

doi: 10.7690/bgzd.2025.10.017

基于风险规避 RRT*算法的机器人路径规划

曹子唯, 陈益超, 黄彬彬, 王子涵, 王晓东, 齐 辉

(杭州供电公司输电运检中心, 杭州 310000)

摘要: 为解决随机非线性机器人系统的控制问题, 提出一种 2 阶段风险规避架构改进 RRT*算法。通过求解 NLP 引入非线性动力学, 采用近似状态分布和分布鲁棒性碰撞检测来量化风险, 从而完成路径规划。通过演示 3 种控制器: 参考轨迹周围具有线性动力学的 LQR、含鲁棒性乘性噪声项的 LQR 以及非线性模型预测控制器。在独轮车动力学模型中施加具有重尾特性的 Laplace 噪声。结果表明, 该算法具有可行性。

关键词: RRT*算法改进; 非线性动力学; 分布鲁棒性; 模型预测控制; 独轮车模型

中图分类号: TP24 **文献标志码:** A

Robot Path Planning Based on Risk Aversion RRT * Algorithm

Cao Ziwei, Chen Yichao, Huang Binbin, Wang Zihan, Wang Xiaodong, Qi Hui

(Transmission Operation and Maintenance Center of Hangzhou Power Supply Company, Hangzhou 310000, China)

Abstract: In order to solve the control problem of stochastic nonlinear robot systems, an improved RRT * algorithm with two-stage risk aversion architecture is proposed. The nonlinear dynamics is introduced by solving the NLP, and the approximate state distribution and distribution-robust collision detection are used to quantify the risk, so as to complete the path planning. Three kinds of controllers are demonstrated: LQR with linear dynamics around the reference trajectory, LQR with robust multiplicative noise term, and nonlinear model predictive controller. Laplace noise with heavy-tailed characteristics is imposed in the unicycle dynamics model. The results show that the algorithm is feasible.

Keywords: RRT * algorithm improvement; nonlinear dynamics; distributional robustness; model predictive control; unicycle model

0 引言

21 世纪以来, 随着各种高新技术的不断发展, 移动机器人也逐渐走入现实生活。路径规划技术在很多领域都具有广泛的应用。在高科技领域的应用有: 巡航导弹躲避雷达搜索、防反弹袭击、完成突防爆破任务等^[1]。在工业领域, 移动机器人能够对工厂的各个环节进行有效操作, 实现自动化生产, 提高产品质量, 节省劳动力和人力成本^[2]。路径规划技术作为移动机器人自主导航的关键技术^[3-4], 旨在研究如何为移动机器人生成一条从起点到终点的最优路径, 使其具备自主避障能力。

因实际机器人系统存在本质非线性和随机性, 无法确保安全性。为此, Blackmore 等^[5-6]提出了考虑随机分量的运动规划方法以提供概率安全保障, 包括基于 CVaR 的线性动力学风险感知规划^[7]和基于 Wasserstein 度量的规划框架^[8]。

RRT 因其特有的采样方式被广泛用于路径规划研究, 具有快速收敛、高效计算和大范围探索等优点。但受限于其无导向的随机采样特性, 该算法仍

存在冗余节点过多和路径非渐进最优等不足。Karaman 等^[9]在算法的基础上提出了渐进最优的 RRT*算法。RRT 和 RRT*的概率约束版本为维持计算可行性, 对系统不确定性做了高斯分布的强假设, Summers 等^[10]证明此类假设会导致风险误判, 进而提出基于矩的模糊集构建方法, 开发了分布鲁棒 RRT 变体 DR-RRT。Renganathan 等^[11]扩展了该框架, 结合 LQR 和卡尔曼滤波状态估计, 设计了具有渐近最优性的输出反馈 DR-RRT*算法。笔者首次尝试为非线性机器人系统设计基于非线性转向的风险感知运动规划。

LQR 控制通过动态规划最小化状态偏差与控制能耗的二次型成本, 是最常见方案。Gravell 等^[12]进一步提出鲁棒 LQR, 通过引入乘性噪声实现参数不确定性下的均方稳定。笔者使用 NMPC 跟踪基于 NLP 转向函数生成的风险规避 RRT 轨迹, 其与底层 NMPC 控制器具有更高一致性。

1) RANS-RRT*算法: 提出一种面向非线性机器人系统的基于采样的运动规划器, 通过构建满足

收稿日期: 2024-10-09; 修回日期: 2024-11-15

第一作者: 曹子唯(1998—), 男, 河南人, 硕士。

分布鲁棒状态约束的动态可行轨迹来增强安全性。

2) 实验验证: 以独轮车模型为测试平台, 通过蒙特卡洛仿真对比研究 3 种底层跟踪控制器(标准 LQR 控制、鲁棒 LQR 控制和非线性模型预测控制), 其评估指标涵盖了避碰成功率、状态偏差成本、控制能耗、计算开销和不同扰动强度下的性能表现。

1 高层规划器: RANS-RRT*

笔者提出了一种基于快速探索随机树的近似解决方案, 并提出了 RANS-RRT*算法: 一种风险规避的非线性转向 RRT*规划器。

1.1 非线性规划(NLP)转向

RANS-RRT*算法采用非线性转向策略, 计算一条由状态—输入对构成的轨迹 $J=\{(s_0, u_0), \dots, (s_N, u_N)\}$ (长度为 N), 将初始状态 S_{init} 驱动至目标状态 S_{des} 。初始状态属于 RANS-RRT*树 T , 目标状态可为采样状态或其他树节点。NLP 转向定义如下:

定义 (NLP 转向): 给定初始状态 S_{init} 、目标状态 S_{des} 及转向步长 N , NLP 转向的解是一组无风险约束的确定性优化问题的状态序列 $\{s_0, \dots, s_N\}$ 与控制序列 $\{u_0, \dots, u_{N-1}\}$ 。

$$J_{NLP} = \min_{u[0:N-1]} \sum_{k=0}^{N-1} u_k^T R[k] u_k \quad (1)$$

1.2 均值与协方差传播

NLP 转向生成的轨迹为 $J=\{(s_0, u_0), \dots, (s_{N-1}, u_{N-1}), (s_N)\}$ 。将其作为节点间轨迹加入 RANS-RRT*树 T 。由于直接集成反馈控制至 RANS-RRT*轨迹极具挑战性, 因此, 采用 NLP 轨迹的开环控制策略, 并强制要求: RANS-RRT*轨迹的均值状态必须与 NLP 轨迹 J 完全匹配并且基于固定均值状态与输入, 通过无迹变换估计状态协方差。

1.3 风险处理

对于 RANS-RRT*轨迹中的状态—协方差对 $(\hat{x}[k], \hat{\Sigma}[k])$, 其风险约束表述为 $\inf_{p_k \in \mathcal{P}_k^x} \mathbb{P}_k^x(x[k] \in X_{free}) \geq 1 - \alpha_k \Leftrightarrow \sup_{p_k \in \mathcal{P}_k^x} \mathbb{P}_k^x(x[k] \in \mathbb{C} \cup X^c) \leq \alpha_k$, 其中 $\mathbb{C}$ 为障碍物集合的并集。

进入障碍物 $\mathcal{O}_i$ 的风险限值为 $\alpha_k/n_{ob_i}/n_{total}$, 超出环境边界的风险限值为 $\alpha_k n_{env}/n_{total}$, 得到风险约束

的分解式:

$$\sum_{i=1}^F \sup_{p_k^x \in \mathcal{P}_k^x} \mathbb{P}_k^x(x[k] \in \mathbb{O}_i) + \sup_{p_k^x \in \mathcal{P}_k^x} \mathbb{P}_k^x(x[k] \in X^c) \leq \sum_{i=1}^F \alpha_k n_{ob_i}/n_{total} + \alpha_k n_{env}/n_{total} = \alpha_k \quad (2)$$

并且有具体的约束条件, 现在需要满足以下 2 个核心不等式:

$$\begin{aligned} \sup_{p_k^x \in \mathcal{P}_k^x} \mathbb{P}_k^x(x[k] \in X^c) &\leq \alpha_k n_{env}/n_{total}; \\ \sup_{p_k^x \in \mathcal{P}_k^x} \mathbb{P}_k^x(x[k] \in \mathcal{O}_i) &\leq \alpha_k n_{ob_i}/n_{total}. \end{aligned} \quad (3)$$

应用布尔不等式(9)可得:

$$\sum_{j=1}^{n_{env}} \sup_{p_k^x \in \mathcal{P}_k^x} \mathbb{P}_k^x(a_{env,j}^T x[k] > b_{env,j}) \leq \alpha_k n_{env}/n_{total}. \quad (4)$$

根据式(3)可以得到:

$$\begin{aligned} \inf_{p_k^x \in \mathcal{P}_k^x} \mathbb{P}_k^x\left(\bigvee_{j=1}^{n_{ob_i}} a_{ob_i,j}^T x > b_{ob_i,j}\right) &> 1 - \alpha_k n_{ob_i}/n_{total}; \\ \inf_{p_k^x \in \mathcal{P}_k^x} \mathbb{P}_k^x\left(\bigvee_{j=1}^{n_{ob_i}} a_{ob_i,j}^T x > b_{ob_i,j}\right) &> \inf_{p_k^x \in \mathcal{P}_k^x} \mathbb{P}_k^x(a_{ob_i,j}^T x > b_{ob_i,j}). \end{aligned} \quad (5)$$

根据弗雷歇—布尔下界推导, 若满足

$$\begin{aligned} \inf_{p_k^x \in \mathcal{P}_k^x} \mathbb{P}_k^x(a_{ob_i,j}^T x[k] > b_{ob_i,j}) &> 1 - \alpha_k/n_{total} \forall j \text{ 则可推出} \\ \inf_{p_k^x \in \mathcal{P}_k^x} \mathbb{P}_k^x(x[k] \notin \mathcal{O}_i) &> 1 - \alpha_k/n_{total} \geq 1 - \alpha_k n_{ob_i}/n_{total}. \end{aligned}$$

当以下 2 个条件满足时, 约束成立。

$$\begin{aligned} \sup_{p_k^x \in \mathcal{P}_k^x} \mathbb{P}_k^x(a_{env,j}^T x[k] > b_{env,j}) &\leq \alpha_k/n_{total} \forall j; \\ \inf_{p_k^x \in \mathcal{P}_k^x} \mathbb{P}_k^x(a_{ob_i,j}^T x[k] > b_{ob_i,j}) &> 1 - \alpha_k/n_{total} \forall i, j. \end{aligned} \quad (6)$$

2 跟踪控制器

采用状态偏差与控制输入的二次型作为阶段成本函数, 确保所有控制器优化目标一致:

$$\begin{aligned} g_{ll}[k](\bar{x}[k], x[k], u[k]) &= \delta_x[k]^T Q[k] \delta_x[k] + u[k]^T R u[k]; \\ g_{ll}[T](\bar{x}[T], x[T]) &= \delta_x[T]^T Q[T] \delta_x[T]. \end{aligned} \quad (7)$$

式中: $\delta_x[k] = x[k] - \bar{x}[k]$ 为状态跟踪误差; 惩罚矩阵 $Q[k]$ 与 $R[k]$ 为对称正定矩阵($\forall k$)。

2.1 机器人动力学模型

通过对连续时间动力学进行前向欧拉离散化, 得到离散时间非线性动力学方程:

$$\begin{aligned} p_x[k+1] &= p_x[k] + \cos(\theta[k])v[k]\Delta t + w_x[k]\Delta t; \\ p_y[k+1] &= p_y[k] + \sin(\theta[k])v[k]\Delta t + w_y[k]\Delta t; \\ \theta[k+1] &= \theta[k] + \omega[k]\Delta t + w_\theta[k]\Delta t. \end{aligned} \quad (8)$$

式中 $p_x[k]$ 、 $p_y[k] \in \mathbb{R}$ 为机器人在地面坐标系中的水

平与垂直位置； $\theta[k] \in \mathbb{R}$ 为机器人相对于世界坐标系 x 轴的航向角； $v[k]$ 、 $\omega[k] \in \mathbb{R}$ 分别为线速度和角速度控制输入； $w_x[k]$ 、 $w_y[k]$ 、 $w_\theta[k]$ 为各状态量对应的扰动项； Δt 为时间戳 k 到 $k+1$ 的采样时间，s。

2.2 线性二次型调节器 (LQR)

LQR 是一种经典最优控制方法，其核心步骤为：

1) 系统线性化：

$$A[k] = \left. \frac{\partial f}{\partial x} \right|_{\bar{x}[k], \bar{u}[k]} ; B[k] = \left. \frac{\partial f}{\partial u} \right|_{\bar{x}[k], \bar{u}[k]} . \quad (9)$$

2) LQR 问题构建：最小化代价函数 J_{ll} ，约束条件为：

$$\delta_x[k+1] = A[k]\delta_x[k] + B[k]\delta_u[k]. \quad (10)$$

3) 离线求解：通过动态规划关系式预计算时变反馈参数序列：

$$\{K[k], L[k], e[k]\}_{k=0}^{T-1}. \quad (11)$$

4) 在线控制律：实时控制输入计算式为：

$$u[k] = K[k]\delta_x[k] + L[k] \begin{bmatrix} \bar{x}[k] \\ \bar{u}[k] \end{bmatrix} + e[k] + \bar{u}[k]. \quad (12)$$

2.3 鲁棒 LQR 控制

针对因过程扰动导致系统状态偏离参考轨迹而引起的状态空间矩阵线性化误差，提出以下鲁棒性增强方法：

雅可比矩阵边界构造：基于独轮车模型特性，可在 A 和 B 矩阵空间构建 2 维边界框以包含所有可能的雅可比矩阵。以 B 矩阵为例：

$$B = \begin{bmatrix} B_{11} \\ B_{21} \end{bmatrix} = \begin{bmatrix} \cos(\theta) \\ \sin(\theta) \end{bmatrix}. \quad (13)$$

鲁棒性区域定义：通过图 1 所示的三角关系，建立矩阵不确定性描述：

$$A[k] \in \{A : A = \bar{A}[k] + \mu_{A_1}[k]A_1[k] + \mu_{A_2}[k]A_2[k]\};$$

$$B[k] \in \{B : B = \bar{B}[k] + \mu_{B_1}[k]B_1[k] + \mu_{B_2}[k]B_2[k]\}. \quad (14)$$

方向矩阵与缩放因子：定义方向矩阵及其缩放边界：

$$A_1[k] = \begin{bmatrix} 0 & 0 & -s[k] \\ 0 & 0 & c[k] \\ 0 & 0 & 0 \end{bmatrix};$$

$$A_2[k] = \begin{bmatrix} 0 & 0 & -c[k] \\ 0 & 0 & -s[k] \\ 0 & 0 & 0 \end{bmatrix};$$

$$B_1[k] = \begin{bmatrix} -s[k] & 0 \\ c[k] & 0 \\ 0 & 0 \end{bmatrix};$$

$$B_2[k] = \begin{bmatrix} c[k] & 0 \\ s[k] & 0 \\ 0 & 0 \end{bmatrix}. \quad (15)$$

式中 $c[k] = \cos(\theta[k])$ ， $s[k] = \sin(\theta[k])$ ，且缩放因子满足 $|\mu_{A_1}[k]| \leq \sigma_{A_1}[k]$ ， $|\mu_{B_1}[k]| \leq \sigma_{B_1}[k]$ 。具体边界值计算式为：

$$\sigma_{A_1}[k] = v[k]\Delta t \sin(\delta\theta_{\max});$$

$$\sigma_{A_2}[k] = v[k]\Delta t [1 - \cos(\delta\theta_{\max})];$$

$$\sigma_{B_1}[k] = \sin(\delta\theta_{\max});$$

$$\sigma_{B_2}[k] = 1 - \cos(\delta\theta_{\max}). \quad (16)$$

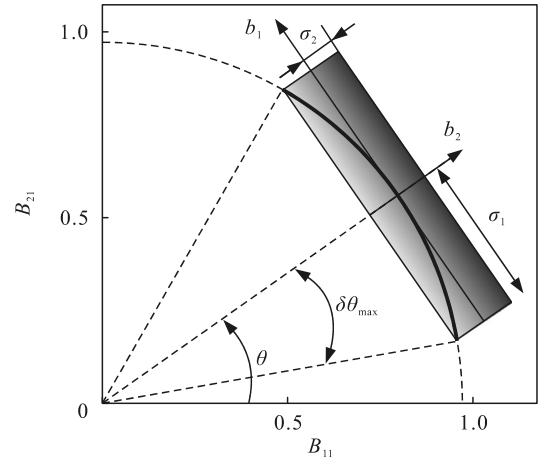


图 1 不同状态线性化下的 B 矩阵几何关系

2.4 非线性模型预测控制 (NMPC)

定义 (NMPC 跟踪 NLP)：给定参考轨迹 $\bar{x}[k], k \in [0:T]$ 和规划时域 N_{ll} ，在时间步骤 t 求解满足约束的最小化问题：

$$\min_{u[t:T+N_{ll}-1]} \sum_{k=t}^{t+N_{ll}-1} g_{ll}[k](\bar{x}[k], x[k], u[k]) +$$

$$g_{ll}[T](\bar{x}[T], x[T]), x[k] \in \mathcal{X};$$

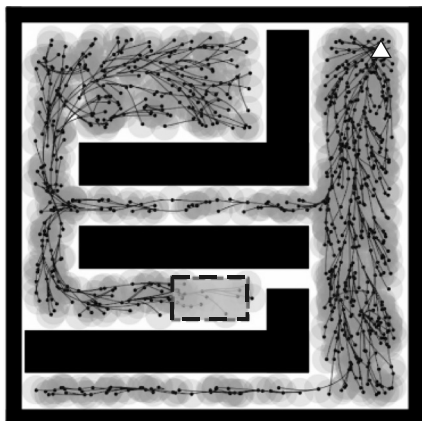
$$x[k+1] = f(x[k], u[k]). \quad (17)$$

最终的控制输入集合是 $u_{0:T-1}^* = [u[0], \dots, u[T-1]]$ 。

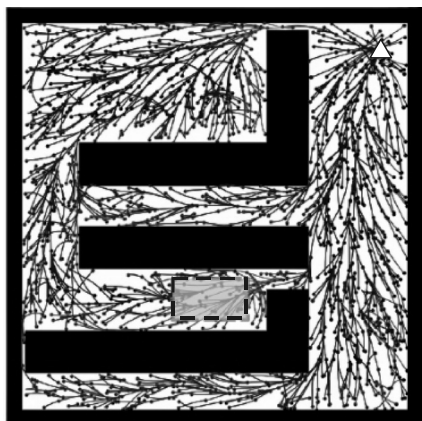
3 数值仿真结果

RANS-RRT*和底层蒙特卡洛仿真的硬件配置分别为 Intel Core i7 6700K CPU/16GB RAM 和 Ryzen 7 2700X/64GB RAM，NLP 建模和求解器的软件配置为 CasADi Opti 和 IPOPT^[13]。如图 2 所示

本环境参数为根节点(白色三角形)、目标区域(虚线矩形)、 10×10 环境: 4 个边界障碍(黑色边框)+5 个矩形障碍(黑色实体)。机器人被视为质点。它的线速度边界为 ± 0.5 s, 角速度边界为 $\pm \pi$ rad/s, RANS-RRT* 转向时域 N 为 30 并且 NMPC 规划时域 N_H 为 10, 离散时间步长 δt 为 0.2 s。规划器控制代价为 $R[k] = \text{diag}([1, 1])$ 。跟踪控制器(LQR/鲁棒 LQR/NMPC)代价为 $Q[k] = \text{diag}([100, 100, 10])$, $R[k] = \text{diag}([1, 1]) (\forall k \in [0: T-1])$, 并且 $Q[T] = 10Q[0]$ 。



(a) RANS-RRT* 树有 942 个节点



(b) RRT* 树有 1 284 个节点

图 2 RANS-RRT 与标准 RRT 算法生成的树结构对比

采用高层规划风险边界 $\beta=0.1$, 该风险值均分至 $T_{\max}=1\ 000$ 个时间步及各障碍物约束。过程噪声分布 $\mathbb{P}_k^w$ 采用零均值多元拉普拉斯分布, 其协方差矩阵 $\Sigma^w = \sigma_\omega^2 I_n$, 其中噪声水平 $\sigma_\omega^2 = 5 \times 10^{-7}$ 。在轨迹跟踪阶段, 采用了更高噪声水平下评估了该算法性能。

图 2(a)所示 RANS-RRT 树的构建耗时约 28 min。算法初始随机采样 2 000 个节点, 但仅有 865 个节点通过可行性及安全性检验并被加入树中。作为对比, 相同环境下采用标准 RRT* 算法构建的树结构图 2(b)会探索此类高风险通道, 从而可能生成不符合安全要求的轨迹。

4 结论

笔者提出一种面向随机非线性机器人系统安全控制的风险规避控制架构。该架构包含新型 RANS-RRT 算法和底层参考轨迹跟踪控制器 2 个核心组件。新型 RANS-RRT 算法是基于非线性转向机制的 RRT 改进版本, 通过分布鲁棒碰撞检测实现风险感知; 底层参考轨迹跟踪控制器通过独轮车动力学模型的系统实验。笔者对 3 种控制器进行了对比测试, 结果表明非线性模型预测控制器(NMPC)的性能优于各类 LQR 改进方案。

本文中证实, 尽管高层规划器采用了极小的噪声水平假设, 底层控制器在中度和强干扰环境下仍表现出色。未来研究将着重从建立完整的非线性传感器模型、在自主系统各层级的优化问题中严格集成分布鲁棒(DR)风险约束, 以实现精确的风险评估和通过代码生成工具缩短 NMPC 的计算耗时进行。

参考文献:

- [1] 张广林, 胡小梅, 柴剑飞, 等. 路径规划算法及其应用综述[J]. 现代机械, 2011(5): 85-90.
- [2] 鲁宇明, 周羽达, 郭鑫, 等. 改进 Informed RRT* 算法移动机器人路径规划[J]. 计算机工程与应用, 2025, 61(8): 283-293.
- [3] 朱大奇, 颜明重. 移动机器人路径规划技术综述[J]. 控制与决策, 2010, 25(7): 961-967.
- [4] 王旭, 朱其新, 朱永红. 面向二维移动机器人的路径规划算法综述[J]. 计算机工程与应用, 2023, 59(20): 51-66.
- [5] BLACKMORE L, ONO M, WILLIAMS B C, et al. Chance-constrained optimal path planning with obstacles[J]. IEEE Transactions on Robotics, 2011, 27(5): 1080-1094.
- [6] ZHU H, ALONSO-MORA J. Chance-constrained collision avoidance for mavs in dynamic environments[J]. IEEE Robotics and Automation Letters, 2019, 4(2): 776-783.
- [7] HAKOBYAN A, KIM G C, YANG I. Risk-aware motion planning and control using CVaR-constrained optimization[J]. IEEE Robotics and Automation Letters, 2019, 4(4): 3924-3931.
- [8] HAKOBYAN A, YANG I. Wasserstein distributionally robust motion planning and control with safety constraints using conditional value-at-risk[C]//2020 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2020: 490-496.
- [9] KARAMAN S, FRAZZOLI E. Sampling-based algorithms for optimal motion planning[J]. arXiv, 2011, 30(7): 846-894.