An Efficient Algorithm for Self-Organized Terminal Arrival in Urban Air Mobility

城市空中出行中自组织终端到达的高效算法

Conference Paper - January 2020
会议论文 - 2020 年 1 月

DOI: 10.2514/6.2020-0660

An Efficient Algorithm for Self-Organized Terminal Arrival in Urban Air Mobility

城市空中出行中自组织终端到达的高效算法

Joshua R. Bertram Peng Wei
Joshua R. Bertram Peng Wei
Iowa State University
爱荷华州立大学
Ames, IA 50011
Ames, IA 50011
{bertram1, pwei}@iastate.edu
{bertram1, pwei}@iastate.edu

Urban Air Mobility (UAM) is a concept for future air transportation where air taxis move passengers between vertical take-off and landing sites known as vertiports. While some form of a structured airspace is likely, it is expected that UAM systems will be required to deal with high traffic densities and will need to respond to conflicts due to a dynamically changing environment. Due to the difficulty of predicting demand of air taxi usage, it may also be more difficult to predict the required demand on any given vertiport. We explore an airspace design which can regulate the flow of aircraft landing at a vertiport, maintaining the aircraft in sequence until capacity is available at the vertiport. We demonstrate an implementation of the airspace design using a highly-efficient Markov Decision Process (MDP) based algorithm to provide separation and collision avoidance for a UAM terminal arrival sequencing problem. The ver-tiport maintains basic information about capacity and sequencing, and the aircraft seamlessly perform guidance in a self-organized distributed manner performing conflict avoidance while waiting to land.

城市空中出行 (UAM) 是未来航空运输的一个概念，其中空中出租车在被称为垂直起降场的垂直起降地点之间运送乘客。尽管可能会存在某种形式的有序空域，但预计 UAM 系统将需要处理高交通密度，并且需要对动态变化的环境中的冲突作出响应。由于预测空中出租车的使用需求较为困难，因此预测任何给定垂直起降场的需求可能也会更加困难。我们研究了一种空域设计方案，该方案可以调节在垂直起降场降落的飞机流，保持飞机按顺序排列，直到垂直起降场有空闲容量。我们展示了一种空域设计实现的示例，使用基于高效马尔可夫决策过程 (MDP) 的算法为 UAM 终端到达排序问题提供分离和避障。垂直起降场维护有关容量和排序的基本信息，飞机在等待着陆时以自组织的分布式方式无缝执行引导，同时进行冲突避免。

I. Introduction

I. 引言

Urban Air Mobility (UAM) is a concept for future air transportation in which partially or fully autonomous air vehicles transport cargo and passengers through dense urban environments. To minimize the footprint required on the ground in urban environments, UAM concepts explore Vertical Take Off and Landing (VTOL) aircraft at sites known as vertiports. UAM aircraft are envisioned as navigating from vertiport to vertiport without established schedules or routes which will make terminal area capacity planning more difficult than for airports where scheduled commercial traffic and registered flight plans are available. In addition to UAM aircraft's inherent irregularity stemming from ad-hoc passenger requests, UAM aircraft can be expected to regularly change their flight path or destination due to changing passenger needs. For example, a passenger may wish to change the destination vertiport mid-flight due to a last-minute meeting cancellation. Similarly, a UAM vehicle may need to switch vertiports mid-flight due to a passenger pickup cancellation leading to the air taxi being dispatched to an alternate pickup location.

城市空中出行 (UAM) 是一个关于未来航空运输的概念，部分或完全自主的航空器在密集的城市环境中运输货物和乘客。为了最小化在城市环境中所需的地面占用面积，UAM 概念探索了垂直起降 (VTOL) 飞机，这些飞机在被称为垂直机场的地点起降。UAM 飞机被设想为在垂直机场之间导航，没有固定的时刻表或路线，这将使得终端区域容量规划比那些有计划商业交通和注册飞行计划的机场更为困难。除了 UAM 飞机固有的不规律性，源于临时乘客请求之外，预计 UAM 飞机会因乘客需求的改变而经常改变其飞行路径或目的地。例如，乘客可能因临时取消会议而希望在飞行途中改变目的地的垂直机场。同样，UAM 车辆可能因乘客取消接机而在飞行途中切换垂直机场，导致空中出租车被派往备选的接机地点。

While we expect some form of terminal area air traffic control to manage each vertiport, we should also expect that the aircraft themselves respond to congestion appropriately by avoiding collision and adapting to circumstances as they develop. The system should also respond well in the case of communications datalink loss, which on aircraft can easily occur as the aircraft banks and turns temporarily blocking the propagation of radio signals due to the orientation of the airframe, airborne and ground based antennas.

虽然我们期望某种形式的终端区域空中交通控制来管理每个垂直机场，我们也应该期望飞机本身能够通过避免碰撞和适应发展中的情况来适当地响应拥堵。此外，系统还应该能够在通信数据链路丢失的情况下良好响应，这种情况在飞机上很容易发生，因为飞机在转弯和倾斜时会暂时阻挡由于机身方向、机上和地面天线传播的无线电信号。

In dealing with congestion, we also expect to transfer some ideas from commercial aviation airspace design that help organize incoming traffic into manageable flows that are designed to reduce conflict. One such idea is the use of metering fixes or arrival gates which serve as staging points that concentrate arriving aircraft into flows which can be sequenced. Aircraft are sequenced in order to provide an orderly flow of aircraft from the arrival gates to the landing areas. Whatever solution is implemented, it must also be simple and efficient enough that it can be deployed on embedded processors found on airborne vehicles.

在处理拥堵问题时，我们也期望借鉴商业航空空域设计的某些理念，这些理念有助于将进入的流量组织成可管理的流动，旨在减少冲突。其中一个理念是使用计量修正或到达门，作为集结点，将到达的飞机集中到可以排序的流动中。飞机被排序，以便从到达门到着陆区域提供有序的飞机流。无论实施何种解决方案，它都必须足够简单高效，以便可以部署在飞行器上的嵌入式处理器上。

In this paper we combine an airspace design with a real-time computational guidance algorithm that also performs collision avoidance. We propose a simple concentric rings based structure for the terminal airspace that regulates the flow of incoming aircraft. We demonstrate the concept using a 3D simulation environment which models the aircraft with a pseudo-6dof model and tracks Near Mid-Air Collisions (NMACs) during execution.

在本文中，我们将空域设计与实时计算引导算法相结合，该算法还执行避撞操作。我们为终端空域提出了一种基于同心环的简单结构，以调节进入飞机的流量。我们使用一个 3D 模拟环境来演示这个概念，该环境使用伪 6 自由度模型模拟飞机，并在执行过程中跟踪近空中碰撞 (NMACs)。

II. Related Work

II. 相关工作

NASA, Uber and Airbus have been exploring the use of vertical takeoff and landing (VTOL) aircraft for Urban Air Mobility (UAM) [1-5]. In general, the UAM concept calls for UAM aircraft taking off and departing from small-scale airports known as vertiports where VTOL aircraft depart and arrive.

美国国家航空航天局 (NASA)、Uber 和 Airbus 一直在探索垂直起降 (VTOL) 飞机在都市空中出行 (UAM) 中的应用 [1-5]。一般来说，UAM 概念要求 UAM 飞机从小型机场起飞和离开，这些机场被称为垂直机场，VTOL 飞机在这里起飞和到达。

Researchers have examined structured airspace approaches. In [6] a vertiport is defined with two arrival and two departure metering fixes to separate climbing and descending traffic, though the focus of the paper is primarily on scheduling time of arrival for electric VTOL with limited battery charge rather than guidance or collision avoidance. Similarly, [7] also examines the time of arrival scheduling with a simple vertiport model. In [8], multiple vertiports over an urban area are modelled for purposes of scheduling network traffic, but the airspace structure itself is not. [9] discusses the overall UAM concept and describes example urban areas showing potential vertiport placement in example urban areas. In [10], an airspace concept known as streams is described which attempts to break airspace into separately managed, related streams of traffic bound for the same destination. In [11], the FAA performed a study of the North Texas Metroplex for the purposes of optimizing the airspace for precision based navigation. The study analyzes the airspace in the metroplex and describes a number of proposed changes to the airspace, including alterations to the standard terminal arrival route (STAR) metering fixes around DFW and DAL airports to better utilize the airspace and to eliminate identified inefficiencies. This report also examines airspace flows and describes trade-offs against different alternatives.

研究人员已经研究了一种结构化的空域方法。在文献 [6] 中，定义了一个带有两个到达和两个出发计量修正点的垂直机场，以分离上升和下降交通，尽管该论文的重点主要是调度电动垂直起降飞机的到达时间，其电池电量有限，而不是指导或避障。同样，文献 [7] 也研究了使用简单垂直机场模型的时间调度。在文献 [8] 中，为了调度网络流量，对一个城市区域内的多个垂直机场进行了建模，但没有对空域结构本身进行建模。文献 [9] 讨论了整体城市空中出行 (UAM) 概念，并描述了示例城市区域中潜在的垂直机场位置。在文献 [10] 中，描述了一种称为“流”的空域概念，试图将空域划分为分别管理的、相关的、流向同一目的地的交通流。在文献 [11] 中，美国联邦航空管理局 (FAA) 对北德克萨斯都会区进行了研究，以优化用于精确导航的空域。该研究分析了都会区的空域，并描述了对空域的一些建议变更，包括改变达拉斯/沃斯堡 (DFW) 和达拉斯 (DAL) 机场周围的标准化终端到达路线 (STAR) 计量修正点，以更好地利用空域并消除已确定的低效之处。这份报告还检查了空域流量，并描述了针对不同替代方案的权衡。

The method in this paper phrases the problem as a Markov Decision Process (MDP). MDPs are known to be difficult or intractable for large problems, but a recently discovered method for efficiently computing them was described in [12]. We use this underlying algorithm to provide guidance and collision avoidance while adding to it an airspace design to demonstrate efficient terminal arrival.

本文中的方法将问题表述为马尔可夫决策过程 (MDP)。MDP 在大型问题中是已知的困难或不可处理的，但在文献 [12] 中描述了一种新发现的计算 MDP 的高效方法。我们使用这个基础算法来提供指导和避障，同时增加了一个空域设计，以展示有效的终端到达。

III. Method

III. 方法

Figure 2 shows a conceptual design for a vertiport terminal area controller (VTAC). The terminal area airspace is composed of one or more concentric rings $r_i \in \{r_1, \cdots, r_n\}$. Each ring can support a limited number of aircraft known as the capacity c_i which is determined by the circumference and ideal separation distance. A region around the vertiport within the innermost rings is reserved for vertical take off and landing (VTOL) operations at the vertiport and is assumed to be handled by some other controller outside the scope of this paper. The vertiport capacity c_v is the number of aircraft the vertiport can simultaneously allow to land. The VTAC can thus support a total capacity of $C = c_v + \sum_{i=1}^{n} c_i$ aircraft.

图 2 展示了一个垂直起降场终端区域控制器 (VTAC) 的概念设计。终端区域空域由一个或多个同心环 $r_i \in \{r_1, \cdots, r_n\}$ 组成。每个环能够支持有限数量的飞机，称为容量 c_i ，它由周长和理想分离距离决定。在最近的环内围绕垂直起降场的区域预留用于垂直起降 (VTOL) 操作，并且假设由本文范围之外的

某些其他控制器处理。垂直起降场的容量 c_v 是垂直起降场可以同时允许着陆的飞机数量。因此，VTAC 可以支持的总容量为 $C = c_v + \sum_{i=1}^{n} c_i$ 架飞机。

An approach threshold is defined at a fixed radius from the vertiport beyond which approaching aircraft operate in free flight where they navigate to their goals while avoiding collision with other aircraft. k approach gates are defined along the circumference of the approach threshold. If an aircraft wishes to land at the vertiport, it is assumed it communicates its intent to the vertiport, the vertiport verifies it has capacity, and the vertiport grants permission to enter the pattern. The approaching aircraft then proceeds to the nearest approach gate. Once the aircraft crosses the approach threshold, it is then under the control of the VTAC and is assigned a sequence number in a first-come-first-served manner.

在距离垂直起降场固定半径处定义了一个进近阈值，超过该阈值的进近飞机在自由飞行中操作，它们在导航至目标的同时避免与其他飞机碰撞。k 在进近阈值的圆周上定义了进近门。如果一架飞机希望降落在垂直起降场，假设它会向垂直起降场传达其意图，垂直起降场验证其是否有容量，并授予进入进近模式的许可。然后进近的飞机前往最近的进近门。一旦飞机穿过进近阈值，它就处于 VTAC 的控制之下，并按照先来先服务的顺序分配一个序列号。

The VTAC examines the capacity of the vertiport and the rings to determine where the aircraft should be assigned. In general, the aircraft will be assigned to the inner-most ring with spare capacity. If all rings are empty, then the aircraft will be directed to proceed to the vertiport for final approach.

VTAC 检查垂直起降场和环的容量，以确定飞机应该被分配到何处。通常，飞机将被分配到最内层的有空闲容量的环。如果所有环都为空，那么飞机将被指引继续前往垂直起降场进行最终进近。

All motion around the rings is in the same direction (assumed to be counter-clockwise in this paper). When an aircraft is assigned to a ring, the aircraft is responsible for entering the ring while maintaining separation from other aircraft. Other aircraft already in the ring are also responsible for maintaining separation. It is assumed a mechanism or sensor exists which can communicate or broadcast the position of all aircraft, possibly based off GPS positions, ground based radar, or ADS-B. The VTAC maintains a first-in-first-out FIFO queue for each ring so that the location of each aircraft within the airspace structure is known. When extracting the next entry from the FIFO queue, it will always be the aircraft with the lowest sequence number.

环绕环的所有运动都在同一方向 (本文假设为逆时针方向)。当一架飞机被分配到一个环时，该飞机负责进入环并与其他飞机保持间隔。环内其他飞机也负责保持间隔。假设存在一种机制或传感器，能够传达或广播所有飞机的位置，可能基于 GPS 位置、地面雷达或 ADS-B。VTAC 为每个环维护一个先进先出 FIFO 队列，以便知道每个飞机在空域结构中的位置。当从 FIFO 队列中提取下一个条目时，总是具有最低序列号的飞机。

As aircraft flow through this airspace structure, first the inner most ring is filled, then the next ring is filled, and so on until all rings are full. If all rings are full, then the VTAC cannot allow any additional aircraft to enter the pattern and any new requests will be denied.

随着飞机流经这个空域结构，首先最内层的环被填满，然后是下一个环，依此类推，直到所有环都满。如果所有环都满了，那么 VTAC 不能允许任何额外的飞机进入该模式，任何新的请求都将被拒绝。

As the vertiport is able to accept new aircraft, aircraft in the inner most ring are selected for landing and are placed in a final approach queue. The aircraft then leave the inner most ring and approach the vertiport's VTOL region. Once they cross the threshold into this region, they are no longer under control of the VTAC and control transitions to the VTOL controller (outside the scope of this paper.)

当垂起机场能够接受新的飞机时，最内层环中的飞机被选中着陆并放入最终进近队列。然后飞机离开最内层环并接近垂起机场的 VTOL 区域。一旦他们越过这个区域的门槛，他们就不再受 VTAC 的控制，控制权转移给 VTOL 控制器 (不在本文的范围内)。

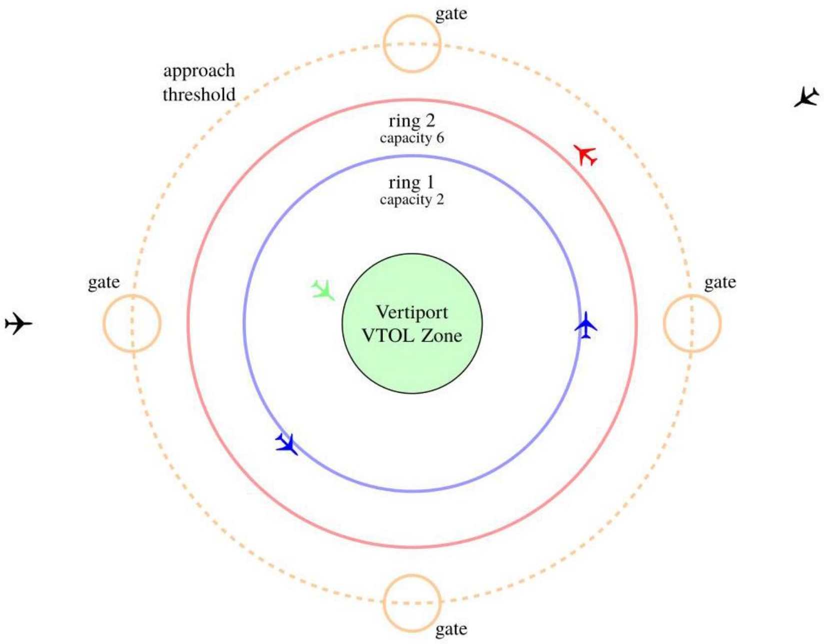

Fig. 1 Terminal arrival airspace design showing two rings and four gates. Ring 1 has a capacity of 2 while ring 2 has a capacity of 6, meaning that the vertiport can keep up to 8 aircraft in the pattern while waiting for the vertiport to allow another aircraft to land. Traffic enters the airspace and approaches the closest arrival gate using free flight. Once each aircraft passes the approach threshold, it is assigned a sequence number and is under the direct control of the vertiport controller. The vertiport controller assigns the aircraft to a ring with spare capacity. The aircraft joins the ring avoiding traffic to ensure separation. As the vertiport has capacity to absorb additional aircraft, the aircraft with the next sequence number is directed to land at the vertiport. As inner rings gain spare capacity, aircraft are cycled from outer rings to inner rings in sequenced order. A separate vertiport control scheme (outside the scope of this paper) manages VTOL ascent and descent.

图 1 终端到达空域设计，展示两个环和四个门。环 1 的容量为 2，而环 2 的容量为 6，这意味着在等待 vertiport 允许另一架飞机着陆时，vertiport 可以在模式中保持最多 8 架飞机。交通进入空域并使用自由飞行接近最近的到达门。一旦每架飞机通过接近阈值，它将被分配一个序列号并处于 vertiport 控制员的直接控制之下。Vertiport 控制员将飞机分配到有剩余容量的环。飞机加入环时避开交通以确保分离。随着 vertiport 有能力吸收额外的飞机，下一个序列号的飞机被指引在 vertiport 着陆。随着内环获得剩余容量，飞机按顺序从外环转移到内环。一个单独的 vertiport 控制方案 (不在本文范围内) 管理 VTOL 的上升和下降。

The altitude of the rings could take different forms. We suggest that the outer rings be at the same or slightly higher altitude than inner rings. We envision that departing aircraft will depart below the altitude of the rings to ensure separation between arriving and departing aircraft. Alternatively, the departing VTOL aircraft could ascend above the altitude of the rings before moving into forward mode.

环的高度可以采取不同的形式。我们建议外环与内环高度相同或略高。我们设想出发的飞机将在环的高度以下出发，以确保到达和出发飞机之间的分离。或者，出发的 VTOL 飞机可以在移动到前进模式之前上升到环的高度以上。

Now that we have defined the concept of operation of this vertiport terminal area controller (VTAC), we next describe an implementation of the controller based off a highly efficient algorithm that can perform collision avoidance while navigating to goals.

现在我们已经定义了 vertiport 终端区域控制器 (VTAC) 的操作概念，接下来我们将描述一个基于高效算法的控制器实现，该算法可以在导航至目标时执行避障。

A. FastMDP Algorithm

A. FastMDP 算法

To implement a demonstration of this airspace design, we build on a recently proposed algorithm [12, 13] that can navigate through complex airspace environments to goals while avoiding collisions with aircraft and terrain. The problem is formulated as a Markov Decision Process (MDP) which accepts a set of positive and negative rewards. We provide a very brief overview of Markov Decision Processes and refer the reader to [13] for a more detailed description of the algorithm and aircraft dynamics. We can summarize by saying that a pseudo-6DOF model is used for the aircraft dynamics and that it is constrained with limits on the dynamics and actions to represent air taxi flight characteristics. While this model is not aerodynamically comprehensive, it is sufficient to describe aircraft motion suitable for examining our airspace design without loss of generality.

为了实现这一空域设计的演示，我们在最近提出的算法 [12, 13] 的基础上进行构建，该算法能够导航穿越复杂的空域环境，到达目标的同时避免与飞机和地形发生碰撞。问题被构建为一个马尔可夫决策过程 (MDP)，它接受一组正负奖励。我们提供了关于马尔可夫决策过程的非常简短概述，并指引读者查阅 [13] 以获取关于算法和飞机动力学的更详细描述。我们可以总结说，使用了一个伪六自由度模型来模拟飞机动力学，并且对其进行了限制，以反映限定在动力学和动作上的空中出租车飞行特性。虽然这个模型在空气动力学上不是全面的，但它足以描述飞机运动，适合检查我们的空域设计而不失一般性。

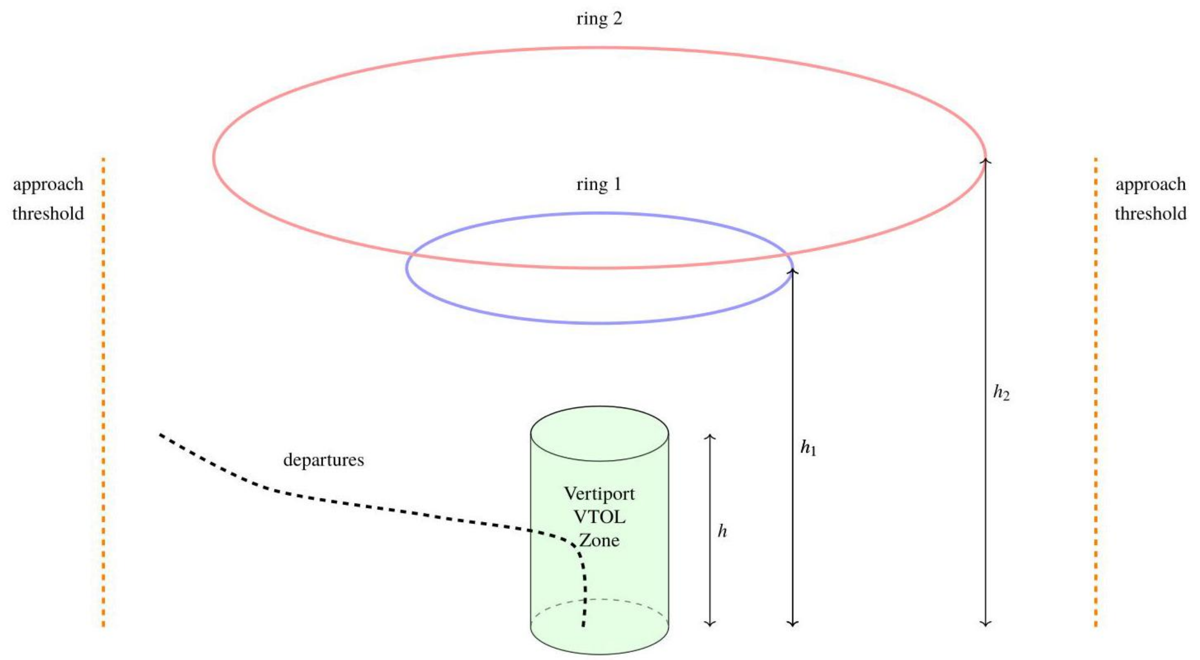

Fig. 2 Profile view of vertiport terminal airspace showing relative altitudes of arrival and departures. The rings may be placed co-altitude (that is, h_1 and h_2 are the same) or outer rings may be at higher altitudes than inner rings. In this diagram, ring altitudes and diameters are not to scale. We envision that departing aircraft would use lower altitudes than ring traffic, but if required departing vehicles could ascend through the center of the rings to a higher altitude than the rings.

图 2 垂直机场终端空域的剖面视图，显示了到达和出发的相对高度。这些环可以放置在相同高度 (即 h_1 和 h_2 是相同的)，或者外环可以比内环高度更高。在这个图中，环的高度和直径并未按比例绘制。我们设想出发的飞机将使用比环交通低的飞行高度，但如果需要，出发的飞行器可以通过环的中心上升到一个比环更高的高度。

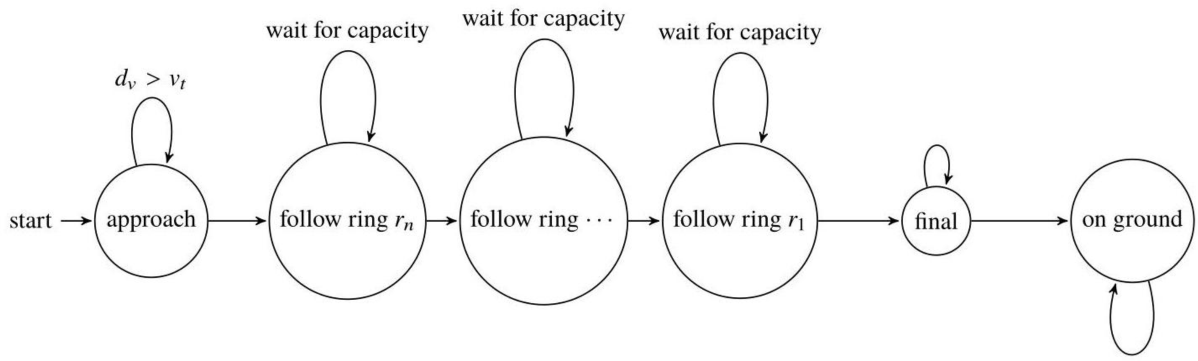

Fig. 3 Per-aircraft arrival state machine. In the figure d_V is the distance to the vertiport and v_t is the approach threshold radius defined around the vertiport. The rewards defined for the MDP will differ depending on the per-aircraft state in this state machine.

图 3 单架飞机到达状态机。在图中 d_V 是到垂直机场的距离，v_t 是围绕垂直机场定义的接近阈值半径。MDP 定义的奖励将根据状态机中每架飞机的状态有所不同。

B.MDP Formulation

B. MDP 构建

Markov Decision Processes (MDPs) are a framework for sequential decision making with broad applications to finance, robotics, operations research and many other domains [14]. MDPs are formulated as the tuple (s_t, a_t, r_t, t) where $s_t \in S$ is the state at a given time $t, a_t \in A$ is the action taken by the agent at time t as a result of the decision process, r_t is the reward received by the agent as a result of taking the action a_t from s_t and arriving at s_{t+1} , and $T(s_t, a, s_{t+1})$ is a transition function that describes the dynamics of the environment and capture the probability $p(s_{t+1} \mid s_t, a_t)$ of transitioning to a state s_{t+1} given the action a_t taken from state s_t . A policy π can be defined that maps each state $s \in S$ to an action $a \in A$. From a given policy $\pi \in \Pi$ a value function $V^\pi(s)$ can be computed that computes the expected return that will be obtained within the environment by following the policy π .

马尔可夫决策过程 (MDPs) 是一种用于顺序决策的框架，广泛应用于金融、机器人学、运筹学以及许多其他领域 [14]。MDPs 被构建为一个元组 (s_t, a_t, r_t, t) ，其中 $s_t \in S$ 是在给定时间的状态，$t, a_t \in A$ 是代理在时间 t 由于决策过程而采取的行动，r_t 是代理因采取行动 a_t 从 s_t 到达 s_{t+1} 而获得的奖励，$T(s_t, a, s_{t+1})$ 是一个转移函数，描述环境的动态并捕捉给定行动 a_t 从状态 s_t 转移到状态 s_{t+1} 的概率 $p(s_{t+1} \mid s_t, a_t)$ 。一个策略 π 可以定义为将每个状态 $s \in S$ 映射到一个行动 $a \in A$ 。从给定的策略 $\pi \in \Pi$ 可以计算出一个价值函数 $V^\pi(s)$ ，该函数计算在遵循策略 π 的情况下在环境中将获得的预期回报。

The solution of an MDP is termed the optimal policy π^* , which defines the optimal action $a^* \in A$ that can be taken from each state $s \in S$ to maximize the expected return. From this optimal policy π^* the optimal value function $V^*(s)$ can be computed which describes the maximum expected value that can be obtained from each state $s \in S$. And from the optimal value function $V^*(s)$, the optimal policy π^* can also easily be recovered.

MDP 的解决方案被称为最优策略 π^* ，它定义了从每个状态 $s \in S$ 可以采取的最优行动 $a^* \in A$ ，以最大化预期回报。从最优策略 π^* 可以计算出最优价值函数 $V^*(s)$ ，该函数描述了从每个状态 $s \in S$ 可以获得的最大预期价值。并且从最优价值函数 $V^*(s)$ 也可以很容易地恢复出最优策略 π^* 。

The same state space and action space as defined in [13] are used here. The reward function however is very different.

这里使用了与 [13] 中定义的相同的状态空间和动作空间。然而，奖励函数是非常不同的。

C. Reward Function

C. 奖励函数

The aircraft receives a different reward function depending on whether it is on approach, following a ring, or on final approach. In general, each aircraft receives a unique reward function. The reward function is designed to attract the aircraft to its goal(s) but treats all other aircraft and obstacles as risks that should be avoided.

飞机根据其是处于进近、跟随环状路径还是最后进近阶段，接收不同的奖励函数。通常，每架飞机都会接收到一个独特的奖励函数。奖励函数旨在吸引飞机朝向其目标，但将所有其他飞机和障碍物视为应避免的风险。

We model this as a simple state machine for each aircraft as shown in Figure 3. Each state in the state machine will result in a different reward function (discussed below). First we define a set of primitives that will help describe the rewards that are define for each state in the state machine.

我们将此建模为如图 3 所示的每架飞机的简单状态机。状态机中的每个状态将导致不同的奖励函数(如下讨论)。首先，我们定义一组原语，以帮助描述状态机中每个状态定义的奖励。

1. Reward Primitives

1. 奖励原语

We model the goal(s) as a positive reward of 100 . We allow the goal number and locations to vary over time, and each goal is modeled as a separate positive reward. We use a discount factor of 0.999 for the MDP. This provides a strong attraction to the goal globally over the state space.

我们将目标建模为 100 的正奖励。我们允许目标数量和位置随时间变化，每个目标都被建模为单独的正奖励。我们对 MDP 使用 0.999 的折扣因子。这在全球状态空间上对目标产生了强烈的吸引力。

To model the risk of colliding with another aircraft, we define a ”risk well” as a negative reward of -1000 which decays at a rate of 0.995 for up to 2000 meters from the center of the well. For every aircraft, we place a risk well at the location the aircraft will be in 5 seconds, where it currently as at, and where it was 5 seconds in the past. This assumes that the other aircraft will maintain a constant heading and velocity and is used as a way to model the risk over the next 5 seconds.

为了模拟与其他飞机碰撞的风险，我们定义了一个“风险井”，作为-1000 的负奖励，其衰减率为 0.995，距离井中心 2000 米以内。对于每架飞机，我们在飞机将在 5 秒后到达的位置、当前所在位置以及 5 秒前所在位置放置一个风险井。这假设其他飞机将保持恒定的航向和速度，并用作在接下来的 5 秒内模拟风险的一种方式。

When an aircraft is assigned to a ring, to discourage it from crossing the center region of the ring we define ”terminal well” as a negative reward of -1000 at the center of the ring with a radius of the ring’s radius. Thus if the aircraft strays inside the radius of the ring, it will receive a negative penalty and will desire to stay outside of the ring’s radius normally. Note that in the event that another aircraft is too close, the collision avoidance penalty will override the penalty of straying inside the ring. The ring’s radius is considered a soft constraint and collision avoidance is considered a hard constraint.

当一架飞机被分配到一个环内时，为了阻止它穿越环的中心区域，我们定义“终端井”为在环中心的半径等于环半径的区域内给予-1000 的负奖励。因此，如果飞机偏离进入环的半径内，它将会接收到一个负惩罚，并通常会希望停留在环的半径之外。注意，在另一架飞机过于接近的情况下，避障惩罚将覆盖飞机偏离环内的惩罚。环的半径被视为软约束，而避障被视为硬约束。

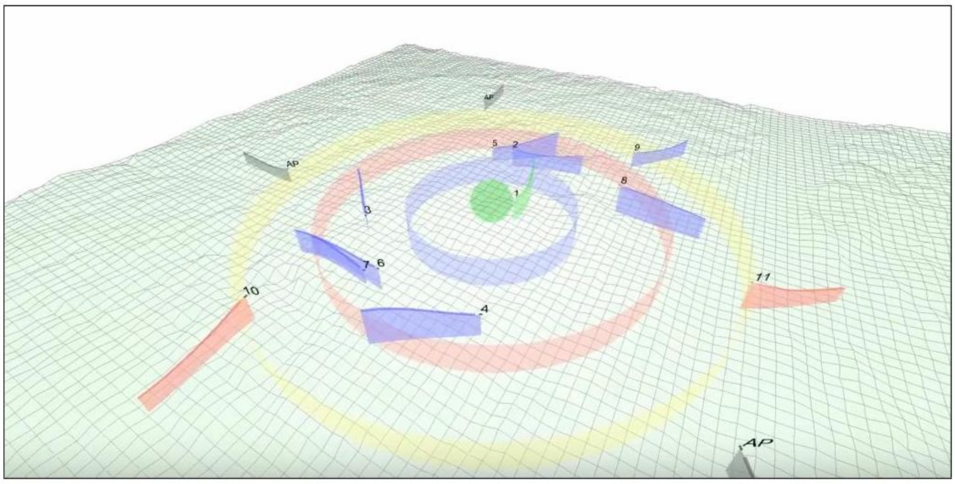

Fig. 4 Screenshot of simulation showing a vertiport terminal area with two rings. Aircraft without numbers assigned are in the approach state; aircraft with numbers have been sequenced. Aircraft are color coded to match the ring that they are assigned to. Gray aircraft have not yet crossed the approach threshold colored yellow. Green aircraft are on final approach to the vertiport.

图 4 模拟截图显示了一个带有两个环的垂直起降场终端区域。未分配编号的飞机处于进近状态；带有编号的飞机已经被排序。飞机的颜色编码与它们被分配到的环相匹配。灰色的飞机尚未飞越黄色的进近阈值。绿色的飞机正处于对垂直起降场的最后进近阶段。

When an aircraft is assigned to a ring, we define a "future bearing" reward as a positive reward that is placed at a bearing that the aircraft will reach in the future if it were to follow the ring's direction (e.g., counter clockwise.) For example, if an aircraft is currently at a bearing of 180 degrees (due south) of the vertiport, its future bearing will be some number of degrees α around the ring (perhaps at 140 degrees, south/south-east). As the aircraft moves around the circle changing its bearing with respect to the vertiport, the future bearing is also moving around the circle at the same rate. We model this as a positive reward of 100 placed on the circumference of the ring at the future bearing. This results in very smooth motion and is a simple and natural way to express the desired behavior for our aircraft. Note that the bearing and future bearing can be determined by the aircraft during flight and are not generated by the VTAC.

当一架飞机被分配到一个环时，我们定义了一个“未来方位角”奖励，这是一种正向奖励，放置在飞机如果沿着环的方向 (例如，逆时针) 前进将会在未来达到的方位角上。例如，如果一架飞机当前位于航站楼的 180 度方位角 (正南方向)，它的未来方位角将是环上某个度数 α (可能在 140 度，即南/东南方向)。随着飞机绕圆移动，改变其相对于航站楼的方位角，未来方位角也以相同的速度绕圆移动。我们将其建模为在环的圆周上放置的 100 点正向奖励，位于未来方位角处。这导致了非常平滑的运动，并且是一种简单而自然的方式来表达我们飞机的期望行为。请注意，飞机在飞行过程中可以确定方位角和未来方位角，它们不是由 VTAC 生成的。

Next we describe how these reward primitives are used in each state to build the desired behavior in the airspace.

接下来，我们描述在每个状态下如何使用这些奖励原语来构建空域中的期望行为。

2. Rewards per state in state machine

2. 状态机中每个状态下的奖励

Depending on the state that the aircraft is in in Figure 3, the rewards that are provided to the MDP differ as follows: 1) When in the "approach" state:

根据图 3 中飞机所处的状态，提供给 MDP 的奖励有以下不同:1) 当处于“接近”状态时:

- A terminal well is placed at the vertiport so that the aircraft will avoid the airspace around the vertiport.
- 在航站楼放置一个终端井，以便飞机避开航站楼周围的空域。
- A positive reward is placed at each approach gate. The MDP solution will naturally draw the aircraft to the nearest goal.
- 在每个接近门处放置一个正向奖励。MDP 解决方案将自然地引导飞机飞向最近的目标。
- Every other aircraft is modeled with risk wells to avoid collision with other aircraft.
- 其他所有飞机都被建模为风险井，以避免与其他飞机相撞。

2) When in the "ring following" state:
2) 当处于“跟随环”状态时:

- A terminal well is placed at the center of the ring with the same radius as the ring so that the aircraft will be discouraged from going within the ring's radius.
- 在环的中心放置一个与环半径相同的终端井，以便飞机不会进入环的半径范围内。
- A positive reward is placed at the future bearing of the aircraft. The MDP solution will naturally draw the aircraft to the future bearing location.

在飞机的未来航向位置放置一个正奖励。MDP 解决方案将自然引导飞机前往未来航向位置。

- Every other aircraft is modeled with risk wells to avoid collision with other aircraft.

其他所有飞机都使用风险井模型来避免与其他飞机相撞。
3) When in the ”final” state:
3) 当处于“最终”状态时:

- A goal is placed at the vertiport so that the aircraft will be attracted to the vertiport VTOL zone.

在垂直起降场放置一个目标，以便吸引飞机前往垂直起降场的 VTOL 区域。

- Every other aircraft is modeled with risk wells to avoid collision with other aircraft.

其他所有飞机都使用风险井模型来避免与其他飞机相撞。
4) When the aircraft reaches the VTOL zone it is removed from the simulation and is considered in the ”on ground”
4) 当飞机到达 VTOL 区域时，它将从模拟中移除，并被认为是“在地面上”

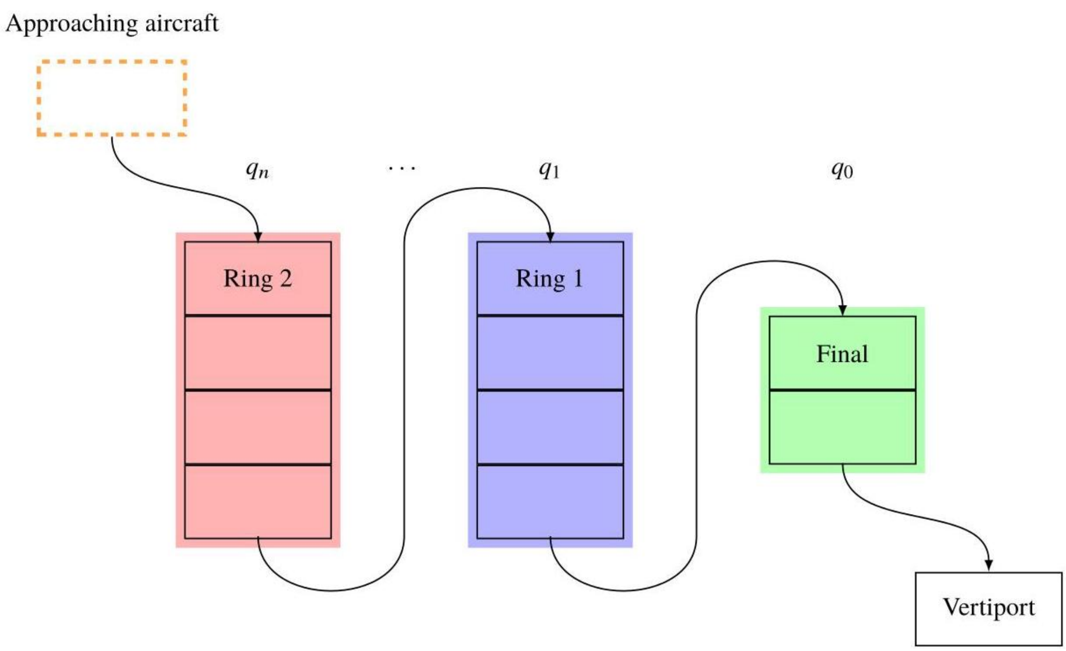

Fig. 5 Depiction of aircraft flow through VTAC queues. Colors are chosen to be consistent with the ring colors from other diagrams.
图 5 展示了飞机通过 VTAC 队列的流程。颜色选择与其它图中的环形颜色保持一致。
state. Any time taken to descend vertically to the airport is not considered in this simulation. In reality, some additional delay will be incurred by any ground service required to offload passengers, refuel/recharge, and load new passengers.
状态。在此模拟中不考虑飞机垂直下降至机场所花费的时间。实际上，任何地面服务所需的卸载乘客、加油/充电和装载新乘客都会产生一些额外的延误。

D. Sequencing

D. 排序

As an aircraft transitions from the approach state to the final state, it is sequenced. Aircraft are sequenced in order that they cross the approach threshold. To implement sequencing, once an aircraft is sequenced we maintain a series of queues of all aircraft under VTAC control. When capacity is available at the vertiport, the aircraft with lowest sequence number is selected first.

当飞机从进近状态过渡到最终状态时，它将被排序。飞机按照它们穿过进近阈值的顺序进行排序。为了实施排序，一旦飞机被排序，我们维护一系列受 VTAC 控制的飞机队列。当垂直起降场有空余容量时，选择序列号最低的飞机优先。

While this method should be able to support other sequencing and prioritization schemes such as prioritizing landing for aircraft with the least amount of fuel/charge, this could potentially cause issues with aircraft moving through multiple rings and is left for future study.

虽然这种方法应该能够支持其他排序和优先级方案，例如优先让燃料/电量最少的飞机着陆，但这可能会在飞机穿越多个环形区域时引起问题，因此留待未来研究。

E. Ring Management

E. 环管理

The VTAC maintains a queue for each ring $\{q_1, \cdots, q_n\}$ and a queue for the final approach to the vertiport q_0 , will all queues referred to as $Q = \{q_0, q_1, \cdots, q_n\}$. Each queue is a first-in-first-out (FIFO) queue and has a fixed capacity. As aircraft cross the approach threshold, they are added to the outermost queue. At each update cycle, the vertiport attempts to move aircraft from outer rings to inner rings and from the innermost ring into the final queue. The queues are arranged as shown in Figure 5.

VTAC 为每个环维护一个队列 $\{q_1, \cdots, q_n\}$ 以及一个通往垂起降场的最终接近队列 q_0 ，所有队列统称为 $Q = \{q_0, q_1, \cdots, q_n\}$ 。每个队列都是先进先出 (FIFO) 队列，并且具有固定的容量。当飞机越过接近阈值时，它们会被添加到最外层的队列。在每个更新周期，垂起降场尝试将飞机从外环移动到内环，并将最内环的飞机移入最终队列。队列的排列如图 5 所示。

F. Algorithm

F. 算法

The algorithm for the simulation for managing the vertiport is shown in Algorithm 1. We demonstrate this planner in a 3D aircraft simulation showing a perspective view of the aircraft, the rings, the approach threshold, and the vertiport as shown in Figure 4. The simulation covers a 25 km by 25 km by 25 km volume which contains a configurable number of aircraft and gates. The location of the gates are chosen to lie at regular intervals along a circle surrounding the vertiport.

管理垂起降场的模拟算法如算法 1 所示。我们在一个 3D 飞机模拟中展示了这个规划器，显示了飞机、环、接近阈值和垂起降场的透视图，如图 4 所示。模拟覆盖了一个 25 km 乘以 25 km 乘以 25 km 的体积，其中包含可配置数量的飞机和门。门的位置被选择在围绕垂起降场的圆周上以规则间隔排列。

Algorithm 1 Vertiport simulation

算法 1 垂起降场模拟

procedure VERTIPORTSIM

过程 VERTIPORTSIM

$\mathbf{S}_t \leftarrow$ randomized initial aircraft states

$\mathbf{S}_t \leftarrow$ 随机初始化飞机状态

$\mathbf{A} \leftarrow$ aircraft actions (precomputed)

$\mathbf{A} \leftarrow$ 飞机动作 (预计算)

$\mathbf{L} \leftarrow$ aircraft limits (precomputed)

$\mathbf{L} \leftarrow$ 飞机限制 (预计算)

$\mathbf{S}_{t+1} \leftarrow$ allocated space

$\mathbf{S}_{t+1} \leftarrow$ 分配的空间

while aircraft remain do

当仍有飞机时

// Update the vertiport queues

// 更新垂直港口队列

for $q_i \in Q, \forall i = \{0, \cdots, n\}$ do

对于 $q_i \in Q, \forall i = \{0, \cdots, n\}$ 执行

while not q_i .full do

当 q_i 不满时循环

// Look for aircraft in outer rings to pull inward

// 在外环寻找飞机拉入内环
for $q_j \in Q, \forall j=\{i+1, \cdots, n\}$ do
对于 $q_j \in Q, \forall j=\{i+1, \cdots, n\}$ 执行
if not q_j .empty then
如果 q_j 不为空则
$q_t \leftarrow q_j$ .remove( )
$q_t \leftarrow q_j$ .remove( )
q_i .add (q_t)
q_i . 添加 (q_t)
break out of for q_j loop
跳出 for q_j 循环
// Aircraft states may have changed due to queue processing above
// 由于上述队列处理，飞机状态可能已发生变化
for each aircraft do
对每架飞机执行
$s_t \leftarrow \mathbf{S}_t$ [aircraft]
$s_t \leftarrow \mathbf{S}_t$ [航空器]
// Build peaks based off the flight phase
// 根据飞行阶段构建峰值
if approach phase then
如果是进近阶段那么
$\mathbf{P}^{+} \leftarrow$ build pos rewards from arrival gates
$\mathbf{P}^{+} \leftarrow$ 从到达门构建正奖励
$\mathbf{P}^{-} \leftarrow$ build neg rewards in Standard Positive Form to avoid collisions
$\mathbf{P}^{-} \leftarrow$ 以标准正形式构建负奖励以避免碰撞
else if ring following phase then
否则如果是在环跟随阶段那么
$\mathbf{P}^{+} \leftarrow$ build pos rewards from future bearing
$\mathbf{P}^{+} \leftarrow$ 从未来方位构建正奖励
$\mathbf{P}^{-} \leftarrow$ build neg rewards in Standard Positive Form to avoid collision and terminal well
$\mathbf{P}^{-} \leftarrow$ 以标准正形式构建负奖励以避免碰撞和终端井
else if final approach phase then
否则如果是在最后进近阶段那么
$\mathbf{P}^{+} \leftarrow$ build pos rewards at vertiport
$\mathbf{P}^{+} \leftarrow$ 在垂直机场构建正奖励
$\mathbf{P}^{-} \leftarrow$ build neg rewards in Standard Positive Form to avoid collision
$\mathbf{P}^{-} \leftarrow$ 以标准正形式构建负奖励以避免碰撞
// Use the FastMDP algorithm to solve for the next action and determine the next state
// 使用 FastMDP 算法来求解下一个动作并确定下一个状态
$s_{t+1} = \text{FastMDP}\left(s_t, \mathbf{A}, \mathbf{L}, \mathbf{P}^{+}, \mathbf{P}^{-}\right)$
$\mathbf{S}_{t+1}$ [aircraft] $\leftarrow s_{t+1}$
// Now that all aircraft have selected an action, apply it
// 现在所有飞机都已选择了一个动作，执行它
$\mathbf{S}_t \leftarrow \mathbf{S}_{t+1}$

Simulation begins with all aircraft spawned randomly around the terminal area. Each aircraft creates and solves its own MDP at each timestep, and then uses the solution of the MDP to select its next action. The actions of all aircraft are performed simultaneously in the simulation at the beginning of the next time step, simulation then advances by one time step (0.1 seconds), and the vertiport terminal area controller (VTAC) state machine is updated. During the simulation, we track the number of near mid-air collisions (NMACs) that occur.

模拟从所有飞机在终端区域随机生成开始。每架飞机在每个时间步长创建并解决自己的 MDP，然后使用 MDP 的解决方案来选择其下一个动作。在模拟中，所有飞机的动作在下一个时间步长开始时同时执行，模拟随后前进一个时间步长 (0.1 秒)，并且垂直起降场终端区域控制器 (VTAC) 的状态机更新。在模拟过程中，我们跟踪发生的接近中空碰撞 (NMACs) 的数量。

IV. Results

IV. 结果

Simulations were run for two environments:

对两种环境进行了模拟:

1) 2 rings, 15 aircraft

1) 2 个环，15 架飞机

- vertiport capacity 1
- 垂直起降场容量 1
- ring 1 capacity 2 with radius 2000
- 环 1 容量 2，半径 2000
- ring 2 capacity 6 with radius 4000
- 环 2 容量 6，半径 4000
- approach threshold radius of 5500
- 接近阈值半径为 5500

2) 4 rings, 45 aircraft

2) 4 个环，45 架飞机

- vertiport capacity 1
- 垂直机场容量 1
- ring 1 capacity 8 with radius 2000
- 环形区域 1 容量 8，半径 2000
- ring 2 capacity 12 with radius 3500
- 环形区域 2 容量 12，半径 3500
- ring 3 capacity 15 with radius 5000
- 环形区域 3 容量 15，半径 5000
- ring 4 capacity 20 with radius 6500
- 环形区域 4 容量 20，半径 6500

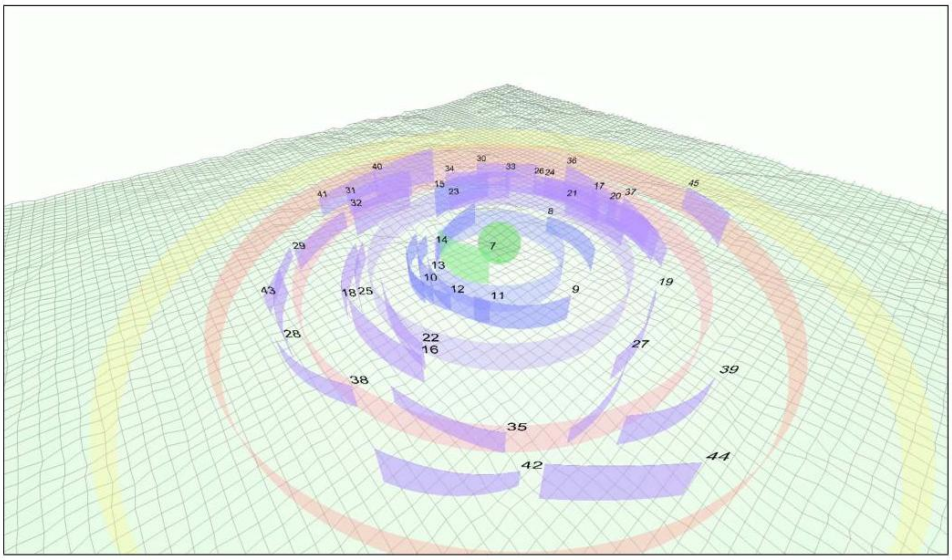

Fig. 6 Screenshot of simulation showing a vertiport terminal area with four rings and 45 aircraft.
图 6 模拟屏幕截图，显示带有四个环形区域和 45 架飞机的垂直机场终端区域。

- approach threshold radius of 7500
- 接近阈值半径为 7500

The aircraft positions were randomly initialized to lie in uniformly distributed in a region around the vertiport and were initialized with random initial headings. The number of aircraft were chosen to be less than the total capacity of the VTAC capacity C .

飞机位置被随机初始化在垂直机场周围的均匀分布区域内，并初始化为随机初始航向。飞机的数量被选择为小于 VTAC 容量的总数 C 。

Simulation results demonstrate that the algorithm is able to successfully sequence aircraft without collisions. An example video with 15 aircraft and 2 rings is available at the URL below which demonstrates the airspace design, state machine, and priority sequencing. The aircraft currently successfully navigate to the arrival gates, are sequenced by the VTAC, and then follow the ring based airspace design while approaching the vertiport. Over 10 runs of this setup with randomized initial conditions 0 NMACs occurred and 0 collisions occurred.

模拟结果显示算法能够成功地对飞机进行排序，避免碰撞。以下 URL 提供了一个包含 15 架飞机和 2 个环形区域的示例视频，展示了空域设计、状态机和优先级排序。飞机目前成功导航至到达门，由 VTAC 进行排序，然后在接近垂直机场时遵循基于环形的空域设计。在这种设置下进行的超过 10 次随机初始条件的运行中，0 次发生了接近冲突 (NMACs) 和 0 次碰撞。

Video: https://youtu.be/J2CNj5VsHQ4

视频:https://youtu.be/J2CNj5VsHQ4

Figure 6 shows a screen shot of a simulation with 4 rings and 45 aircraft with a video available at the following URL. This example shows the difficulty of sequencing large numbers of aircraft with increased density. Despite the complexity in a single run 9 NMACs occurred and 0 collisions occurred. The video is rendered with additional indicators showning the intent of each aircraft. Green lines extend from each aircraft to its goal and help understand the decision making that occurs.

图 6 显示了一个包含 4 个环形区域和 45 架飞机的模拟屏幕截图，以下 URL 提供了视频。这个示例显示了在增加密度的情况下对大量飞机进行排序的难度。尽管在单次运行中复杂性增加，但发生了 9 次接近冲突 (NMACs) 和 0 次碰撞。视频以附加指示器的形式渲染，显示了每架飞机的意图。绿色线条从每架飞机延伸至其目标，有助于理解决策过程。

Note that the 9 NMACs that occur appear to be caused when aircraft transition from an outer ring to an inner ring. Sometimes this results in what appears to be an undesirable density resulting from excessive closure rate to the ring. Once the closure happens, the rate at which the aircraft separate also appears to be too low. This needs to be investigated in future work to see if a change to the algorithm can better manage this issue.

注意，发生的 9 次 NMAC 似乎是在飞机从外环过渡到内环时引起的。有时这会导致由于过快的接近速率而出现看似不希望的密度。一旦接近发生，飞机分离的速率也似乎过低。这需要在未来的工作中进行调查，以查看算法的更改是否可以更好地管理这个问题。

Video: https://youtu.be/P8R_JaB5pdI

视频:https://youtu.be/P8R_JaB5pdI

V. Conclusion

V. 结论

In this paper we have demonstrated an airspace design concept to handle an uncertain number of UAM aircraft approaching a vertiport. The airspace design uses a concentric ring pattern along with queues to sequence the aircraft. The computational burden is split between the vertiport which manages aircraft's assignment to rings and to the final queue and the aircraft which manage their own motion around the rings while avoiding collisions. The result is a hybrid centralized / distributed algorithm in which the aircraft self-organize with little direct coordination required of the vertiport's central controller.

在本文中，我们展示了一种处理不确定数量的城市空中交通 (UAM) 飞机接近垂直机场的空域设计概念。该空域设计使用同心环模式以及队列来安排飞机的顺序。计算负担在垂直机场 (负责管理飞机分配到环和最终队列) 和飞机 (在避免碰撞的同时管理自己在环上的运动) 之间分配。结果是混合集中式/分布式算法，在该算法中，飞机自我组织，而垂直机场的中央控制器需要直接协调的很少。

In the future, one way to improve the algorithm will be to alter the first-come-first-served nature of the sequencing to a priority based scheme sorted by a metric such as minutes of fuel/charge remaining. This should improve the safety margin by allowing aircraft which may run out of fuel to land earlier. However, this will introduce additional congestion as a high priority aircraft move towards the inner rings through existing traffic. This may be solved with altitude differences between rings or by introducing speed control.

在未来，改进算法的一种方法将是改变序列的先来先服务性质，转而采用基于优先级的方案，该方案按照例如剩余燃料/充电时间的指标排序。这应该通过允许可能耗尽燃料的飞机早点着陆来提高安全边际。然而，这将会引入额外的拥堵，因为高优先级飞机穿过现有交通向内环移动。这可以通过环之间的海拔差异或引入速度控制来解决。

Additionally, future work may allow for an ability to add new rings dynamically. The code actually currently supports this, but it was found that by expanding the number of rings and thus the approach threshold, some aircraft may be directed to reverse course resulting in unnecessary conflicts as aircraft caught within this newly created ring attempt to sort themselves out. It was decided for this paper to leave this case out and explore at a later time.

此外，未来的工作可能允许动态添加新的环。实际上，代码目前支持这一点，但发现通过增加环的数量从而提高接近阈值，一些飞机可能被指示掉头，导致在新的创建环内被困的飞机尝试整理自己时出现不必要的冲突。本文决定将这种情况留待以后探讨。

Also in general separation could be improved if speed control were part of the guidance provided to each aircraft. Currently all aircraft move at the same max speed in the simulation. Improved separation could result if aircraft close to each other also received additional guidance to change speeds to achieve a better distribution around the ring. It is unclear if this functionality would be better as a centralized vertiport function or if an altered reward structure would allow this behavior to emerge.

通常情况下，如果速度控制成为向每架飞机提供的引导的一部分，分离度可以得到改善。目前，在模拟中所有飞机都以相同的最大速度移动。如果接近彼此的飞机也接收到额外的指导以改变速度，以实现在环周围的更好分布，那么分离度可以得到提高。尚不清楚这种功能作为集中式垂直起降场功能是否会更好，或者改变奖励结构是否能够使这种行为出现。

References

参考文献

[1] Gipson, L., "NASA Embraces Urban Air Mobility, Calls for Market Study," https://www.nasa.gov/aero/nasa-embraces-urban-air-mobility, 2017. Accessed: 2018-01-19.

[2] "Uber Elevate | The Future of Urban Air Transport," https://www.uber.com/info/elevate/, 2017. Accessed: 2018-08-13.

[3] Holden, J., and Goel, N., "Fast-Forwarding to a Future of On-Demand Urban Air Transportation," San Francisco, CA, 2016.

[4] "Urban Air Mobility," http://publicaffairs.airbus.com/default/public-affairs/int/en/our-topics/ Urban-Air-Mobility.html, 2018. Accessed: 2018-08-13.

[5] "Future of Urban Mobility," http://www.airbus.com/newsroom/news/en/2016/12/My-Kind-Of-Flyover.html, 2017. Accessed: 2018-08-13.

[6] Kleinbekman, I. C., Mitici, M. A., and Wei, P., "eVTOL Arrival Sequencing and Scheduling for On-Demand Urban Air Mobility," 2018 IEEE/AIAA 37th Digital Avionics Systems Conference (DASC), IEEE, 2018, pp. 1-7.

[7] Pradeep, P., and Wei, P., "Energy Optimal Speed Profile for Arrival of Tandem Tilt-Wing eVTOL Aircraft with RTA Constraint," IEEE CSAA Guidance, Navigation and Control Conference, 2018.

[8] Bosson, C., and Lauderdale, T. A., "Simulation evaluations of an autonomous urban air mobility network management and separation service," 2018 Aviation Technology, Integration, and Operations Conference, 2018, p. 3365.

[9] Holden, J., and Goel, N., "Fast-forwarding to a future of on-demand urban air transportation," San Francisco, CA, 2016.

[10] Wei, P., Surakitbanharn, C., Landry, S., and Sun, D., ”Workload evaluation of sectorized air traffic control and stream management,” 2012 Integrated Communications, Navigation and Surveillance Conference, IEEE, 2012, pp. N2-1.

[11] FAA, ”Optimization of Airspace and Procedures in the Metroplex (OAPM) Study Team Final Report, North Texas Metroplex,” Proceedings of the 2011 Global PBN Summit, 2011.

[12] Bertram, J. R., Yang, X., Brittain, M., and Wei, P., ”Online Flight Planner with Dynamic Obstacles for Urban Air Mobility,” 2019 Aviation Technology, Integration, and Operations Conference, 2019.

[13] Bertram, J. R., and Wei, P., ”Distributed Computational Guidance for High-Density Urban Air Mobility with Cooperative and Non-Cooperative Collision Avoidance,” AIAA SciTech 60th Aerospace Sciences Meeting, 2020.

[14] Sutton, R. S., and Barto, A. G., Reinforcement learning: An introduction, Vol. 1, MIT press Cambridge, 1998.