SCIENTIA SINICA Informationis

面向特殊应用场景的无人机智能决策与控制专刊・论文



面向复杂未知多障碍环境的多无人机分布式在线轨 迹规划

张学伟1,田栢苓1*,鲁瀚辰2,谌宏鸣1,宗群1

1. 天津大学电气自动化与信息工程学院, 天津 300072

2. 北京机电工程研究院复杂系统控制与智能协同技术重点实验室, 北京 100074

* 通信作者. E-mail: bailing_tian@tju.edu.cn

收稿日期: 2022-01-29; 修回日期: 2022-04-05; 接受日期: 2022-05-06; 网络出版日期: 2022-09-09

国家优秀青年科学基金 (批准号: 62022060) 和国家自然科学基金面上项目 (批准号: 62003236, 61873340, 61903349) 资助

摘要 考虑复杂未知多障碍环境对无人机实时轨迹规划性能的影响,提出了基于 Tube-MPC 和模型 预测路径积分 (model predictive path integral, MPPI) 控制相结合的多无人机分布式实时轨迹规划框 架与方法. 首先,考虑无人机在多障碍环境下的避碰避障需求,构造代价函数表征轨迹规划过程中的 约束条件,将多无人机的轨迹规划问题转化为随机最优控制问题.其次,借鉴 Tube-MPC 思想,设计并 实现了多无人机分布式轨迹规划框架,通过将低频标称控制器与高频辅助控制器串联保证了系统的实 时性和鲁棒性. 再次,为避免传统方法在求解过程中的维数灾难,提出基于 MPPI 的多无人机异步轨 迹规划方法,该方法通过基于 GPU 的并行蒙特卡洛 (Monte-Carlo) 随机前向采样技术,将多无人机随 机最优控制问题的求解转化为给定代价函数下对采样轨迹期望的求解,进而获得最优控制序列,其显 著特点是求解速度快且避免了基于梯度求解方法对约束条件和代价函数连续性及凸特性的要求. 最 后,通过 Gazebo 虚拟仿真平台,在复杂未知多障碍环境下对算法的有效性进行了验证.

关键词 未知多障碍环境,多无人机,在线轨迹规划,模型预测路径积分,GPU并行加速

1 引言

近年来,由于多无人机可以充分利用有限的单机资源共同完成复杂的任务,其在情报工作、搜索救援、侦察预警、目标捕捉等方面正发挥着越来越重要的作用,已经成为未来军事战场中不可或缺的力量.与此同时,由于地下空间、城市楼宇及丛林等典型复杂未知多障碍环境在未来城市巷战及野外低空作战中的重要作用,面向复杂未知多障碍场景的多无人机在线协同规划受到广泛关注,被认为是提高多无人机在复杂环境下自主性和灵活性的关键技术^[1~3].

引用格式: 张学伟,田栢苓,鲁瀚辰,等. 面向复杂未知多障碍环境的多无人机分布式在线轨迹规划. 中国科学: 信息科学, 2022, 52: 1627–1641, doi: 10.1360/SSI-2022-0048
 Zhang X W, Tian B L, Lu H C, et al. Multi-UAV decentralized online trajectory planning in complex unknown obstacle-rich environments (in Chinese). Sci Sin Inform, 2022, 52: 1627–1641, doi: 10.1360/SSI-2022-0048

ⓒ 2022《中国科学》杂志社

多无人机的轨迹规划本质上是一个受多约束、多变量及非线性综合影响的复杂最优控制问题,目 前该问题按照求解方式的不同,主要有集中式和分布式两类方法. 文献 [4] 考虑多障碍环境对多机器 人轨迹求解的影响,提出了基于混合整数线性规划 (mixed integer linear programming, MILP) 的多机 器人轨迹规划方法,该方法通过采用二元变量表示碰撞约束,保障了多机器人执行任务过程中能耗最 小. 进一步, 为了提升轨迹的动力学可行性, Kumar 等^[5,6] 先后提出了基于混合整数二次规划 (mixed integer quadratic program, MIQP) 和二次规划 (quadratic program, QP) 的方法构建优化问题求解多 无人机轨迹, 解决了异构无人机编队重构与轨迹规划问题. 基于优化求解思想, 文献 [7] 考虑固定时 间内多无人机避碰轨迹生成问题,提出了基于序列凸规划 (sequential convex programming, SCP) 的 求解方法,该方法通过采用凸约束逼近非凸约束迭代求解策略,实现了空旷环境下多无人机的避碰飞 行. Chen 等^[8]为了进一步提升实时性,研究了基于增量序列凸规划 (incremental sequential convex programming, ISCP) 的多无人机轨迹规划方法, 该方法将 SCP 方法与贪婪求解策略结合, 实现了非凸 约束条件下多无人机避碰避障飞行.随后,Robinson等^[9]为进一步提升非凸约束下轨迹求解成功率, 提出了基于 LS-PS (level set-pseudospectral) 的多智能体轨迹规划方法, 仿真结果验证了该算法求解非 凸问题的有效性. 为进一步提升求解规模, 文献 [10,11] 采用基于点 - 线构成的路图标注环境中障碍 物,并通过图搜索方法离线得到无人机轨迹,然后基于梯度下降方法优化无人机轨迹,从而实现了无人 机集群规划. 从提升集群轨迹离线求解速率的角度出发, Hamer 等 [12] 提出了基于 GPU 加速的方法, 实现了多无人机轨迹的高效率求解.进一步,为改进优化问题求解过程对梯度信息的依赖, Williams 等^[13]提出了 MPPI 方法, 避免了多无人机轨迹求解时梯度计算. 虽然上述集中式求解方法有效解决 了多无人机轨迹规划问题,但是随着无人机数量增加,计算复杂度会急剧增加.

近年来,基于分布式的多无人机轨迹规划算法得到了广泛研究.文献 [14,15] 考虑多智能体避碰问题,提出了基于速度障碍 (velocity obstacle, VO) 的多智能体避碰轨迹生成方法,实现了多智能体轨迹的快速生成. Manocha 等^[16] 将该方法与模型预测控制 (model predictive control, MPC) 方法结合,提出了一种满足动力学约束的多无人机分布式轨迹规划方法,实现了多无人机避碰轨迹的在线求解. 进一步,为了降低通讯网络对分布式规划的影响,Zhou 等^[17]提出了基于 BVC (buffered Voronoi cell) 的多智能体避碰方法,该方法在多智能体轨迹求解时仅需要机器人之间彼此交互相对位置信息,解决了通讯带宽受限时多无人机轨迹在线规划问题. 为进一步提升安全性,文献 [18]提出了 PBVC (probabilistic buffered Voronoi cell)方法,在传感器测量存在噪声时,实现了多智能体路径规划. 从定位不确定性角度考虑,Zhu 等^[19]提出了基于 B-UAVC (buffered uncertainty-aware Voronoi cell)的多机器人分布式在线轨迹规划方法,该方法实现了多智能体存在定位误差情况下的轨迹规划. Luis等^[20,21]为进一步提高轨迹求解成功率,研究了分布式模型预测控制 (distributed model predictive control, DMPC)方法,实现了多无人机点对点飞行. 随后,文献 [22]考虑多无人机避碰避障问题,研究了基于优先级策略的轨迹搜索算法,并通过样条曲线平滑轨迹,保障了多障碍环境下无人机的安全飞行,且进一步提升了多无人机在线轨迹生成的实时性.

国内外学者在多无人机轨迹规划领域进行了大量的研究工作,并取得了丰硕的研究成果,但是出于计算实时性以及轨迹可行性要求,现有工作在对多无人机轨迹进行求解时往往依赖如下假设:(1)时间同步,假设所有无人机采用一致的全局时间;(2)代价连续,假设表征轨迹所受约束的代价函数或约束条件是连续的甚至是凸函数;(3)环境已知,假设环境中障碍物信息完全已知或部分已知.然而,实际飞行中,很难保证多机时间完全同步和代价函数的处处连续及凸特性.因此,迫切需要研究更普适的多无人机轨迹规划方法.为此,本文将我们前期工作^[23,24]进行了单机向多机的推广,提出了面向复杂未知多障碍环境的多无人机分布式在线轨迹规划方法.本文的主要创新如下.(1)设计了多无人

Method	Reciprocal collision avoidance	Obstacle avoidance	Asynchronous	Cost-function continuity	Runtime
[10, 11]	\checkmark	\checkmark	×	Required	Very slow
[13]	\checkmark	\checkmark	×	Not required	Fast
[14, 15]	\checkmark	\checkmark	×	Required	Very fast
[16]	\checkmark	×	\checkmark	Required	Fast
[20]	\checkmark	×	×	Required	Slow
[22]	\checkmark	\checkmark	\checkmark	Required	Fast
Proposed	\checkmark	\checkmark	\checkmark	Not required	Fast

表 1 多无人机轨迹规划方法对比

Table 1	Comparison	of multi-UAV	trajectory	planning	methods
---------	------------	--------------	------------	----------	---------

机分布式轨迹规划框架,该框架借鉴 Tube-MPC 思想,通过将低频的标称控制器与高频的辅助控制器 串联保证了系统的实时性与鲁棒性. (2) 提出了基于 MPPI 的多无人机异步轨迹规划方法,该方法通 过基于 GPU 的并行蒙特卡洛随机采样技术保证了无人机轨迹生成的高效性与安全性,其显著特点是 不依赖于代价函数和约束条件的梯度信息,避免了对其连续性和凸特性的要求. 将本文所提方法与其 他多无人机轨迹规划方法从相互避碰、避障、时间异步、代价函数连续、运行时间 5 个方面进行对比, 结果如表 1 所示.

本文的内容安排如下:第2节阐述本文解决的问题,并构建无人机动态方程与约束条件,将无人 机轨迹规划问题转化为一个随机最优控制问题,第3节介绍复杂未知多障碍环境下的多无人机分布式 在线轨迹规划框架,第4节介绍基于 MPPI 的多无人机异步轨迹规划方法,第5节利用 ROS-Gazebo 虚拟仿真平台对本文所提算法的性能进行测试验证,第6节对本文研究工作进行归纳总结与展望.

2 问题描述

考虑一个具有 $N \in \mathbb{Z}$ 个无人机的编组,其所在环境空间表示为 $W \subseteq \mathbb{R}^3$,且假设空间完全未知.无 人机在执行任务过程中利用机载传感器对未知空间探索,并将环境中已经探索到的障碍物表示为 O, 未探索的空间表示为 U,那么可用于无人机生成轨迹的安全空间为 $F = W \setminus O \setminus U$.若无人机位于空间 W 中某一位置 $x \in \mathbb{R}^3$,则用 $\mathcal{R}(x)$ 表示,并将第 i 架无人机的起始位置和目标位置分别表示为 $s_i \in F$ 和 $g_i \in F$.假设编组中任意两架无人机 i, j 都具有不同的起始点和目标点,即满足 $\mathcal{R}(s_i) \cap \mathcal{R}(s_j) = \emptyset$ 和 $\mathcal{R}(g_i) \cap \mathcal{R}(g_j) = \emptyset$.对于第 i 架无人机,定义连接起点 s_i 与终点 g_i 的轨迹为 Q_i^m ,该轨迹共包含 m 个航点 $\mathrm{pos}_i^l \in Q_i^m, \forall l \in [0,m)$,且满足 $\mathrm{pos}_i^0 = s_i$ 与 $\mathrm{pos}_i^{m-1} = g_i$.多无人机轨迹规划的目标是生成一 条从起始点到目标点的安全轨迹,确保避开环境中障碍物且在同一时刻不会与其他无人机发生碰撞, 因此需满足以下条件:

$$\mathcal{R}\left(\mathrm{pos}_{i}^{l}\right) \in \mathcal{F}, \ \forall l \in [0, m), \ \forall i \in [0, N),$$
(1)

$$\mathcal{R}(t_i, \mathrm{pos}_i) \cap \mathcal{R}(t_j, \mathrm{pos}_j) = \emptyset, \ \forall i \neq j \in [0, N), \ t_i = t_j,$$
(2)

其中, pos_i , pos_j 分别表示第 $i \neq j$ 架无人机生成轨迹的所有航迹点, $\mathcal{R}(t, x \in \mathbb{R}^3)$ 表示无人机在时刻 t的位置为 x.

注释1 本文算法推导以第 i 架无人机为例, 为表述方便, 在不产生歧义的情况下, 忽略了角标 i.

2.1 无人机动态方程

本文以四旋翼无人机为例,构建其动态方程.由于四旋翼无人机具有微分平坦特性^[25],因此可以用 4 个平坦输出 $[p_x, p_y, p_z, \psi]^T$ 及其微分表示该系统的状态 **X** 和控制输入 U = u + v.四旋翼无人机的动态方程如下:

$$\dot{\boldsymbol{X}} = \boldsymbol{A}\boldsymbol{X} + \boldsymbol{B}\boldsymbol{U},$$

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{0}_{6\times3} \ \boldsymbol{a} \\ \boldsymbol{0}_{4\times3} \ \boldsymbol{0}_{4\times7} \end{bmatrix}, \ \boldsymbol{B} = \begin{bmatrix} \boldsymbol{0}_{6\times4} \\ \boldsymbol{I}_{4\times4} \end{bmatrix}, \ \boldsymbol{a} = \begin{bmatrix} \boldsymbol{I}_{6\times6} \\ \boldsymbol{0}_{6\times1} \end{bmatrix}^{\mathrm{T}},$$
(3)

其中, $X = [p_x, p_y, p_z, v_x, v_y, v_z, a_x, a_y, a_z, \psi]^T$ 表示无人机的位置、速度、加速度以及偏航角, $u = [\dot{a}_x, \dot{a}_y, \dot{a}_z, \dot{\psi}]^T$ 表示无人机的控制输入, 即位置的三阶导数与偏航角速率, v 表示随机控制输入.

2.2 约束条件分析

无人机轨迹规划过程中,由于自身动力学限制以及避碰避障需求,为保证生成轨迹的安全性与平 滑性,轨迹规划过程要满足一些约束条件,主要包括引导约束、避障约束、避碰约束和动力学约束.

2.2.1 引导约束

为确保无人机轨迹迭代求解过程中具有较好的初值,本文采用混合 A* 算法^[26] 搜索初始轨迹. 以 混合 A* 搜索得到的轨迹终点状态作为期望目标状态,以本文提出的基于 MPPI 规划方法得到的控制 输入序列积分得到的预测状态作为无人机实际状态,构造引导代价函数 q_t 惩罚无人机预测状态与期 望状态之间的偏差,形式如下:

$$q_t = k_p \|p - p_d\| + k_v \|v - v_d\|,$$
(4)

其中, p 和 v 分别表示无人机预测轨迹的位置和速度, p_d 和 v_d 分别表示期望位置和速度, k_p 与 k_v 表示权重.

2.2.2 避障约束

为确保安全飞行,无人机轨迹需要避开环境中障碍物,构造避障代价函数 q_c 惩罚距离障碍物较近的轨迹点, q_c 具体表达形式如下:

$$q_{c} = \begin{cases} k_{\rm col}, & d_{\rm obs} < d_{\rm min}^{\rm obs}, \\ k_{\rm col} \left(\frac{d_{\rm max}^{\rm obs} - d_{\rm obs}}{\delta}\right)^{p}, \ d_{\rm min}^{\rm obs} \leqslant d_{\rm obs} \leqslant d_{\rm max}^{\rm obs}, \\ 0, & d_{\rm obs} > d_{\rm max}^{\rm obs}, \end{cases}$$
(5)

其中, d_{obs} 表示无人机到最近障碍物的距离, d_{min}^{obs} 与 d_{max}^{obs} 是常值, 分别表示最小和最大的距离阈值, $\delta = d_{max}^{obs} - d_{min}^{obs}$, p > 0. 当轨迹点到最近障碍物距离大于最大距离阈值时, 认为轨迹安全, 不会发生 碰撞; 当轨迹点到最近障碍物距离小于最小距离阈值时, 认为轨迹发生碰撞概率很大, 给予一个代价 为 $k_{col} > 0$ 的惩罚; 当轨迹点到最近障碍物距离处于最小和最大距离阈值之间时, 给予一个代价处于 [0, k_{col}] 之间的惩罚, 且惩罚代价与 d_{obs} 的值成反比.

2.2.3 避碰约束

为确保无人机之间不会发生碰撞,构造避碰代价函数 q_m^{ij} 惩罚与其他无人机距离较近的轨迹点. 对于第 *i* 架无人机, q_m^{ij} 具体表达形式如下:

$$q_{m}^{ij} = \begin{cases} k_{m}, & d_{ij} < d_{\min}^{\text{thr}}, \\ k_{m} \left(\frac{d_{\max}^{\text{thr}} - d_{ij}}{d_{\max}^{\text{thr}} - d_{\min}^{\text{thr}}} \right)^{k}, d_{\min}^{\text{thr}} \leqslant d_{ij} \leqslant d_{\max}^{\text{thr}}, \\ 0, & d_{ij} > d_{\max}^{\text{thr}}, \end{cases}$$

$$d_{ij} = \left\| E^{1/2} \left[\text{pos}_{i}^{l} - \text{pos}_{j}^{m} \right] \right\|, \text{ pos}_{i}^{l} = Q_{i} \left(\tau_{i} \right), \text{ pos}_{j}^{m} = Q_{i} \left(\tau_{j(i)} \right), \end{cases}$$

$$\tau_{j(i)} = \begin{cases} \tau_{i} - t_{s}^{j}, \tau_{i} - t_{s}^{j} \geqslant 0, \\ 0, & \tau_{i} - t_{s}^{j} < 0, \end{cases}$$
(6)

其中, *j* ∈ ∀(*N**i*) 表示编组中其他无人机标号, *d_{ij}* 表示考虑下洗气流对无人机飞行状态的影响之后的 两架无人机之间的欧氏距离, 取 *E* = diag (1,1, λ), λ < 1. 在同一时刻下, 当 *d_{ij}* 大于阈值 *d*^{thr}_{max} 时, 认 为彼此不会发生碰撞; 当 *d_{ij}* 小于阈值 *d*^{thr}_{min} 时, 认为彼此发生碰撞概率很大, 给予一个代价为 *k_m* > 0 的惩罚; 当 *d_{ij}* ∈ [*d*^{thr}_{min}, *d*^{thr}_{max}] 时, 给予一个代价处于 [0, *k_m*] 之间的惩罚, 且惩罚代价与 *d_{ij}* 的值成反比 (*k* > 0 ∈ *C*). 此外, 假设第 *i* 架无人机局部轨迹的时间轴为全局时间轴, τ_i 表示第 *i* 架无人机当前轨迹 点所处的时刻, *tⁱ_s* 表示第 *j* 架无人机局部轨迹开始时刻, $\tau_{j(i)}$ 表示与第 *i* 架无人机对应于同一全局时 刻 τ_i 时第 *j* 架无人机在自己时间轴所处的时刻. 基于以上时间表示, 在时刻 τ_i , 无人机 *i* 所处的局部 轨迹为 *Q_i*(τ_i), 对应的轨迹点为 pos^{*i*}_{*i*}; 无人机 *j* 所处的局部轨迹为 *Q_i*($\tau_{j(i)}$), 对应的轨迹点为 pos^{*i*}_{*i*}.

2.2.4 动力学约束

为确保生成的轨迹可以被无人机执行,不超过其动力学限制,构造代价函数 q_d 对速度、加速度超过动力学范围的轨迹点进行惩罚, q_d 具体表达形式如下:

$$q_d = \begin{cases} k_{\rm dyn}, \ \|s\| > s_{\rm max}, \\ 0, \ \|s\| \leqslant s_{\rm min}, \end{cases}$$
(7)

其中, *s* 表示无人机速度、加速度状态 (s_v , s_a), s_{max} 与 s_{min} 表示无人机状态阈值 (s_{max}^v 和 s_{min}^v 分别 对应最大和最小速度值, s_{max}^a 和 s_{min}^a 分别对应最大和最小加速度值), $k_{dyn} > 0 \in C$.

2.3 最优控制问题构建

在满足无人机动态方程的前提下,通过设计最优控制问题中的代价函数,可以将未知多障碍环境 下多无人机轨迹规划问题转化为一个随机最优控制问题.具体转化过程如下:假设 *v* ~ N(0,σ²)服从 均值为 0,方差为 σ² 的正态分布,用于产生随机传播过程中的前向采样.通过前向采样固定时域为 *T* 的多组控制序列,代入状态方程中生成多条预测轨迹,并构造代价函数评估预测轨迹.此外,将多无人 机轨迹规划中各种约束条件作为代价函数的优化代价项.代价函数表示为如下形式:

$$V\left(\boldsymbol{X}_{t},t\right) = \min_{\left\{\boldsymbol{u}_{t_{0}},\dots,\boldsymbol{u}_{t_{0}+T}\right\}} \mathbb{E}_{Q}\left[\phi\left(\boldsymbol{X}_{t_{0}+T},t_{0}+T\right) + \int_{t_{0}}^{t_{0}+T} \left(q\left(\boldsymbol{X}_{t},t\right) + \frac{1}{2}\boldsymbol{u}(\boldsymbol{X}_{t},t)^{\mathrm{T}}\boldsymbol{R}\boldsymbol{u}\left(\boldsymbol{X}_{t},t\right)\right) \mathrm{d}t\right],$$

$$q\left(\boldsymbol{X}_{t},t\right) = q_{t} + q_{c} + q_{m}^{ij} + q_{d},$$
(8)

1631

其中, $\phi(\mathbf{X}_t, t)$ 定义为终端代价; $q(\mathbf{X}_t, t)$ 定义为运行代价, 由式 (4)~(7) 构成; **R** 是一个正定矩阵. 预 测轨迹的时间间隔为 $[t_0, t_0 + T]$, \mathbf{X}_t , \mathbf{U}_t 分别表示 $t \in [t_0, t_0 + T]$ 时刻系统的状态和控制输入. 此外, 定义 $\mathbb{E}_{\mathbb{Q}}[\cdot]$ 为多条预测轨迹代价值的加权期望. 则无人机轨迹规划问题可以定义为如下的固定时域最 优控制问题:

$$\arg\min_{U^*} V(\boldsymbol{X}_t, t) \quad \text{s.t. } \dot{\boldsymbol{X}} = \boldsymbol{A}\boldsymbol{X} + \boldsymbol{B}\boldsymbol{U}, \tag{9}$$

其中, $U^* = \{U^*_{t_0}, \dots, U^*_{t_0+r}\}$ 是经过轨迹评估后得到的最终的控制输入序列, 通过求解该控制输入序列, 即可实现无人机轨迹的快速生成.

3 多无人机分布式在线轨迹规划框架

本文提出的面向复杂未知多障碍环境的多无人机分布式在线轨迹规划算法的整体结构框架如 图 1 所示, 借鉴 Tube-MPC 的思想 [27], 该框架将标称控制器与辅助控制器串联来保证系统的实时 性与鲁棒性. 标称控制器忽略系统扰动, 针对理想状态求解固定时域最优控制问题, 从而实现期望轨 迹生成;辅助控制器作用于实际系统,实现无人机动作对期望轨迹的跟踪.其中,标称控制器的输入是 第 2.2 小节介绍的 3 个约束条件 (其他无人机轨迹 – 避碰约束, EDT 地图 – 避障约束, 引导轨迹 – 引 导约束) 以及当前无人机的位姿信息, 辅助控制器的输入是无人机期望轨迹. 由于标称控制器主要基 于模型预测路径积分控制算法,需要大量的并行计算,因此以 10 Hz 频率运行,保证实时性;而辅助控 制器以 50 Hz 频率运行,保证鲁棒性.关于标称控制器的具体实现,本文第4节会详细介绍;关于辅助 控制器的具体实现,请参考我们前期的工作^[28].由于采用 Gazebo 仿真环境验证多机轨迹规划算法, 无人机的状态估计信息可以通过相关插件直接获取.此外,在无人机平台上搭载深度相机,获取周围 环境信息,并采用栅格法将环境细分为等体积的正方体.结合无人机位置信息,通过射线追踪方法判断 空间中栅格占据状态,获取占据栅格地图和欧氏距离转换 (Euclidean distance transform, EDT) 地图. 占据栅格地图用于表述代表环境信息的正方体格子被障碍物占据情况,用于轨迹生成过程中安全性检 查. 若栅格被占据,则处在占据栅格中的轨迹视为无效; 若栅格空闲,则处在占据栅格中的轨迹有效; 若栅格状态未知,则视为占据. EDT 地图用于表述栅格到最近障碍物的距离,用于轨迹求解过程中最 优性考虑.根据无人机所在栅格位置距离障碍物远近,避障代价函数提供不同大小的斥力.基于我们 前期工作^[24]中的建图策略,生成上述两种地图,该策略的主要优点是可以为每一个栅格分配独立线 程进行计算, 与本文所提出的轨迹规划算法均可通过 GPU 计算平台实现并行加速. 多无人机间通讯 网络主要负责无人机轨迹的发送与接收. 当前无人机一旦生成轨迹, 就向通讯网络发送; 同时, 从通讯 网络接收其他无人机发来的轨迹信息,并将接收到的轨迹按照自己的时钟时间进行同步,用于无人机 间避碰代价计算.为减少通讯带宽,轨迹信息仅包含航点位置、相应时间、无人机标号.接下来重点介 绍本文的核心内容: 基于 MPPI 的多无人机分布式在线轨迹规划算法.

4 基于 MPPI 的多无人机异步轨迹规划方法

4.1 模型预测路径积分控制方法

本文第 2.3 小节将多无人机轨迹规划问题转化为一个随机最优控制问题.为求解该最优控制问题,将边界条件 $V(\mathbf{X}_{t_0+T}, t_0 + T) = \phi(\mathbf{X}_{t_0+T}, t_0 + T)$ 代入式 (8),构建哈密顿 – 雅可比 – 贝尔曼

1632



图 1 (网络版彩图) 多无人机分布式在线轨迹规划框架 Figure 1 (Color online) Multi-UAV decentralized online trajectory planning framework

(Hamilton-Jacobi-Bellman, HJB) 方程为

$$-\frac{\partial V}{\partial t} = q\left(\boldsymbol{X}_{t}, t\right) + \left(\boldsymbol{A}\boldsymbol{X}_{t}\right)^{\mathrm{T}} \frac{\partial V}{\partial \boldsymbol{X}_{t}} - \frac{1}{2} \left(\frac{\partial V}{\partial \boldsymbol{X}_{t}}\right)^{\mathrm{T}} \boldsymbol{B}\boldsymbol{R}^{-1} \boldsymbol{B}^{\mathrm{T}} \left(\frac{\partial V}{\partial \boldsymbol{X}_{t}}\right) + \frac{1}{2} \mathrm{tr} \left(\boldsymbol{B}\boldsymbol{B}^{\mathrm{T}} \frac{\partial^{2} V}{\partial \boldsymbol{X}_{t}^{2}}\right).$$
(10)

容易求得该偏微分方程的解为

$$\boldsymbol{U}^{*}\left(\boldsymbol{X}_{t},t\right) = -\boldsymbol{R}^{-1}\boldsymbol{B}^{\mathrm{T}}\frac{\partial V}{\partial\boldsymbol{X}_{t}}.$$
(11)

因此,为得到控制输入 $U^*(X_t,t)$,需要求解代价函数关于无人机状态的偏微分方程 (partial differential equation, PDE). 传统方法求解上述偏微分方程会受到系统状态维度影响,而四旋翼无人机包含多个状态变量,因此采用传统方法难以求解上述方程.本文采用路径积分控制 (path integral control, PIC) 方法将其转化为求解多条预测轨迹的期望.为采用费曼 – 卡茨 (Feynman-Kac) 方程建立 PDE 与 PIC 之间的联系,需要将等式 (10) 简化为如下形式的线性偏微分方程 (linear partial differential equation, LPDE):

$$\frac{\partial \Psi}{\partial t} = \frac{\Psi}{\lambda} q \left(\boldsymbol{X}_{t}, t \right) - \left(\boldsymbol{A} \boldsymbol{X}_{t} \right)^{\mathrm{T}} \frac{\partial \Psi}{\partial \boldsymbol{X}_{t}} - \frac{1}{2} \mathrm{tr} \left(\boldsymbol{B} \boldsymbol{B}^{\mathrm{T}} \frac{\partial^{2} \Psi}{\partial \boldsymbol{X}_{t}^{2}} \right),$$

$$\Psi \left(\boldsymbol{X}_{t}, t \right) = \mathrm{e}^{-\frac{V\left(\boldsymbol{X}_{t}, t \right)}{\lambda}},$$
(12)

其中, $\lambda > 0, \lambda \in C$. 具体过程可参考文献 [13]. 该 LPDE 的解为

$$\Psi\left(\boldsymbol{X}_{t},t\right) = \mathbb{E}_{\mathbb{P}}\left[\exp\left(-\frac{1}{\lambda}\int_{t}^{t_{0}+T}q\left(\boldsymbol{X}_{\tau},\tau\right)\mathrm{d}\tau\right)\Psi\left(\boldsymbol{X}_{t_{0}+T},t_{0}+T\right)\right],\tag{13}$$

其中, ℝ_P[·] 表示对不受控系统的多条预测轨迹的代价函数值的期望.

1633

将
$$\Psi(\mathbf{X}_{t_0+T}, t_0 + T) = e^{-\frac{V(\mathbf{X}_{t_0+T}, t_0+T)}{\lambda}} = e^{-\frac{\phi(\mathbf{X}_{t_0+T}, t_0+T)}{\lambda}}$$
代入式 (13) 可得
 $\Psi(\mathbf{X}_t, t) \approx \mathbb{E}_{\mathbb{P}}\left[\exp\left(-\frac{S(t)}{\lambda}\right)\right],$
 $S(t) = \phi(\mathbf{X}_{t_0+T}, t_0 + T) + \int_{t_0}^{t_0+T} q(\mathbf{X}_{\tau}, \tau) d\tau.$
(14)

上式为代价函数 (式 (8)) 提供了路径积分表达形式.为求代价函数的最优解,传统方法需要通过式 (11) 计算其对状态 **X** 的梯度,易受到维度灾难影响,而基于 MPPI 的轨迹规划方法避免对状态 **X** 解析梯度的计算,具体过程如下.

首先将 $\Psi(X_t, t)$ 对 X_{t_0} 求解析梯度可得连续时间下路径积分的最优解的表达形式:

$$\boldsymbol{u}^* dt = \boldsymbol{R}^{-1} \boldsymbol{B}^{\mathrm{T}} \left(\boldsymbol{B} \boldsymbol{R}^{-1} \boldsymbol{B}^{\mathrm{T}} \right)^{-1} \frac{\mathbb{E}_{\mathbb{P}} [\exp(-\frac{S(t)\boldsymbol{B}}{\lambda} d\boldsymbol{v})]}{\mathbb{E}_{\mathbb{P}} [\exp(-\frac{S(t)}{\lambda})]}.$$
(15)

其次,为方便前向采样过程,对无人机状态方程离散化处理可得

$$\mathbf{X}_{t} = \mathbf{A}_{d} \mathbf{X}_{t-1} + \mathbf{B}_{d} \mathbf{U}_{t-1},
\mathbf{A}_{d} = \begin{bmatrix} \mathbf{I}_{3 \times 3} & \mathbf{I}_{3 \times 3} \Delta t & \mathbf{I}_{3 \times 3} \frac{\Delta t^{2}}{2} & \mathbf{0}_{3 \times 1} \\ \mathbf{0}_{3 \times 3} & \mathbf{I}_{3 \times 3} & \mathbf{I}_{3 \times 3} \Delta t & \mathbf{0}_{3 \times 1} \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 1} \\ \mathbf{0}_{1 \times 3} & \mathbf{0}_{1 \times 3} & \mathbf{0}_{1 \times 3} & 1 \end{bmatrix}, \quad \mathbf{B}_{d} = \begin{bmatrix} \mathbf{I}_{3 \times 3} \frac{\Delta t^{3}}{6} & \mathbf{0}_{3 \times 1} \\ \mathbf{I}_{3 \times 3} \frac{\Delta t^{2}}{2} & \mathbf{0}_{3 \times 1} \\ \mathbf{I}_{3 \times 3} \Delta t & \mathbf{0}_{3 \times 1} \\ \mathbf{0}_{1 \times 3} & \Delta t \end{bmatrix}.$$
(16)

最终得到离散化形式的最优控制输入为

$$\boldsymbol{u}_{t}^{*} = \boldsymbol{u}_{t-1} \frac{\mathbb{E}[\exp(-\frac{S(t_{0}+T)}{\lambda}\boldsymbol{v}_{t})]}{\mathbb{E}[\exp(-\frac{S(t_{0}+T)}{\lambda})]},$$

$$S(t_{0}+T) = \phi(\boldsymbol{X}_{t_{0}+T}, t_{0}+T) + \sum_{k=1}^{K} q(\boldsymbol{X}_{\tau}, \tau) \Delta t,$$
(17)

其中, K 表示蒙特卡洛前向采样数量.

4.2 多无人机异步轨迹规划方法

路径积分控制方法通过将大量采样轨迹的代价求期望, 解决了偏微分方程求解过程中维数灾难问题 (式 (14)), 而该期望可通过蒙特卡洛前向采样近似求解 (式 (17)). 模型预测控制方法通过构建代价 函数评价采样, 从而计算出相应时刻的最优控制输入 *u*^{*}. MPPI 结合上述两种方法, 基于模型预测结构, 根据路径积分控制方法更新每一步预测迭代过程.本小节基于 MPPI 框架, 详细介绍本文提出的 多无人机异步轨迹规划方法.

以第 *i* 架无人机为例,介绍其轨迹生成方法. 首先,对于随机控制输入 *v*,定义 v_k^{ρ} 为第 *k* 次 采样过程中第 $\rho \in \mathbb{Z}$ 步对应的随机输入向量. 其次,对应于第 *k* 次采样,给定初始控制输入序列: $U_{init}^k = [u_{t_0}^k, u_{t_0+\Delta t}^k, \dots, u_{t_0+T}^k]$ (初始化为零向量),将 $U = u_{t_0+\rho\Delta t}^k + v_k^{\rho}, \rho \in [0,T]$ 代入无人机状态方 程中前向迭代 *T* 步,即可得到第 *k* 条采样轨迹. 在第 ρ 步,相应的代价函数值为 *S*($t_{0+\rho\Delta t}$),将 *T* 步 代价值相加得到第 *k* 条采样轨迹的代价. 在实际运算时,基于 GPU 多核特性,多条采样轨迹并行前 向传播 *T* 步. 然后,通过比较 *K* 条轨迹的代价,得到最小代价值,并通过式 (17) 计算权重,得到最优 控制序列;并结合 MPC 框架,完成控制序列的迭代更新.最后,将最优控制序列代入无人机状态方程 中,得到无人机轨迹. 整体基于 MPPI 的多无人机异步轨迹规划方法的伪代码如算法 1 所示.

算法 1 The asynchronous multi-UAV trajectory planning method based on MPPI

Input: $U_{\text{init}}^{k \in [0,K]}$: initial control sequence; K: Monte-Carlo sampling times; T: number of propagate steps; Δt : time interval;

```
Main iteration: Q_i^m = f(U_{\text{init}}^{k \in [0,K]}, K, T, \Delta t);
  1: for k = 0 to K - 1 do
             V_k = \{\boldsymbol{v}_k^0, \boldsymbol{v}_k^1, \dots, \boldsymbol{v}_k^T\};
  2:
             for \rho = 0 to T do
  3:
                  if d_{\text{obs}} < d_{\min}^{\text{obs}} \parallel d_{ij} < d_{\min}^{\text{thr}} then
  4:
  5:
                        \boldsymbol{X}_{t_0+(\rho+1)\Delta t} = \boldsymbol{X}_{t_0+\rho\Delta t};
  6:
                  else
                        \boldsymbol{X}_{t_0+(\rho+1)\Delta t} = \boldsymbol{A}_d \boldsymbol{X}_{t_0+\rho\Delta t} + \boldsymbol{B}_d \left( \boldsymbol{u}_{t_0+\rho\Delta t} + \boldsymbol{v}_k^\rho \right);
  7:
  8:
                  end if
  9:
                  S_k + = S(t_0 + (\rho + 1)\Delta t);
10:
             end for
11: end for
12: S_{\min} = \min[S_0, S_1, \dots, S_{K-1}];
13: \xi = \sum_{k=0}^{K-1} \exp\left(-\frac{1}{\lambda} \left(S_k - S_{\min}\right)\right);
14: for k = 0 to K - 1 do
15: \varpi_k = \frac{1}{\xi} \exp\left(-\frac{1}{\lambda}\left(S_k - S_{\min}\right)\right);
16: end for
17: for \rho = 0 to T do
             \boldsymbol{u}_{t_0+\rho\Delta t}+=\sum_{k=0}^{K-1}\varpi_k \boldsymbol{v}_k^{\rho};
18:
19: end for
20: \boldsymbol{u}_{t}^{*} = \boldsymbol{u}_{t}, t \in [0, T\Delta t];
21: for \rho = 0 to T do
22:
           \boldsymbol{u}_t = \boldsymbol{u}_{t+\Delta t};
23: end for
24: U^* = [\boldsymbol{u}_{t_0}, \boldsymbol{u}_{t_0+\Delta t}, \dots, \boldsymbol{u}_{t_0+T\Delta t}],
25: for \rho = 1 to T - 1 do
26:
             \boldsymbol{X}_{t_0+(\rho+1)\Delta t} = \boldsymbol{A}_d \boldsymbol{X}_{t_0+\rho\Delta t} + \boldsymbol{B}_d \boldsymbol{u}_{t_0+\rho\Delta t};
             \operatorname{pos}_{i}^{\rho+1} = \boldsymbol{X}_{t_0+(\rho+1)\Delta t};
27:
28: end for
29: Q_i^m = \operatorname{connect} \left[ s_i, \operatorname{pos}_i^1, \operatorname{pos}_i^2, \dots, \operatorname{pos}_i^T \right];
Output: Q_i^m: the optimal trajectory of the ith UAV.
```

在通过代价函数评价每一步的采样轨迹时,需要考虑引导约束、避障约束、避碰约束、无人机动 力学约束等.引导约束对无人机轨迹提供引力场,期望实际轨迹尽量接近引导轨迹 (式 (4)),避免其陷 入局部极小值;避碰、避障约束 (式 (5) 和 (6))对无人机轨迹提供斥力场,期望其尽量远离障碍物和 其他无人机,确保飞行的安全性.由于斥力场是软约束,并不能完全确保轨迹安全,因此对于距离障碍 (*d*obs < *d*^{obs})或者其他无人机较近 (*dij* < *d*^{thr}_{min})的轨迹点,给予一个较大的惩罚并停止当前轨迹在当 前步的前向传播,以上一步控制输入作为当前步控制输入;对于距离障碍或者其他无人机较远的轨迹 点,给予一个与距离成反比的适当惩罚后前向传播.无人机动力学约束惩罚速度、加速度绝对值超过 阈值的轨迹点 (式 (7)).此外,每一架无人机都有各自的时钟,作为当前轨迹参考时间.无人机之间考 虑避碰时,以当前无人机时钟时间作为参考 (全局时间),把其他无人机的时钟时间转换到当前无人机 时钟系下.上述多无人机异步轨迹规划方法如图 2 所示.



图 2 (网络版彩图) 多无人机异步轨迹规划方法 Figure 2 (Color online) Asynchronous multi-UAV trajectory planning method

表 2 8 架无人机起始与目标位置

Table 2 Starting and target positions of eight UAVs

	UAV1	UAV2	UAV3	UAV4	UAV5	UAV6	UAV7	UAV8
Starting	(-20, 0, 1.5)	(20, 0, 1.5)	(0, -20, 1.5)	(0, 20, 1.5)	(-20, -20, 1.5)	(20, 20, 1.5)	(-20, 20, 1.5)	(20, -20, 1.5)
Target	(20, 0, 1.5)	(-20, 0, 1.5)	(0, 20, 1.5)	(0, -20, 1.5)	(20, 20, 1.5)	(-20, -20, 1.5)	(20, -20, 1.5)	(-20, 20, 1.5)

5 **仿真验证**

本文算法实现基于机器人操作系统 (robot operating system, ROS), 采用 C++ 语言编程, 并运 行于 CPU 为 Intel i9-10900K, GPU 为 NVIDIA GeForce RTX 3090 的 Ubuntu18.04 系统. 本文采用 Gazebo 仿真平台, 搭建了具有复杂障碍物的场景, 使用带有深度相机的四旋翼无人机模型验证本文所 提算法的有效性. 本节包括多无人机避碰测试与多无人机避碰避障测试两个部分. 测试过程中, 选择 蒙特卡洛采样次数 K = 2560, 步长 T = 35, 步间距 $\Delta t = 0.1$ s.

5.1 多无人机避碰测试

在无障碍场景下,采用 8 架无人机点对点飞行方法测试本文所提算法性能,并与 ORCA 方法^[15] 在飞行距离 (*d*_{fy})、飞行时间 (*t*_{fy})、轨迹求解时间 (*t*_{sol})、安全性方面进行比较.测试过程中,无人机 最大速度均设为 2 m/s (*s^v*_{max} = 2),最大加速度设为 1 m/s² (*s^a*_{max} = 1),无人机间发生相互避碰行为的 距离阈值均设为 1 m (*d^{thr}*_{max} = 1,*d^{thr}*_{min} = 0.6). 8 架无人机的起始与目标位置如表 2 所示. 仿真测试结 果如表 3 所示,表中数据均为平均值.其中安全性指标采用相互碰撞概率 (*p*_{col}) 表示,定义为测试过程 中发生碰撞的无人机个数与无人机总数的百分比.从表 3 数据可得,所提出的算法在飞行距离、飞行 时间与安全性方面均优于 ORCA;虽然轨迹求解效率低于 ORCA,但是每架无人机平均求解时间优于 8 ms,满足实时性要求.仿真过程中无人机轨迹如图 3 所示.从图 3 可得,本文所提算法求解得到的 无人机轨迹更加平滑.

			Ĩ		
Method	d_{fly} (m)	$t_{\rm fly}~({ m s})$	$t_{\rm sol}~({\rm ms})$	$p_{ m col}$ (%)	
ORCA	53.40	62.60	0.027	43.75	
Proposed	49.06	39.59	7.98	0	

表 3 多无人机避碰测试结果

 Table 3
 Collision avoidance test results for multiple UAVs



图 3 (网络版彩图) 多无人机点对点飞行轨迹 Figure 3 (Color online) Point-to-point flight trajectories of multiple UAVs



图 4 (网络版彩图) 多障碍场景 Figure 4 (Color online) Multi-obstacle scenes

表 4 多障碍环境下所提算法性能测试

Table 4	Performance of	the proposed	algorithm in	obstacle-rich	environments
---------	----------------	--------------	--------------	---------------	--------------

	d_{fly} (m)	$t_{\rm fly}$ (s)	$t_{\rm sol} \ ({\rm ms})$	$d_{\rm safe}$ (m)	$p_{\rm col}~(\%)$
Scene I	109.23	84.96	5.17	0.69	0
Scene II	52.90	46.68	6.27	0.73	0

5.2 多无人机避碰避撞测试

为测试基于 MPPI 的多无人机异步轨迹规划方法在多障碍环境下的性能,本文基于 Gazebo 仿真



图 5 (网络版彩图) 场景 I 全局点云地图与无人机飞行轨迹 Figure 5 (Color online) Global point cloud map of scene I and flight trajectories of UAVs



图 6 (网络版彩图) 场景 II 全局点云地图与无人机飞行轨迹 Figure 6 (Color online) Global point cloud map of scene II and flight trajectories of UAVs

平台搭建 2 个场景, 如图 4 所示. 基于场景 I, 进行 4 架无人机长距离飞行实验. 从 UAV1 到 UAV4, 选择 4 架无人机的起点分别为 (-50, -9, 1.5), (-50, -3, 1.5), (-50, 3, 1.5), (-50, 9, 1.5); 终点分别为 (50, 9, 1.5), (50, 3, 1.5), (50, -3, 1.5), (50, -9, 1.5). 基于场景 II, 进行 8 架无人机点对点飞行实验, 无人 机起点与目标点位置如表 2 所示. 测试过程中, 无人机最大速度设为 2 m/s ($s_{max}^a = 1$), 最大加速度设 为 1 m/s² ($s_{max}^a = 1$), 避碰约束的最大和最小距离阈值分别为 $d_{max}^{thr} = 1$ 和 $d_{min}^{thr} = 0.6$, 避障约束的最 大和最小距离阈值分别为 $d_{max}^{thr} = 1.5$ 和 $d_{min}^{obs} = 0.8$. 测试结果如表 4 所示, 表中数据均为多次测试中



Figure 7 (Color online) Desired and actual trajectories of UAV5 in scene II

多架无人机的平均值. 从表 4 可以得出, 在场景 I 和 II 飞行过程中, 无人机平均速度分别约为 1.29 和 1.133 m/s; 与最近障碍物的距离 (*d*_{safe}) 约为 0.7 m 且无人机之间没有发生相互碰撞, 符合安全性要求; 轨迹求解时间均小于 7 ms, 满足规划实时性要求. 在多机规划过程中, 每一架无人机实时建图且没有 共享彼此地图信息. 为了方便可视化, 将多架无人机地图手动拼接在一起形成了全局地图并删除了地 面点. 图 5 和 6 分别显示了场景 I 和 II 的全局点云地图和无人机飞行轨迹. 测试过程中所有无人机 均采用相同的辅助控制器, 以场景 II 中 UAV5 点对点飞行为例, 其运行轨迹和期望轨迹如图 7 所示. 从图 7 可见, 无人机实际运行轨迹能够很好地跟踪期望轨迹. 更多实验细节参见网址¹⁾.

6 总结与展望

以提高复杂未知多障碍环境下多无人机在线轨迹生成的安全性和灵活性为目的,本文提出了基于 Tube-MPC 和 MPPI 融合的多无人机分布式在线轨迹规划框架与方法.该方法通过并行蒙特卡洛随 机采样技术,将多无人机轨迹优化问题的求解转化为对大量采样轨迹的期望求解,其显著特点是无人 机在线轨迹求解过程中不依赖于梯度信息,有效避免了传统方法对代价函数和约束条件的连续性及凸 性要求.仿真验证表明本文提出的算法,在保证多无人机飞行安全(避碰避障)的前提下,多无人机的 在线轨迹生成时间优于 8 ms.未来我们尝试将本文所提算法移植到自主搭建的四旋翼无人机平台,在 树林或楼宇等多障碍场景中予以验证.

参考文献 -

¹ Augugliaro F, Lupashin S, Hamer M, et al. The flight assembled architecture installation: cooperative construction with flying machines. IEEE Control Syst Mag, 2014, 34: 46–64

² Spurný V, Báča T, Saska M, et al. Cooperative autonomous search, grasping, and delivering in a treasure hunt scenario by a team of unmanned aerial vehicles. J Field Robotics, 2019, 36: 125–148

³ Saikin D A, Baca T, Gurtner M, et al. Wildfire fighting by unmanned aerial system exploiting its time-varying mass. IEEE Robot Autom Lett, 2020, 5: 2674–2681

¹⁾ https://www.bilibili.com/video/BV1PP4y1N7Gw.

- 4 Schouwenaars T, de Moor B, Feron E, et al. Mixed integer programming for multi-vehicle path planning. In: Proceedings of European Control Conference, 2001. 2603–2608
- 5 Mellinger D, Kushleyev A, Kumar V. Mixed-integer quadratic program trajectory generation for heterogeneous quadrotor teams. In: Proceedings of IEEE International Conference on Robotics and Automation, Bielefeld, 2012. 477–483
- 6 Tang S, Thomas J, Kumar V. Hold or take optimal plan (HOOP): a quadratic programming approach to multi-robot trajectory generation. Int J Robotics Res, 2018, 37: 1062–1084
- 7 Augugliaro F, Schoellig A P, D'Andrea R. Generation of collision-free trajectories for a quadrocopter fleet: a sequential convex programming approach. In: Proceedings of International Conference on Intelligent Robots and Systems, Algarve, 2012. 1917–1922
- 8 Chen Y, Cutler M, How J P. Decoupled multiagent path planning via incremental sequential convex programming.
 In: Proceedings of IEEE International Conference on Robotics and Automation, Washington, 2015. 5954–5961
- 9 Robinson D R, Mar R T, Estabridis K, et al. An efficient algorithm for optimal trajectory generation for heterogeneous multi-agent systems in non-convex environments. IEEE Robot Autom Lett, 2018, 3: 1215–1222
- 10 Preiss J A, Honig W, Ayanian N, et al. Downwash-aware trajectory planning for large quadrotor teams. In: Proceedings of International Conference on Intelligent Robots and Systems, Vancouver, 2017. 250–257
- 11 Honig W, Preiss J A, Kumar T K S, et al. Trajectory planning for quadrotor swarms. IEEE Trans Robot, 2018, 34: 856–869
- 12 Hamer M, Widmer L, D'andrea R. Fast generation of collision-free trajectories for robot swarms using GPU acceleration. IEEE Access, 2018, 7: 6679–6690
- 13 Williams G, Aldrich A, Theodorou E A. Model predictive path integral control: from theory to parallel computation. J Guidance Control Dyn, 2017, 40: 344–357
- 14 Fiorini P, Shiller Z. Motion planning in dynamic environments using velocity obstacles. Int J Robotics Res, 1998, 17: 760–772
- 15 Berg J, Guy S J, Lin M, et al. Reciprocal n-body collision avoidance. In: Robotics Research. Berlin: Springer, 2011. 3–19
- 16 Arul S H, Manocha D. DCAD: decentralized collision avoidance with dynamics constraints for agile quadrotor swarms. IEEE Robot Autom Lett, 2020, 5: 1191–1198
- 17 Zhou D, Wang Z, Bandyopadhyay S, et al. Fast, on-line collision avoidance for dynamic vehicles using buffered Voronoi cells. IEEE Robot Autom Lett, 2017, 2: 1047–1054
- 18 Wang M, Schwager M. Distributed collision avoidance of multiple robots with probabilistic buffered Voronoi cells.
 In: Proceedings of International Symposium on Multi-Robot and Multi-Agent Systems (MRS), 2019. 169–175
- 19 Zhu H, Alonso-Mora J. B-UAVC: buffered uncertainty-aware Voronoi cells for probabilistic multirobot collision avoidance. In: Proceedings of International Symposium on Multi-Robot and Multi-Agent Systems (MRS), 2019. 162–168
- 20 Luis C E, Schoellig A P. Trajectory generation for multiagent point-to-point transitions via distributed model predictive control. IEEE Robot Autom Lett, 2019, 4: 375–382
- 21 Luis C E, Vukosavljev M, Schoellig A P. Online trajectory generation with distributed model predictive control for multi-robot motion planning. IEEE Robot Autom Lett, 2020, 5: 604–611
- 22 Zhang X, Shen H, Xie G, et al. Decentralized motion planning for multi quadrotor with obstacle and collision avoidance. Int J Intell Robot Appl, 2021, 5: 176–185
- 23 Lu H, Zong Q, Lai S, et al. Real-time perception-limited motion planning using sampling-based MPC. IEEE Trans Ind Electron, 2022, 69: 13182–13191
- 24 Lu H, Zong Q, Lai S, et al. Flight with limited field of view: a parallel and gradient-free strategy for micro aerial vehicle. IEEE Trans Ind Electron, 2022, 69: 9258–9267
- 25 Mellinger D, Kumar V. Minimum snap trajectory generation and control for quadrotors. In: Proceedings of IEEE International Conference on Robotics and Automation, Shanghai, 2011. 2520–2525
- 26 Dolgov D, Thrun S, Montemerlo M, et al. Path planning for autonomous vehicles in unknown semi-structured environments. Int J Robotics Res, 2010, 29: 485–501
- 27 Yu S, Maier C, Chen H, et al. Tube MPC scheme based on robust control invariant set with application to Lipschitz nonlinear systems. Syst Control Lett, 2013, 62: 194–200
- 28 Shen H, Zhang X, Lu H, et al. State estimation and control for micro aerial vehicles in GPS-denied environments.

In: Proceedings of International Conference on Wireless Communications and Signal Processing (WCSP), Wuhan, 2020. 1070–1075

Multi-UAV decentralized online trajectory planning in complex unknown obstacle-rich environments

Xuewei ZHANG¹, Bailing TIAN^{1*}, Hanchen LU², Hongming SHEN¹ & Qun ZONG¹

1. School of Electrical and Information Engineering, Tianjin University, Tianjin 300072, China;

2. Science and Technology on Complex System Control and Intelligent Agent Cooperation Laboratory, Beijing Electro-Mechanical Engineering Institute, Beijing 100074, China

* Corresponding author. E-mail: bailing_tian@tju.edu.cn

Abstract In consideration of the effect of complex unknown obstacle-rich environments on the performance of the real-time trajectory planning of unmanned aerial vehicles (UAVs), this paper proposes a framework and method for multi-UAV decentralized real-time trajectory planning based on Tube-MPC and model predictive path integral (MPPI) control. First, the cost function representing constraints is formulated by taking into account the requirements of the reciprocal collision avoidance and the obstacle avoidance in an obstacle-rich scenario such that the trajectory planning problem is transformed into a stochastic optimal control problem. Furthermore, a decentralized trajectory planning framework for multiple UAVs is designed and established based on Tube-MPC, wherein the low-frequency nominal controller and the high-frequency auxiliary controller are connected in series to ensure the real-time characteristic and robustness of the system. In addition, an asynchronous multi-UAV trajectory planning method based on MPPI for avoiding dimensional disaster in solution through traditional methods is presented. Through the parallel Monte-Carlo random forward sampling technology based on GPU, the solution of the stochastic optimal control problem for multiple UAVs is transformed into the expectation of sampling trajectory under a given cost function. Then, the optimal control sequence is obtained. Its notable feature is quick solution and avoidance of the requirements of continuity and convex properties of the constraint condition and the cost function that are indispensable for the gradient-based method. Finally, the effectiveness of the algorithm in a complex, unknown, and obstacle-rich environment is verified using the Gazebo virtual simulation platform.

Keywords unknown obstacle-rich environment, multi-UAV, online trajectory planning, model predictive path integral, GPU parallel acceleration