、飞行员响应时间，以及至关重要的通信中断。只有在应对所有这些因素时都能保持安全间隔的解，才会被认定为有效。

Mallard 以与人工管制员相同的方式发出指令：针对爬升、下降和速度调整发布标准许可。在系统化空域中，横向管制得到简化：智能体不再发出任意航向指令，而是引导航空器沿特定的预定义航道飞行，即构成空域结构的 GPS 引导航路。这种基于航道的方法将横向管制问题从连续引导简化为离散航道选择，从而能够通过模拟验证安全性，并在设计上保持几何间隔。它还能减轻管制员的认知负荷，因为航空器始终沿着可预测的路径飞行，所需监视更少，同时仍能确保安全间隔。

通过模拟发现冲突后，Mallard 会运用其冲突消解逻辑来识别冲突类型，然后提出策略。Mallard 会从一系列冲突消解策略中进行选择，尝试找出冲突原因并提出解决办法。由于每项决策都可以追溯到应用于特定冲突几何关系的、可识别的战术模式，因此该智能体的推理过程始终透明：人类操作员不仅能理解提出了什么管制许可，还能明白为何选择了这种特定方案。

A. Mallard 如何运作：完整周期

图 1 i　展示了 Mallard 的运行周期；在整个运行过程中，该周期每隔几秒便会重复一次　　：

基准规划（蓝色）：智能体接收当前空域状态——扇区内所有航空器的位置、速度和飞行计划，以及预计在 15 分钟内进场的航空器信息。对于每架航空器，Mallard 都会制定一份基准计划，代表效率最高的路径：爬升至首选高度，沿指定的 GPS 引导航路飞行，并在最佳位置下降，以满足协调离场条件。这些计划假定不存在冲突，并将运行效率最大化。

安全验证（黄色）： Mallard 通过向前模拟接下来一小时的运行情况来测试这些计划。数字孪生会在多种情景下推演所有航空器的轨迹：正常条件、现实的风况变化、飞行员响应延迟，以及 15 分钟的通信中断情景；在此情景下，航空器会继续执行最后收到的管制许可，不再接收任何后续指令。该验证既能揭示当前冲突，也能发现可能在未来几分钟内出现的后续交互。

Mallard 依次解决冲突，将每次间隔丧失视为涉及两架航空器的问题。对于每次冲突，智能体都会向两架航空器分配互补管制许可（例如，两架航空器都向左偏置至平行航道，或一架爬升、另一架下降）。随后，系统会将每种候选策略纳入相应的飞行计划，并重新模拟整个空域，以评估后续影响并确认不会引入新冲突：例如，刚刚偏置的航空器可能与相邻航线上第三架航空器发生冲突，或者时间调整可能改变后续交互。如果出现新冲突，系统会依照同一流程逐一处理：找出相关的计划航段、评估策略并重新模拟。搜索会递归进行，并可能逐步尝试更深层次的策略组合，直到找到完整的解——一份协调的空域计划，使所有航空器能够同时保持安全间隔。在整个过程中，只会修改导致冲突的特定计划航段；每架航空器其余部分高效的基准轨迹则保持不变。

管制许可发布（绿色）：一旦获得经过验证的安全解，Mallard 就会监控空域，并在适当时机发布管制许可。管制许可与几何条件绑定，而非固定时间，例如“横向间隔超过 5 海里时左转”，即使时机发生变化，也能确保安全。这种基于状态的方法意味着，管制许可逻辑会根据实际轨迹而非预测的计划自动调整。

持续适应　　：这一循环：计划、验证、求解、发布每隔几秒重复一次。随着航空器移动，新的飞行进入，且条　　件发生变化，Mallard 会通过前向仿真重新评估，并据此进行调整。

B. 关键组件

智能体 Mallard 的方法整合了五个关键组成部分：

- **系统化空域结构：** Mallard 利用预先定义的 GPS 制导航路结构，构建“空中高速公路”（见图2）。通过限定航空器沿具有已知几何间隔的离散横向航路飞行，基于航路道的结构将冲突消解问题从连续的四维轨迹优化转化为对航路道、高度层和相对速度的离散选择。
- **不确定性轨迹仿真：** 空域的高保真数字孪生可仿真航空器在拟议控制动作下的轨迹演变，从而在存在环境不确定性的情况下，对安全性和运行可接受性进行前瞻性评估。
- **基于状态的条件策略：** 控制干预被编码为动态策略（详见第 III.E.4.II 节），并由局部空域几何形态（例如间隔阈值或相遇类型）触发。这种基于状态的表述将安全性与精确时机解耦，确保即使风等随机因素导致沿航迹变化，飞行意图仍然有效。
- **专家知情决策：** Mallard 采用以专家空中交通管制员的推理为基础的决策逻辑（表 2）。通过将解空间限制在一组按优先级排序的标准冲突消解策略中，迭代式回溯搜索能够生成本质上易于理解的计划，使人类操作员不仅能够核验结果，还能核验每次干预背后的策略依据。
- **因果可追溯性：** 这种层级结构为每次管制干预都建立了明确的审计追踪。由于每个动作都与特定的触发条件和上级机动密不可分地关联，系统得以保留其决策的逻辑溯源信息。这使人类操作员不仅能够审查发出了什么指令，还能准确了解为何以及何时系统判定有必要发出该指令，从而使决策过程透明且可核验（第三节.E.4.I，图 6）。

以下各节将详细阐述这些组件的架构和实现细节，并讨论如何评估此类智能体的性能。

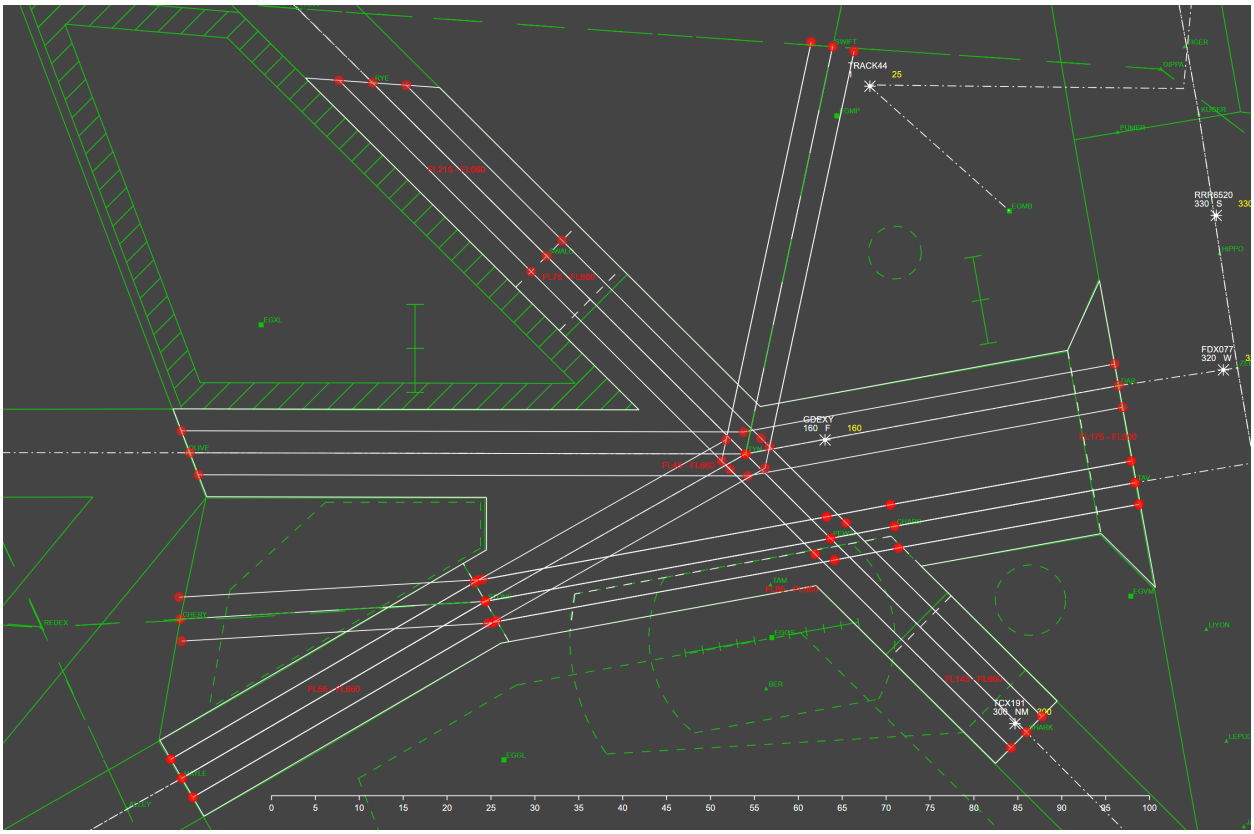

图 2 训练扇区中一组已解除冲突的航道。每条航线的右侧和左侧航道在几何上彼此分离，不会发生冲突。

C. 实时监控与轨迹预测

数字孪生平台纳果英国 Project BluebirdDT [31, 32] 的一部分。BluebirdDT 提供真实世界的运行数据，用于基于实际数据的验证 [33]，并纳入空中交通管制员培训的相关要素 [34]，使智能体性能能够依据源自人工空中交通管制员评估法律要求的指标进行评估 [34]。随着设计逐步成熟，这种集成旨在帮助 Mallard 不仅符合技术标准，也能在结构化环境中满足运行预期。

测试期间，Mallard 会对空域进行前向模拟，评估名义轨迹和随机反事实。模拟涵盖扇区内的所有航空器，以及预计在 15 分钟内进入扇区的航空器，从而能够主动探测未来可能发生的交互。最长一小时的预测时域有助于确保可追溯性；同时，假设不再发出任何指令，每个反事实也会向前推演 15 分钟，在通信中断时提供保守的通信中断安全缓冲。

这些时间范围经过选择，以匹配航路运行中典型的扇区前视时间和通信中断程序。通过对名义轨迹施加采样变化来表示执行过程中的不确定性，变化涉及航空器性能和环境条件等关键因素；BluebirdDT [31, 35, 36] 中的概率模型为此提供了支持。因此，每次展开都对应于在不同执行条件下对同一控制计划的一种合理实现。

以下各节将详细介绍该架构的各个组成部分。

D. 分层计划表示

Mallard 将每架航空器的意图表示为结构化的层级结构，而非固定的坐标序列。计划不仅编码航空器应处于何处，还编码它应采取哪些动作，以及这些动作应何时因空域状态的演变而触发。

这种表示有两个关键用途。首先，它能够支持在多个抽象层级上进行推理：智能体 nt

表 1 控制结构层级。该架构将高层意图分解为原子基元。请注意，计划动作 **是一个容器，用于将特定逻辑门（条件）绑定到特定控制输出（动作）。**

层	控制对象	功能	示例
原子	条件 & 动作	构成计划的不可再分逻辑门和控制动词。	C =横向间隔 > 5海里 A =恢复导航(定位点)
战术	计划动作	一个绑定了触发条件、动作和完成条件的容器。	$\mathrm{P} = C_{\text{trigger}}, A, C_{\text{completion}}$
战略	机动	从中选出的多步骤解决方案 专家库中选出的多步骤解。 一系列计划动作，通常是完整的——一个计划动作的条件是另一个动作的触发条件	**策略B**: $(\mathrm{P}_n \rightarrow \mathrm{P}_{n+1})$
全局	飞行计划	每架航空器的端到端计划，由一系列相互关联的机动动作组成。	$FP = (\mathrm{P}_1 \rightarrow \mathrm{P}_2 \ldots \rightarrow \mathrm{P}_N)$
全系统	空域计划	所有无冲突的 航空器飞行计划所组成的协调集合，在扇区内同时生效。	$S = \{FP_1, FP_2, \ldots, FP_M\}$

可以调整高层机动，而无需重构其组成部分的管制许可；也可以修改单个计划动作，而不影响飞行计划的其余部分。其次，它确保计划在存在执行不确定性的情况下仍然有效，无需持续重新规划。

Mallard 不会在固定时间指定动作，例如“14:23 转向至航向 270”，而是将动作与几何条件或基于状态的条件关联起来，例如“当横向间隔超过 5 海里时，转向至航向 270”。条件根据实际空域状态，而非预测的时间表，决定动作应在何时发生。这一设计具有一项关键特性：如果风、性能变化或飞行员延误改变了航空器的时间安排，计划在逻辑上仍然正确。动作会在其条件变为真时触发，而不是在可能已不再合适的预定时间触发。轨迹随时间以不同方式展开，但其底层意图依然不变。在 Mallard 持续重新仿真扇区的过程中，同一计划会在新条件下重新验证，而非重新构建。

该层级结构将航空器的完整计划逐步分解为更简单的组成部分（表 1）。在扇区层面，空域计划协调所有航空器。每架航空器都有一个飞行计划，即从进入到离开的一系列机动动作。机动动作代表多阶段的战术解（例如“向左偏置、保持间隔、恢复航线”），并可分解为计划动作——由闭环触发、动作和完成环节构成的单元。最基础的是原子条件和动作，即构成所有决策逻辑基本构件的布尔谓词和控制指令。以下各节将介绍这些基本要素，并展示它们如何组合成易于理解且稳健的计划。

基础原语：条件与动作

Mallard 的控制逻辑由两个基本要素构成：条件和动作。当且仅当满足相应条件时，才会执行动作。条件（C）由描述空域状态的布尔谓词构成，其中包括时间检查以及诸如横向通过航空器之类的相对几何触发条件。这种几何表述将安全与时间不确定性解耦，确保即使随机因素引入沿航迹变化，意图仍然有效。动作 A 是以标准用语表达的离散空中交通管制指令，例如爬升至飞行高度层 *330* 或保持航向 *270*。

独立控制轴

在 Mallard 的控制模型中，横向和垂直维度被视为独立控制轴。每个轴都有各自的一组条件和动作，使智能体能够分别推理水平几何结构（例如航道选择、偏置）和垂直意图（例如爬升、下降）。机动动作可以结合两个轴上的动作，但底层表示仍保留这种区分，从而简化因果归因、支持局部计划修改，并实现单调轴约束等机制。通过隔离对冲突有影响的轴，智能体可以只调整计划中相关的维度，同时保持另一维度不变，从而减少非预期交互，并使整体计划易于理解。

计划动作

一个计划动作将这些基本要素绑定为一个闭环单元：

$$\mathcal{P} = \langle C_{\text{trigger}}, A, C_{\text{completion}} \rangle.$$

智能体仅在满足 C_{trigger} 时才发出 A，并监控 $C_{\text{completion}}$ 以验证执行情况，从而能够检测偏差，例如飞行员响应延迟或未能达到指令剖面。

机动合成

计划动作组合成机动动作——用于实施高层策略（例如横向或垂直间隔）的结构化参数序列。这与专家空中交通管制员的分层任务分解相似 [37], ，使智能体能够从战术意图层面进行推理，而非逐项考虑控制输入。

例如，一个横向偏置机动最多包含三个阶段：平行阶段（指挥两架航空器进入平行偏置航道）、保持（保持偏置，直至间隔超过阈值），以及可选的恢复（重新并入标称航路）。平行阶段根据当前空域状态立即触发，后续阶段则使用几何触发条件（例如“横向通过航空器”），而非固定时序。此机动通过参数进行设定，在本例中，参数包括为每架航空器分配的航道（例如两架均向左、均向右或向相反方向偏置）以及恢复所需的间隔阈值；这些参数由冲突几何关系和可用空域决定。

飞行计划：从入口到出口的机动排序

计划动作组合成机动动作，而机动动作又组合成飞行计划——即从扇区入口到出口的完整单架航空器控制序列。飞行计划表示如下：

$$FP = (\mathcal{P}_1 \rightarrow \mathcal{P}_2 \rightarrow \ldots \rightarrow \mathcal{P}_N)$$

其中，每个 P_i 都是一个计划动作，一个动作的完成条件作为下一动作的触发条件，从而形成逻辑门控许可指令的时间有序链。

智能体的规划过程始于为每架受控航空器构建标称飞行计划。该基准计划生成从扇区入口到预先协调的出口条件之间的运行效率最优路径。利用系统化空域结构，标称计划指示航空器截获其指定的基于性能导航航路，爬升至（或保持）其首选飞行高度层（PFL）以尽量减少燃油消耗，并执行连续下降运行（CDO），该运行经过计算，可尽可能晚地满足出口定位点约束（一个经过协调的飞行高度层）。该标称计划充当参考意图——即在不存在冲突的情况下航空器将遵循的轨迹。

空域计划：全扇区策略

在最高层级上，Mallard 维护一份空域计划：该计划是扇区内所有同时处于活动状态的飞行计划的完整集合：

$$S = \{FP_1, FP_2, \ldots, FP_M\}$$

其中，M 表示当前处于管制之下或预计将在未来 15 分钟内进入的航空器数量。空域计划代表整个扇区的协调状态：无冲突的解要求所有 M 份飞行计划在一同向前模拟时，在所有不确定性场景下都保持安全间隔。

当向前仿真检测到两架航空器之间存在冲突时，Mallard 不会丢弃它们的飞行计划并从头开始重新优化。相反，Mallard 会使用一种称为计划拼接的局部修复机制（详见第 III.E.4.II 节），仅在受影响飞行计划的冲突航段中插入修正机动动作。由于每个

计划表示为有序的计划动作序列，而非固定坐标列表，因此智能体可以在该序列中加入解冲突机动（例如横向偏置），同时保留上游和下游逻辑。例如，为解决巡航期间的冲突，智能体会识别出导致冲突的具体计划动作（例如“在中心线航路上保持 PFL”），并拼接入一个偏置机动，使航空器暂时横向偏移，同时保持原计划的其他所有方面不变。

图 3 展示了 BluebirdDT 模拟器中的雷达显示画面，其中多架航空器根据初始计划沿系统化航路飞行。这些基准规则引导每架航空器沿预定路径飞行；如果预测到冲突，Mallard 的冲突消解逻辑（第三节 E）会相应调整计划。

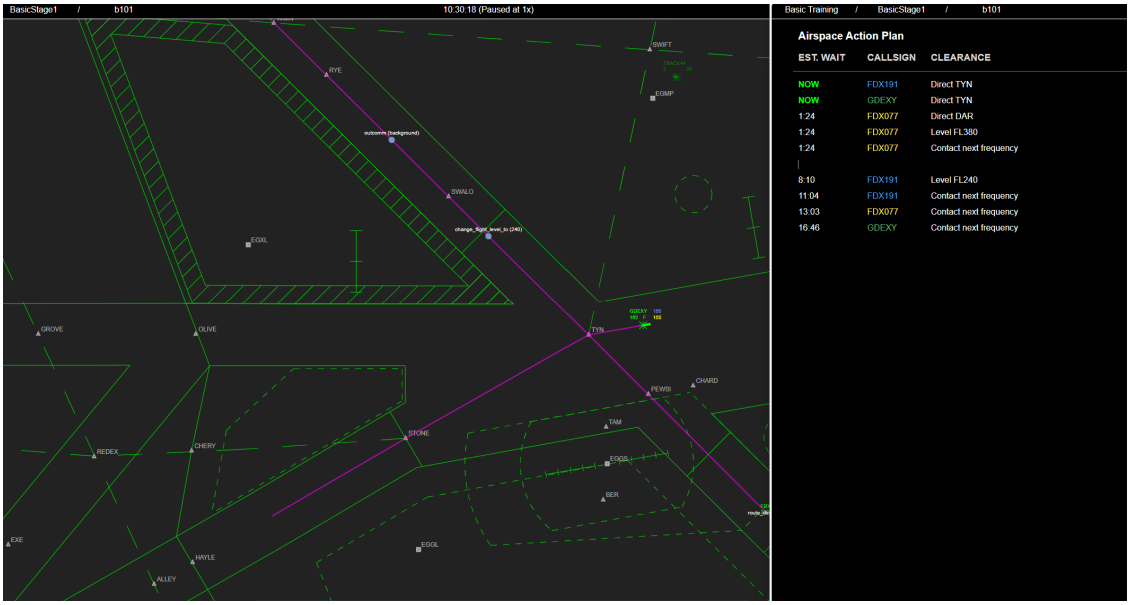

图 3 Agent Mallard 在 BluebirdDT 数字孪生界面中的初始飞行计划示例可视化，展示了基于系统化空域原则的预定航路。

E. 冲突解决

Mallard 持续向前模拟所有航空器计划，并将任何违反最小间隔的轨迹对，或在失去与扇区内所有航空器的无线电通信时会导致间隔违规的轨迹对，标记为成对冲突。我们将通信中断后产生的轨迹称为“反事实轨迹”，因为它们代表了另一种未来，即通信中断时会 发生什么。在这种情况下，航空器会继续沿其航线飞行，并保持其所在航道的偏置。

1. 顺序解决逻辑

在模拟过程中，Mallard 会为所有航空器编制一份技术安全记录 （TSR）。冲突按顺序解决：首先处理预测时间最早的冲突，随后重新模拟空域，以更新后续交互。重新模拟可避免求解器处理已不复存在的冲突，或忽略干预引入的新冲突。

冲突按航空器对进行考量，并根据两架航空器在名义轨迹或反事实轨迹中的预测最近接近点（CPA）处的相对垂直状态、航向和速度进行分类。类似地，解也针对航空器对，并提供相互补充的动作来解决冲突。

2. 冲突分类

Agent Mallard 通过整合以下三个维度对冲突进行分类：相对垂直状态（4 种组合）、相对航向（3 种类型）和相对速度（3 种状态），见图 4。由此形成一套全面的冲突类别分类体系（例如，平行 / 同高度 / *AC1* 更快 表示超越或追赶）。对于每一种具体类别，智能体

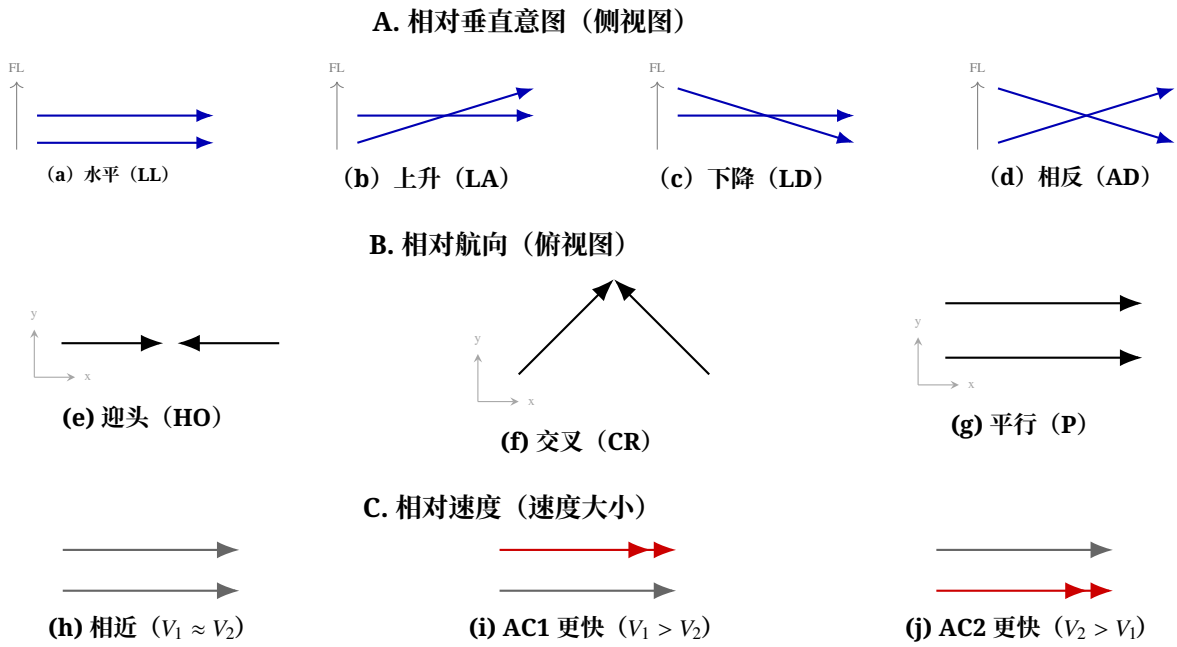

图4 全面冲突分类体系。智能体从三个维度对交互进行分类：（A）垂直状态（蓝色／侧视图）、（B）横向几何结构（黑色／平面图）和（C）速度差（红色／运动视图）。这种多维分类能够精确选择冲突消解策略。

会调用一份预先排序的冲突消解机动列表——例如，在超越场景中优先采用速度控制或临时航道偏置，而在迎头冲突中优先采用垂直间隔或横向变道。

3. 基于车道的冲突消解

系统化空域以预先定义的基于性能的导航（PBN）引导航道为特征（例如，RNAV1 可确保航空器始终处于指定航线 1 海里范围内），从而实现结构化且可预测的冲突消解。图5展示了其一项关键优势：在迎头冲突中（图5a），通过“双航道变换”（图5c），可安全地将两架航空器转移至间隔已知的平行航道。若航道间距适当（例如中心线之间相距 7 海里），RNAV1 可确保航空器在整个过程中至少保持 5 海里的间隔。

基于航道的冲突消解相较于战术引导具有显著的安全优势，因为航空器始终接受持续的基于性能的导航引导。战术引导（图5b）会产生开放式航向；一旦发生通信中断，航空器可能因此进入非受控航迹，甚至飞出扇区边界或陷入次生冲突。相比之下，航道变更可使航空器始终沿预定义航路飞行，并具有已知的几何形态和确定的端点。如果通信中断，航空器只需继续沿分配的PBN航道飞行至下一个定位点，始终处于管制空域内，并通过设计保持间隔。这种结构化行为能够更加高效地利用横向空间，并大幅简化冲突消解问题。

4. 通过回溯搜索实现全空域冲突消解

尽管每次单独的冲突消解都针对特定的航空器对安全问题，Mallard仍旨在为管制扇区内的所有航空器制定安全计划。这一要求带来了显著的算法复杂度，因为解决一个冲突可能会在空域其他位置引发、消除或改变其他冲突。因此，智能体必须考虑相互依赖的冲突链，并协调多个航空器对之间的解决方案。

基础机制 在介绍完整的解脱算法之前，我们先介绍三种能够有效解决冲突的关键机制：

I. **因果归因（搜索空间剪枝）：**

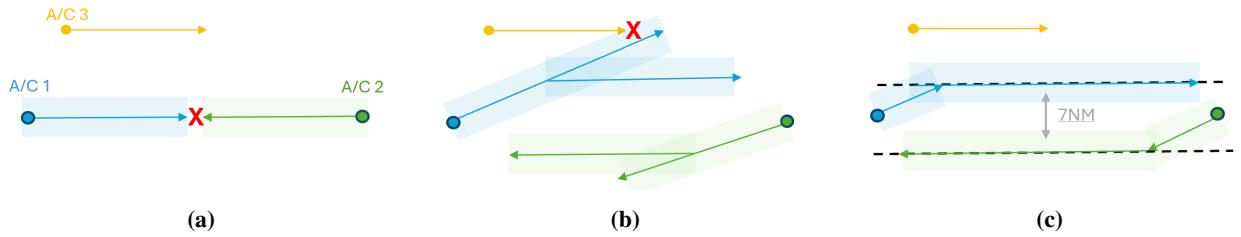

图 5 (a) 蓝色和绿色航空器之间的预测冲突（红色“X”）。(b) 战术引导会引入非标准航迹和次生冲突（例如与黄色飞机的冲突）。(c)“双航道变换”将两架航空器移至预先定义的平行航道，从而保持可预测的间隔。讨论请参见正文。

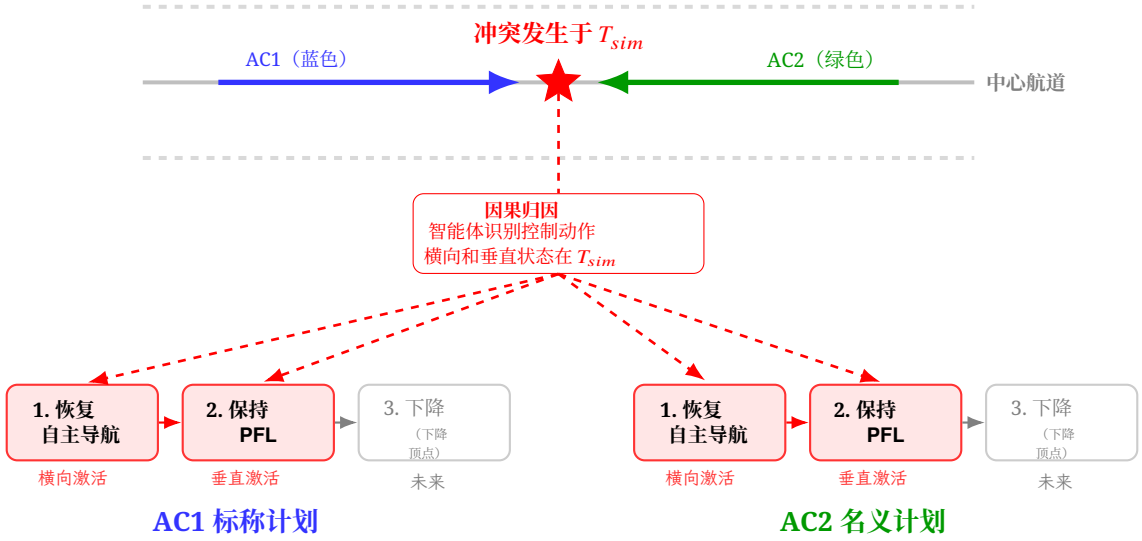

图 6 因果归因与搜索空间剪枝。数字孪生检测到迎头冲突。智能体将其追溯至控制当前状态的活动动作。这些节点被标记为需要修改（红色），而未来/待执行动作（灰色）则保持不变。请注意，名义计划中不会发布速度动作。

检测到冲突时，Mallard 会沿轨迹向后追溯，进行因果归因，以识别直接导致不安全状态的计划动作 （图 6）。隔离这一因果要素后，智能体便可只修改计划中相关的部分，同时保留其余部分。一旦找到安全修订，原先后续的动作仍然有效，从而减少不必要的更改并缩小搜索空间。

II. **计划拼接：**
一旦选定有效策略，Mallard 就会通过计划拼接来应用该策略（图 7）。按顺序排列的计划动作结构使智能体能够仅替换发生冲突的片段，将其替换为新的机动，而该机动本身也是一系列计划动作。在图示示例中，通过对两架航空器计划中的横向计划动作进行互补替换，解决了冲突。

III. **单调轴约束：**
为减少振荡行为并促进收敛至稳定计划，Mallard 采用单调轴约束。一旦干预修改了某个特定轴，例如发出向右偏置指令，该轴便受到约束。此后，任何干预都只能强化该决定（例如增大偏置），不得将其逆转。每个控制轴都分别适用此规则。每项约束都关联一个完成条件，用于指示相应安全问题何时已消除。在图 7 中，τ 标记显示航空器彼此通过、横向约束解除的位置。通过阻止矛盾机动，单调性消除了在不兼容的修正之间循环的现象，并引导搜索趋向稳定解。

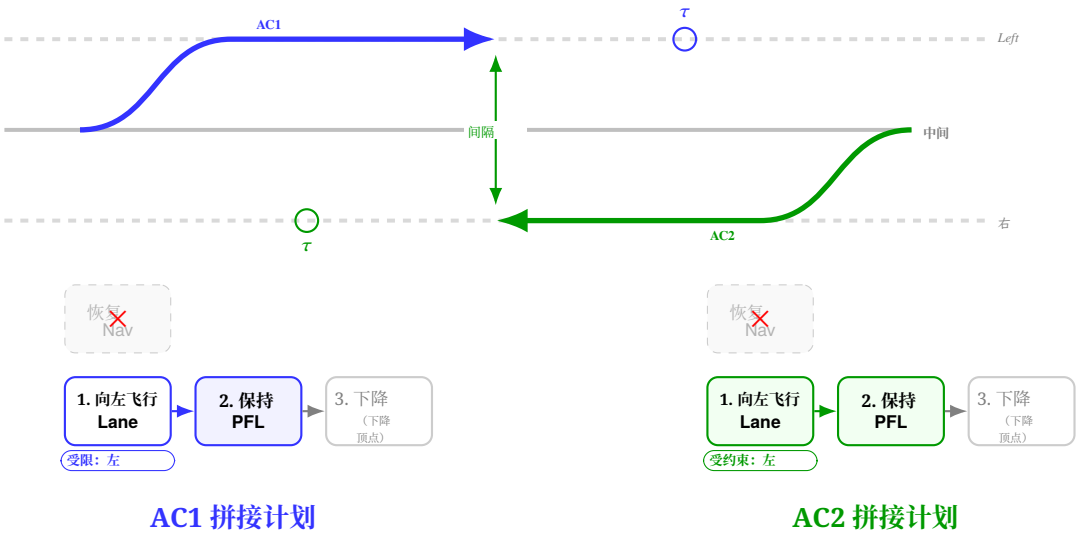

图 7 冲突的“恢复导航”动作（已划掉）被绕过。一个新指令（沿左侧航道飞行）被拼接到序列中，从而约束横向轴。后续的垂直逻辑（保持 PFL）和未来逻辑（下降）得以保留。 τ 表示横向约束解除的未来时刻，使该航空器能够自由采用该轴上的其他冲突消解策略。

算法结构：通过回溯进行顺序解决建立这些基础机制（图 6 和图 7）后，下面将说明 Mallard 如何协调这些机制，形成完整的解决策略。Mallard 提出一种深度受限回溯搜索算法，用于探索每个冲突的排序策略，并协调顺序冲突解决。令 S_0 表示初始空域计划，其中包含扇区内所有航空器的标称飞行计划；令 X_0 表示通过监视观测到的当前空域状态。该算法在算法 1 中形式化描述。

运行流程 该算法通过策略应用与全空域验证的协同循环运行。数字孪生会模拟整个 1 小时规划窗口内的所有航空器计划，并识别预测的间隔丧失，再按时间排序，形成时间优先队列。为保持因果一致性，算法会优先处理最早发生的冲突，确保智能体不会定位那些随后干预本可消除的冲突。搜索采用有界递归深度，其中深度 d 表示自初始空域状态以来连续应用的冲突解决次数。算法从深度 $d = 0$的基线名义计划开始。对于当前深度选定的冲突，算法会按照优先级顺序考虑库中的候选解决策略（表 2）。因果归因（第 III.E.4.I 节）会识别责任计划片段，而单调轴约束（第 III.E.4.III 节）则在评估每项策略之前剪枝不兼容选项。

每个候选策略都通过计划拼接（第 III.E.4.II 节）来应用，仅修改因果片段。随后重新模拟整个空域，以检测级联效应，例如新的冲突或几何形态变化。如果仍存在冲突，算法会递归至深度 $d + 1$，在更新后的状态下解决下一个预测冲突。

如果当前深度下没有任何策略能够解决所有下游冲突，或者达到深度限制 $d_{\max} = 3$，算法就会回溯：撤销最近一次计划修改，并尝试深度 d 下的下一个策略。在某一深度下尝试完所有策略后，算法会进一步回溯到深度 $d - 1$，直到找到完整的安全计划，或所有选项均告失败。需要注意的是，最大深度是在测试过程中选定的；原则上，深度可以按需设置得任意大。

采用贪心接受的策略排序该算法结合了两种设计原则 :

策略排序 每次出现冲突时，都会按优先级顺序考虑策略库中的策略（表 2）。深度限制可避免过多的级联干预。实际上，这种优先级顺序意味着更简单的解（所需干预更少）往往会更早找到，这与操作上偏好低复杂度计划相符。贪心接受首个解： 一旦找到任何完整且安全的航路计划，搜索便会终止。该

算法 1全空域冲突消解

输入： 当前计划集 S、当前空域状态 X、最大深度 $d_{max} = 3$
输出： 完整安全计划集 S' 或回退方案

功能 空域消解(S, X, $depth$)
    $conflicts \leftarrow$ 模拟与检测(S, X) ▷ 向前模拟所有航空器
    如果 $conflicts = \emptyset$ **那么**
        返回 S ▷ 成功：已找到安全计划集
    结束如果
    如果 $depth \geq d_{max}$ **那么**
        返回 NULL ▷ 超出深度限制
    结束 if
    $c \leftarrow$ 最早冲突($conflicts$) ▷ 时间优先级
    $S_{strat} \leftarrow$获取策略(c)
    $S_{strat} \leftarrow$ 按轴约束筛选(S_{strat})
    对所有 $s \in S_{strat}$ **执行** ▷ 按优先级顺序尝试策略
        $S' \leftarrow$ 应用策略(S, s, c) ▷ 对因果片段进行计划拼接（第 III.E.4.II 节）
        $solution \leftarrow$ 空域消解(S', X, $depth + 1$) ▷ 递归调用
        如果 $solution \neq$ 空，**则**
            返回 $solution$ ▷ 第一个完整的安全计划胜出
        结束如果
        撤销策略(S, s, c) ▷ 回溯：恢复先前计划
    结束循环
    返回 空值 ▷ 所有策略均已用尽
结束功能

主程序：
$solution \leftarrow$ 空域消解(S_0, X_0,0)
如果 $solution =$ 空**则**
    返回 回退策略() ▷ 保守垂直间隔
结束如果
返回 $solution$

算法不会穷举所有可能的解来选出全局最优解。这种贪心接受方式可显著节省计算资源：一旦发现有效计划，就不会再进行回溯或策略探索。

例如，如果策略1（例如横向间隔）在深度 d=1 处解决了迎头冲突，且没有产生新的冲突，算法便会立即接受该方案，即使策略3理论上可能生成稍短的路径。

由于策略按优先级顺序进行评估，且搜索受到深度限制，因此搜索往往能找到使用高优先级策略的浅层解，这符合对低复杂度计划的操作偏好。在终止前探索到的解中，第一个完整的安全计划通常是干预深度最浅的计划之一，但无法保证它是全局干预深度最小或全局最优的解。

计算属性 最佳情况：如果每个冲突都能通过排名最高的策略顺利解决（且不会引发次生冲突），则算法的复杂度为 $O(k)$，其中 k 表示初始冲突的数量。实际应用中，设计良好的策略优先级会提高出现这种最佳情况的概率：优先级最高的策略往往能够成功解决冲突，而不会造成后续复杂问题。

最坏情况：在每种策略都会产生需要回溯的新冲突的对抗场景中，复杂度接近 $O(b^{d_{max}})$，其中 d_{max} 是最大搜索深度，b 是分支因子（每种冲突类别对应的策略数量）。不过，有几个因素可以缓解这种指数级的最坏情况：(1) 深度限制（$d_{max} = 3$）限定了搜索范围；(2) 轴约束会剪除许多分支；(3) 大多数现实冲突都能通过简单的解决方案化解；(4) 贪心接受可避免穷举搜索。

终止保证： 该算法保证终止。深度上限确保递归不会无限进行，而有限的策略库确保每个冲突节点的分支因子都有上限。如果搜索空间内不存在安全的航路计划，该算法会穷尽所有可能性并调用回退机制。

回退机制 如果搜索穷尽了深度达 d_{max} 的所有策略组合，仍未找到完整且安全的航路计划，Mallard 就会启动保守的回退方案：立即执行垂直间隔机动（为两架航空器找出未占用的飞行高度层，并指引它们前往），以提供简单、确定的安全缓冲区。系统还会同时向人工管制员发出警报，将该情景标记为需要专家干预的边缘情况。这一故障安全机制旨在应对超出智能体解决能力的高度受限情景，优先确保间隔，同时明确保留人工监督权限。在我们的测试中，此回退方案被视为失败，只是尝试近似一种合理的故障恢复模式。

表 2 冲突消解策略库。识别因果动作后，智能体会按照优先级顺序评估已知策略，以选择一种解决方案。每种策略在实施前都会经过前向仿真测试。

优先级	策略	描述
1	横向间隔（同侧）	用于同一飞行高度层上的对头方向，通过分配中心线同侧的横向偏置 .
2	高度层变更（下降）	一架航空器下降至冲突飞行高度层以下。
3	高度层变更（爬升）	一架航空器爬升至冲突飞行高度层以上；用于较低的高度层不可用。
4	交换高度层	航空器横向分离、垂直方向相互交错，然后重新并入原横向航路。
5	匹配速度并跟随	将两架航空器都设为速度控制，并让一架跟在另一架后面同一航迹上的其他目标 .
6	附加策略	附加策略或参数化变体。

回溯搜索算法通过前向仿真提供了一种评估安全性的机制。然而，仅有安全性不足以获得运行认可。管制员必须能够理解、验证并适当地信任智能体的推理过程。以下章节将介绍 Mallard 的架构设计如何实现这种透明性。

F. 可解释性与人机协同

许多已提出的人工智能空中交通管制智能体存在一个局限，即其不透明性 [12, 13]。对此类系统的信任不仅取决于成功的结果，也取决于底层决策过程的透明性 [26]。Mallard 通过架构设计来应对这一问题：分层计划表示（第三节 D）保留了决策溯源，因果归因（第三节 E.4.I）可识别应负责的计划航段，而完整的解脱算法（算法 1）则保持策略选择历史。为使这种透明性在实际运行中发挥作用，我们开发了可视化工具，旨在帮助人们深入了解算法的运行方式，以及选择解决策略的原因。

1. 用于训练和验证的可视化

为支持空中交通管制员熟悉系统并开展智能体验证，我们提出了一种详细的可视化界面，可从多个抽象层级呈现 Mallard 的内部推理过程：

- **空域计划视图**：显示扇区范围内所有航空器轨迹（图 3），包括名义计划和任何拼接的冲突解脱区段，使管制员能够检查空域的完整预期状态。
- **单航空器计划检查器**：详细显示所选航空器的计划动作序列，以及支配飞行各阶段的条件—动作—完成链。
- **冲突解脱历史**：一种可随时间浏览和拖动查看的可视化，将每个检测到的冲突与其分类、从排序策略库（表 2）中选取的策略以及最终的计划修改关联起来。该轨迹旨在让管制员逐步回放智能体的决策过程（见图 8）。

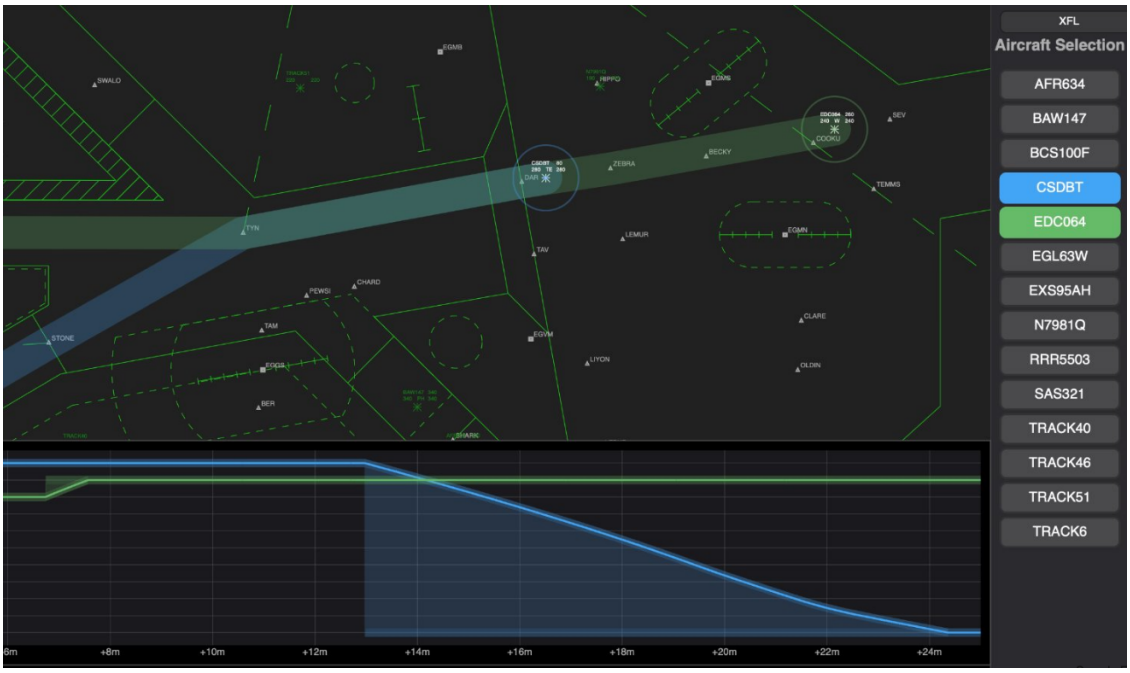

图 8 人机界面（HMI）中的冲突可视化。雷达屏幕上显示横向位置（含安全缓冲区），下方窗格显示垂直意图。系统会自动选中两架存在冲突的航空器，其计划显示约 13 分钟后将出现不安全程序。届时，蓝色航空器将下降穿过绿色航空器，两者的横向位置也将不安全。管制员可以拖动时间线，查看轨迹并确认安全问题。

这些详细视图旨在用于离线分析：了解智能体在训练场景中的推理模式，识别策略选择可能不理想的边缘情况，并通过透明性建立管制员信任。可拖动查看的冲突历史旨在支持场景后复盘，让教员和管制员能够检查级联冲突是如何得到解决的，以及智能体的策略选择是否符合专家判断。

虽然这种详尽的透明性对于验证和建立信任至关重要，但在日常运行中呈现如此详细的信息会造成不必要的认知负担。

2. 用于运行的最简显示

为实现实时人机协同，我们提出一种刻意保持简洁的运行界面。管制员将看到：

- 所有航空器的预期轨迹，以及标示触发条件满足并发布管制许可位置的预期动作点
- 可供管制员审核和发布的管制许可
- 冲突警报及建议的解决方案

这种运行视图旨在聚焦于意图而非推理：管制员可以确认计划的轨迹保持了间隔，并符合扇区协调要求，而无需在日常运行中检查底层的策略选择逻辑。如果管制员希望了解智能体做出特定决策的决策依据，仍可使用详细的可视化工具，但默认情况下不会显示这些工具。两种可视化模式的原型实现均已在 BluebirdDT 环境中开发完成，但评估其对管制员的实际效果仍是未来的工作（第七节D）。这种双模式设计体现了可解释性在不同情境下具有不同作用：一方面，通过透明验证建立信任并进行能力评估；另一方面，在实时运行中仅显示必要的意图信息，以保持态势感知并避免认知过载。

随着从轨迹预测、冲突解决到决策的完整方法建立起来

透明性方面，我们现在说明将如何系统地评估 Mallard 的表现。

IV. 验证方法

智能体性能将使用机器基础训练（MBT） [34], 进行评估。这是一种以人为中心的评估方法，由 Project Bluebird 开发，依据的是 NATS学院对空中交通管制员（ATCO）学员所采用的严格训练和评估标准。MBT 将根据与空中交通管制员培训标准相一致的四个运行维度评估智能体：

- **安全**：在现实的执行不确定性下维持最小间隔
- **规划**：策略质量、最大限度减少运行影响、预防后续冲突
- **协调**：扇区边界处的航空器状态、移交清晰度
- **可解释性**：与标准用语保持一致、逻辑连贯

评估场景取自 BluebirdDT 数字孪生中的空中交通管制员培训材料（第三节 C），涵盖不同的交通密度、冲突几何形态以及级联式多航空器交互。空中交通管制员教员会审查智能体生成的解，评估其运行可接受性、类人性和战略适切性。

五、当前状态与初步验证

Agent Mallard 目前正处于积极开发阶段，核心架构组件已实现，并开始在 BluebirdDT 环境中接受初步验证。初步评估侧重于确立该方法在运行层面的可信度：

A. 与资深空中交通管制员开展逻辑验证

分层机动结构和冲突解决策略已通过与在职英国管制员及空中交通管制员教员进行的结构化演练接受审查。这些环节包括：

- 介绍冲突分类系统和排序策略库
- 在模拟器上演练示例场景，以验证逻辑一致性
- 征求有关机动参数化和完成条件的反馈
- 评估分层计划表示的可解释性

管制员的反馈令人鼓舞：在试验场景中，专家表示，Mallard的决策逻辑总体上类似于他们熟悉的操作推理模式，并且针对所呈现冲突几何形态提出的机动在操作上是合理的。这些咨询还促成了策略排序和参数化方面的若干改进。

B. 部分实现与测试

Mallard架构的关键组件已经实现：

- 分层计划表示（条件、动作、计划动作、机动动作）
- 用于轨迹仿真的数字孪生集成
- 冲突检测与分类逻辑
- 针对常见冲突类别的核心冲突消解策略库

针对少量MBT场景开展的初步测试表明，仿真在环方法能够检测冲突、生成候选机动，并在数字孪生模型下解决冲突而不引入次生违规。系统性的基于MBT的评估活动仍有待开展。

C. 后续步骤：全面MBT评估

通过 MBT框架进行的完整验证仍在进行中 , 目标如下：

1) **场景覆盖：** 针对完整的基于模型的训练课程进行系统测试 2) **量化指标：** 收集安全性、效率和计算性能数据 3) **专家评估：** 由空中交通管制员教员对智能体解进行正式审查 4) **比较分析：** 在匹配场景下与人工管制员表现进行基准比较 5) **失效模式分析：** 识别边界条件和边缘情况

VI. 设计假设与已知局限

Agent Mallard 当前的设计基于一些关键假设，这些假设界定了其适用范围，并明确了未来发展的方向：

系统化空域环境 Mallard 旨在系统化空域中执行战术管制，该空域边界明确，航空器沿基于性能的导航引导航路飞行。该智能体假设：

- 航空器遵守已发布的管制许可，并保持 RNAV1 性能
- 航道结构在几何上保持有效，且间距适当
- 运行条件正常（无严重天气、系统故障或空域结构失效）

当这些假设不再成立时（例如，大范围天气扰动导致需要采用自由航路），当前的 Mallard 设计不能直接照常使用；此时，管制将交由人工空中交通管制员执行，或改用更适合自由航路运行的替代工具。

成对冲突解脱 核心冲突消解逻辑针对成对相遇进行了优化。尽管该架构能够通过迭代解脱处理某些级联冲突，但在航空器高度密集的多航空器集群场景中，可能会超出现有能力范围。这是当前实现已知的局限。

专家知识引导的策略空间 作为一个由专家知识引导的基于规则的系统，Mallard 的策略库体现了编码后的空中交通管制员实践。除预设库中的内容外，它不会生成新的机动类型。这一设计选择优先考虑可解释性和运行一致性，而非不受约束的解空间探索。

仿真保真度依赖 系统性能取决于数字孪生所提供的航空器状态数据、轨迹模型和环境条件的准确性与及时性。数据质量下降或建模误差会传递到智能体的决策中。

搜索终止与回退 如果在这些策略中找不到安全解，系统会将该冲突标记为需要人工干预，并发出“安全飞行高度层”机动指令，旨在迅速将两架航空器引导至更安全的飞行高度层，同时明确放弃协调脱离或飞行经济性考量。

VII. 未来工作

Agent Mallard 正在积极开发中。以下领域是推动系统迈向运行评估与部署的优先事项：

A. 全面验证

当前最紧迫的优先事项是按照完整的 MBT场景大纲 [34]开展系统评估。MBT 提供了一套结构化的运行场景课程，每个场景都代表训练扇区中约三十分钟的模拟交通流量。这些场景按照具体的评估标准设计，旨在系统地培养智能体能力，从基础的冲突解决逐步推进到日益复杂的战术情境。目前的测试仅涵盖少量基础场景，且冲突几何形态彼此独立。要完成验证，必须做到：

- **渐进式复杂度：** 系统地循序推进完整的 MBT 课程，涵盖包含多个并发冲突、级联交互和受限解空间的场景
- **运行真实性：** 在真实的扇区交通流条件下进行测试，而非采用人为构造的成对相遇；测试还应涵盖正常的进入／离开模式、不同的航空器性能，以及混合飞行剖面（爬升、巡航、下降）
- **评估标准覆盖度：**展示在所有 MBT 评估维度上的能力——安全（不确定性下的间隔保障）、规划（策略质量和效率）、协调（退出状态管理）以及可解释性（解的运行清晰度）
- **定量性能指标：**收集完整场景集中的间隔保障率、轨迹效率（燃油消耗、时间偏差、路径延伸率）、计算性能和策略选择分布
- **正式专家评估：**由空中交通管制员教员对智能体生成的解进行结构化审查，在匹配场景中将 Mallard 的方法与人工管制员解决方案进行比较

这项系统性评估将确定性能边界、失效模式，以及在考虑实际部署之前需要扩展策略库或改进算法的场景。

B. 策略库扩展与专业知识引出

当前策略库（表 2）涵盖了通过初步专家咨询确定的基本冲突类别。然而，运行专业技能分散在经验水平、扇区专业领域和区域实践各不相同的管制员之中。通过结构化的空中交通管制员参与系统地扩展策略库，既可作为一种实用的完善机制，也能让我们了解空中交通管制中战术推理的多样性。

拟议的引出方法：

鉴于 Mallard 是在 MBT场景中进行评估的，管制员可以查看 Mallard 提出的解并提供结构化反馈：认可该方案在运行上合理，或提出他们在相同情境下更倾向采用的替代策略。由于 Mallard 的分层计划表示具有可解释性——明确展示冲突分类、策略选择和参数化机动序列——管制员可以有针对性地评议决策依据，而不仅仅是结果。这将有助于策略发现（识别当前策略库中尚未收录的有效方案，并记录管制员倾向于采用替代方案的条件）、参数化细化（观察管制员如何在特定情境下调整机动参数）、策略排序校准（记录专家偏好与当前排序存在差异之处及其原因），以及识别依赖情境的适应规则（总结在特定运行场景下应如何调整策略的规律，例如出口协调、流量管理、相邻扇区约束、天气或临时限制）。

潜在的研究洞见：

这一迭代式完善过程有望进一步揭示战术专业知识的结构。通过系统记录管制员的偏好在何处以及为何不同于编码后的策略库，这种方法或许有助于刻画以下方面：管制员群体在战术决策中的共识程度与差异程度；策略选择主要由冲突几何关系驱动（这意味着存在普适原则），还是由运行情境驱动（这意味着需要扇区特定专业知识）；以及不同约束条件（安全裕度、效率、协调复杂度、工作负荷）在形成策略偏好方面的相对重要性。Mallard 的可解释性对这一过程至关重要：由于该智能体能够在冲突分类、策略选择和机动分解层面展现其推理过程，管制员可以在适当的抽象层级提供反馈，调整高层策略选择，而非低层控制参数。因此，策略库开发不仅是系统改进，也可能成为研究安全关键领域中专家战术推理的一种方法。

C. 算法增强

利用现代多核架构同时评估多个候选策略，可以缩短实际运行求解时间。这与已定义的深度受限回溯搜索完全兼容。

D. 人机协作协议

虽然第三节.F节所述的可视化架构支持透明性，但系统性的实证评估仍有待开展。尚待解决的关键问题包括：实际运行中最优的信息密度，以及管制员覆盖智能体建议时所采用的干预协议。应通过结构化的用户研究对这些问题进行调查，并结合客观绩效指标与主观信任和工作负荷评估。

最终目标并非取代人工管制员，而是为常规战术冲突解脱提供可靠、透明的决策支持，使管制员能够将注意力分配到战略协调、非正常情况和高层级流量管理上；在这些方面，人的判断仍不可或缺。

致谢

本研究得到了 由英国工程与自然科学研究委员会（EP/V056522/1）和 N 提供支持 ATS.

参考文献

[1] 美国联邦航空管理局，《FAA航空航天预测表：2024—2044财年》，2024年。网址 https://www.faa.gov/data_research/aviation/aerospace_forecasts.[2] 欧洲航空安全组织，《2040年欧洲航空：增长面临的挑战》，技术报告，欧洲航空安全组织，2018年10月。网址 https://www.eurocontrol.int/publication/challenges-growth-2018.[3] NATS，《英国空中交通管制中的 iFACTS：飞行员须知》，2014年5月。网址 https://www.ukfsc.co.uk/files/External% 20Meetings/NATS%20SPA% 202012/NATS%20SPA%20iFacts%20May%202014.pdf.[4] 美国联邦航空管理局，《用户请求与评估工具（URET）》，2014年。网址 https://automation. forthillgroup.com/story/URET.[5] 欧洲航空安全组织，《中期冲突探测欧洲航空安全组织规范》，技术报告 EUROCONTROL-SPEC- 0139，欧洲航空安全组织，2017年3月。网址 https://www.eurocontrol.int/sites/default/files/publication/ files/EUROCONTROL-SPEC-0139%20MTCD%20Ed% 202.0.pdf.[6] 美国国家航空航天局，《空域技术演示2（ATD-2）：进场/离场/地面综合交通管理（IADS）》，2019年4月。网址 https://aviationsystems.arc.nasa.gov/ research/atd2/ATD2_FS-2019-04-03-ARC_Final.pdf.[7] SESAR联合事业，《AISA——推动自动化发展的人工智能态势感知基础》，2022年。网址 https:// www.sesarju.eu/projects/aisa.[8] 艾伦·图灵研究所，《Project Bluebird：用于空中交通管制的人工智能系统》，2025年。网址 https://www.turing.ac. uk/research/research-programmes/project-bluebird.[9] Ghosh, S., Laguna, S., Lim, S. H., Wynter, L., 和 Poonawala, H.，《一种用于空中交通管制的深度集成多智能体强化学习方法》，*arXiv*预印本 *arXiv:2004.01387*，2020年。网址 https://arxiv.org/abs/2004.01387.[10] Brittain, M., 和 Wei, P.，《自主空中交通管制员：一种深度多智能体强化学习方法》，*arXiv*预印本 *arXiv:1905.01303*，2019年。网址 https://arxiv.org/abs/1905.01303.[11] Nilsson, J., Unger, J., 和 Eilertsen, G.，《有限指令条件下用于空中交通管制冲突解决的自优先级多智能体强化学习》，航空航天，第12卷第2期，2025年，第88页。 https://doi.org/10.3390/aerospace12020088.[12] (EASA), E. U. A. S. A.，《人工智能路线图：航空领域人工智能的以人为中心的方法》，技术报告，欧盟航空安全局（EASA），2020年2月。网址 https://www.easa.europa.eu/sites/default/files/dfu/EASA-AI- Roadmap-v1.0.pdf.[13] IFATCA联合认知人机系统团队，《欧洲空中交通管理系统引入人工智能所面临的挑战：人机视角》，技术报告，IFATCA联合认知人机系统团队，2022年7月。网址 https://www.atcguild.in/iwen/iwen3222/General/IFATCA_JCHMS_TEAM_PAPER_FINAL% 20220708.pdf.[14] Perez-Cerrolaza, J., Englund, C., Donzella, C., 和 Borg, M.，《工业与交通领域安全关键系统中的人工智能：综述》，*ACM*计算调查，2023年。https://doi.org/10.1145/3626314.[15] NATS，《空域的未来》，2025年。网址 https://www.nats.aero/airspace/future/.[16] Suárez, M. Z., 等，《理解有关空中交通管制员工作负荷的研究：系统综述》，应用人体工程学，第102卷，2024年，第103736页。https://doi.org/10.1016/j.apergo.2022.103736.[17] Histon, J. M., 和 Hansman, R. J.，《缓解空中交通管制复杂度：基于结构的抽象的作用》，博士论文，麻省理工学院航空学系……，2008年.[18] Djokić, J.，《作为工作负荷驱动因素的空中交通管制复杂度》，《交通研究*C*辑：新兴技术》，第18卷，2010年，1020 1034 10 1016 2010 03 005 pp. – . https://doi.org/ . /j.trc. . . .[19] 国际民用航空组织，《区域导航（RNAV）简介》，技术报告，国际民用航空组织，2014年10月。网址 https://www.icao.int/WACAF/Documents/Meetings/2014/OPS-Approval/14%20October%202014/01%20- % 20Introduction%20to%20RNAV.pdf.[20] Administration, F. A.，《NextGen年度报告：2022财年》，技术报告，美国联邦航空管理局，2022年10月。网址 https://www.faa.gov/nextgen/NextGen-Annual-Report-2022.pdf.

[21] 欧洲航空安全组织，“点汇聚：改进并协调进场运行”，，2025年。网址：https://www.eurocontrol.int/concept/point-merge。[22] Undertaking, S. . J.,“2022–2023年度双年度工作计划：第四次修订版”，技术报告，SESAR 3 联合事业，2023年4月。网址：https://www.sesarju.eu/sites/default/files/2023-06/Bi-Annual%20Work%20Programme% 20for%20years%202022-2023%20-%20Fourth%20amended%20version_0.pdf。[23] Delahaye, D., Peyronne, C., Mongeau, M., and Puechmorel, S.,“通过遗传算法和B样条近似解决飞机冲突”，*ENRI* 空中交通管理/通信、导航与监视国际研讨会（*EIWAC*）第二届会议论文集，日本东京，2010年。[24] Durand, N., and Alliot, J.-M.,“航路冲突的最优解决”，美欧空中交通管理研发研讨会第一届会议论文集，法国萨克雷，1998年。[25] Isufaj, R., Aranega Sebastia, D., and Angel Piera, M.,“利用深度多智能体强化学习实现冲突解决”，《航空运输杂志》，第30卷，第3期，2022年，第71–80页。[26] Mohamed, E., Carvell, B., Procter, R., De Ath, G., and Everson, R., 面向空中交通管制的透明人工智能智能体，美国航空航天学会，2026年。[27] Carvell, B., De Ath, G., Benjamin, E., and Everson, R., 在线动作堆叠提升空中交通管制强化学习的性能，美国航空航天学会，2026年。[28] Callantine, T. J.,“基于CATS的空中交通管制员智能体”，技术报告，美国国家航空航天局，2002年。[29] Callantine, T.,“用于支持超高密度运行研究的TRAC试验规划与场景生成”，美国航空航天学会建模与仿真技术会议，2009年，第5836页。[30] Prevot, T.,“探索分布式空中交通管理的多种视角：多机控制系统MACS”，*HCI-Aero*会议论文集，第149卷，2002年，第154页。[31] Pepper, N., Keane, A., Dewi Gould, E. H., Hodgkin, A., Lauritsen, L., Vlahos, C., De Ath, G., Everson, R., Cannon, R., Castro, A. S., Korna, J., Carvell, B., and Thomas, M., 用于训练和评估空中交通管制人工智能智能体的英国航路空域概率数字孪生，美国航空航天学会，2026年。[32] NATS，“Project Bluebird：探索人工智能在空中交通管制中的应用”，，2025年。网址：https://www.nats.aero/about-us/research/n/project- bluebird/。[33] Keane, A., Pepper, N., Burr, C., Hodgkin, A., Gould, D., Korna, J., and Thomas, M., 用于确保英国航路空域人工智能数字孪生准确性和保真度的框架，美国航空航天学会，2026年。[34] Carvell, B., Thomas, M., De Ath, G., Everson, R., Pepper, N., Keane, A., and Cannon, R., 基于受监管评估框架的空中交通管制人工智能智能体人在回路测试，美国航空航天学会，2026年。[35] Pepper, N., De Ath, G., Carvell, B., Hodgkin, A., Dodwell, T., Thomas, M., and Everson, R.,“用于四维飞机轨迹生成的扇区特定概率方法”，《交通研究*C*辑：新兴技术》，第179卷，2025年，第105291页。https://doi.org/10.1016/j.trc.2025.105291。[36] Hodgkin, A., Pepper, N., and Thomas, M.,“基于物理信息机器学习方法的飞机下降概率仿真”，，2025年。网址：https://arxiv.org/abs/2504.02529。[37] Seamster, T. L., Redding, R. E., Cannon, J. R., Ryder, J. M., and Purcell, J. A.,“空中交通管制专业技能的认知任务分析”，《国际航空心理学杂志》，第3卷，第4期，1993年，第257–283页。