

基于 SSPF 算法的移动机器人全局定位研究^{*}

罗 颖, 谭冠政

(中南大学 信息科学与工程学院, 湖南 长沙 410083)

摘 要: 传统粒子滤波器(PF)算法在移动机器人全局定位时会出现粒子退化和粒子耗尽问题, 导致定位失败。提出了一种改进的 PF 算法即智能采样 PF(SSPF)算法, 该方法采用 Unscented 卡尔曼滤波器(UKF)算法和基于相关熵的自适应重采样算法相结合的方法, 较好解决了传统 PF 算法所引起的粒子退化问题。由于重采样的一种极端情况会带来粒子耗尽问题, 采用 MH(metropolis hastings)算法, 较好地解决了粒子耗尽问题。与以往方法相比, 该方法可以实现准确性较高的移动机器人全局定位。

关键词: 粒子滤波器; 移动机器人; 全局定位; 智能采样 PF 算法; Unscented 卡尔曼滤波器; MH 算法

中图分类号: TP242.6 **文献标识码:** A **文章编号:** 1000-9787(2008)05-0031-04

Research on global localization of mobile robot based on SSPF algorithm^{*}

LUO Ying, TAN Guan-zheng

(School of Information Science and Engineering Central South University, Changsha 410083, China)

Abstract: Traditional particle filter(PF) algorithm can lead to particle degeneracy and particle depletion problems in global localization of mobile robot to cause localizing defeat. A improved PF named smart sampling particle filter(SSPF) algorithm is proposed by combining Unscented Kalman filter(UKF) algorithm and relative entropy for adaptive PF algorithm, which solves the particle degeneracy of traditional PF algorithm. Since one of the extreme conditions of resampling can cause particle depletion, this problem is solved by using metropolis hastings(MH) approach. Comparing with previous methods, the algorithm can implement global localization of mobile robot with high precise.

Key words: partide filter(PF); mobile robot; global localization; smart sampling particle fliter(SSPF) algorithm; Unscented Kalman filter(UKF); metropolis hastings(MH) algorithm

0 引 言

移动机器人定位是自治式智能机器人领域中的一个基本但又非常重要的问题, 其定位过程包括位置跟踪和全局定位^[1]。位置跟踪是指初始位置已知, 通过纠正里程计累积误差估计位姿; 而全局定位要求机器人能够在初始位置不确定的情况下, 通过具有噪声的传感器感知信息估计位姿, 因此, 全局定位更具挑战性。目前, 解决全局定位有很多方法。Julier S J 等人^[2]提出 UKF 算法, 但 UKF 算法仍存在很多约束。近年发展起来的基于 PF 算法的定位方法^[3]是一种全新的定位方法。但传统 PF 算法也存在许多局限性, 第一, 如果在正确状态附近没有足够多的粒子, 将会导致滤波器发散; 第二, 粒子经常快速收敛, 导致 PF 算法仅能获得次优结果。如何提高滤波器的准确性、有效性并防止粒子退化(退化问题是指只有少数粒子的权值不为

0, 其余粒子的权值近似为 0)和粒子耗尽(耗尽问题是指粒子集退化成仅含有同一个粒子的多个拷贝)一直是研究者关心的问题。为了使粒子更好地表示后验概率密度, Thrun S 等人^[4]提出的混合蒙特卡洛法和 Fox D^[5]提出的基于自适应采样的方法, 虽然提高了滤波器的有效性, 但也加重了计算负担。

相对于基本的 PF 算法, 本文提出 SSPF 算法在以下 3 个方面做了改进: 1)在采样阶段, 采用了 UKF 算法来产生重要性提议分布, 从而使采样点同实际分布更为接近, 较好地解决了粒子退化问题; 2)在重采样阶段, 采用了基于相关熵的自适应重采样方法, 该方法可有效减少重采样的执行步骤, 使 PF 算法更具智能性, 并在一定程度上可缓解重采样步骤损失粒子多样性的问题; 3)由于重采样有可能会产生粒子耗尽这一极端情况, 采用了 MH 算法较好地解

决了粒子耗尽问题,使得定位更加准确。

1 传统 PF 算法

为了更好地理解和分析改进算法,先简要介绍一下传统 PF 算法。

假设动态系统状态空间模型如下所示

$$\begin{cases} \mathbf{x}(k) = \mathbf{f}[\mathbf{x}(k-1), \mathbf{u}(k-1), \epsilon(k-1)] & (\text{运动模型}) \\ \mathbf{z}(k) = \mathbf{h}[\mathbf{x}(k), \eta(k)] & (\text{观测模型}) \end{cases}, \quad (1)$$

式中 $\mathbf{x}(k)$ 为系统状态向量; $\mathbf{z}(k)$ 为系统观测向量; $\mathbf{u}(k)$ 为输入向量; $\epsilon(k)$ 为系统噪声; $\eta(k)$ 为观测噪声; $\epsilon(k)$, $\eta(k)$ 分别为独立同分布噪声序列; $\mathbf{f}(\cdot)$, $\mathbf{h}(\cdot)$ 既可为线性函数也可为非线性函数; 系统输入 $\mathbf{u}(k)$ 已知。

传统 PF 算法实现步骤如下:

1) 初始化: 当时刻 $k=0$ 时, 从先验概率密度函数 $p(\mathbf{x}_0)$ 中提取 N 个样本点 \mathbf{x}_0^i , 并设置初始权重 $w_0^i = 1/N$, $i=1, 2, \dots, N$;

2) 重要性采样、计算权重: 当时刻 $k=1$ 时, 从转换先验密度函数 $p(\mathbf{x}_k | \mathbf{x}_{k-1}^i)$ 中提取 N 个样本点 \mathbf{x}_k^i , 通过公式 $w_k^i = w_{k-1}^i p(\mathbf{z}_k | \mathbf{x}_k^i)$ 计算权重 (公式中的 $p(\mathbf{z}_k | \mathbf{x}_k^i)$ 为似然密度函数), 用公式 $\tilde{w}_k^i = w_k^i / \sum_{i=1}^N w_k^i$ 来对权值进行归一化;

3) 重采样: 设置门限样本点数为 N_{th} (通常情况下 N_{th} 与粒子数 N 相等), 有效样本点数由公式 $N_{eff} = 1 / \sum_{i=1}^N (\tilde{w}_k^i)^2$ 确定, 当 $N_{eff} < N_{th}$ 时, 抛弃低权值的采样点, 并在保持样本点总数为 N 的前提下, 用残余采样法^[2]从权值较高的采样点中衍生出多个子采样点。各样本点权重在重采样后均为 $w_k^i = 1/N$, $i=1, 2, \dots, N$ 。执行完后, 令 $k=k+1$, 转到(2), 开始下一轮循环。直到 $k=T$ 时, 循环结束。

2 改进算法描述

2.1 UKF 设计提议分布

选择一个好的提议分布在 PF 设计中非常重要, Doucet A 等人^[6]已经证明 $p(\mathbf{x}_k | \mathbf{x}_{k-1}, \mathbf{z}_k)$ 是最优提议分布。传统 PF 算法是把转换先验密度函数 $p(\mathbf{x}_k | \mathbf{x}_{k-1})$ 作为最优提议分布 $p(\mathbf{x}_k | \mathbf{x}_{k-1}, \mathbf{z}_k)$ 的近似形式并从中采样。然而, 以转换先验密度函数做提议分布的最大缺陷在于 $p(\mathbf{x}_k | \mathbf{x}_{k-1})$ 并没有考虑最近的传感器感知信息 \mathbf{z}_k 。如果观测概率呈尖峰形状 (传感器信息较为可靠时发生) 或者预测状态在后验分布的尾部 (状态预测不准确时发生), 则会引起严重的退化问题。为了有效地利用观测信息, 本文提出使用 UKF 算法来产生提议分布, UKF 算法采用确定性采样, 而非 PF 算法的随机采样, 从而避免了 PF 算法的粒子点退化问题。

在 UKF 算法中, 由于有噪声项, 需要对系统进行扩维处理, 针对式(1)和式(2)定义的系统, 令 $\mathbf{x}^a = [\mathbf{x}^T \ \epsilon^T \ \eta^T]^T$

相应的 sigma 矩阵为 $\Sigma^a = [(\Sigma^x)^T (\Sigma^\epsilon)^T (\Sigma^\eta)^T]^T$ 。具体的算法流程如下:

1) 计算 sigma 点

$$\mathbf{x}_{k-1}^a = [\mathbf{x}_{k-1}^x \ \mathbf{x}_{k-1}^\epsilon \ \mathbf{x}_{k-1}^\eta \pm \sqrt{(n_a + \lambda) \mathbf{P}_{k-1}^a}] \quad (2)$$

2) 时间更新

$$\begin{cases} \mathbf{x}_{k|k-1}^x = \mathbf{f}(\mathbf{x}_{k-1}^x, \mathbf{u}_{k-1}, \eta_{k-1}^\epsilon) \\ \bar{\mathbf{x}}_{k|k-1} = \sum_{i=0}^{2n_a} \mathbf{w}_i^{(m)} \mathbf{x}_{k|k-1}^i \\ \mathbf{P}_{k|k-1} = \sum_{i=0}^{2n_a} \mathbf{w}_i^{(c)} [\mathbf{x}_{k|k-1}^i - \bar{\mathbf{x}}_{k|k-1}] [\mathbf{x}_{k|k-1}^i - \bar{\mathbf{x}}_{k|k-1}]^T \cdot (3) \\ \mathbf{z}_{k|k-1} = \mathbf{h}(\mathbf{x}_{k|k-1}^x, \eta_{k-1}^\eta) \\ \bar{\mathbf{z}}_{k|k-1} = \sum_{i=0}^{2n_a} \mathbf{w}_i^{(m)} \mathbf{z}_{i,k|k-1} \end{cases}$$

3) 观测更新

$$\begin{cases} \mathbf{P}_{\bar{\mathbf{z}}_{k-1} \bar{\mathbf{z}}_{k-1}} = \sum_{i=0}^{2n_a} \mathbf{w}_i^{(c)} [\mathbf{z}_{i,k|k-1} - \bar{\mathbf{z}}_{k|k-1}] [\mathbf{z}_{i,k|k-1} - \bar{\mathbf{z}}_{k|k-1}]^T \\ \mathbf{P}_{\mathbf{x}_{k-1} \bar{\mathbf{z}}_{k-1}} = \sum_{i=0}^{2n_a} \mathbf{w}_i^{(c)} [\mathbf{x}_{i,k|k-1} - \bar{\mathbf{x}}_{k|k-1}] [\mathbf{z}_{i,k|k-1} - \bar{\mathbf{z}}_{k|k-1}]^T \\ \mathbf{K}_k = \mathbf{P}_{\mathbf{x}_{k-1} \bar{\mathbf{z}}_{k-1}} \mathbf{P}_{\bar{\mathbf{z}}_{k-1} \bar{\mathbf{z}}_{k-1}}^{-1} \\ \bar{\mathbf{x}}_k = \bar{\mathbf{x}}_{k|k-1} + \mathbf{K}_k (\mathbf{z}_k - \bar{\mathbf{z}}_{k|k-1}) \\ \mathbf{P}_k = \mathbf{P}_{k|k-1} - \mathbf{K}_k \mathbf{P}_{\bar{\mathbf{z}}_{k-1} \bar{\mathbf{z}}_{k-1}} \mathbf{K}_k^T \end{cases} \quad (4)$$

上述公式中的 \mathbf{x}^a , $\mathbf{w}_i^{(m)}$, \mathbf{P} 等参数的定义参见文献[2]。

UKF 把当前观测信息吸收到状态估计中, 使粒子向高概率区域移动, 从而减少了粒子退化。

2.2 基于相关熵的自适应重采样算法

重采样方法是解决权值退化问题的主要方法。本文提出一种基于相关熵的自适应重采样算法, 减少了重采样方法的执行步骤, 并且, 缓解了重采样步骤损失粒子多样性的问题。其基本思想是通过相关熵衡量前后 2 个时刻规范化权值的差异, 如果 2 个时刻规范化权值差异较大, 则认为权值退化比较严重, 需执行重采样步骤, 否则, 继续执行重要性采样方法。算法如下:

相关熵主要用来描述 2 个概率分布 $p(x)$ 和 $q(x)$ 之间的距离, 距离越小, 说明 2 个概率分布差异越小。相关熵的定义如式(5)所示, 并且, 当且仅当 $p(x) = q(x)$ 时, $K(p, q)$ 为 0, 否则, 大于 0 即

$$K(p, q) = \sum_x p(x) \log \frac{p(x)}{q(x)} \quad (5)$$

假定在 k 时刻第 i 个粒子的权值为 w_k^i 。时刻 k 与时刻 $k-1$ 粒子规范化权值的差异使用相关熵度量, 并考虑在 PF 执行过程中可能会出现 $\log 0$ 的问题, 得到公式(6)

$$K_m(w_k, w_{k-1}) = \sum_{i=1}^{N_s} w_k^i (w_k^i - w_{k-1}^i)^2, \tag{6}$$

式中 K_m 表示时刻 k 与时刻 $k-1$ 规一化权值的差异, 如果 2 个时刻规一化权值的差异在可接受范围内, 则继续执行重要性采样步骤, 如果差异较大, 则认为规一化权值衰退比较快, 为了有效遏制其衰退, 则需执行重采样步骤。在 PF 执行重要性采样步骤之后, 计算 $K_m(w_k, w_{k-1})$, 如果 K_m 大于阈值 σ , 则执行重采样步骤。

2.3 MH 算法

重采样在极端情况下会导致粒子耗尽问题, 为了克服这个问题, 可以增加粒子集的数量, 但这样会使计算量过大。本文采用在重采样步骤之后引入 MH 算法。MH 算法通过建立一个平稳分布为 $\pi(x)$ 的马尔科夫链来得到 $\pi(x)$ 的样本和在此基础上的统计推断, 其基本过程是产生具有马尔科夫依赖性质的样本, 在给定当前状态 x 的情况下根据提议分布 $q(x^*|x)$ 产生一个候选状态 x^* , 然后, 以概率公式(7)移动到状态 x^* , 或者保持在原状态, 不断迭代, 这个过程就形成了马尔科夫链。当样本趋于无限时, 该链的状态分布趋于目标分布

$$\alpha(x, x^*) = \min \left[1, \frac{\pi(x^*)q(x|x^*)}{\pi(x)q(x^*|x)} \right]. \tag{7}$$

MH 算法的思想非常简单, 但是, 它需要提议分布 $q(x^*|x)$ 的详细设计, 选择不同的提议分布产生不同的 MH 方法。

3 实验结果和分析

本文做了 2 个实验, 均在 Pentium (R) 4 CPU 2.80 GHz 512M 的内存的 PC 机上采用 Matlab 7 编程实现。实验一将本文提出的 SSPF 算法同 UKF 算法, 传统的 PF 算法以及 PFMH 算法(此算法为基本的 PF 算法与 MH 算法相结合的算法) 3 种常用的滤波算法进行比较; 实验二用 SSPF 算法实现了移动机器人的全局定位。

3.1 实验一

状态空间通过如下的非线性方程定义

$$\begin{cases} x_t = 0.5x_{t-1} + 25 \frac{x_{t-1}}{1+x_{t-1}^2} + 8\cos 1.2t + \epsilon_t \\ z_t = \frac{x_t^2}{20} + \eta_t \end{cases}. \tag{8}$$

系统状态初值取为 1, ϵ_t 和 η_t 分别取为方差 $\sigma_{\epsilon}^2 = 10$ 和 $\sigma_{\eta}^2 = 1$ 的高斯白噪声, 根据上式, 仿真了 $t = 1, \dots, 60$ 的数据, 从公式(8)中可以看出: 该动态模型的运动模型和观测模型都是非线性的, 使用 SSPF 的目的是从具有噪声的观测中估计真实状态, 实验中, 滤波器在每个事件点运行 200 个粒子, 分别对 4 种滤波器算法进行了 100 次独立实验。4 种滤波器算法的性能如表 1 所示, 表中比较了 100 次独立

实验中 4 种滤波算法的均方误差 (MSE) 的均值和方差, 可以看出: SSPF 算法在性能上最优。

表 1 性能对照表

Tab 1 Comparison sheet of performance

算法	均方误差	
	均 值	方 差
UKF	0.2646	0.0094
PF	0.4540	0.0535
PFMH	0.4209	0.0545
SSPF	0.0679	0.0061

3.2 实验二

3.2.1 实验内容

为了实现全局定位, 必须建立运动模型和感知模型, 一旦这 2 个模型确定, 就可以用 SSPF 算法实现全局定位。

1) 运动模型

对于两轮差分驱动移动机器人, 其概率运动模型 $p(X_t|X_{t-1}, a)$ 可以利用运动学模型建立。对于带有里程计的移动机器人, 运动行为 a 通常可以由里程计信息 (u) 表示。设机器人从位姿 $X(t-1) = \{x(t-1), y(t-1), (t-1)\}$ 处开始, 移动一段位移后到达 $X(t)$ 处。可以由里程计得到这段位移量为 $u(t) = [T(t) \ \Delta\theta(t)]^T$, 其中, $T(t)$ 为该段位移的路程长, $\Delta\theta(t)$ 为机器人方向角的改变量, $\epsilon = [\epsilon_x, \epsilon_y, \epsilon_\theta]^T$ 是运动噪声, 则可以得到机器人的运动模型如公式(9)所示

$$X(t) = X(t-1) + \begin{bmatrix} \cos(\theta(t-1)) & 0 \\ \sin(\theta(t-1)) & 0 \\ 0 & 1 \end{bmatrix} u(t-1) + \epsilon(t-1). \tag{9}$$

2) 观测模型

应用运动模型预测机器人的位姿后, 还需要更多的传感器观测信息更新估计获得更可靠的位姿。本次仿真实验中, 在机器人工作环境中, 设置了 6 个路标且位置固定, 并将 6 只射频传感器 (radio frequency tags)^[7] 分别固定在这 6 个路标上, 同时, 在移动机器人上安装了射频收发器。当机器人移动时, 射频收发器周期性地发送一定频率的射频信号作为询问请求, 在射频信号范围内的射频传感器就会发送一个独一无二的 ID 号作为响应。因为每个射频传感器发送一个独一无二的 ID 号, 所以, 得到的距离数据就会自动地与恰当的信标相匹配, 所以, 在此不需要数据匹配。假设 t 时刻接收到信息的射频路标的位置为 (x_b, y_b) , 则其观测模型可以表示成如方程(10)所示

$$Z(t) = \begin{bmatrix} \sqrt{(x_b - x(t))^2 + (y_b - y(t))^2} \\ \sqrt{(x(t) - x(t-1))^2 + (y(t) - y(t-1))^2} \\ \theta(t) - \theta(t-1) \end{bmatrix} +$$

$$\eta(t) = \begin{bmatrix} r(t) \\ T(t) \\ \Delta\theta(t) \end{bmatrix} + \eta(t), \quad (10)$$

式中 $x(t)$, $y(t)$ 和 $\theta(t)$ 为机器人在 t 时刻的位置和方向; $r(t)$ 为从射频路标到机器人所处位置的距离; $T(t)$ 和 $\Delta\theta(t)$ 分别为从 $t-1$ 时刻到 t 时刻机器人运动的位移和方向角的改变量; $\eta(t)$ 为观测噪声。

3.2.2 仿真结果

图 1 给出了仿真结果, 仿真环境大小为 $40\text{ m} \times 40\text{ m}$ 。在图 1(a), (b), (c) 中的弧线表示机器人预先设定好的运动轨迹; + 表示机器人的真实位置; ° 为粒子云集; ○ 表示估计的机器人的位置; ◇ 表示路标; * 表示在第 n ($n=1, \dots, 80$) 步做出响应的射频信标; 大圈表示射频信

标所能检测到的范围。设定实验运行的步数为 80 步 (每检测到一次路标为一步)。在开始阶段, 1 000 个粒子集均匀地分布在整個定位空间, 并且, 机器人的初始状态是未知的。当 $n=2$ 时, 粒子已经在开始逐渐收敛。随着机器人的移动, 粒子很快会收敛到少数几个集中区域。当 $n=39$ 时, 此时机器人的真实位置为 (0.98, 4.22), 而估计出的机器人的位置为 (0.99, 4.26), 其坐标值相差很小, 说明定位精度较高。在此之后的定位过程相当于一个位置跟踪的过程, 所以, 这之后只需很少的粒子就可以实现机器人的自定位。其各步的定位误差如图 2 所示。仿真结果表明: SSPF 算法可以实现准确性较高的移动机器人全局定位。

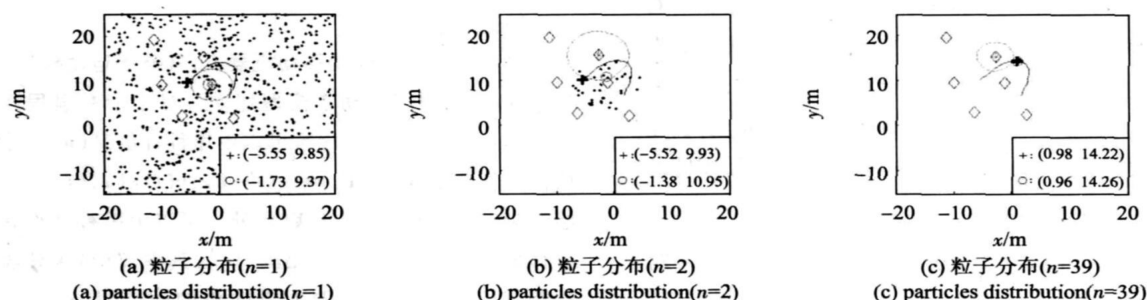


图 1 SSPF 算法定位在第 n 步的采样分布

Fig 1 Sample distribution of SSPF at n step

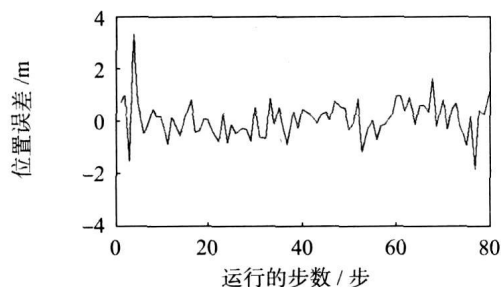


图 2 定位误差

Fig 2 Localizing errors

4 结 论

针对机器人定位这一非线性、非 Gauss 动态系统的状态估计问题, 在基本 PF 算法的基础上, 综合使用了 UKF 算法和自适应重采样算法等技术, 提出了一种定位精度更高的 SSPF 算法。并通过仿真实验, 充分验证了此方法的定位精度优于其他 PF 方法, 并且, 此方法能够较快较准确地实现机器人的全局定位问题。

参考文献:

- [1] Fox D, Burgard W, Thrun S. Markov localization for mobile robots in dynamic environments[J]. Journal of Artificial Intelligence Research, 1999, 11(3): 391-427.

- [2] Julier S J. The scaled unscented transformation[C] // Proceedings of American Control Conference, Jefferson City, MO, USA, 2002: 4555-4559.
- [3] 胡士强, 敬忠良. 粒子滤波算法综述[J]. 控制与决策, 2005, 20(4): 361-365, 371.
- [4] Thrun S, Fox D, Burgard W. Robust monte carlo localization for mobile robots[J]. Artificial Intelligence, 2001, 128(1-2): 99-141.
- [5] Fox D. Adapting the sample size in particle filters through KLD-sampling[J]. International Journal of Robotic Research, 2003, 22(12): 985-1004.
- [6] Doucet A, Gordon N. Sequential monte carlo in practice[M]. New York: Springer-Verlag, 2001.
- [7] Kantor G, Singh S. Preliminary results in range-Only localization and mapping[C] // Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Washington, DC, USA: IEEE Press, 2002: 1818-1823.

作者简介:

罗 颖(1982-), 女, 湖南邵阳人, 硕士研究生, 研究方向为智能机器人系统与应用。