

文章编号 1004-924X(2015)增-0560-07

基于改进迭代无迹粒子滤波的同时定位与地图构建

罗 元,熊 艳*,张 毅

(重庆邮电大学 信息无障碍工程研发中心,重庆 400065)

摘要:针对移动机器人同时定位与地图构建(SLAM)中的粒子滤波不一致问题,提出了新的改进算法。利用迭代无迹粒子滤波精确设计了粒子滤波器的提议分布,以迭代更新方式将当前激光传感器信息和里程计信息融入重要性采样过程,实现了改进迭代无迹粒子滤波算法,降低了滤波器预测阶段机器人位姿的不确定性并有效地减少了所需粒子的数量。使用配有 URG 激光传感器的 Pioneer3-DX 在机器人操作系统平台上与 Fast SLAM 算法进行了比较实验。结果表明,创建相同一致性地图时,改进算法仅使用 10 个粒子构建地图,平均消耗时间为 325 s,除此之外还减少了所需粒子数量,提高了地图创建效率;同时机器人航向误差为 -1.486° ,降低了机器人位姿的不确定性。此外,对两种算法方差的比较可以看出改进算法的稳定度高于 FastSLAM 算法。

关键词:移动机器人;同时定位与地图构建;迭代无迹粒子滤波;重要性采样;机器人操作系统

中图分类号:TP242.6 **文献标识码:**A **doi:**10.3788/OPE.20152313.0560

Simultaneous localization and mapping implementation based on improved iterated unscented particle filter

LUO Yuan, XIONG Yan*, ZHANG Yi

(Engineering Research & Development Center of Information Accessibility,
Chongqing University of Posts and Telecommunications, Chongqing 400065, China)

* Corresponding author, E-mail: xiongy1211@163.com

Abstract: For particle filter inconsistency in mobile robot Simultaneous Localization and Mapping (SLAM), an improved unscented particle filter algorithm is proposed in this paper. To overcome this phenomenon mentioned above, the algorithm utilizes an iterated sigma point particle filter to generate more accurate proposal distribution, which fuses the robot's odometer information and laser information into sequential importance sampling routine through iterated update processing. The algorithm effectively improves the filter consistency and state estimation accuracy, and requires smaller number of particles. Based on the robot operating system, this algorithm is performed on a platform of Pioneer3-DX robot equipped with a URG laser range finder to compare with the traditional fast SLAM algorithm. Experimental results show that it creates a same consistency map and the improved algorithm with only 10 particles and consumes 325 s reduces the number of particles needed and improves the

收稿日期:2015-05-08;修订日期:2015-05-19.

基金项目:重庆市教委科学技术研究基金资助项目(No. KJ130512)

mapping efficiency. At the same time, the robot heading error is $-1.486 1^\circ$, showing a lower uncertainty of the robot pose. In addition, it indicates that the stability of the improved algorithm is higher than that of the FastSLAM algorithm by comparing their variances.

Key words: mobile robot; Simultaneous Localization and Mapping(SLAM); iterated unscented particle filter; importance sampling; laser range finder

1 引言

移动机器人同时定位与地图构建(Simultaneous Localization and Mapping, SLAM)是机器人在未知环境中依靠自身携带的传感器构建环境地图,并同时确定自身在该地图中位姿的过程。SLAM技术已被认为是解决移动机器人自主导航的关键^[1]。

基于 Rao-blackwellised 粒子滤波的 FastSLAM 算法是学者 Montemerlo 提出的^[2], 相比于传统的卡尔曼滤波 SLAM 方法, FastSLAM 具有计算代价小、适用于大尺度环境、对数据关联误差鲁棒等优点, 已经成为 SLAM 的流行方法框架^[3]。FastSLAM 算法根据 SLAM 问题的条件独立特性, 将高维的移动机器人轨迹和环境地图联合后验概率密度估计解耦为低维状态估计, 从而解决了状态空间 SLAM 算法的维数灾难问题, 提高了 SLAM 的求解效率。Kim 等^[4]提出了 UFastSLAM 算法, 其核心在于利用无迹变换来估计 SLAM 中的转移概率密度; 在 UFastSLAM 的基础上, Kim 等在文献^[5]中还对粒子滤波器提议分布设计问题进行了探讨, 更正了原始 UFastSLAM 中的过信估计问题, 给出了改进的 UFastSLAM 算法。

UFastSLAM 应用无迹粒子滤波技术(UPF)计算更加精确的方差和机器人的不确定性, UPF 技术考虑了加权粒子的线性回归。最初的 UPF 假定在每一个时间步骤中获取观测值, 然而, 在 SLAM 问题中, 缺乏一个观测值或多个观测值可能频繁发生, 因此本文提出以迭代更新方式融合当前激光传感器信息和里程计信息的方法来改进提议分布, 使得采样时考虑最新的观测信息, 从而提高了地图创建和位姿估算的精度, 同时还减少

了采样所需的粒子数目, 提高了计算性能。

2 FastSLAM 算法

FastSLAM 算法的核心思想在于利用机器人的历史控制信息 $\mathbf{u}^{t-1} = u_1, \dots, u_{t-1}$ 和机器人的历史环境观测信息 $\mathbf{z}^t = z_1, \dots, z_t$ 来估计机器人轨迹 $\mathbf{x}^t = x_1, \dots, x_t$ 和环境地图 $\Theta = \theta_1, \dots, \theta_M$ 的联合后验概率分布。联合后验概率分布可以表示为:

$$p(\mathbf{x}^t, \Theta | \mathbf{z}^t, \mathbf{u}^{t-1}) = p(\mathbf{x}^t | \mathbf{z}^t, \mathbf{u}^{t-1}) \cdot \prod_{m=1}^M p(\theta_m | \mathbf{x}^t, \mathbf{z}^t, \mathbf{u}^{t-1}). \quad (1)$$

上式表达的条件独立性使得笔者可以先对机器人进行估计, 再结合观测模型对地图进行更新。对于轨迹后验概率 $p(\mathbf{x}^t | \mathbf{z}^t, \mathbf{u}^{t-1})$ 采用粒子滤波器估计, 在机器人轨迹已知的前提下, 求解环境路标后验概率已有很好的解决办法^[6]。

3 改进的无迹粒子滤波 SLAM

在提议分布中融入当前环境观测信息, 使采样粒子集中分布于高观测似然区域, 是有效避免粒子集退化, 提高粒子滤波器性能的有效途径。本文采用迭代 Sigma 点无迹粒子滤波实现提议分布中的观测信息融合, 提出无迹粒子滤波算法, 其提议分布表达为高斯分布:

$$q(x_t | x_{t-1}, z_t, u_{t-1}) = N(x_t, P_t). \quad (2)$$

3.1 重要性采样

无迹粒子滤波的重要性采样过程包括两个阶段: 无迹粒子预测和 Sigma 点粒子更新。

3.1.1 无迹粒子预测

(1) 联合控制作用, 增广粒子状态

$$\mathbf{x}_{t-1}^{[m]} = \begin{bmatrix} \mathbf{x}_{t-1}^{[m]} \\ 0 \end{bmatrix}, P_{t-1}^{[m]} = \begin{bmatrix} \mathbf{P}_{t-1}^{[m]} & 0 \\ 0 & \mathbf{Q}_t \end{bmatrix}, \quad (3)$$

式中: $\mathbf{x}_{t-1}^{[m]}$ 为增广状态向量, $\mathbf{x}_{t-1}^{[m]}$ 为 $t-1$ 时刻第 m 个粒子代表的机器人的位姿参数, \mathbf{Q}_t 为控制噪

声协方差矩阵, $\mathbf{P}_{t-1}^{[m]}$ 为 $t-1$ 时刻第 m 个粒子的增广状态协方差矩阵。

(2) 计算对称 Sigma 支撑点集

利用一组表示高斯概率密度的对称 Sigma 支撑点集来传播不确定性, Sigma 支撑点集采样规则如下:

$$\begin{cases} \mathbf{x}_{t-1}^{[i][m]} = \\ \mathbf{x}_{t-1}^{[m]}, (i=0) \\ \mathbf{x}_{t-1}^{[m]} + (\sqrt{(L+\lambda)\mathbf{P}_{t-1}^{[m]}})_i, (i=1,2,\dots,L) \\ \mathbf{x}_{t-1}^{[m]} - (\sqrt{(L+\lambda)\mathbf{P}_{t-1}^{[m]}})_{i-L}, (i=L+1,\dots,2L) \end{cases}, \quad (4)$$

其中: i 表示矩阵 $\mathbf{P}_{t-1}^{[m]}$ 的第 i 列, $\lambda = \alpha^2(L + \kappa) - L$, $\alpha(0 < \alpha < 1)$ 应该为一个小值, 以避免非线性增强时的取样非局部效应 ($\alpha = 0.002$ 较为合适)。 κ 为尺度参数, 决定了 Sigma 支撑点在样本均值附近的分布情况。

(3) 利用非线性函数传播 Sigma 支撑点集

每一个 Sigma 点 $\mathbf{x}_{t-1}^{[i][m]}$ 包含了状态、控制和观测分量, 表示为:

$$\mathbf{x}_{t-1}^{[i][m]} = [\mathbf{x}_t^{[i][m]}, \mathbf{u}_t^{[i][m]}, \mathbf{y}_t^{[i][m]}]^T. \quad (5)$$

运动模型 $f(\cdot)$ 为一个非线性函数, 该模型中每个 Sigma 点的当前控制 $\mathbf{u}_t^{[m]}$ 具有增广控制噪声分量 $\mathbf{u}_t^{[i][m]}$, 一系列的 Sigma 点 $\mathbf{x}_{t-1}^{[i][m]}$ 由运动模型来传播:

$$\begin{aligned} \bar{\mathbf{x}}_t^{[i][m]} &= f(\mathbf{u}_t^{[m]} + \mathbf{u}_t^{[i][m]}, \mathbf{x}_{t-1}^{[i][m]}) = \\ &[\bar{\mathbf{x}}_{x,t}^{[i][m]}, \bar{\mathbf{x}}_{y,t}^{[i][m]}, \bar{\mathbf{x}}_{\beta,t}^{[i][m]}]^T, \end{aligned} \quad (6)$$

其中: $\bar{\mathbf{x}}_t^{[i][m]}$ 为经过非线性函数传播后机器人状态的 Sigma 点集。

(4) 粒子状态预测和状态协方差预测

利用经传播后的 Sigma 点集 $\bar{\mathbf{x}}_t^{[i][m]}$ 预测粒子状态和协方差矩阵, 由 $\bar{\mathbf{x}}_t^{[i][m]}$ 的线性加权回归计算得到:

$$\mathbf{x}_{t-1}^{[m]} = \sum_{i=0}^{2L} \omega_g^{[i]} \bar{\mathbf{x}}_t^{[i][m]}, \quad (7)$$

$$\mathbf{P}_{t-1}^{[m]} = \sum_{i=0}^{2L} \omega_c^{[i]} (\bar{\mathbf{x}}_t^{[i][m]} - \mathbf{x}_{t-1}^{[m]}) (\bar{\mathbf{x}}_t^{[i][m]} - \mathbf{x}_{t-1}^{[m]})^T, \quad (8)$$

式中: 权重系数 $\omega_g^{[i]}$ 在计算方差时使用, 权重系数 $\omega_c^{[i]}$ 在恢复高斯分布的协方差时使用。权重的计算公式为:

$$\begin{cases} \omega_g^{[0]} = \frac{\lambda}{(L+\lambda)} (i=0) \\ \omega_c^{[0]} = \frac{\lambda}{(L+\lambda)} + (1 - \alpha^2 + \beta) (i=0) \\ \omega_g^{[i]} = \omega_c^{[i]} = \frac{\lambda}{2(L+\lambda)} (i=1,2,\dots,2L) \end{cases}, \quad (9)$$

其中, 参数 β 用来吸收后验分布高阶分量的知识。对一个高斯分布, β 最佳值为 2。

3.1.2 迭代 Sigma 点机器人状态更新

迭代更新能够有效降低滤波对非线性系统线性化处理造成的高阶截断误差。基于无迹变换的统计学线性化代替迭代扩展卡尔曼滤波 (Iterated Extended Kalman Filter, IEKF) 中的泰勒一阶线性化, 第 m 个粒子的迭代 Sigma 点更新步骤为:

(1) 联合观测, 增广预测粒子状态:

$$\mathbf{x}_{t-1}^{[m]} = \begin{bmatrix} \mathbf{x}_{t-1}^{[m]} \\ 0 \end{bmatrix}, \mathbf{P}_{t-1}^{[m]} = \begin{bmatrix} \mathbf{P}_{t-1}^{[m]} & 0 \\ 0 & \mathbf{R}_t \end{bmatrix}, \quad (10)$$

式中: $\mathbf{x}_{t-1}^{[m]}$ 为增广状态向量, $\mathbf{x}_{t-1}^{[m]}$ 为粒子迭代后的状态, \mathbf{R}_t 为测量噪声协方差矩阵。

(2) 计算对称 Sigma 支撑点集:

$$\mathbf{x}_t^{[i][m]} = [\mathbf{x}_t^{[i][m]}, \mathbf{x}_t^{[i][m]} \pm (\sqrt{(L+\lambda)\mathbf{P}_t^{[m]}})]. \quad (11)$$

其中各个参数定义如式 (4)。

(3) 利用 Sigma 支撑点集进行观测预测:

$$\bar{\mathbf{Z}}_t^{[i][m]} = h(\mathbf{x}_t^{[i][m]}, \mathbf{x}_t^{[i][m]}) + \mathbf{x}_t^{[i][m]}, \quad (12)$$

$$\hat{\mathbf{z}}_t^{[m]} = \sum_{i=0}^{2L} \omega_g^{[i]} \bar{\mathbf{Z}}_t^{[i][m]}, \quad (13)$$

$\bar{\mathbf{Z}}_t^{[i][m]}$ 为经观测函数 $h(\cdot)$ 传播后的 Sigma 点, 具有非线性函数的特点, 其中增加了观测噪声分量 $\mathbf{x}_t^{[i][m]}$ 。 $\hat{\mathbf{z}}_t^{[m]}$ 为第 m 个粒子对环境特征的观测预测。

(4) 状态迭代更新

利用 Sigma 支撑点集计算协方差矩阵:

$$\bar{\mathbf{S}}_t^{[m]} = \sum_{i=0}^{2L} \omega_c^{[i]} (\bar{\mathbf{Z}}_t^{[i][m]} - \hat{\mathbf{z}}_t^{[m]}) (\bar{\mathbf{Z}}_t^{[i][m]} - \hat{\mathbf{z}}_t^{[m]})^T, \quad (14)$$

$$\bar{\Sigma}_t^{[m]} = \sum_{i=0}^{2L} \omega_c^{[i]} (\bar{\chi}_t^{[i][m]} - \mu_{n_i,t}^{[m]}) (\bar{\mathbf{Z}}_t^{[i][m]} - \hat{\mathbf{z}}_t^{[m]})^T. \quad (15)$$

式中: $\bar{\mathbf{S}}_t^{[m]}$ 为新的协方差矩阵, $\bar{\Sigma}_t^{[m]}$ 为互协方差矩阵。

第 m 个粒子在观测更新中的滤波增益为:

$$\mathbf{K}_i^{[m]} = \bar{\mathbf{S}}_i^{[m]} (\bar{\mathbf{S}}_i^{[m]})^{-1}. \quad (16)$$

机器人在时刻 t 的状态更新为:

$$\mathbf{x}_i^{[m]} = \mathbf{x}_{i|t-1}^{[m]} + \mathbf{K}_i^{[m]} (\mathbf{z}_i - \hat{\mathbf{n}}_i^{[m]}). \quad (17)$$

其状态协方差矩阵更新为:

$$\mathbf{P}_i^{[m]} = \mathbf{P}_{i|t-1}^{[m]} - \mathbf{K}_i^{[m]} \bar{\mathbf{S}}_i^{[m]} (\mathbf{K}_i^{[m]})^T. \quad (18)$$

滤波更新完毕后,执行重要性采样:

$$\mathbf{x}_i^{[m]} \sim N(\mathbf{x}_i^{[m]}, \mathbf{P}_i^{[m]}). \quad (19)$$

如果机器人同时观测到多个环境特征,则对粒子集中的每一个粒子均执行多次迭代 Sigma 点状态更新后在执行(19)式的重要性采样。

3.2 重要性权值的计算和重采样策略

无迹粒子滤波 SLAM 的重要性权值在计算时需要考虑传感器的最新观测,通过下面等式给出:

$$\omega_i^{[m]} = |2\pi\mathbf{L}_i^{[m]}|^{-(1/2)} e^{\{-1/2\}(\mathbf{z}_i - \hat{\mathbf{z}}_i^{[m]})^T (\mathbf{L}_i^{[m]})^{-1} (\mathbf{z}_i - \hat{\mathbf{z}}_i^{[m]})}. \quad (20)$$

其中:

$$\mathbf{L}_i^{[m]} = \left(\sum_t \bar{\mathbf{S}}_t^{[m]} \right)^T (\mathbf{P}_i^{[m]})^{-1} \sum_t \bar{\mathbf{S}}_t^{[m]} + \bar{\mathbf{S}}_i^{[m]}. \quad (21)$$

重采样是影响粒子滤波器性能的一个主要因素,在重采样过程中低权值粒子被高权值粒子代替,重采样技术使用有效粒子数作为标准,粒子的有效粒子数计算公式为:

$$N_{\text{eff}} = \frac{1}{\sum_{m=1}^M (\hat{\omega}^{[m]})^2}, \quad (22)$$

其中: M 为粒子总数, $\hat{\omega}^{[m]}$ 为第 m 个粒子的归一化权值。如果重要性权值的协方差增加,则 N_{eff} 减小。将粒子总数的 50% 作为阈值,当 N_{eff} 小于该阈值时,进行重采样,这将解决协方差增加造成的粒子退化问题。

3.3 改进算法的流程

提出的改进无迹粒子滤波 SLAM 算法的具体过程为:

Step1 滤波初始化

首先,确定机器人状态的初始分布 $N(x_0, P_0)$,控制噪声 \mathbf{Q} 与观测噪声 \mathbf{R} 以及 Sigma 点迭代更新次数;然后,从机器人状态初始分布采集

N 个粒子,即 $\{x_0^{[m]}\} \sim N(x_0, P_0)$ 。

Step2 滤波执行

(1)粒子集预测:利用机器人运动模型 f 、控制输入 u 和噪声协方差 \mathbf{Q} ,根据 3.1.1 节的无迹粒子预测过程预测每个粒子状态 $\mathbf{x}_{i-1}^{[m]}$ 与协方差 $\mathbf{P}_{i-1}^{[m]}$;

(2)判断:如果没有观测到环境信息,转步骤(5),否则继续向下执行;

(3)迭代更新:首先,对每一个粒子,根据其状态预测 $\mathbf{x}_{i-1}^{[m]}$ 、协方差预测 $\mathbf{P}_{i-1}^{[m]}$ 与当前观测 \mathbf{z}_i 、观测噪声 \mathbf{R} ,执行 3.1.2 节中的迭代 Sigma 点粒子状态更新,得到该粒子的提议分布;然后,按式(19)从提议分布中采集新一代粒子,并计算粒子权重;

(4)重采样:粒子集权重归一化处理后进行重采样;

(5)根据机器人位姿 $\mathbf{x}_i^{[m]}$ 和观测信息 \mathbf{z}_i 计算 $\mathbf{m}^{[i]}$,并更新地图。

4 ROS 系统上基于改进无迹粒子滤波算法的 SLAM 实现

4.1 ROS 上的 SLAM 系统构建

Robot Operating System(ROS)是 2010 年 Willow Garage 公司发布的一种用于机器人的后操作系统^[7]。它提供了一个标准的操作系统环境,包括硬件抽象、底层设备控制、通用功能的执行、进程间消息传递和功能包发行包管理等,也提供了一些工具和库用于获取、建立、编写和执行多机构融合的程序。ROS 是一种分布式的进程框架^[8],它可执行若干种类型的通信,包括基于服务的同步 RPC 的通信、基于主题的异步数据流及参数服务器上的数据储存。

计算图是 ROS 处理数据的一种点对点的网络形式^[9],其中基本概念包括节点(Node)、节点管理器(Master)、参数服务器(Parameter Server)、消息(Message)、主题(Topic)、服务(Service)和消息记录包(Bag)。Node 是执行运算任务的进程,Message 是由类型域构成的数据结构,可包含任何嵌套的结构和阵列,Node 之间通过传递消息进行通讯。Topic 用于识别消息内容,是一个

带有发布和订阅消息功能的传输系统,节点可通过它进行消息传送。Master 提供名称注册和查找功能。Service 能够允许其它节点直接与某个节点进行交互。Bag 是一种用于保存和回放消息数据的文件格式。

本文在 ROS 平台上完成实验,主要 Nodes 如图 1 所示。其中, / base_link 为公共的 Topic,所有的 Message 都在这上面进行交互;Odometer 节点、Laser Scan 节点用来记录里程计信息、激光信息并发布到 Topic;rosaria 节点提取的 RosAria (由发布的信息在 Topic 上转换得来)信息用于控制 Pioneer3 机器人;Master 用于对各节点进行注册和查找;Map Building 提取里程计和激光信息并通过提出的改进算法进行地图构建。

另外,ROS 中提供了一个 3D 可视化工具 Rviz,可以在显示面板区域选择需要显示的选项。它还包括一个 3D 可视化区域,用于显示所加载的所有项,本文使用它来实时在线显示所构建的地图。

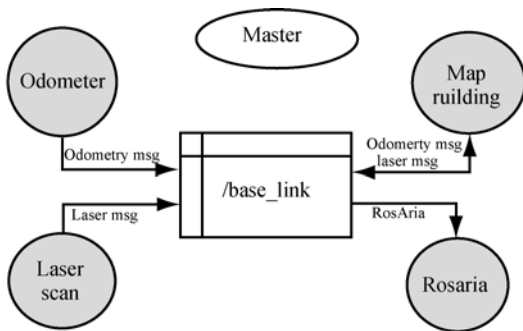


图 1 ROS 上节点运行图

Fig. 1 Node operation diagram on ROS

4.2 ROS 上改进无迹粒子滤波算法的 SLAM 实现

本实验在 Pioneer3-DX 移动机器人上进行。该机器人配备一个 URG-HUKUYO 激光测距仪和一个用串口线连接的 PC,PC 为 Linux(Ubuntu12.04)上运行的 ROS 系统,通过串口线来控制机器人移动,并采集里程计和激光数据进行在线地图构建和显示。为完成本实验,依次需运行的节点有:

```
$ roscore
$ rosrn ROSARIA rosaria
$ rosrn SLAM_mapping transform
```

```
$ rosrn SLAM_mapping mapping
$ rosrn Rviz rviz
```

其中 roscore 用于打开整个系统,rosaria 节点用于 PC 与机器人交互,transform 节点把激光坐标与里程计坐标转换到统一坐标系下,mapping 节点为本文的重点,它将根据激光和里程计提供测量信息,利用本文提出的改进算法完成 SLAM,最后通过 Rviz 实时显示,如图 2 所示。其中白色区域 W 为实验环境,黑色区域 B 表示障碍物,D 为激光扫描点集,P 为机器人建图过程中的移动路径。

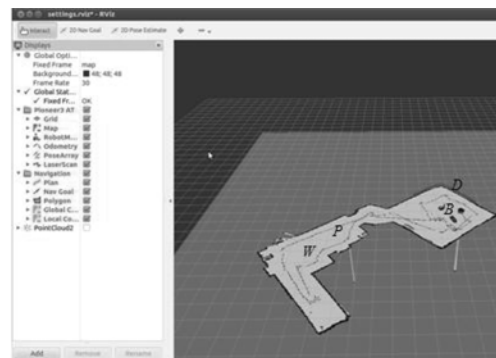


图 2 Rviz 实时建图显示

Fig. 2 Real-time map of Rviz

本实验分别使用传统 FastSLAM 算法和改进算法进行路径规划。图 3 为传统 FastSLAM 算法采用 40 个粒子构建的环境地图,由于传统算法是从运动模型中进行采样的而没有考虑新的观测值,从而导致算法误差较大,构建的地图会出现不一致现象(如图中 a,b,c,d,e 处)。图 4 为改进算法使用 10 个粒子构建的高精度地图。

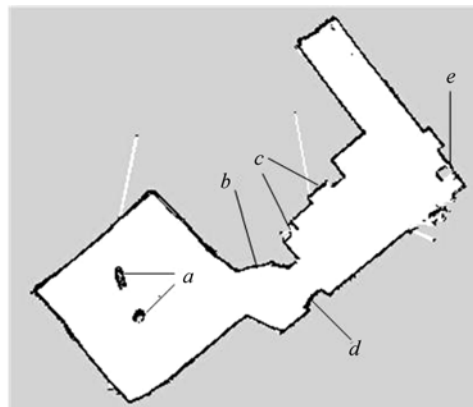


图 3 传统 FastSLAM 算法构建的地图

Fig. 3 Map built by traditional FastSLAM algorithm

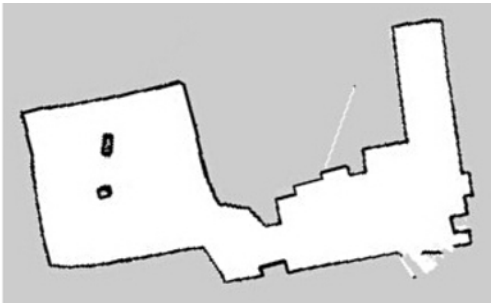


图 4 改进算法构建的地图

Fig. 4 Map built by improved algorithm

4.3 实验结果分析

表 1 比较了传统算法与改进算法创建同一一致性地图所需要的粒子数和时间,可以看出改进算法需要的粒子数远小于传统算法,创建地图的时间也较短。这是因为本文是采用迭代 Sigma 点无迹粒子滤波实现提议分布中的观测信息融合,使采样粒子集中地分布于高观测似然区域,从而提高了滤波器的一致性和状态估计精度,相比于 FastSLAM 方法,需要较少的粒子数。另外,因为算法的处理时间随着粒子的减少而减少,所以改进后的算法不仅提高了算法效率,还缩短了处理时间。

表 1 创建一致性地图参数表

Tab. 1 Parameters list when building a consistency map

Algorithm	Number of particles/个	Elapsed time/s
FastSLAM	40	530
本文算法	10	325

参考文献:

- [1] 陈晨,程荫杭. 基于迭代无迹卡尔曼滤波的 SLAM 算法仿真研究[J]. 系统仿真学报, 2012, 24(8): 1643-1650.
CHEN CH, CHENG Y H. Simulation research of SLAM algorithm based on iterated unscented Kalman filter [J]. *Journal of System Simulation*, 2012, 24(8):1643-1650. (in Chinese)
- [2] MONTEMERLO M, THRUN S, KOLLER D, et al. FastSLAM: a factored solution to the simultaneous localization and mapping problem[C]. *Proceedings of the 18th National Conference on Arti-*

表 2 两种算法估计精度比较

Tab. 2 Estimation accuracy comparison of two kinds of algorithms

Algorithm	MSE/m		Heading errors/(°)
	mean	var	
FastSLAM	0.783	0.500	-2.692 7
本文算法	0.083	0.076	-1.486 1

表 2 是在两种算法均使用 10 个粒子时的均值、方差和航向误差的结果比较,可以看出本文提出的改进算法的估计精度高于传统的 FastSLAM 算法。

5 结 论

本文提出了一种改进迭代无迹粒子滤波 SLAM 算法,利用迭代无迹粒子滤波来精确设计粒子滤波器的提议分布,以迭代更新方式将观测模型融入重要性采样过程,使得采样粒子更接近于真实分布,降低了滤波器预测阶段机器人位姿的不确定性,并有效减少了所需的粒子的数量。实验结果证明,创建相同的一致性地图,改进算法仅使用了 10 个粒子,平均消耗时间为 325 s,其航向误差为 $-1.486 1^\circ$,而 FastSLAM 算法需要 40 个粒子,平均消耗时间为 530 s,其航向误差为 $-2.697 2^\circ$ 。实验结果体现了改进算法的高效性和创建地图的精确性,通过对两种算法方差的比较可以看出改进算法的稳定度高于 FastSLAM 算法。

ficial Intelligence, Cambridge, MA: MIT Press, 2002: 593-598.

- [3] 宋宇,李庆玲,康轶非,等. 平方根容积 Rao-Blackwellised 粒子滤波 SLAM 算法[J]. 自动化学报, 2014, 40(2):357-367.
SONG Y, LI Q L, KANG Y F, et al. SLAM with square-root cubature rao-Blackwellised particle filter [J]. *Acta Automatica Sinica*, 2014, 40(2): 357-367. (in Chinese)
- [4] KIM C, SAKTHIVEL R, Chung W K. Unscented FastSLAM: a robust and efficient solution to the SLAM problem[J]. *IEEE Transactions on Robotics*, 2008, 24(4): 808-820.
- [5] KIM C, KIM H K, CHUNG W K. Exactly Rao-

- Blackwellized unscented particle filters for SLAM [C]. *Proceedings of the 2011 IEEE International Conference on Robotics and Automation, Shanghai, China*; IEEE, 2011, 3589-3594.
- [6] MORAVEC H P. Sensor fusion in certainty grids for mobile robots [J]. *AI Magazine*, 1988, 9(2): 61-75.
- [7] 张建伟,张立新,胡颖,等. 开源机器人操作系统-ROS[M]. 北京:科学出版社,2012.
ZHANG J W, ZHANG L X, HU Y, et al. *The Open Source Robot Operating System-ROS* [M]. Beijing: Science Press,2012. (in Chinese)
- [8] AARON M E F á ndez. *Learning ROS for Robotics Programming*[M]. Beijing: China Machine Press, 2014.
- [9] 罗元,傅有力,程铁凤. 基于改进 Rao-Blackwellized 粒子滤波器的同时定位与地图构建[J]. *控制理论与应用*, 2015, 32(2):267-272.
LUO Y, FU Y L, CHENG T F. Simultaneous localization and mapping implementation based on the improved Rao-Blackwellized particle filter[J]. *Control Theory & Applications*, 2015, 32(2): 267-272. (in Chinese)

作者简介:



罗 元(1972—),女,湖北宜昌人,博士,教授,1996年于重庆邮电大学信号与信息处理专业获得硕士学位,2003年于重庆大学仪器科学与技术精密仪器及机械系获得博士学位,主要研究领域为机器视觉、人机交互、基于图像视频处理的测试。E-mail: luoyuan@cqupt.edu.cn



熊 艳(1989—),女,贵州贵阳人,硕士研究生,2013年于重庆邮电大学获得学士学位,主要从事移动机器人自主导航方面的研究。E-mail: xiongy1211@163.com



张 毅(1965—),男,重庆人,博士,教授,1997年于合肥工业大学获得精密仪器及机械硕士学位,2002年于华中科技大学获得机械制造及其自动化博士学位,现为重庆邮电大学国家信息无障碍工程研发中心主任,中欧智能系统及机器人联合实验室副主任,重庆邮电大学智能系统及机器人研究所所长,主要从事智能机器人、人机交互、机器视觉和智能控制等方面的研究。E-mail: zhangyi@cqupt.edu.cn