arXiv:2302.02293v1 [cs.RO] 5 Feb 2023

利用配备有限视场传感器的无人机快速进行未知环境建图的自主探索方法

赵英豪、严力，*IEEE* 会员，谢洪、戴继成和魏鹏程

摘要—**自主探索是实现快速自主建图和目标搜索的重要环节之一。然而，现有的大多数方法都面临效率低下的问题，这主要是由轨迹质量不佳或往返机动造成的。为提高未知环境中的探索效率，本文提出了一种快速自主探索规划器（FAEP）。与现有方法不同，我们首先设计了一种新颖的前沿探索序列生成方法，以获得更合理的探索路径；该方法在非对称旅行商问题（ATSP）中不仅考虑飞行层级因素，还考虑前沿层级因素。随后，根据探索序列和前沿的分布，我们提出了一种自适应偏航规划方法，通过在探索行程中改变偏航角覆盖更多前沿。此外，为了提高飞行速度和平稳性，我们还采用了动态重规划策略。我们在仿真环境中进行了充分的对比和评估实验。实验结果表明，与典型方法和最先进方法相比，所提出的探索规划器在飞行时间和飞行距离方面表现更佳。此外，我们还在真实环境中进一步评估了所提出方法的有效性。**

索引词—**自主探索、建图、探索序列、自适应偏航规划**

I 引言

我NTELLIGENT 机器人一直是许多领域追求的目标，经过众多研究人员的努力，已经出现了许多应用案例 [1], [2]。近年来，无人机（UAV）凭借其独特优势，作为数据采集和物资运输平台，已广泛应用于测绘 [3], 环境保护 [4],救援 [5], 军事 [6], 等领域[7] 。然而，在大多数作业场景中，无人机仍处于人工操作状态，尚缺乏自主操作能力。作为无人机自主建图能力的关键组成部分之一，自主探索已引起广泛关注，并涌现出许多优秀的自主探索算法 [8], [9]。

稿件收稿日期：2xxx年xx月；修订日期：xxxx年xx月；录用日期：xxxx年x月。本工作得到中国国家重点研发计划（项目编号：2020YFD1100200）和湖北省科技重大专项（项目编号：2021AAA010）的资助。（通信作者：Li Yan和谢洪）所有作者均来自中国湖北省武汉市武汉大学测绘学院，邮编430079（电子邮箱：zhaoying-hao@whu.edu.cn；lyan@sgg.whu.edu.cn；hxie@sgg.whu.edu.cn；daijicheng@whu.edu.cn；wei.pc@whu.edu.cn）。

A. 现有问题

尽管现有探索方法可以利用前沿或采样视点探索环境，但仍有许多问题亟待解决。采样视点的方法能够轻松生成候选目标，但总是会导致较低的探索率和探索效率 [8]。而且，其中大多数采用贪心策略，只关注局部信息增益，却忽略了全局效率。文献 [10] 采用基于前沿的策略引导“下一最佳视点”规划，从而提高了效率。然而，这些方法没有考虑无人机的动力学特性，会导致探索轨迹不平滑、低速飞行以及大量走走停停机动。尽管采用前沿的方法 [9]，[11] 可以通过搜索前沿并生成探索序列来快速探索整个环境，但寻找和描述前沿的过程通常计算代价高昂。FUEL [12] 是一种先进的快速自主探索算法。其启发式框架通过设计前沿信息结构（FIS）和分层规划，实现了无人机快速、高效的探索，并且能够以高频率生成平滑、高速的探索轨迹。然而，尽管与其他算法相比，该算法已显著提高了探索率和探索效率，但仍面临一些影响其探索效率的问题，例如探索过程中的往返机动。

B. 主要贡献

为减少第 I-A 节中提到的会导致探索效率低下的往返机动，本文基于 FUEL 的框架，提出了一种新型快速自主探索规划器（FAEP）。为提高探索轨迹的合理性和探索效率，我们针对全局巡回规划设计了一种综合探索序列生成方法。该方法不仅考虑飞行层级因素，还创新性地考虑前沿层级因素（前沿的空间特征），以减少往返探索机动。根据空间特征，我们评估前沿导致往返机动的潜在可能性，并基于该可能性获得更合理的探索序列。在确定局部探索目标后，一种自适应偏航

规划策略被设计用于通过飞行过程中的偏航变化实现高效探索，其中包含两种偏航规划模式：常规规划模式和两阶段规划模式。根据前沿的分布及其自身的运动状态，可以选择适当的模式以覆盖更多区域。此外，为提高飞行的稳定性和速度，还采用了动态重规划。

我们在不同的仿真环境中，将本方法与三种典型且先进的方法进行了比较。实验结果表明，本方法的性能优于其他三种方法。与两种典型方法 NBVP 和 Aeplanner 相比，本方法的探索过程快 3–7 倍。与先进方法 FUEL 相比，本方法将飞行时间和飞行距离减少了 20% 以上。此外，我们还通过机载真实环境探索验证了本方法的有效性。本文的贡献总结如下：

1）一种全面的前沿探索序列生成方法，不仅考虑飞行层面的因素，还考虑前沿层面的因素，从而生成更合理的全局路径。我们创新性地考虑前沿的空间位置及其导致低效率机动的可能性。可能导致往返机动的区域将被优先探索，以提高效率。2）一种自适应偏航规划策略，包含两种偏航规划模式：常规规划模式和两阶段规划模式。在飞往局部目标时，系统将根据不同模式生成平滑的偏航路径，以覆盖前沿。具体而言，两阶段规划模式会在合理的时机通过偏航变化来覆盖更多区域或特殊的小区域。

3）在仿真中进行了充分的定量对比实验。此外，还在各种环境中开展了真实世界实验，以验证我们的方法。有关实验的更多详情，请访问 https: // youtu.be/0Y671mEwJ_A。

II. 相关工作

近年来，许多学者对自主探索问题进行了研究，并从多个角度提出了大量方法，主要分为以下三类：基于采样的探索、基于前沿的探索，以及近年来兴起的基于机器学习的算法。本文仅讨论前两类算法，它们已广泛应用于各种探索任务。

A. 基于采样的探索方法

基于采样的探索方法在自由空间中随机采样视点，并通过获取信息增益最高的路径来寻找下一个最佳视点 [13]。文献 [8] 提出了一种滚动时域“下一最佳视点”规划（NBVP），通过考虑整条路径所获得的信息来探索三维环境。NBVP 是首个以滚动时域方式将下一最佳视点概念用于探索的方法，许多方法

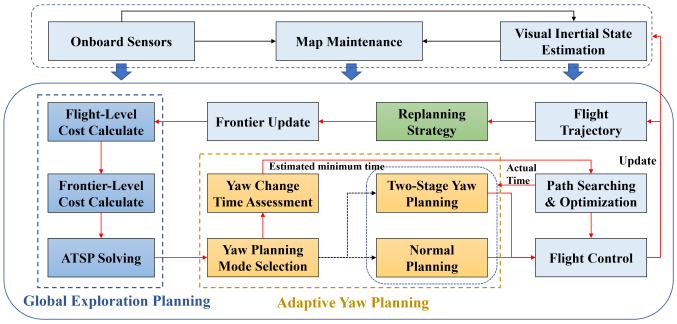


图 1. 所提出的快速自主探索规划器概览。规划器的主要流程以红线表示。我们的主要贡献以深色（蓝色、黄色和绿色）标出。

均由此方法衍生。这些方法在增量式快速探索随机树（RRT）中选择信息增益最高的路径，供无人机执行。文献 [10] 中的自主探索规划器（Aeplanner）结合了前沿探索与 NBVP，以避免在大型环境中陷入停滞、无法探索所有区域；该方法还利用前几轮迭代中的缓存点，加快了潜在信息增益的估算过程。文献 [14] 使用增量采样和概率路线图来提高规划效率。方法 [15] 将采样方法与基于前沿的方法相结合，以降低在大型场景中寻找未探索区域所带来的影响。还有一些两阶段方法 [16], [17]，通过在全局地图和局部地图中采用不同的规划策略，高效覆盖整个环境。

B. 基于前沿的探索方法

相比之下，基于前沿的方法主要由两个过程组成：寻找前沿（已知区域与未知区域之间的边界），以及求解访问各前沿的全局路径序列问题。 [18] 提出了首个用于探索通用二维环境的基于前沿的探索方法，该方法选择最近的前沿作为下一个目标。随后， [19] 提出了一种基于随机微分方程的探索算法，用于实现三维环境中的探索。为实现高速飞行， [11] 的作者提出了一种在视场（FOV）内提取前沿，并选择使速度变化最小的前沿的方法。为寻找合理的前沿探索序列， [9]采用了旅行商问题（TSP）。 [20]通过采用信息驱动探索策略来选择合适的探索目标。然而，许多方法都面临全局覆盖效率低、飞行轨迹保守以及决策频率低等问题。为解决这些问题， [12] 的作者采用增量前沿结构和分层规划，实现了快速探索。

III. 所提出的方法

为提高探索建图的效率，我们提出了一种快速自主探索规划器。该规划器的主要操作流程如图 1 所示。首先，我们根据

无人机状态和传感器数据进行地图维护并更新前沿。其次，综合考虑飞行层级因素和前沿层级因素，进行全局探索规划。第三，根据全局规划结果，执行路径规划和包含两种规划模式的自适应偏航规划，以生成高质量的局部路径。最后，我们将规划结果提交给飞行控制系统执行，并根据重规划策略持续执行上述流程。

A. 前沿探索序列生成

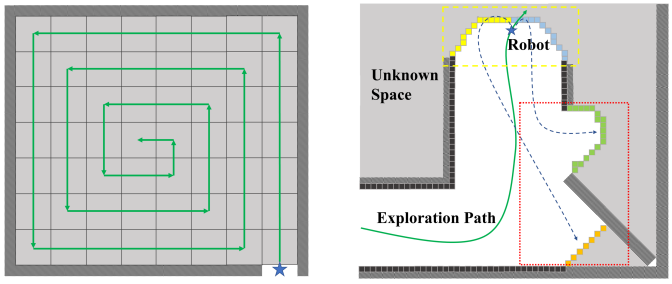
前沿探索序列对于基于前沿的探索方法至关重要，其合理性直接影响整个探索过程的效率。许多方法使用旅行商问题来获取探索序列。然而，大多数方法仅将前沿之间的欧几里得距离作为旅行商问题的代价，虽然这种做法简单，却不够充分。尽管有些方法在欧几里得距离的基础上考虑偏航变化和速度方向等因素，以提高探索序列的质量，但仍存在一些不足。与这些方法不同，本文首先不采用传统的旅行商问题，而是使用更合理的非对称旅行商问题（ATSP）进行求解。其次，本文不仅将飞行层级因素（前沿与无人机之间的欧几里得距离、偏航变化和速度方向变化）作为代价，还将前沿层级因素纳入代价函数，通过求解 ATSP 生成更优的全局探索序列。

根据完全覆盖路径规划（CCPP）问题的评价标准 [21], ，往返机动越少，效率越高。因此，当需要快速探索未知区域时，使用螺旋填充算法 [22] 是最佳策略之一。如图 2(a) 所示，螺旋算法在实现高覆盖率的同时，路径重叠较少。此外，为避免不必要的往返机动，不应遗漏已经经过的较小未知区域。否则，如图 2(b) 所示，会产生大量需要机动绕行的区域。受此启发，本文认为，当某个前沿接近探索区域的边界或独立小区域时，应提高其相应的探索优先级。如果不优先探索该区域，就会导致往返机动，降低全局探索效率。因此，不同于仅考虑当前飞行高度因素的其他算法，本方法还将前沿的空间特征纳入考量，通过优先处理边界区域和独立小区域，生成更合理的前沿探索序列；其中，边界是指任务所确定的探索区域边缘。

为实现边界区域优先级，我们为 FIS 中的每个前沿计算边界代价 $c_b(k)$:

$$c_b(k) = \begin{cases} d_{min}(k) & D_k < r_s \\ d_{min}(k) \cdot \left(1 + w_d \cdot \dfrac{D_k - r_s}{r_s}\right) & D_k \geq r_s \end{cases} \quad (1)$$

$$d_{min}(k) = min(d_x^k, d_y^k, d_z^k) \quad k \in \{1, 2, \cdots, N_{cls}\} \quad (2)$$



(a) (b)

图 2. 子图 (a) 为使用螺旋填充算法进行探索的过程示意图。子图 (b) 表示未采用独立小区域优先级的探索过程。(b) 中的黄色方框表示正在探索的前沿。(b) 中的红色方框表示将导致未来出现往返路径（虚线）的前沿。

其中，N_{cls} 表示前沿的数量。如图3所示，d_x^k、d_y^k、d_z^k 分别表示沿 X、Y、Z 轴从前沿到边界的最短距离，X、Y、Z 轴与无人机起飞时的初始机体坐标系一致。r_s 是传感器的最大量程，D_k 是前沿 F_k 的视点 V_k 与当前位置 p_0之间的距离。当该前沿靠近当前位置（$D_k < r_s$）时，我们直接将最小距离 $d_{min}(k)$视为 $c_b(k)$。前沿距离边界越远，边界代价 $c_b(k)$就越大。为提高边界前沿的优先级，我们将 $c_b(k)$作为非对称旅行商问题代价函数中的一项，从而得到一个优先探索靠近边界的前沿的序列。同时，为避免探索靠近边界但远离当前位置的前沿，当 D_k 大于 r_s时，我们通过正惩罚因子 w_d 对这些前沿施加惩罚。这样，D_k 越大，前沿代价就越大。

为实现独立小区域优先级，我们设计了一种名为底部射线的方法，如图3所示。我们使用射线（红线）粗略检测前沿后方未知区域的范围，然后根据检测结果评估该区域为独立小区域的概率 c_s^k。主要流程如下：首先，获取视点$V_k(p_k, \xi_k)$与无人机当前位置 p_0 之间的距离 D_k 小于 D_{thr}的前沿，其中 D_{thr} 为距离阈值，用于避免提高距离过远的前沿的优先级。我们使用 [12] 中的相同方法生成视点 V_k，其中包含一个位置p_k 和一个偏航角 ξ_k。该方法分为三个步骤：1）在原点位于前沿中心的柱坐标系中均匀采样点。2）保留覆盖率高于阈值的点，并按覆盖率降序排列。3）选择所需偏航变化和飞行距离较小的视点作为最终视点。其次，如果一个前沿在垂直于 Z 轴的两个方向上都与障碍物相邻，我们认为该前沿后方的未知区域是类似房间或走廊的独立区域，应优先探索。对于这种情况，我们设定 c_s^k = 1

$\overrightarrow{p_k p_a^k}$

(k)。否则，执行下一步。第三，计算从视点 V_k 到 F_k 的平均点 p_a^k的向量（图3中的红线），并将其延伸一个地图分辨率

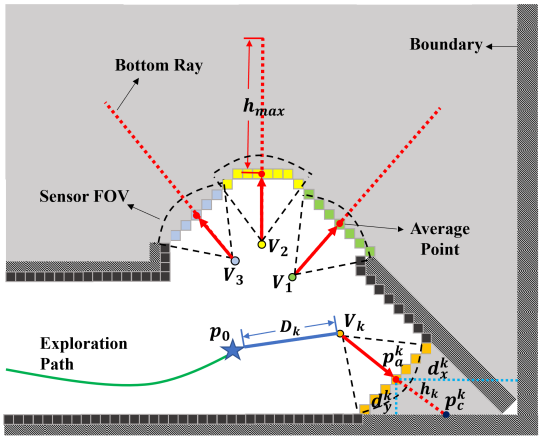


图3. 前沿空间特征评估示意图。V_1、V_2、V_3和V_k是覆盖各前沿的对应最优视点。根据前沿F_k（橙色体素）的平均点p_a^k，我们计算从F_k到探索区域边缘的距离（浅蓝色线）。此外，我们使用底部射线（红色线，由V_k和p_a^k的向量方向生成）粗略评估各前沿（彩色体素）后方未知区域（灰色区域）的范围。当射线触及占据体素、空闲体素、边界或最大距离h_{max}时，我们将射线的距离h_k（红色虚线）视为该前沿的范围。

直到其触及占据体素、空闲体素、边界或超过最大距离h_{max}，从而得到一个底部点p_c^k以及p_a^k与p_c^k之间的距离h_k（图3中的红色虚线）。然后，我们通过以下公式计算这些前沿的独立小区域概率$c_s(k)$：

$$c_s(k) = \frac{h_{max} - h_k}{h_{max}} \tag{3}$$

最后，我们将$c_s(k)$和$c_b(k)$视为前沿层级因素，并将其与[12]中使用的飞行层级因素整合到非对称旅行商问题的成本矩阵M_{tsp}中，如下所示：

$$\begin{aligned} M_{tsp}(0,k) &= t_{lb}(V_0,\ V_k) + w_c \cdot c_c(V_k) \\ &+ w_b \cdot c_b(k) - w_f \cdot c_s(k) \end{aligned} \tag{4}$$

$$t_{lb}(V_0,\ V_k) = \max\left\{ \frac{length(p_0, p_k)}{v_{max}},\ \frac{\min(|\xi_0 - \xi_k|, 2\pi - |\xi_0 - \xi_k|)}{\dot{\xi}_{max}} \right\} \tag{5}$$

$$c_c(V_k) = \cos^{-1}\frac{(p_k - p_0)\cdot v_0}{\|p_k - p_0\|\,\|v_0\|} \tag{6}$$

其中，V_0表示无人机的当前状态，包括当前位置p_0和偏航角ξ_0。V_k表示前沿F_k的视点。v_0是无人机的当前速度。$t_{lb}(V_0, V_k)$和$c_c(V_k)$表示距离、偏航变化和速度变化等飞行层级因素。w_c、w_b和w_f分别是三个代价项对应的权重。根据公式4，代价最小的前沿很可能会导致往返机动，因此应优先探索该前沿，以减少后期的低效探索。其余M_{tsp}的计算方法如下：

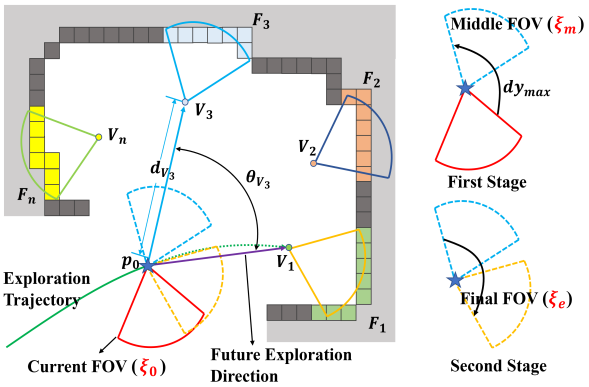


图4. 局部范围内存在多个视点时，所提出的两阶段偏航规划示意图。第一阶段，我们根据偏航变化的幅度计算中间偏航角ξ_m（蓝色视场），然后执行从当前偏航角ξ_0（红色视场）到中间偏航角的相应偏航规划，以便在飞行过程中覆盖更多区域（绿色虚线）。第二阶段，执行从中间偏航角到最终偏航角ξ_e（黄色视场）的偏航规划，以探索目标前沿（绿色网格）。

$$\begin{aligned} &M_{tsp}(k_1,k_2) = M_{tsp}(k_2,k_1) \\ &= t_{lb}(V_{k_1}, V_{k_2}), k_1, k_2 \in \{1,2,\cdots,N_{cls}\} \end{aligned} \tag{7}$$

$$M_{tsp}(k,0) = 0, k \in \{0,1,2,\cdots,N_{cls}\} \tag{8}$$

B. 自适应偏航规划

当无人机配备视场有限的传感器时，除了路径规划，偏航规划也至关重要。通过大量实验，我们发现，无人机飞往局部目标时，往往存在多个区域，只需改变偏航角即可覆盖这些区域，而不会影响飞行方向。若偏航规划得当，无人机便能在一次规划任务中探索更多区域，从而提高探索效率。

然而，覆盖更多区域通常需要更大的偏航变化，也意味着耗时更长。如果规划器在探索过程中始终采用这一策略，探索过程将会极其缓慢。因此，采用这一策略的时机也很重要。基于此，本文设计了一种自适应偏航规划策略。与其他偏航规划方法不同，该策略采用两种规划模式：常规偏航规划和两阶段偏航规划。常规偏航规划根据机器人的当前偏航角以及目标前沿的视点，规划一条平滑的偏航轨迹。当两阶段偏航变化所需时间小于或约等于当前轨迹的飞行时间时，两阶段偏航规划利用更大范围的偏航变化来覆盖更多前沿。其主要流程见自适应偏航规划策略，其中VPs是包含所有视点的集合。V_1和X_0分别是下一个目标视点和当前运动状态。

起初，由于传感器的范围有限，只有与当前位置p_0的距离小于传感器范围r_s的区域才能被覆盖。因此，我们使用**ViewpointsIn-Local()**来计算满足以下条件的视点数量V_k：其

与当前位置 p_0 之间的距离 D_k 小于阈值 d_{thr} （$d_{thr} < r_s$），且这两个点相互可见。此外，$\overrightarrow{p_0V_k}$ 与 $\overrightarrow{p_0V_1}$ 之间的角度 θ_{V_k} 必须小于 90 度，以确保覆盖能力（第 1 行）。接下来，如果存在满足上述条件的视点，我们将采用两阶段偏航规划模式（第 2–11 行）；否则，采用常规偏航规划方法（第 12–15 行）。如采用两阶段规划，如图4所示，我们使用 **FindMiddleYaw()** 计算每个视点与当前状态之间的偏航变化，并找出满足时间条件且偏航变化最大 ξ_m 的视点（蓝色视场）（第 3 行）。随后，根据 ξ_m、当前偏航角 ξ_0、目标前沿的偏航角 ξ_e 与最大角速度$\xi_{\max}$之间的几何关系，使用**EstimateMinYawTime()**粗略计算两次偏航变化所需的最短时间 T_{min} ：

$$T_1 = \frac{\min\left(|\xi_m - \xi_0|, 2\pi - |\xi_m - \xi_0|\right)}{\dot{\xi}_{\max}} \tag{9}$$

$$T_2 = \frac{\min\left(|\xi_e - \xi_m|, 2\pi - |\xi_e - \xi_m|\right)}{\dot{\xi}_{\max}} \tag{10}$$

$$T_{min} = \tau \cdot (T_1 + T_2) \tag{11}$$

其中，τ是可用于提高偏航变化可行性的扩展系数。为避免偏航变化对飞行速度产生过大影响，我们仅在偏航变化时间小于飞行时间时执行两阶段偏航规划（第6行）。然而，一旦根据第III-A节判断该区域可能是一个小区域，为避免后续可能出现的来回机动，我们将毫不犹豫地采用两阶段偏航规划来覆盖该区域（第6行）。随后，如果满足上述条件，则将 T_{min}、当前运动状态 X_0 以及下一个目标视点的位置 p_e 提供给第III-C节介绍的 **PathPlanningAndOpt()** ，以生成飞行路径（第7行），其中 T_{min} 被视为最短飞行时间约束。最后，如果 **PathPlanningAndOpt()** 生成的实际飞行时间 T_{real}大于 T_{min}，则将通过**YawPlanning()**执行两阶段偏航规划（第 8–11行、第12–15行）。在该函数中，我们使用均匀 B 样条表示偏航角轨迹 ϕ（t），该轨迹由N+1 个控制点 Φ：$= \{\phi_0, \ldots \phi_n\}$ 以及节点区间 δt_ϕ参数化。 T 是轨迹的总时间。由于 B 样条具有凸包性质，可以通过改变控制点 Φ 并求解以下问题来优化轨迹的平滑性和动力学可行性：

$$\begin{aligned}\arg\min_{\xi_{cp}} \ & \gamma_1 f_s + \gamma_2 \left(\phi\left(t_0\right) - \xi_0\right) + \gamma_3 \left(\phi(T) - \xi_e\right) \\ & + \gamma_4 \left(f_{\dot{\xi}} + f_{\ddot{\xi}}\right)\end{aligned} \tag{12}$$

其中，f_s 表示平滑性。第二项和第三项是软航点约束，要求 ϕ（t）经过当前偏航角 ξ_0 和目标偏航角 ξ_e。最后两项是针对角速度和加速度动力学可行性的软约束。γ_1、γ_2、γ_3和 γ_4 分别是各项对应的权重。f_s、$f_{\dot{\xi}}$和 $f_{\ddot{\xi}}$的计算方法与 [23]类似。

算法 1： 自适应偏航规划策略

输入： VPs, V_1 (p_e, ξ_e) X_0 (V_0, v_0, a_0) D_k, c_s (k)

输出： 偏航轨迹 Y

1: $N_v \leftarrow$ **局部视点**（VPs）
2: **如果** $N_v > 1$ **则**
3: $\xi_m \leftarrow$**查找中间偏航角**(N_v)
4: $T_1, T_2 \leftarrow$ **估计最小偏航时间**(ξ_0, ξ_m, ξ_e)
5: $T_{min} \leftarrow \tau \cdot (T_1 + T_2)$ $R \leftarrow T_1/T_{min}$
6: **如果** $T_{min} <= D_k/v_0 \ || \ c_s(k) > 0.5$ **则**
7: $T_{real} \leftarrow$ **路径规划与优化**(X_0, p_e, T_{min})
8: **如果** $T_{real} >= T_{min}$ **则**
9: $Y_1 \leftarrow$ **偏航规划**$(\xi_0, \xi_m, T_{real} * R)$
10: $Y_2 \leftarrow$ **偏航规划**$(\xi_m, \xi_e, T_{real} * (1 - R))$
11: **return** $Y(Y_1, Y_2)$
12: $T_{min} \leftarrow$ **估计最小偏航时间**(ξ_0, ξ_e)
13: $T_{real} \leftarrow$ **路径规划与优化**(X_0, p_e, T_{min})
14: $Y \leftarrow$ **偏航规划**(ξ_0, ξ_e, T_{real})
15: **返回** Y

C. 路径搜索与优化

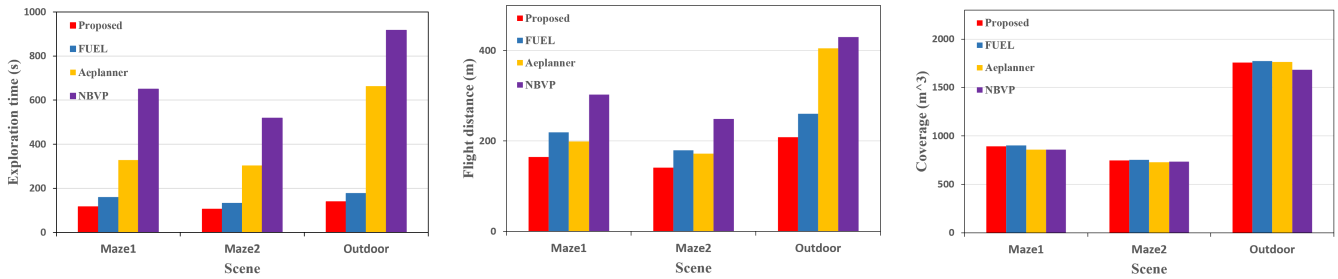
实现快速探索的另一个核心环节是路径规划，其用于生成安全、平滑且满足动力学可行性的轨迹，以实现快速飞行。根据内容，路径规划可分为两部分：路径搜索和路径优化。在路径搜索中，广泛采用运动学-动力学路径搜索 [24] ，利用符合无人机动力学特性的运动基元，寻找无碰撞且满足动力学可行性的路径。然而，当无人机处于特殊场景中，例如在房屋内从室内向室外搜索路径，或穿过狭窄空间时，如果仅采用传统的运动学-动力学路径搜索，由于离散控制空间的限制，搜索过程会耗时较长，甚至可能失败。为减少这种情况的发生，我们采用此前的工作 [25] 来搜索初始路径，该方法利用引导路径提升运动学-动力学路径搜索的稳定性。在路径优化中，我们采用B样条轨迹优化[24]对初始路径进行优化。该方法还利用均匀B样条的凸包性质，通过最小化代价函数生成平滑、安全且满足动力学可行性的轨迹。

D. 自适应动态重规划

目标点的速度通常默认设为零，而每次重新规划的耗时是动态且未知的。因此，低频率的重规划策略会因走走停停机动而导致探索速度较低。此外，如果将当前位置用作规划起点，那么当规划过程耗时较长时，由于新路径的起点与无人机当前位置之间的距离较远，飞行会不稳定。为解决这些问题，本文受到 [26] 的启发，在探索重新规划中采用自适应动态起点。在第 i 个规划步骤中，我们不将当前位置用作规划起点，而是选择未来 t_i 时刻的位置

表 I 三种场景的探索统计

场景	方法	探索时间（s）				飞行距离（m）				覆盖率（m^3）			
		Avg	Std	Max	Min	Avg	Std	Max	Min	Avg	Std	Max	Min
迷宫1	Aeplanner	328.1	43.7	400.9	222.07	199.0	26.3	249.1	**140.9**	859.7	24.0	899.0	805.3
	NBVP	650.3	139.8	928.9	418.2	302.7	60.4	413.3	198.6	859.8	41.5	**950.3**	756.8
	FUEL	158.9	8.2	175.6	146.6	219.0	10.5	242.4	200.6	**901.8**	3.2	909.0	**896.0**
	所提出的方法	**116.3**	**6.4**	**128.3**	**103.1**	**164.6**	**10.0**	**182.9**	141.6	892.9	**2.3**	898.4	889.2
迷宫2	Aeplanner	302.0	31.0	346.9	235.0	171.7	17.1	198.2	126.7	726.8	14.3	745.4	680.2
	NBVP	517.9	76.3	653.7	336.2	248.7	37.8	320.8	159.6	733.3	23.5	**768.9**	689.0
	FUEL	132.4	7.6	145.9	117.5	179.7	10.8	196.2	150.5	**753.3**	3.2	762.7	**748.2**
	所提出的方法	**105.5**	**6.0**	**116.0**	**95.5**	**141.1**	**10.0**	**159.1**	**122.2**	747.1	**2.1**	751.9	742.8
室外	Aeplanner	661.2	101.0	856.2	463.8	405.2	52.5	505.0	308.3	1763.5	9.7	**1783.7**	1746.1
	NBVP	917.1	185.9	1390.4	630.5	429.9	76.4	591.6	300.8	1683.3	76.5	1777.3	1501.9
	FUEL	177.0	9.5	201.7	161.1	260.1	14.4	298.5	236.4	**1773.9**	**4.3**	1781.7	**1767.6**
	所提出的方法	**139.7**	**8.5**	**154.8**	**121.6**	**208.2**	**11.7**	**232.0**	**186.4**	1755.9	10.5	1776.9	1735.0



(a) (b) (c)

图 5。(a)、(b) 和 (c) 分别 分别为表 I 中呈现的平均探索时间、平均飞行距离和平均覆盖率 ly.

表 II
仿真参数

最大速度	2.0 m/s	最大加速度	2.0 m/s^2
最大偏航角速度	1.0 弧度/秒	相机范围	4.5 米
相机视场角	[80,60] 度	ROS 版本	ROS Melodic

作为规划起点，且 t_i 并非恒定值，而是由上一次规划结果决定：

$$t_i = \max\left(\rho \cdot t_{i-1}, t_{min}\right) \tag{13}$$

其中， t_i， t_{i-1} 分别表示第 i 步和第 i-1 步规划步骤的耗时。 t_{min} 是一次规划所需的最短时间。 ρ 是用于提高可靠性的扩展系数。如果规划成功且实际规划时间小于 t_i，则在时间 t_i 后根据规划结果更新路径。否则，执行重新规划。此外，为了保持飞行速度和流畅性，当剩余飞行路径的持续时间少于 1s 时，我们会进行重新规划。

IV. 实验结果

A. 基准比较

在仿真实验中，我们在三种不同环境下将所提出的方法与三种最先进方法进行比较。这三种方法分别是 FUEL [12],Aeplanner[10], 和 NBVP [8]。对于所有方法，我们均采用其开源实现及默认配置（Aeplanner 和 NBVP 的最大节点数设为 200）。在每种场景中，这些方法均在 5 个不同的初始位置进行测试（位置之间相距 5 米）并且每种方法在每个位置测试 5 次。实验中使用的配置如表 II 所示。我们在配备 Intel Core i9-9900K@ 3.6GHz 和 64GB 内存的计算机上测试这些方法。

备注： 尽管有许多传感器能够观测较大范围的区域，但它们本质上仍属于有限视场传感器。 Moreover，由于我们关注的是探索方法的质量，因此我们认为，即使使用观测范围较窄的传感器，该方法仍能取得更好的性能，这比使用观测范围较宽的传感器更具挑战性。

1）迷宫场景：迷宫是验证探索方法效率的最有效环境。因此，我们首先比较两种迷宫环境中的探索效率：矩形迷宫（迷宫1）和正方形迷宫（迷宫2）。迷宫1的场景范围为 $30 \times 16 \times 2\ m^3$,，迷宫2的场景范围为 $20 \times 20 \times 2\ m^3$。实验结果如图6和表I所示。可以看出，NBVP耗时最长、飞行距离最远，而且效率也不稳定。Aeplanner是NBVP的一种改进方法，它将前沿探索策略与NBVP相结合，从而提高了效率。由于具备高效的全局覆盖路径和最短时间飞行路径，所提出的方法和FUEL相较于上述两种方法具有明显优势。所提出的方法不仅飞行路径更加平滑，而且在确保较高覆盖率的同时，飞行时间和飞行距离也更短。同时，得益于更少的往返机动和偏航规划策略，所提出的方法比FUEL实现了更高效的探索。与FUEL相比，我们的方法在迷宫1中的探索时间和飞行距离分别减少了26.8%和24.8%，在迷宫2中分别减少了’20.3%和21.5%，且探索比率的变化趋于线性。

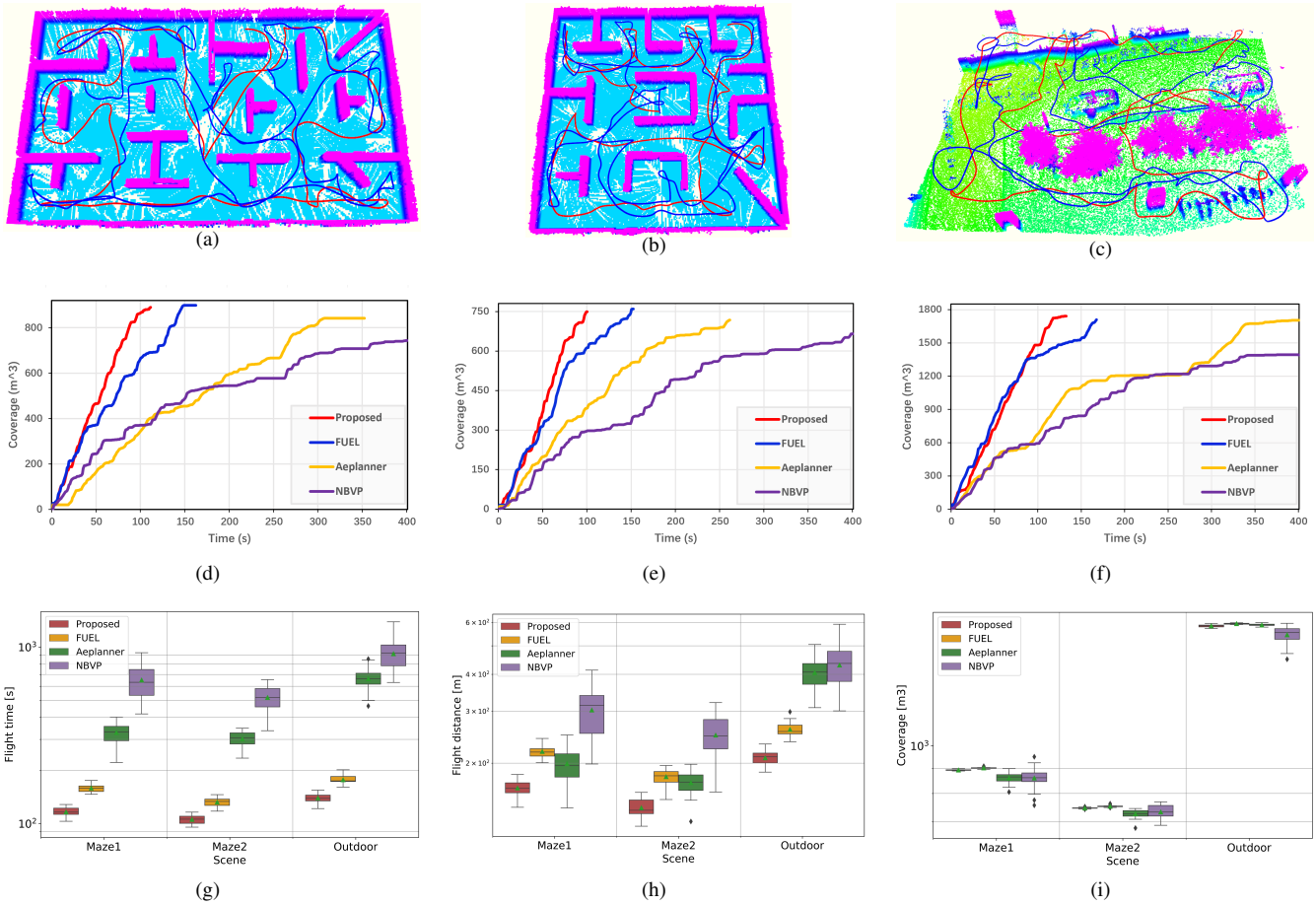


图 6. 不同场景下实验结果的基准比较。(a)、(b) 和 (c) 分别对应迷宫1、迷宫2和室外场景（其中红色和蓝色轨迹分别表示所提出的方法和 FUEL 在相同初始位置下的最佳表现。为清晰起见，我们仅展示上述两种方法）。(d)、(e) 和 (f) 分别展示了四种对比实验在三种不同场景下的覆盖率表现。(g)、(h) 和 (i) 分别展示了三种不同场景下四种方法的实验结果（见表 I）中的飞行时间、飞行距离和覆盖率的具体分布。更多比较请参见 https://github.com/Zyhlibrary/FAEP。

2）室外场景：我们还在自然场景中比较了这四种方法，以验证其稳定性和效率。该场景包含树木、汽车、围栏和其他物体，范围为 $20 \times 30 \times 3$ m^3。实验结果同样如图6和表I所示。结果表明，由于场景复杂度增加，与迷宫环境相比，四种方法的探索时间和距离均有所增加，但所提出的方法与其他方法相比仍具有明显优势。与NBVP和Aeplanner相比，我们的方法速度提高了5至7倍。与FUEL相比，我们的方法在探索时间和飞行距离方面仍分别保持21.1%和20%的优势。此外，我们可以看到，我们的方法和FUEL的效率基本相同，或者FUEL在探索初期表现更好；但在中后期，我们的方法逐渐显现出优势。这是因为在探索初期，无论优先探索哪个前沿，都能获得较高的信息增益，因此全局探索序列的质量不会影响初期的探索效率。由于我们的方法会及时探索那些信息增益可能较低、但会导致往返机动的前沿，因此在探索初期，我们的方法效率可能不是最高的。然而，

表 III 不同场景下的平均计算时间

方法	三个场景下的实验结果（毫秒）			
	迷宫1	迷宫2	室外	平均值
Aeplanner	623	756	761	713
NBVP	794	912	926	877
FUEL	29	25	33	29
所提出的方法	24	21	31	25

随着探索的推进，由低质量全局探索序列引发的往返机动逐渐显现，导致 FUEL 的效率降低。由于探索序列质量较高，如图6(a)6(b)6(c)所示，我们的方法往返机动显著更少，因此在整体探索效果方面表现更佳。

3）进一步评估：我们还可以看到，所提出的方法在不同初始位置下生成的结果波动范围非常有限，这表明不同初始位置对所提出方法最终探索性能的影响有限。这是因为一旦确定了探索区域，无论初始位置在哪里，它都只会影响早期阶段的探索区域和探索轨迹，而所提出的方法会引导无人机按照我们的探索原则探索

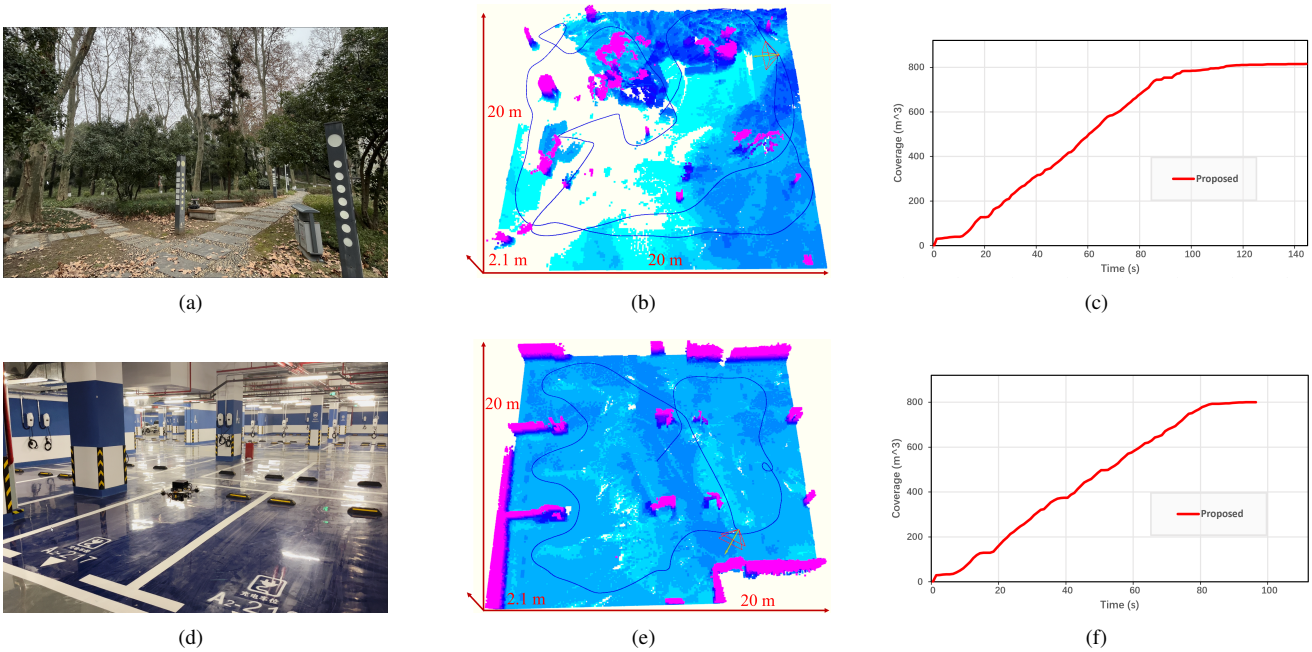


图7。真实世界实验的结果。(a)、(b)和(c)为公园中的实验结果。(d)、(e)和(f)为地下停车场中的结果。需要注意的是，(b)和(e)中的路径是实际的探索轨迹，通过记录同步定位与建图提供的无人机位置获得。(e)和(f)展示了探索覆盖率相对于探索时间的变化，该覆盖率可通过已知体素栅格的数量获得。实验视频可在 https://youtu.be/0Y671mEwJ–A 。

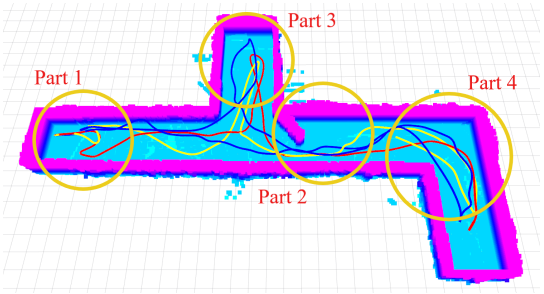


图8。用于清晰解释我们方法改进效果的走廊场景。红色轨迹和蓝色路径分别是我们的方法和FUEL的实际探索轨迹。黄色轨迹是未采用自适应偏航规划部分时我们方法的探索轨迹。橙色圆圈表示导致效率差异的关键部分。

该区域。此外，我们比较了所提出的方法与三种最先进方法的计算时间，比较结果如表 III 所示。我们可以看到，由于采用了包含前沿信息结构的启发式框架，所提出的方法和 FUEL 能够轻松地在整个空间中维护关键信息。因此，所提出的方法和 FUEL 在计算时间方面表现更佳。

此外，为了更清楚地展示我们的贡献，如图8所示，我们在走廊中设计了一项简单而有效的对比实验，对比我们的方法、未采用自适应偏航规划的方法（为便于说明，称为方法-1）以及 FUEL。完成探索时，FUEL 耗时 45.4 s，行进距离为 74.4 m；方法-1 耗时 32.2 s，行进距离为 60.5 m。我们的方法表现最佳，耗时 30.7 s，行进距离为 55.9 m。从图8可以看出，我们的轨迹更合理，往返机动更少，这主要与

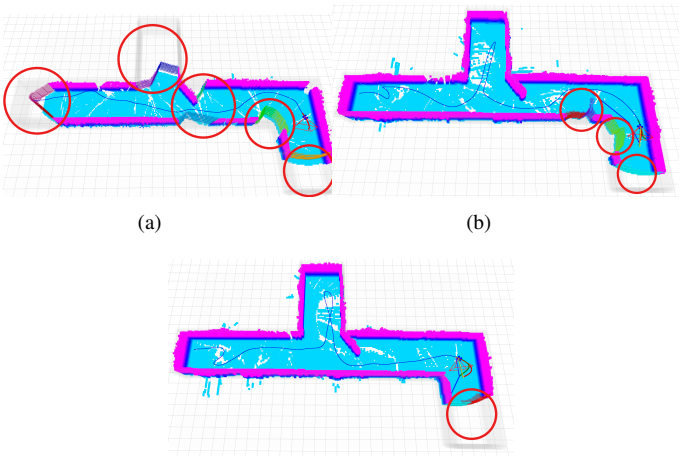


图9. 无人机飞往第 4 部分时的全局探索状态。(a)、(b) 和 (c) 分别为 FUEL、方法-1 和我们的方法的结果。红色圆圈标示尚未探索的区域。

四个部分：第1部分、第 2 部分、第 3 部分和第 4 部分。第1部分是探索开始的区域。由于视场的限制，一些小区域会被遗漏。第 2 部分包含一个拐角，该拐角会导致某个区域难以及时探索。第 3 部分是一个突出走廊。第 4 部分是一个包含较大拐角的区域，转弯时很容易遗漏部分区域。对于第1部分和第 2 部分，由于我们优先探索第III-A节中提到的小型独立区域，因此，即使在这些区域停留时间较长，我们的方法和方法-1也会及时探索这些区域，这会略微牺牲局部效率，但可以避免往返机动造成的后续更大影响。因此，可以看到，我们的方法和方法-1的轨迹会明显绕行，以覆盖该区域

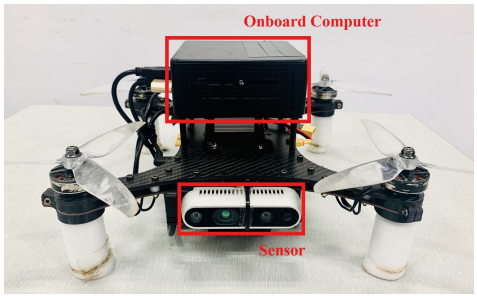


图 10. 真实世界实验中使用的定制四旋翼无人机。

表 IV
定制四旋翼无人机的系统参数

轴距	330 毫米	飞行控制	CUAV V5+
电机	TMOTOR F90 KV1300	螺旋桨尺寸	7 英寸
传感器	Intel RealSense D435	计算机	英特尔 NUC

这两个部分中的区域。对于第 3 部分，由于采用了图 2(a)所示的螺旋填充概念，我们的方法（包括方法-1）会优先探索该部分，以减少低效路径。在第 4 部分，由于我们的方法采用了第III-B节中提到的自适应偏航规划，我们将通过偏航变化覆盖更多前沿。因此，与方法-1和FUEL相比，我们不会在第 4 部分留下需要返回并再次探索的未探索区域。为证明上述分析，如图 9所示，我们还提供了无人机飞行至第 4 部分时三种方法的全局探索状态，未探索区域的分布与我们的分析一致。

B. 复杂度分析

我们的方法包含三个主要部分：基于非对称旅行商问题的全局路径求解、局部路径规划与偏航规划，以及地图维护。第一部分中，我们在全局规划中使用 LKH [27] 求解非对称旅行商问题。根据 [27], 中的证明，LKH 的复杂度为 $O(n^2)$。因此，这部分的计算复杂度为 $O(N_{cls}^2)$，其中 N_{cls} 表示前沿的数量，其大小取决于已知区域与未知区域之间边界的长度。当边界长度超过距离阈值（约 2 米）时，边界将被分割为两个前沿。按照这一原则，可以确定 N_{cls} 。在规划部分，我们使用 B 样条生成局部轨迹和偏航路径，因此这部分的计算复杂度为 $O(N_c)$，其中 N_c 是 B 样条控制点的数量， N_c约等于规划时域除以相邻控制点之间的距离间隔（约 0.5 米）。在地图维护中，由于我们只更新局部体素栅格，时间复杂度为 $O(N_v)$，其中 N_v 是受传感器数据影响的体素栅格数量，其大小由传感器的视场决定。

C. 真实世界实验

为了验证所提出的方法的有效性，我们还使用定制四旋翼无人机，在公园和地下停车场进行了两项真实世界实验，如图 10 所示。我们为无人机配备了一个视场受限的传感器，并使用 [28] 提供四旋翼无人机的状态。所有模块均运行在一台机载计算机 NUC 上，该计算机配备英特尔酷睿 i5-1135G7@ 2.40GHz、16GB 内存和 ROS Melodic。详细的系统参数如表 IV 所示。在实验中，我们将动态限制设为 $v_{max} = 1.0$m/s、 $a_{max} = 1.0$ m/s^2 和 $\dot{\xi}_{max} = 1.0$ rad/s。

首先，为了验证我们的方法在自然场景中的表现，我们在公园进行了探索实验。该场景中包含树木、灌木、石凳及其他物体。我们用一个 $20 \times 20 \times 2.1$ m^{3}的盒子界定了探索场景。探索结果如图7的子图(a)、(b)和(c)所示。整个过程的探索时间为144.5 s，飞行距离为131.0 m。需要注意的是，我们预先设定的探索区域最小高度为 -0.1 m，但公园地面并不平坦，其中有一大片低洼区域，因此图7子图(b)中出现了视觉上的空白区域（没有点云）。随后，为了验证我们的方法在地下场景中的表现，我们还在地下停车场进行了探索实验，并用一个 $20 \times 20 \times 2.1$ m^{3}的盒子界定场地。实验结果如图7的子图(d)、(e)和(f)所示。整个探索过程的探索时间和飞行距离分别为94.3 s和90.2 m。此外，两次真实世界实验采集的数据均存储为分辨率为0.1 m的占据栅格地图中（图7子图(b)和(e)），两张地图的大小分别为1.82 MB和3.13 MB。上述两项实验表明，我们的方法能够使用有限视场传感器，在自然场景和室内环境中高效、安全地完成探索任务。我们还在图7中提供了视频演示，供读者了解更多细节。

V. 结论与未来工作

本文提出了一种自主探索规划器，用于由配备有限视场传感器的无人机快速完成未知环境建图。首先，我们设计了一种全面的前沿探索序列生成方法，不仅考虑飞行层面的代价（距离、偏航变化和速度方向变化），还考虑前沿的空间特征，以减少往返机动。其次，根据前沿分布和无人机的飞行状态，提出了一种自适应偏航规划方法；当两阶段偏航角变化所需的时间小于或近似等于当前轨迹的飞行时间时，该方法能够覆盖更多前沿。最后，通过选择动态起点和相应的重规划策略，采用路径规划和动态重规划来提高飞行过程的稳定性和流畅性。仿真和真实世界实验均验证了所提出方法的有效性。

参考文献

[1]X. Zhang、Y. Chu、Y. Liu、X. Zhang 和 Y. Zhuang，“一种采用均匀采样的四旋翼无人机新型信息驱动自主探索策略”，*IEEE*工业电子学汇刊，2021年。[2] H. Wang、S. Zhang、X. Zhang、X. Zhang 和 J. Liu，“满足摄影测量约束的四旋翼无人机近最优三维视觉覆盖”，*IEEE*工业电子学汇刊，第69卷，第2期，第1694–1704页，2021年。

[3] X. Zheng、F. Wang 和 Z. Li，“用于三维高分辨率建筑模型重建的多无人机协同路径规划方法”，*ISPRS*摄影测量与遥感学报，第146卷，第483–494页，2018年。[4] S. W. Chen、G. V. Nardari、E. S. Lee、C. Qu、X. Liu、R. A. F. Romero 和 V. Kumar，“SLOAM：用于森林资源清查的语义激光雷达里程计与建图”，*IEEE*机器人与自动化快报，第5卷，第2期，第612–619页，2020年。[5] M. Erdelj、E. Natalizio、K. R. Chowdhury 和 I. F. Akyildiz，“来自天空的助力：利用无人机进行灾害管理”，*IEEE*普适计算，第16卷，第1期，第24–32页，2017年。[6] P. Ramesh 和 J. M. L. Jeyan，“运行参数对微型无人机军事与民用应用影响的比较分析”，载于*AIP*会议论文集，第2311卷，第1期，页码 030034。AIP出版有限责任公司，2020年。[7] J. Jiang、J. Yuan、X. Zhang 和 X. Zhang，“DVIO：一种基于优化的紧耦合直接视觉惯性里程计”，*IEEE*工业电子学汇刊，第68卷，第11期，第11 212–11 222页，2020年。[8] A. Bircher、M. Kamel、K. Alexis、H. Oleynikova 和 R. Siegwart，“用于三维探索的滚动时域‘下一最佳视点’规划器”，载于*2016*年*IEEE*国际机器人与自动化会议（*ICRA*），第1462–1468页。IEEE，2016年。[9] Z. Meng、H. Qin、Z. Chen、X. Chen、H. Sun、F. Lin 和 M. H. Ang，“用于三维未知环境探索与结构重建的两阶段优化下一视点规划框架”，*IEEE*机器人与自动化快报，第2卷，第3期，第1680–1687页，2017年。[10] M. Selin、M. Tiger、D. Duberg、F. Heintz 和 P. Jensfelt，“大规模三维环境的高效自主探索规划”，*IEEE*机器人与自动化快报，第4卷，1706，2019年 2，第1699页。[11] T. Cieslewski、E. Kaufmann 和 D. Scaramuzza，“多旋翼飞行器快速探索：一种面向高速飞行的边界选择方法”，载于*2017*年*IEEE/RSJ*智能机器人与系统国际会议（*IROS*），第2135–2142页。IEEE，2017年。[12] B. Zhou、Y. Zhang、X. Chen 和 S. Shen，“FUEL：采用增量前沿结构和分层规划的无人机快速探索”，*IEEE*机器人与自动化快报，第6卷，第2期，第779–786页，2021年。[13] C. Connolly，“下一最佳视点的确定”，载于论文集：*1985*年*IEEE*国际机器人与自动化会议，第2卷，第432–435页。IEEE，1985年。[14] Z. Xu、D. Deng 和 K. Shimada，“通过增量采样和概率路线图实现动态环境中的无人机自主探索”，*IEEE*机器人与自动化快报，第6卷，第2期，第2729–2736页，2021年。[15] V. M. Respall、D. Devitt、R. Fedorenko 和 A. Klimchik，“快速[23] B. Zhou、J. Pan、F. Gao 和 S. Shen，“RAPTOR：面向四旋翼无人机快速飞行的鲁棒感知轨迹重规划”，*IEEE*机器人学汇刊，第37卷，第6期，第1992–2009页，2021年。

基于采样的微型飞行器下一最佳视点探索算法，”载于*2021*年 *IEEE* 国际机器人与自动化会议（*ICRA*），第89–95页。IEEE，2021年。[16] C. Cao、H. Zhu、H. Choset 和 J. Zhang，“Tare：一种用于高效探索复杂三维环境的分层框架”，载于机器人：科学与系统会议（*RSS*），线上，2021年。[17] H. Zhu、C. Cao、Y. Xia、S. Scherer、J. Zhang 和 W. Wang，“Dsvp：一种通过动态扩展实现快速探索的双阶段视点规划器”，载于*2021*年 *IEEE/RSJ*智能机器人与系统国际会议（*IROS*），第7623–7630页。IEEE，2021年。[18] B. Yamauchi，“一种用于自主探索的基于前沿的方法”，载于*1997*年 *IEEE*机器人与自动化计算智能国际研讨会论文集 *CIRA'97*——“迈向机器人与自动化的新计算原理”，第146–151页。IEEE，1997年。[19] S. Shen、N. Michael 和 V. Kumar，“基于随机微分方程的微型飞行器室内自主三维探索算法”，《国际机器人研究杂志》，第31卷，第12期，第1431–1444页，2012年。[20] P. Zhong、B. Chen、S. Lu、X. Meng 和 Y. Liang，“面向空中机器人的信息驱动快速行进自主探索”，*IEEE*机器人与自动化快报，第7卷，第2期，第810–817页，2021年。[21] E. Galceran 和 M. Carreras，“机器人覆盖路径规划综述”，《机器人与自主系统》，第61卷，第12期，第1258–1276页，2013年。[22] E. Gonzalez、O. Alvarez、Y. Diaz、C. Parra 和 C. Bustacara，“BSA：一种完全覆盖算法”，载于*2005*年 *IEEE*国际机器人与自动化会议论文集，第2040–2044页。IEEE，2005年。[24] B. Zhou、F. Gao、L. Wang、C. Liu 和 S. Shen，“用于快速自主飞行的稳健高效四旋翼轨迹生成”，*IEEE*机器人与自动化快报，第4卷，第4期，第3529–3536页，2019年。[25] Y. Zhao、L. Yan、Y. Chen、J. Dai 和 Y. Liu，“基于引导路径的四旋翼快速自主飞行稳健高效轨迹重新规划”，《遥感》，第13卷，第5期，第972页，2021年。[26] J. Tordesillas、B. T. Lopez 和 J. P. How，“FASTER：一种用于未知环境飞行的快速安全轨迹规划器”，载于*2019*年 *IEEE/RSJ*智能机器人与系统国际会议（*IROS*），第1934–1940页。IEEE，2019年。[27] K. Helsgaun，“林–克尼汉旅行商启发式算法的有效实现”，《欧洲运筹学杂志》，第126卷，第1期，第106–130页，2000年。[28] T. Qin、P. Li 和 S. Shen，“VINS-Mono：一种稳健且通用的单目视觉惯性状态估计器”，*IEEE*机器人学汇刊，第34卷，第4期，第1004–1020页，2018年。