arXiv:2303.00851v1 [cs.RO] 1 Mar 2023

自主竞速无人机集群的激进轨迹生成

Yuyang Shen[1], Jinming Xu[1], Jin Zhou[1],Danzhe Xu[2],Fangguo Zhao[3],Jiming Chen[1],和 Shuo Li[1]

摘要— 自主无人机竞速正逐渐成为挑战四旋翼飞行器自主技术的绝佳平台，涵盖规划、导航和控制技术。然而，该领域的大多数研究主要聚焦于单架无人机的场景。本文介绍一种新颖的时间最优轨迹生成方法，可为四旋翼集群生成时间最优轨迹，使其以最大机动性无碰撞地飞过预定义航点。我们在 Gazebo 仿真中验证了该方法：由5 架四旋翼飞行器组成的集群能够在 $35m \times 35m$ 空间内以最高速度 $14m/s$飞过一条包含6个航点的复杂竞速赛道。我们还在一个 $4m \times 2m$ 飞行场地中开展了两架四旋翼飞行器穿越3 个航点的飞行测试，以证明所提方法在真实世界中的可行性。仿真和真实世界的飞行测试均表明，所提方法能够为自主竞速无人机集群生成最优激进轨迹。该方法也可轻松迁移至其他类型的机器人集群。

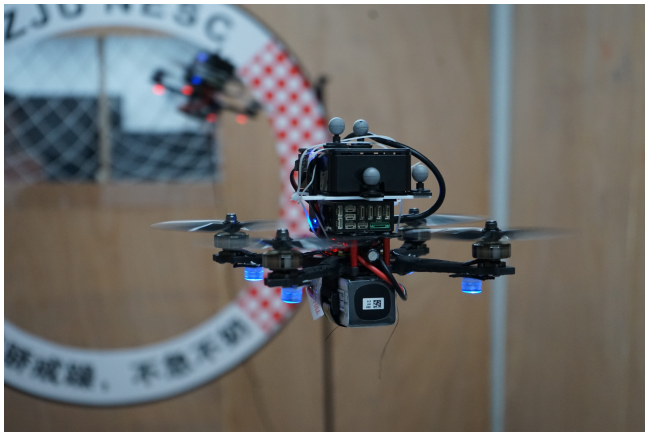


图1. 实验中使用的飞行平台穿过航点。

I. 引言

自主无人机竞速是挑战无人机自主激进飞行技术的绝佳平台，涉及环境感知、轨迹规划、状态估计与控制等。竞速规则相当简单：无人机必须完全自主地飞行，并按特定顺序穿过门框，最快者获胜。必须承认，尽管自主竞速无人机的速度已显著提升，但专业人类竞速飞行员与最先进的自主无人机竞速技术之间仍存在巨大差距。其中一项挑战，是如何为单架竞速无人机或由多架竞速无人机组成的集群，在复杂竞速赛道上进行时间最优轨迹生成。

2016 年以前，尚无关于自主无人机竞速的研究，但激进轨迹生成与控制技术已发展成熟。其中，基于微分平坦性理论的方法在生成四旋翼飞行器穿越预定义航点的敏捷轨迹方面展现出优势 [1], [2]。时至今日，这些方法仍是最常用的四旋翼飞行器轨迹生成方法之一。尽管这些方法能够生成敏捷轨迹，但由于其多项式特性，生成的轨迹并非时间最优，因此竞速无人机无法以极限速度飞行。自主无人机竞速的一个里程碑是 IROS 2016 自主无人机竞速旨在为研究团队和业余爱好者团队提供交流机会，让他们展示并比拼最新技术，最终战胜人类飞手 [3]。从此以后，IROS 自主无人机竞速成为一年一度的赛事，一直举办到 2019 年。在这一早期阶段，自主竞速无人机的速度逐年提升，但仍远不及人类飞手。例如，获胜者的速度在 $0.6m/s$ 的比赛中约为 2016 [4] ，在 2017 [5]的比赛中约为 $0.7m/s$ 。代尔夫特理工大学的 MAVLab 让一架 Bebop 四旋翼无人机在复杂的地下室环境中飞越一条设有五个门的竞速赛道，最高速度达到 $1.7m/s$ ，时间为 2018 [6]。之后，他们研发了一架重 72 克的自主竞速无人机，能够以最高 $2.6m/s$ [7]的速度飞过一条简单的四门赛道。IROS 2019 自主无人机竞速的冠军——来自苏黎世大学的机器人与感知组（RPG）——将飞行速度提升至 $2.5m/s$ [8]。一个重要的转折点是 2019 年人工智能机器人竞速比赛（又称 AlphaPilot 挑战赛）：冠军代尔夫特理工大学的 MAVLab 达到了 $9.2m/s$ [9] 的最高速度，而亚军 RPG 达到了 $8m/s$ [10]的最高速度。遗憾的是，根据报告，自主竞速无人机的速度虽已非常接近人类飞手，但仍未能超越他们。自主无人机竞速的另一个里程碑是 Scaramuzza 在 2021 年的研究：他们开发了一种名为 CPC 的新型轨迹生成方法，能够生成真正的时间最优轨迹，并引导四旋翼飞行器以 $19m/s$ [11]的最高速度飞行。他们的方法最终在一场无人机竞速

[1]作者来自中国杭州310027浙江大学控制科学与工程学院 shuo.li@zju.edu.cn[2]Danzhe Xu来自中国杭州310023浙江工业大学自动化系。[3]Fangguo Zhao来自中国西安710072西北工业大学自动化学院

任务。

从上述竞赛和研究来看，只有一架无人机飞过竞速赛道。以第一人称视角（FPV）无人机竞速为例，人类飞手可以同时与多达 5 名对手竞赛 [12]。因此，如何让多架自主竞速无人机同时参赛，是机器人学界一个非常有趣且具有挑战性的课题。据作者所知，目前尚无针对自主竞速无人机集群生成时间最优轨迹的研究。目前，相关研究主要聚焦于四旋翼无人机集群，例如编队飞行。大多数研究采用优化多项式来引导四旋翼飞行器，同时避免彼此碰撞，但无法保证所生成轨迹的最优性 [13], [14]。另一些研究则采用分布式方法来完成一些复杂任务。例如，在 [15], 中，研究人员使用微型四旋翼无人机集群在未知环境中执行搜救任务；Duisterhof等人则使用四旋翼集群在杂乱环境中搜寻气体泄漏 [16]。Zhou等人实现了四旋翼集群以 $3m/s$ [17]的速度飞过一片竹林。然而，“时间最优”并非这些方案的目标，而且四旋翼飞行器的飞行速度仍远未达到其飞行包线的边界。

因此，在本文中，我们

1）将 CPC 扩展至自主竞速无人机集群，使四旋翼飞行器凭借极强的机动性，完全自主地飞越竞速赛道，并以最短飞行时间抵达目标点。2）重新表述竞速无人机集群的优化问题，使得在生成时间最优的激进轨迹时能够施加避碰约束。3）在仿真中验证并分析我们的方法，其中 5 架四旋翼飞行器以 $14m/s$ 的最大速度飞越 6 个航点；此外，我们还通过真实世界飞行测试展示所提方法的可行性。

II. 方法

A. 四旋翼无人机模型

本文采用的四旋翼无人机动力学模型与 [11] 中所述模型相同。为方便读者，我们在此列出该动力学模型。需要注意的是，我们在变量/状态左侧添加下标 i，以表示它们属于 i^{th} 四旋翼飞行器。

$$ {}_i\dot{\mathbf{x}} = \mathbf{f}_{dyn}({}_i\mathbf{x}, {}_i\mathbf{u}) = \begin{cases} {}_i\mathbf{v} \\ \mathbf{g} + \frac{1}{{}_im}\mathbf{R}({}_i\mathbf{q}){}_i\mathbf{T} \\ \frac{1}{2}\Lambda({}_i\mathbf{q}) \begin{bmatrix} 0 \\ {}_i\boldsymbol{\omega} \end{bmatrix} \\ {}_i\mathbf{J}^{-1}({}_i\boldsymbol{\tau} - {}_i\boldsymbol{\omega} \times {}_i\mathbf{J}\,{}_i\boldsymbol{\omega}) \end{cases} \tag{1} $$

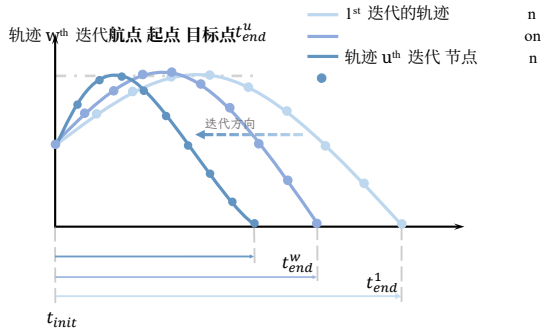
其中

$$ {}_i\mathbf{T} = \begin{bmatrix} 0 \\ 0 \\ \sum {}_iT_s \end{bmatrix} $$

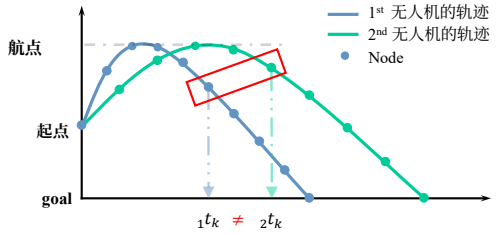
是四旋翼飞行器的推力向量，且

$$ {}_i\boldsymbol{\tau} = \begin{bmatrix} {}_il/\sqrt{2}({}_iT_1 + {}_iT_2 - {}_iT_3 - {}_iT_4) \\ {}_il/\sqrt{2}(-{}_iT_1 + {}_iT_2 + {}_iT_3 - {}_iT_4) \\ {}_ic_\tau({}_iT_1 - {}_iT_2 + {}_iT_3 - {}_iT_4) \end{bmatrix} $$

是四旋翼飞行器的力矩向量。在上面的方程中，${}_i\mathbf{v}$、${}_i\mathbf{q}$ 和 ${}_i\boldsymbol{\omega}$ 分别是 i^{th} 四旋翼飞行器的速度、四元数和角速度。$\mathbf{R}({}_i\mathbf{q})$是旋转矩阵。 ${}_il$ 是机臂长度。 ${}_iT_s$ 是旋翼的推力，它们是动力系统（1）的输入。



（a）在优化过程中，到达时间 t_{end} 缩短，因此各节点之间的时间间隔也随之缩短。

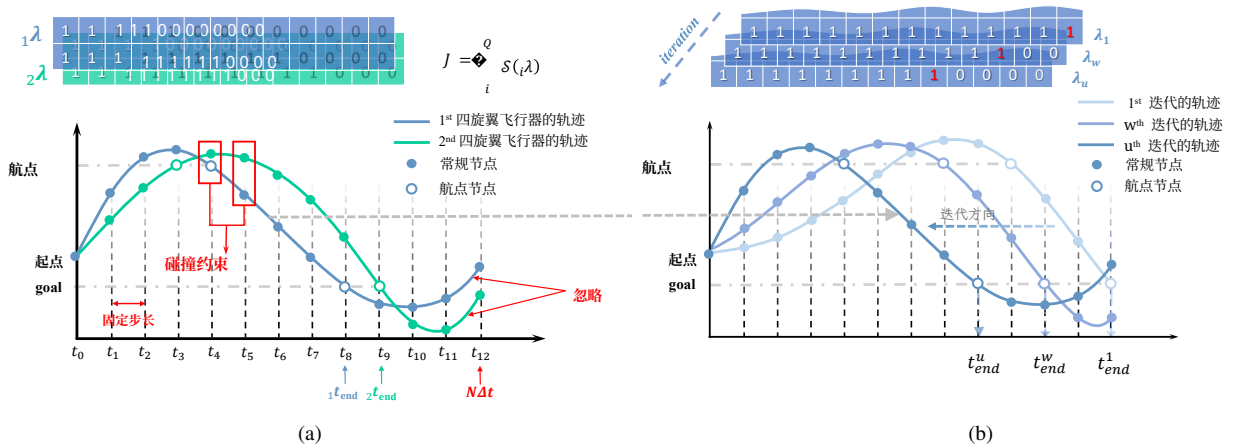


（b）在多无人机竞速场景中，不同的到达时间会导致对应节点的时间不匹配。

图 2. CPC 及其在多无人机竞速场景中的缺陷示意图。

B. 优化问题

本文旨在为自主竞速无人机集群生成时间最优轨迹，使它们在飞行过程中能够穿过规定的航点，同时相互避让，并以最短时间抵达目标点。在 CPC 中，他们在优化问题中加入进度度量变量矩阵 $\boldsymbol{\lambda}$ 和进度变化矩阵 $\boldsymbol{\mu}$，以控制航点的分配。具体而言，$\boldsymbol{\lambda}$ 用于确保四旋翼飞行器能够按特定顺序穿过预定义航点，而 $\boldsymbol{\mu}$ 用于确保轨迹上分配点与对应航点之间的距离在较小的容差范围内。在他们的方法中，节点数量 N



(a) (b)

图3。所提方法示意图。时间步长是固定的，因此可以为多个四旋翼飞行器添加碰撞约束。 ${}_i\boldsymbol{\lambda}$ 的右下方是空闲的。因此，四旋翼飞行器可以在这种固定时间步长策略下自由“选择”其到达节点。因此，优化目标是在满足动力学约束和碰撞约束等条件的同时，最小化 ${}_i\boldsymbol{\lambda}$ 之和。

是预先确定的，并强制四旋翼飞行器在最后一个节点抵达目标点。因此，在优化过程中，随着飞行时间 t_{end} 的缩短，各节点之间的时间步长 $\Delta t = (t_{end} - t_{init})/N$ 也会逐渐减小（图2(a)）。这一策略在单架竞速无人机场景中效果很好。然而，对于多架竞速无人机，由于节点数量已确定，难以在各四旋翼飞行器之间加入碰撞约束；当四旋翼飞行器的到达时间 ${}_it_{end}$不同时，对应节点的时间也不同（图2(b)）。

为了解决上述问题，我们首先固定时间步长 Δt，并定义相对较多的节点 N，使无人机应在节点 ${}_in_{end} < N$处到达目标点，这也意味着到达时间 ${}_it_{end} < N\Delta t$。图3(a)展示了两个竞速无人机从同一点出发，飞经一个航点并同时到达同一目标点的场景示意图。可以看出，由于 Δt 是固定的，四旋翼飞行器的节点是同步的。因此，可以将四旋翼飞行器在每个节点处的碰撞约束写为

$$\|\mathbf{E}({}_i\mathbf{P}_k - {}_r\mathbf{P}_k)\|_2^2 - \delta_{col} \geq 0 \qquad i \neq r \tag{2}$$

其中，${}_i\mathbf{P}_k$ 表示 i^{th} 四旋翼飞行器在时间 t_k的位置，$\delta_{col} > 0$ 是确保两架四旋翼飞行器不会相互碰撞的容差，而$\mathbf{E}$ 是用于降低下洗气流风险的矩阵 [18]。

与单架竞速无人机场景类似，每架四旋翼飞行器都有自己的 ${}_i\boldsymbol{\lambda}$ 矩阵和 ${}_i\boldsymbol{\mu}$，以确保四旋翼飞行器按预定义的顺序飞过航点，并确实到达指定节点的航点。因此，我们对进度变量 $\boldsymbol{\lambda}$ 和进度变化 $\boldsymbol{\mu}$ 施加与 CPC 相同的约束。

$$\begin{cases} {}_i\lambda_k^j \leq {}_i\lambda_k^{j+1} \\ {}_i\lambda_{k+1}^j - {}_i\lambda_k^j + {}_i\mu_k^j = 0 \\ {}_i\mu_k^j(\|{}_i\mathbf{P}_k - {}_i\mathbf{P}^{wj}\|_2^2 - {}_i\nu_k^j) := 0 \end{cases} \tag{3}$$

其中，${}_i\lambda_k^j$ 是一个布尔变量，表示 i^{th} 四旋翼飞行器是否在时间 t_k 经过 j^{th} 航点。${}_i\mu_k^j$ 表示 i^{th} 四旋翼飞行器是否正在经过 j^{th} 航点。${}_i\nu_k^j$ 是一个容差松弛变量。运算符“:=”表示 NAND（与非）函数。

需要注意的是，在所提方法中，我们不要求四旋翼飞行器在最后一个节点到达目标点，这与 CPC 不同。这意味着只要可行，四旋翼飞行器可以在任意时间步到达目标点。图3(a)展示了两架竞速无人机轨迹的一个示例。可以看到，第一架四旋翼飞行器在 ${}_1t_{end}$到达目标点。由于四旋翼飞行器已完成任务，因此 ${}_1t_{end}$ 之后的节点将被忽略。这样一来，优化目标实际上是在所有优化状态均满足约束（包括四旋翼飞行器的动力学约束和碰撞约束等）的前提下，尽可能将四旋翼飞行器的到达节点 ${}_in_{end}$向左推。图3(b)展示了一架四旋翼飞行器在优化过程中的 1^{st}、w^{th} 和 u^{th} $(1 < w < u)$ 次迭代示例。换言之，优化目标是最小化所有四旋翼飞行器的 ${}_i\boldsymbol{\lambda}$ 矩阵中*1* 的数量。我们定义一个算子 $\mathbb{S}(\bullet)$ 来计算矩阵中所有元素的总和，并将优化目标定义为:

$$\min_{\mathbf{X}} J = \sum_{i}^{Q} \mathbb{S}({}_i\boldsymbol{\lambda}) \tag{4}$$

其中，Q 是集群中四旋翼飞行器的数量。$\mathbf{X}$ 是由所有四旋翼飞行器的优化变量${}_i\mathbf{x}$ 组成的集合，而${}_i\mathbf{x} = [{}_i\mathbf{x}_0, {}_i\mathbf{x}_1, ..., {}_i\mathbf{x}_{N-1}]$ where

$${}_i\mathbf{x}_k = \left[{}_i\mathbf{p}_k \quad {}_i\mathbf{v}_k \quad {}_i\mathbf{q}_k \quad {}_i\boldsymbol{\omega}_k \quad {}_i\mathbf{u}_k \quad {}_i\boldsymbol{\lambda}_k \quad {}_i\boldsymbol{\mu}_k \quad {}_i\boldsymbol{\nu}_k\right]$$

是优化变量。通过求解满足动力学约束（1）、碰撞约束（2）、航路点约束（3）、输入约束和初始约束的优化问题（4），我们应该能够得到每架四旋翼飞行器经过优化的无碰撞轨迹。与 CPC 相比，为了同步四旋翼飞行器之间的节点，我们必须固定时间步长，然后估算一个相对较大的节点数 N，以确保四旋翼飞行器能在允许的时间内到达目标点，这会导致 ${}_it_{end}$ 之后的节点被浪费。因此，准确估算所需的节点数对于提高求解效率至关重要。

III. 仿真结果与分析

在本节中，我们分析优化结果，并在无人机竞速赛道中将所提方法与 CPC 进行比较，以展示其性能。此外，我们还在 Gazebo 环境中测试最优轨迹，以展示其在四旋翼飞行器上的运行效果。首先，我们在一个 $35m \times 35m$ 空间内设计一条竞速赛道作为飞行场地，并在该场地内设置 6 个航点。航点的位置列于表 I 中。

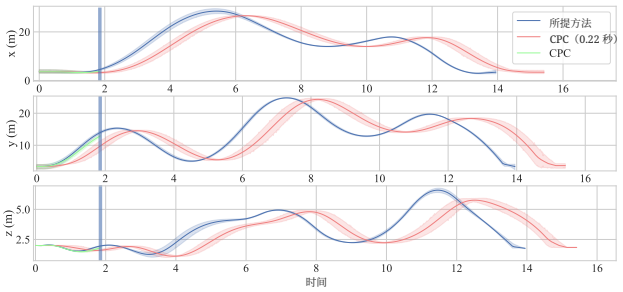


图 4. 针对所设计赛道（表 I），不同轨迹生成方法结果的比较。实线表示四旋翼飞行器的平均位置，阴影表示四旋翼飞行器的上下界。蓝色曲线为所提方法的优化结果。绿色曲线为采用基准方法独立生成的 5 架四旋翼飞行器的轨迹。需要注意的是，在没有避碰约束的情况下，5 架四旋翼飞行器会在约 $2s$ 处发生碰撞。红色曲线与绿色曲线类似，不同之处在于 5 架四旋翼飞行器以 $0.22s$ 的时间间隔依次起飞。

表 I 航点位置

航点编号	$x[m]$	$y[m]$	$z[m]$
1	5	15	2
2	25	5	3
3	20	25	5
4	14	14	2
5	18	18	6
6	5	14	4

我们通过求解优化问题 (4)，演示了 5 架竞速无人机飞越该赛道的过程。除表 I 所列的航点位置外，详细参数见表 II。

优化结果如图 4 所示（蓝色曲线）。为了使图表更加清晰，我们使用带阴影的曲线来

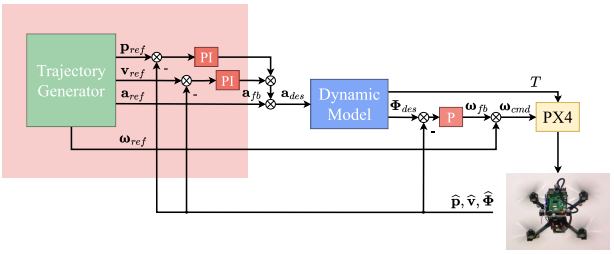


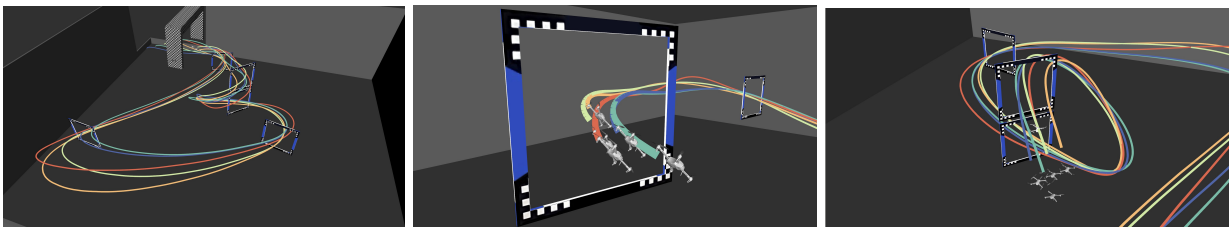
图 5. 轨迹跟踪的控制结构。前馈信号来自生成的最优轨迹，反馈信号用于修正偏差。它们通过 ROS 节点实现，低层速率环控制器为 PX4 控制器。二者通过 MAVLink协议进行通信。

表 II 仿真所用参数

参数	数值/范围	参数	数值/范围
N	550	${}_iT_s[N/s]$	$[-120, 120]$
$\Delta t[s]$	0.03	${}_i\theta, {}_i\phi[deg]$	$[-60, 60]$
$\delta_{col}[m]$	0.25	${}_i\psi[deg]$	$[-5, 5]$
${}_iT_s[N]$	$[1, 7.35]$	$\mathbf{E}$	$diag(1, 1, 1/3)$

表示四旋翼集群的位置，其中实线曲线表示四旋翼飞行器的平均位置，阴影边界表示四旋翼飞行器轨迹的上界和下界。据作者所知，目前尚无与本文主题相同的研究，即让多架四旋翼飞行器同时进行竞速。很难找到可与所提方法进行比较的基准方法。因此，我们使用 CPC 作为基准方法，以展示所提方法的性能；在该基准方法中，每架四旋翼飞行器独立规划自己的时间最优轨迹。不出所料，这些四旋翼飞行器会在某个时刻发生碰撞。结果如图 4 所示（绿色曲线）：它们在起飞后 $2s$ 发生碰撞。直接将 CPC 应用于多无人机竞速的一种直观方式，是让四旋翼飞行器以一定的时间间隔起飞。我们发现，$0.22s$ 的时间间隔可以确保这些四旋翼飞行器不会彼此碰撞，结果如图 4 所示（红色曲线）。可以看出，所提方法和具有 0.22 时间间隔的基准方法都能够引导四旋翼飞行器到达目标点，但所提方法的总到达时间比基准方法$0.9s$ 更短。需要指出的是，我们人为地为四旋翼飞行器添加了时间间隔，这有助于它们避免碰撞。然而，与简单的往返轨迹类似，该方法无法保证在所有场景下都不会发生碰撞。

随后，我们在 Gazebo 仿真器中测试这些轨迹，以展示轨迹跟踪的性能。我们使用常见的开源 PX4 进行底层控制。高层控制器以 ROS 节点的形式编写，并通过 ROS 话题与 PX4 通信。该控制器采用经典的前馈与反馈策略：生成的时间最优转速和推力作为前馈信号，而经典的双环 PID 控制器

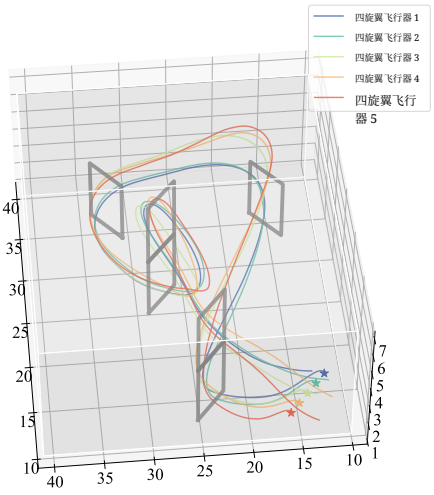


(a) (b) (c)

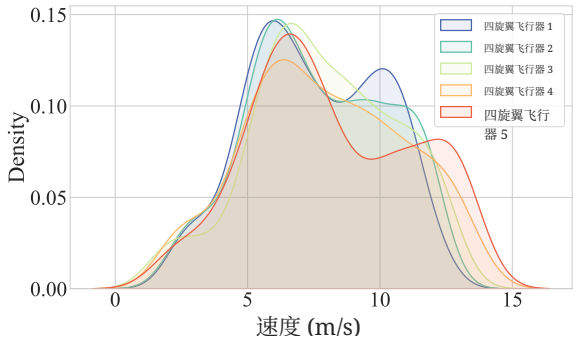
图 6. 5 架四旋翼飞行器在 Gazebo 仿真环境中竞速的图片。补充材料中提供了视频。

则作为反馈控制器来纠正偏差（图 5）。

Gazebo 中的仿真场景如图 6 所示。飞行场地是一个 $35m \times 35m$ 空间，航点（竞速门）部署在表 I 所列的位置。5 架四旋翼飞行器同时出发，并按照预定义的顺序无碰撞地穿过门框。图 7 展示了这 5 架四旋翼飞行器的轨迹及其速度分布。可以看出，在此场景中，最大飞行速度达到 $14m/s$。



(a) 5 架四旋翼飞行器同时出发，无碰撞地飞过各航点



(b) 四旋翼飞行器的速度分布。最大速度达到 $14m/s$

图 7. 仿真的飞行数据。



图 8. 两架四旋翼飞行器在场地内飞行的画面。

IV. 实验设置与结果

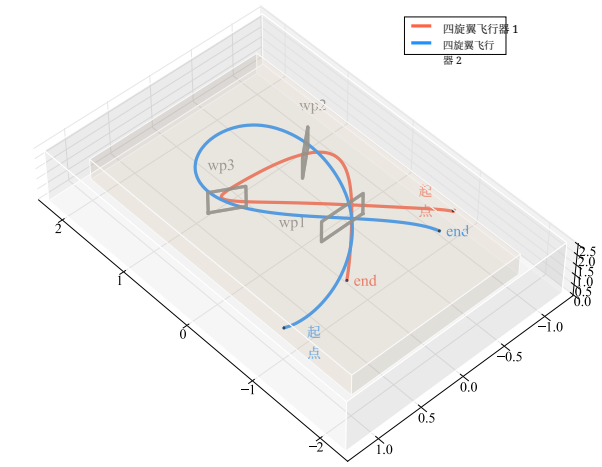
本节展示所提方法在真实世界中的飞行性能。飞行平台是一架自制四旋翼飞行器，机载树莓派运行 Ubuntu 20.04 和 ROS Noetic，以运行高层控制器（图 1）。一台运行 PX4 的 CUAV nora+自动驾驶仪用于底层控制。高层控制器通过 MAVLink协议向自动驾驶仪发送角速度和油门指令。自动驾驶仪执行这些指令，控制四旋翼飞行器跟踪规划轨迹。该飞行平台重量为 $713g$，推重比为5。实验中使用 Opti-track系统提供精确的位置反馈。

由于 Opti-track系统的覆盖范围有限（$4.6m \times 2.6m \times 2.5m$），并且需要留出必要的安全裕度，自由飞行空间仅为 $4m \times 2m \times 1m$。因此，我们只有3个航点用于生成轨迹，以展示所提方法在真实世界中的性能。四旋翼飞行器的航点列于表III中。

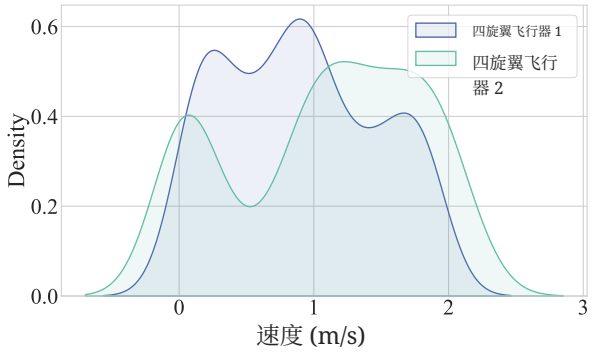
表 III 真实世界实验中的航点位置

四旋翼飞行器编号	航点 1	航点 2	航点 3
1	(0，0，1。5)	(1，−1，1。5)	(1，1，1。5)
2	(0, 0, 1.5)	(1, 1, 1。5)	(1, −1, 1.5)

在真实世界实验中，四旋翼飞行器必须在极其有限的机动空间内加速和减速，最高飞行速度为 $2m/s$。轨迹和速度分布如图9所示，其中我们使用



(a) 真实世界实验中两架四旋翼飞行器的无碰撞轨迹。



(b) 四旋翼飞行器的速度分布。最大速度为$2m/s$

图9. 两架四旋翼飞行器沿3航点轨迹飞行的结果。

门框表示航点。可以看出，两架四旋翼飞行器能够飞过指定航点，同时避开彼此，并以最短时间抵达各自的目标点。

V. 结论

本文提出了一种新颖的自主多无人机竞速轨迹生成方法，该方法能够生成依次通过各航点且避免碰撞的时间最优轨迹。我们还通过仿真展示了所提方法能够在一个 $35m \times 35m$ 空间中为5架四旋翼飞行器生成经过6个航点的轨迹。在这一具有挑战性的环境中，四旋翼飞行器能够达到 $14m/s$的最大速度。我们还在真实世界实验中测试了所提方法；由于飞行场地的尺寸极为有限，四旋翼飞行器仍能以 $2m/s$的最高速度无碰撞地飞过各航点，这证明了所提方法在真实世界中的可行性。

未来还有许多研究方向值得探索。例如，如何提高优化效率，甚至将优化过程移至四旋翼飞行器机载运行，都值得探讨。此外，如何提高激进机动下的轨迹跟踪性能，也是一个值得研究的有趣课题。对于微型飞行器而言，如何仅利用机载资源实现高速定位，仍是一项亟待解决的重大挑战；解决这一问题，才能让机器人飞出实验室。

参考文献

[1] D. Mellinger 和 V. Kumar，“四旋翼飞行器的最小加加速度轨迹生成与控制”，载于 2011年IEEE国际机器人与自动化会议，第 2520–2525 页，IEEE，2011。[2] M. Faessler、A. Franchi 和 D. Scaramuzza，“考虑旋翼阻力的四旋翼飞行器动力学微分平坦性：实现高速轨迹的精确跟踪”，IEEE《机器人与自动化快报》，第 3 卷，第 2 期，第 620–626 页，2017。[3] H. Moon、Y. Sun、J. Baltes 和 S. J. Kim，“IROS 2016竞赛 [竞赛],”，IEEE《机器人与自动化杂志》，第 24 卷，第 1 期，第 20–29 页，2017。[4] S. Jung、S. Cho、D. Lee、H. Lee 和 D. H. Shim，“基于直接视觉伺服的2016年IROS自主无人机竞速挑战赛框架”，《野外机器人学杂志》，第 35 卷，166，2018 1，146 页，第 – 期。[5] H. Moon、J. Martinez-Carranza、T. Cieslewski、M. Faessler、D. Falanga、A. Simovic、D. Scaramuzza、S. Li、M. Ozo、C. De Wagter 等，“自主无人机竞速中的挑战与所用技术”，《智能服务机器人》，第 1–12 页，2019。[6] S. Li、M. M. Ozo、C. De Wagter 和 G. C. de Croon，“自主无人机竞速：一种计算高效的基于视觉的导航与控制策略”，《机器人与自主系统》，第 133 卷，文章编号 103621，2020。[7] S. Li、E. van der Horst、P. Duernay、C. De Wagter 和 G. C. de Croon，“用于计算高效自主竞速的72克无人机视觉模型预测定位”，《野外机器人学杂志》，第 37 卷，第 4 期，第 667–692 页，2020。[8] E. Kaufmann、M. Gehrig、P. Foehn、R. Ranftl、A. Dosovitskiy、V. Koltun 和 D. Scaramuzza，“美人与野兽：最优方法与学习相结合，实现无人机竞速”，载于 2019年国际机器人与自动化会议（ICRA），第 690–696 页，IEEE，2019。[9] C. De Wagter、F. Paredes-Vall´es、N. Sheth 和 G. de Croon，“2019年人工智能机器人竞速比赛获胜团队背后的感知、状态估计与控制”，《野外机器人》，第 2 卷，第 1263–1290 页，2022。[10] P. Foehn、D. Brescianini、E. Kaufmann、T. Cieslewski、M. Gehrig、M. Muglikar 和 D. Scaramuzza，“AlphaPilot：自主无人机竞速”，《自主机器人》，第 46 卷，第 1 期，第 307–320 页，2022。[11] P. Foehn、A. Romero 和 D. Scaramuzza，“四旋翼航点飞行的时间最优规划”，《科学·机器人》，第 6 卷，第 56 期，文章编号 eabh1221，2021。[12] D. Hanover、A. Loquercio、L. Bauersfeld、A. Romero、R. Penicka、Y. Song、G. Cioffi、E. Kaufmann 和 D. Scaramuzza，“自主无人机竞速：综述”，arXiv 电子预印本，第 arXiv–2301 页，2023。[13] A. Kushleyev、D. Mellinger、C. Powers 和 V. Kumar，“迈向灵巧微型四旋翼飞行器集群”，《自主机器人》，第 35 卷，第 4 期，第 287–300 页，2013。[14] W. H¨onig、J. A. Preiss、T. S. Kumar、G. S. Sukhatme 和 N. Ayanian，“四旋翼无人机集群的轨迹规划”，IEEE《机器人学汇刊》，第 34 卷，第 4 期，第 856–869 页，2018。[15] K. McGuire、C. De Wagter、K. Tuyls、H. Kappen 和 G. C. de Croon，“微型飞行机器人集群探索未知环境的最简导航方案”，《科学·机器人》，第 4 卷，第 35 期，文章编号 eaaw9710，2019。[16] B. P. Duisterhof、S. Li、J. Burgu´es、V. J. Reddi 和 G. C. de Croon，“Sniffy bug：在杂乱环境中自主搜寻气体的纳米四旋翼飞行器集群”，载于 2021年IEEE/RSJ智能机器人与系统国际会议（IROS），第 9099–9106 页，IEEE，2021。[17] X. Zhou、X. Wen、Z. Wang、Y. Gao、H. Li、Q. Wang、T. Yang、H. Lu、Y. Cao、C. Xu 等，“野外微型飞行机器人集群”，《科学·机器人》，第 7 卷，第 66 期，文章编号 eabm5954，2022。

[18] X. Zhou、J. Zhu、H. Zhou、C. Xu 和 F. Gao，“Ego-swarm：杂乱环境中的全自主分布式四旋翼无人机集群系统”，载于 *2021*年*IEEE* 国际机器人与自动化会议 *(ICRA)*，第 4101–4107 页，IEEE，2021。