arXiv:2304.00342v1 [cs.RO] 1 Apr 2023

多智能体基于采样的运动规划分解

Alessandro Zanardi[1*]、Pietro Zullo[1*]、Andrea Censi[1]、Emilio Frazzoli[1]

摘要——**现代机器人学通常涉及多个在共享环境中运行的具身智能体。在这些情况下，路径规划比单智能体场景复杂得多。尽管可以使用标准的基于采样的算法（SBA）在机器人的联合空间中搜索解，但随着智能体数量增加，这种方法很快会变得计算上难以处理。为了解决这一问题，我们将因子分解的概念融入基于采样的算法，只需对现有方法进行最少的修改。在搜索解的过程中，一旦通过分解启发式确认不同智能体子集未来的解彼此独立，我们就可以将这些子集解耦（即因子分解），使其进入相互独立的低维搜索空间。因此，我们逐步构建一个精简的超图，其中某些（超）边将智能体划分到彼此独立的子图中。在最佳情况下，这种方法能够将搜索空间维度随智能体数量增长的复杂度从指数级降至线性级。在平均情况下，找到高质量解所需的样本更少，同时保持 SBA 的最优性、完备性和随时性。我们介绍了一种通用的因子分解采样算法实现，推导了概率路图*在样本复杂度方面的解析增益，并展示了RRG的实证结果。**

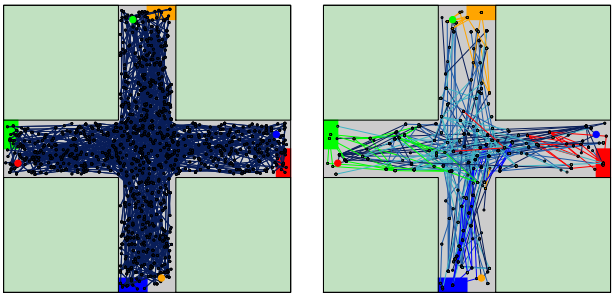


图 1. 一个包含 4 个智能体的 MA-MMP 示例。每个智能体从一条走廊出发，必须到达另一条走廊中的目标区域。左侧为在联合配置空间中运行 RRG 的结果。右侧则采用其因子化 形式（FactRRG）：随着参与者彼此独立（即可以确定从该点开始，它们未来的解相互独立），将其因子化到各自独立的低维图中。可以直观看出， FactRRG 收敛到更精简的图，平均所需迭代次数显著减少，同时代价与原解相当。

一、引言

多智能体运动规划问题（MA-MMP）因其有望解决自动驾驶和物流等领域中的复杂现实问题，而日益受到重视。在共享环境中进行规划的一个关键方面，是找到高效且稳健的协调方法，使智能体能够像人类每天在道路上一样，在其他智能体共存的情况下顺畅通行。

多个智能体在避免碰撞的同时追求各自的目标，因此多智能体运动规划问题（MA-MMP）自然成为这一挑战的数学表述。有趣的是，许多在单智能体场景中表现出色的运动规划方法，在扩展到多智能体情形时却难以有效扩展 [1], [2]。普遍认为，一个主要的制约因素是处理高维空间中搜索问题的能力 [3]。

基于采样的算法在相对低维的搜索空间中解决规划问题方面取得了巨大成功 [4]。然而，它们也无法避免多智能体系统带来的维数灾难。搜索空间会随着智能体数量的增加而呈指数级增长，使得即便只有少量智能体，该问题也难以求解。

人们已做出许多尝试来克服这些局限性。其中一些方法引入启发式方法，将 MA-MMP 转化为一系列单智能体问题，例如，通过为智能体设定特定顺序 [5], [6] ，或开发局部碰撞消解方法 [7], [8]。这些方法往往会牺牲最优性或完备性，尤其是在智能体需要穿行于狭小、拥挤的环境中时 [9]。

贡献声明：在这项工作中，我们展示了如何将 [10] 中为动态博弈提出的因子分解概念应用于多智能体运动规划问题的基于采样的算法。随着解的求得，我们提出的方法能够自然地将参与者从联合配置空间解耦到低维搜索空间中——在极限情况下，每个智能体对应一个搜索空间。我们方法的一大优势是可应用于大多数常见的现成先进规划器。此外，它在利用为单智能体情形开发的所有性质（最优性、完备性等）和工具（例如，扩展到动力系统）的同时，还可能带来显著的计算收益（具体取决于多智能体运动规划问题本身可因子分解的程度）。

A. 相关工作

为解决多智能体运动规划问题，我们确定了三种主要方法：集中式、分布式和混合式方法。集中式方法将所有智能体的并集视为一个系统 [11]–[13]。这些方法在联合空间中进行规划，通常保留与相应单智能体方法相同的完备性和最优性性质。然而，这会导致搜索空间维度呈指数级增长，使计算很快变得难以处理 [1]。例如， [11], 将快速探索随机树 RRT* 加以改造，用于解决多智能体运动规划问题。

去中心化方法尝试通过先求解个体的单体问题，随后组合其解 [5]–[8]，来求解 MA-MMP。自然地，

*贡献相同。[1]A. Zanardi、P. Zullo、A. Censi 和 E. Frazzoli 任职于瑞士苏黎世联邦理工学院动态系统与控制研究所，电子邮箱：azanardi@ethz.ch。本研究获瑞士国家科学基金会支持，资助项目为国家自动化研究能力中心（NCCR Automation），资助协议编号为 51NF40 180545。

第一步忽略了智能体之间的耦合，而在组合个体解时，还需要额外步骤来解决冲突。已有方法包括在智能体之间强制实施“规划层次结构” [5], [6]，以及在求解单体问题后进行局部冲突消解或重新规划 [7], [8]。例如， [5] 按顺序构建路径，将先前的智能体视为移动障碍物。在 [7] 中，每个智能体的路径分别求得；如果两个智能体在某一点发生碰撞，则重新规划其余路线。尽管计算效率更高，但与其集中式对应方法相比，这些方法通常缺乏同等的最优性和完备性保证。

最后，混合式方法 旨在结合集中式 方法和分布式 方法各自的优点。与我们在此提出的方法类似，这类算法通常利用主要针对单个智能体各自问题的数据结构和算法，但在出现冲突时可以切换到联合表示。 [8],提出的 dRRT* 就是一个例子：这是一种渐近最优的基于采样的算法，无需显式表示联合运动图，便可扩展至大量智能体。一个关键思路通常是识别较小的独立子问题，并分别求解。例如， [14] 首先为每个智能体单独计算最优路径。如果发现各智能体的路径存在冲突，就将发生冲突的智能体编为一组，并将组内智能体视为一个整体，重新计算解。类似地， [9]将单个策略之间的碰撞视为彼此独立（更准确地说，是彼此依赖）的指标，并将这一概念纳入路径构建阶段，从而形成一个可适用于快速探索随机树和概率路图等增量式规划器的框架。

II. 预备知识

A. 多智能体运动规划问题

我们首先回顾与 [1, 第 7 章2]类似的多智能体运动规划问题（MA-MMP）的定义。设有一个包含 $|\mathcal{A}|$个智能体 $\mathcal{A}$ 的有限集。每个智能体都有一个维度为 $d^i \in \mathbb{N}$ 的关联配置空间 $\mathcal{X}^i$ ；为简便起见，我们假设所有智能体的配置空间 d^i 相同。因此，智能体的联合配置空间为 $\mathcal{X} = [0,1]^{d^i|\mathcal{A}|}$ ，其对应的一个子集是障碍区域 $\mathcal{X}_{\text{obs}}$。其补集定义为差集$\mathcal{X}_{\text{free}} = \text{cl}(\mathcal{X}/\mathcal{X}_{\text{obs}})$的集合闭包，即无障碍区域 。每个智能体都有一个关联的初始状态 $x^i_{\text{init}} \in \mathcal{X}_{\text{free}}$和一个目标区域 $\mathcal{X}^i{}_{\text{目标}} \subseteq \mathcal{X}$。请注意，我们借用博弈论记法来区分智能体某个（子）群体特有的量与联合量。上标用于指明所关注的智能体；没有上标则表示联合量。例如，第 i 个智能体的配置空间为 $\mathcal{X}^i$，而 $\mathcal{X}$ 表示联合配置空间（ $= \Pi_{i\in\mathcal{A}}\mathcal{X}^i$ ）。

MA-MMP 随后归结为：为每个智能体寻找一条无碰撞路径，使其从给定的初始配置到达各自的目标。更正式地，设第 i 个智能体的路径是形如 $\sigma^i\colon [0,1] \to \mathcal{X}^i$ 的函数。为避免病态情况，我们仅考虑具有连续有界变差 [15]的路径。

定义 1 （多智能体运动规划问题）。设 $\mathcal{A}$ 是一个智能体有限集，且有三元组 $\left\langle \mathcal{X}_{\text{free}}、\ \{\mathcal{X}^i_{\text{goal}}\}_{i\in\mathcal{A}}、\ \{x^i_{\text{init}}\}_{i\in\mathcal{A}} \right\rangle$。我们将多智能体运动规划问题（*Multi-Agent Motion Planning Problem*，简称 $\mathcal{P}(x^{\mathcal{A}}_{\text{init}})$ ）定义为寻找 $\sigma = \{\sigma^i\}_{i\in\mathcal{A}}$ 使得：

1) 每个智能体的路径均可行

$$\sigma^i(0) = x^i_{\text{init}} \ \wedge\ \sigma^i(1) \in \text{cl}(\mathcal{X}^i_{\text{goal}}) \quad \forall i \in \mathcal{A}; \tag{1}$$

2) 路径相对于障碍物无碰撞：

$$\sigma^i(\tau) \in \mathcal{X}_{\text{free}} \quad \forall i \in \mathcal{A}\ \forall \tau \in [0,1]; \tag{2}$$

3) 路径相对于其他智能体无碰撞 *ts:*

$$\text{coll}(\sigma^i(\tau), \sigma^j(\tau)) = \textsc{False} \quad \forall_{i\neq j} i,j \in \mathcal{A};\ \forall \tau \in [0,1]. \tag{3}$$

如果不存在这样的路径集合，则报告失败。我们用 $\textsc{Sol}(\mathcal{P}(x^{\mathcal{A}}_{\text{init}}))$ 表示解路径集合。

通常，我们不仅希望获得一个 解，还希望根据某个代价函数获得最优路径，由此定义：

定义 2 (最优 MA-MMP). 给定一个多智能体运动规划问题 *(MA-MMP)* $\mathcal{P}(x^{\mathcal{A}}_{\text{init}})$ 及代价函数 $c\colon \Pi_{i\in\mathcal{A}}\Sigma^i \to \mathbb{R}_{\geq 0}$，若解 σ^* 满足以下条件，则该解为最优解

$$\sigma^* \in \underset{\sigma\in\textsc{Sol}(\mathcal{P}(x^{\mathcal{A}}_{\text{init}}))}{\arg\min} c(\sigma). \tag{4}$$

若不存在这样的路径，则报告失败。我们将ϵ-最优路径的集合记为$\textsc{Sol}^{\epsilon}(\mathcal{P}(x^{\mathcal{A}}_{\text{init}}))$。
That is, $\{\sigma \mid (c(\sigma) - c(\sigma^*))/c(\sigma^*) \leq \epsilon\}$ *) for* $\epsilon \geq 0$.

本文将各智能体的个体代价之和（即社会代价）作为 MA-MMP 的代价函数，即 $c(\sigma) = \sum_{i\in\mathcal{A}} c^i(\sigma)$。此外，我们将智能体的个体代价函数定义为路径长度 ，其形式为 $c^i : \Sigma^i \to \mathbb{R}_{\geq 0}$。这一选择简化了本文的表述，并将智能体之间的交互限制为仅发生碰撞。尽管如此，在更一般的情况下，智能体之间也可能在代价层面存在耦合（例如安全距离惩罚） [16]。只要 MA-MMP 的代价函数可按智能体分离，本文提出的概念仍然适用（可能只需稍作调整）。

最后，为确保 SBA 的适定性 [15]，我们假设 c, c^i 是单调的，即 $\forall\, \sigma, \sigma' \in \Sigma,\ c(\sigma) \leq c(\sigma \cup \sigma')$；并且是有界的，即存在 $\exists\, k_c$ 使得 $c(\sigma) \leq k_c TV(\sigma)\ \ \forall \sigma \in \Sigma$ ，其中 k_c 是一个正实数常量， TV 表示总变差。在本文其余部分中，我们将采用路径长度 作为当前问题的代价函数。

B. 基于采样的运动规划算法

基于采样的算法（SBA） [1, 第 187] 页已成为机器人学中广泛使用的运动规划策略。SBA 通过对智能体的配置空间进行随机采样，并构建图或路图来工作，这些图或路图表示规划环境中的连通性和可行路径。在本文中，我们考虑算法1所示形式的 SBA。需要注意的是，在本工作的范围内，RRG、概率路图、概率路图*、快速探索随机树均符合这一

方案。RRT* [15]、FMT* [17]、BIT* [18] 等更复杂的变体需要重新定义用于因子分解的特定额外操作，例如重新布线。因此我们不再考虑这些变体。我们确定了四个宏操作，即采样、近邻、无碰撞、添加边。

采样：从预先绘制的点集（如概率路图）中获取下一个待处理的样本，或从$\mathcal{X}_{\text{free}}$ 中抽取新样本（如RRG，快速探索随机树*）。

近邻：算法从图中查询位于新随机样本邻域内的节点。同样，根据算法不同，查询可能返回单个样本（快速探索随机树）、一定半径内的一组邻居（概率路图，概率路图*、 RRG），或 *k*近邻 顶点（k-概率路图、 k-概率路图*, k-RRG）。

无碰撞：_该函数检查 x_{rand} 与 x_{near} 之间的转换是否与障碍物及多智能体情形下的智能体本身发生碰撞。

添加边：最后，若无碰撞，则将 x_{rand} 添加到图中，并与 x_{near}建立一条边。根据算法不同，图中的边可以是有向 或无向的。

假设1。为简单起见，在本文余下部分中，我们将满足*算法1*的算法称为基于采样的算法（*SBA*）。

在满足假设1的算法中，我们选取 RRG 和概率路图 * 作为运行示例，因为它们具有最优性保证。

算法1 基于采样的算法

$\mathcal{G}$.添加节点(x_{init})
对于 $i = 0, \ldots, N-1$ **执行**
  x_{rand} =采样($\mathcal{X}_{\text{free}}$)
  X_{near} =近邻(x_{rand})
  对于 $x_{\text{near}} \in X_{\text{near}}$ **do**
    如果 无碰撞(x_{near}, x_{rand}) **则**
      $\mathcal{G}$.添加边(x_{near}, x_{rand})
返回 $\mathcal{G}$

III. SBA 的因子分解

我们介绍将因子分解应用于 SBA 的思想。由此可以更高效地以“混合式方法”求解 MA-MMPs[1]，同时保留在联合空间中运行的 SBA 的完备性和最优性保证。具体做法是在可能的情况下，利用智能体的因子分解（解耦），将其划分到彼此独立的低维搜索空间中。理解因子分解思想的关键在于：从智能体的某些联合配置出发，剩余的解可以独立搜索。当联合问题的解可以通过组合分别计算出的较小智能体子群的解来找到时（计算时假定其他智能体不存在），这一点便可以实现。

定义3（（最优）运动规划问题的独立性）。给定一个 *MA-MMP* $\mathcal{P}(x^{\mathcal{A}})$ 以及至少两个互不相交的智能体集合 $\mathcal{A}'$、$\mathcal{A}'' \subseteq \mathcal{A}$，我们称 $\mathcal{P}(x^{\mathcal{A}'})$ 和 $\mathcal{P}(x^{\mathcal{A}''})$ 关于 ϵ-最优性相互独立，记作 $\mathcal{P}(x^{\mathcal{A}'}) \perp^{\epsilon} \mathcal{P}(x^{\mathcal{A}''})$，当且仅当：

$$\begin{aligned} &\sigma^{\mathcal{A}} = \sigma^{\mathcal{A}'} \cup \sigma^{\mathcal{A}''} \in \textsc{Sol}^{\epsilon}(\mathcal{P}(x^{\mathcal{A}})), \\ &\forall \sigma^{\mathcal{A}'} \in \textsc{Sol}^{\epsilon}(\mathcal{P}(x^{\mathcal{A}'})),\ \forall \sigma^{\mathcal{A}''} \in \textsc{Sol}^{\epsilon}(\mathcal{P}(x^{\mathcal{A}''})). \end{aligned} \tag{5}$$

尽管定义 3 中给出的独立性概念具有一般性，但有时并不实用，因为要检验独立性，就必须知道因子化问题和联合问题的解。不过，我们发现了一些计算成本较低且足以推断独立性的启发式方法，例如基于智能体未来的资源占用情况 [10]。因此，我们将分解启发式视为一种函数：给定两个配置后，该函数判断是否可以推断出定义 3 中的条件。

定义4.（因子分解启发式）给定一个 *MA-MMP*$\mathcal{P}(x^{\mathcal{A}}_{\text{init}})$ 以及任意两个不相交的智能体集合 $\mathcal{A}', \mathcal{A}'' \subset \mathcal{A}$，我们称函数factorization heuristic 为 ϵ 最优性的因子分解启发式，其形式为

$$h : \mathcal{X}^{\mathcal{A}'} \times \mathcal{X}^{\mathcal{A}''} \to \texttt{Bool}, \tag{6}$$

使得

$$h(x^{\mathcal{A}'}_{\text{init}}, x^{\mathcal{A}''}_{\text{init}}) = 1 \implies \mathcal{P}(x^{\mathcal{A}'}_{\text{init}}) \perp^{\epsilon} \mathcal{P}(x^{\mathcal{A}''}_{\text{init}}). \tag{7}$$

我们认识到，寻找良好的因子分解启发式本身就可以成为一个完整的研究领域。实践中采用何种因子分解启发式，取决于具体应用；合理的选择通常需要结合对环境和其他智能体的额外认识与假设（例如，沿车道行驶的汽车的可达性分析、障碍物的类型、其他动态障碍物的行为以及拥挤程度）。

启发式方法的不同选择将决定问题被因子化的程度，以及算法1的每次迭代在计算上会增加多少负担。极端情况下，如果直接评估定义3，就会得到“完美因子分解”，也就是说，问题会被尽可能充分地因子化。在实际中，这或许适用于不拥挤的环境，因为分别为每个智能体计算解、且这些解已经无碰撞的概率较高；但在其他情况下，这种做法可能会带来沉重的计算负担。另一个极端是，永不进行因子分解的启发式仍是一个有效函数，显然计算成本很低，但也相当无用，因为它总是返回原始问题，始终在所有智能体的联合空间中求解。显然，这两种极端之间存在一整条帕累托前沿。

A. 因子分解例程

为了将因子分解思想嵌入 SBA，我们需要寻找一个具有以下签名的函数：FACTORIZE：$\mathcal{X} \to \text{Part}(\mathcal{X})$，其中 Part 表示划分的集合[2]。例如，考虑由 3 个智能体组成的联合状态 $x = [x^1, x^2, x^3]$，其划分为 $\{[x^1, x^2, x^3]\}$、$\{[x^1, x^2], [x^3]\}$、$\{[x^1], [x^2, x^3]\}$、$\{[x^1, x^3], [x^2]\}$ 或 $\{[x^1], [x^2], [x^3]\}$。

[1]从 I-A 节的意义上说。

[2]请注意，集合的划分是将其元素分组成非空子集，并确保每个元素恰好包含在一个子集中。

尽管在本文其余部分中，我们假设已有可用的启发式方法和 FACTORIZE 函数，但我们在此提供三个具体示例：

- 第一个示例通过 [10] 利用机器人的时空占据来实现。给定一个状态，分别计算每个机器人的前向可达集；如果两个集合不相交，则可将这些智能体视为相互独立。将此过程应用于所有智能体后，即可构建独立性图 [10, 定义 7]，其连通分量表示智能体的划分。

- 对前一点进行推广，如果针对当前问题还能获得其他信息，就可以基于这些信息进行因子分解。我们将在第V节给出一个示例：假设已知给定状态下的最优解位于我们与目标之间的一个锥内。我们设想，类似的思路也可用于知情搜索 [19] 。

- 结合独立智能体划分的定义，学习预测该划分。给定一个联合状态，我们尝试依据预测出的智能体划分对其进行因子分解。

B. 因子分解采样算法

下文将展示如何将第III-A节的因子分解例程集成到算法1中。有趣的是，由此得到的运动图是一个超图，其中包含有向超边；当智能体被因子化时，这些超边可从一个源节点连接到多个目标节点（图2b）。这种结构要求对部分标准例程进行调整时格外谨慎，例如搜索因子分解状态的最近邻。

一种因子分解采样算法，其形式与算法2相同，但对主要操作进行了如下重述：

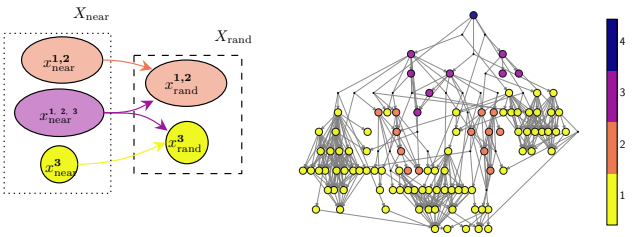
1) 采样：此例程保持不变，样本仍从联合空间中抽取。它们可能会在下一步中被因子化。

2) 因子化：此操作会按照第 III-A 节所述，将新样本映射到其某个划分中，从而对其进行因子化。我们用大写字母 X 表示状态集合。例如，X_{rand} 是

$$X_{\text{rand}} = \{x^{\mathcal{A}_i} : \mathcal{P}(x^{\mathcal{A}_i}) \perp \mathcal{P}(x^{\mathcal{A}_j}) \; \forall i \neq j\}. \tag{8}$$

算法 2 FactSBA

1: $\overrightarrow{X}_{\text{init}}$ =FACTORIZE(x_{init})
2: $\overrightarrow{\mathcal{H}}$.添加节点($X_{\text{init}}$)
3: **对于** $i = 0, \ldots, N-1$ **执行**
4: x_{rand} =采样($\mathcal{X}_{\text{free}}$)
5: X_{rand} =FACTORIZE(x_{rand})
6: **对** $\widehat{x}_{\text{rand}} \in$幂集$(X_{\text{rand}})$ **执行**
7: $X_{\text{near}} = \mathcal{A}$-近邻($\widehat{x}_{\text{rand}}$)
8: **对** $x_{\text{near}} \in X_{\text{near}}$ **执行**
9: **如果**无碰撞(x_{near}, $\widehat{x}_{\text{rand}}$) **则**
10: $\overrightarrow{\mathcal{H}}$.添加边($x_{\text{near}}$, $X^{\mathcal{A}(\widehat{x}_{\text{rand}})}$ rand)
11: **返回** $\overrightarrow{\mathcal{H}}$



(a)

(b) 图 2 $\{[x^1_{\text{rand}}, x^2_{\text{rand}}], [x^3_{\text{rand}}]\}$ x_{rand} $X_{\text{rand}} =$。在图 a 中，一个随机样本被因子化为 X_{rand}。重要的是，可以从两类邻接节点到达 的节点：一类是包含相同智能体的节点，另一类是包含智能体（部分）并集的节点。后者正是智能体发生因子化的转换。此结构要求在 pow(X_{rand}) 中搜索邻居。图 2b 展示了在四智能体问题中运行 RRG（算法2）的因子化版本后得到的完整图示例。初始状态是一个四维状态，但很快就解耦为一个三智能体节点和一个单智能体节点，最终图被因子化为四个单智能体组件。转换过程中智能体的解耦用一个小黑点表示。

3) 近邻：要理解此操作与算法1中的操作有何不同，关键在于注意到，X_{rand} 中的节点既可以通过包含相同智能体集合的邻居到达（例如，$\{x^3\} \to \{(x^3)^+\}$），也可以通过包含该智能体集合超集的节点到达（例如，$\{(x^3)^+\}$可以由节点 $\{x^{1,2,3}\}$ 到达；如图2a所示，该节点会在转换过程中被因子化）。概括而言，X_{rand}中所包含的智能体划分的所有可能组合，对应于 X_{rand}的幂集 pow(·)，其中既包含单个元素，也包含它们的组合。因此，与第6—7行一致，我们为幂集中的每个元素定义 $\mathcal{A}$-近邻操作，该操作实际上用于查找包含与 $\widehat{x}_{\text{rand}}$相同智能体集合的最近邻。需要特别注意的是，$\widehat{x}_{\text{rand}}$ 必须作为联合状态处理。

一般来说，X_{near} 仍然是一个集合，通常由连接半径或 k个最近节点确定。正如 [15] 中所观察到的，最优连接半径是搜索空间维数 d 的递减函数；在因子化情形下，该维数会因节点而异。可以在线调整连接半径，使其适应输入状态的维数；不过，只要将该半径设为最大联合搜索空间维数的函数，就足以保证收敛。最后，从计算角度来看，由于我们会检查幂集中的每个元素，每次迭代平均需要执行的最近邻查询次数会增加。尽管如此，这些查询仅涉及整个图中的一个较小子集。在我们的案例研究中，我们发现这并未增加每次迭代的平均计算时间。

4) 无碰撞：该例程与算法1相比没有实质性变化。我们仅指出，即使边会使智能体解耦，仍必须在 $\widehat{x}_{\text{rand}}$ 中将转换视为智能体之间的联合转换。这可以确保解耦转换无碰撞。

5) 添加边：与算法1不同，添加边因子化算法的操作会添加一条超边。源节点始终是单个节点 x_{near}，而目标节点可以是 X_{rand}中的一个节点（子集）。我们用 $X^{\mathcal{A}(\widehat{x}_{\text{rand}})}$ rand 包含与 $\widehat{x}_{\text{rand}}$ 相同智能体的 X_{rand} 的子集。换句话说，$X^{\mathcal{A}(\widehat{x}_{\text{rand}})}$ rand 是 $\widehat{x}_{\text{rand}}$ 的因子化版本，

该联合状态用于最近邻搜索和碰撞检测。因此，当 $X^{\mathcal{A}(\widehat{x}_{\text{rand}})}{}_{\text{rand}}$ 仅包含状态 $\widehat{x}_{\text{rand}}$ 时，超边将产生一条标准边；当 $X^{\mathcal{A}(\widehat{x}_{\text{rand}})}{}_{\text{rand}}$ 包含更多状态时，则会产生一条分裂边。不同情况如图 2a 所示。

重要的是，我们指出，选择某些因子化启发式方法可能会为新边引入额外的“一致性”约束。例如，若将搜索空间中的“未来资源”用作启发式方法，则目标节点必须属于起点的未来资源，才能保持解耦。最后，与非因子化版本一样，智能体转移的代价与边相关联。对于本文，分裂边的代价取各个转移代价之和。

C. 因子分解 *SBAs* 的性质

下文将证明，渐近最优 SBA 的因子分解对应算法保留了其最优性，即随着样本数量趋于无穷，该算法收敛到最优解。

命题 1. （因子分解 *SBAs* 的收敛性）渐近最优 *SBA* 的因子分解对应算法保留了其最优性，即随着样本数量趋于无穷，该算法收敛到最优解。

证明。 关键的直觉在于观察到，FactSBA 找到的路径可以看作是在相同或更低维搜索空间中运行的标准 SBA 算法所找到的解的组合。首先，注意在算法2中，当 $\widehat{x}_{\text{rand}}$ 与 $X^{\mathcal{A}(\widehat{x}_{\text{rand}})}{}_{\text{rand}}$ 重合时——即 $X^{\mathcal{A}(\widehat{x}_{\text{rand}})}{}_{\text{rand}}$ 未被因子化时——整个过程（近邻,添加边）与算法1中的过程一致。这仅适用于 $\vec{\mathcal{H}}$ 的子图，其中节点对应于智能体$\mathcal{A}(\widehat{x}_{\text{rand}})$的（联合）状态。事实上，如果我们从不因子化任何状态，算法2就与算法1相同。

现在考虑以下情况：根据构造，算法2经过 n 次迭代后找到的最优路径仅包含有限个（如果有的话）分裂超边（例如，图2b）。这些分裂超边的数量有限，因为可供分解的参与者数量有限。因此，对于给定的 $\epsilon \geq 0$，考虑由任意满足定义 4 的启发式构造的 FACTORIZE。根据构造，非平凡的 FACTORIZE 会将联合空间的部分分解为其“边缘”，即仅相对于特定参与者子集的投影；我们将这些子集称为块，以呼应划分中的块。随后，算法2对每个块运行算法1，寻找从该块的初始节点到后续划分（或目标）的最优路径。请注意，对于第一个块，其“初始节点”是固定的（x_{init}）。

最后，如果让算法1运行足够长的时间，那么对于每个块，找到的路径几乎必然至少是 ϵ-最优的，因此满足解耦条件。由于这对任意块都成立，整体路径也至少是 ϵ-最优的，因为如定义2所述，因子化 ϵ-最优路径的乘积对于联合问题仍然是ϵ-最优的（引理1）， ϵ-最优路径的拼接也同样如此（引理2）。■

该证明强调了一个重要方面。如果我们根据 ϵ-最优性进行 FACTORIZE，解耦可确保可行性但只有当我们在子问题中找到至少达到 ϵ-最优的解时才成立。实际上，我们观察到，较大的 epsilon 虽然允许进行较少的 FACTORIZE，但能够更快地找到有保证可解耦的解。然而，在线选择和估计 ϵ 可能并不实际。

不过，也存在一些“变通方法”，可以确保找到的任意解都是可行的（而不只是至少 ϵ-最优的解）。例如，可以使用启发式方法，为每个节点关联一组状态，并在这些集合不相交时进行因子化。随后，仅当 $X^{\mathcal{A}(\widehat{x}_{\text{rand}})}{}_{\text{随机}}$ 属于该集合时才添加连接（算法2第10行）；对于 x_{near}，如果这些集合是正向不变的，且每个集合都包含于其父节点的集合中，那么我们可以保证，在该块中找到的任何可行路径都已经实现了有效解耦（例如，[10]）。

IV. 通过因子分解获得概率路图* 的采样增益

下面我们介绍样本复杂度增益。形式上，它定义为在有因子分解和无因子分解的情况下，达到给定最优性所需的最少样本数之比，其中达到该最优性的概率相同。显然，这种增益取决于当前问题内在的“可因子化性”，即联合配置空间中有多大比例可以进行因子分解。重要的是，它衡量了在联合空间中运行的 SBA 收敛速度较慢所造成的改进；这种收敛速度在搜索空间的维度上通常至少呈指数级增长 [1,页 324]。

定义5 （通过因子分解获得的样本复杂度增益）。设 $\mathcal{P}$（$x_{\text{init}}^{\mathcal{A}}$）为定义*2*中定义的最优 *MA-MMP*。接着，任取期望的最优性 $\epsilon > 0$ 和概率水平 $\bar{p} \in [0,1]$，使得 $\mathbb{P}[\,(c\,(\sigma^{\texttt{ALG}}) - c\,(\sigma^*)\,)\,/c\,(\sigma^*) \leq \epsilon] \geq \bar{p}$。我们将因子分解带来的样本复杂度增益定义为：因子分解采样算法为达到 ϵ-最优性且成功概率至少为 $\bar{p}$而节省的样本所占的百分比。该量表示为

$$\mathsf{g}^{\mathsf{Fact}} := 1 - \frac{N_{\epsilon,\bar{p}}^{\mathsf{Fact}}}{N_{\epsilon,\bar{p}}}. \tag{9}$$

接下来，我们针对概率路图*这一具体情形推导定义 5。首先回顾一下，对于大多数 SBA，要求达到某种最优性等价于使样本序列的离散度足够低（定义 8）[3]。因此，我们考虑:

定义 6 （达到目标离散度所需的最少样本数）。 设S 是 $[0,1]^d$上的一个稠密（极限意义下）序列。固定 $\bar{D}$、$\bar{p} \in (0,1]$ 分别为目标离散度和概率水平。我们将$N_{\bar{D},\,\bar{p}} \in \mathbb{N}$ 记为满足 $\mathbb{P}\,(D\,(S) < \bar{D}) \geq \bar{p}$的 S 样本数的最小值。

对于 ℓ_∞-离散度和均匀采样，我们通过取满足命题 3 的最小 N 来近似定义 6。现在，我们引入因子分解因子的定义，用于量化某个 MA-MMP 的可因子分解程度。

定义 7 (因子分解因子)。 给定定义在（联合）自由空间 $\mathcal{X}_{\text{free}}$上的 *MA-MMP*$\mathcal{P}(x_{\text{init}}^{\mathcal{A}})$，对于智能体的一种可能划分 Part，我们将其因子分解

[3]这里稍微滥用一下记号，我们将“采样序列”和“点集”这两个术语互换使用。

因子定义为 $\mathcal{X}_{\text{free}}$ 中可根据 Part 进行因子化的部分所占的比例，且不超过此范围——记作 $\mathcal{X}_{\text{free}}^{\mathsf{Part}}$。我们将其记为

$$\mathsf{f}^{\mathsf{Part}} := \frac{\mu(\mathcal{X}_{\text{free}}^{\mathsf{Part}})}{\mu(\mathcal{X}_{\text{free}})}. \tag{10}$$

命题 2.（*FactPRM*的样本复杂度增益*）给定定义 *5* 中的一个最优 *MA-MMP* $\mathcal{P}(x_{\text{init}}^{\mathcal{A}})$ 以及任意期望的最优性 $\epsilon \geq 0$ 和概率水平$\bar{p} \in [0,1]$。采用均匀采样，并以概率路图* 作为 *SBA*。为简化起见，假设 $\mathcal{X}_{\text{free}} = [0,1]^d$中比例为f 的部分是“完全可因子分解”的[4]，，则通过因子分解获得的样本复杂度增益为：

$$\mathsf{g}_{\mathsf{FactPRM}^*}^{\mathsf{Fact}} = \mathsf{f} - \frac{(1-\mathsf{f})\log(1-\mathsf{f})}{\log\left(\frac{\mu}{\bar{D}^{d^i|\mathcal{A}|}(1-\bar{p})}\right)} + \mathcal{O}(\bar{D}^{d^i(|\mathcal{A}|-1)}). \tag{11}$$

证明。令 FactPRM* 表示¯概率路图*的因子化版本。令 $D = D_\infty(S)$ 为满足最优性要求 ϵ 的充分离散度，该要求根据 [20,定理 2]得到。注意，$N_{\bar{D},\,\bar{p}}$ 可由命题 3 得出。因此，PRM* 在联合空间中运行所需的最少样本数为：

$$N_{\bar{D},\bar{p}}^{\mathsf{PRM}^*} = \frac{\mu(\mathcal{X}_{\text{free}})}{\bar{D}^{d^i|\mathcal{A}|}} \log\left(\frac{\mu(\mathcal{X}_{\text{free}})}{\bar{D}^{d^i|\mathcal{A}|}(1-\bar{p})}\right). \tag{12}$$

根据假设，因子化 算法 FactPRM* 能够完美地解耦属于“完全可因子分解”区域的样本。然后，我们可以将 FactPRM* 的搜索空间分为两部分：第一，联合空间中“不可因子分解”的子区域，其勒贝格测度为 $(1-\mathsf{f})\mu(\mathcal{X}_{\text{free}})$，维度为 $d^i|\mathcal{A}|$。第二，与 $|\mathcal{A}|$ 个子问题相对应的因子分解子区域，每个子问题的维度为 d^i，勒贝格测度为 $(\mathsf{f}\mu(\mathcal{X}_{\text{free}}))^{\frac{1}{|\mathcal{A}|}}$。请注意，为确保整体以概率 $\bar{p}$ 满足最优性要求，我们需要提高每个子问题的概率要求，使这些事件同时发生的概率为 $\bar{p}$。对于每个子问题，我们将 $\bar{p}^i := \bar{p}^{\frac{1}{|\mathcal{A}|}}$ 定义为提高后的概率水平。

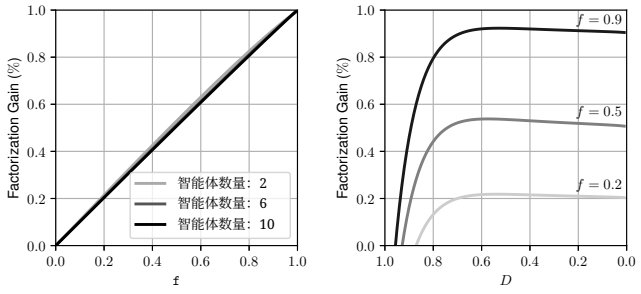
为求得所需的最少样本数，我们将各子问题所需的样本数相加；但由于一个样本被因子化后，已经提供了 $|\mathcal{A}|$ 个更低维度的样本，因此有：

$$N_{\bar{D},\bar{p}}^{\mathsf{FactPRM}^*} = N_{\bar{D},\bar{p}}^{\text{joint}} + N_{\bar{D},\bar{p}^i}^{\text{i}}, \tag{13}$$

其中，$N_{\bar{D}\,\bar{p}}^{\text{joint}}$、以及 $N_{\text{i}\bar{D}、\bar{p}^i}$分别对应于关于“联合区域”和“单个智能体”的定义 6。请注意，(13) 确实保留了整体最优性，因为根据引理 1 和引理 2，每个子问题都保证满足最优性要求，因此它们的组合也满足该要求。应用命题 3，并使用 μ 作为 $\mu(\mathcal{X}_{\text{free}})$ 的简写，可得：

$$\begin{aligned} N_{\bar{D},\bar{p}}^{\mathsf{FactPRM}^*} = {} & \frac{(1-\mathsf{f})\mu}{\bar{D}^{d^i|\mathcal{A}|}} \log\left(\frac{(1-\mathsf{f})\mu}{\bar{D}^{d^i|\mathcal{A}|}(1-\bar{p})}\right) \\ & + \frac{(\mathsf{f}\mu)^{\frac{1}{|\mathcal{A}|}}}{\bar{D}^{d^i}} \log\left(\frac{(\mathsf{f}\mu)^{\frac{1}{|\mathcal{A}|}}}{\bar{D}^{d^i}(1-\bar{p}^{\frac{1}{|\mathcal{A}|}})}\right) \end{aligned} \tag{14}$$

[4]所有智能体被因子化为独立集合的划分。



) 图 3。图 3a展示了因式分解增益确实几乎与问题的“可因子化性”呈线性关系。显然，尽管该增益几乎不随参与者数量变化，但样本的绝对数量显然并非如此。在图 3b 中，我们展示了因式分解增益随所需最优性（因此为较低的 ℓ_∞-离散度）变化的关系。可以观察到，$\mathsf{g}_{\mathsf{FactPRM}^*}^{\mathsf{Fact}} \to \mathsf{f}$ 随着 $\bar{D} \to 0$ 而变化。总体而言，我们发现这些图对参数的不同选择不太敏感；图中所示结果固定了 $\bar{p} = .7$、$d^i = 2$、$\bar{D} = .7$（图 3a）以及 $|\mathcal{A}| = 5$（图 3a）。

Lastly substituting (14) and (12) into (9) we have $\frac{N_{\epsilon,\bar{p}}^{\mathsf{Fact}}}{N_{\epsilon,\bar{p}}} =$

$$\frac{\frac{(1-\mathsf{f})\mu}{\bar{D}^{d^i|\mathcal{A}|}}\log\left(\frac{(1-\mathsf{f})\mu}{\bar{D}^{d^i|\mathcal{A}|}(1-\bar{p})}\right) + \frac{(\mathsf{f}\mu)^{\frac{1}{|\mathcal{A}|}}}{\bar{D}^{d^i}}\log\left(\frac{(\mathsf{f}\mu)^{\frac{1}{|\mathcal{A}|}}}{\bar{D}^{d^i}(1-\bar{p}^{\frac{1}{|\mathcal{A}|}})}\right)}{\frac{\mu}{\bar{D}^{d^i|\mathcal{A}|}}\log\left(\frac{\mu}{\bar{D}^{d^i|\mathcal{A}|}(1-\bar{p})}\right)}, \tag{15}$$

由此可得式（11）。 ■

V. 实验

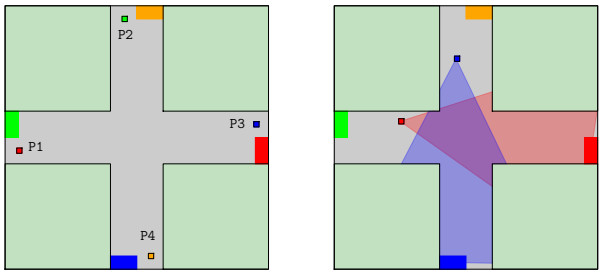
我们对 RRG [15] 上 SBA 的因子分解进行基准测试；我们将其因子分解对应算法命名为 FactRRG。我们选择 RRG 作为运行示例，因为我们认为，在无需重新布线或在算法1基础上增加任何复杂机制的情况下，它是具备最优性和随时性属性的 SBA 中最简单的一种。

评估场景如图4a所示，我们分别考虑包含2、3和4个智能体的三种情况。智能体的初始条件设定在走廊末端，目标区域则位于与之相对的某条走廊中。中央交叉口迫使智能体在大部分空间内“保持耦合”。智能体为完整约束，且没有动力学约束。每个智能体的状态由其在二维平面上的位置构成，因此 $x \in \mathbb{R}^2$；控制输入为沿 x 和 y 方向的速度$[v_x, v_y] \in \mathbb{R}^2$。显然，可以扩展到具有动力学约束的更复杂智能体和场景，但这超出了本文的讨论范围。

实现细节： RRG 和 FactRRG 的实现分别对应算法1和算法2。在两个版本中，我们都将连接半径 设为 RRG 的最优值 [15]。

其形式为 $r_{\mathtt{RRG}} = \min\left(\gamma_{\mathtt{RRG}}\left(\frac{\log(n)}{n}\right)^{\frac{1}{d}}, \eta_{\mathtt{RRG}}\right)$，其中$n$ 为节点数，用户定义的参数设为 $\gamma_{\mathtt{RRG}} = 100$, $\eta_{\mathtt{RRG}} = 100$。此外，为加快收敛，两个算法中的采样过程均设定为以 0.1 的概率返回目标区域内的一个点。

每次试验中，两个算法都使用相同的随机种子依次运行，以确保两个算法采用相同的采样序列。在实验中，我们将固定数量的图节点作为停止准则；更准确地说，当找到一个解且相应图中的节点数超过 $1k$时，我们就停止执行。



(a) 场景　　(b) 未来资源

图 4. 图 4a 展示了 4 智能体情况下的评估场景。2 智能体和 3 智能体的情况通过依次选取参与者得到：2 个参与者（P1，P2）、3 个参与者（P1，P2，P3）和 4 个参与者（P1，P2，P3，P4）。图 4b 展示了实验中使用的未来资源计算实例。

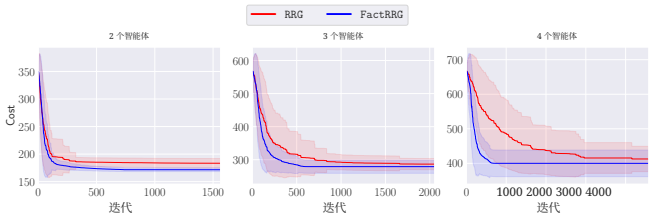


图 5. RRG 算法及其因子分解对应算法 FactRRG的代价随迭代次数的变化。两种算法使用相同的参数和随机序列，依次运行，每种算法重复试验 100 次。平均代价和相对 1 个标准差带均表明，随着参与者数量增加，因子分解对应算法的效果更好。

采用的 因子分解启发式 如图 4b 所示。它依赖于我们对环境所作的额外假设，即每个智能体与其目标之间的锥形区域内都存在一条解路径。给定智能体坐标后，我们考虑一个以当前位置至相应目标区域的直线路径为中心、向周围延伸的锥。这隐含地假设，在此环境中，最优解与直线路径大致“对齐”。显然，这种特定选择适用于此环境，但在迷宫等其他环境中则会失效。尽管可以构造出问题“可因子化性”为零的病态案例，但我们认为，许多有趣的应用都允许定义实用的因子分解启发式（例如，自动驾驶中的可达集、仓库中的走廊等）。在此场景中，我们利用了这样一个假设：智能体无需反向行进或离开锥形区域，即可找到通往目标的最优路径。

A. 结果与讨论

RRG 与 FactRRG 的性能对比见图 5。在全部三个实例中，可以看出，因子化 算法优于基线算法，能更快找到代价更低的路径，而且随着智能体数量增加，性能差距也随之扩大。值得注意的是，尽管每次迭代的计算成本略有增加，仍然会出现这一结果。例如，在 4 个智能体的情形下，计算成本从 80($\pm$11) ms 增至 98($\pm$13) ms($+22\%$)，但以少得多的迭代次数实现了显著更低的代价（-75%）。

我们注意到，在样本数较少时，标准算法几乎无法比其因子化版本更早找到解。这是由于采用了这种分解启发式，它阻止 FactRRG 在已因式分解节点与未因式分解节点之间建立连接（充分利用了启发式提供的信息：“从这里开始，解都位于锥内”）。由于 RRG 无需遵守未来资源约束，因此在样本数较少时会创建更多图中的边，这一点在表 I 的完整图中也有所体现。尽管如此，随着样本数增加，因子分解的优势便逐渐显现。

最后，表 I 展示了 3 个智能体的情况下，两种算法每次运行结束时图的平均维度。结果表明，因子化算法 FactRRG 在解的代价方面优于 RRG，同时还能大幅降低图的维度。边数减少的首要原因是 RRG 创建的图中包含许多双向边，尽管如此，边数比例仍远低于 50%。第二个也是最重要的原因如下：假设存在 a 个单个智能体节点，每个节点有 m 条出边；若要在智能体联合的情况下表示该图，则需要 m^a 条边，其数量随智能体数量呈指数增长，而在解耦表示中则仅呈线性增长（am）。

表 I

# 智能体	2	3	4
RRG(# 边)	$90.4k(\pm 2.8k)$	$60.4k(\pm 2.4k)$	$34.6k(\pm 5.9k)$
FactRRG（# 条边）	**13.0k**($\pm$**0.9k**)	**9.7k**($\pm$**0.7k**)	**8.3k**($\pm$**0.8k**)

VI. 结论

通过将因子分解 思想应用于标准SBA，我们表明可以自然地构建出更适用于多智能体规划问题的因子分解SBA。最重要的是，所得算法通常能够保留原算法的特性（完备性、最优性、随时可用性.....），同时大幅降低计算负载（这一点已通过 PRM* 的解析分析和 RRG 的实证研究得到验证）。

因子分解的有效性取决于启发式函数，该函数能够判断在剩余问题中玩家解何时彼此解耦。尽管我们提供了具体示例，但我们认为这仍是最值得探索的未来研究方向之一。另一个可能的拓展是证明因式分解原理也适用于更复杂的 SBA。总之，我们相信因子分解的全部潜力尚待发掘。

参考文献

[1] S. M. LaValle，《规划算法》。剑桥大学出版社，2006年11月。
[2] T. Standley 和 R. Korf，“协同寻路问题的完备算法”，载于第二十二届国际人工智能联合会议论文集——第一卷，系列：IJCAI’11。AAAI出版社，2011年，第668–673页。[3] J. E. Hopcroft、J. T. Schwartz 和 M. Sharir，“多个独立物体的运动规划复杂性；‘仓库管理员问题’的 PSPACE-困难性”，《国际机器人研究杂志》，第3卷，第4期，第76–88页，1984年。

[4] S. Karaman 和 E. Frazzoli，“最优运动规划的基于采样的算法”，载于《国际机器人研究杂志》，第30卷，第7期，2011年6月，第846–894页。[5] M. A. Erdmann 和 T. Lozano-Perez，“关于多个移动物体”，《算法学》，第2卷，第477–521页，1986年。[6] M. ˇC´ap、P. Nov´ak、A. Kleiner 和 M. Seleck´y，“多移动机器人轨迹协调的优先级规划算法”，《IEEE自动化科学与工程汇刊》，第12卷，第3期，第835–849页，2015年。[7] D. Silver，“协同寻路”，载于第一届 *AAAI*人工智能与互动数字娱乐会议论文集，系列：AIIDE'05。AAAI出版社，2005年，第117–122页。[8] R. Shome、K. Solovey、A. Dobson、D. Halperin 和 K. E. Bekris，“dRRT*：可扩展且具有知情性的渐近最优多机器人运动规划”，《自主机器人》，第44卷，467，2020年3月3日4日，443号。第–页。[9] G. Wagner 和 H. Choset，“多机器人路径规划的子维度扩展”，《人工智能》，第219卷，第1–24页，2015年2月。[10] A. Zanardi、S. Bolognani、A. Censi、F. Dorfler 和 E. Frazzoli，“时空资源上的动态博弈因子分解”，载于*2022*年 *IEEE/RSJ* 智能机器人与系统国际会议（*IROS*）。IEEE，2022年10月，第13 159–13 166页。[11] M. ˇC´ap、P. Nov´ak、J. Vok ˇr´ınek 和 M. P ˇechou ˇcek，“多智能体 RRT*：基于采样的协同寻路（扩展摘要）”，载于第十二届自主智能体与多智能体系统国际会议论文集，*AAMAS'13*，2013年。[12] G. Sanchez 和 J.-C. Latombe，“使用 PRM规划器比较多机器人系统的集中式规划与解耦规划”，载于*2002*年 *IEEE*机器人与自动化国际会议论文集（目录号：*02CH37292*），第2卷，2002年，第2112–2119页。[13] A. Dreves 和 M. Gerdts，“自动驾驶汽车最优控制问题的广义纳什均衡方法”，《最优控制应用与方法》，第39卷，第1期，第326–342页，2018年。[14] T. Standley，“求解协同寻路问题的最优解”，《AAAI人工智能会议论文集》，第24卷，第1期，第173–178页，2010年7月。[15] S. Karaman 和 E. Frazzoli，“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。[16] A. Zanardi、E. Mion、M. Bruschetta、S. Bolognani、A. Censi 和 E. Frazzoli，“具有字典序偏好和社会有效纳什均衡的城市驾驶博弈”，《IEEE机器人与自动化快报》，第6卷，第3期，第4978–4985页，2021年7月。[17] L. Janson、E. Schmerling、A. Clark 和 M. Pavone，“快速行进树：一种用于高维最优运动规划的快速行进采样方法”，《国际机器人研究杂志》，第34卷，第7期，第883–921页，2015年6月。[18] J. D. Gammell、S. S. Srinivasa 和 T. D. Barfoot，“批量知情树（BIT*）：通过对隐式随机几何图进行启发式引导搜索实现基于采样的最优规划”，载于*2015*年 *IEEE*机器人与自动化国际会议（*ICRA*）。IEEE，2015年5月，第3067–3074页。[19] ——，“知情 RRT*：通过直接采样可采纳椭球启发式实现聚焦的最优基于采样的路径规划”，载于*2014*年 *IEEE/RSJ* 智能机器人与系统国际会议。IEEE，2014年9月，第2997–3004页。[20] L. Janson、B. Ichter 和 M. Pavone，“确定性基于采样的运动规划：最优性、复杂度与性能”，《国际机器人研究杂志》，第37卷，61，2018年1月1日，46号。第–页。

VII. 附录

定义 8 (ℓ_p-离散度)**.** 对于包含在具有正勒贝格测度的 d 维紧欧几里得子空间 $\mathcal{X}$ 中的有限非空点集 S，其 ℓ_p-离散度$D_p(S)$定义为

$$D_p(S) = \sup_{x\in\mathcal{X}} \left\{ \min_{s\in S} \|x-p\|_p \right\}. \tag{16}$$

直观而言，离散度量化了一组点 S 对空间的覆盖程度，其依据是能够接触到这些点中任何一点的最大开欧几里得球。

命题 3（采样数量的充分条件，适用于ℓ_∞-离散度）。设 $\mathcal{X}$ 是一个测度为 $\mu(\mathcal{X})$、维数为 d的勒贝格可测空间。设 S 是在 $\mathcal{X}$上抽取的一列 N独立同分布均匀样本，对于任意 $D, \bar{p} \in (0,1]$, ，使得$\mathbb{P}[D_\infty(S) \le D] \ge \bar{p}$ 成立的充分样本数量 $N > 0$ 为:

$$N \ge \frac{\mu(\mathcal{X})}{\bar{D}^d} \log\left(\frac{\mu(\mathcal{X})}{\bar{D}^d(1-\bar{p})}\right). \tag{17}$$

证明。将 $\mathcal{X}$ 在 $m = \lceil \frac{\mu(\mathcal{X})}{\bar{D}^d} \rceil$ 个相同的 d 维超立方体（单元）中细分。注意，每个立方体的边长小于或等于 D。此外，注意到每个单元至少有一个样本，是满足 $D_\infty(S) \le D$ 的充分条件。设 A_i 为第 i 个单元不包含来自 S 的任何样本这一事件，则 A_i 的概率为 $\mathbb{P}(A_i) = \left(1 - \frac{1}{m}\right)^N$。因此，应用并集界后，至少有一个单元为空的概率为

$$\mathbb{P}[\cup_i A_i] \le \sum_i \mathbb{P}[A_i] = m\left(1-\frac{1}{m}\right)^N. \tag{18}$$

因此，考虑其补事件“所有单元都至少包含一个样本”——该事件可确保达到所需的差异——我们得到:

$$\mathbb{P}[D_\infty(S) \le \bar{D}] \ge 1 - m\left(1-\frac{1}{m}\right)^N \ge 1 - m\exp\left(-\frac{N}{m}\right), \tag{19}$$

其中，第二个不等式由 $1 + x \le e^x$ 对任意 $x \in \mathbb{R}$ 均成立得出。最后，对于给定的 $\bar{p}$，代入 m 并求解 N，可得:

$$N \ge \frac{\mu(\mathcal{X})}{\bar{D}^d} \log\left(\frac{\mu(\mathcal{X})}{\bar{D}^d(1-\bar{p})}\right), \tag{20}$$

其中我们利用了 $1 - m\exp\left(-\frac{N}{m}\right)$ 随 m 增大而递减这一事实，以及 $m = \lceil \frac{\mu(\mathcal{X})}{\bar{D}^d} \rceil \ge \frac{\mu(\mathcal{X})}{\bar{D}^d}$。■

引理 1 (ϵ-解的乘积的最优性)。给定一个 *MA-MMP*，令 σ^* 为其最优解，并将 $\sigma = \{\sigma_i\}_{i\in\mathcal{A}}$ 视为该问题的一个可行解。那么:

$$\epsilon = \frac{c(\sigma) - c(\sigma^*)}{c(\sigma^*)} \le \max_{i\in\mathcal{A}}(\epsilon^i) \tag{21}$$

其中 $\epsilon^i = \frac{c^i(\sigma) - c^i(\sigma^*)}{c^i(\sigma^*)}$。

证明。根据联合代价和 ϵ-最优的定义:

$$\begin{aligned}\epsilon &= \frac{c(\sigma)-c(\sigma^*)}{c(\sigma^*)} = \frac{\sum_{i\in\mathcal{A}} c^i(\sigma) - \sum_{i\in\mathcal{A}} c^i(\sigma^*)}{\sum_{i\in\mathcal{A}} c^i(\sigma^*)} \\ &= \frac{\sum_{i\in\mathcal{A}}\left[c^i(\sigma)-c^i(\sigma^*)\right]}{\sum_{i\in\mathcal{A}} c^i(\sigma^*)} \le \frac{\sum_{i\in\mathcal{A}}\left[\epsilon^i c^i(\sigma^*)\right]}{\sum_{i\in\mathcal{A}} c^i(\sigma^*)} \le \max_{i\in\mathcal{A}} \epsilon^i.\end{aligned} \tag{22}$$

■

引理 2 (ϵ-路径拼接的最优性)**.** 引理 *1* 也适用于路径拼接。

Proof. Analogous to Lem. 1. ■