arXiv:2302.01833v1 [cs.RO] 3 Feb 2023

© 2022 IEEE

允许个人使用本材料。对于所有其他用途，包括在任何现有或未来媒介中重印或再版本材料以用于广告或促销目的、创建新的合集作品、转售或重新分发至服务器或列表，或在其他作品中重复使用本作品的任何受版权保护部分，均须获得 IEEE 的许可。

SphereMap：用于快速安全感知无人机规划的动态多层图结构

Tom´aˇs Mus,Matˇej Petrl´ık,Martin Saska

摘要—本文提出了一种灵活的拓扑表示：在无人机（UAV）上机载构建双层图结构，通过不断用相交球体填充占据地图中的自由空间来实现。大多数最先进的规划方法在规划最短路径时，会使无人机与障碍物保持预先设定的距离。采用所提出的结构进行规划，仅在必要时才达到这一预设距离，其他情况下则保持更安全的间距，同时速度比其他最先进的方法快几个数量级。此外，我们展示了如何将这种图表示转换为轻量级、可共享的环境拓扑—体积地图，从而实现去中心化的多机器人协作。我们在数公里范围的真实地下环境中成功验证了所提出的方法，包括洞穴、受损的工业建筑，以及 DARPA SubT 挑战赛决赛中严苛复杂的场景。该赛事旨在尽可能真实地模拟搜索与救援任务的条件，而我们的方法在虚拟赛道中取得了第 2nd 名。

索引词—自主飞行器导航、不确定环境下的规划、建图、航空系统：感知与自主性、危险环境中的机器人技术

多媒体材料

本文附有多媒体材料，可访问 mrs.felk.cvut.cz/papers/ral2022spheremap 获取。

一、引言

THANKS 于机载软件成本日益降低且发展迅速，无人机正越来越多地用于搜索与救援（SAR）任务 [1] 以及工业 [2] 或历史 [3] 建筑的巡检。无人机的高机动性使其成为应急救援人员和救援队的理想工具。快速飞越未知且可能存在危险的环境，可让操作员迅速获得态势感知，从而做出更好的决策；在安全状况未知的未探索环境中进行探索时，也能避免危及人类生命。

文献中已有通过无人机完成自主任务的系统 [4], [5]，可部署于各种场景中，例如巡检 [6]、消防 [7], [8] 和辐射定位 [9]。这些应用通常依赖全球导航卫星系统进行无人机定位，并借助额外的传感器实现大多较为简单的反应式行为。另一方面，在缺乏可靠全球导航卫星系统数据的大型非结构化环境中，无人机导航仍是一个尚待解决的课题。近期，DARPA SubT 隧道赛道[10]、城市赛道 [11],和洞穴赛道 [12]的推动加速了这一领域的发展。

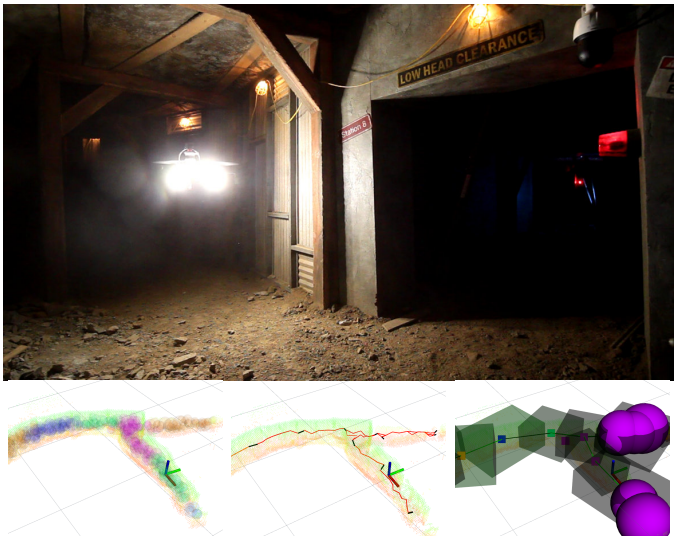

图 1：DARPA SubT 挑战赛最终赛事期间，在赛道隧道区域中构建的 SphereMap。自由空间由球体填充（左图），形成一个以球心为顶点的无向图。中图展示了近似凸区域之间的门户（黑色）以及缓存的区域内路径（红色）。右图展示了 LTVMap 的包围盒，颜色越深表示区域探索／检查得越少，紫色球体标记探索目标。

在任何环境中寻找无碰撞路径，都需要对自由空间和障碍物进行表示。基于八叉树数据结构构建的体素占据栅格（例如 OctoMap [13] 或 UFOMap[14]实现）通常用于将这些信息编码为单元格的状态。尽管基于八叉树的占据地图内存效率高且访问速度快，但在占用栅格上进行朴素规划，可能无法满足实时长距离规划（全局导航）的要求，尤其是在地下环境典型的复杂拓扑结构中。现有基于超时的变通方法 [11], 会输出一个终点与初始目标之间欧几里得距离最短的路径。这种方法避免了等待规划结果时长时间空等，但在拓扑结构较复杂的情况下，可能导致根本无法到达目标。许多全局导航问题的解决方法会利用环境拓扑结构构建地图，以实现快速路径规划，例如 [15]–[17]。然—

稿件于2022年2月24日收到；2022年6月20日修订；2022年7月18日录用。经副编辑和审稿人评审后，编辑 Tamim Asfour 推荐本文发表。本研究得到美国国防高级研究计划局（DARPA）、捷克科学基金会（GACR）项目编号20-29531S、TAČR项目编号FW03010020、NAKI II计划下项目编号DG18P02OVV069、布拉格捷克理工大学资助项目编号SGS20/174/OHK3/3T/13，以及由OP VVV资助的项目CZ.02.1.01/0.0/0.0/16 019/0000765“信息学研究中心”的支持。作者就职于布拉格捷克理工大学电气工程学院控制论系，地址：166 36 布拉格6，{tomas.musil |matej.petrlik |martin.saska}@fel.cvut.cz。数字对象标识符（DOI）：见本页顶部。

© IEEE，2022。已被 IEEE ROBOTICS AND AUTOMATION LETTERS (RA-L) 接收。数字对象标识符：10.1109/LRA.2022.3195194

而，这些方法中的大多数都是在首次遍历环境后才构建拓扑地图，而非在任务过程中逐步构建。因此，它们无法部署在进入此前未绘制地图的环境中的应急响应无人机上，而这正是搜索与救援场景中的典型需求。那些能够逐步构建地图的方法，往往也不会将障碍物间隙信息纳入地图，以进行安全感知规划。因此，当无人机需要迅速决定是探索附近碰撞风险较高的狭窄通道，还是前往更远处存在更安全路径的目标时，这些方法仍然无法派上用场。

本文介绍了一种增量构建的图结构（称为 SphereMap），它由一个稀疏拓扑图和一个底层的相交球体稠密图构成；稀疏拓扑图表示大致凸形的自由空间区域。任务执行期间，该结构可在无人机上持续机载构建，并使其能够实时权衡路径安全性、路径长度及剩余飞行时间，从而对任务目标作出决策。此外，该结构还能生成一种轻量级的环境表示（称为 LTVMap），由表示环境体积的凸形状图构成，可随时与其他机器人或人工操作员共享。

A. 相关工作

文献 [16] 提出了一种为快速路径规划而设计的轻量级拓扑地图表示。在该研究中，作者基于机器人在环境中运行一趟后离线生成的欧几里得符号距离场（ESDF），逐步扩展自由空间的凸簇。随后利用生成的地图在环境中快速规划路径。由于这些簇是凸的，路径规划在导航图上进行，图中的顶点是簇彼此重合处的点（称为门户）。边则是穿过一个簇、连接两个门户的直线路径。该方法要求簇采用固定的体素尺寸，这在环境同时存在极窄区域和开阔区域时可能带来问题。此外，离线生成 ESDF 的做法妨碍了在未知环境中的探索性部署，也无法应对动态变化环境。

另一种创建环境稀疏图表示的方法见 [15]。该方法根据欧几里得符号距离场地图创建三维沃罗诺伊图，并将其转换为稀疏图，从而能够在欧几里得符号距离场上实现比 A* 算法或 RRT* [18] 快几个数量级的规划。然而，该稀疏图并非动态构建，而是在初始探索任务完成后才构建。此外，图中也未纳入净空信息，不过作者表示将在未来工作中处理这一问题。

一种用于自主探索的拓扑图构建流行方法 GBPlanner 在 [19]中提出。在该方法中，全局图会在任务执行过程中逐步构建，并用于返航，以及在抵达死胡同时将无人机重新定位至新的前沿点。该方法的缺点是，无法轻松查询占据栅格地图中任意两点之间的路径，而且构建图的目的在于提供最短返航路径。因此，在规划前往新前沿点的路线时，无法权衡路径长度与路径安全性。

一种类似的方法在 [17], 中提出，作者逐步构建由覆盖点组成的稀疏拓扑图，从这些覆盖点发射射线穿过占据栅格地图，并在每个覆盖点处创建一个凸多面体。如果某个多面体的质心位于另一个多面体内部，这些覆盖点也会合并。这是一种增量式且非常轻量的拓扑表示，但不包含隐式净空信息，而且作者提出的路径规划仅基于距离。

[20], 提出了一种安全感知路径规划技术，作者构建了路径优化准则，其中包含路径上给定点到障碍物的距离信息，最大距离为 d_{max} 。在该方法中，作者通过构建与障碍物相距 d_{max} 的图（或位于两个障碍物之间、距离更小的图）来求解优化问题。路径沿该图生成，因此无人机要么与障碍物保持 d_{max}的距离飞行，要么穿过两个障碍物之间的脊线，要么沿直线飞过两个障碍物之间。该方法难以适应较大规模的环境，因为它仅维护尺寸为传感器量程 3–4 倍的局部地图，因此无法在更大的环境中实现全局导航。与我们在室外和室内/地下环境中均经过验证的方法不同，该研究仅考虑室外环境。

为在局部地图边界之外进行规划， [21] 构建了一种稀疏图，其中包含两类顶点：机器人过去位置的历史记录，用于在已探索环境中快速导航；以及表示通往新空间的潜在路径的可见性节点。通过加入碰撞概率惩罚项来确保局部轨迹的安全性。该惩罚项基于运动基元选择代价函数中的位姿估计协方差。所提出的方法缺少在动态环境中重新规划稀疏图的技术，而此前畅通的路径可能会突然受阻。该方法的评估也仅在简化的仿真中进行，且未考虑传感器数据不完善、地图漂移、控制误差和外部扰动等常见的现实问题。

B. 主要贡献

结合上述讨论的相关文献，本文的贡献可概括如下：

- 一种新颖的动态构建图结构（SphereMap），包含净空信息，可在已知空间中的任意两点之间实现快速、安全感知的路径规划，并能够应对结构任意且不断变化的环境，
- 一种快速生成环境轻量级体积图表示（LTVMap）的方法，该表示可通过低带宽通信与其他机器人共享，并承载任务关键信息，
- 实验评估表明，对于长距离规划，在 SphereMap 上进行规划的速度可比最先进的方法快几个数量级，
- 在真实世界场景中验证所提出的方法，尤其是在 DARPA SubT竞赛中；在该竞赛中，机器人被部署到真实和模拟的

此前未访问的环境 [22]。

II. 问题描述

本文所解决的规划问题是在实时条件下，计算穿越由占据栅格地图表示的已知空间的长距离路径，同时兼顾路径长度和路径安全性。假设无人机被建模为半径为 r_{min} 的球体。由于所得路径将作为局部规划器的输入，因此不考虑无人机的动态约束。形式化地，我们将规划问题定义为：寻找由 $k \in [0,1]$ 参数化的解路径 $\mathbf{x}(k)$，使准则

$$J(\mathbf{x}(k)) = L(\mathbf{x}(k)) + Z(\mathbf{x}(k)), \tag{1}$$

分为路径长度 $L(\mathbf{x}(k))$ 和路径风险 $Z(\mathbf{x}(k))$，定义如下

$$L(\mathbf{x}(k)) = \int_0^1 \|\mathbf{x}'(k)\|\, dk \tag{2}$$

$$Z(\mathbf{x}(k)) = \xi \int_0^1 \|\mathbf{x}'(k)\| \left[\max(0, d_{max} - d(\mathbf{x}(k)))\right]^2 dk \tag{3}$$

同时满足

$$\min_{k\in(0,1)} d(\mathbf{x}(k)) > r_{min} \tag{4}$$

其中，$d(\mathbf{x})$ 是点 $\mathbf{x}$ 处的最近障碍物距离，ξ 是一个参数，用于指定路径安全性在路径总体代价中的权重。代价截断距离 d_{max} 根据无人机的尺寸以及扰动和控制误差可能造成的最大偏差确定。

需要解决的第二个问题是，能够随时高效生成所访问环境部分的轻量级拓扑体积表示，其中包含有关环境各部分的任务相关信息，以便与机器人与人类团队的其他成员共享。我们进一步为该问题规定以下要求和假设：

1）任务：探索给定环境，并在通信可用时将探索信息和任务相关信息共享给其他无人机和人工操作员。2）环境：无结构且未知的动态地下或室外环境，其中包含狭窄通道、低带宽通信不可靠的区域，且无法使用全球导航卫星系统。

3）无人机平台：假定搭载深度传感器（激光雷达或深度相机），并在机载端构建环境的占据栅格地图，同时使用同步定位与建图方法进行自身定位。无人机被建模为球体，不考虑平台的物理碰撞容差，例如 [23] 中所述。4）规划时间：该方法必须能够找到通往数百米外目标点的路径。完成这一过程必须只需几十毫秒，这样无人机就能随时评估通往所有潜在目标点的路径，因为有时，与只能通过高风险通道到达的较近目标相比，较远但安全的目标可能更可取。

III. SphereMap

本节将介绍所提出的多层图结构、逐步构建该结构的方法，以及如何从该结构中快速提取轻量级拓扑体积地图。

A. 拓扑地图表示

我们提出将环境的拓扑表示为由球体构成的无向图 $\mathcal{G}$，并将环境划分为大致凸的分段 $\mathcal{S}$。球面图上的每个节点 ν 都被赋予一个位置 $\mathbf{p}_\nu$、一个半径 r_ν（等于距离 $\mathbf{p}_\nu$最近的障碍物距离），以及一个标签 σ_ν，用于指明该节点所属的分段。对于交集半径大于 r_{min} 的任意两个球体，我们都将它们连接起来。对于规划而言，图3所示的球面图与 [24] 中使用的欧几里得符号距离场（ESDF）类似，因为它本身就编码了与障碍物之间的距离。然而，与 [24], 不同的是，所提出的地图表示不受固定分辨率网格的限制；当体素尺寸过大时，狭窄通道可能因离散化误差而显得被阻塞，而体素过小时，规划速度则会大幅降低。球面图还以八叉树结构存储，以便快速进行最近邻查询，并缩短插入和删除所需的时间。

球体图被划分为由节点 $\mathcal{S}$ 组成的近似凸分段，从而形成一个拓扑图。凸性条件并不严格，因为门户之间的路径是第 III-B3 节所述的无碰撞段内路径，而非 [16] 中的直线。对于任意两个包含相互连通节点的区域，我们都会在拓扑图中添加一条连接，并存储球体交集半径最大的那对节点 $\nu_1 \in \sigma_1$，$\nu_2 \in \sigma_2$。如 [16] 所示，我们将这些分段之间的节点对称为*门户*。为实现高效的远距离规划，我们还会迭代计算并存储给定分段中任意两个门户之间的最优路径（根据式（1）），如图3所示。

B. 扩展*SphereMap*

所提出的结构在任务执行过程中以较低的 2 Hz 频率增量式构建。球体图逐步扩展，使球体覆盖环境中整个可达已知空间，并从球面图中剪除冗余节点，以加快计算。每次更新迭代中，仅修改以无人机为中心、尺寸预先设定的立方体内的地图部分。更新迭代只需以无人机的当前位置以及立方体空间的最新占据信息作为输入（即立方体内某一点是被占据、空闲还是未知）。任何可在机载设备上运行的占据地图构建方法都可用于提供此类信息，例如高效的占据栅格地图实现 OctoMap [13]或 UFOMap [14]。

为了能够快速查询任意给定点到最近障碍物的距离，我们根据附近空间立方体中的占据点（例如，占用栅格中占据体素的质心）构建 k-D树。为防止球体扩展到未知空间，我们还将自由空间与未知空间边界上的点加入 k-D树。在大多数

© IEEE，2022。已被 IEEE机器人与自动化快报（RA-L）接收。数字对象标识符：10.1109/LRA.2022.3195194

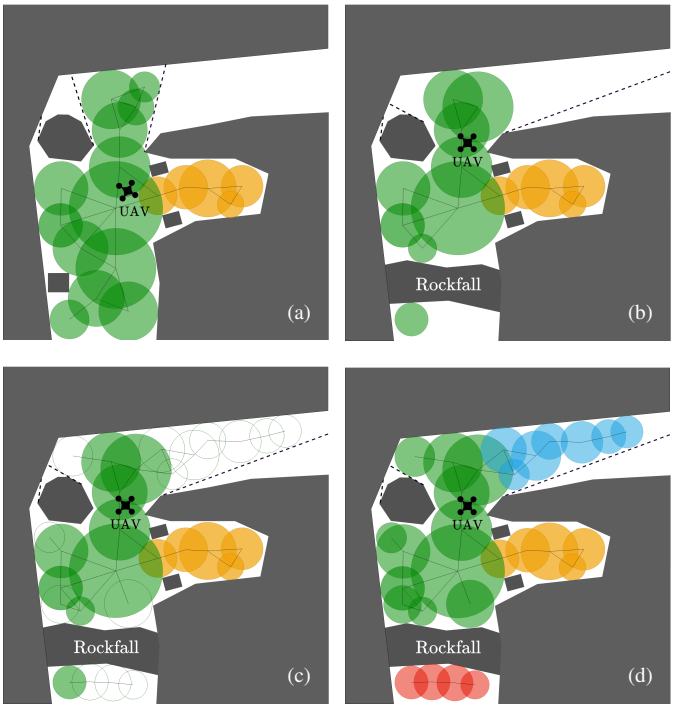

图 2：SphereMap更新算法一次迭代的概要。虚线表示已知自由空间与未知空间的边界。(a)：上一次迭代结束时的地图。(b)：更新球体半径，并剪枝不安全和冗余的节点。(c)：采样、创建新节点并将其连接到图中。(d)：将不连通的（ ）分段拆分为两个分段（，），扩展现有分段并生成一个新的分段（ ）。

占据地图中，查找这些边界点（称为前沿点）并非易事，且通常比构建 k-D树更耗时，但这对于安全路径规划是必要的，对探索也很有帮助。

SphereMap 通过近似恒定时间的迭代构建；随着给定区域内迭代次数增加，这些迭代会收敛到球体的最优分布。因此，球体偶尔可能无法覆盖整个可达空间，导致某些点无法通过 SphereMap 到达，或找到的路径远非最优。不过，这可以由任务控制处理。一次 SphereMap更新迭代（详见图 2）按以下顺序包含三个步骤：

1）重新计算与剪枝步骤

由于安全规划需要最新的占据信息，我们首先重新计算每个附近节点的半径 $\nu \in \mathcal{N}_{near}$，并更新半径发生变化的节点之间的连接，如算法1所示。半径的计算方法是：针对球面图更新部分中的每个节点，查询其到当前障碍物 k-D 树 K 中最近点的距离。由于我们假设环境是动态的，一些节点可能会变得不可通行。这可能是因为环境中的障碍物移动，或传感器数据存在噪声。这些节点会从图中剪除。

此外，由于发现了更多自由空间，一些节点可能会扩大，从而使其他节点变得过时。如果存在一个更大的节点，且其覆盖该节点体积的比例至少达到某个阈值，我们就将该节点标记为冗余节点。随后，所有冗余节点都会被剪除。

算法1： SphereMap 半径更新与剪枝步骤。仅更新节点 $\mathcal{N}_{near}$ 位于无人机周围包围盒内的节点。

过程重新计算球面映射
  输入：
    $\mathcal{G}$ ▷ SphereMap 图
    K ▷ 障碍物 k-D 树
    r_{min} ▷与障碍物的最小距离
  对于 $\nu \in \mathcal{N}_{near}$ **执行** ▷ 无人机周围立方体内待更新的节点
    $r_\nu \leftarrow$ **最近障碍物距离(**$\mathbf{p}_\nu$, K**)**
    如果 $r_\nu < r_{min}$ **then**
      $\mathcal{G} \leftarrow \mathcal{G} \setminus \nu$
    else
      更新节点连接(ν, $\mathcal{G}$**)**
  对于 $\nu \in \mathcal{N}_{near}$ **do**
    如果 冗余性判断(ν, $\mathcal{G}$**) 则**
      $\mathcal{G} \leftarrow \mathcal{G} \setminus \nu$

2) 扩展步长

在下一步中（如算法 2 所示），通过在无人机附近采样点，将新的非冗余球体添加到图中。采样点的数量 S 会极大地影响这一步的运行时间。采样大量点可以确保可达空间被球体填满，但需要更多计算资源。更稀疏的采样方法可以加快扩展步长，但在新发现的大范围区域中，可能导致部分可达空间未被球面图覆盖。在我们的实现中，我们选择对所用占据八叉树中的每个体素采样一个点，同时从无人机向随机方向发射射线，并沿射线采样点。

算法 2： SphereMap 扩展步长。在采样位置将不在现有节点内部的新节点添加到图中。

过程EXPANDSPHEREMAP
  输入：
    $\mathcal{G}$ ▷ SphereMap 图
    K ▷ 障碍物 k-D 树
  $\mathcal{P}_s \leftarrow$**近自由空间采样点(**$\mathbf{p}_{uav}$**)**
  对 $\mathbf{p}_s \in \mathcal{P}_s$ **执行**
    $\nu_{new} \leftarrow$**初始化节点()**
    $\mathbf{p}_{\nu_{new}} \leftarrow \mathbf{p}_s$
    $r_{\nu_{new}} \leftarrow$ **最近障碍物距离(**$\mathbf{p}_s$, K**)**
    $\sigma_{\nu_{new}} \leftarrow \emptyset$
    如果 $r_{\nu_{new}} \geq r_{min}$ **且非冗余性判断(**$\nu_{new}, \mathcal{G}$**) 则**
      $\mathcal{G} \leftarrow \mathcal{G} \cup \nu_{new}$
      更新节点连接($\nu_{new}, \mathcal{G}$**)**
      对 $\nu' \in$ **相连节点(**ν_{new}, $\mathcal{G}$**) 执行**
        如果 isRedundant(ν', $\mathcal{G}$**) 则**
          $\mathcal{G} \leftarrow \mathcal{G} \setminus \nu'$

3) 分割步骤

在最后一步（如算法 3 所示）中，我们更新地图的分割。图中的每个节点 ν，除当前扩展步长中新创建的节点外，都属于某个区域 σ_ν。在此步骤中，我们首先将每个新节点分配到一个分段。具体做法是，先在球面图上使用泛洪填充扩展现有区域，并将区域的最大包围球半径限制为 r_{exp}。然后，我们遍历其余尚未分配分段的新添加节点，并以这些节点为起点初始化并扩展新的分段（同样受 r_{exp} 限制），直到所有节点都属于某个分段。

由于该方法能够对图进行非常细粒度的分割，我们尝试将此步骤中所有经过修改的区域与其相邻区域合并。只有在合并后的区域不超过包围球半径 r_{merge}，且各分段的中心彼此可见时，才允许进行合并。通过使用这些条件

并将 r_{exp} 设置得远低于 r_{merge}，我们可以以较低的计算开销实现大致的凸分割。

此外，对于每次迭代中发生修改的每个分段，我们都会根据准则 (1) 重新计算并存储其中每对门户之间的最优路径$\mathcal{C}_{(p_1,p_2)}$ （p_1，p_2）。分割步骤或更新与剪枝步骤都可能导致分段发生修改，这意味着其中一些节点被更改，或节点之间的一些边被删除。下文将这些路径称为段内路径；它们使用第 IV 节中描述的路径代价和启发式函数，通过A*算法计算，然后存入缓存，以便快速进行按需长距离路径规划。

算法3： SphereMap 分割步骤。门户之间的段内路径存储在缓存中 $\mathcal{C}$，以便加快规划。

过程分割SPHEREMAP
  输入：
    $\mathcal{S}_{\mathcal{G}}$ ▷SphereMap 图分段
    r_{exp} ▷ 分段扩展边界球的最大半径
    r_{merge} ▷ 分段合并边界球的最大半径
  $\mathcal{S}_{near} \leftarrow$**获取邻近线段(**$\mathcal{S}_{\mathcal{G}}$**)**
  $\mathcal{S}_{near} \leftarrow$**拆分不连通线段(**$\mathcal{S}_{near}$**)**
  对 $\sigma \in \mathcal{S}_{near}$ **执行** ▷ 尝试扩展现有线段
    $\sigma \leftarrow$ **扩展线段(**σ, r_{exp}**)**
    重新计算线段门户(σ**)**
  对 $\nu \in \mathcal{N}_{near}, \sigma_\nu = \emptyset$ **执行** ▷ 扩展新分段
    $\sigma_{new} \leftarrow$**初始化线段(**ν**)**
    $\sigma_{new} \leftarrow$**扩展线段(**σ_{new}**)**
    $\mathcal{S}_{near} \leftarrow \mathcal{S}_{near} \cup \sigma_{new}$
    重新计算线段门户(σ_{new}**)**
  对于$(\sigma_1,\ \sigma_2) \in$ **getAdjacentSegmentPairs(**$\mathcal{S}_{near}$**) 执行** ▷ 合并线段
    如果可合并线段($\sigma_1,\ \sigma_2,\ r_{merge}$**) 则**
      $\sigma_1 \leftarrow$ **合并线段(**$\sigma_1,\ \sigma_2$**)**
      $\mathcal{S}_{near} \leftarrow \mathcal{S}_{near} \setminus \sigma_2$
  对于 $\sigma \in \mathcal{S}_{near}$ **执行** ▷ 重新计算段内路径
    如果线段是否已更改(σ**)，则**
      对于 $(p_1,\ p_2) \in$ **获取线段门户(**σ**)，执行**
        $\mathcal{C}_{(p_1,p_2)} \leftarrow$ **计算线段内路径(**$\sigma,\ p_1,\ p_2$**)**

C. 轻量级地图创建与共享

球面图分割的另一项关键特性是，可以用凸三维形状来表示近似凸的空间区域（在我们的实现中，我们使用由宽度、高度、深度和 z 轴旋转角确定的 4DOF 边界框，如图 7 所示）。在 SphereMap 的每次更新迭代之后，对于任何发生修改的邻近分段，我们都会计算并存储该分段球体的最佳拟合凸形状包络。因此，环境的拓扑体积图表示（LTVMap）可以随时提取。LTVMap 图中的每个顶点表示一个分段包络的位姿和尺寸；对于任意两个相连的分段，图中都会添加一条边。

这种表示比 SphereMap 本身和底层占据栅格地图都更轻量，因此可用于无法共享完整信息的低带宽场景。对于搜索与救援任务，可以轻松地为每个区域附加额外信息（例如无人机摄像机覆盖的表面百分比，或给定区域内前沿点的信息）。在任务期间，这些信息随后可以与其他机器人共享以便协作，也可以与人工操作员共享；这种地图表示能让他们清晰了解环境和任务进展 [22]。

D. 计算复杂度

每次更新迭代的复杂度主要取决于两个因素：确定给定点周围球体的半径，其复杂度为 $\mathcal{O}(\log|K|)$；以及检查最近球体是否可能相交，其复杂度为 $\mathcal{O}(n_c \log|\mathcal{N}_{near}|)$。其中，$n_c$ 是需要检查的最近球体的平均数量，具体取决于球体密度以及更新包围盒中的最大球体。因此，整个更新的复杂度为$\mathcal{O}((|\mathcal{N}_{near}|+|S|)(n_c \log\ |\mathcal{N}_{near}|+\log|K|))$。由于只更新无人机周围的小范围区域，更新时间不会随 SphereMap 中球体的总数增加，而取决于局部区域的球体密度。通过对球面图进行剪枝，该方法使 n_c 和 $|\mathcal{N}_{near}|$ 保持在合理范围内，从而也能维持较低的运行时间。

IV. SphereMap中的规划

对于使用优化准则 (1) 在图或网格上规划路径，需要先将该准则离散化。假设路径上两个相邻点$\mathbf{p}_1$,$\mathbf{p}_2$ 之间有一段，其到最近障碍物的距离分别为 $r_{\mathbf{p}_1}$,$r_{\mathbf{p}_2}$。显然，长度增量可写为

$$\Delta L_{\mathbf{p}_1\mathbf{p}_2} = \|\mathbf{p}_1 - \mathbf{p}_2\|, \tag{5}$$

但如果我们只能通过 k-D树获取障碍物距离，那么安全增量 ΔZ 的计算就比较棘手。通过将增量定义为以下形式来近似该积分：

$$\Delta Z_{\mathbf{p}_1\mathbf{p}_2} = \xi\left[\max\left(0, d_{max} - \frac{r_{\mathbf{p}_1} + r_{\mathbf{p}_2}}{2}\right)\right]^2 \|\mathbf{p}_1 - \mathbf{p}_2\|. \tag{6}$$

规划过程中，我们还会确保点$\mathbf{p}_1$ 与$\mathbf{p}_2$ 之间的连线上不存在比最小允许距离 r_{min}更靠近障碍物的点（通过计算$\mathbf{p}_1$和$\mathbf{p}_2$ 周围半径分别为 $r_{\mathbf{p}_1}$ 和$r_{\mathbf{p}_2}$的球体的交集大小）。我们使用 A*算法，两个点之间的转移代价为

$$g(\mathbf{p}_1, \mathbf{p}_2) = \Delta L_{\mathbf{p}_1\mathbf{p}_2} + \Delta Z_{\mathbf{p}_1\mathbf{p}_2} \tag{7}$$

以及一个启发式函数

$$h(\mathbf{p}_2) = \|\mathbf{p}_2 - \mathbf{p}_{goal}\| \tag{8}$$

用于 SphereMap 中的规划。

对门户之间的最优路径进行分割和缓存的主要优势在于，每当查询点 A 与点 B 之间的路径时，我们只需计算从起点和终点到其所在分段的门户、跨越球面图的路径。其余路径则可以利用缓存的段内路径进行规划，如图3所示。

如表 I 所示，与在整个球面图上计算路径相比，这种技术可将计算速度提升两个数量级；但生成的路径必须经过门户，而最优路径未必会经过这些门户。

© IEEE，2022。已被 IEEE机器人与自动化快报（RA-L）接收。数字对象标识符：10.1109/LRA.2022.3195194

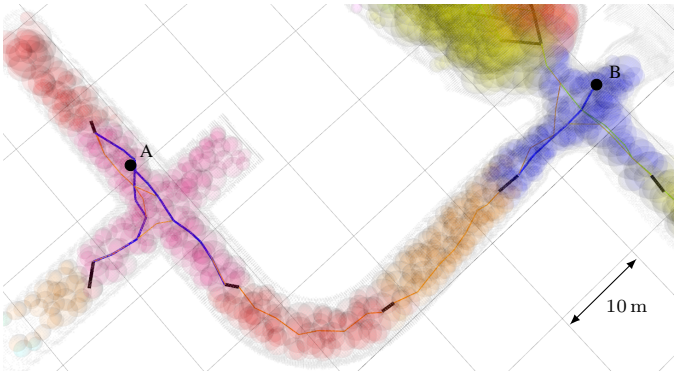

图3：所提出的规划方法示意图。查询从 A 到 B 的路径时，只需计算少量跨越球面图的路径（），即从 A 到其所在分段的门户（）的路径；到达门户后，再计算从 B 所在分段的门户到 B 的路径。其余计算则使用门户之间缓存的段内路径（细彩线）。

V. 实验

本节旨在多个仿真环境中验证所提出的方法，并将 SphereMap 与最先进的方法进行比较。此外，我们分析 SphereMap 生长步骤的运行时间，并说明由 SphereMap 生成的轻量级地图远小于二进制占用八叉树，即使其分辨率较低，同时仍保留环境连通性以及各区域的尺寸和几何形状细节。我们还展示了所提出的方法在地下和室外环境中部署于真实无人机平台上的情况。

A. 规划分析

为展示 SphereMap 快速生成优化准则（1）的路径的能力，我们在仿真环境中进行了两项实验，并将所提出的规划方法与以下最先进的规划方法进行比较：

- 基于占用栅格的 *A**算法：一种简单的路径规划方案是在具有固定体素尺寸的占用栅格上运行 A*算法。使用第 IV 节中定义的转移代价和启发式函数，并采用较小的体素尺寸，这种方法能够找到比 SphereMap 方法更接近准则（1）所定义最优解的路径；但如表 I 所示，由于计算时间过长，该方法无法用于长距离规划。
- 通过占用栅格使用 *RRT**：此外，我们将所提出的规划方法与一种流行的规划算法——渐近最优 RRT* [18]——进行了比较。用于比较的路径是初始解，即找到解后不再继续进行树生长以进一步优化路径；继续树生长虽可生成更接近最优解的结果，但会增加计算时间。

首先，我们比较了无人机需要规划到大量目标点的路径，且目标点最远可达无人机 300 米外的情况下，各种规划方法的表现。我们使用一台标准笔记本电脑，以 0.2 米的分辨率构建 SphereMap 占据八叉树，并以 0.8 米的分辨率进行基于栅格的 A*算法计算。基于安全性的规划参数设为 $\xi = 7$，$d_{max} = 2$米 $r_{min} = 0.8$ 米。该比较结果见表 I，找到的路径如图 4 (a) 所示。结果显而易见，现有最先进方法在这种情况下速度过慢，无法满足本文提出的要求。

在第二个规划实验中，我们分析了各比较方法找到的通往单一目标的路径长度和风险（定义见式 (1)）。为更清晰地展示不同方法之间的差异，基于网格的 A* 使用了 0.4 米的分辨率。实验采用了与第一次实验相同的基于安全性的规划参数。分析结果见图 4 (b) 和表 II。可以看出，我们的方法远胜于当前最先进的方法，同时仍接近基于网格的 A* 找到的最优解。

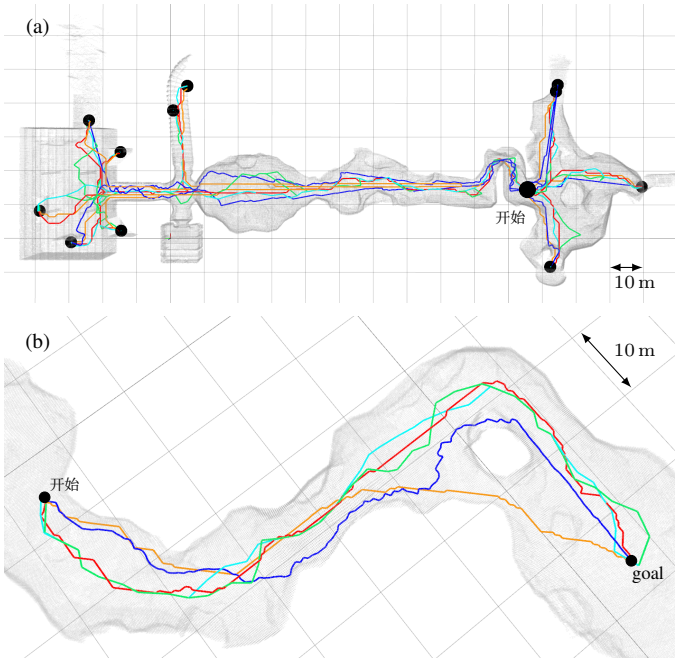

图 4：使用以下方法规划至多个潜在探索目标（a）和单个目标（b）的路径可视化：使用缓存的段内路径，在 SphereMap 上由 A*算法生成的路径（）；通过整个球面图规划的路径（）；使用代价函数 (1) 的基于网格的 A*（）；在体素分辨率为 $r = 0.8$m（a）和 $r = 0.4$m（b）的占用栅格上，以路径长度作为路径代价的基于网格的 A*（）；RRT* 路径（）。

规划器	时间 [毫秒]	找到的路径	平均代价 [-]
网格 A*	59524	11/11	270.36
仅长度网格 A*	58795	11/11	477.39
RRT*	96830	6/11	385.05
SphereMap	1868	11/11	**257.77**
SphereMap 缓存	**22**	11/11	322.73

表 I：图 4 (a) 所示仿真环境中的多目标路径规划实验定量结果。每条路径的成本通过对离散化转移代价 (7) 求和计算得出。计算时间为所有找到的路径所耗时间之和。即使设置了 10 s 的规划超时，RRT* 仍未能找到其中 5 条路径。与需要数十秒才能找到路径的最先进方法相比，在 SphereMap 的整个球面图上进行规划仅需 1.8 s；若使用段内路径缓存，则仅需 22 ms 即可找到这些路径。

规划器	时间 [毫秒]	长度 [米]	风险 [-]	成本 [-]
网格 A*	28064	101.11	33.88	134.99
仅长度网格 A*	15376	80.33	376.21	456.54
RRT*	544	96.79	413.61	510.40
SphereMap	56	**104.14**	**45.98**	**150.12**
SphereMap 缓存	**0.496**	110.46	49.37	159.83

表 II：图 4 (b) 所示模拟洞穴环境中的单目标规划实验定量结果。每条路径的成本通过对离散化转移代价 (7) 求和计算得出。我们的方法比 RRT* 找到初始解快 3 个数量级，比分辨率为 0.4 m 的基于网格的 A* 快 5 个数量级。仅针对路径长度进行 A* 优化，自然会找到最短路径；但如图 4 (b) 所示，该路径会经过风险更高的狭窄通道。

B. 迭代运行时间分析

为了展示构建 SphereMap 所需的计算负载较低，我们在一次模拟的探索任务中进行了实验。该任务穿越一个规模与其他实验中所示环境相同的广阔洞穴环境，并测量了 SphereMap 生长过程各个部分的运行时间。

该实验的结果见图 5。实验表明，SphereMap 的更新迭代平均计算时间约为 150 毫秒。需要注意的是，整体生长运行时间很大程度上取决于扩展步长，并且可以通过采用更稀疏的采样方法来加快。

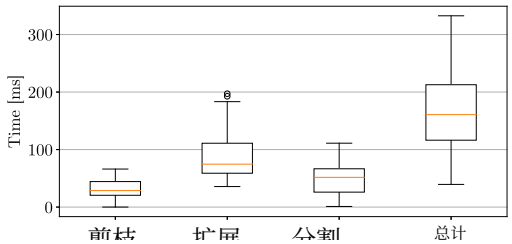

图 5：在大型洞穴环境中的探索任务期间，SphereMap 一次生长迭代各个部分运行时间的箱线图。离群计算时间通常出现在无人机探视拐角并发现一大片自由空间区域时。

C. 地图压缩分析

为了展示由 SphereMap 生成的 LTVMap 的实用性和轻量化特点，我们在一个绵延数百米的模拟洞穴环境中进行了实验，并将 LTVMap 与构建 SphereMap 所用的原始二进制占据八叉树（0.2米分辨率）进行了比较。我们还比较了 LTVMap 与通过将原始八叉树降至 1米分辨率而生成的占据八叉树的大小。整个任务期间这三种表示的大小变化如图 6 所示。实验结束时的 LTVMap 和原始地图如图 7 所示。

根据图 6 中，降低分辨率后的占据八叉树似乎是进行体积信息共享的不错选择。然而，降低分辨率可能会导致狭窄通道在生成的地图中被阻塞，使其他机器人错误地判断环境的连通性。另一方面，LTVMap 在保留连通性信息的同时，也比降低分辨率后的地图更小。

所示 LTVMap 的一个缺点是，由于我们使用 4DOF 边界框进行区域形状近似，因此这些包围盒中有很大一部分空间在原始占据八叉树中并非空闲空间。采用不同的边界形状（如多边形或 6DOF 边界框）可以减少误分类空间，因为它们能更准确地近似分段形状，但代价是计算复杂度和通信带宽更高。尽管如此，对于分散式协同规划，我们发现 4DOF 边界框足以应对 DARPA SubT 挑战赛 [22]。

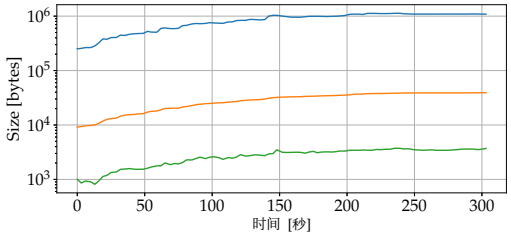

图 6：V-C 节所述实验中，LTVMap（）、SphereMap 所用的二进制占据八叉树（）（由 OctoMap 库实现[13]），以及通过将原始八叉树降至 1米分辨率而生成的八叉树（）的大小变化。

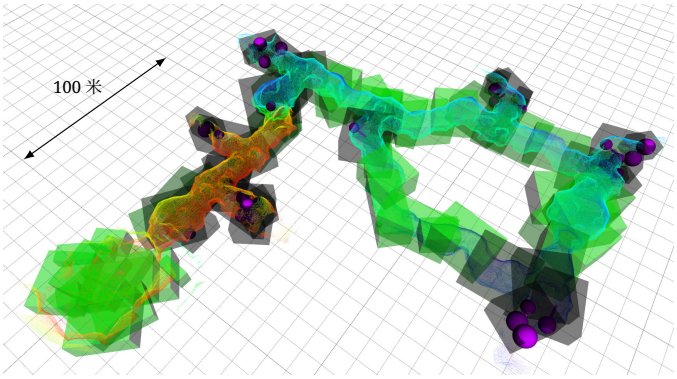

图 7：实验 V-C 中所述实验里的原始占据八叉树（以编码点高度的热力图为点云着色）以及根据该八叉树生成的 SphereMap 构建的 LTVMap 的可视化。紫色球体表示添加到 LTVMap 中的潜在探索目标，方块的颜色表示它们与未探索空间的接近程度。

D. 真实环境部署

本文所述的 SphereMap 成功应用于 DARPA SubT 挑战赛决赛（图 1），用于机器人之间以及机器人与操作员之间的导航、协同探索、目标路径规划和地图共享。已部署系统中的控制和状态估计部分详见 [4]。在 DARPA SubT 挑战赛决赛的虚拟赛道中，也使用了形式较为粗略的 SphereMap（未对节点进行剪枝）；我们的 CTU-CRAS-NORLAB 团队在比赛中获得了 2nd 名 [22]。

最后，我们还在一处大型室外环境中对 SphereMap 进行了测试。在那里，无人机自主探索了一栋房屋周围的区域，该房屋四周灌木丛生；无人机穿过一座桥梁，飞入一片广阔的开阔地，并安全返航。只需进行少量微调（仅降低采样点数量，以补偿庞大的自由空间体积），

© IEEE，2022。已被 IEEE机器人与自动化快报（RA-L）接收。数字对象标识符：10.1109/LRA.2022.3195194

在配备英特尔酷睿 i7 处理器的真实无人机平台上，除了运行 SphereMap 之外，还运行着同步定位与建图、状态估计、视觉检测和控制；SphereMap 的总增长运行时间平均约为 200 毫秒。这表明，即使在大型室外区域，所提出的方法也能在常见硬件上使用。室外探索实验以及虚拟和实体 DARPA SubT 决赛的视频输出可在补充材料中查看。

VI. 结论

本文提出了一种方法，可在飞行过程中灵活构建带有缓存的段内路径的分段球体图，用于快速安全感知路径规划，并按需生成已探索环境的轻量级拓扑体积图表示。所提出的方法无需根据对环境的先验知识对地图分辨率进行微调。因此，为了实现对未知非结构化环境的安全、灵活的多机器人探索，该方法只需获取无人机附近的占据空间、自由空间和未知空间地图；每次地图更新时，机载传感器都能高效地提供此类地图。基于所提出的 SphereMap 进行规划，旨在寻找兼顾安全性和长度的路径。由该结构生成的拓扑体积 LTVMap，即使在降低分辨率的情况下，所占带宽也远小于占据栅格地图，同时仍保留环境连通性及其大致形状的信息。这对于地下环境的实时多机器人探索至关重要，因为机器人之间的可靠通信是其瓶颈。

在使用扫描洞穴、矿井和工业综合体的世界模型进行的大量仿真和统计分析中，所提出的方法还在多次探索任务期间搭载于真实无人机平台上得到了验证；与基于网格的 A*或 RRT*等最先进的规划算法相比，该方法寻找路径的速度快了3个数量级，能够使机器人在环境中安全导航，并支持共享轻量级地图的无人机之间进行协作。采用所提出方法的整个无人机系统还与由入选 DARPA SubT 挑战赛决赛的顶尖机器人团队组成的参赛队伍进行了对比。该挑战赛为在洞穴、矿井和地铁等高难度真实环境中比较不同解决方案提供了可靠条件。所提出的方法成功地在系统赛道中引导一支无人机队伍穿越充满尘土、狭窄且路径动态受阻的隧道，并且是部署于虚拟赛道系统中的关键组成部分，最终获得了第 2nd 名。

参考文献

[1] J. Delmerico 等，“救援机器人学的现状与未来展望”，《野外机器人学》期刊，第36卷，第7期，第1171–1191页，2019年。[2] R. Carvalho 等，“一种基于无人机的框架，用于采矿业带式输送机的半自动热成像巡检”，Sensors，第20卷，第8期，第2243页，2020年。[3] P. Petr´a˘cek等，“Dronument：在大型历史古迹的黑暗区域可靠部署微型飞行器的系统”，IEEE机器人与自动化快报，第5卷，第2期，第2078–2085页，2020年4月。[4] T. Baca等，“MRS无人机系统：推动自主无人机可复现研究、现实环境部署和教育的前沿”，智能与机器人系统期刊，第102卷，第26期，第1–28页，2021年5月。[5] J. L. Sanchez-Lopez等，“Aerostack：空中机器人学的架构与开源软件框架”，发表于国际无人机系统会议（ICUAS），2016年，第332–341页。[6] V. Kr´atk´y等，“由无人飞行器编队利用分布式照明进行自主航拍”，IEEE机器人与自动化快报，第6卷，第4期，第7580–7587页，2021年10月。[7] V. Spurny等，“无人机在建筑物内自主消防”，IEEE Access，第9卷，第15 872–15 890页，2021年1月。[8] J. Quenzel等，“MBZIRC 2020中无人机-无人地面车辆编队自主灭火”，发表于国际无人机系统会议（ICUAS），2021年，第934–941页。[9] T. Baca等，“利用微型单探测器康普顿事件相机为微型飞行器定位伽马辐射源”，发表于国际无人机系统会议（ICUAS），2021年6月，第338–346页。[10] M. Petrl´ík等，“适用于受限环境作业的鲁棒无人机系统”，IEEE机器人与自动化快报，第5卷，第2期，第2169–2176页，2020年4月。[11] V. Kr´atk´y等，“用于快速探索大型复杂室内环境的自主无人机系统”，《野外机器人学》期刊，第38卷，第8期，第1036–1058页，2021年5月。[12] P. Petracek等，“无人机对洞穴环境的大规模探索”，IEEE机器人与自动化快报，第6卷，第4期，第7596–7603页，2021年10月。[13] A. Hornung等，“OctoMap：一种基于八叉树的高效概率三维建图框架”，Autonomous Robots，第34卷，第3期，第189–206页，2013年。[14] D. Duberg等，“UFOMap：一种兼顾未知区域的高效概率三维建图框架”，IEEE机器人与自动化快报，第5卷，第4期，第6411–6418页，2020年。[15] H. Oleynikova等，“用于微型飞行器规划的稀疏三维拓扑图”，IEEE/RSJ智能机器人与系统国际会议（IROS），第1–9页，2018年。[16] F. Bl¨ochliger等，“Topomap：基于视觉SLAM地图的拓扑建图与导航”，IEEE机器人与自动化国际会议（ICRA），第1–9页，2018年。[17] T. Dang等，“用于地下环境自主机器人探索的基于图的路径规划”，IEEE/RSJ智能机器人与系统国际会议（IROS），第3105–3112页，2019年。[18] S. Karaman等，“用于最优运动规划的基于采样的算法”，《国际机器人研究杂志》，第30卷，第7期，第846–894页，2011年。[19] T. Dang等，“用于地下环境自主机器人探索的基于图的路径规划”，IEEE/RSJ智能机器人与系统国际会议（IROS），第3105–3112页，2019年。[20] H. Cover等，“稀疏切向网络（SPARTAN）：微型飞行器运动规划”，IEEE机器人与自动化国际会议（ICRA），第2820–2825页，2013年。[21] M. Collins等，“利用在线稀疏拓扑图，为无人机在未知环境中的高速飞行进行高效规划”，发表于IEEE机器人与自动化国际会议（ICRA），2020年，会议论文，第11 450–11 456页。[22] M. Petrlik等，“UAVs Beneath the Surface：DARPA SubT地下搜索与救援中的协同自主能力”，预印本 arXiv:2206.08185，2022年，投稿至《野外机器人学》特刊：DARPA地下挑战赛。[23] P. De Petris等，“受限环境中的鲁棒耐碰撞导航”，发表于IEEE机器人与自动化国际会议（ICRA），2021年，第2286–2292页。[24] H. Oleynikova等，“Voxblox：用于机载微型飞行器规划的增量式三维欧氏带符号距离场”，IEEE/RSJ智能机器人与系统国际会议（IROS），第1366–1373页，2017年。