中国科学: 技术科学 2026 年 第 56 卷 第 10 期: 2436 ~ 2449
SCIENTIA SINICA Technologica techcn.scichina.com
《中国科学》杂志社 SCIENCE CHINA PRESS
论 文 低空技术与工程专刊

未知环境下四旋翼无人机快速安全避障飞行控制

王梦云[1], 王博[1], 牛轶峰[1*], 于宏渤[1], 王菖[1*], 张涛[2]

1. 国防科技大学智能科学学院, 长沙 410073
2. 清华大学自动化系, 北京 100080
* E-mail: niuyifeng@nudt.edu.cn; wangchang07@nudt.edu.cn

收稿日期: 2025-10-16; 接受日期: 2026-02-05; 网络版发表日期: 2026-09-28
国家自然科学基金(批准号: 62576349, T2521006, 62576354, 62533010)和湖南省研究生科研创新项目(编号: CX20240005)资助项目

摘要 实现四旋翼无人机在未知复杂环境中的快速、安全自主飞行是推动其在搜救、勘探等时间敏感型任务中应用的关键挑战. 针对传统分层式导航方法存在的误差累积、飞行策略次优以及端到端学习方法难以提供严格安全保证等局限性, 本文提出了一种融合实时感知、模型预测轮廓控制(model predictive contouring control, MPCC)与控制障碍函数(control barrier functions, CBF)的统一飞行控制框架. 该框架以MPCC为核心, 将局部轨迹规划与飞行控制整合于单个非线性优化问题中, 通过最大化任务进度来实现近似时间最优的快速飞行. 同时, 将基于CBF的反应式安全保障能力与MPCC的前瞻性最优控制相结合. 系统仅依赖机载深度相机的实时视觉输入, 在线识别高风险障碍物并构建CBF安全约束, 从而在无须全局无碰撞参考路径和先验地图的条件下, 实时生成兼顾速度、安全与动力学可行性的控制指令. 在仿真与真实物理平台上的大量对比实验验证了所提方法的有效性. 结果表明, 与主流分层式方法相比, 本方法在复杂障碍物场景下的平均飞行速度和成功率显著提高, 并有效克服了传统方法的“走-停”现象, 实现了更加快速、流畅、连贯的飞行轨迹. 在户外树林场景和真实四旋翼平台上的试验成功验证了所提出方法在资源受限的机载系统上的实时性与鲁棒性. 本研究为四旋翼在未知环境中执行快速、安全的自主任务提供了一种高效、可靠的统一解决方案.

关键词 四旋翼无人机, 快速飞行, 安全避障, 控制障碍函数, 模型预测轮廓控制

1 引言

四旋翼(quadrotors)因其高机动性、低成本和部署灵活性, 在搜索救援、物流运输、环境监测等领域展现出巨大的应用潜力[1]. 在许多时间敏感型任务中, 例如灾后快速搜救或紧急物资配送, 飞行速度是决定任务成败的关键因素. 与此同时, 飞行安全是任何自主机器人系统必须满足的首要前提[2]. 因此, 实现四旋翼在未知复杂环境中的快速、安全自主飞行, 是推动其实用化进程的核心挑战之一.

为实现四旋翼的自主导航, 研究人员提出了多种方法, 主要可分为传统分层式方法与端到端的学习方法. 传统分层式方法遵循“感知-规划-控制”的级联架构, 其模块化设计使得系统架构清晰、可解释性强, 并且易于显式地集成安全约束[3]. 然而, 这种方法的固有缺陷在于各模块间的误差累积效应. 例如, 规划模块

引用格式: 王梦云, 王博, 牛轶峰, 等. 未知环境下四旋翼无人机快速安全避障飞行控制. 中国科学: 技术科学, 2026, 56: 2436–2449
Wang M Y, Wang B, Niu Y F, et al. Fast and safe obstacle avoidance flight control for quadrotors in unknown environments (in Chinese). Sci Sin Tech, 2026, 56: 2436–2449, doi: 10.1360/SST-2025-0301

© 2026《中国科学》杂志社

生成的轨迹可能因未充分考虑底层控制器的跟踪能力而导致实际飞行偏差，而控制模块通常只能被动跟踪预先设定的速度剖面，难以充分发掘旋翼的极限动力学性能, 导致飞行策略的次优性.

以模仿学习(imitation learning, IL)[4]和强化学习(reinforcement learning, RL)[5]为代表的机器学习方法, 试图通过数据驱动的方式直接学习从感知到控制的映射. 这类方法在特定场景下展现了优越的性能, 但其应用仍面临诸多挑战. 首先, 在安全性保障方面存在天然缺陷, 由于难以在神经网络中直接编码硬约束, 其安全性通常依赖于惩罚项的间接引导，这在训练分布之外的未知状态空间中难以得到严格保证. 其次，这类方法通常依赖海量数据进行训练，样本效率较低，且学习到的策略泛化能力受限于训练数据的多样性. 此外, 神经网络策略通常是“黑箱”模型, 缺乏可解释性, 为系统调试带来困难.

鉴于上述方法的局限性，模型预测控制(model predictive control, MPC)[6]因其在线优化和处理复杂约束的能力而备受关注. 其中, 模型预测轮廓控制(model predictive contouring control, MPCC)[7]作为MPC的一个重要分支，尤为适合解决快速飞行问题. 与传统MPC跟踪时间参数化的轨迹不同，MPCC以弧长参数化的几何路径为参考，通过在优化目标中同时最小化路径跟踪误差和最大化任务进度，将规划与控制整合在统一的滚动优化框架内. 这种设计使飞行器从被动的轨迹追随者转变为能够主动决策以最大化任务进度的智能体, 有效减小了误差累积, 并能够生成近似时间最优的激进轨迹. 然而, 现有基于MPCC的研究大多聚焦于已知环境下的竞速飞行任务，依赖精确的全局地图来规划参考路径或构建安全约束. 部分工作引入了控制障碍函数(control barrier function, CBF)[8]来保证安全，但其应用场景仍局限于对已知的全局无碰撞路径进行跟踪，CBF的避障作用未能得到充分发挥，极大地限制了该类方法在未知环境中的应用. 综上所述, 如何在未知复杂环境中，将实时感知与兼顾速度和安全的统一规划控制框架有效结合，是当前四旋翼自主飞行领域亟待解决的关键问题.

针对上述挑战，本文提出了一种面向未知环境的四旋翼安全快速飞行框架. 本文以MPCC为核心, 将局部轨迹规划与飞行控制统一于一个非线性优化问题中. 通过引入基于CBF的安全约束, 使无人机能够在无先验地图的情况下, 仅依赖机载视觉感知信息, 实时生成兼顾速度、安全与动力学可行性的控制指令. 主要贡献总结如下. (1) 提出了一种在未知复杂环境下快速安全飞行的方法. 该方法以模型预测轮廓控制为基本框架，将局部重规划与跟踪控制整合在一个优化问题中，以进度最大化为目标，实现近似时间最优的快速飞行. (2) 通过从机载深度相机实时识别障碍物并在线构建控制障碍函数，将安全约束嵌入优化问题中, 使无人机在无需预先构建地图或规划全局路径的情况下实现前瞻性的安全避障. (3) 通过在多种复杂度的仿真环境和真实户外树林中的大量对比实验，验证了所提方法的有效性、鲁棒性和实时性. 实验结果表明, 我们的方法在飞行速度和成功率上均显著优于其他对比方法.

2 相关工作

2.1 四旋翼自主导航

目前，四旋翼自主导航的技术路线可以分为传统分层式框架与端到端学习框架.

传统分层式框架遵循“感知-规划-控制”的级联架构[9]. 这类方法通过模块化的设计, 将复杂的导航问题分解为一系列子问题，使得系统逻辑清晰、可解释性强, 并且便于显式地集成安全约束. 例如, 规划模块可以生成由无碰撞多面体构成的“飞行走廊”来保证路径安全[10]. 然而，这种分层架构的固有缺陷在于各模块间的误差累积与信息隔断. 规划器生成的轨迹往往未充分考虑底层控制器的动力学限制，而控制器只能被动跟踪预设的速度剖面，难以充分发掘飞行器的极限性能, 导致整体飞行策略的次优性.

端到端学习框架，特别是基于IL和RL的方法，试图通过数据驱动的方式直接学习从原始传感器输入到控制指令的映射[11]. 这些方法在特定场景下展现了出色的反应能力. 然而，其在安全至上的飞行任务中应用仍面临严峻挑战[12]. 首先，其安全性难以得到严格保证. 由于神经网络难以直接编码硬约束，安全通常依赖于奖励函数中的惩罚项进行间接引导，这在训练分布之外的未知区域中可能失效[2]. 其次，这类方法(尤其是无模型RL)通常样本效率低下，依赖海量训练数据, 且学习到的策略泛化能力有限. 此外, 其“黑箱”特性也为系统的调试、验证与部署带来了困难[13].

2.2 模型预测轮廓控制

为实现更快速的飞行，研究人员将目光投向了MPC[14]，特别是其适用于轮廓跟踪任务的分支——MPCC[15]. 与跟踪时间参数化轨迹的传统MPC不同, MPCC以最大化沿几何路径的任务进度为主要优化目标, 从而将规划与控制统一为一个问题, 能够生成近似时间最优的飞行策略.

早期的MPCC研究主要集中在已知环境下的竞速任务. 例如，Romero等人[16]首次将MPCC成功应用于无人机竞速，在预先扫描构建的赛道上实现了极限飞行. Krinner等人提出的MPCC++[17]则通过设计虚拟隧道约束, 在MPCC框架内加入了安全保障, 但这种方法依然局限于赛道等结构化环境，不适用于未知空间的探索. 近期，PE-planner[18]将CBF引入MPCC中以增强安全性，但其应用前提是已经规划好了一条全局无碰撞的参考轨迹，环境信息完全已知. 在这种设定下，CBF的避障作用未能得到充分发挥，且未能解决核心的未知环境感知与导航问题.

综上，尽管MPCC在生成快速轨迹方面潜力巨大, 但现有工作大多依赖于对环境的先验知识，如何将其与实时感知结合以适应未知复杂环境，是亟待解决的开放问题.

2.3 飞行安全保障

飞行安全是自主系统成功完成任务的基础. 除了上文提到的飞行走廊等规划层面的方法，控制层面的安全保障机制也得到了广泛研究. 其中，CBF因其能提供前瞻性的、有理论保证的安全性而备受关注.

目前，CBF在无人机领域的应用通常以安全过滤器的形式出现. 例如，Chen等人[19]设计了一种基于MPC的安全过滤器，用于修正学习策略生成的可能不安全的轨迹，使其保持在安全边界内. 类似地，Xu等人[20]将RL策略的输出速度限制在预先计算的安全速度范围内. 这些方法虽然有效, 但它们将安全模块作为一个独立的、后置的“干预”环节，增加了系统的复杂度与计算延迟. 这种分层式的安全保障方式与飞行性能优化是解耦的，可能导致过于保守的行为，并且与追求极致性能和模块统一的控制思想相悖. Saviolo等人[21]将CBF约束纳入非线性模型预测控制跟踪优化, 构建了反应式避障框架，但通常仅适用于障碍物稀疏的低速场景. 如何将CBF所代表的形式化安全保证, 无缝、高效地内生于一个统一的规划控制框架中，是提升系统整体性能的关键.

3 未知环境下快速安全飞行控制方法

本文提出的方法框架可以分为感知、规划控制、安全约束构建三个部分. 系统首先利用机载深度相机实时感知环境，通过聚类算法识别并定位高风险障碍物; 随后, 根据感知结果在线构建CBF安全约束, 并将其无缝集成到MPCC优化问题中; 最后, 通过滚动求解该优化问题，实现对任意无需考虑避障的参考路径进行快速、安全的跟与避障. 该框架将反应式的避障能力与前瞻性的最优控制有机结合，无需预先规划全局无碰撞路径，赋予了四旋翼在未知环境中高度的自主决策与适应能力.

3.1 问题描述

如图1所示，在本文的场景想定中，四旋翼无人机搭载传感器和计算平台，需要在未知的复杂环境中实现自主快速飞行的同时安全躲避潜在的障碍物，以高效完成各种时间敏感任务. 由于环境是未知的，无人机无法预先得到障碍物的分布情况或者任何全局参考路径, 在飞行过程中仅有简单的参考飞行方向指导. 无人机需要通过自身传感器对障碍物进行识别判断，并利用机载计算单元优化生成的控制信号，实现安全快速飞行. 在本文中，四旋翼通过相机、IMU、激光雷达等传感器对自身状态和周围环境进行感知，传感器数据经过感知模块和快速安全飞行控制模块处理生成角速度和推力信号组成的控制指令，再发送给飞行控制器完成底层闭环控制.

首先对四旋翼无人机进行动力学建模. 假设四旋翼是质量为m、惯性矩阵为$\mathbf{J}$的刚体，则其动力学模型可以表示为

$$\begin{aligned}&\dot{\mathbf{p}} = \mathbf{v},\\&m\dot{\mathbf{v}} = \mathbf{R}_b^e\mathbf{T}_b + \mathbf{f}_D + m\mathbf{g},\\&\dot{\mathbf{R}}_b^e = \mathbf{R}_b^e[\boldsymbol{\omega}_b]_\times,\end{aligned} \tag{1}$$

其中，$\mathbf{p} = [p_x, p_y, p_z]^{\mathrm{T}}$和$\mathbf{v} = [v_x, v_y, v_z]^{\mathrm{T}}$为世界系下的无人机位置向量和速度向量，$\mathbf{R}_b^e$是从机体系到世界系的旋转矩阵，$\boldsymbol{\omega}_b$是机体系下的角速度，$[\boldsymbol{\omega}_b]_\times$是$\boldsymbol{\omega}_b$的反对称矩阵，$\mathbf{T}_b = [0, 0, f_{\mathrm{T}}]^{\mathrm{T}}$为由4个螺旋桨产生的机体系的

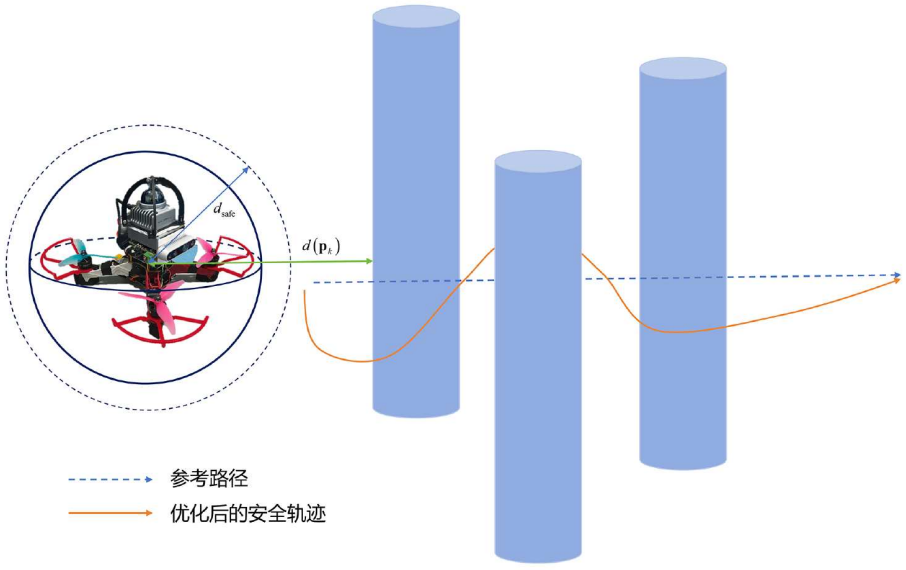


图 1　(网络版彩图)四旋翼根据任意参考路径优化得到安全避障轨迹
Figure 1　(Color online) The quadrotor optimizes its flight path along any reference trajectory to achieve safe obstacle avoidance.

推力, $\mathbf{g}$是重力加速度向量.

综上所述, 控制问题的状态输入为t时刻的RGB图像$\mathbf{I}^t \in \mathbb{R}^{H\times W\times 3}$、深度图像$\mathbf{D}^t \in \mathbb{R}^{H\times W}$和四旋翼状态$\mathbf{x}^t = [\mathbf{p}^{\mathrm{T}}, \mathbf{q}^{\mathrm{T}}, \mathbf{v}^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{10}$, 控制输出为四旋翼的质量归一化推力$t_{\mathrm{mn}} = f_{\mathrm{T}} / m$和机体角速度$\boldsymbol{\omega}_b$.

3.2　基于模型预测轮廓控制的快速飞行

MPC被广泛用于轨迹跟踪问题, 给定每一离散时刻的参考状态$\mathbf{x}_{k,\mathrm{ref}}$和参考控制信号$\mathbf{u}_{k,\mathrm{ref}}$, MPC可以表示为滚动求解以下优化问题:

$$J_{\mathrm{MPC}} = \sum_{k=0}^{N}\left(\left\|\Delta\mathbf{x}_k\right\|_{\mathbf{Q}}^2 + \left\|\Delta\mathbf{u}_k\right\|_{\mathbf{R}}^2\right). \tag{2}$$

与传统模型预测控制最小化到参考状态的欧氏距离不同, 模型预测轮廓控制最小化到参考路径的距离的同时最大化沿着该路径上的进度. MPCC问题只需要一个三维轮廓线作为参考, 而不需要可执行的参考轨迹, 因此被称作轮廓控制.

定义沿着参考路径的进度为从参考路径起始点开始的弧长θ, 与MPC中使用时间参数化轨迹不同, MPCC的参考路径是使用进度θ参数化的. 定义轮廓误差为当前位置$\mathbf{p}_k$到参考路径$\mathbf{p}^d(\theta_k)$的距离, $e_c(\theta_k) = \min\limits_{\theta_k}\left\|p_k - p^d(\theta_k)\right\|_2$. 由于计算轮廓误差本身就是一个优化问题, 需要使用一定的近似方法对其进行估计. 使用$e_c(\theta_k)$在平面$\pi(\hat{\theta}_k)$上的投影$\hat{\mathbf{e}}_c(\hat{\theta}_k)$的模$\left\|\hat{\mathbf{e}}_c(\hat{\theta}_k)\right\|$对$e_c(\theta_k)$进行近似, 如图2所示. 而使用$\hat{\theta}_k$对$\theta_k$的近似误差为滞后误差$e_l(\theta_k) = \left\|\theta_k - \hat{\theta}_k\right\|$, 该段弧长同样可以使用$\left\|\hat{\mathbf{e}}_l(\hat{\theta}_k)\right\|$近似. 定义$\mathbf{p}^d(\hat{\theta}_k)$到当前位置$\mathbf{p}_k$的向量为$\mathbf{e}(\hat{\theta}_k)$, 则轮廓误差$e_c(\theta_k) \approx \left\|\hat{\mathbf{e}}_c(\hat{\theta}_k)\right\|$和滞后误差$e_l(\theta_k) \approx \left\|\hat{\mathbf{e}}_l(\hat{\theta}_k)\right\|$可以由$\mathbf{e}(\hat{\theta}_k)$计算得到:

$$\hat{\mathbf{e}}_l(\hat{\theta}_k) = [\mathbf{t}(\hat{\theta}_k)^{\mathrm{T}}\mathbf{e}(\hat{\theta}_k)]\mathbf{t}(\hat{\theta}_k), \tag{3}$$

$$\hat{\mathbf{e}}_c(\hat{\theta}_k) = \mathbf{e}(\hat{\theta}_k) - \hat{\mathbf{e}}_l(\hat{\theta}_k), \tag{4}$$

其中$\mathbf{t}(\hat{\theta}_k)$为$\hat{\theta}_k$处的切线向量. 可以证明, 当$\hat{\mathbf{e}}_l(\hat{\theta}_k) = 0$时, 近似值$\hat{\theta}_k$, $\hat{\mathbf{e}}_l(\hat{\theta}_k)$和$\hat{\mathbf{e}}_c(\hat{\theta}_k)$等于真实值, 为了保证这种近似估计的准确性, 需要最小化滞后误差.

对于进度项$\hat{\theta}$, 我们通过最大化$v_{\hat{\theta}} = \mathrm{d}\hat{\theta} / \mathrm{d}t$来间接最大化$\hat{\theta}$, 其离散动力学可以写作

$$\hat{\theta}_{k+1} = \hat{\theta}_k + v_{\hat{\theta},k}\Delta t. \tag{5}$$

除了沿着参考路径快速飞行的目标外, 四旋翼还

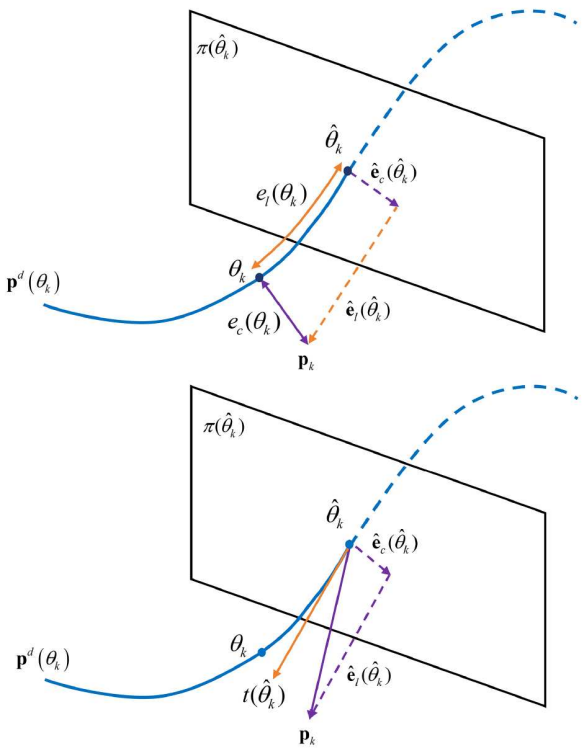


图 2 轮廓误差定义及其近似计算

Figure 2 (Color online) Definition of contour error and its approximate calculation.

需要自主对环境进行感知, 因此需要机头(即相机)方向沿着飞行速度方向, 定义偏航角误差$e_{\text{yaw}}(\hat{\theta}_k) = \left\|\hat{\theta}_k - \theta_k^d\right\|$.

综上所述, MPCC的优化目标是最小化轮廓误差和滞后误差, 同时最大化进度θ, 从而保证快速而精准地跟踪参考路径. 为了方便后续表述, 我们取消所有近似变量上的帽子符号, 则MPCC的优化目标可以写为

$$\begin{aligned} J_{\text{MPCC}} = \sum_{k=0}^{N}\Big[&\left\|e_c(\theta_k)\right\|_{q_c}^2 + \left\|e_l(\theta_k)\right\|_{q_l}^2 + \left\|e_{\text{yaw}}(\theta_k)\right\|_{q_{\text{yaw}}}^2 \\ &+\left\|\mathbf{u}_k\right\|_{\mathbf{R}_u}^2 + \left\|\Delta\mathbf{u}_k\right\|_{\mathbf{R}_{\Delta u}}^2 - \mu v_{\theta,k}\Big]. \end{aligned} \tag{6}$$

通过在飞行过程中实时求解以上优化问题, 即可将规划控制模块一体化求解.

为了求解该优化问题, 需要已知参考路径$\mathbf{p}^d(\theta)$, 对于任意一段使用参数t表示的三次样条曲线$\mathbf{q}(t) = [q_x(t), q_y(t), q_z(t)]$, 其中存在$n$+1个控制点$\{t_0, t_1, \dots, t_n\}$和$n$段样条线, 需要将其转为弧长参数化曲线$\mathbf{p}(\theta)$. 弧长参数化过程如下.

首先计算每一段曲线的弧长:

$$l_i = \int_{t_i}^{t_{i+1}} \sqrt{[q'_x(t)]^2 + [q'_y(t)]^2 + [q'_z(t)]^2}\,\mathrm{d}t, \tag{7}$$

其中$i = 0, \dots, n-1$, 则总弧长$L = \sum_{i=0}^{n-1} l_i$. 将总弧长等分成m段, 则每一小段$\Delta l = L/m$. 由此可以得到m+1个控制点, 所在弧长分别为$0, \Delta l, 2\Delta l, \dots, m\Delta l$, 根据弧长坐标计算这些点对应的参数$\tilde{t}_0, \tilde{t}_1, \dots, \tilde{t}_m$, 其满足

$$\int_{t_0}^{\tilde{t}_i} \frac{\mathrm{d}s}{\mathrm{d}t}\mathrm{d}t = i\Delta l. \tag{8}$$

为了计算$\tilde{t}_i$, 首先找到满足$\sum_{k=0}^{j-1} l_k \leqslant i\Delta l < \sum_{k=0}^{j} l_k$的索引$j$, 该条件能保证$t_j \leqslant \tilde{t}_i < t_{j+1}$, 则公式可以写为

$$\int_{t_0}^{\tilde{t}_i} \frac{\mathrm{d}s}{\mathrm{d}t}\mathrm{d}t = \int_{t_0}^{t_j} \frac{\mathrm{d}s}{\mathrm{d}t}\mathrm{d}t + \int_{t_j}^{\tilde{t}_i} \frac{\mathrm{d}s}{\mathrm{d}t}\mathrm{d}t = \sum_{k=0}^{j-1} l_k + \int_{t_j}^{\tilde{t}_i} \frac{\mathrm{d}s}{\mathrm{d}t}\mathrm{d}t = i\Delta l, \tag{9}$$

则$\int_{t_j}^{\tilde{t}_i} (\mathrm{d}s/\mathrm{d}t)\mathrm{d}t = i\Delta l - \sum_{k=0}^{j-1} l_k$. 使用二分法即可求得$\tilde{t}_i$. 弧长参数化的算法具体流程如算法1所示.

算法1 弧长参数化算法

输入: 原始样条线的每段弧长$l_1, l_2, \dots, l_{n-1}$

输出: 均等分m段后每段弧长对应的$\tilde{t}_0, \tilde{t}_1, \dots, \tilde{t}_m$

for i = 0,⋯, m **do**
  计算j使其满足$\sum_{k=0}^{j-1} l_k \leqslant i\Delta l < \sum_{k=0}^{j} l_k$, 则$\tilde{t}_i \in [t_j, t_{j+1}]$
  $t_l \leftarrow t_j$, $t_r \leftarrow t_{j+1}$, $t_m \leftarrow \frac{t_l + t_r}{2}$
  $\Delta s \leftarrow \int_{t_j}^{t_m} \frac{\mathrm{d}s}{\mathrm{d}t}\mathrm{d}t$
  while $\left|\Delta s - i\Delta l - \sum_{k=0}^{j-1} l_k\right| > 1 \times 10^{-3}$
    if $\Delta s < i\Delta l - \sum_{k=0}^{j-1} l_k$
      $t_l \leftarrow t_m$
    else
      $t_r \leftarrow t_m$
    end if
    $\Delta s \leftarrow \int_{t_j}^{t_m} \frac{\mathrm{d}s}{\mathrm{d}t}\mathrm{d}t$
  $\tilde{t}_i \leftarrow t_m$
end for
返回$\tilde{t}_0, \tilde{t}_1, \dots, \tilde{t}_m$

为了突出本文方法的实时避障能力，我们设定了连接无人机起始位置与目标点的直线作为参考路径. 该直线路径在输入到MPCC控制器之前，会经过算法1所述的弧长参数化处理，转换为以弧长θ为变量的路径$\mathbf{p}^r(\theta)$.

3.3　基于控制障碍函数的安全避障约束

为了在飞行期间避免与障碍物发生碰撞，需要在上述优化问题中加入避障约束. 相比简单的障碍物安全距离约束，CBF可以保持系统处于一个安全的不变集里，更具备前瞻性和安全性保障. 假设无人机中心位置$\mathbf{p}_k$与障碍物之间的距离为$d(\mathbf{p}_k)$，则CBF可以设计为

$$h(\mathbf{x}_k) = d^2(\mathbf{p}_k) - d_{\text{safe}}^2, \tag{10}$$

其中，d_{safe}为无人机中心距离障碍物表面的安全距离，小于该距离则认为发生碰撞. 构建离散形式的二次CBF函数:

$$\begin{aligned} &h^0(\mathbf{x}_k) = h(\mathbf{x}_k),\\ &h^1(\mathbf{x}_k) = h^0(\mathbf{x}_{k+1}) + (c_1 - 1)h^0(\mathbf{x}_k),\\ &h^2(\mathbf{x}_k) = h^1(\mathbf{x}_{k+1}) + (c_2 - 1)h^1(\mathbf{x}_k),\\ &h^3(\mathbf{x}_k) = h^2(\mathbf{x}_{k+1}) + (c_3 - 1)h^2(\mathbf{x}_k). \end{aligned} \tag{11}$$

当以上函数和$\mathbf{u}_k$满足

$$\begin{aligned} &h^p(\mathbf{x}_0) \geqslant 0,\ p = 0,\ 1,\ 2,\ 3,\\ &h^3(\mathbf{x}_k, \mathbf{u}_k) \geqslant 0,\ k = 1,\ 2, \ldots,\ N-1 \end{aligned} \tag{12}$$

时，根据不变集理论，后续所有$\mathbf{x}_k$都满足$h(\mathbf{x}_k) \geqslant 0$，即飞行状态满足安全约束.

3.4　未知环境的障碍物感知

由于本文环境是未知的，无法直接获得安全约束中的障碍物位置坐标，需要从图像中识别并输出障碍物的位置. 考虑到森林环境中障碍物具有密集分布纹理复杂，以及在不同光照下与背景区分度低等特点，传统的基于几何特征或颜色信息的障碍物检测方法难以取得稳定可靠的效果. 本节提出了一种基于深度学习与几何聚类相结合的障碍物三维位置感知方法. 该方法首先利用轻量化的深度神经网络对深度图像进行语义分割，提取障碍物像素区域；然后，通过聚类算法将分割后的像素集划分为独立的障碍物实例；最后，结合相机模型与无人机位姿信息，解算出每个障碍物在世界坐标系下的三维坐标.

考虑到机载计算平台的算力限制，本文选用MobileUNetV2作为分割网络，如图3所示，其结合了MobileNetV2的倒残差结构和线性瓶颈层以及UNet的编码器-解码器和跳跃连接结构. 前者作为编码器，能够以较低的计算成本高效提取图像特征，后者则能有效融合深层语义信息和浅层空间细节信息，实现对障碍物的精确分割. 该模型在保持较高分割精度的同时，显著降低了参数量与计算复杂度，满足了实时性要求. 为了确保感知网络的鲁棒性并有效实现从仿真到现实的迁移，我们的数据集由仿真和真实数据两部分构成.

分割网络的训练依赖于大量高质量的标注数据. 像素级手动标注成本高昂且效率低下，我们设计了一套自动标签生成方法，利用几何先验自动识别并剔除地面，从而生成障碍物掩码. 生成的掩码作为监督学习的标签训练分割网络.

假设对分割掩码进行聚类得到一簇障碍物$\{C_1, C_2, \ldots, C_K\}$及其对应的像素坐标:

$$(u_i, v_i) = \left[\frac{1}{|C_i|}\sum_{(u,v)\in C_i} u, \frac{1}{|C_i|}\sum_{(u,v)\in C_i} v\right],\ i = 1,\ 2, \ldots,\ K, \tag{13}$$

则障碍物在相机坐标系下的位置$P_C = (X_i, Y_i, Z_i)$为

$$\begin{aligned} X_i &= \frac{(u_i - c_x)Z_i}{f_x},\\ Y_i &= \frac{(v_i - c_y)Z_i}{f_y}, \end{aligned} \tag{14}$$

其中Z_i为障碍物的平均深度$Z_i = \left(\sum_{(u,v)\in C_i} d\right) / |C_i|$. 障碍物在世界坐标系下的位置$P_W$可以通过坐标系转换得到:

$$P_W = R_{WB}(R_{BC}P_c + t_{BC}) + t_{WB}, \tag{15}$$

其中R_{BC}和t_{BC}分别是相机坐标系到机体坐标系的旋转矩阵和平移向量，R_{WB}和t_{WB}分别是机体坐标系到世界坐标系的旋转矩阵和平移向量，完整的障碍物感知流程如图4所示.

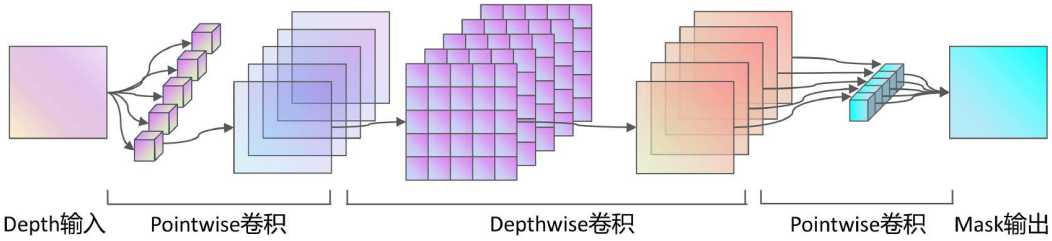


图 3 (网络版彩图) MobileNetV2网络基本结构
Figure 3 (Color online) Basic architecture of MobileNetV2 network.

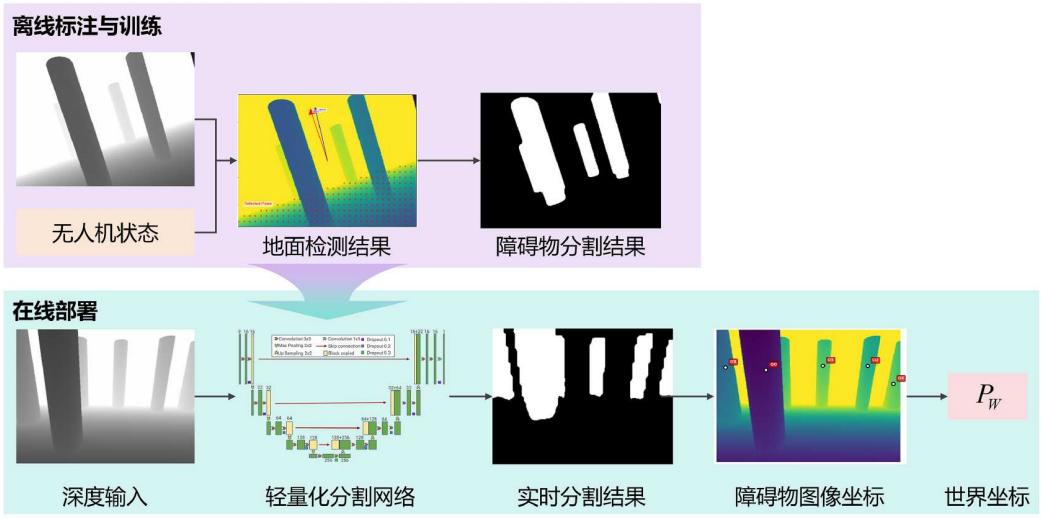


图 4 (网络版彩图)从深度图像中计算障碍物坐标的算法流程
Figure 4 (Color online) General process for calculating obstacle coordinates from depth images.

4 仿真与试验结果

在本节中, 我们进行了一系列的仿真和实物试验验证本文提出的控制策略, 一切试验设计旨在于回答以下问题. (1) 与其他基线方法对比, 提出的控制方法在飞行性能、成功率、计算消耗等方面的性能如何? (2) 设计的控制障碍函数安全约束有什么优势? (3) 提出的轻量化障碍物感知模块是否能够满足快速飞行的实时性要求, 其性能对整体框架有何影响? (4) 所提算法框架能否有效部署于计算资源受限的真实四旋翼平台, 并在真实环境中稳定运行?

4.1 实现细节

所有仿真实验均基于FlightBench仿真平台, 在一台配备Intel Core i9-13900K CPU与NVIDIA RTX 4070Ti GPU的计算机上进行. 为模拟未知复杂环境, 我们随机生成了三种不同障碍物密度的场景, 障碍物数量分别为30 (简单)、50 (中等)和70 (困难). 所有场景中, 参考路径均设置为一条直接贯穿障碍物区域的直线, 以此考验算法的安全避障能力. 感知模块使用Pytorch实现, MPCC部分使用C++和NLopt库实现. 实验中算法超参数的设置如表1所示. 为全面评估本文方法的性能, 我们选取了以下几种代表性的规划方法作为基线.

(1) Fast-planner[22]: 一种基于快速随机搜索树(rapidly-exploring random tree, RRT*)的经典方法, 在前端进行路径搜索.

(2) Ego-planner[3]: 一种基于轨迹优化的方法, 通

表 1　算法超参数取值

Table 1　Hyperparameter settings for the proposed approach

变量	取值
q_c	0.5
q_{yaw}	0.3
μ	30
c_1	0.9
c_3	0.9
q_l	35
$\mathbf{R}_u$	diag{0.4,0.4,0.1,0.2}
$\mathbf{R}_{\Delta u}$	diag{0.6,0.6,0.2,3.0}
c_2	0.9
d_{safe}	0.3

过梯度信息优化轨迹以规避障碍物.

(3) PE-planner[18]: 从ESDF地图中获取最近的危险点, 构建CBF保证安全避障.

(4) 本文方法-完美感知: 我们还设置了一个“完美感知”的本文方法变体, 该变体直接从仿真环境中获取障碍物的精确位置信息, 作为性能的理论上界.

4.2　仿真试验

4.2.1　飞行轨迹质量定性分析

如图5所示展示了本文方法在三种不同难度场景下生成的飞行轨迹. 轨迹颜色代表飞行速度, 可以看出, 即便在障碍物密集的困难场景中, 本文方法依然能够生成平滑、连贯的轨迹, 完美规避所有障碍物, 并全程保持较高速度.

为了更直观对比, 图6可视化了在中等密度场景下不同方法的典型飞行轨迹. 可以看出, 本文方法生成的轨迹(蓝色)最为平滑和连贯, 能够在高速前进的同时, 以最小的绕行代价优雅地规避障碍物. 这得益于MPCC框架的前瞻性与CBF约束的平滑引导. 相比之下, 传统分层式方法则表现出明显的“走-停”行为. 在接近障碍物时, 它们需要紧急减速甚至悬停以进行重规划, 导致轨迹出现频繁的停顿和急转.

图7中的速度曲线也直观展示了这种现象. 在所有场景中, 本文方法的速度曲线均能维持在较高水平且波动平缓. 而基线方法的速度则呈现出周期性的剧烈下降, 每一次下降都对应着一次代价高昂的重规划过程, 这在障碍物密集的困难场景中严重影响了整体飞行效率. 一个有趣的现象是, 图7中本文方法的速度曲线在任务的末端阶段存在一个尖峰现象. 这一行为是因为当无人机在接近终点时, 其前向路径上无障碍物, 则算法以最大化任务进度为核心的目标函数将驱动无人机在动力学极限内进行最终加速, 以实现近似时间最优的到达. 而紧随其后的快速减速, 则是由于任务的终端条件要求无人机在到达目标点后立即悬停.

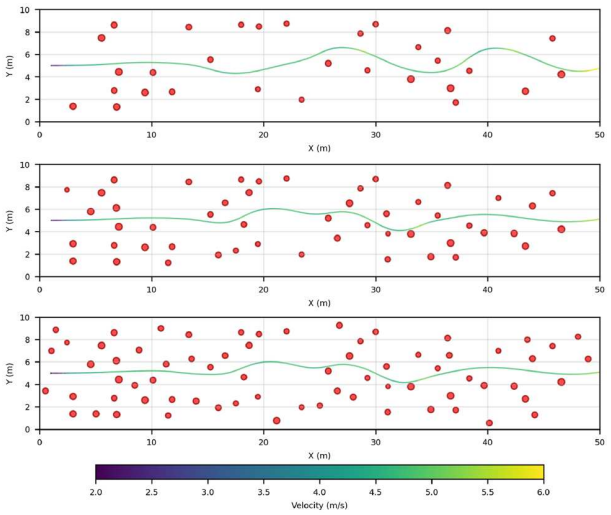


图 5　(网络版彩图)三种难度场景下的飞行轨迹, 轨迹颜色表示速度大小

Figure 5　(Color online) Flight trajectories under three difficulty scenarios, with trajectory colors indicating speed magnitude.

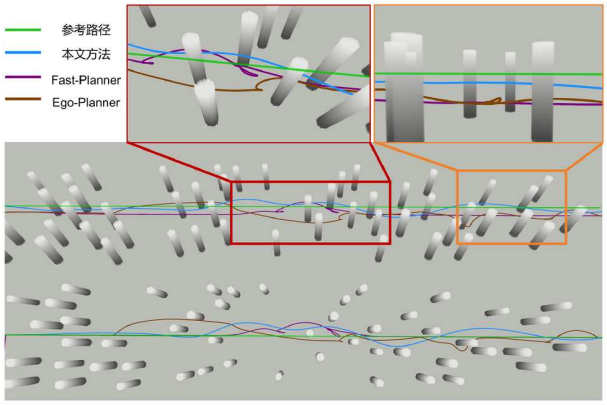


图 6　(网络版彩图)不同对比算法得到的飞行轨迹(图中分别为侧视和俯视两个视角)

Figure 6　(Color online) Flight trajectories obtained by different comparison algorithms (shown in side view and top view perspectives).

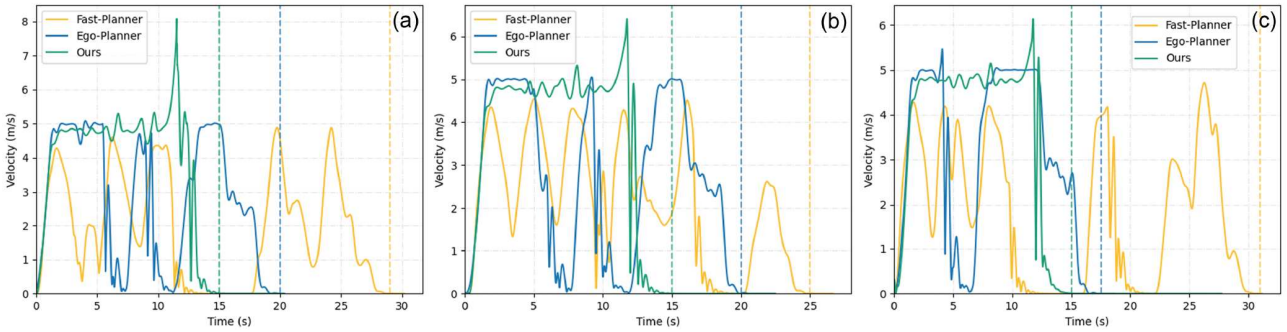


图 7 (网络版彩图)三种场景下各种对比算法的速度变化曲线以及任务完成时间. (a) 低障碍物密度场景; (b) 中障碍物密度场景; (c) 高障碍物密度场景

Figure 7 (Color online) Speed variation curves of various comparison algorithms under different scenarios and task completion times. (a) Low obstacle density scene; (b) the scene of medium obstacle density; (c) high obstacle density scene.

4.2.2 飞行关键性能指标定量分析

表1展示了不同方法在三种场景下的定量飞行指标, 我们从飞行效率、安全性与计算开销三个维度进行深入分析. 本文方法和传统方法的最大飞行速度均设置为6 m/s.

结合图8中的平均飞行速度柱状图和表格中的最大速度结果分析, 在飞行效率方面, 尽管几种方法的最大速度差异不大, 但是本文方法在所有场景下的平均飞行速度均显著领先于基线方法. 尤其在困难场景中, 本文方法的平均速度高达4.62 m/s, 分别是Fast-planner (1.92 m/s)和Ego-planner (2.27 m/s)的2.4倍和2.0倍. 这一巨大差异充分证明了本文统一框架的优越性, 即MPCC能够通过将规划与控制统一、并以最大化任务进度为优化目标, 能有效克服分层方法的次优性, 从而更充分地发掘无人机的动力学潜能.

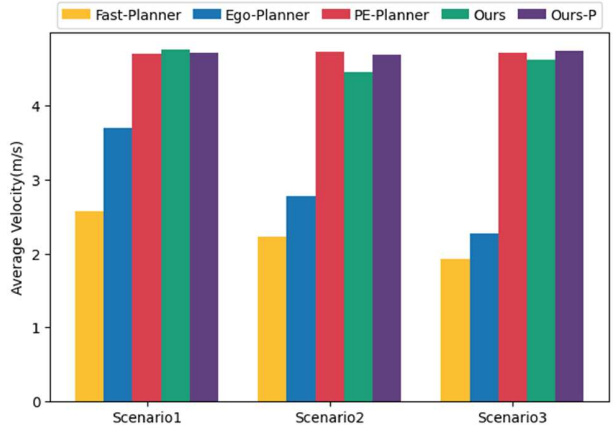


图 8 (网络版彩图)不同对比算法的平均飞行速度

Figure 8 (Color online) Average flight speed of different contrast algorithms.

在安全性与成功率方面, 本文方法同样展现出卓越的鲁棒性. 如图9所示, 随着环境复杂度的提升, 基线方法的成功率急剧下降, 在困难场景下仅为10%. 这是因为其反应式的重规划机制在高速飞行下面临巨大挑战, 感知或规划的微小延迟都可能导致碰撞. 而本文方法得益于CBF约束提供的前瞻性安全保障, 即使在困难场景下, 依然能维持70%的高成功率. 特别地, 当提供完美感知信息时, 成功率可进一步提升至80%~100%, 证明了快速飞行方法框架本身的可靠性. 值得注意的是, PE-planner在本实验设置下成功率非常低. 而在高质量的全局无碰撞参考路径的引导下, PE-planner*在所有场景下都达到了90%以上的成功率, 并且其平均速度也远高于原始的PE-planner. 其根本原因在于, 当PE-planner从预先构建的ESDF地图中读取危险碰撞点坐标时, 由于ESDF地图的特性仅能同时获取一个危险点, 在原论文中参考轨迹为全局规划得到的高质量无碰撞轨迹时, 该方法能够保证安全, 然而在本文仅提供任意直线参考路径时, 该方法的成功率较低.

从表2的时间消耗数据来看, 本文方法的规划控制耗时(约4 ms)远低于Fast-planner (约13 ms)和Ego-planner (约18 ms). 这是因为MPCC在滚动时域内求解一个规模相对固定的优化问题, 而基线方法则可能需要进行复杂的路径搜索或轨迹优化. 在未知环境中进行快速飞行, 系统的总延迟, 即从传感器捕捉到新信息到执行相应控制指令的时间差, 直接决定了无人机的反应

表 2　不同场景下各种对比算法的定量指标, 每种场景进行10次试验

Table 2　Quantitative metrics for various contrast algorithms across different scenarios, with 10 trials conducted for each scenario

场景	方法	飞行性能				时间消耗	
		平均速度 (m/s)	最大速度 (m/s)	最小障碍物距离 (m)	成功率 (%)	预处理 (ms)	规划控制 (ms)
简单	Fast-planner	2.56899	4.81834	0.258738	80	42.1	13.7327
	Ego-planner	3.69639	5.07531	0.141417	90	34.4	15.3243
	PE-planner	4.70224	5.35515	0.069276	20	715.428	4.5280
	PE-planner*	7.746800	12.875508	0.276712	100	1203.173	3.45562
	本文方法	4.75453	5.78887	0.10912	80	12.2	3.81605
	本文方法-完美感知	4.71768	5.36702	0.136004	100	–	3.504
中等	Fast-planner	2.222968	4.53847	0.123545	60	52.2952	10.5607
	Ego-planner	2.774857	5.04957	0.113689	90	45.2668	16.8037
	PE-planner	4.7235	5.56235	0	0	731.245	4.6701
	PE-planner*	6.103296	9.214615	0.171814	100	1232.200	4.2337
	本文方法	4.45516	5.42706	0.154464	80	11.5	4.29512
	本文方法-完美感知	4.68325	5.60545	0.142489	100	–	4.1457
困难	Fast-planner	1.919524	4.715924	0.297574	10	48.7813	10.9552
	Ego-planner	2.265157	5.468904	0.108505	10	42.9798	17.8141
	PE-planner	4.71502	5.33616	0	0	732.587	3.7736
	PE-planner*	5.836548	9.048533	0.1629	90	1250.512	4.1107
	本文方法	4.61906	5.43806	0.12328	70	11.4	3.7925
	本文方法-完美感知	4.74233	5.60159	0.101511	80	–	3.4802

距离.

尽管本文方法需要额外的感知预处理时间(约12 ms), 但其总耗时仍具有竞争力, 且感知与控制可分

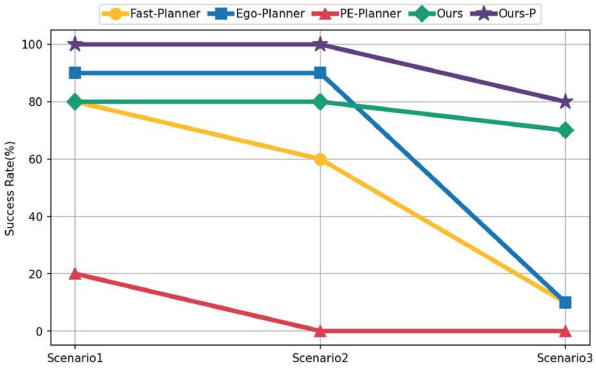


图 9　(网络版彩图)随着任务难度提升, 不同对比算法的成功率变化

Figure 9　(Color online) As task difficulty increases, the success rate of different contrast algorithms changes.

别在机载计算机的GPU和CPU上并行执行, 完全满足实时性要求. 如此短的决策时间使得无人机能够进行连续、平滑的轨迹调整, 实现了前瞻性的避障. 相比之下, 传统分层方法的瓶颈通常在于感知与建图环节, 其耗时普遍在30~50 ms之间. 为了保证安全, 系统唯一的选择就是在探测到潜在碰撞时采取保守的紧急制动策略, 等待高延迟的规划模块计算出新的安全路径. 而依赖显式建图的如PE-planner方法在构建ESDF地图时需要巨大的预处理开销(超过700 ms), 不适用于未知环境下的快速反应任务.

4.3　实物飞行试验

为验证所提框架在真实物理世界中的可行性与鲁棒性, 我们开展了具有挑战性的室外飞行试验. 我们自行搭建了如图10所示的四旋翼实验平台. 该平台采用250 mm轴距的机架, 保证了飞行的灵活性与机动性.

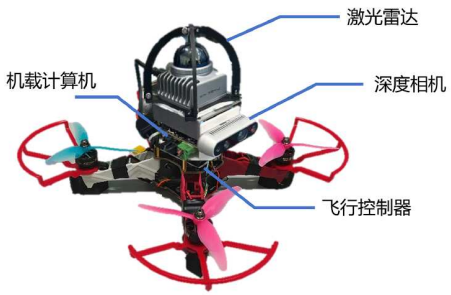


图 10 (网络版彩图)自行搭建的四旋翼无人机结构组成
Figure 10 (Color online) Structure of a self-made quadrotor.

底层姿态控制与电机驱动由雷讯Nora+飞行控制器负责, 提供稳定可靠的底层飞行能力. 所有上层算法, 包括感知、状态估计与MPCC求解, 均在高性能的英伟达Jetson Orin NX嵌入式计算机上运行. 该平台代表了当前资源受限机载系统的主流配置. Intel RealSense D435i深度相机作为主感知传感器, 为算法提供实时的环境深度信息. 在室外环境中, 我们使用Mid-360激光雷达通过FAST-LIO算法[23]进行高精度的实时定位, 为系统提供全局一致的状态估计. 需要强调的是, 激光雷达仅用于定位, 所有用于避障的障碍物信息完全来自于深度相机, 这与我们在未知环境中的设定完全一致.

实验在一个真实户外树林场景中进行, 如图11(a)所示. 该场景包含了大量随机分布、形态各异的树木, 以及复杂多变的光照条件, 对算法的感知鲁棒性和实时决策能力构成了严峻考验. 实验任务设置为: 在无任何先验地图的情况下, 命令无人机跟踪一条穿越树林的直线参考路径, 要求其自主、安全地规避所有路径上的障碍, 并最终到达指定终点.

图11(b)记录了一次典型飞行试验中的三维状态轨迹. 从中可以清晰地看到, 尽管黄色直线表示的参考路径直接穿过了多个用灰色圆柱体表示的障碍物, 但无人机实际飞行的轨迹(红色曲线)显示, 无人机在保持沿着参考路径方向高速前进的同时, 能够主动绕开障碍. 例如, 在轨迹的中间点1和中间点3附近, 为了躲避前方的树木, 无人机进行了两次机动, 整个过程没有出现任何因重规划导致的急转或停顿. 图11(c1)~(c3)

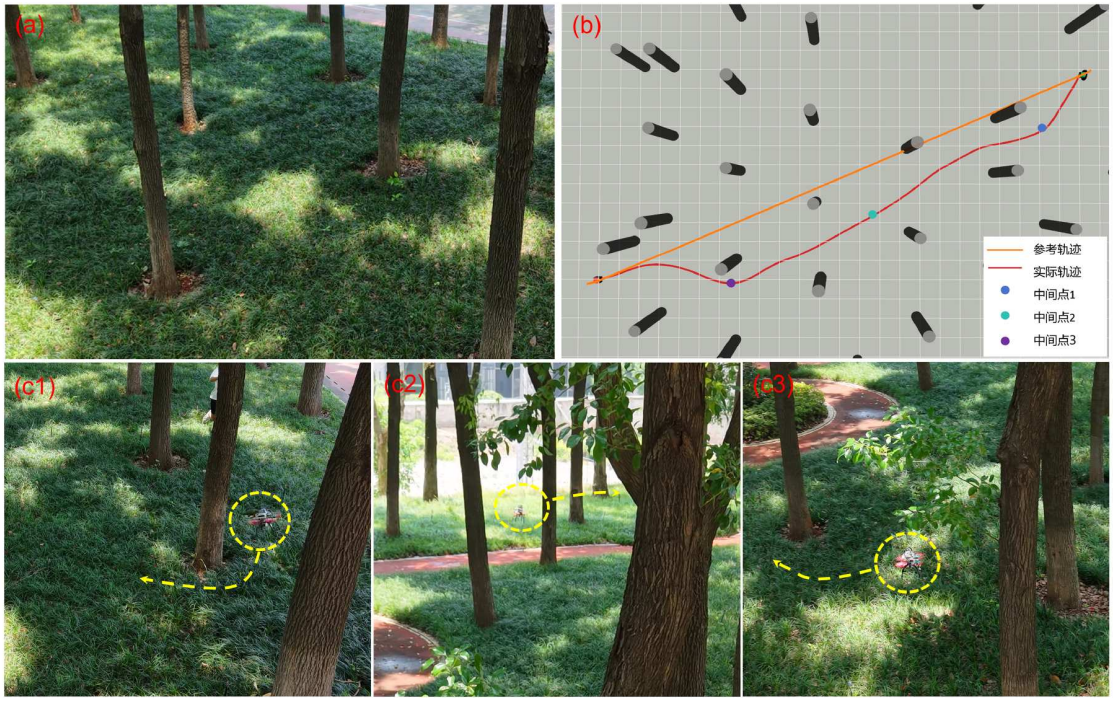


图 11 (网络版彩图)四旋翼无人机在户外树林环境中快速安全飞行. (a) 飞行测试的森林场景; (b) 典型飞行试验中的三维轨迹; (c1)~(c3) 三个不同时刻的飞行截图
Figure 11 (Color online) Quadrotor flies quickly and safely in outdoor forest environments. (a) The forest scene of the flight test; (b) three-dimensional trajectories in typical flight tests; (c1)–(c3) three screenshots of the flight at different moments.

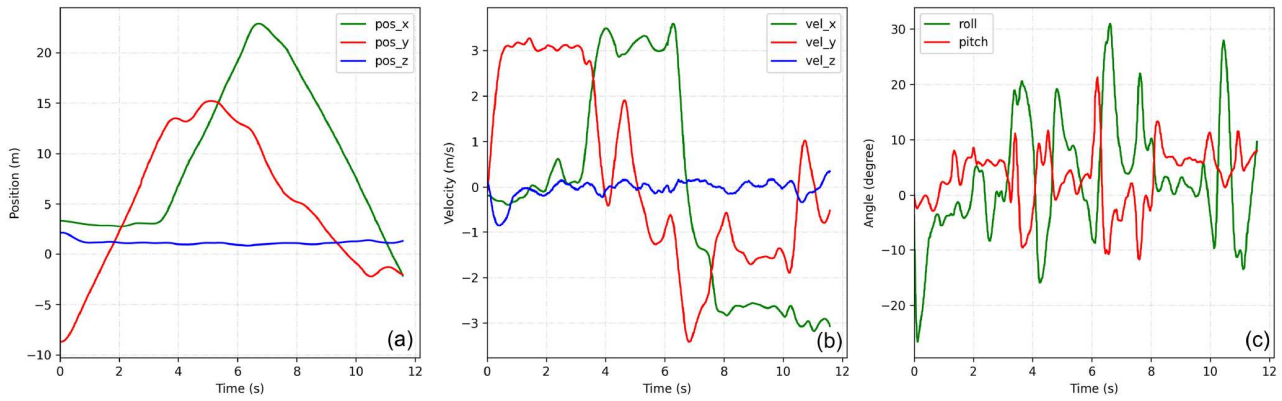


图 12　(网络版彩图)室外飞行试验的状态曲线. (a) 飞行过程中的无人机位置变化情况; (b) 飞行过程中的无人机速度变化情况; (c) 飞行过程中的无人机滚转角和俯仰角变化情况

Figure 12　(Color online) State curves for outdoor flight experiments. (a) The position changes of the unmanned aerial vehicle during flight; (b) the speed changes of the unmanned aerial vehicle during flight; (c) the changes in the roll Angle and pitch Angle of the unmanned aerial vehicle during flight.

展示了图11(b)中三个中间点处的飞行截图.

图12展示了飞行过程中的状态变化曲线. 位置曲线再次证实了轨迹的平滑性. 速度曲线表明, 在整个飞行过程中, 无人机的前向速度始终保持在较高水平(平均速度达到3.5 m/s以上), 仅在执行避障机动时有平缓的动态调整, 这证明了MPCC以最大化任务进度为目标的有效性. 同时, 欧拉角曲线平稳且有界, 表明MPCC求解生成的控制指令是动力学可行且能够被底层飞控精确跟踪的. 这验证了我们整个控制框架的内外环协同工作的有效性, 从高层决策到底层控制能够稳定执行.

综上所述, 物理试验结果有力地证明了本文提出的整体框架能够在计算资源和功耗受限的机载平台上实时、稳定地运行. 它成功地将仿真环境中观察到的快速、安全、平滑的避障能力迁移到了充满不确定性的真实物理世界中, 充分验证了本研究成果在现实场景中的应用潜力和价值.

5　结论与讨论

本文针对四旋翼在未知复杂环境中实现快速安全飞行的核心挑战, 提出了一种融合实时视觉感知、MPCC与CBF的统一控制框架. 通过将局部规划与飞行控制整合于单个非线性优化问题中, 并引入前瞻性的CBF安全约束, 本框架能够在无先验地图的条件下, 仅依赖机载视觉输入, 实时生成兼顾速度、安全与动力学可行性的控制指令. 充分的仿真与物理平台实验结果表明, 该框架不仅在飞行速度和任务成功率等关键性能指标上显著优于主流的分层式方法, 更重要的是, 它从根本上克服了传统方法因高延迟和误差累积导致的“走-停”式非流畅飞行问题, 实现了真正意义上的快速、平滑、连贯的自主避障飞行.

本框架的优势主要源于其内在的协同设计. 首先, MPCC通过最大化任务进度的优化目标, 赋予了四旋翼主动发掘自身动力学潜能以缩短任务时间的能力, 使其从被动的轨迹跟踪者转变为智能的决策者. 其次, CBF的引入在优化求解的层面提供了严格的安全保障, 它将避障行为从一种被动的、反应式的操作, 转变为一种主动的、前瞻性的决策, 使得飞行器能够以更平滑、更高效的方式绕过障碍. 最后, 轻量化的实时感知模块与高效的MPCC求解器相结合, 保证了整个系统的低延迟(<20 ms), 这是实现高速反应与流畅飞行的基础. 本研究的成功实践, 验证了将最优控制理论与现代感知技术深度融合, 是解决复杂环境下机器人自主导航问题的有效途径.

未来工作可以聚焦于三个方面. 在对障碍物的鲁棒感知方面, 可以探索多传感器融合技术; 在MPCC预测模型方面, 可以采用数据驱动在线辨识提高模型的预测精度; 在安全避障约束方面, 可以探索如何将动态障碍物的运动预测整合进CBF优化中.

致谢 感谢在飞行试验过程中协助我的刘运卓、张鸿、陈众楷、刘天晴、胡芸等同学, 在此深表谢忱.

参考文献

1 Zuo Z, Liu C, Han Q L, et al. Unmanned aerial vehicles: Control methods and future challenges. IEEE CAA J Autom Sin, 2022, 9: 601–614
2 Brunke L, Greeff M, Hall A W, et al. Safe learning in robotics: From learning-based control to safe reinforcement learning. Annu Rev Control Robot Auton Syst, 2022, 5: 411–444
3 Zhou X, Wang Z, Ye H, et al. EGO-planner: An ESDF-free gradient-based local planner for quadrotors. IEEE Robot Autom Lett, 2021, 6: 478–485
4 Jin Y, Liu X, Shao Y, et al. High-speed quadrupedal locomotion by imitation-relaxation reinforcement learning. Nat Mach Intell, 2022, 4: 1198–1208
5 Song Y, Romero A, Müller M, et al. Reaching the limit in autonomous racing: Optimal control versus reinforcement learning. Sci Robot, 2023, 8: eadg1462
6 Falanga D, Foehn P, Lu P, et al. PAMPC: Perception-aware model predictive control for quadrotors. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). 2018. 1–8
7 Lam D, Manzie C, Good M. Model predictive contouring control. In: Proceedings of the IEEE Conference on Decision and Control (CDC). 2010. 6137–6142
8 Jang I, Kim H J. Safe control for navigation in cluttered space using multiple Lyapunov-based control barrier functions. IEEE Robot Autom Lett, 2024, 9: 2056–2063
9 Wang Z, Zhou X, Xu C, et al. Geometrically constrained trajectory optimization for multicopters. IEEE Trans Robot, 2022, 38: 3259–3278
10 Ren Y, Zhu F, Liu W, et al. Bubble planner: Planning high-speed smooth quadrotor trajectories using receding corridors. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). 2022. 6332–6339
11 Kaufmann E, Bauersfeld L, Loquercio A, et al. Champion-level drone racing using deep reinforcement learning. Nature, 2023, 620: 982–987
12 Wang M, Jia S, Niu Y, et al. Agile flights through a moving narrow gap for quadrotors using adaptive curriculum learning. IEEE Trans Intell Veh, 2024, 9: 6936–6949
13 Wang B D, Zhu H, Zhu X Z, et al. Agile control of a quadrotor with disturbance observation based on graph neural networks (in Chinese). Sci Sin Tech, 2025, 55: 1490–1501 [王宝栋, 祝海, 朱效洲, 等. 时延条件下多无人机系统鲁棒编队跟踪控制. 中国科学: 技术科学, 2025, 55: 1490–1501]
14 Wang M, Wang B, Niu Y, et al. PI-WAN: A physics-informed wind-adaptive network for quadrotor dynamics prediction in unknown environments. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). 2025. 5413–5419
15 Brito B, Floor B, Ferranti L, et al. Model predictive contouring control for collision avoidance in unstructured dynamic environments. IEEE Robot Autom Lett, 2019, 4: 4459–4466
16 Romero A, Sun S, Foehn P, et al. Model predictive contouring control for time-optimal quadrotor flight. IEEE Trans Robot, 2022, 38: 3340–3356
17 Krinner M, Romero A, Bauersfeld L, et al. MPCC++: Model predictive contouring control for time-optimal flight with safety constraints. In: Proceedings of the Robotics: Science and Systems Conference (RSS). 2024. 1–12
18 Qiu J, Liu Q, Qin J, et al. PE-planner: A performance-enhanced quadrotor motion planner for autonomous flight in complex and dynamic environments. IEEE Robot Autom Lett, 2024, 9: 5879–5886
19 Chen S, Li Y, Lou Y, et al. Learning agile quadrotor flight in restricted environments with safety guarantees. IEEE Trans Intell Veh, 2024, 9: 5642–5649
20 Xu Z, Han X, Shen H, et al. NavRL: Learning safe flight in dynamic environments. IEEE Robot Autom Lett, 2025, 10: 3668–3675
21 Saviolo A, Picello N, Verma R, et al. Reactive collision avoidance for safe agile navigation. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). 2025. 16125–16132
22 Zhou B, Gao F, Wang L, et al. Robust and efficient quadrotor trajectory generation for fast autonomous flight. IEEE Robot Autom Lett, 2019, 4: 3529–3536
23 Xu W, Cai Y, He D, et al. FAST-LIO2: Fast direct LiDAR-inertial odometry. IEEE Trans Robot, 2022, 38: 2053–2073

Fast and safe obstacle avoidance flight control for quadrotors in unknown environments

WANG MengYun[1], WANG Bo[1], NIU YiFeng[1], YU HongBo[1], WANG Chang[1] & ZHANG Tao[2]

[1] *College of Intelligence Science and Technology, National University of Defense Technology, Changsha 410073, China*
[2] *Department of Automation, Tsinghua University, Beijing 100080, China*

Achieving fast, safe, and autonomous flight for quadrotors in unknown and complex environments is a critical challenge for their deployment in time-sensitive missions such as search and rescue or exploration. Addressing the limitations of traditional hierarchical navigation methods, including error accumulation and suboptimal flight policies, as well as the inability of end-to-end learning methods to provide strict safety guarantees, this paper proposes a unified flight control framework that integrates real-time perception, model predictive contouring control (MPCC), and control barrier functions (CBF). The framework is centered around MPCC, which unifies local trajectory planning and flight control into a single nonlinear optimization problem, achieving near-time-optimal fast flight by maximizing task progress. Concurrently, it innovatively combines the reactive safety assurance of CBFs with the proactive optimal control of MPCC. The system relies solely on real-time visual input from an onboard depth camera to identify high-risk obstacles and construct CBF safety constraints online. This enables the real-time generation of control commands that balance speed, safety, and dynamic feasibility, all without requiring a global collision-free reference path or a prior map. Extensive comparative experiments in both simulation and on a physical platform validate the effectiveness of the proposed method. The results demonstrate that, compared to mainstream hierarchical methods, our approach significantly improves the average flight speed and success rate in complex, obstacle-rich environments. Furthermore, it effectively overcomes the "stop-and-go" phenomenon inherent in traditional methods, achieving substantially faster, smoother, and more coherent flight trajectories. Successful experiments in an outdoor forest setting on a real quadrotor platform further validate the real-time performance and robustness of the proposed method on resource-constrained onboard systems. This research provides an efficient, reliable, and unified solution for enabling quadrotors to perform fast and safe autonomous tasks in unknown environments.

quadrotor, fast flight, safe obstacle avoidance, control barrier function, model predictive contouring control

doi: 10.1360/SST-2025-0301