

基于多假设跟踪的移动机器人自适应蒙特卡罗定位研究

张恒¹ 樊晓平¹ 瞿志华^{1,2}

摘要 针对移动机器人蒙特卡罗定位 (Monte Carlo localization, MCL) 算法在含有对称和自相似结构的环境中容易失败的问题, 提出了一种基于多假设跟踪的自适应蒙特卡罗定位改进算法. 该算法根据粒子间空间相似性采用核密度树聚类算法对粒子群进行聚类, 每簇粒子代表一个位姿假设并用一个独立的 MCL 算法进行跟踪, 总体上形成了一组非等权的粒子滤波器, 很好地克服了普通粒子滤波器由于粒子贫乏而引起的过度收敛问题. 同时运用该核密度树实现了自适应采样, 提高了算法的性能. 针对机器人“绑架”问题对该算法作了进一步的改进. 实验结果证明了该算法的有效性.

关键词 移动机器人, 蒙特卡罗定位, 多假设跟踪, 核密度树
中图分类号 TP24

Mobile Robot Adaptive Monte Carlo Localization Based on Multiple Hypothesis Tracking

ZHANG Heng¹ FAN Xiao-Ping¹ QU Zhi-Hua^{1,2}

Abstract This paper presents an improved algorithm that extends Monte Carlo localization (MCL) to solve the problem of localization failure in symmetric and/or self-similar environments. The algorithm clusters the particles adaptively according to their spatial similarity by using a kernel density (kd)-tree-based cluster algorithm. Each cluster of particles denotes a pose hypothesis and is traced by an individual MCL process so as to form a group of unequally weighted particle filters in general, thus overcoming the over-convergence problem due to lack of the particle sets. The kd-trees are also used for adaptive sampling to improve the algorithm performance. Further improvement to the algorithm makes it possible to solve the kidnapped robot problem as well, and the experimental results show that it has higher efficiency than the standard MCL algorithm.

Key words Mobile robot, Monte Carlo localization, multiple hypothesis tracking, kernel density tree

1 引言

移动机器人的定位问题是指移动机器人通过传感器信息估算自身位置和姿态的过程. 更具体地说是利用先验环境地图信息、机器人位姿的当前估计以及传感器的观测值等输入信息, 经过一定的处理和变换, 产生更加准确的对机器人当前位姿的估计. 机器人定位问题分为两类: 位姿跟踪问题和全局定位问题. 前者指机器人的初始位姿已知, 根据传感器的观测信息来补偿码盘的误差; 如果机器人的初始位姿未知, 或者在跟踪过程中出现了“绑架”现象 (即在未通知机器人的情况下改变机器人的位姿), 那么就成了全局定位问题. 近年来, 越来越多的研究者把概率理论应用到移动机器人定位中^[1]. 对于解

决全局定位问题, 最具有代表性的是蒙特卡罗定位 (Monte Carlo localization, MCL) 算法^[2] 和马尔可夫定位 (Markov localization) 算法^[3]. 其中 MCL 算法具有实现简单、定位精度高、所需计算资源少等优点而受到研究者的青睐, 并对其提出了许多改进的方法, 代表性的有 Mixture-MCL^[4]、自适应采样 MCL^[5,6]、遗传粒子滤波^[7] 等.

粒子贫化现象是 MCL 算法的一个主要障碍, 在要得到同样精度的条件下, 所需的粒子数目随搜索空间大小成倍增长, 而为了满足定位的实时性要求, 一般只能选用一定数量的样本 (粒子) 来代表机器人的位姿. 这样势必导致机器人定位过快地收敛于高概率的位姿, 严重时会导致定位失败, 特别是在含有对称结构和自相似结构的环境中. 为了克服这种困难, 我们提出一种基于自适应动态粒子聚类的新的 MCL 算法, 其基本思想是: 适时地对等权粒子根据其在空间上的相似性进行聚类, 每簇粒子代表一个机器人的位姿假设, 在其后的一段时间里每簇粒子对应于一个独立的 MCL 算法, 直到满足一定的判定条件再从全局抽取等权粒子进行聚类, 如此反复. 文献 [8] 首先提出了类似的思想, 但本文算法与之相比有如下特点: 1) 不改变粒子滤波的建议分布, 算

收稿日期 2006-5-26 收修稿日期 2006-12-15
Received May 26, 2006; in revised form December 15, 2006
国家自然科学基金 (69975003) 资助
Supported by National Natural Science Foundation of China (69975003)

1. 中南大学信息科学与工程学院 长沙 410075 2. 美国中佛罗里达大学电子与计算机工程系 奥兰多 FL 32816-2450
1. School of Information Science and Engineering, Central South University, Changsha 410075 2. Department of Electrics and Computer Engineering, University of Central Florida, Orlando, FL 32816-2450, U. S. A.
DOI: 10.1360/aas-007-0941

法更加简单、清晰; 2) 粒子类别数在初始给定后可以不时地通过从全局抽取等权粒子后聚类获得, 即粒子动态聚类; 3) 在本算法框架下可以容易地将其他对粒子滤波器的改进算法应用进来, 从而进一步提高算法的效率和性能, 如在本文的算法实现过程中就应用了文献 [5, 6] 中提出的自适应采样方法; 4) 更有效地解决了机器人“绑架”问题。

2 蒙特卡罗定位

移动机器人定位问题可以描述为: 在机器人和其所处环境组成的动态系统中, 根据机器人初始状态概率分布 $p(\xi_0)$ 、截止到 t 时刻对机器人的控制序列 $\mathbf{u}^t = \{\mathbf{u}_t | t = 1, 2, \dots, t\}$ 和机器人对环境的观测序列 $\mathbf{z}^t = \{\mathbf{z}_t | t = 0, 1, \dots, t\}$, 估计机器人 t 时刻的状态 ξ_t 。其中 $\xi_t = (x_t, y_t, \theta_t)^T$ 表示 t 时刻机器人相对于地图的位姿, x_t, y_t 为笛卡尔坐标, θ_t 为机器人朝向角。从统计学的观点看, 对 ξ_t 的估计是一个贝叶斯滤波问题, 可通过估计后验分布 $p(\xi_t | \mathbf{z}^t, \mathbf{u}^t)$ 来实现。贝叶斯滤波器假设系统是一个马尔可夫过程, $p(\xi_t | \mathbf{z}^t, \mathbf{u}^t)$ 可以通过以下两步递推获得:

1) 预测。根据机器人运动模型预测机器人在下一时刻的状态, 即计算先验概率密度 $p(\xi_t | \mathbf{z}^{t-1}, \mathbf{u}^t)$:

$$\begin{aligned} p(\xi_t | \mathbf{z}^{t-1}, \mathbf{u}^t) &= \int p(\xi_t | \xi_{t-1}, \mathbf{z}^{t-1}, \mathbf{u}^t) p(\xi_{t-1} | \mathbf{z}^{t-1}, \mathbf{u}^t) d\xi_{t-1} \\ &= \int p(\xi_t | \xi_{t-1}, \mathbf{u}_t) p(\xi_{t-1} | \mathbf{z}^{t-1}, \mathbf{u}^{t-1}) d\xi_{t-1} \quad (1) \end{aligned}$$

其中 $p(\xi_t | \xi_{t-1}, \mathbf{u}_t)$ 为机器人的运动模型, 可通过机器人的运动学方程获得。

2) 更新。根据机载传感器 (如激光、超声等距离传感器) 的观测模型, 利用新的观测信息更新系统的状态。即通过如下贝叶斯公式计算后验概率密度 $p(\xi_t | \mathbf{z}^t, \mathbf{u}^t)$:

$$\begin{aligned} p(\xi_t | \mathbf{z}^t, \mathbf{u}^t) &= \frac{p(\mathbf{z}_t | \xi_t, \mathbf{u}^t) p(\xi_t | \mathbf{z}^{t-1}, \mathbf{u}^t)}{p(\mathbf{z}_t | \mathbf{z}^{t-1}, \mathbf{u}^t)} \\ &= \frac{p(\mathbf{z}_t | \xi_t) p(\xi_t | \mathbf{z}^{t-1}, \mathbf{u}^t)}{p(\mathbf{z}_t | \mathbf{z}^{t-1}, \mathbf{u}^t)} \quad (2) \end{aligned}$$

其中 $p(\mathbf{z}_t | \xi_t)$ 为机器人传感器的观测模型, 可通过机器人传感器和地图模型获得, $p(\mathbf{z}_t | \mathbf{z}^{t-1}, \mathbf{u}^t)$ 为归一化常量。

对于机器人全局定位问题, 由于模型的非线性和状态分布的复杂性, 直接通过解析方式求解这样的递推问题是困难的。而近年来发展起来的粒子滤波器可以很好地解决这样的问题^[9], 即蒙特卡罗定位 (Monte Carlo localization, MCL)。

MCL 的基本思想就是用 N 个带有权值的样本 $S_t = \{(\xi_t^{(j)}, w_t^{(j)}) | j = 1, 2, \dots, N\}$ 来近似地表示后验概率密度 $p(\xi_t | \mathbf{z}^t, \mathbf{u}^t)$ 。其中 $\xi_t^{(j)}$ 是机器人在 t 时刻的一个可能状态, $w_t^{(j)}$ 是一个非负实数称为重要性因子 (粒子的权值), 表示 t 时刻机器人状态为 $\xi_t^{(j)}$ 的概率, $\sum_{j=1}^N w_t^{(j)} = 1$ 。这样机器人的后验分布可以近似地表示为

$$p(\xi_t | \mathbf{z}^t, \mathbf{u}^t) \approx \sum_{j=1}^N w_t^{(j)} \delta(\xi_t - \xi_t^{(j)}) \quad (3)$$

对于全局定位, 机器人初始位姿是未知的, 其状态的先验概率一般用整个机器人可行空间上的均匀分布表示。基本的 MCL 算法见算法 1。

算法 1. 基本 MCL 算法。

Input: $t-1$ 时刻的带权粒子集 S_{t-1} , 控制输入 \mathbf{u}_t , 机器人对环境的观测 \mathbf{z}_t 。

Output: t 时刻的带权粒子集 S_t 。

1. S_t 置空;
2. for $i = 1$ to N do
3. 从 S_{t-1} 根据重要性因子随机抽取一样本 $\xi_{t-1}^{(n)}$;
4. 采用 \mathbf{u}_t , 状态 $\xi_{t-1}^{(n)}$, 根据运动模型抽取粒子 $\xi_t^{(i)} \sim p(\xi_t | \xi_{t-1}^{(n)}, \mathbf{u}_t)$;
5. 根据观测模型计算粒子的重要性因子 $w_t^{(i)} = p(\mathbf{z}_t | \xi_t^{(i)})$, 将带权粒子 $(\xi_t^{(i)}, w_t^{(i)})$ 加入到 S_t ;
6. end;
7. 对 S_t 中的重要性因子 $w_t^{(i)}$ 作归一化处理;
8. 返回 S_t 。

3 粒子聚类蒙特卡罗定位

在机器人全局定位开始一段时间后, 代表机器人位姿的粒子常聚于若干簇, 这表示机器人的位姿有多个假设。有时在一段时间内仅根据机器人对环境的观测信息不能消除这种多假设局面, 特别是在对称或含有自相似结构的环境中。运用基本的 MCL 算法常常会由于粒子贫乏使得这些假设中的某一个成为最终选择, 而这个选择常是错误的。出现该问题的原因在于概率越大的假设抽取的粒子越多, 最终使得其他概率小的假设跟踪丢失。为解决这种问题, 我们的基本思想是在不能完全否定假设 (即消除该簇粒子) 的情况下, 每个假设对应于至少一定数目的粒子, 使得滤波器不会太快地否定小概率的假设。这样做的结果是重采样后的粒子非等权。

设在 $t-1$ 时刻可用 K 簇粒子 $\{(C_{k,t-1}, B_{k,t-1}) | k = 1, 2, \dots, K\}$ 表示机器人位姿的后验分布。其中, $C_{k,t-1} = \{(\xi_{k,t-1}^{(j)}, w_{k,t-1}^{(j)}) | j = 1, 2, \dots, M_k\}$ 为 $t-1$ 时刻第 k 簇粒子的集合; $B_{k,t-1}$ 为 $C_{k,t-1}$ 包含真实机器人位姿的概率, 记为 $p(k_{t-1} | \mathbf{z}^{t-1}, \mathbf{u}^{t-1})$, 其中 $\sum_{k=1}^K B_{k,t-1} = 1$, $\sum_{j=1}^{M_k} w_{k,t-1}^{(j)} = 1$ 。现在的问题是如何根据控制输入

\mathbf{u}_t , 机器人对环境的观测 \mathbf{z}_t , 得到 t 时刻的 K 簇粒子 $\{(C_{k,t}, B_{k,t})|k = 1, 2, \dots, K\}$. $C_{k,t}$ 的求解过程同算法 1 所示的基本 MCL 算法一样, 这里重点讨论 $B_{k,t}$ 的求解过程.

类似于式 (1) 和 (2), 运用贝叶斯公式和全概率公式, 有

$$\begin{aligned} B_{k,t} &= p(k_t|\mathbf{z}^t, \mathbf{u}^t) = \\ & \frac{p(\mathbf{z}_t|k_t, \mathbf{u}^t)p(k_t|\mathbf{z}^{t-1}, \mathbf{u}^t)}{p(\mathbf{z}_t|\mathbf{z}^{t-1}, \mathbf{u}^t)} = \\ & \frac{p(\mathbf{z}_t|k_t)p(k_t|\mathbf{z}^{t-1}, \mathbf{u}^t)}{p(\mathbf{z}_t|\mathbf{z}^{t-1}, \mathbf{u}^t)} \propto \\ & p(\mathbf{z}_t|k_t)p(k_t|\mathbf{z}^{t-1}, \mathbf{u}^t) \propto \\ & p(\mathbf{z}_t|k_t) \sum_{k_{t-1}=1}^K p(k_t|\mathbf{u}_t, k_{t-1})p(k_{t-1}|\mathbf{z}^{t-1}, \mathbf{u}^{t-1}) \propto \\ & p(\mathbf{z}_t|k_t) \sum_{k_{t-1}=1}^K p(k_t|\mathbf{u}_t, k_{t-1})B_{k,t-1} \end{aligned} \quad (4)$$

由于对应于各簇粒子的滤波算法是独立运行的, 粒子不会改变其所在的簇类别, 于是

$$p(k_t|\mathbf{u}_t, k_{t-1}) = \begin{cases} 0, & \text{若 } k_t \neq k_{t-1} \\ 1, & \text{若 } k_t = k_{t-1} \end{cases} \quad (5)$$

从而,

$$B_{k,t} \propto p(\mathbf{z}_t|k_t)B_{k,t-1} \quad (6)$$

$p(\mathbf{z}_t|k_t)$ 可通过第 k 簇粒子中所有粒子的加权观测概率之和获得, 即

$$p(\mathbf{z}_t|k_t) \propto \sum_{(\xi_{k,t}^{(i)}, w_{k,t}^{(i)}) \in C_{k,t}} w_{k,t}^{(i)} p(\mathbf{z}_t|\xi_{k,t}^{(i)}) \quad (7)$$

将式 (7) 代入式 (6) 中, 得到

$$B_{k,t} \propto B_{k,t-1} \sum_{(\xi_{k,t}^{(i)}, w_{k,t}^{(i)}) \in C_{k,t}} w_{k,t}^{(i)} p(\mathbf{z}_t|\xi_{k,t}^{(i)}) \quad (8)$$

具体的粒子聚类 MCL 算法见算法 2.

算法 2. 粒子聚类 MCL 算法.

Input: $t-1$ 时刻的粒子簇 $\{(C_{k,t-1}, B_{k,t-1})|k = 1, 2, \dots, K\}$, 控制输入 \mathbf{u}_t , 机器人对环境的观测 \mathbf{z}_t .

Output: t 时刻的粒子簇 $\{(C_{k,t}, B_{k,t})|k = 1, 2, \dots, K\}$.

1. for $k = 1$ to K do
2. $C_{k,t}$ 置空;
3. for $i = 1$ to M_k do
4. 从 $C_{k,t-1}$ 根据重要性因子随机抽取一样本 $\xi_{k,t-1}^{(n)}$;
5. 采用 \mathbf{u}_t , 状态 $\xi_{k,t-1}^{(n)}$, 根据运动模型抽取粒子 $\xi_{k,t}^{(i)} \sim p(\xi_t|\xi_{k,t-1}^{(n)}, \mathbf{u}_t)$;

6. 根据观测模型计算粒子的权值 $w_{k,t}^{(i)} = p(\mathbf{z}_t|\xi_{k,t}^{(i)})$, 将带权粒子 $(\xi_{k,t}^{(i)}, w_{k,t}^{(i)})$ 加入到 $C_{k,t}$;
7. end;
8. $C_{k,t}$ 中粒子重要性因子归一化;
9. end;
10. 根据式 (8) 计算 $\{B_{k,t}|k = 1, 2, \dots, K\}$;
11. 返回 $\{(C_{k,t}, B_{k,t})|k = 1, 2, \dots, K\}$.

可见, 当 $K = 1$ 时粒子聚类 MCL 算法等价于基本算法.

4 自适应粒子聚类蒙特卡罗定位

第 3 节理论上解决了基于粒子聚类的 MCL 问题, 但在多假设跟踪的过程中, 还有一些重要的问题需要解决: 1) 何时对粒子重聚类? 2) 如何对粒子重聚类? 3) 每簇粒子在重采样的过程中采多少样本才合适? 对于问题 1, 可简单地度量粒子簇的包罗体各边是否超过预定值, 如超过则对粒子再聚类. 对于问题 2, 首先将滤波器中所有粒子的权值在整体上归一化: $w_{k,t}^{(j)} = B_{k,t}w_{k,t}^{(j)}$, 即组合成为单个粒子集, 然后在此基础上抽取等权粒子构建核密度树 (Kernel density tree, kd-tree)^[4], 最后基于该 kd-tree 对其进行分类形成新的粒子簇, 具体分类过程在此省略. 对于问题 3, 我们引用文献 [5, 6] 中基于 Kullback-Leibler 距离的抽样方法. 具体实现时同样采用前面的 kd-tree, 也就是说该 kd-tree 是在采样的过程中同时构建. 这样得到的粒子簇检测与重聚类算法见算法 3.

算法 3. 粒子簇检测与重聚类算法.

Input: t 时刻旧的粒子簇 $\{(C_{k,t}, B_{k,t})|k = 1, 2, \dots, K\}$, 参数: $\alpha_x, \alpha_y, \alpha_\theta, N$.

Output: t 时刻新的粒子簇 $\{(C'_{k,t}, B'_{k,t})|k = 1, 2, \dots, L\}$.

/* $(x_{k,t}^{(j)}, y_{k,t}^{(j)}, \theta_{k,t}^{(j)}) = \xi_{k,t}^{(j)}$ */

/* $\{(\xi_{k,t}^{(j)}, w_{k,t}^{(j)})|j = 1, 2, \dots, M_k\} = C_{k,t}$ */

1. key = 0;

2. for $k = 1$ to K do

3. $dx = \max(\{x_{k,t}^{(j)}|j = 1, 2, \dots, M_k\}) - \min(\{x_{k,t}^{(j)}|j = 1, 2, \dots, M_k\})$;

4. $dy = \max(\{y_{k,t}^{(j)}|j = 1, 2, \dots, M_k\}) - \min(\{y_{k,t}^{(j)}|j = 1, 2, \dots, M_k\})$;

5. $dc = \max(\{\theta_{k,t}^{(j)}|j = 1, 2, \dots, M_k\}) - \min(\{\theta_{k,t}^{(j)}|j = 1, 2, \dots, M_k\})$;

6. if $dx > \alpha_x$ or $dy > \alpha_y$ or $dc > \alpha_\theta$ then

7. key = 1;

8. break;

9. else

10. continue;

11. end

12. end

13. if key == 1 then

14. $n = 0$; 清空 kd-tree: T ;
15. while $n < N$ do
16. 基于 $\bigcup_{k=1}^K \bigcup_{j=1}^{M_k} (\xi_{k,t}^{(j)}, B_{k,t} w_{k,t}^{(j)})$
抽取样本 $\xi_t^{(n)}$;
17. 将 $\xi_t^{(n)}$ 插入 T ;
18. 将 $n = n + 1$;
19. if $(n > LIMIT(T))$ then break;
/* 函数 $LIMIT(T)$ 是关于 T 的叶子节点的函数, 返回基于 KL 距离的抽样数目 */
20. 归一化得等权粒子集
 $\{(\xi_t^{(j)}, 1/n) | j = 1, 2, \dots, n\}$;
21. 基于 T 得到 $\{(C'_{k,t}, B'_{k,t}) | k = 1, 2, \dots, L\}$;
22. end
23. else
24. $\{(C'_{k,t}, B'_{k,t}) | k = 1, 2, \dots, L\} =$
 $\{(C_{k,t}, B_{k,t}) | k = 1, 2, \dots, K\}$;
25. end
26. 返回 $\{(C'_{k,t}, B'_{k,t}) | k = 1, 2, \dots, L\}$.

对于不需要重聚类而运行的各独立粒子滤波器同样采用上面的基于 kd-tree 的抽样方法. 在定位过程当某簇粒子的整体概率 $B_{k,t}$ 小于给定值 α 时, 我们可认为该假设不成立, 从而可将它从粒子簇群中删除. 在环境较为复杂、传感器精度不高的情况下, 可能会出现一些新的假设 (即簇可能会分离), 这时可通过上面的粒子簇检测与重聚类算法重获聚类粒子, 再启动粒子聚类蒙特卡罗定位算法. 这样形成了一个完整的自适应粒子聚类蒙特卡罗定位算法, 其流程如图 1 所示.

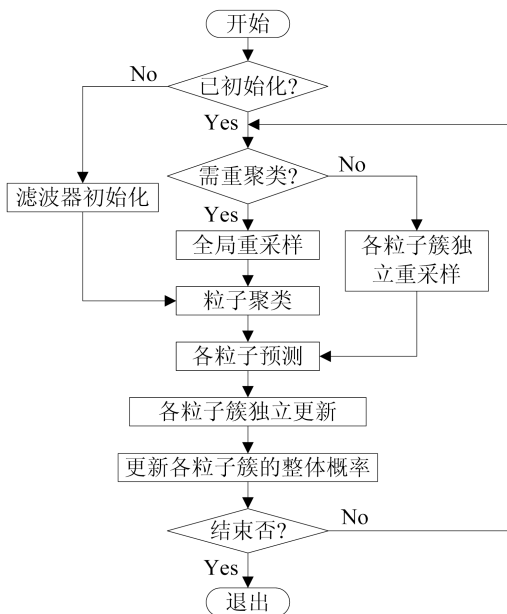


图 1 自适应粒子聚类蒙特卡罗定位类算法流程图

Fig. 1 Flow chart for adaptive particle clustered Monte Carlo localization algorithm

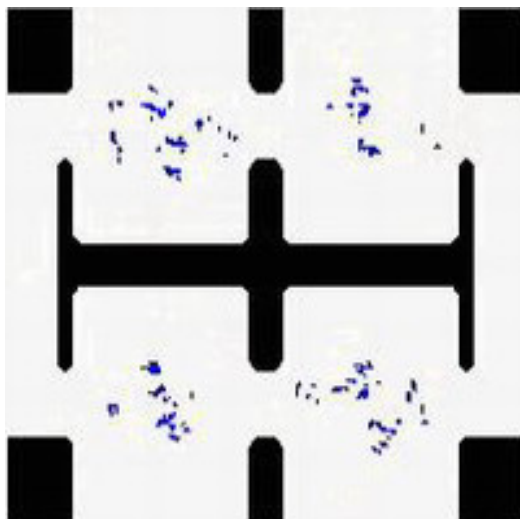
5 算法的进一步改进

本文提出的自适应聚类粒子蒙特卡罗定位算法能很好地解决机器人在含有对称或自相似结构的环境中全局定位的问题. 但当出现机器人“绑架”现象时, 或由于地图不精确, 机器人起始位置周围的环境与地图严重不匹配时, 将会使得没有一簇粒子可以包含机器人的真实位姿. 针对这种情况, 可充分利用本文算法中各粒子簇独立演化的特点, 再加入一些随机粒子簇而不影响原粒子簇的演化. 具体做法是: 每隔一段时间在原有粒子簇之外启动一个类似于初始定位时的 MCL 系统, 不同的是在该系统中采用的粒子数目很少, 且将传感器误差作一定程度的放大, 当该系统的粒子聚于一簇时 (用粒子包罗体的大小衡量) 检测是否与原粒子簇中某簇相吻合, 如果不与任何簇相吻合, 则认为此时机器人已出现“绑架”问题, 将该簇粒子加入到聚类粒子 MCL 的粒子簇中独立成为一簇, 并给其赋一较小的整体概率值. 另外, 由于各粒子簇对应的 MCL 系统相对独立, 可以将其它一些已有的为提高粒子滤波性能的方法应用其中, 如基于当前统计模型的算法^[10], 遗传粒子滤波器^[7], 多次观测合成的实时粒子滤波器^[11]等.

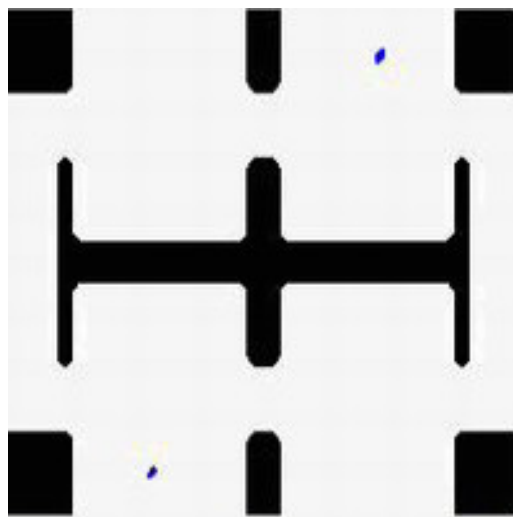
6 仿真实验及结果

基于 PlayerStage (参见以下网址: <http://playerstage.sourceforge.net>) 移动机器人平台, 我们做了对比仿真实验: PlayerStage 中已有的自适应 MCL (amcl 驱动器) 和本文提出的算法. 机器人模型为 Pioneer 机器人上装有激光测距仪. 环境地图为图 2 (见下页) 所示的对称环境. 机器人初始位姿位于左下角的方块中, 并在该区域作随机避障运动. 用 PlayerStage 中的 amcl 做了大量实验, 每次实验在第 10 步以前滤波器就会收敛于左下区域 (正确定位) 或右上区域 (定位失败), 而不能保持对这两个假设的完全跟踪. 同样用本文的算法也做了大量实验, 而每次实验都能一直保持对两个假设的正确跟踪. 图 2 为应用本文算法的一次实验结果.

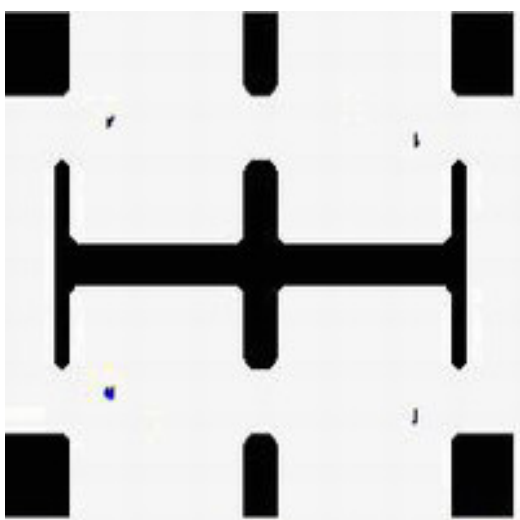
对于机器人“绑架”问题, 我们也做了大量的仿真实验. 用 PlayerStage 中的 amcl 由于没有考虑机器人“绑架”问题, 每次实验都会失败, 而本文提出的方法在绝大多数情况下都可以校正定位的结果. 不能成功的情况主要是由于环境太大, 辅助粒子滤波器的粒子密度相对过小, 或者是传感器误差放大程度过大. 简单的解决途径是生成更多的粒子并减小传感器误差放大程度, 但这样的后果是计算量太大 (几乎是加了一个初始定位过程), 严重影响定位的实时性.



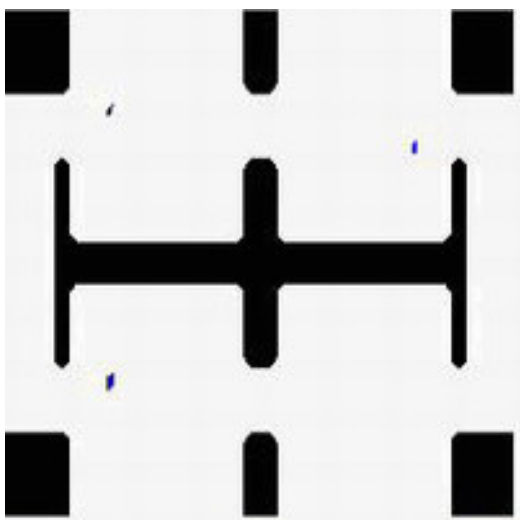
(a) Step 3



(d) Step 40



(b) Step 10



(c) Step 13

图 2 自适应粒子聚类蒙特卡罗定位实验结果

Fig. 2 Results of the simulation of adaptive particle clustered Monte Carlo localization algorithm

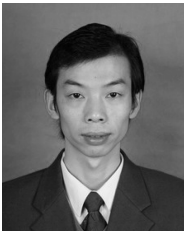
7 结论

本文针对移动机器人 MCL 算法在含有对称和自相似结构的环境中容易失败的问题, 构造了一类基于动态粒子聚类的两层结构、非等权粒子滤波器, 从而形成了一种基于多假设跟踪的 MCL 改进算法. 该算法很好地克服了普通粒子滤波器由于粒子贫乏而引起的过度收敛问题, 算法进一步改进后同时可以解决机器人“绑架”问题. 此外, 由于该算法框架的特点, 其他对粒子滤波器的改进算法也能较方便地应用到其中, 以进一步提高算法的性能. 实验结果证明了该算法的有效性.

References

- 1 Li Mao-Hai, Hong Bing-Rong. Progress of probabilistic localization methods in mobile robots. *Robot*, 2005, **27**(4): 380~384
(厉茂海, 洪炳榕. 移动机器人的概率定位方法研究进展. *机器人*, 2005, **27**(4): 380~384)
- 2 Fox D, Burgard W, Dellaert F, Thrun S. Monte Carlo localization: efficient position estimation for mobile robots. In: *Proceedings of National Conference on Artificial Intelligence*. Orlando, Florida, USA: AAAI/MIT Press, 1999. 343~349
- 3 Fox D. Markov Localization: A Probabilistic Framework for Mobile Robot Localization and Navigation [Ph. D. dissertation], University of Bonn, 1998
- 4 Thrun S, Fox D, Burgard W, Dellaert F. Robust Monte Carlo localization for mobile robots. *Artificial Intelligence*, 2001, **128**(1-2): 99~141
- 5 Fox D. Adapting the sample size in particle filters through KLD-sampling. *The International Journal of Robotic Research*, 2003, **22**(12): 985~1003

- 6 Jiang Zheng-Wei, Gu Yuan-Tao. Novel adaptive particle filters in robot localization. *Acta Automatica Sinica*, 2005, **31**(6): 833~838
(蒋正伟, 谷源涛. 机器人定位中的自适应粒子滤波算法. 自动化学报, 2005, **31**(6): 833~838)
- 7 Luo Rong-Hua, Hong Bing-Rong, Piao Song-Hao, Dai Hu-Ming. Coevolution-based adaptive particle filters for global localization. *Chinese Journal of Electronics*, 2005, **14**(3): 458~462
- 8 Milstein A, Sanchez J N, Williamson E. Robust global localization using clustered particle filtering. In: Proceedings of National Conference on Artificial Intelligence. Edmonton, Alberta, Canada: AAAI Press, 2002. 581~586
- 9 Hu Shi-Qiang, Jing Zhong-Liang. Overview of particle filter algorithm. *Control and Decision*, 2005, **20**(4): 361~365
(胡士强, 敬忠良. 粒子滤波算法综述. 控制与决策, 2005, **20**(4): 361~365)
- 10 Deng Xiao-Long, Xie Jian-Ying, Wang Lin-Su. Improved particle filter based on current statistical model. *Control and Decision*, 2005, **20**(5): 567~570
(邓小龙, 谢剑英, 王林苏. 基于当前统计模型的改进粒子滤波算法. 控制与决策, 2005, **20**(5): 567~570)
- 11 Kwok C, Fox D, Meila M. Real-time particle filters. *Proceedings of the IEEE*, 2004, **92**(3): 469~484



张 恒 中南大学信息科学与工程学院博士研究生. 2001 年获中南大学信息学院计算机系学士学位, 主要研究方向为智能移动机器人系统, 机器视觉. 本文通信作者.

E-mail: hbzhangheng@sohu.com
(**ZHANG Heng** Ph. D. candidate at School of Information Science and Engineering, Central South University. He received his bachelor

degree from Central South University in 2001. His research interest covers intelligent mobile robot system and machine vision. Corresponding author of this paper.)



樊晓平 中南大学教授, 1998 年在华南理工大学获博士学位. 主要研究方向为机器人控制, 智能控制, 智能交通系统.
E-mail: xpfan@mail.csu.edu.cn

(**FAN Xiao-Ping** Professor at Central South University. He received his Ph. D. degree from South China University of Technology in 1998. His research

interest covers robot control, intelligent control, and intelligent transport systems.)



瞿志华 美国中佛罗里达大学教授和中南大学长江学者讲座教授, 1990 年在美国乔治亚理工学院获博士学位. 主要研究方向为非线性系统与控制, 机器人, 学习控制, 电力系统.

E-mail: qu@pegasus.cc.ucf.edu

(**QU Zhi-Hua** Professor at Central Florida University, USA and Central South University, P. R. China. He received his Ph. D. degree from Georgia Institute of Technology in 1990. His research interest covers nonlinear systems and controls, robotics, learning control, and power systems.)