http://bhxb.buaa.edu.cn jbuaa@buaa.edu.cn
DOI: 10.13700/j.bh.1001-5965.2024.0662

不确定环境下基于POMDP和参数化MPC的无人机在线路径规划

院金彪[1]，刘海艳[2]，马博迪[3]，赵玏洋[4]，高红岗[1]，刘贞报[1,*]

(1. 西北工业大学 民航学院，西安 710072；2. 上海民用航空控制与导航系统有限公司，上海 200241；3. 长安大学 信息学院，西安 710016；4. 深圳大学 建筑与城市规划学院，深圳 518060)

摘 要：无人机在未知环境中自主飞行时，定位与感知层面存在着难以消除的不确定性因素，开展航路规划方法以在不确定性干扰下增强航路有效性并提升无人机（UAVs）目标跟踪能力具有重要意义。当前，名义信念状态优化（NBO）为目标跟踪的不确定性提供了理论依据，基于管道的模型预测控制（Tube-based-MPC）则为规划与控制的分离架构提供了支持，从而使模型预测控制（MPC）能够应用于实时航路规划系统。基于此，提出一种在线航路规划方法以解决未知环境下的目标跟踪、避障问题。将航路规划问题抽象成一个部分可观测马尔可夫决策过程（POMDP）模型，并运用卡尔曼滤波对置信状态进行估计，将目标定位的不确定性度量融入到框架中；基于简化的三维运动模型构造了Tube-based-MPC优化框架对POMDP进行求解，通过构建包含置信状态和未知障碍物的非线性优化问题，实现未来一定连续时间跨度的航路规划。在优化问题中充分考虑无人机和障碍物形状，并引入全维度避障距离度量，显著拓展优化问题的可行域。引入Q-Learning强化学习框架，以MPC作为策略函数逼近器，提升目标跟踪的闭环性能。仿真实验和硬件在环实验验证了所提方法的有效性。

关 键 词：部分可观测马尔可夫决策过程；基于管道的模型预测控制；航路规划；强化学习；目标跟踪

中图分类号：V249

文献标志码：A **文章编号**：1001-5965(2026)11-3941-17

在当今快速发展的科技领域，无人机(unmanned aerial vehicles, UAVs)技术正迅速成为解决多领域挑战的前沿工具之一。随着无人机的广泛应用，如救援任务[1-2]、智能油气管道巡检[3]、探索勘测和灾害响应[4-5]等，对其智能化和自主性的需求日益迫切。面对不断增长的需求和挑战，研究者们致力于提升无人机的感知、规划和控制能力，以确保其在未知环境中能够高效、安全地执行任务。

航路规划作为无人机自主飞行的关键技术，通过综合考虑环境中的障碍物、地形等因素，能够为无人机生成安全可行的飞行路径，避免与障碍物发生碰撞，最大程度降低事故风险，也可以根据具体环境特征调整飞行路径，使无人机能够适应不同的工作场景，提高其应用的多样性。尤其在进行远距离，大范围作业时，飞手对于飞行环境中的阵风、团雾、移动威胁等信息的掌握是不准确的(环境状态

收稿日期：2024-09-06；录用日期：2024-11-15；网络出版时间：2024-11-22 17:09
网络出版地址：link.cnki.net/urlid/11.2625.V.20241122.1600.001
基金项目：国家自然科学基金 (52072309,12102345)；陕西省重点研发计划 (2019ZDLGY14-02-01,2022FP-40)；深圳市基础研究资助项目 (JCYJ20190806152203506)；航空科学基金 (ASFC-2018ZC53026)；苏州市科学技术局资助项目 (ZXL2023177)；太仓市基础研究计划资助项目 (TC2023JC13)

*通信作者. E-mail：liuzhenbao@nwpu.edu.cn

引用格式：院金彪，刘海艳，马博迪，等. 不确定环境下基于POMDP和参数化MPC的无人机在线路径规划[J]. 北京航空航天大学学报，2026，52（11）：3941-3957. Yuan J B，Liu H Y，Ma B D，et al. UAV online path planning based on POMDP and parameterized MPC in uncertain environments[J]. Journal of Beijing University of Aeronautics and Astronautics，2026，52（11）：3941-3957（in Chinese）.

的不确定性),再加上机载传感器、执行器动作和飞行动力学方面的不确定性,当面临这些突发风险时,需要无人机进行在线航路规划以完成自主避障。

部分可观测马尔可夫决策过程(partially observable Markov decision process, POMDP)为不确定性下的规划提供了一种数学上的严格方法[6]。POMDP为基于智能体对状态空间的不完全观测而进行的一系列马尔可夫决策过程,其通过置信度描述各个可能状态的概率值,从而获得某一规划时刻的置信状态,进而反复和环境交互推理得到动作序列,其在无人机领域也有一定的应用。Trotti等[7]提出一种轨迹规划方法,使无人机能够到达目标的同时避开禁飞区域,优化燃料消耗、飞行时间等,并对飞行角度进行约束,研究中考虑了固定翼飞机的动力学,并添加了不确定性模型误差以考虑近似情况。该研究基于蒙特卡罗树搜索(Monte Carlo tree search, MCTS)进行在线求解,减轻了POMDP模型求解的复杂性。Morere等[8]提出一种基于POMDP框架并采用贝叶斯优化(Bayesian optimization, BO)的轨迹规划(简称BO-POMDP)算法,所得BO-POMDP算法通过使用MCTS进行求解,并设计了平衡探索和利用奖励函数,无人机空对地监控实验验证表明,BO-POMDP算法具有较好的性能。Deng等[9]针对无人机辅助的车辆通信问题进行了研究,采用联合优化无人机轨迹设计和车辆分配的方案,将车辆轨迹预测和联合优化问题建模为POMDP。为解决POMDP的非马尔可夫性,设计基于深度Q网络(deep Q-network, DQN)算法和长短时记忆(long short-term memory, LSTM)网络层,构建了一个新的深度循环Q网络(deep recurrent Q-network, DRQN)框架,仿真结果表明,该基于DRQN框架的方案收敛速度快,且在总频谱效率方面优于基准方案。Aniceto dos Santos等[10]针对无人机在未知环境中收集信息即信息路径规划(informative path planning, IPP)和自主探索(autonomous exploration, AE)问题,提出一种使用具有POMDP的顺序BO进行森林监测的探索和IPP的建模策略,基于BO-POMDP在不确定性环境下进行研究,开发了一个顺序决策模型。

以上研究中针对POMDP的求解主要体现为2类,即基于MCTS[7-8,10]的算法和基于强化学习[9]的算法。尽管对POMDP在无人机领域的应用进行了研究和探索,取得了一定的效果,但针对MCTS类算法,其性能直接受动作的离散程度影响,权衡决策树的复杂度和算法搜索时间仍是一个难点。针对强化学习类算法,由于对环境的依赖性较高,不同场景间的通用性差,不利于无人机在大范围3D空间高效的航路规划需求。

此外,针对POMDP的计算复杂度问题(随着求解深度增加,规划范围呈指数级增长,计算效率低),很多学者对求解效率进行了改进。目前常见的求解方式有部分可观察蒙特卡罗规划(partially observable Monte Carlo planning, POMCP)算法[11]、确定稀疏部分可观测树(determinized sparse partially observable tree, DESPOT)算法[12-13]、自适应策略树(adaptive behavior tree, ABT)算法[14]、DESPOT-α算法[15]、实时调整POMDP解决方案(简称TAPIR)算法[16]、多极决策(multi-policy decision making, MPDM)算法[17]、领域特定的闭环策略树(domain-specific closed-loop policy tree, DCPTree)算法[18]、强化学习算法[19-20]、名义信念状态优化(nominal belief state optimization, NBO)算法[21]等。其中,文献[11-16]算法都适用于离散状态和离散动作空间的规划,通常会构造一棵策略树,在此基础上按照一定的准则搜索策略,并在理论上提供精确解,文献[17-18]算法是预先给出未来一定规划周期内可能的若干方案,从而构造离散的状态-动作或离散的状态-动作-观察树,大大减少了无效策略,在结构化环境中具有较好的应用。文献[19-20]算法基于强化学习框架通过与环境不断交互学习不同置信状态下可能的动作方案,其优势为可以学习到较优的离散或连续动作方案,但对环境依赖性强。此外,针对无人机在开放空间中的航路规划应用,离散化的动作空间会大大缩小无人机的求解可行域,且离散粒度太大会增加POMDP模型的复杂程度,造成维度灾难,增大求解难度。NBO算法作为POMDP的近似求解方案,具有很好的处理连续动作的能力,对于无人机在不确定性环境中的在线决策有很好的辅助作用,且相比其他近似算法计算效率高。文献[22-24]在NBO算法基础上进一步拓展,将动力学模型及各种感兴趣的特征纳入框架,展示了算法的强大和灵活性,并对多种跟踪控制和规避威胁场景进行了研究,证明了其实用性。

模型预测控制(model predictive control, MPC)作为一种流行的控制算法框架,能够轻松构建涵盖多种约束的复杂优化问题,并实现对未来一定时间跨度的在线轨迹生成,以应对未知的环境变化,在航路规划领域应用广泛。本文通过MPC框架对NBO算法进行优化问题的构建。然而,在该优化问题中,由于目标函数、约束条件及状态转移方程中的系数通常保持不变,导致其本身存在的误差难以得到改善,进而影响目标跟踪的闭环性能。近年

来, 基于 MPC 的强化学习[25-29]为进一步提升求解质量带来了可能。其核心在于将 MPC 作为策略输出逼近器, 在强化学习框架下, 利用经验池样本逐步调整 MPC 中的参数, 以实现优化的目的。与传统将深度神经网络(deep neural network, DNN)作为策略逼近器的强化学习框架相比, 基于 MPC 的强化学习具有更少的参数, 适应性更强。

本文着眼于未知环境中目标和模型的不确定性, 同时结合无人机作业时可能遭遇的未知静态或动态威胁场景, 提出一种在线航路规划方法。该方法充分考虑了目标状态的不确定性、动力学模型的不确定性, 以及以未知多面体形式呈现的静态障碍物和动态障碍物。本文方法基于 NBO 的理念完成了优化问题建模, 借助 MPC 框架融入了基于全维度距离度量[30]的避障约束条件, 并引入基于 Q-Learning 的强化学习框架来优化模型参数, 旨在提升目标跟踪的闭环性能; 利用 Casadi 求解该优化问题, 获取了不同置信状态下的连续动作输出, 达成了在连续动作空间中进行一定时间跨度的在线轨迹优化。为便于表述, 将本文提出的基于 Q-Learning 的 NBO 算法命名为 QL-NBO。本文的主要创新点包括:

1) 构建了基于 MPC 的 NBO 优化求解框架, 充分发挥 NBO 在处理目标状态不确定性方面的优势, 达成了未知环境中的在线轨迹规划。在这一框架内, 针对避障问题采用了考虑障碍物形状的全维度距离度量[30]方法, 相较于将障碍物等效为球形或椭球形等标准几何体的方式, 该方法扩大了问题求解的可行域, 且相比基于对偶变换的避障[31-32]轨迹规划, 具备更快的求解速度。

2) 引入了基于 MPC 的强化学习算法, 对优化模型进行参数化处理, 将 MPC 用作策略函数逼近器, 于 Q-Learning 框架中实时调整模型的权重系数, 有效改善了障碍物增多时避障航路有效性(到障碍物距离大于 0)降低的状况, 提高了 MPC 优化模型的求解质量及目标跟踪的闭环性能。

1　无人机系统

1.1　问题描述

本文选择电动垂直起降飞行器(electric vertical take-off and landing, eVTOL)作为研究平台, 重点考虑其在固定翼模式下的运动特性。设定的场景为复杂的三维环境, 包含静态和动态障碍物, 这些障碍物在规划初期是未知的。因此, 需要通过在线决策实时生成路径, 引导无人机在避障的同时尽可能完成设定任务。此外, 为确保路径的可行性, 还需考虑无人机的运动学约束。因此, 在线航路规划问题成为一个需要同时满足多种约束条件的优化问题。

1.2　三维运动学模型

针对航路规划本文只考虑运动学模型, 该模型主要用来约束航点以满足无人机运动学特点, 从而引导无人机以较小代价完成自主导航。通常三维空间下的运动学模型为[33]

$$\begin{cases} \dot{x} = v\cos\phi\cos\varphi \\ \dot{y} = v\cos\phi\sin\varphi \\ \dot{z} = v\sin\phi \\ \dot{v} = u \\ \dot{\varphi} = \omega_\varphi \\ \dot{\phi} = \omega_\phi \end{cases} \tag{1}$$

式中: (x,y,z)为无人机的实时位置; v为无人机的巡航速度; ϕ为无人机的航迹倾角; φ为无人机的航迹偏角; ω_ϕ和ω_φ分别为航迹倾角速度和航迹偏角速度; u为加速度, 即无人机的合加速度。

为确保生成路径能够方便无人机以较小的代价跟踪, 该运动学模型需要遵守如下可行性约束:

$$\begin{cases} \omega_{\phi\min} < \omega_\phi < \omega_{\phi\max} \\ \omega_{\varphi\min} < \omega_\varphi < \omega_{\varphi\max} \\ \phi_{\min} < \phi < \phi_{\max} \\ u_{\min} \leqslant u \leqslant u_{\max} \\ v_{\min} \leqslant v \leqslant v_{\max} \end{cases} \tag{2}$$

2　POMDP 模型设计

POMDP 模型的作用是在环境状态不确定、动作不确定的情况下能够找到一组动作序列, 使该智能体在这组序列下的价值回报最大[6], 典型的 POMDP 模型由一个 7 元组 $<S,A,Z,T,O,R,\gamma>$组成。其中, S为状态空间, A为动作空间, Z为观察空间, T为状态转移条件概率 $T(S,A,S')=P(S'|S,A)$, S'为转移后的状态空间, O为观察条件转移概率 $O(S',A,Z)=P(\boldsymbol{z}|\boldsymbol{s}',\boldsymbol{a})$, $\boldsymbol{a}$为动作, $\boldsymbol{s}'$为转移后的状态, $\boldsymbol{z}$为观察, 每当执行一个动作后, 环境会给智能体返回一个奖励R, 从当前时刻开始, 向前执行多步动作后, 每一步的奖励会以折扣γ^H递减, 其中, H为规划时段跨度, 且$0\leqslant\gamma<1$。

2.1　置信状态

考虑实际环境状态的不完全观测, POMDP 模型引入了置信状态的概念, 置信状态可以描述为不同时刻下所有状态的概率分布。令时刻t时$\boldsymbol{s}$的置信状态为$\boldsymbol{b}_t(\boldsymbol{s})$, 当执行动作$\boldsymbol{a}_t$后得到观察为$\boldsymbol{z}_t$, 则$t+1$时刻的置信状态为

$$\boldsymbol{b}_{t+1}(\boldsymbol{s}') = \eta O(\boldsymbol{s}',\boldsymbol{a}_{t+1},\boldsymbol{z}_{t+1})\sum_{\boldsymbol{s}\in S} T(\boldsymbol{s},\boldsymbol{a}_{t+1},\boldsymbol{s}')\boldsymbol{b}_t(\boldsymbol{s}) \tag{3}$$

式中: $\eta\in(0,1)$; 转移概率T遵循贝叶斯[34]规则。

设置信状态更新函数为 τ，则有 $\boldsymbol{b}_{t+1}=\tau(\boldsymbol{b}_t,\boldsymbol{a}_{t+1},\boldsymbol{z}_{t+1})=\cdots=\tau(\cdots\tau(\tau(\boldsymbol{b}_0,\boldsymbol{a}_1,\boldsymbol{b}_1),\boldsymbol{a}_2,\boldsymbol{b}_2),\cdots,\boldsymbol{a}_t,\boldsymbol{z}_t)$，从中可以看出每个置信状态都包含了从最初时刻到当前时刻所有历史动作和观察信息 $(\boldsymbol{a}_1,\boldsymbol{z}_1,\boldsymbol{a}_2,\boldsymbol{z}_2,\cdots,\boldsymbol{a}_{t+1},\boldsymbol{z}_{t+1})$，反映出置信状态在转移过程中是根据历史观测和动作来推断当前状态的概率分布。

当智能体在执行策略 π 时，H 个规划时间段跨度下，智能体获得的期望总折扣奖励为

$$V_\pi(\boldsymbol{b})=E\left(\sum_{t=0}^{H}\gamma^t R(\boldsymbol{s}_t,\pi(\boldsymbol{b}_t))|\boldsymbol{b}_0=\boldsymbol{b}\right) \tag{4}$$

式中：E 为期望。

某一置信状态下的最优动作可以通过评价价值最大时所执行的动作来确定：

$$V^*(\boldsymbol{b})=\max_{\boldsymbol{a}\in A}\left\{\sum_{s\in S}\boldsymbol{b}(\boldsymbol{s})R(\boldsymbol{s},\boldsymbol{a})+\gamma\sum_{z\in Z}P(z|\boldsymbol{b},\boldsymbol{a})V^*(\tau(\boldsymbol{b},\boldsymbol{a},z))\right\} \tag{5}$$

2.2 组成成分对应

本文的任务是在未知场景下能够使无人机考虑环境状态的不确定性避开一切静态和动态障碍。将该问题映射到 POMDP 模型中需要在如下几方面进行处理。

1）状态 S。状态是智能体在环境中所处的当前情况或配置，规划时间跨度 k 的状态可表示为 $\boldsymbol{s}_k=(\boldsymbol{X}_k,\boldsymbol{G}_k,\boldsymbol{\xi}_k,\boldsymbol{P}_k,\boldsymbol{O}_{\mathrm{bs}k})$，其中，$\boldsymbol{X}_k$ 为无人机运动学模型状态 $(\boldsymbol{x}_k,\boldsymbol{y}_k,\boldsymbol{z}_k,\boldsymbol{v}_k,\phi_k,\varphi_k)$，$\boldsymbol{G}_k$ 为期望的无人机目标状态 $(\boldsymbol{x}_G,\boldsymbol{y}_G,\boldsymbol{z}_G,\boldsymbol{v}_G,\phi_G,\varphi_G)$，$\boldsymbol{\xi}_k$ 和 $\boldsymbol{P}_k$ 分别为跟踪状态下后验均值向量和协方差矩阵，其计算可参考卡尔曼滤波算法[35-36]，$\boldsymbol{O}_{\mathrm{bs}k}$ 为时间跨度 k 下的观察空间。

2）动作 A。这里的动作为运动学模型的输入，因此，第 k 个规划时间跨度的动作定义为 $\boldsymbol{a}_k=(\omega_{\phi_k},\omega_{\varphi_k},\boldsymbol{u}_k)$。

3）观察空间 Z 和条件概率 O。考虑测量的不确定性，第 k 个规划时间跨度的观测 $\boldsymbol{z}_k$ 可通过式(6)计算：

$$\begin{cases}\boldsymbol{z}_k^{\boldsymbol{X}}=\boldsymbol{X}_k\\ \boldsymbol{z}_k^{\boldsymbol{G}}=\boldsymbol{G}_k+\boldsymbol{e}_k\\ \boldsymbol{z}_k^{\boldsymbol{\xi}}=\boldsymbol{\xi}_k\\ \boldsymbol{z}_k^{\boldsymbol{P}}=\boldsymbol{P}_k\end{cases} \tag{6}$$

式中：$\boldsymbol{z}_k^{\boldsymbol{X}}$ 为系统状态观测值；当目标可视时，$\boldsymbol{z}_k^{\boldsymbol{G}}$ 为目标状态观测值，否则，$\boldsymbol{z}_k^{\boldsymbol{G}}$ 不可测量；$\boldsymbol{e}_k$ 为测量噪声；$\boldsymbol{z}_k^{\boldsymbol{\xi}}$ 和 $\boldsymbol{z}_k^{\boldsymbol{P}}$ 为跟踪状态观测值。

可以看出观察空间 $\boldsymbol{z}_k^{\boldsymbol{X}}$、$\boldsymbol{z}_k^{\boldsymbol{\xi}}$ 和 $\boldsymbol{z}_k^{\boldsymbol{P}}$ 与状态空间是相等的，假定了实际当中无人机的状态和跟踪状态是完全可观察的，而目标状态是不完全观察的，这也是不确定性来源。

4）状态转移规则。当无人机从某一状态 $\boldsymbol{X}_k$ 执行动作 $\boldsymbol{a}_{k+1}$ 后环境会返回新的状态 $\boldsymbol{X}_{k+1}$，其中，系统状态转移通过运动学模型来确定。目标状态转移 $\boldsymbol{G}_{k+1}=\boldsymbol{F}\boldsymbol{G}_k+\boldsymbol{v}_k$，其中，矩阵 $\boldsymbol{F}$ 为跟踪目标的状态转移矩阵，$\boldsymbol{v}_k$ 表示测量中的随机噪声。跟踪状态转移则通过标准卡尔曼滤波算法确定。

5）奖励函数。无人机在总时间跨度为 H 的规划周期内，执行每个动作都会产生一个奖励，这些奖励的总和就是总时间跨度内的奖励，本文以无人机当前位置到目标位置的距离作为代价，当时间跨度为 k 时的奖励值计算如下：

$$C(\boldsymbol{s}_k,\boldsymbol{a}_k)=E_{e,v}\left(\|\boldsymbol{G}_{k+1}-\boldsymbol{\xi}_{k+1}\|_2^2\,|\,\boldsymbol{s}_k,\boldsymbol{a}_k\right) \tag{7}$$

式中：$C(\boldsymbol{s}_k,\boldsymbol{a}_k)$ 为跟踪状态和目标状态之间的均方差的期望。

6）折扣回报 γ。折扣回报可自行设定，其范围为[0,1)之间。

当确定了 POMDP 模型的各个组成成分后就可以进行置信状态的计算，第 k 个时间跨度的置信状态为

$$\boldsymbol{b}_k=\left(\boldsymbol{b}_k^{\boldsymbol{X}},\boldsymbol{b}_k^{\boldsymbol{G}},\boldsymbol{b}_k^{\boldsymbol{\xi}},\boldsymbol{b}_k^{\boldsymbol{P}}\right) \tag{8}$$

式中：$\boldsymbol{b}_k^{\boldsymbol{X}}(\boldsymbol{s})=\delta(\boldsymbol{X}-\boldsymbol{X}_k)$，$\delta(\cdot)$ 为概率密度函数；$\boldsymbol{b}_k^{\boldsymbol{\xi}}(\boldsymbol{s})=\delta(\boldsymbol{\xi}-\boldsymbol{\xi}_k)$；$\boldsymbol{b}_k^{\boldsymbol{P}}(\boldsymbol{s})=\delta(\boldsymbol{P}-\boldsymbol{P}_k)$；而 $\boldsymbol{b}_k^{\boldsymbol{G}}(\boldsymbol{s})$ 可以通过贝叶斯公式计算，也可以通过对目标点或障碍物的轨迹预测进行计算。

3 NBO 近似解和 MPC 优化

关于 POMDP 的求解，其核心目标在于探寻一组动作序列，以较小代价或奖励较大的决策方式完成任务。以无人机的应用场景来说，鉴于环境变化迅速且对实时性有着较高要求，无需开展过长时间跨度的预测。故而，能够将目标函数设定为规划时间跨度总长为 H 的累积期望：

$$J_H=E\left(\sum_{k=0}^{H-1}C(\boldsymbol{s}_k,\boldsymbol{a}_k)\right) \tag{9}$$

在每个时间跨度 k 下，动作的选择都需要依赖之前所有历史状态和动作。假设问题(9)基于状态 $\boldsymbol{s}_k(k=0,1,\cdots,H-1)$ 存在最优动作序列，则相对于置信状态 $\boldsymbol{b}_k(k=0,1,\cdots,H-1)$ 来说，该最优动作序列也一样存在，因此，可以将式(9)用置信状态和动作对表示为

$$J_H=E\left(\sum_{k=0}^{H-1}c(\boldsymbol{b}_k,\boldsymbol{a}_k)|\boldsymbol{b}_0\right) \tag{10}$$

式中：$c(\boldsymbol{b}_k,\boldsymbol{a}_k)=\int C(\boldsymbol{s}_k,\boldsymbol{a}_k)\boldsymbol{b}(\boldsymbol{s})\mathrm{d}\boldsymbol{s}$，即置信状态中每一状态执行动作 $\boldsymbol{a}_k$ 后的期望，这里用积分表示是因为置信状态本身的连续性。设最优策略为 π_k^*，则有动

作$\boldsymbol{a}_k^*=\pi_k^*(\boldsymbol{b}_k)$。进一步, 式(10)根据贝尔曼方程可写成如下形式:

$$J_H^*(\boldsymbol{b}_k)=c(\boldsymbol{b}_k,\boldsymbol{a})+E\left(J_{H-1}^*(\boldsymbol{b}_{k+1})|\boldsymbol{b}_k,\boldsymbol{a}\right) \tag{11}$$

将式(10)等号右边部分表示为动作价值, 则有

$$Q_H(\boldsymbol{b}_k,\boldsymbol{a})=c(\boldsymbol{b}_k,\boldsymbol{a})+E\left(J_{H-1}^*(\boldsymbol{b}_{k+1})|\boldsymbol{b}_k,\boldsymbol{a}\right) \tag{12}$$

因此, 时间跨度为k时, 最优策略为

$$\pi_k^*(\boldsymbol{b}_k)=\underset{\boldsymbol{a}}{\operatorname{argmin}}Q_H(\boldsymbol{b}_k,\boldsymbol{a}) \tag{13}$$

所有时间跨度对应的最优动作$\boldsymbol{a}$组成的序列则为整个规划周期的最优策略。

3.1 NBO 近似

由于置信状态本身及无人机运动学模型中动作空间的连续性, 造成 POMDP 问题的维度灾难, 增加了求解难度。正如引言中分析的, POMDP 的求解采用当前一些较新的求解器时, 由于都对动作状态进行了离散化处理, 但在大范围空间中进行航路规划时, 随着环境中的障碍物数量的增加, 问题的维度呈指数上升, 导致求解慢, 不适合无人机在空中高速运动的场景。因此, 本文采用 NBO 算法, 实现了对连续动作的优化。该算法对目标函数做了如下近似:

$$J_H(\boldsymbol{b}_0)\approx\sum_{k=0}^{H-1}c\left(\hat{\boldsymbol{b}}_k,\boldsymbol{u}_k\right) \tag{14}$$

式中: $\hat{\boldsymbol{b}}_k$为名义信念状态, 通过卡尔曼滤波算法中的$\left(\hat{\boldsymbol{\xi}}_k,\hat{\boldsymbol{P}}_k\right)$进行更新:

$$\hat{\boldsymbol{b}}_k^G(\boldsymbol{G})=\mathcal{N}\left(\boldsymbol{G}-\hat{\boldsymbol{\xi}}_k,\hat{\boldsymbol{P}}_k\right) \tag{15}$$

式中: $\hat{\boldsymbol{\xi}}_k$和$\hat{\boldsymbol{P}}_k$的更新如下:

$$\begin{cases}\hat{\boldsymbol{\xi}}_{k+1}=\boldsymbol{F}_k\hat{\boldsymbol{\xi}}_k\\ \hat{\boldsymbol{P}}_{k+1}=\begin{cases}\left[\left(\boldsymbol{F}_k\hat{\boldsymbol{P}}_k\boldsymbol{F}_k^{\mathrm{T}}+\boldsymbol{Q}_k\right)^{-1}+\boldsymbol{S}_{k+1}\right]^{-1}\\ \left(\boldsymbol{F}_k\hat{\boldsymbol{P}}_k\boldsymbol{F}_k^{\mathrm{T}}+\boldsymbol{Q}_k\right)^{-1}\end{cases}\end{cases} \tag{16}$$

式中: $\hat{\boldsymbol{P}}_{k+1}$等式右侧上方表示目标可视, 下方为其他情况; $\boldsymbol{F}_k$为目标的状态转移矩阵; $\boldsymbol{Q}_k$为噪声矩阵; $\boldsymbol{S}_{k+1}=\boldsymbol{H}_{k+1}^{\mathrm{T}}\left[\boldsymbol{R}_{k+1}\left(\hat{\boldsymbol{\xi}}_k,\boldsymbol{X}_k\right)\right]^{-1}\boldsymbol{H}_{k+1}$, $\boldsymbol{H}_k$为观测矩阵, $\boldsymbol{R}_k$的计算式为

$$\boldsymbol{R}_k=\boldsymbol{M}_k\begin{bmatrix}\sigma_{\text{range}}^2(k)&0&0\\0&\sigma_{\text{angle_}\phi}^2(k)&0\\0&0&\sigma_{\text{angle_}\varphi}^2(k)\end{bmatrix}\boldsymbol{M}_k^{\mathrm{T}} \tag{17}$$

式中: $\sigma_{\text{range}}(k)=(p/100)r_{\text{eff}}(k)$; $\sigma_{\text{angle_}\phi}(k)=qr_{\text{eff}}(k)$; $\sigma_{\text{angle_}\varphi}(k)=\mu r_{\text{eff}}(k)$; p、q和μ分别为无人机当前位置、目标位置的距离和纵向方位角与横向方位角的不确定性度量, $r_{\text{eff}}(k)=\sqrt{r_k^2+\tilde{b}^2}$, r_k为当前位置到目标点的距离, $\tilde{b}$为非 0 常数, 表示距离测量误差; $\boldsymbol{M}_k$的计算式为

$$\boldsymbol{M}_k=\begin{bmatrix}\cos\phi_k&0&\sin\phi_k\\0&1&0\\-\sin\phi_k&0&\cos\phi_k\end{bmatrix}\begin{bmatrix}\cos\varphi_k&-\sin\varphi_k&0\\\sin\varphi_k&\cos\varphi_k&0\\0&0&1\end{bmatrix} \tag{18}$$

由式(16)可以看出名义误差协方差矩阵$\hat{\boldsymbol{P}}_{k+1}$取决于未来时刻对目标观测有效性, 即目标是否在感知范围内。当目标不在感知范围内时, 能通过假设目标在第 k+1 时间跨度的位置为$\hat{\boldsymbol{\xi}}_{k+1}$, 确保无人机跟踪状态和目标状态在该时间跨度下的均方误差最小, 因此, 有

$$c\left(\hat{\boldsymbol{b}}_k,\boldsymbol{u}_k\right)=\mathrm{Tr}\hat{\boldsymbol{P}}_{k+1} \tag{19}$$

式中: $\mathrm{Tr}\hat{\boldsymbol{P}}_{k+1}$为无人机对目标跟踪的不确定性度量, $\mathrm{Tr}\hat{\boldsymbol{P}}_{k+1}$中的不确定性最小同时也反应了跟踪状态和目标状态的均方误差最小。

此外, $\boldsymbol{X}_k$和$\boldsymbol{G}_k$的计算如下:

$$\begin{cases}\boldsymbol{X}_{k+1}=f(\boldsymbol{X}_k,\boldsymbol{a}_k)+\boldsymbol{n}_k & \boldsymbol{n}_k\sim\mathcal{N}(\boldsymbol{0},\boldsymbol{N}_k)\\ \boldsymbol{G}_{k+1}=\boldsymbol{F}_k\hat{\boldsymbol{\xi}}_k+\boldsymbol{m}_k & \boldsymbol{m}_k\sim\mathcal{N}(\boldsymbol{0},\boldsymbol{M}_k)\end{cases} \tag{20}$$

式中: $f(\boldsymbol{x}_k,\boldsymbol{a}_k)$为无人机系统的运动学模型; $\boldsymbol{N}_k$为系统误差矩阵。

3.2 Tube-based-MPC

文献 [21-23] 均运用 MATLAB 的 fmin 函数进行仿真求解。为将该求解方式进一步拓展至更通用的平台, 本文采用当前流行的 MPC[37-41] 框架来构造并求解优化问题。规划所生成的轨迹作为控制系统外环的参考输入, 以引导无人机飞行。

针对 MPC 优化问题的求解, 存在众多方便使用的库。本文选用了 C++版本的 Casadi(https://gitee.com/lanqx/casadi.git), 该库具备强大的功能集及简洁的语言设计, 易于学习和使用。本文构造的 MPC 为一个收紧约束的优化问题, 亦称基于管道的 MPC(Tube-based-MPC) [42], 通过减小可行域, 提高了规划结果的安全性, 使其更易跟踪。本文构造的 MPC 优化问题如下:

$$\begin{cases}\min\limits_{\boldsymbol{a}_k}\sum\limits_k c\left(\hat{\boldsymbol{b}}_k,\boldsymbol{a}_k\right)+d_k\\ \text{s.t.}\begin{cases}\boldsymbol{x}_0=\boldsymbol{x}_{\text{S}},\boldsymbol{x}_{H+1}=\boldsymbol{x}_{\text{F}}\\ \hat{\boldsymbol{P}}_{k+1}=\begin{cases}\left[\left(\boldsymbol{F}_k\hat{\boldsymbol{P}}_k\boldsymbol{F}_k^{\mathrm{T}}+\boldsymbol{Q}_k\right)^{-1}+\boldsymbol{S}_{k+1}\right]^{-1}\\ \left(\boldsymbol{F}_k\hat{\boldsymbol{P}}_k\boldsymbol{F}_k^{\mathrm{T}}+\boldsymbol{Q}_k\right)^{-1}\end{cases}\\ h(\boldsymbol{x}_k,\boldsymbol{a}_k)\leqslant 0\qquad k=0,1,\cdots,H\end{cases}\end{cases} \tag{21}$$

式中: $\boldsymbol{x}_0$为无人机的初始状态; $\boldsymbol{x}_{\text{S}}$为对应的初始状态取值; $\boldsymbol{x}_{H+1}$为终点状态; $\boldsymbol{x}_{\text{F}}$为终点状态取值; $h(\boldsymbol{x}_k,\boldsymbol{a}_k)$为运动学模型的状态和输入约束; d_k为关于避障距离的度量, 其计算式为

$$d_k=\{|\boldsymbol{X}_k-\boldsymbol{m}_k||m\in O_{\text{bs}k}^M\} \tag{22}$$

式中: $O_{\text{bs}k}^M$表示时间跨度为k时无人机感知范围内的M个障碍物; d_k为第k个时间跨度下无人机与 M 个障碍

物的距离集合，其计算方式采用全维度距离进行表示。

3.3 全维度距离表示

通常在式(21)中有关避障约束，大多会将障碍物等效为规则的几何图形，并通过解析式来实现避障约束。然而，在实际情况中，大多数障碍物都是不规则的多边形或多面体。若将这些不规则形状等效成规则的球形、圆柱、椭圆等形状，将会缩小可行域，致使求解失败或求解效果不理想。尽管文献[31-32]采用对偶变换的方式很好地从理论上解决了避障约束，但由于引入了新的变量和约束条件，导致在障碍物增多时求解时间迅速增加，无法在障碍物密集的环境中进行在线规划。而文献[30]通过将避障约束放到目标函数中，显著提升了求解性能。因此，本文在文献[30]二维平面的避障距离度量方法基础上进行了拓展，以满足三维空间中无人机航路规划的需求。首先，避障约束表示为

$$\begin{cases} s_{\mathrm{d}}(D,B):\ =d_{\mathrm{ist}}(D,B)-p_{\mathrm{en}}(D,B) \\ d_{\mathrm{ist}}(D,B)=\min_T\{\|T\|_2:(D+T)\cap O\neq\varnothing\} \\ p_{\mathrm{en}}(D,B)=\min_T\{\|T\|_2:(D+T)\cap O=\varnothing\} \end{cases} \tag{23}$$

式中：D为无人机当前的位置和姿态；B为凸多边形障碍物的位置和姿态；$d_{\mathrm{ist}}(D,B)$为当无人机和障碍物未重叠时，两者之间的最短距离；$p_{\mathrm{en}}(D,B)$为当无人机和障碍物重叠时，两者分开的最短距离。

这些距离的度量均由无人机和障碍物的顶点与顶点及顶点和面之间的距离所组成集合中的最小值构成，因此，导数不连续，无法充当约束条件或目标函数通过MPC框架进行优化。虽然整数规划能够解决此类问题，但其无法提供平滑解以供无人机跟踪。文献[30]针对式(23)所描述的避障问题从理论层面进行了推导，并提供了相应的解决方案。设有2个集合$X_{\mathrm{u}},Y_{\mathrm{u}}\in\mathbf{R}$，其闵可夫斯基差为

$$X_{\mathrm{u}}-Y_{\mathrm{u}}=\{(x-y)\in\mathbf{R}:x\in X_{\mathrm{u}},y\in Y_{\mathrm{u}}\} \tag{24}$$

式中：集合$X_{\mathrm{u}}-Y_{\mathrm{u}}$为一个以0点为原点的多边形或多面体，集合到0点的距离包含A到B的最短距离，因此，无人机到障碍物的距离可以表示为

$$s_{\mathrm{d}}(D,B)=s_{\mathrm{d}}(0,B-D) \tag{25}$$

考虑方向性，则有

$$\begin{aligned} &\max_{\mathcal{P}_{f_{\mathrm{e}}},\mathcal{P}_{f_{\mathrm{o}}}}\left\{s_{\mathrm{d}}\left(0,B-\mathcal{P}_{f_{\mathrm{e}}}\right),s_{\mathrm{d}}\left(0,\mathcal{P}_{f_{\mathrm{o}}}-D\right)\right\}\leqslant s_{\mathrm{d}}(D,B) \\ &\quad\forall f_{\mathrm{e}}=\{1,2,\cdots,K_{\mathrm{e}}\},\forall f_{\mathrm{o}}=\{1,2,\cdots,K_{\mathrm{o}}\} \end{aligned} \tag{26}$$

式中：$\mathcal{P}_{f_{\mathrm{e}}}$和$\mathcal{P}_{f_{\mathrm{o}}}$分别为无人机的$f_{\mathrm{e}}$面和$f_{\mathrm{o}}$面；$K_{\mathrm{e}}$为无人机的面的个数；$K_{\mathrm{o}}$为障碍物的面数。且$s_{\mathrm{d}}\left(0,B-\mathcal{P}_{f_{\mathrm{e}}}\right)$和$s_{\mathrm{d}}\left(0,\mathcal{P}_{f_{\mathrm{o}}}-D\right)$的计算式分别为

$$\begin{cases} s_{\mathrm{d}}\left(0,B-\mathcal{P}_{f_{\mathrm{e}}}\right)=\dfrac{1}{\|\boldsymbol{H}^{f_{\mathrm{e}}}\|_2}\left(-\boldsymbol{h}^{f_{\mathrm{e}}}+\min_{\boldsymbol{u}_{\mathrm{T}}}\left(\boldsymbol{H}^{f_{\mathrm{e}}}\right)^{\mathrm{T}}\boldsymbol{u}_{\mathrm{T}}\right) \\ s_{\mathrm{d}}\left(0,\mathcal{P}_{f_{\mathrm{o}}}-D\right)=\dfrac{1}{\|\boldsymbol{G}^{f_{\mathrm{o}}}\|_2}\left(-\boldsymbol{g}^{f_{\mathrm{o}}}+\min_{\boldsymbol{y}_{\mathrm{r}}}\left(\boldsymbol{G}^{f_{\mathrm{o}}}\right)^{\mathrm{T}}\boldsymbol{y}_{\mathrm{r}}\right) \end{cases} \tag{27}$$

式中：$\boldsymbol{h}^{f_{\mathrm{e}}}$和$\boldsymbol{g}^{f_{\mathrm{o}}}$分别为无人机和障碍物各个面的法向量；$\boldsymbol{H}^{f_{\mathrm{e}}}$和$\boldsymbol{G}^{f_{\mathrm{o}}}$分别为无人机和障碍物的面$f_{\mathrm{e}}$和面$f_{\mathrm{o}}$的法向量且方向向外；$\boldsymbol{u}_{\mathrm{T}}$和$\boldsymbol{y}_{\mathrm{r}}$分别为无人机和障碍物的各个顶点。式(27)的物理意义是无人机和障碍物各自顶点到对方面f_{e}或面f_{o}的最短距离。

3.4 最短距离约束

避障要求在每个时间跨度k的无人机和周围一定范围内的U个障碍物之间的最短距离均大于d_{m}，则有

$$s_{\mathrm{d}}(D(k),B_u(k))\geqslant d_{\mathrm{m}}\qquad u=1,2,\cdots,U \tag{28}$$

式中：$D(k)$为第k个时间跨度对应的无人机的位置，$B_u(k)$为第k时间跨度时，无人机附近U个障碍物中的第u个障碍物。联立式(25)和式(27)，则有

$$\begin{cases} s_{\mathrm{d}}(D(k),B_u(k))\geqslant \\ \quad\max\left\{\max_{f_{\mathrm{e}}}\min_{v_{\mathrm{ero}}}\boldsymbol{H}\left(\boldsymbol{R},\mathcal{P}_{f_{\mathrm{e}}}\right)^{\mathrm{T}}\left(v^u_{v_{\mathrm{ero}}}(k)-v_{f_{\mathrm{e}}}(k)\right)\right. \\ \left.\max_{f_{\mathrm{o}}}\min_{v_{\mathrm{ere}}}\boldsymbol{H}\left(\boldsymbol{R}_n(k),\mathcal{P}^u_{f_o}\right)^{\mathrm{T}}\left(v_{v_{\mathrm{ere}}}(k)-v^u_{f_{\mathrm{o}}}(k)\right)\right\}\geqslant d_{\mathrm{m}} \\ v_{\mathrm{ere}}\in\{1,2,\cdots,N_{\mathrm{e}}\},v_{\mathrm{ero}}\in\left\{1,2,\cdots,N^u_{\mathrm{o}}\right\} \\ f_{\mathrm{e}}\in\{1,2,\cdots,K_{\mathrm{e}}\},f_{\mathrm{o}}\in\left\{1,2,\cdots,K^u_{\mathrm{o}}\right\} \end{cases} \tag{29}$$

式中：$\boldsymbol{H}(\boldsymbol{R},\mathcal{P})$为无人机或障碍物面$\mathcal{P}$上由内指向外的单位法向量；$\boldsymbol{R}$为无人机或障碍物的旋转矩阵；$v_{f_{\mathrm{e}}}(k)$为无人机面$f_{\mathrm{e}}$上的任意一点；$v_{v_{\mathrm{ere}}}(k)$为无人机的一个顶点；$v^u_{f_{\mathrm{o}}}(k)$为第$u$个障碍物的面$f_{\mathrm{o}}$上的任意一点；$v^u_{v_{\mathrm{ero}}}(k)$为第$u$个障碍物的一个顶点。

以$s_{\mathrm{d}}\left(0,B-\mathcal{P}_{f_{\mathrm{e}}}\right)$为例，其计算示意图如图1所示。图1中，以障碍物顶点到f_{e2}面的距离为例，可以计算UAV的其他面到障碍物各个顶点的距离。反之$s_{\mathrm{d}}\left(0,\mathcal{P}_{f_{\mathrm{o}}}-D\right)$可以通过计算UAV各个顶点到障碍物各个面的距离，采用式(29)进行计算。与文献[30]一样，对式(29)采用对数求和指数(log_sum_exp, LSE)进行平滑处理，LSE能够增加数值稳定性，防止数值溢出，并增加计算效率。以$\max\{\gamma\}$为例，LSE处理后的取值会限定在如下范围中：

$$\max\{\gamma\}+\frac{\ln\Omega}{\alpha'}\geqslant L_{\mathrm{SE}(\alpha'>0)}(\gamma)\geqslant\max\{\gamma\} \tag{30}$$

式中：$\dfrac{\ln\Omega}{\alpha'}$为一个很小的项，$\alpha'>0$表示对max进行平滑，$\alpha'<0$表示对min进行平滑。

LSE处理后的形式为

$$\begin{cases} U(D(k),B_u(k))=L_{\mathrm{SE}(\alpha'>0)}(\boldsymbol{d})-\dfrac{\ln\left(K_{\mathrm{e}}+K^u_{\mathrm{o}}\right)}{\alpha'} \\ \boldsymbol{d}=\left[\boldsymbol{d}_U^{\mathrm{T}},\boldsymbol{d}_E^{\mathrm{T}}\right]^{\mathrm{T}}\in\mathbf{R}^{K_{\mathrm{e}}+K^u_{\mathrm{o}}} \\ \boldsymbol{d}_U=\left[d_U^1,\ldots,d_U^{f_{\mathrm{e}}},\ldots d_U^{K_{\mathrm{e}}}\right]^{\mathrm{T}}\in\mathbf{R}^{K_{\mathrm{e}}} \\ \boldsymbol{d}_E=\left[d_E^1,\ldots,d_E^{f_{\mathrm{o}}},\ldots d_E^{K^u_{\mathrm{o}}}\right]^{\mathrm{T}}\in\mathbf{R}^{K^u_{\mathrm{o}}} \end{cases} \tag{31}$$

式中：$d_U^{f_{\mathrm{e}}}=s_{\mathrm{d}}\left(0,B_u(k)-\mathcal{P}_{f_{\mathrm{e}}}(k)\right)$；$d_E^{f_{\mathrm{o}}}=s_{\mathrm{d}}\left(0,\mathcal{P}^u_{f_{\mathrm{o}}}(k)-D(k)\right)$，同样对$d_U^{f_{\mathrm{e}}}$和$d_E^{f_{\mathrm{o}}}$通过LSE平滑结果为

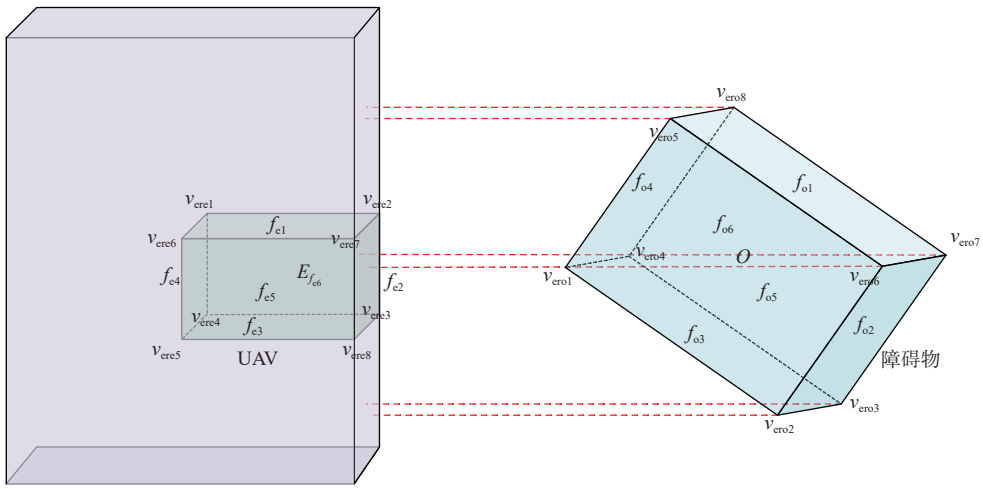


图 1　$s_{\mathrm{d}}\left(0, B-\mathcal{P}_{f_{\mathrm{e}}}\right)$ 计算示意图

Fig. 1　Schematic diagram of calculation of $s_{\mathrm{d}}\left(0, B-\mathcal{P}_{f_{\mathrm{e}}}\right)$

$$\begin{cases} d_U^{f_{\mathrm{e}}} = L_{\mathrm{SE}(\alpha'<0)}\left(\left[d_{U_1}^{f_{\mathrm{e}}}, \cdots, d_{U_{\mathrm{vero}}}^{f_{\mathrm{e}}}, \cdots, d_{U_{N_{\mathrm{o}}^u}}^{f_{\mathrm{e}}}\right]^{\mathrm{T}}\right) + \tilde{d}_U^{f_{\mathrm{e}}} \\ d_E^{f_{\mathrm{o}}} = L_{\mathrm{SE}(\alpha'<0)}\left(\left[d_{E_1}^{f_{\mathrm{o}}}, \cdots, d_{E_{\mathrm{vere}}}^{f_{\mathrm{o}}}, \cdots, d_{E_{N_{\mathrm{e}}}}^{f_{\mathrm{o}}}\right]^{\mathrm{T}}\right) + \tilde{d}_E^{f_{\mathrm{o}}} \end{cases} \tag{32}$$

式中: $d_{U_{\mathrm{vero}}}^{f_{\mathrm{e}}} = \boldsymbol{H}\left(\boldsymbol{R}(k), \mathcal{P}_{f_{\mathrm{e}}}\right)^{\mathrm{T}} \boldsymbol{R}_u(k) \boldsymbol{l}_{v_{\mathrm{vero}}}^u$, 其物理含义为无人机 f_{e} 面的单位法向量上障碍物顶点 $\boldsymbol{R}_u(k)\boldsymbol{l}_{v_{\mathrm{vero}}}^u$ 的投影; $\tilde{d}_U^{f_{\mathrm{e}}} = \left|\boldsymbol{H}\left(\boldsymbol{R}, \mathcal{P}_{f_{\mathrm{e}}}\right)^{\mathrm{T}}\right| \left(B_{\mathrm{o}}^u(k) - v_{f_{\mathrm{e}}}(k)\right)$, 其物理含义为障碍物中心点 $B_{\mathrm{o}}^u(k)$ 到无人机 $\boldsymbol{H}\left(\boldsymbol{R}, \mathcal{P}_{f_{\mathrm{e}}}\right)^{\mathrm{T}}$ 所在面上的任意一点 $v_{f_{\mathrm{e}}}(k)$ 的距离在该方向量上的投影; 同理, $d_{E_{\mathrm{vere}}}^{f_{\mathrm{o}}} = \left|\boldsymbol{H}\left(\boldsymbol{R}_u(K), \mathcal{P}_{f_{\mathrm{o}}}^u\right)^{\mathrm{T}}\right| \boldsymbol{R}\boldsymbol{l}_{v_{\mathrm{vere}}}$, $\tilde{d}_E^{f_{\mathrm{o}}} = \left|\boldsymbol{H}\left(\boldsymbol{R}_u(K), \mathcal{P}_{f_{\mathrm{o}}}^u\right)^{\mathrm{T}}\right| \cdot \left(D_{\mathrm{e}}(k) - v_{f_{\mathrm{o}}}^u(k)\right)$。

设 $d_k = d_{\mathrm{m}} - U\left(D(k), B_u(k)\right)$, 本文设计了一个平滑函数代替无人机和障碍物间距离的惩罚, 具体为

$$c_k = \begin{cases} 0.5x^2 & x < d_k, x > 0 \\ x + 0.5d_k^2 - d_k & x \geqslant d_k \\ 0 & x \leqslant 0 \end{cases} \tag{33}$$

将式(33)代入式(22), 最终目标函数为

$$\begin{cases} \min\limits_{\boldsymbol{a}_k} \sum\limits_k c\left(\hat{\boldsymbol{b}}_k, \boldsymbol{a}_k\right) + w_1 c_k + w_2 \sum\limits_i \tilde{s}_i \\ \text{s.t.} \begin{cases} \boldsymbol{x}_0 = \boldsymbol{x}_{\mathrm{S}}, \boldsymbol{x}_H = \boldsymbol{x}_{\mathrm{F}} \\ \hat{\boldsymbol{P}}_{k+1} = \begin{cases} \left[\left(\boldsymbol{F}_k \hat{\boldsymbol{P}}_k \boldsymbol{F}_k^{\mathrm{T}} + \boldsymbol{Q}_k\right)^{-1} + \boldsymbol{S}_{k+1}\right]^{-1} \\ \left(\boldsymbol{F}_k \hat{\boldsymbol{P}}_k \boldsymbol{F}_k^{\mathrm{T}} + \boldsymbol{Q}_k\right)^{-1} \end{cases} \\ h_i\left(\boldsymbol{x}_k, \boldsymbol{a}_k\right) \leqslant s_i \qquad k = 0, 1, \cdots, H, i = 0, 1, \cdots, H \end{cases} \end{cases} \tag{34}$$

式中: w_1 和 w_2 分别为避障条件 c_k 和松弛变量 $\tilde{s}_i$ 的系数。

将未知环境中无人机与所感知的静态和动态障碍物间的距离均以符号距离的方式进行表示, 此时满足了任意一个时间跨度下的优化需要, 对于动态障碍物还需考虑其时间跨度总数为 H 的可能的运动位置和姿态。如图 2 所示。

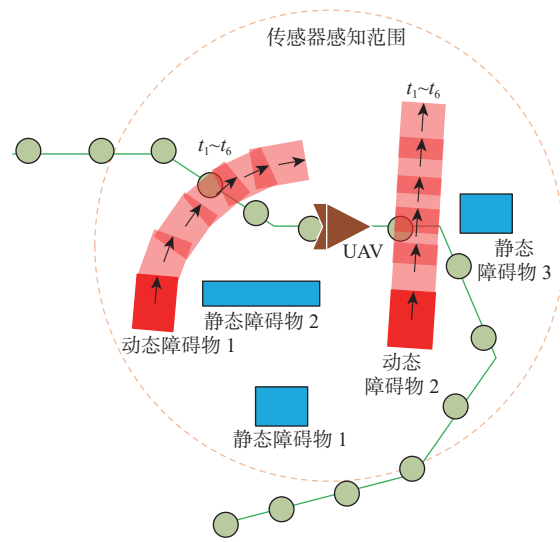


图 2　未知场景避障示意图

Fig. 2　Schematic diagram of obstacle avoidance in unknown scenarios

图 2 中无人机通过机载传感器感知到周围一定范围内有 2 个动态障碍物和 3 个静态障碍物, 其中, 动态障碍物考虑了 $t_1 \sim t_6$ 时间跨度的位置和姿态。绿色的折线为任务指定航线, 圆点为航点, 目的是尽量靠近任务航线朝着航点序列飞行。在 MPC 优化过程中, 针对不同的时间跨度 k 考虑的动态障碍物的位置和姿态是不同的。本文旨在对航路规划框架进行验证, 并未对障碍物的感知和识别及动态障碍物轨迹预测进行研究, 该内容属于其他领域。本文在处理动态障碍物时提前设定了轨迹, 模拟轨迹预测过程, 从轨迹队列中, 按照时间跨度获取其位置和姿态。针对障碍物的形状则都设定为高低不等的长方体(非凸多面体也需分割成多个凸多面进行计算)。

4 基于MPC的强化学习

MPC控制器的闭环性能取决于其组成部分，尤其是预测模型[29]。动力学中的不匹配会削弱MPC可靠预测系统行为的能力，从而影响控制器的性能及其避免违反约束的能力。本文采用的三维空间运动模型突显了该问题。本节将展示如何通过利用在线闭环数据，通过强化学习自动学习MPC中的参数来解决这些缺点。

4.1 MPC参数化

MPC作为策略输出逼近器，为融合到强化学习框架中，需要对其进行参数化处理，基于式(34)，参数化后的优化问题为

$$\begin{cases}\min\limits_{\boldsymbol{a}_k}\left(\theta_{\mathrm{p}}^2\sum\limits_k c\left(\hat{\boldsymbol{b}}_k,\boldsymbol{a}_k\right)+\theta_{\mathrm{u}}^2\|\boldsymbol{u}_{k+1}-\boldsymbol{u}_k\|_2+\theta_{\mathrm{o}}^2 w_1 c_k\right)\gamma^k+\\ \qquad w_2\sum\limits_i s_i\\ \text{s.t.}\begin{cases}\boldsymbol{x}_0=\boldsymbol{x}_{\mathrm{S}},\boldsymbol{x}_H=\boldsymbol{x}_{\mathrm{F}}\\ \hat{\boldsymbol{P}}_{k+1}=\begin{cases}\left[\left(\boldsymbol{F}_k\hat{\boldsymbol{P}}_k\boldsymbol{F}_k^{\mathrm{T}}+\boldsymbol{Q}_k\right)^{-1}+\boldsymbol{S}_{k+1}\right]^{-1}\\ \left(\boldsymbol{F}_k\hat{\boldsymbol{P}}_k\boldsymbol{F}_k^{\mathrm{T}}+\boldsymbol{Q}_k\right)^{-1}\end{cases}\\ h_i(\boldsymbol{x}_k,\boldsymbol{a}_k)\leqslant s_i\qquad k=0,1,\cdots,H,i=0,1,\cdots,H\end{cases}\end{cases}\tag{35}$$

式中：$c\left(\hat{\boldsymbol{b}}_k,\boldsymbol{a}_k\right)$包含了时间跨度$k$时的无人机位置和目标点位置，其计算可参考式(19)；θ_{p}为与目标相关的参数；θ_{u}为与输入变化相关的参数；θ_{o}为与避障相关的参数；w_1和w_2均为固定常数，用来稳定优化问题求解过程。使用松弛变量一方面表示这个约束的小违反(尽管是不希望的)是可以容忍的；另一方面确保在学习过程中既能放松约束，又能对违反进行惩罚，以便使智能体学会区分安全和不安全的策略[29]。此外，在优化问题中，有时存在一些硬约束条件(必须严格满足的条件)，但由于各种实际原因，完全满足这些条件可能会很困难或导致问题无解。通过引入松弛变量，可以将硬约束转化为等式约束加上松弛变量的非负约束，从而提升优化问题的求解成功率。

MPC每求解一次，无人机向前前进一步并记录一次阶段成本，然后将该成本值连同状态和动作及下一个状态存储到经验池中，以便强化学习框架通过构造时序差分(temporal difference, TD)误差损失函数进行参数调整。本文的阶段成本代价s_{gc}考虑了状态越界代价s_{c}，输入成本变换率最小代价a_{rc}及距离障碍物的距离d_{oc}三部分。具体如下：

$$s_{\mathrm{gc}}(k)=w_{\mathrm{s1}}s_{\mathrm{c}}+w_{\mathrm{s2}}a_{\mathrm{rc}}+w_{\mathrm{s2}}d_{\mathrm{oc}}\tag{36}$$

$$s_{\mathrm{c}}=\max(0,x_k(n)-x_{\mathrm{ub}},x_{\mathrm{lb}}-x_k(n))\tag{37}$$

$$a_{\mathrm{rc}}=a_k-a_{\mathrm{last}}\tag{38}$$

$$d_{\mathrm{oc}}=d(x_k(k),O)\tag{39}$$

总体而言，可调参数矢量$\boldsymbol{\theta}$为

$$\boldsymbol{\theta}=(\theta_{\mathrm{p}},\theta_{\mathrm{u}},\theta_{\mathrm{o}})\tag{40}$$

$\boldsymbol{\theta}$在强化学习过程中根据阶段成本会逐步调整优化问题权重，提升求解质量。当参数足够多的情况下，参数化的MPC框架有能力获得最优策略[26]。

4.2 基于二阶最小二乘时序差的Q学习

基于LSTD-Q的强化学习方法能高效利用数据，且往往比更基础的时序差分学习方法收敛得更快[29]。针对$\boldsymbol{\theta}$构建贝尔曼剩余误差的最小二乘如下：

$$\min_{\boldsymbol{\theta}}E\left(\|Q_*(\boldsymbol{s},\boldsymbol{a})-Q_{\boldsymbol{\theta}}(\boldsymbol{s},\boldsymbol{a})\|_2^2\right)\tag{41}$$

式中：$Q_*(\boldsymbol{s},\boldsymbol{a})\approx L(\boldsymbol{s},\boldsymbol{a})+\gamma Q_{\boldsymbol{\theta}}(\boldsymbol{s}_+,\pi(\boldsymbol{s}_+))$，为当前状态$\boldsymbol{s}$时，执行动作$\boldsymbol{a}$后的动作价值，$\boldsymbol{s}_+$为转移后的新状态，$L(\boldsymbol{s},\boldsymbol{a})$为执行动作$\boldsymbol{a}$后的立即回报；$Q_{\boldsymbol{\theta}}(\boldsymbol{s},\boldsymbol{a})$为执行动作$\boldsymbol{a}$前，现有策略模型对动作价值的评估。

二阶最小二乘时序差的Q学习则是通过对式(41)进行偏导数计算，关于$\boldsymbol{\theta}$的更新如下：

$$\begin{cases}\boldsymbol{\theta}\leftarrow\boldsymbol{\theta}-\alpha_{\mathrm{L}}\boldsymbol{A}^{-1}\boldsymbol{b}\\ \boldsymbol{A}=E\left(\delta(\nabla\boldsymbol{\theta})^2Q_{\boldsymbol{\theta}}+\nabla\boldsymbol{\theta}Q_{\boldsymbol{\theta}}\nabla\boldsymbol{\theta}^{\mathrm{T}}\delta\right)\\ \boldsymbol{b}=E\left(\delta\dfrac{\partial Q_{\boldsymbol{\theta}}}{\partial\boldsymbol{\theta}}\right)\end{cases}\tag{42}$$

式中：$\delta=L(s_i,\boldsymbol{a}_i)+\gamma V_{\boldsymbol{\theta}}(s_{i+1})-Q_{\boldsymbol{\theta}}(s_i,\boldsymbol{a}_i)$为TD误差；$V_{\boldsymbol{\theta}}(s_{i+1})$为状态价值；$\alpha_{\mathrm{L}}$为学习率。

在式(42)中关于$Q_{\boldsymbol{\theta}}$的导数和海森矩阵的计算需要通过优化问题最优值的灵敏度来获得。可以构造如下函数：

$$\mathcal{L}_{\boldsymbol{\theta}}=\Phi_{\boldsymbol{\theta}}+\boldsymbol{\lambda}^{\mathrm{T}}G_{\boldsymbol{\theta}}+\boldsymbol{\mu}^{\mathrm{T}}H_{\boldsymbol{\theta}}\tag{43}$$

式中：$\Phi_{\boldsymbol{\theta}}$为优化问题的目标函数；$\boldsymbol{\lambda}$和$\boldsymbol{\mu}$分别为与等式约束$G_{\boldsymbol{\theta}}$和不等式约束$H_{\boldsymbol{\theta}}$相关的拉格朗日乘子。

设$z=\{p,\boldsymbol{\lambda},\boldsymbol{\mu}\}$，$p=\{X,\overline{U}\}$，其中$X$和$\overline{U}$分别为与优化问题中的状态和输入相关的变量，可得梯度$\nabla\boldsymbol{\theta}Q_{\boldsymbol{\theta}}$和海森矩阵$(\nabla\boldsymbol{\theta})^2Q_{\boldsymbol{\theta}}$为

$$\begin{cases}\nabla\boldsymbol{\theta}Q_{\boldsymbol{\theta}}=\dfrac{\partial Q_{\boldsymbol{\theta}}}{\partial\boldsymbol{\theta}}=\dfrac{\partial\mathcal{L}_{\boldsymbol{\theta}}(\boldsymbol{s},\boldsymbol{a},z^*)}{\partial\boldsymbol{\theta}}\\ \nabla\boldsymbol{\theta}^2Q_{\boldsymbol{\theta}}=\boldsymbol{H}_{\mathrm{h}}(Q_{\boldsymbol{\theta}})=\dfrac{D}{D\boldsymbol{\theta}}\left(\dfrac{\partial\mathcal{L}_{\boldsymbol{\theta}}(\boldsymbol{s},\boldsymbol{a},z^*)}{\partial\boldsymbol{\theta}}\right)\approx\dfrac{\partial^2\mathcal{L}_{\boldsymbol{\theta}}(\boldsymbol{s},\boldsymbol{a},z^*)}{\partial\boldsymbol{\theta}^2}\end{cases}\tag{44}$$

计算得到的海森矩阵$\boldsymbol{H}_{\mathrm{h}}$总是通过加上一个单位矩阵的倍数来被修改为正定的[29]，因此，对参数需要施加上下限，原始的更新规则被表述为以下优化问题：

$$\begin{cases}\Delta\boldsymbol{\theta}^{*}=\arg\min\limits_{\Delta\boldsymbol{\theta}}\dfrac{1}{2}\Delta\boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{H}_{\mathrm{h}}\Delta\boldsymbol{\theta}+\alpha\boldsymbol{F}^{\mathrm{T}}\Delta\boldsymbol{\theta}\\ \text{s.t.}\begin{cases}\boldsymbol{\theta}_{\mathrm{lb}}\leqslant\boldsymbol{\theta}+\Delta\boldsymbol{\theta}\leqslant\boldsymbol{\theta}_{\mathrm{ub}}\\ \Delta\boldsymbol{\theta}_{\mathrm{lb}}\leqslant\Delta\boldsymbol{\theta}\leqslant\Delta\boldsymbol{\theta}_{\mathrm{ub}}\end{cases}\end{cases}\tag{45}$$

式中: $\boldsymbol{F}=\nabla\boldsymbol{\theta}Q_{\boldsymbol{\theta}}$; $\boldsymbol{H}_{\mathrm{h}}=\nabla\boldsymbol{\theta}^{2}Q_{\boldsymbol{\theta}}$; 梯度的更新量为 $\Delta\boldsymbol{\theta}^{*}$, 因此, 式(42)中关于参数 $\boldsymbol{\theta}$ 的更新如下:

$$\boldsymbol{\theta}\leftarrow\boldsymbol{\theta}+\Delta\boldsymbol{\theta}^{*}\tag{46}$$

综合以上, 本文 QL-NBO 算法流程如图 3 所示。该算法核心在于避障优化模型的构建及基于最小二阶时序差的 Q-Learning 进行实时的参数调整。其中, 针对前者主要包含 3 部分, 分别为 POMDP 框架下基于 NBO 的目标函数近似、多面体避障距离度量和基于最小二乘平滑 LSE 的避障距离可微性处理及运动学模型状态和输入约束。针对后者主要实现对优化模型中需要学习的参数进行动态调整。详细流程如图 3 所示。可以看出, 首先, 构造避障优化问题模型; 其次, 对模型进行参数化处理, 并对相关参数进行初始化; 然后, 采用 Casadi 求解库计算模型输出策略(平滑的参考轨迹); 接着, 无人机按照该策略引导向前飞行, 此时判断无人机是否到达目标位置, 如到达则结束飞行, 如未到达则记录当前时间跨度前后的环境状态及奖励等信息($<\boldsymbol{s},\boldsymbol{a},\boldsymbol{s}',\boldsymbol{r},\mathrm{done}>$, 其中, done 为结束标志, $\boldsymbol{s}$ 和 $\boldsymbol{s}'$ 为相邻的 2 个环境状态, $\boldsymbol{a}$ 为动作策略, $\boldsymbol{r}$ 为奖励值)并存入经验池中; 最后, 基于 Q-Learning 强化学习框架进行相关参数调整。

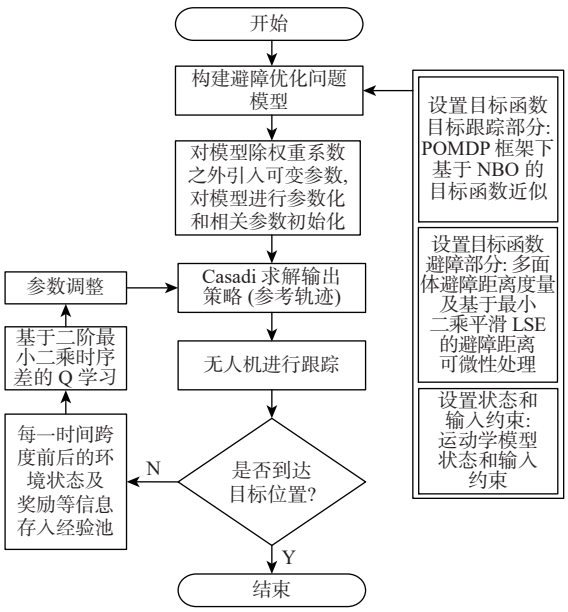


图 3　本文 QL-NBO 算法流程

Fig. 3　Flowchart of the QL-NBO algorithm

5　规划框架

本节设计了一个规划框架对本文算法进行验证, 如图 4 所示。框架支持软件在环和硬件在环仿真, 基于文献 [43] 和文献 [44] 搭建, 整个框架分为 L1、L2 和 L3 这 3 层。L1 包含地图模块、任务模块、规划模块和感知模块, 内部通过 MAVROS 进行通信。地图模块负责在场景中生成静态和动态障碍物, 以及订阅飞控中的任务, 并通过 RVIZ 进行可视化处理; 任务模块负责手动切换 OffBoard 和发布位置三元组(针对执行固定航路的任务场景, 可用来判断当前位置和航线任务间关系), 以便进行航路切换(参考了 px4 源代码 git clone https://github.com/PX4/

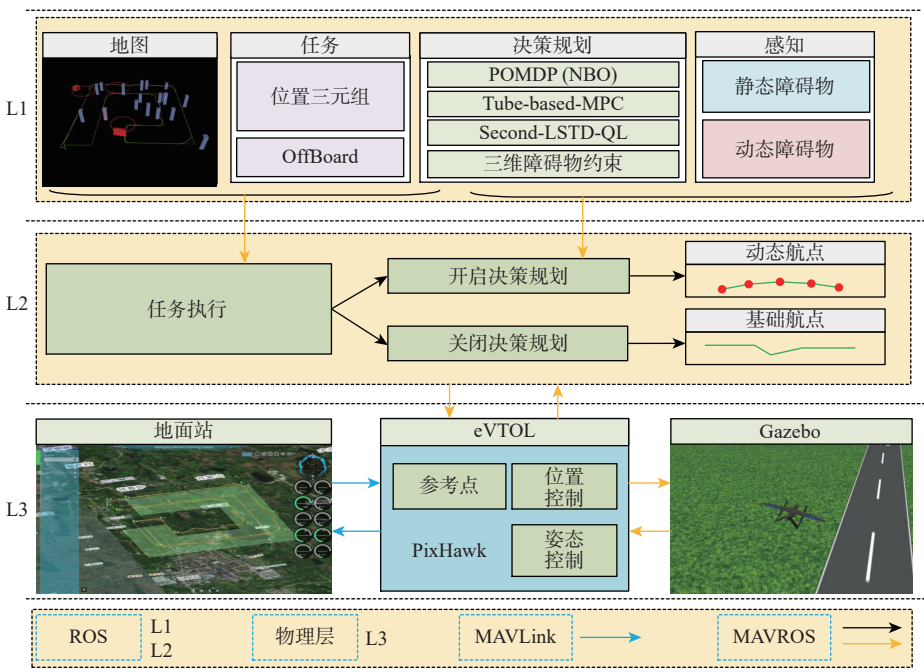


图 4　航路规划框架

Fig. 4　Path planning framework

PX4-Autopilot.git)到飞控端;规划模块采用NBO对POMDP近似处理,采用Tube-based-MPC(收紧约束的优化问题)框架进行优化问题构造,基于Q-Learning强化学习框架通过构造二次最小二乘问题实现对参数化MPC的更新,同时针对障碍物约束,在三维空间中采用考虑多面体形状的全维度避障度量,并进行LSE平滑处理,对未来5个时间跨度的轨迹点进行预测,每个时间跨度为2.5 s;感知模块负责将检测到的静态和动态障碍物及未来5个时间跨度的轨迹通过MAVROS发送给规划模块。L2负责整个规划框架主体数据装配和整个软件的任务执行,活跃在整个系统的生命周期内。无人机在飞行过程中会一直感知地图中的障碍物信息,该模块判断当前环境中不存在障碍物时按照任务航点执行,一旦发现有障碍物则会启动规划模块,通过任务模块借助三元组话题将避障轨迹发送给无人机跟踪。L3为物理层,包含地面站、无人机和Gazebo环境。地面站基于QGroundControl开发,增加了3D可视化模块及其他容易操作的功能;无人机模块负责接收任务执行模块发送的航点任务,通过位置和姿态控制完成轨迹跟踪;Gazebo负责飞行环境可视化,将实时的位置和速度等信息提供给飞控单元。

6 实验分析

6.1 仿真平台及参数设置

本文设计的规划框架运行在Ubuntu18.04操作系统上,飞控单元采用PixHawk平台,无人机是自定义的eVTOL模型,并将其加载到Gazebo环境中。计算单元采用i7处理器,硬盘为500 GB纯固态。在场景中,以长方体替代所有凸多面体作为障碍物,原因在于无论是凹多面体还是大型凸多面体,皆可分割为多个凸多面体。对于不规则形状障碍物的测试,文献[30]已完成。本文创建的场景中障碍物数量众多,为便于测试,统一设置为高度随机的长方体,主要聚焦于其有效性与避障性能分析。实验参数设置如表1所示。

表1 参数设置

Table 1 Parameter settings

参数	数值
$\boldsymbol{P}$	$\mathrm{diag}(1.0)_{6\times6}$
$\boldsymbol{Q}_k$	$\mathrm{diag}(10^{-4})_{6\times6}$
$\boldsymbol{H}_k$	$[1.0_{2\times2}\quad 0.0_{2\times4}]$
$\boldsymbol{R}$	$\mathrm{diag}(1.0)_{3\times3}$
$\boldsymbol{F}_k$	$\mathrm{diag}(1.0)_{6\times6}$
d_m/m	100.0
感知范围/m	200.0
d_t/s	1.0
时间跨度	5
$v/(\mathrm{m\cdot s^{-1}})$	17~25
θ_p	1.0
θ_u	1.0
θ_o	1.0
α	5×10^6
w_1	5×10^3
w_2	1×10^6
$w_{\mathrm{s}1}$	0.1
$w_{\mathrm{s}2}$	0.1
$w_{\mathrm{s}3}$	0.1
p	10
q	10
μ	10
$\widetilde{b}$	10

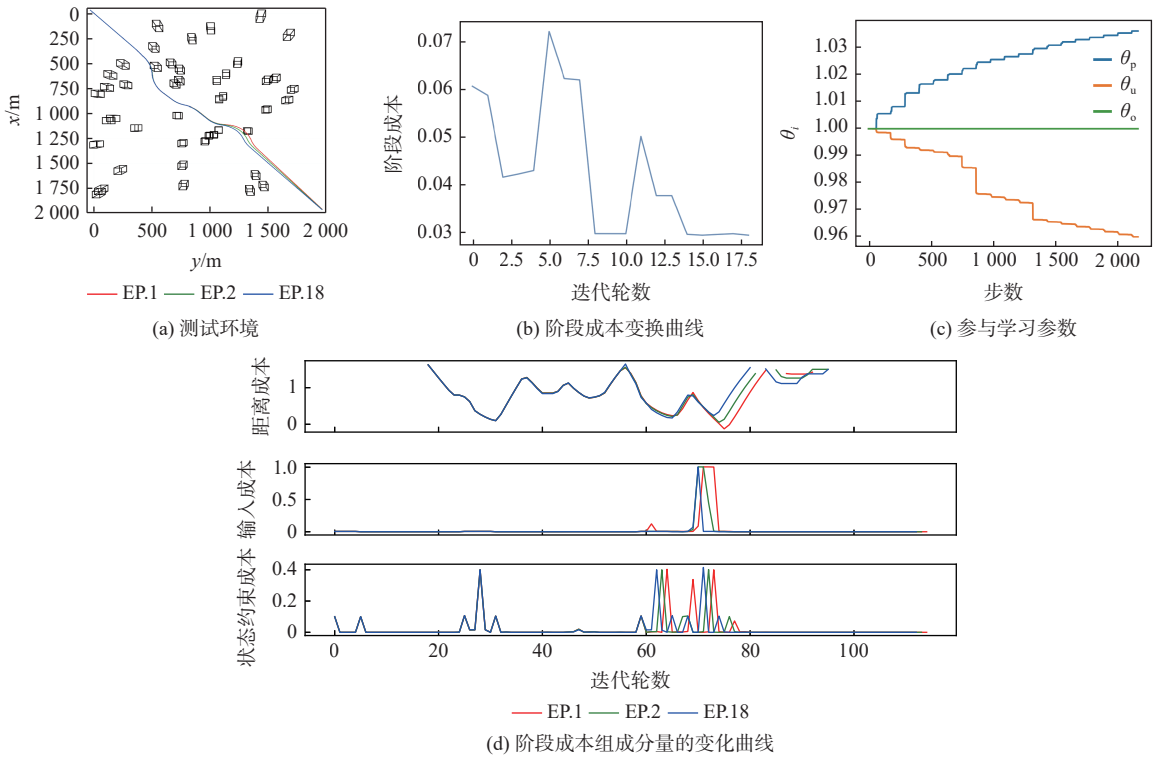
6.2 Q-Learning对NBO的有效性

为验证Q-Learning对NBO的有效性,本文设计了如图5(a)所示的场景,障碍物为随机生成的长方体,无人机飞行起点为(20, 20, 80)m,终点为(2 000, 2 000, 80)m,无人机实时感知周围200 m范围内的障碍物,构造当前位置到目标点NBO定点目标跟踪问题。每个迭代中,无人机都从相同的起点朝着目标点飞行,并实时对感知到的障碍物进行避障,以便直观地看出参数调整后的跟踪效果。整个过程逐步通过Q-Learning强化学习框架进行MPC策略模型优化,结果如图5(b)~图5(d)所示。

从图5(a)可以看出,初始迭代中无人机和障碍物发生了碰撞,在后续目标跟踪过程中,无人机逐渐远离了障碍物,3条路径(红、绿、蓝)分别对应图5(d)中的EP.1、EP.2、EP.18这3次迭代的结果,同时图5(d)中的距离成本、输入成本及状态约束成本分别为图5(b)阶段成本的3个组成分量。从图5(b)可以看出,随着迭代次数的增加,整个迭代的阶段成本总和呈下降趋势。取其中的第1、第2、第18次迭代进行对比,从图5(d)中的距离成本可看出无人机逐渐远离了障碍物,图5(d)中的输入成本可看出输入的变化率在逐渐变小,图5(d)中的状态约束成本可看出违反约束的情况在逐渐缩小。图5(c)为参数化MPC中3个动态调整参数θ_p、θ_u、θ_o的变化趋势,能够看出θ_p和θ_o均发生了调整,θ_u一直维持在初始值状态。综合以上可以看出Q-Learning强化学习框架对NBO问题的结果产生了正向影响。

6.3 不同障碍物数量影响下的鲁棒性分析

针对QL-NBO算法鲁棒性分析,本文设计了障



(a) 测试环境　(b) 阶段成本变换曲线　(c) 参与学习参数

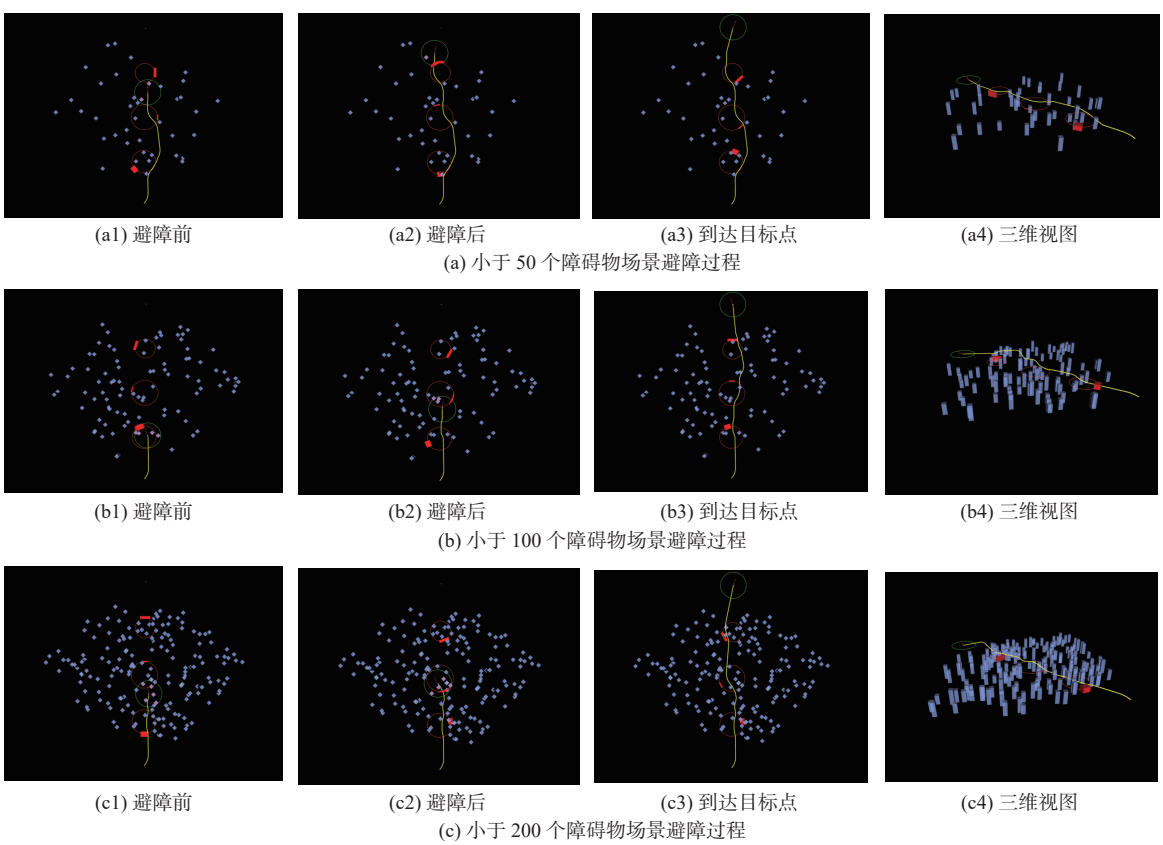
(d) 阶段成本组成分量的变化曲线

图 5　Q-Learning 对 NBO 的有效性仿真结果

Fig. 5　Simulation results of effectiveness of Q-Learning on NBO

碍物数量分别为 50、100、200 以内 3 种随机场景，并添加了 3 个以半径分别为 180、100、200 m 的圆形轨迹运动的动态障碍物。场景大小为 2 000 m×2 000 m×500 m, 规划模型和跟踪模型均为式 (1) 中的运动学模型。

实验结果如图 6 所示，其中，图 6(a)场景中包



(a1) 避障前　(a2) 避障后　(a3) 到达目标点　(a4) 三维视图

(a) 小于 50 个障碍物场景避障过程

(b1) 避障前　(b2) 避障后　(b3) 到达目标点　(b4) 三维视图

(b) 小于 100 个障碍物场景避障过程

(c1) 避障前　(c2) 避障后　(c3) 到达目标点　(c4) 三维视图

(c) 小于 200 个障碍物场景避障过程

图 6　3 种不同障碍物数量的仿真场景

Fig. 6　Presents three simulation scenarios with different numbers of obstacles

含障碍物数量在 50 以内，图 6(a1)和图 6(a2)表示仿真过程中避障前后状态，图 6(a3)和图 6(a4)分别为仿真完成后的顶视图和 3D 视图。图 6(b)和图 6(c)分别为障碍物数量为 100 和 200 以内的场景。每张图中的 3 个大小不同的红色方块是动态障碍物，红色圆圈是提前设定的动态障碍物轨迹，黄色的线是无人机在运动学模型作用下走过的航路，在无人机朝向目标点跟踪的过程中，会实时以半径 200 m 的圆(图中的绿色圆圈)向周围环境感知静态和动态障碍物，感知到的障碍物则显示为粉色实体矩形，未被感知到的障碍物则为半透明的蓝色矩形，场景中矩形障碍物的高度均在 100～120 m 范围内。此外，可在图 6(a2)、图 6(b1)、图 6 (b2) 和图 6 (c2) 看出，当无人机在靠近动态障碍物时，对障碍物未来若干个时间跨度进行了预测，本文设置的预测步长为 5。

图 7 展示了如图 6 所示 3 种障碍物场景下整个仿真期间无人机避障距离的变化过程。能够发现，在这 3 种场景的整个仿真进程中，无人机与障碍物的最小距离均大于 0，这表明本文 QL-NBO 算法具备良好的避障能力。图中当障碍物数量小于 50 和 100 时，出现了曲线断裂的现象，其原因在于此时无人机感知到 200 m 范围内的静态和动态障碍物数量均为 0，致使避障距离变为无穷大，这同时也体现出该场景中障碍物的分布相对较为稀疏。而当障碍物数量小于 200 时，曲线保持连续，这说明无人机在每一时间跨度下均能感知到静态或动态障碍物。结合图 6 和图 7 可以看出，3 种障碍物场景下本文 QL-NBO 算法均能完成避障，体现了其在 3 种场景下的求解成功率较高，具有较好的鲁棒性，较为稳定。

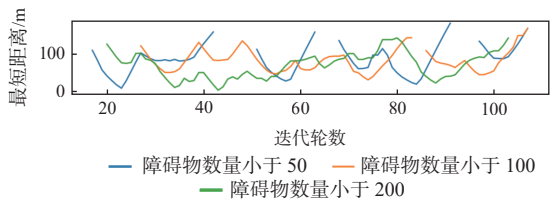


图 7 无人机与障碍物的最短距离

Fig. 7 Minimum distance between the UAV and obstacles

6.4 不同障碍物场景下求解时间对比

对算法求解时间的度量对于优化资源利用具有重要意义，尤其在资源有限的状况下，有助于使用者遴选出高效的算法。本文分别于障碍物数量小于 50、80、100、120、150、200 的障碍物场景下，针对 HybridA*算法[45]、NBO(Dual)算法[31-32]、NBO(LSE)算法[30]及本文 QL-NBO(LSE)算法，从起点至终点执行完一次定点目标跟踪任务，对其平均计算时间展开对比，结果如表 2 所示。

表 2 多种算法不同障碍物场景下的平均单次计算时间对比

Table 2 Comparison of average single calculation time of multiple algorithms in different obstacle scenarios

算法	时间/s					
	障碍物数量小于50	障碍物数量小于80	障碍物数量小于100	障碍物数量小于120	障碍物数量小于150	障碍物数量小于200
HybridA*(a_{cc} = 0.25; t = 5.0 s)	0.01	0.01	–	–	–	–
HybridA*(a_{cc} = 0.125; t = 5.0 s)	0.02	0.33	3.51	–	–	–
HybridA*(a_{cc} = 0.1; t = {1,2,3,4,5} s)	0.02	0.52	8.31	–	–	–
NBO(Dual)	0.22	0.21	0.24	0.83	0.84	0.83
NBO(LSE)	0.05	0.08	0.08	0.08	0.14	0.14
QL-NBO(LSE)	0.08	0.08	0.14	0.14	0.15	0.15

注：“–”表示搜索失败。

表 2 中的障碍物数量分别为各个场景中各自障碍物的总数，当障碍物越多时无人机所感知到的障碍物数量会越多，因此，每一时间跨度参与到优化中的障碍物数量会相应增加。此外，障碍物的数量都用小于进行表述是因为障碍物在生成时按照指定个数随机生成，当靠近起点或终点 500 m 范围内则不允许障碍物生成，因此，通常情况下生成的障碍物数量少于设定数量。

表 2 为 HybridA* 算法设定的 3 种离散加速度 a_{cc} 与时间步长的组合形式，当障碍物数量处于 80 以下时，所有算法均可实现规划任务。然而，当障碍物数量渐增至 100 时，HybridA*算法第 1 种参数无法在规定时间内完成计算；而当障碍物数量进一步增加至 120 时，HybridA*算法第 2 种和第 3 种参数组合也未能在规定时间内达成计算。通过对这 3 种参数的比较分析可知，Hybrid A* 算法受离散加速度的影响颇为显著。在障碍物相对稀少的场景中，仅需离散程度较低的加速度即可搜索出完整的从起点至终点的路径。但随着障碍物数量的增多，为搜索到路径，搜索维度需相应提升，所耗费的时间也随之增加。

反观 NBO(Dual)算法、NBO(LSE)算法及本文 QL-NBO(LSE)算法，其在所有障碍物环境中的表现均较为稳定。其中，与 NBO(Dual)算法相比，其余 2 种算法的计算耗时更少，展现出了较好的实时性。在应对小于 200 个障碍物时，每个迭代的平均

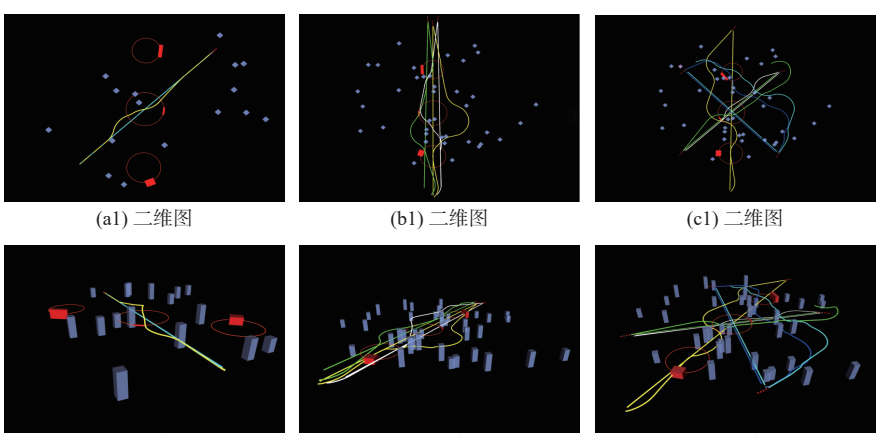
耗时大致维持在 0.1 s 左右, 具有较高的计算效率。此外, 由于本文 QL-NBO 算法 属于在线实时规划算法, 相较于 NBO(LSE) 算法, 该算法在计算时间方面的优势并不突出, 其优势主要体现在求解质量及整个跟踪过程的闭环性能上(6.2 节实验 已表明该点)。

进一步深入剖析 HybridA* 算法整体计算成功率较低的另一原因: eVTOL 在固定翼模式下存在非完整性约束, 由于失速速度的限制, 无人机无法将速度降至 0, 这在速度约束方面对算法的可行域造成了限制。当障碍物密集时, 尽管 HybridA*算法通过提高离散程度能够提升搜索成功率, 但其相应的耗时也会增加。对于在未知场景下以固定翼模式飞行的 eVTOL 而言, 该算法无法应用于其实时航路规划之中。需要指出的是, 这仅仅是针对静态障碍物环境而言, 若考虑动态障碍物, 将会面临更为严峻的挑战, 如何平衡其搜索维度和计算时间是一个难题。不过, 当障碍物较少时, 从表 2 可以看出, 其计算实时性较高。因此, 在障碍物较少的简单环境下, HybridA*算法具有显著的实时性优势, 且对于静态环境的离线规划, 或是具备完整性约束的系统, 例如, 能够进行微分平坦处理的四旋翼无人机系统, 由于速度可以为 0, 极大拓展了速度规划的范围, 此时应用 HybridA*算法也具有较好的优势, 关于本文涉及的 HybridA *算法, 详情可参考文献 [45]。

6.5　动态目标跟踪下单机和多机避障

实验 2、实验 3 和实验 4 均为静态目标下的测试, 为测试本文算法在动态目标下的跟踪避障能力及对于单机和多机规划的扩展能力, 本文设计了 1 架、3 架和 5 架无人机分别同时对动态目标跟踪的实验, 场景中放置了少于 50 个静态障碍物和 3 个动态障碍物, 仿真结果如图 8 所示, 避障距离变化如图 9 所示。其中, 针对 1 架无人机场景, 无人机 1 的起终点分别为(20,20,80)m 和(2 000,2 000,80)m, 跟踪目标起点和终点也为(20,20,80)m 和(2 000,2 000,80)m; 针对 3 架无人机场景, 无人机 1、无人机 2 和无人机 3 的起点分别为(20,20,80)m、(200,40,80)m 和(20, 200,80)m, 对应的终点分别为(2 000,2 000,80)m、(2 100,2 000,80)m 和(2 000,2 100,80)m, 相应每架无人机所跟踪的目标的起点分别为(20,20,80)m、(80, 20,80)m 和(20,80,80)m, 对应的终点分别为(2 000, 2 000,80)m、(2 000,2 020,80)m 和(2 020,2 000,80)m; 针对 5 架无人机场景, 无人机 1、无人机 2、无人机 3、无人机 4 和无人机 5 的起点分别为(20,20,80)m、(20,1 000,80)m、(2 000,1 020,80)m、(1 000, 20, 80)m 和(1 020,2 000,80)m, 对应的终点分别为(2 000,2 000, 80)m、(2 000,1 000,80)m、(20,1 020,80)m、(1 000, 2 000, 80)m 和(1 020,20,80)m, 相应每架无人机所跟踪的目标起点分别为(20,20,80)m、(20,1 000, 80)m, (2 000,1 200,80)m、(1 000, 20, 80)m 和(1 200, 2 000,80)m, 对应的终点分别为(2 000,2 000,80)m、(2000,1 000,80)m、(20,1 200,80)m、(1 000,2000,80)m 和(1 200,20,80)m。所有动态目标均以 20 m/s 的速度匀速直线朝对应的终点运动, 实验全程确保目标点是可知的, 但考虑了当前位置和目标位置通视和被障碍物阻挡的情况, 此时目标函数中对目标点的跟踪部分根据式(19)进行计算, 感知模块主要获取当前位置 200 m 范围内的障碍物参与规划。此外, 针对多目标跟踪, 需要将式(35)中目标函数增加关于无人机间距离的 LSE 平滑项, 且该距离的度量也采用本文提到的多面体间的全维度距离计算方式(见式(33))。

图 8 为 1 架、3 架和 5 架无人机在动态目标跟



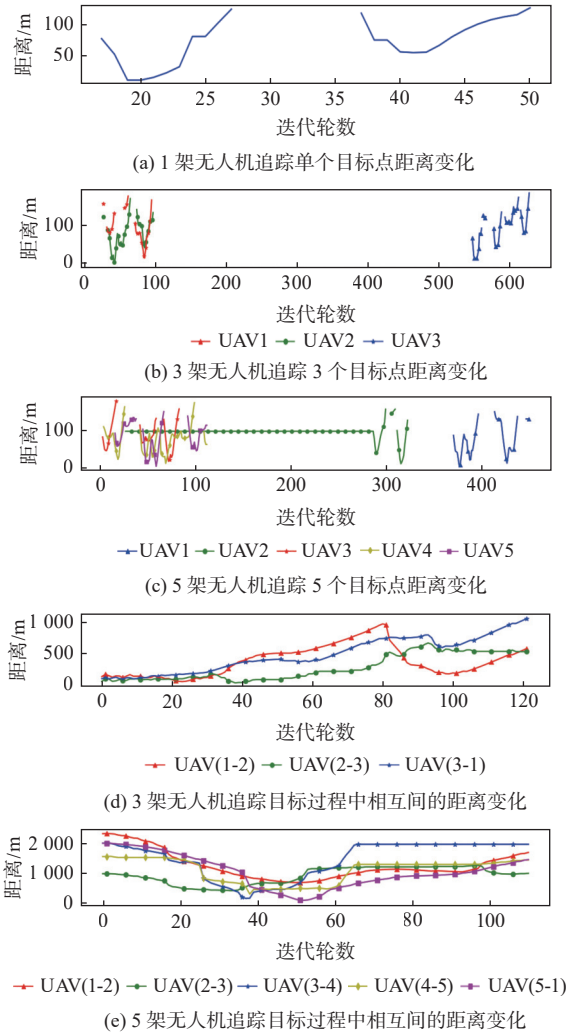
(a1) 二维图　(b1) 二维图　(c1) 二维图

(a2) 三维图　(b2) 三维图　(c2) 三维图

(a) 单架无人机追踪单个目标点　(b) 3 架无人机追踪 3 个目标点　(c) 5 架无人机追踪 5 个目标点

图 8　1 架、3 架、5 架无人机动态目标跟踪效果

Fig. 8　Dynamic target tracking effect of 1, 3, and 5 UAVs



(a) 1 架无人机追踪单个目标点距离变化

(b) 3 架无人机追踪 3 个目标点距离变化

(c) 5 架无人机追踪 5 个目标点距离变化

(d) 3 架无人机追踪目标过程中相互间的距离变化

(e) 5 架无人机追踪目标过程中相互间的距离变化

图 9 不同数量无人机跟踪时的距离变化

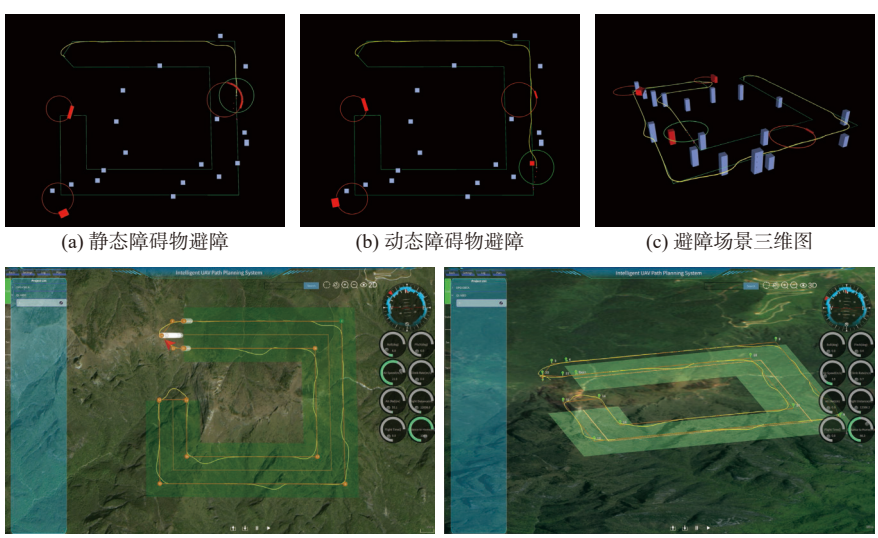
Fig. 9 Distance variations during tracking with different numbers of UAVs

踪场景下的仿真截图，每张图中直线为无人机所跟踪的目标，其中，图 8(b1)、图 8(b2)、图 8(c1)和图 8(c2)中的目标路径颜色和对应无人机路径颜色相同，方便区分。从图 8 可以直观地看出，单架和多架无人机场景下均能对动态目标完成跟踪并避障。图 9 为图 8 中对应的跟踪过程中无人机距离所感知范围内障碍物的最小距离的变化曲线，直观地从图中就能看出所有的避障距离均大于 0，表明无人机既完成了动态目标跟踪，又能躲避障碍物，也考虑了无人机间的距离，展示了本文 QL-NBO(LSE)算法对于单机和多机之间扩展的灵活性。

6.6 航路任务下的硬件在环测试

航线跟踪任务的规划模型使用了式(1)中的运动学模型，跟踪模型为 px4，并采用硬件在环仿真进行验证，控制系统采用 px4 原有的位置控制外环和姿态控制内环[46]，系统采用规划和控制分离的架构，在规划层采用收紧的 MPC，采用 Tube-based-MPC 进行优化问题求解，控制层则采用 eVTOL 本身的动力学模型进行跟踪，该方法的优势在于简化的模型能够加快 MPC 的求解，以提供满足实时性要求的预测参考轨迹引导无人机飞行。硬件在环仿真环节，首先，执行 px4 程序，再启动航路规划系统，然后，在地面站规划任务航线，并上传到飞控，规划系统订阅无人机的航路任务和实时位置，在地面站解锁并使无人机起飞则开始执行任务。该环节可以验证规划框架的稳定性和算法的实用性。无人机开始以旋翼模式飞到 80 m 高度，转换到固定翼模式，此时通过手动控制将无人机切换为 OffBoard 模式。

在整个任务执行过程中，无人机将航路规划系统的位置三元组作为参考，沿着任务航线飞行，当感知到周围存在障碍物时，系统开始实时在航线上生成要跟踪的目标航点 (图 10 中的紫色航点) 以引导无人机飞行，规划模块根据每个时间跨度是否感



(a) 静态障碍物避障 (b) 动态障碍物避障 (c) 避障场景三维图

(d) 地面站端顶视图 (e) 地面站端三维图

图 10 航线任务避障决策过程

Fig. 10 Obstacle avoidance path planning process of the flight mission

知到障碍物而选择开启或停用, 整个任务场景中, 静态和动态障碍物的设置和定点跟踪任务是相同的, 不同之处在于本文将第 3 个时间跨度的航点实时发送给飞控端跟踪, 主要是因为在硬件在环仿真中, 每次无人机前进的位置和所期望的位置会有偏差, 如果将第 2 个航点发送给无人机, 由于受初始航向的影响, 使跟踪效果变差, 避障过程如图 10 所示。

从图 10 可以看出, 无人机在感知到障碍物时会开启规划模块, 且能够生成未来 5 个时间跨度的航点 (可以在图 10(a)中看到有 6 个红色的离散航点), 在生成的航点前方的紫色航点为动态生成的目标点。图 10(a)为无人机正在对动态障碍物进行避障, 可以看到, 红色障碍物向前生成了 5 个时间跨度的半透明红色障碍物, 用来模拟对动态障碍物的轨迹预测, 结合图 10(b)看出, 此时无人机的轨迹变化不大, 表明考虑了动态障碍物的轨迹信息; 图 10(b)为无人机正在对静态障碍物进行避障, 感知到 200 m 范围的障碍物为其右侧高亮长方体, 结合图 10(c)可以看出完成了对静态障碍物的避障; 图 10(c)为任务场景的三维图, 图中无人机虽然感知到了一个静态障碍物(绿色圆圈所包含的红色长方体), 但由于距离障碍物较远, 因此, 对执行轨迹影响不大。图 10(d)和图 10(e)分别为本次任务执行中无人机走过的航路在地面站端的二维和三维可视化。可以看出, 本文规划框架能够综合任务需要和避障需要进行决策规划并完成任务。

7　结　论

1) 本文对未知环境下无人机目标跟踪航路规划问题展开深入研究, 设计并开发了能够随机生成障碍物的仿真环境, 用以对本文 QL-NBO 算法和规划框架进行验证。

2) 本文方法深入探究了 POMDP 模型, 借助 MPC 实现了 NBO 算法。针对避障需求, 引入全维度避障距离度量, 在满足避障基本要求的前提下, 当障碍物数量相同时, 相比基于对偶变换原理的避障优化问题提高了求解效率。

3) 针对 MPC 本身难以消除误差的问题, 提出 QL-NBO 算法, 该算法基于 Q-Learning 强化学习框架达成了 MPC 框架中相关参数的动态调整, 有效提升了目标跟踪的闭环性能, 同时通过实验验证了其从单机到多机的协同避障方面具有良好扩展性。

4) 在系统应用方面, 本文提供了一个实时航路规划系统, 并基于 px4、ROS、RVIZ、Gazebo 开展了软件仿真和硬件在环仿真, 通过实验验证了本文规划框架的可行性、鲁棒性和有效性。

未来, 可基于该框架和算法进一步扩展其在多机任务中的应用, 如集群打击, 协同作业等。可在感知和传感器误差的灵敏性问题、多无人机间的通信覆盖和优化及协同作业中的执行效率等方面进行深入研究。此外, 在硬件在环仿真时, 无人机的跟踪效果受到初始航迹偏角的影响, 导致与软件仿真的结果存在偏差。后续可基于该框架, 进一步从速度优化和控制器设计的角度进行研究以改善其跟踪效果。

参考文献（References）

[1] Dong J, Ota K, Dong M X. UAV-based real-time survivor detection system in post-disaster search and rescue operations[J]. IEEE Journal on Miniaturization for Air and Space Systems, 2021, 2(4): 209-219.

[2] Liu X L, Ansari N. Resource allocation in UAV-assisted M2M communications for disaster rescue[J]. IEEE Wireless Communications Letters, 2019, 8(2): 580-583.

[3] Alharam A, Almansoori E, Elmadeny W, et al. Real time AI-based pipeline inspection using drone for oil and gas industries in Bahrain[C]//Proceedings of the International Conference on Innovation and Intelligence for Informatics, Computing and Technologies. Piscataway: IEEE Press, 2020: 1-5.

[4] Shamsoshoara A, Afghah F, Blasch E, et al. UAV-assisted communication in remote disaster areas using imitation learning[J]. IEEE Open Journal of the Communications Society, 2021, 2: 738-753.

[5] Guo Q, Peng J, Xu W Z, et al. Minimizing the longest tour time among a fleet of UAVs for disaster area surveillance[J]. IEEE Transactions on Mobile Computing, 2022, 21(7): 2451-2465.

[6] Dong W J, Qi X Z, Chen Z X, et al. An indoor path planning and motion planning method based on POMDP[C]//Proceedings of the IEEE International Conference on Robotics and Biomimetics. Piscataway: IEEE Press, 2017: 1564-1570.

[7] Trotti F, Farinelli A, Muradore R. An online path planner based on POMDP for UAVs[C]//Proceedings of the European Control Conference. Piscataway: IEEE Press, 2023: 1-6.

[8] Morere P, Marchant R, Ramos F. Sequential Bayesian optimization as a POMDP for environment monitoring with UAVs[C]//Proceedings of the IEEE International Conference on Robotics and Automation. Piscataway: IEEE Press, 2017: 6381-6388.

[9] Deng D H, Wang C W, Wang W D. Joint air-to-ground scheduling in UAV-aided vehicular communication: a DRL approach with partial observations[J]. IEEE Communications Letters, 2022, 26(7): 1628-1632.

[10] Aniceto Dos Santos M A, Vivaldini K C T. A modeling strategy using Bayesian optimization with POMDP for exploration and informative path planning of UAVs in monitoring of forest[C]//Proceedings of the International Conference on Computational Science and Computational Intelligence. Piscataway: IEEE Press, 2022: 451-456.

[11] Silver D, Veness J. Monte-Carlo planning in large POMDPs [C]//Proceedings of the 24th International Conference on Neural Information Processing Systems - Volume 2. New York: ACM, 2010: 2164-2172.

[12] Ye N, Somani A, Hsu D, et al. DESPOT: online POMDP planning with regularization[J]. Journal of Artificial Intelligence Research, 2017, 58(1): 231-266.

[13] Bai H Y, Cai S J, Ye N, et al. Intention-aware online POMDP planning for autonomous driving in a crowd[C]//Proceedings of the IEEE International Conference on Robotics and Automation. Piscataway: IEEE Press, 2015: 454-460.

[14] Kurniawati H, Yadav V. An online POMDP solver for uncertainty planning in dynamic environment[M]//Robotics Research: The 16th International Symposium ISRR. Berlin: Springer, 2016: 611-629.

[15] Priyadarshini G N, Hsu D, Sun L W. DESPOT-alpha: online POMDP planning with large state and observation spaces[C]//Proceedings of the Robotics: Science and Systems XV. Robotics: Science and Systems Foundation, 2019.

[16] Klimenko D, Song J, Kurniawati H. TAPIR: a software toolkit for approximating and adapting POMDP solutions online[C]//Proceedings of the Australasian Conference on Robotics and Automation. Sydney: Australian Robotics and Automation Association, 2014: 28.

[17] Cunningham A G, Galceran E, Eustice R M, et al. MPDM: multipolicy decision-making in dynamic, uncertain environments for autonomous driving[C]//Proceedings of the IEEE International Conference on Robotics and Automation. Piscataway: IEEE Press, 2015: 1670-1677.

[18] Ding W C, Zhang L, Chen J, et al. EPSILON: an efficient planning system for automated vehicles in highly interactive environments[J]. IEEE Transactions on Robotics, 2022, 38(2): 1118-1138.

[19] Adiprawita W, Ahmad A S, Sembiring J, et al. Reinforcement learning with heuristic to solve POMDP problem in mobile robot path planning[C]//Proceedings of the International Conference on Electrical Engineering and Informatics. Piscataway: IEEE Press, 2011: 1-5.

[20] Lee Y, Cai P P, Hsu D. MAGIC: learning macro-actions for online POMDP planning[C]//Proceedings of the Robotics: Science and Systems XVII. Robotics: Science and Systems Foundation, 2021.

[21] Miller S A, Harris Z A, Chong E K P. A POMDP framework for coordinated guidance of autonomous UAVs for multitarget tracking[J]. EURASIP Journal on Advances in Signal Processing, 2009, 2009(1): 724597.

[22] Ragi S, Chong E K P. UAV path planning in a dynamic environment via partially observable Markov decision process[J]. IEEE Transactions on Aerospace and Electronic Systems, 2013, 49(4): 2397-2412.

[23] Yang Q M, Zhang J D, Shi G Q. Modeling of UAV path planning based on IMM under POMDP framework[J]. Journal of Systems Engineering and Electronics, 2019, 30(3): 545-554.

[24] Eaton C M, Chong E K P, Maciejewski A A. Robust UAV path planning using POMDP with limited FOV sensor[C]//Proceedings of the IEEE Conference on Control Technology and Applications. Piscataway: IEEE Press, 2017: 1530-1535.

[25] Cai W Q, Kordabad A B, Esfahani H N, et al. MPC-based reinforcement learning for a simplified freight mission of autonomous surface vehicles[C]//Proceedings of the 60th IEEE Conference on Decision and Control. New York: ACM, 2021: 2990-2995.

[26] Gros S, Zanon M. Data-driven economic NMPC using reinforcement learning[J]. IEEE Transactions on Automatic Control, 2020, 65(2): 636-648.

[27] Esfahani H N, Kordabad A B, Gros S. Approximate robust NMPC using reinforcement learning[C]//Proceedings of the European Control Conference. Piscataway: IEEE Press, 2021: 132-137.

[28] Lagoudakis M G, Parr R. Least-squares policy iteration[J]. Journal of Machine Learning Research, 2003, 4: 1107-1149.

[29] Airaldi F, De Schutter B, Dabiri A. Reinforcement learning with model predictive control for highway ramp metering[EB/OL]. (2025-02-03)[2026-09-11]. https://doi.org/10.48550/arXiv.2311.08820.

[30] Han Z C, Wu Y W, Li T, et al. An efficient spatial-temporal trajectory planner for autonomous vehicles in unstructured environments[J]. IEEE Transactions on Intelligent Transportation Systems, 2024, 25(2): 1797-1814.

[31] Zhang X J, Liniger A, Borrelli F. Optimization-based collision avoidance[J]. IEEE Transactions on Control Systems Technology, 2021, 29(3): 972-983.

[32] He R X, Zhou J Y, Jiang S, et al. TDR-OBCA: a reliable planner for autonomous driving in free-space environment[C]//Proceedings of the American Control Conference. Piscataway: IEEE Press, 2021: 2927-2934.

[33] Wang Y X, Wang H L, Lun Y B. Shortest path planning of UAV for target tracking and obstacle avoidance in 3D environment[C]//Proceedings of the 39th Chinese Control Conference. Piscataway: IEEE Press, 2020: 3385-3390.

[34] Das M, Ghosh S K. Reducing parameter value uncertainty in discrete Bayesian network learning: a semantic fuzzy Bayesian approach[J]. IEEE Transactions on Emerging Topics in Computational Intelligence, 2021, 5(3): 361-372.

[35] Youn W, Ko N Y, Gadsden S A, et al. A novel multiple-model adaptive Kalman filter for an unknown measurement loss probability[J]. IEEE Transactions on Instrumentation and Measurement, 2021, 70: 8500511.

[36] Partovibakhsh M, Liu G J. An adaptive unscented Kalman filtering approach for online estimation of model parameters and state-of-charge of lithium-ion batteries for autonomous mobile robots[J]. IEEE Transactions on Control Systems Technology, 2015, 23(1): 357-363.

[37] Dong F, Li X C, You K Y, et al. Standoff tracking using DNN-based MPC with implementation on FPGA[J]. IEEE Transactions on Control Systems Technology, 2023, 31(5): 1998-2010.

[38] Bauersfeld L, Spannagl L, Ducard G J J, et al. MPC flight control for a tilt-rotor VTOL aircraft[J]. IEEE Transactions on Aerospace and Electronic Systems, 2021, 57(4): 2395-2409.

[39] Feng Z W, Cao G Y, Grigoriadis K M, et al. Secure MPC-based path following for UAS in adverse network environment[J]. IEEE Transactions on Industrial Informatics, 2023, 19(11): 11091-11101.

[40] Jacquet M, Kivits M, Das H, et al. Motor-level N-MPC for cooperative active perception with multiple heterogeneous UAVs[J]. IEEE Robotics and Automation Letters, 2022, 7(2): 2063-2070.

[41] Liu Y Y, Van Schijndel J, Longo S, et al. UAV energy extraction with incomplete atmospheric data using MPC[J]. IEEE Transactions on Aerospace and Electronic Systems, 2015, 51(2): 1203-1215.

[42] Falugi P, Mayne D Q. Getting robustness against unstructured un-

certainty: a tube-based MPC approach[J]. IEEE Transactions on Automatic Control, 2014, 59(5): 1290-1295.

[43] Xiao K, Tan S C, Wang G H, et al. XTDrone: a customizable multi-rotor UAVs simulation platform[C]//Proceedings of the 4th International Conference on Robotics and Automation Sciences. Piscataway: IEEE Press, 2020: 55-61.

[44] Xiao K, Ma L, Tan S C, et al. Implementation of UAV coordination based on a hierarchical multi-UAV simulation platform[C]// Proceedings of the Advances in Guidance, Navigation and Control. Singapore: Springer Singapore, 2022: 5131-5143.

[45] Zhou B Y, Gao F, Wang L Q, et al. Robust and efficient quadrotor trajectory generation for fast autonomous flight[J]. IEEE Robotics and Automation Letters, 2019, 4(4): 3529-3536.

[46] Yang J, Liu C J, Coombes M, et al. Optimal path following for small fixed-wing UAVs under wind disturbances[J]. IEEE Transactions on Control Systems Technology, 2021, 29(3): 996-1008.

UAV online path planning based on POMDP and parameterized MPC in uncertain environments

YUAN Jinbiao[1]，LIU Haiyan[2]，MA Bodi[3]，ZHAO Leyang[4]，GAO Honggang[1]，LIU Zhenbao[1,*]

(1. School of Civil Aviation，Northwestern Polytechnical University，Xi'an 710072，China；
2. Shanghai Civil Aviation Control and Navigation System Co.，Ltd，Shanghai 200241，China；
3. School of Information Engineering，Chang'an University，Xi'an 710016，China；
4. School of Architecture & Urban Planning，Shenzhen University，Shenzhen 518060，China)

Abstract: It is challenging to eliminate positioning and perception ambiguities when unmanned aerial vehicles (UAVs) operate autonomously in unfamiliar areas. In view of this, it is necessary to conduct research on path planning to enhance the effectiveness of the path under the interference of uncertain factors and thereby improve the target tracking ability of UAVs. The goal of applying model predictive control（MPC）to the real-time path planning system is currently accomplished by Tube-based-MPC, which supports the separation architecture of planning and control, and nominal belief state optimization (NBO), which offers a theoretical foundation for the uncertainty of target tracking. On this premise, this paper proposes an online path planning framework to solve the problems of target tracking and obstacle avoidance in unknown environments. First, the path planning problem is abstracted into a partially observable Markov decision process (POMDP) model, and the Kalman filter is used to estimate the belief state, integrating the uncertainty measure of target positioning into the framework. Secondly, based on the simplified 3-dimensional motion model, a tightened constraint Tube-based-MPC optimization framework is constructed to solve the POMDP. The path planning for a specific continuous time span in the future is achieved by building a nonlinear optimization problem that includes the belief state and unknown barriers. Then, the shapes of UAVs and obstacles are fully considered in the optimization problem, and a full-dimensional obstacle avoidance distance metric is introduced to significantly expand the feasible region of the optimization problem. Finally, the Q-learning reinforcement learning framework is introduced, and MPC is used as the policy function approximator to improve the closed-loop performance of target tracking. Simulation experiments and hardware-in-the-loop experiments verified the effectiveness of the proposed method.

Keywords: partially observable Markov decision process; Tube-based-MPC; path planning; reinforcement learning; target tracking

Received：2024-09-06；**Accepted**：2024-11-15；**Published Online**：2024-11-22 17：09

URL：link.cnki.net/urlid/11.2625.V.20241122.1600.001

Foundation items：National Natural Science Foundation of China (52072309,12102345); Key Research and Development Program of Shaanxi Province (2019ZDLGY14-02-01,2022FP-40); Shenzhen Fundamental Research Program (JCYJ20190806152203506); Aeronautical Science Foundation (ASFC-2018ZC53026); Suzhou Municipal Science and Technology Bureau under Grant (ZXL2023177); Taicang Basic Research Program under Grant (TC2023JC13)

* **Corresponding author**. E-mail：liuzhenbao@nwpu.edu.cn