

基于局部子地图方法的多机器人主动同时定位与地图创建

苑 晶¹, 黄亚楼², 陶 通², 习白羽²

(1. 南开大学信息技术科学学院, 天津 300071; 2. 南开大学软件学院, 天津 300071)

摘要: 研究了多机器人在未知环境下以主动的方式协作完成同时定位与地图创建 (SLAM) 的问题. 引入局部子地图方法, 由每个机器人建立自身周围局部区域的子地图, 使多个机器人之间的地图创建相互独立, 从而对全局环境的 SLAM 问题进行分解. 而每个机器人在建立局部子地图时将主动 SLAM 问题转化为多目标优化问题; 机器人选取最优的控制输入, 使定位与地图创建的准确性、信息增益以及多机器人之间的协调关系得到综合优化. 最后, 通过扩展的卡尔曼滤波器 (EKF) 对子地图进行融合得到全局地图. 仿真结果验证了该方法的有效性.

关键词: 局部子地图; 多机器人协作; 主动同时定位与地图创建

中图分类号: TP24

文献标识码: A

Multi-robot Active Simultaneous Localization and Mapping Based on Local Submap Approach

YUAN Jing¹, HUANG Ya-lou², TAO Tong², XI Bai-yu²

(1. College of Information Technical Science, Nankai University, Tianjin 300071, China;

2. College of Software, Nankai University, Tianjin 300071, China)

Abstract: The cooperative simultaneous localization and mapping (SLAM) finished in an active way by multiple robots in unknown environment is investigated. Local submap strategy is introduced in which each robot carries out map building in the local area around itself so that it can build local submap independent of other robots, then the SLAM problem in global environment can be decomposed into multiple sub-problems. The problem of active SLAM is converted into that of multi-objective optimization when each robot builds its local submap. The robot chooses the optimal control inputs so that the accuracy of localization and mapping, information gain and the cooperative motion will be synthetically optimized. Finally, all the submaps are fused into global map by extended Kalman filter (EKF). Simulation results prove the effectiveness of the presented approach.

Keywords: local submap; multi-robot cooperation; active simultaneous localization and mapping

1 引言 (Introduction)

随着导航问题研究的不断深入和移动机器人应用领域的不断扩展, 同时定位与地图创建 (SLAM) 问题逐渐成为移动机器人领域内备受关注的热点问题.

与单机器人的 SLAM 相比, 多机器人协作完成的 SLAM 具有更准确、更高效和更鲁棒的优势. 然而, 目前有关多机器人 SLAM 问题的研究成果相对较少. 文 [1] 基于 EKF 研究了基于全局地图估计的多机器人 SLAM 问题, 由多个机器人共同维护全局地图的状态及其协方差矩阵, 并利用观测进行全局更新; 文 [2] 采用增量极大似然法估计机器人位姿,

并构造增量地图, 同时利用由蒙特卡洛方法计算出的机器人位姿的后验概率对增量地图进行校正, 然后将该方法应用到多机器人建图中; 文 [3] 将 EKF 的改进算法 SEIF (sparse extended information filter) 应用于多机器人 SLAM 中. 但在上述几种方法中, 多个机器人的运动路径都是事先指定的, 机器人进入环境后跟踪各自的指定路径运动, 完成观测和建图, 而在此过程中多个机器人之间缺乏充分有效的协调. 现有研究中仅有 Fox 基于粒子滤波器研究了结构化环境下的多机器人主动 SLAM 问题 [4,5], 通过决策理论为多机器人规划各自最优的目标点, 使所有决策行为的期望增益尽可能大, 而期望代价尽可能小. 然而该研究仅考虑了多机器人定位的不确

定性,并未对建图的不确定性进行处理.

Williams 基于 EKF 提出了局部子地图 SLAM 方法^[6],又称受约束的局部子地图滤波器(constrained local submap filter, CLSF).在局部子地图方法中,机器人只维护一个由自身及其周围某一局部区域内的路标组成的子地图,并且子地图与全局地图之间是相互独立的,因此机器人每次只需对子地图进行更新,然后周期性地将子地图融合到全局地图中,由于子地图仅包含一小部分路标,因此算法复杂度大大降低;同时,Williams 将该方法应用到多机器人 SLAM 中^[6,7],为多个机器人分别指定运动路径,使它们跟踪各自的路径完成协作建图.但由于多个机器人的运动路径是事先指定的,每个机器人在建图过程中仅跟踪各自的指定路径进行环境探索,因此缺乏与其它机器人之间的在线协调.

本文基于局部子地图思想,提出多机器人主动 SLAM 方法,每个机器人在建立自身局部子地图时采用主动探索的策略,使其运动兼顾定位与地图创建的准确性、信息增益以及与其它机器人之间的协调,从而得到更准确、高效的建图结果.

2 局部子地图方法 (Local submap approach)

2.1 系统状态定义

在局部子地图方法中,系统状态包括四部分:子地图相对于全局地图的位置关系、全局地图中的路标位置、子地图中机器人的位姿和子地图中路标的位置,定义为:

$$\mathbf{X}_s(k) = \begin{bmatrix} \mathbf{G}_{X_L}^T(k) & \mathbf{G}_{X_m}^T(k) & \mathbf{L}_{X_r}^T(k) & \mathbf{L}_{X_m}^T(k) \end{bmatrix}^T \quad (1)$$

其中, $\mathbf{G}_X(k) = [\mathbf{G}_{X_L}^T(k), \mathbf{G}_{X_m}^T(k)]^T$, 为全局坐标系下的系统状态; $\mathbf{L}_X(k) = [\mathbf{L}_{X_r}^T(k), \mathbf{L}_{X_m}^T(k)]^T$, 为局部坐标系下的系统状态; $\mathbf{G}_{X_L}(k)$ 表示子地图坐标系 F_L 在全局坐标系 F_G 中的位姿; $\mathbf{L}_{X_r}(k) = [L_{x_r}(k), L_{y_r}(k), L_{\theta_r}(k)]^T$, 表示机器人在局部子地图中的位姿; $\mathbf{G}_{X_m}(k) = [\mathbf{G}_{X_1}^T(k), \mathbf{G}_{X_2}^T(k) \cdots]^T$ 和 $\mathbf{L}_{X_m}(k) = [\mathbf{L}_{X_1}^T(k), \mathbf{L}_{X_2}^T(k) \cdots]^T$ 分别表示全局地图和子地图, $\mathbf{G}_{X_i}(k) = [G_{x_i}(k), G_{y_i}(k)]^T$

和 $\mathbf{L}_{X_i}(k) = [L_{x_i}(k), L_{y_i}(k)]^T (i = 1, 2, \dots)$ 分别表示路标 l_i 在全局地图和子地图中的位置.局部子地图与全局地图之间的映射是通过 $\mathbf{G}_{X_L}(k)$ 实现的,将局部子地图融合到全局地图时,利用 $\mathbf{G}_{X_L}(k)$ 对局部子地图的状态和协方差矩阵进行变换,而机器人在局部子地图中进行 SLAM 时,子地图和全局地图之间是无关的.

系统状态的协方差表示为:

$$\mathbf{P}_s(k|k) = \begin{bmatrix} \mathbf{G}_P(k|k) & \mathbf{0} \\ \mathbf{0} & \mathbf{L}_P(k|k) \end{bmatrix} = \begin{bmatrix} \mathbf{G}_{P_{LL}}(k|k) & \mathbf{G}_{P_{Lm}}(k|k) & \mathbf{0} & \mathbf{0} \\ \mathbf{G}_{P_{Lm}}^T(k|k) & \mathbf{G}_{P_{mm}}(k|k) & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{L}_{P_{rr}}(k|k) & \mathbf{L}_{P_{rm}}(k|k) \\ \mathbf{0} & \mathbf{0} & \mathbf{L}_{P_{rm}}^T(k|k) & \mathbf{L}_{P_{mm}}(k|k) \end{bmatrix} \quad (2)$$

其中 $\mathbf{G}_{P_{LL}}(k|k)$ 、 $\mathbf{G}_{P_{Lm}}(k|k)$ 和 $\mathbf{G}_{P_{mm}}(k|k)$ 分别表示全局地图中子地图坐标系、子地图坐标系与路标、路标与路标之间的协方差, $\mathbf{L}_{P_{rr}}(k|k)$ 、 $\mathbf{L}_{P_{rm}}(k|k)$ 和 $\mathbf{L}_{P_{mm}}(k|k)$ 分别表示子地图中机器人位姿、机器人与路标、路标与路标之间的协方差.

2.2 基于 EKF 的系统状态估计

假设在子地图中机器人的运动模型为:

$$\mathbf{L}_{X_r}(k+1) = \mathbf{f}(\mathbf{L}_{X_r}(k), \mathbf{u}(k), \mathbf{v}(k)) \quad (3)$$

其中 $\mathbf{v}(k)$ 为零均值高斯白噪声,其协方差矩阵为 $\mathbf{Q}(k)$.

传感器观测模型为:

$$\mathbf{Z}(k+1) = \mathbf{h}(\mathbf{L}_{X_r}(k+1), \mathbf{L}_{X_m}(k+1)) + \mathbf{w}(k+1) \quad (4)$$

其中 $\mathbf{w}(k)$ 为零均值高斯白噪声,其协方差矩阵为 $\mathbf{R}(k)$.

对于激光传感器,当机器人观测到某个路标 l_i 时,有

$$\mathbf{h}(\mathbf{L}_{X_r}(k+1), \mathbf{L}_{X_m}(k+1)) = \begin{bmatrix} \sqrt{(L_{x_r}(k+1) - L_{x_i}(k+1))^2 + (L_{y_r}(k+1) - L_{y_i}(k+1))^2} \\ \arctan \frac{L_{y_r}(k+1) - L_{y_i}(k+1)}{L_{x_r}(k+1) - L_{x_i}(k+1)} - L_{\theta_r}(k+1) \end{bmatrix} \quad (5)$$

与全局地图估计方法相同,局部子地图方法的系统状态估计同样分为预测与更新两步.由于局部

子地图与全局地图相互独立,因此预测与更新均只针对子地图进行操作.

预测:

$$\hat{\mathbf{X}}_s(k+1|k) = \begin{bmatrix} \mathbf{G}_{\hat{x}_L}(k+1|k) \\ \mathbf{G}_{\hat{x}_m}(k+1|k) \\ \mathbf{L}_{\hat{x}_r}(k+1|k) \\ \mathbf{L}_{\hat{x}_m}(k+1|k) \end{bmatrix} = \begin{bmatrix} \mathbf{G}_{\hat{x}_L}(k|k) \\ \mathbf{G}_{\hat{x}_m}(k|k) \\ \mathbf{f}(\mathbf{L}_{\hat{x}_r}(k|k), \mathbf{u}(k)) \\ \mathbf{L}_{\hat{x}_m}(k|k) \end{bmatrix} \quad (6)$$

$$\hat{\mathbf{Z}}(k+1|k) = \mathbf{h}(\mathbf{L}_{\hat{x}_r}(k+1|k), \mathbf{L}_{\hat{x}_m}(k+1|k)) \quad (7)$$

$$\mathbf{P}_s(k+1|k) = \begin{bmatrix} \mathbf{G}_P(k+1|k) & \mathbf{0} \\ \mathbf{0} & \mathbf{L}_P(k+1|k) \end{bmatrix} = \begin{bmatrix} \mathbf{G}_{P_{LL}}(k|k) & \mathbf{G}_{P_{Lm}}(k|k) & \mathbf{0} & \mathbf{0} \\ \mathbf{G}_{P_{Lm}}^T(k|k) & \mathbf{G}_{P_{mm}}(k|k) & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \nabla_r \mathbf{f} \cdot \mathbf{L}_{P_{rr}}(k|k) \cdot \nabla_r \mathbf{f}^T + \nabla_v \mathbf{f} \cdot \mathbf{Q}(k) \cdot \nabla_v^T \mathbf{f} & \nabla_r \mathbf{f} \cdot \mathbf{L}_{P_{rm}}(k|k) \\ \mathbf{0} & \mathbf{0} & (\nabla_r \mathbf{f} \cdot \mathbf{L}_{P_{rm}}(k|k))^T & \mathbf{L}_{P_{mm}}(k|k) \end{bmatrix} \quad (8)$$

其中 $\nabla_r \mathbf{f} = \left. \frac{\partial \mathbf{f}}{\partial \mathbf{L}_{X_r}} \right|_{\mathbf{L}_{X_r} = \mathbf{L}_{\hat{x}_r}(k|k)}$ 和 $\nabla_v \mathbf{f} = \left. \frac{\partial \mathbf{f}}{\partial \mathbf{v}} \right|_{\mathbf{v}=\mathbf{0}}$ 分别表示 $\mathbf{f}(\cdot)$ 相对于机器人位姿 $\mathbf{L}_{X_r}(k)$ 和运动噪声向量 $\mathbf{v}(k)$ 的雅可比矩阵.

更新:

$$\hat{\mathbf{X}}_s(k+1|k+1) = [\mathbf{G}_{\hat{x}}^T(k|k), (\mathbf{L}_{\hat{x}}(k+1|k) + \mathbf{W}(k+1) \cdot (\mathbf{Z}(k+1) - \hat{\mathbf{Z}}(k+1|k)))^T]^T \quad (9)$$

$$\mathbf{P}_s(k+1|k+1) = \begin{bmatrix} \mathbf{G}_P(k|k) & \mathbf{0} \\ \mathbf{0} & \mathbf{L}_P(k+1|k) - \mathbf{W}(k+1) \cdot \mathbf{S}(k+1) \cdot \mathbf{W}^T(k+1) \end{bmatrix} \quad (10)$$

$$\mathbf{S}(k+1) = \nabla \mathbf{h} \cdot \mathbf{L}_P(k+1|k) \cdot \nabla \mathbf{h}^T + \mathbf{R}(k+1) \quad \mathbf{W}(k+1) = \mathbf{L}_P(k+1|k) \cdot \nabla \mathbf{h}^T \cdot \mathbf{S}^{-1}(k+1) \quad (11)$$

其中 $\nabla \mathbf{h} = \left. \frac{\partial \mathbf{h}}{\partial (\mathbf{L}_{X_r}, \mathbf{L}_{X_m})} \right|_{\mathbf{L}_{X_r} = \mathbf{L}_{\hat{x}_r}(k+1|k), \mathbf{L}_{X_m} = \mathbf{L}_{\hat{x}_m}(k+1|k)}$ 表示 $\mathbf{h}(\cdot)$ 相对于机器人和路标 l_i 的雅可比矩阵.

2.3 局部子地图与全局地图之间的变换

子地图坐标系在全局地图中的位姿估计 $\mathbf{G}_{\hat{x}_L}(k|k)$ 建立了从子地图坐标系 F_L 到全局坐标系 F_G 的变换关系 $\mathbf{T}(k)$ ^[6], 使得子地图中的状态估计及相应的协方差可以映射到全局地图中.

$$\mathbf{G}_{\hat{x}_s}(k|k) = \mathbf{T}(k) \cdot \hat{\mathbf{X}}_s(k|k) = \begin{bmatrix} \mathbf{G}_{\hat{x}_L}(k|k) \\ \mathbf{G}_{\hat{x}_m}(k|k) \\ \mathbf{G}_{\hat{x}_L}(k|k) \oplus \mathbf{L}_{\hat{x}_r}(k|k) \\ \mathbf{G}_{\hat{x}_L}(k|k) \oplus \mathbf{L}_{\hat{x}_m}(k|k) \end{bmatrix} \quad (12)$$

$$\mathbf{G}_P(k|k) = \nabla \mathbf{T}(k) \cdot \mathbf{P}_s(k|k) \cdot (\nabla \mathbf{T}(k))^T \quad (13)$$

其中 \oplus 表示向量之间的复合运算^[6], $\nabla \mathbf{T}(k)$ 的计算可参见文 [6].

2.4 局部子地图与全局地图之间的融合

文 [6] 给出了一种局部子地图与全局地图融合的方法, 使用路标的约束关系进行融合, 即对于同

一个路标, 全局地图中的估计和局部子地图中的估计应该是等价的, 从而构成了约束关系. 而在此, 本文给出了一种基于 EKF 的融合方法, 将同一个路标在局部子地图和全局地图中的两个不同的估计结果进行融合.

局部子地图到全局地图的映射关系为 $\mathbf{T}(k)$, 反之, 由全局地图到子地图的映射为 $\mathbf{T}^{-1}(k)$, 则:

$$\mathbf{S}(k+1) = \nabla (\mathbf{T}^{-1}) \cdot \mathbf{G}_P(k|k) \cdot (\nabla (\mathbf{T}^{-1}))^T + \mathbf{L}_P(k+1|k+1)$$

$$\mathbf{W}(k+1) = \mathbf{G}_P(k|k) \cdot (\nabla (\mathbf{T}^{-1}))^T \cdot \mathbf{S}^{-1}(k+1)$$

$$\mathbf{G}_{\hat{x}}(k+1|k+1) = \mathbf{G}_{\hat{x}}(k|k) + \mathbf{W}(k+1) \cdot (\mathbf{L}_{\hat{x}}(k+1|k+1) - \mathbf{T}_L^{-1} \cdot \mathbf{G}_{\hat{x}}(k|k))$$

$$\mathbf{G}_P(k+1|k+1) = \mathbf{G}_P(k|k) - \mathbf{W}(k+1) \cdot \mathbf{S}(k+1) \cdot \mathbf{W}^T(k+1) \quad (14)$$

其中, $\mathbf{L}_{\hat{x}}(k+1|k+1) = [\mathbf{L}_{\hat{x}_r}^T(k+1|k+1) \quad \mathbf{L}_{\hat{x}_m}^T(k+1|k+1)]^T$, 表示局部子地图中的状态估计值, \mathbf{T}_L 为 \mathbf{T} 中对应于 $\mathbf{L}_{\hat{x}}$ 的右下角子阵.

3 基于局部子地图的多机器人主动 SLAM (Multi-robot active SLAM using local submap)

文 [6] 证明了局部子地图与全局地图之间是相互独立的, 因此可以将多机器人 SLAM 分解为多个单机器人 SLAM 的子问题, 当满足一定条件时将多个机器人的局部子地图融合到全局地图中, 实现全局地图的更新.

对于单机器人 SLAM 问题, 已有很多解决方法, 但是在现有研究中机器人通常采用随机方式^[8,9]或预先指定路径的方式^[3,10]探索环境. 对于前者, 机器人随机选取下一步的控制输入, 因此建图存在盲目性的缺陷; 对于后者, 假设在机器人进入环境前已经离线规划出了一条简单路径(贯穿于环境的直线或圆), 机器人进入环境后始终跟踪该路径, 同时完成自身定位与地图创建. 然而, 这种假设在很大程度上与环境的未知性相矛盾, 且机器人跟踪预先指定的路径创建地图时, 不能保证环境中的所有路标都被观测到, 因此创建的地图是不完全的. 为此, 相关研究者提出了基于 EKF 的自适应同时定位与地图创建方法^[11], 即机器人每一步执行的控制输入能将地图创建中的不确定性最小化, 这样机器人的运动考虑了当前定位情况和地图创建程度, 对环境具有较好的自适应性. 但这种方法假设机器人每一时刻都能够观测到环境中的所有路标, 因此不适用于路标数目较多的大规模环境.

本文基于局部子地图策略, 提出一种新的多机器人协作主动 SLAM 方法, 将主动 SLAM 转化为一个最优控制问题进行求解. 由于子地图的创建过程与全局地图无关, 所有机器人均独立完成各自子地图的建立, 因此每个机器人可以独立地选取最优控制输入, 当机器人建立的局部子地图的不确定性小于给定阈值, 且建立局部子地图的步数大于给定阈值时, 将它们融合到全局地图.

3.1 系统状态定义

考虑 M 个机器人 $r_j (j = 1, 2 \dots M)$ 组成的多机器人系统, 系统状态表示为:

$$\bar{\mathbf{X}}_s(k) = [\mathbf{G}_{X_{L_1}}^T(k), \dots, \mathbf{G}_{X_{L_M}}^T(k), \mathbf{G}_{X_m}^T(k), \mathbf{L}_{X_1}^T(k), \dots, \mathbf{L}_{X_M}^T(k)]^T \quad (15)$$

其中 $\mathbf{L}_{X_j}(k) = [\mathbf{L}_{X_{r_j}}^T(k), \mathbf{L}_{X_{m_j}}^T(k)]^T (j = 1, 2 \dots M)$.

系统状态的协方差可表示为:

$$\bar{\mathbf{P}}_s(k|k) = \begin{bmatrix} \bar{\mathbf{G}}_P(k|k) & \mathbf{0} \\ \mathbf{0} & \bar{\mathbf{L}}_P(k|k) \end{bmatrix} \quad (16)$$

其中

$$\bar{\mathbf{G}}_P(k|k) = \begin{bmatrix} \mathbf{G}_{P_{L_1, L_1}}(k|k) & \dots & \mathbf{G}_{P_{L_1, L_M}}(k|k) & \mathbf{G}_{P_{L_1, m}}(k|k) \\ \vdots & & \ddots & \vdots \\ \mathbf{G}_{P_{L_1, L_M}}^T(k|k) & \dots & \mathbf{G}_{P_{L_M, L_M}}(k|k) & \mathbf{G}_{P_{L_M, m}}(k|k) \\ \mathbf{G}_{P_{L_1, m}}^T(k|k) & \dots & \mathbf{G}_{P_{L_M, m}}^T(k|k) & \mathbf{G}_{P_{m, m}}(k|k) \end{bmatrix}$$

$$\bar{\mathbf{L}}_P(k|k) = \text{diag}\{\mathbf{L}_{P_1}(k|k), \dots, \mathbf{L}_{P_M}(k|k)\}$$

$$\mathbf{L}_{P_j}(k|k) = \begin{bmatrix} \mathbf{L}_{P_{j, irr}}(k|k) & \mathbf{L}_{P_{j, mm}}(k|k) \\ \mathbf{L}_{P_{j, mm}}^T(k|k) & \mathbf{L}_{P_{j, mm}}(k|k) \end{bmatrix} (j = 1, 2 \dots M)$$

3.2 优化目标

在探索过程中, 每个机器人均采用前述局部子地图方法建立和维护自己的局部子地图, 而地图创建所采用的运动方式是主动的, 即每个机器人在子地图中通过优化目标函数选取控制输入. 目标函数包含三部分, 即在子地图中的定位与地图创建的准确性、信息增益和多机器人之间的协调项.

对于第 j 个机器人 r_j , 利用其位姿协方差椭球的体积和路标位置协方差椭圆的面积来描述定位与地图创建的准确性^[11]:

$$C(\mathbf{L}_{P_j}(k+1|k+1)) = \pi \prod_{l=1}^3 \sqrt{\lambda_l(\mathbf{L}_{P_{j, irr}}(k+1|k+1))} + \pi \sum_{i=1}^{N_j} \prod_{l=1}^2 \sqrt{\lambda_l(\mathbf{L}_{P_{j, ii}}(k+1|k+1))} = \pi \sqrt{\det(\mathbf{L}_{P_{j, irr}}(k+1|k+1))} + \pi \sum_{i=1}^{N_j} \sqrt{\det(\mathbf{L}_{P_{j, ii}}(k+1|k+1))} \quad (17)$$

其中, $\mathbf{L}_{P_{j, irr}}(k+1|k+1)$ 表示机器人位姿的协方差; $\mathbf{L}_{P_{j, ii}}(k+1|k+1)$ 为 $\mathbf{L}_{P_{j, mm}}(k+1|k+1)$ 主对角线上的 2×2 子阵, 表示每个路标二维位置坐标的协方差; 而 $\lambda_l(\mathbf{L}_{P_{j, irr}}(k+1|k+1))$ 和 $\lambda_l(\mathbf{L}_{P_{j, ii}}(k+1|k+1))$ 分别表示 $\mathbf{L}_{P_{j, irr}}(k+1|k+1)$ 和 $\mathbf{L}_{P_{j, ii}}(k+1|k+1)$ 的特征根; N_j 表示 r_j 已观测到的路标数目. 而 $\mathbf{L}_{P_j}(k+1|k+1)$ 由 (18) 式进行计算:

$$\mathbf{L}_{P_j}(k+1|k+1) = [\mathbf{L}_{P_j}^{-1}(k+1|k) + (\nabla \mathbf{h}_j)^T \cdot \mathbf{R}^{-1}(k+1) \cdot \nabla \mathbf{h}_j]^{-1} \quad (18)$$

其中, $\nabla \mathbf{h}_j = \left. \frac{\partial \mathbf{h}}{\partial (\mathbf{L}_{X_{r_j}}, \mathbf{L}_{X_{m_j}})} \right|_{\mathbf{L}_{X_{r_j}} = \mathbf{L}_{\hat{X}_{r_j}}(k+1|k), \mathbf{L}_{X_{m_j}} = \mathbf{L}_{\hat{X}_{m_j}}(k+1|k)}$.

考虑优化目标的第 2 项—信息增益, 由于机器人传感器探测范围有限, 因此通常情况下, 机器人无法同时观测到环境中的所有路标. 因此, 若仅以机器人定位和地图创建准确性作为目标函数, 机器人很可能陷入某个局部区域进行探索, 因为此时

机器人仅能观测到周围的几个路标, 为使当前定位和对这几个路标的位置估计更准确, 机器人将始终徘徊于这些路标周围进行观测, 而无法向未探索区域运动. 为此, 借助文 [12] 的方法, 在机器人探索环境的过程中计算已探索区域和未探索区域的边界 (frontier), 并以此计算机器人能够探索的新区域面积, 作为机器人的信息增益引入到目标函数中, 促使机器人为获得更多的环境信息而同时向未探索区域运动.

采用占用栅格方法表示环境, 根据文 [12], 与未探索栅格相邻的空白栅格表示已探索区域和未探索区域之间的边界. 机器人从已探索区域不断向边界运动, 可以探索到更多的未知区域. 基于已探索区域和未探索区域的边界, 可以计算出 r_j 预期探索的新区域面积 $A_j(k+1|k)$, 即机器人第 k 步执行某一控制输入后第 $k+1$ 步所能探测到的之前未探索栅格的面积的预测值.

$$A_j(k+1|k) = A_g \cdot N_{g,j}(k+1|k) \quad (19)$$

其中, A_g 为每个栅格的面积, $N_{g,j}(k+1|k)$ 为预测的机器人 r_j 执行控制输入后传感器能够探测到的新栅格数目.

最后, 考虑目标函数的第 3 项—多机器人之间的协调项. 与单机器人 SLAM 相比, 多机器人协作完成 SLAM 的一个重要优势就是多机器人之间可以通过合理的协调高效地完成地图创建, 但当多机器人之间没有协调或协调不合理时会阻碍上述优势的发挥. 例如两个机器人共同完成 SLAM, 若其中一个机器人跟随另一个机器人以相同的运动路径完成地图创建, 则地图创建效率与单机器人的情况相同. 因此, 不同机器人应该选择各自合理的运动方式, 以充分发挥多机器人协作的优势.

机器人在选择运动方式时, 应该尽量避免进入其他机器人的局部子地图, 否则会导致多个机器人探索同一局部区域. 由于多机器人建立的局部子地图之间相互独立, 因此一般情况下, 多个机器人对同一局部区域进行地图创建并不能明显提高定位与地图创建的准确性, 这一点与基于全局地图估计的方法是不同的. 所以应该尽可能避免多个机器人建立的子地图发生重合, 以充分提高多机器人地图创建的效率.

机器人 r_j 与机器人 $r_i (i \neq j)$ 之间的协调体现为 r_j 选择的控制输入应使它远离 r_i 的子地图 $M_i(k|k)$, 本文用 r_j 相对于 $M_i(k|k)$ 边界的最短距离 $d_{r_j, M_i}(k|k)$ 表示 r_j 远离 $M_i(k|k)$ 的程度. 当 r_j 位于子

地图 $M_i(k|k)$ 外部时, $d_{r_j, M_i}(k|k) > 0$; 当 r_j 位于子地图 $M_i(k|k)$ 内部时, $d_{r_j, M_i}(k|k) < 0$.

将 $d_{r_j, M_i}(k)$ 引入到机器人 r_j 的目标函数中, 作为惩罚项, 即若机器人 r_j 因执行某一控制输入而进入机器人 r_i 的子地图 $M_i(k|k)$, 则 r_j 的目标函数会明显减小, 因此优化协调项能使 r_j 尽量避免执行上述控制输入.

3.3 多机器人主动 SLAM

将上述三个优化目标分别进行归一化处理, 然后进行融合, 得到如下目标函数:

$$J(\mathbf{u}_j(k)) = \omega_1 \cdot \frac{C(\mathbf{L}_{P_{\min}})}{C(\mathbf{L}_{P_j}(k+1|k+1))} + \omega_2 \cdot \frac{A_j(k+1|k)}{A_{\max}} + \omega_3 \cdot \frac{1}{M-1} \cdot \sum_{i \neq j} \frac{d_{r_j, M_i}(k+1|k)}{d_t} \quad (j=1, 2, \dots, M) \quad (20)$$

其中, $\mathbf{L}_{P_{\min}}$ 表示定位与地图创建协方差的下界^[8], A_{\max} 表示机器人传感器最大探测范围的面积. d_t 为指定阈值, 只有当机器人 r_j 与子地图 $M_i(k|k)$ 之间的距离较近时, 才考虑在目标函数中引入协调项, 即当 $d_{r_j, M_i}(k+1|k) \leq d_t$ 时, 目标函数的第 3 项才起作用, 否则不再考虑第 3 项.

在式 (20) 中, $d_{r_j, M_i}(k+1|k)$ 表示机器人 r_j 在第 k 步预测执行控制输入 $\mathbf{u}_j(k)$ 后在第 $k+1$ 步距离 $M_i(k|k)$ 边界的最短距离, 在此使用 $M_i(k|k)$ 计算 $d_{r_j, M_i}(k+1|k)$ 而不使用 $M_i(k+1|k)$ 进行计算是因为后者需要考虑 $\mathbf{u}_i(k)$, 导致 $\mathbf{u}_j(k)$ 和 $\mathbf{u}_i(k)$ 的计算相互耦合. 式 (20) 中的 $d_{r_j, M_i}(k+1|k)$ 利用 $M_i(k|k)$ 进行计算, 使多个机器人可以独立地计算各自的最优控制输入, 从而大大简化目标函数的求解.

则 r_j 在当前步的最优控制输入为:

$$\mathbf{u}_j^*(k) = \arg \max_{\mathbf{u}_j(k) \in \mathbf{U}_{\text{feasible}}} J(\mathbf{u}_j(k)) \quad (21)$$

其中 $\mathbf{U}_{\text{feasible}}$ 表示机器人的可选控制输入集.

当执行式 (21) 时, 可能出现陷入局部区域的情况, 即机器人当前定位与地图创建足够准确, 但传感器无法探测到新的未知区域, 此时采用如下方法避免陷入局部探索, 即当 $C(\mathbf{L}_{P_j}) / (N_j + 1) \leq C_t$ 且 $A_j = 0$ 时, 让 r_j 朝着当前距离它最近的边界栅格运动, 其中 N_j 表示当前 r_j 在其局部子地图中观测过的路标数目, C_t 为给定阈值. 若上述两个条件至少有一个不满足, 则继续执行式 (21).

由于本文给出的多机器人 SLAM 方法是基于局部子地图策略实现的, 在地图创建过程中多个机器人的探索相互独立, 因此当某个机器人的运动或

观测受到更大噪声扰动时, 只有对应于该机器人的子地图的准确性受到影响. 而当该子地图融合到全局地图时, 由于本文采用了 EKF 进行融合, 由 EKF 的特性可知, 对同一状态 x 的两个不同估计结果 $\hat{x}_i (i = 1, 2)$ 进行融合, 所得到的估计值 \hat{x} 的不确定性总小于融合前估计值 $\hat{x}_i (i = 1, 2)$ 的不确定性, 即便某个 \hat{x}_i 的不确定性很大, 融合后的结果仍会更加准确. 因此只要其他机器人的子地图的不确定性较小, 融合后得到的全局地图的不确定性就不会受到较大影响. 极端情况下, 当某个机器人无法正常工作而瘫痪时, 其他机器人不会受到过多影响, 仍能按照主动的方式准确地建立自己的子地图, 并最终融合得到全局地图, 因此本文的方法具有一定的鲁棒性.

4 仿真 (Simulation)

假设工作环境为 $100\text{m} \times 80\text{m}$ 的矩形区域, 两个机器人的起始位姿分别为 $\mathbf{X}_{r_1}(0) = [-5, 0, \pi]^T$ 和 $\mathbf{X}_{r_2}(0) = [5, 0, 0]^T$, 控制周期 $T = 0.5 \text{ s}$, $\omega_1 = \omega_2 =$

$\omega_3 = 1/3$, 传感器探测距离范围为 $(0, 15] \text{ m}$, 夹角范围为 $[-\pi/2, \pi/2]$, $\mathbf{Q} = \text{diag}\{0.3^2, 0.3^2, 0.05^2, 0.3^2, \dots, 0.3^2\}$, $\mathbf{R} = \text{diag}\{0.2^2, 0.04^2\}$, $C_l = 0.3$, $d_l = 20 \text{ m}$. 简便起见, 将机器人的线速度和角速度分别离散化为 $u_1 = \{0.5, 1.0, 1.5, 2.0\}$ (单位: m/s) 和 $u_2 = \{-\pi/2, -3\pi/8, -\pi/4, -\pi/8, 0, \pi/8, \pi/4, 3\pi/8, \pi/2\}$ (单位: m/s).

在环境中随意放置 $N = 29$ 个路标, 采用本文提出的主动 SLAM 方法进行定位与地图创建, 数据关联采用最近邻方法, 算法终止条件为两个机器人观测到了所有路标. 图 1 给出了一组仿真结果, 其中 “*” 表示实际的路标位置, 而 “+” 表示估计的路标位置, 椭圆表示定位与地图创建的不确定性. 重复实验 30 次, 结果见表 1, 其中 $C(\mathbf{G}_{P_{\text{mm}}})/N$ 表示 N 个路标的平均不确定性, $C(\mathbf{G}_{P_{\text{mm}}}(k|k)) = \pi \sum_{i=1}^N \prod_{l=1}^2 \sqrt{\lambda_l(\mathbf{G}_{P_{ii}}(k|k))}$, 其中 $\lambda_l(\mathbf{G}_{P_{ii}}(k|k))$ 为 $\mathbf{G}_{P_{\text{mm}}}(k|k)$ 主对角线上的 2×2 子阵

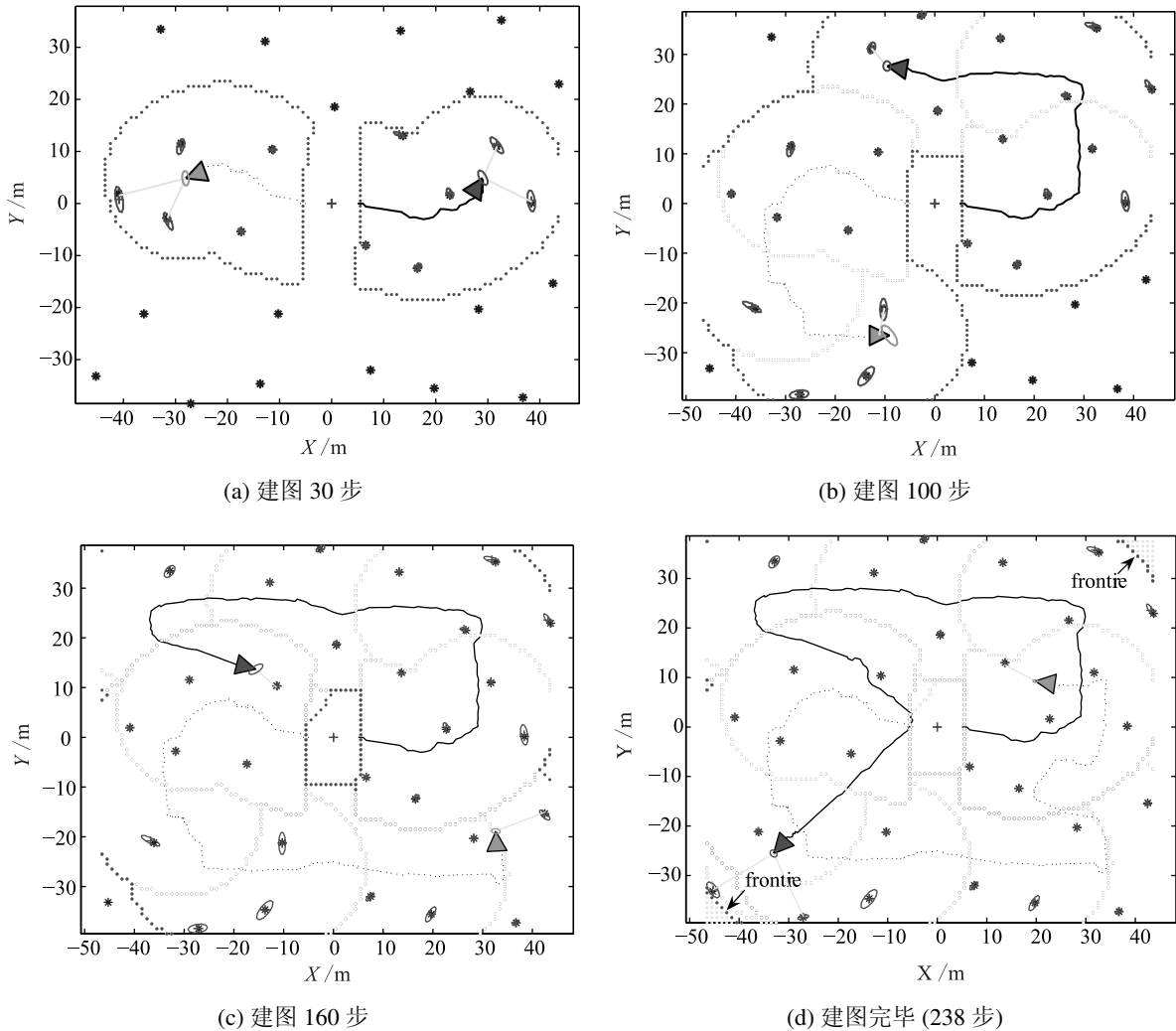


图 1 两机器人主动 SLAM 仿真结果

Fig.1 Simulation results of two-robot active SLAM

表 1 单机器人主动 SLAM 与多机器人主动 SLAM 的比较
Tab.1 Simulation comparison between single-robot and multi-robot active SLAM

	最小建图步数	最大建图步数	平均建图步数	平均建图误差	$C(\mathbf{G}_{P_{mm}})/N$
两机器人	208	371	296	0.282 m	0.327 m ²
单机器人	310	705	503	0.361 m	0.490 m ²

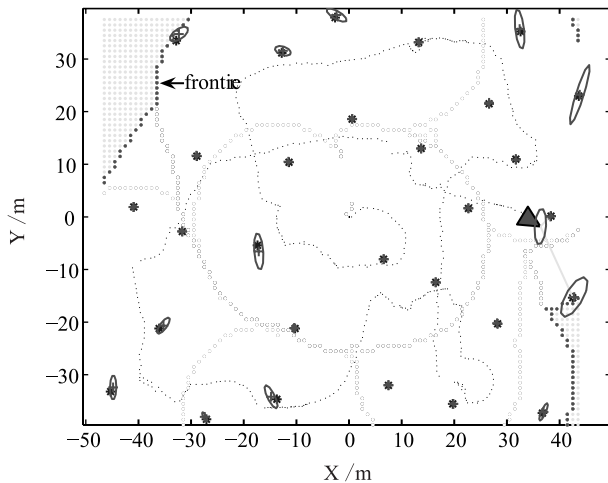


图 2 单机器人主动 SLAM 仿真结果 (431 步)

Fig.2 Simulation result of single-robot active SLAM (431 steps)

$\mathbf{G}_{P_i}(k|k)$ 的特征根. 在 30 次实验中机器人地图创建所用步数相差并不大, 平均为 296 步.

为进行实验比较, 在相同的环境中采用单机器人局部子地图方法, 机器人起始位姿为 $\mathbf{X}_r(0) = [0, 0, 0]^T$, 仿真结果见图 2. 重复实验 30 次, 结果见表 1. 由此可见, 两机器人的协作地图创建较单机器人的地图创建具有更高的效率, 并且由于单机器人建图运动路线较长, 所以误差累积更加明显, 导致其建图误差和不确定性较高.

5 结论 (Conclusion)

本文系统地研究了未知环境下的多机器人主动同时定位与地图创建问题, 通过引入局部子地图策略使每个机器人的建图相互独立, 从而将多机器人 SLAM 问题分解为多个单机器人 SLAM 子问题. 对于每个子问题, 将主动 SLAM 转化为最优控制问题, 通过将最优化目标函数的控制输入施加给机器人, 使机器人的运动兼顾定位与地图创建准确性、信息增益和多机器人之间的协调, 实现了准确、高效和鲁棒的定位与地图创建.

参考文献 (References)

[1] Fenwick J W, Newman P M, Leonard J J. Cooperative concurrent mapping and localization[A]. Proceedings of the IEEE International Conference on Robotics and Automation[C]. Piscataway, NJ, USA: IEEE, 2002. 1810~1817.

- [2] Thrun S, Burgard W, Fox D. A real-time algorithm for mobile robot mapping with applications to multi-robot and 3D mapping [A]. Proceedings of the IEEE International Conference on Robotics and Automation[C]. Piscataway, NJ, USA: IEEE, 2000. 321~328.
- [3] Thrun S, Liu Y F. Multi-robot SLAM with sparse extended information filters[A]. Springer Tracts in Advanced Robotics (vol.15)[M]. Berlin, Germany: Springer, 2003. 254~266.
- [4] Ko J, Stewart B, Fox D, *et al.* A practical, decision-theoretic approach to multi-robot mapping and exploration[A]. Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems[C]. Piscataway, NJ, USA: IEEE, 2003. 3232~3238.
- [5] Fox D. Distributed multi-robot exploration and mapping[A]. Proceedings of the 2nd Canadian Conference on Computer and Robot Vision[C]. Piscataway, NJ, USA: IEEE, 2005. 1325~1339.
- [6] Williams S B. Efficient Solutions to Autonomous Mapping and Navigation Problems[D]. Sydney, Australia: University of Sydney, 2001.
- [7] Williams S B, Dissanayake G, Durrant-Whyte H. Towards multi-vehicle simultaneous localisation and mapping[A]. Proceedings of the IEEE International Conference on Robotics and Automation[C]. Piscataway, NJ, USA: IEEE, 2002. 2743~2748.
- [8] Dissanayake M W M G, Newman P, Clark S, *et al.* A solution to the simultaneous localization and map building (SLAM) problem[J]. IEEE Transactions on Robotics and Automation, 2001, 17(3): 229~241.
- [9] Sim R, Elinas P, Griffin M, *et al.* Vision-based SLAM using the Rao-Blackwellised particle filter[DB/OL]. http://www.cs.ubc.ca/~simra/publications/rur05_sim.pdf, 2005.
- [10] Zhang S, Xie L H, Adams M. An efficient data association approach to simultaneous localization and map building[A]. Proceedings of the IEEE International Conference on Robotics and Automation[C]. Piscataway, NJ, USA: IEEE, 2004. 854~859.
- [11] Feder H J S, Leonard J J, Smith C M. Adaptive mobile robot navigation and mapping[J]. The International Journal of Robotics Research, 1999, 18(7): 650~668.
- [12] Yamauchi B. A frontier-based approach for autonomous exploration[A]. Proceedings of the IEEE International Symposium on Computational Intelligence in Robotics and Automation[C]. Los Alamitos, CA, USA: IEEE Computer Society, 1997. 146~151.

作者简介:

苑 晶 (1980 -), 男, 博士, 讲师. 研究领域: 移动机器人运动控制与导航, 多机器人协调.

黄亚楼 (1964 -), 男, 教授, 博士生导师. 研究领域: 智能机器人系统, 智能信息处理.