

FastSLAM 算法的仿生优化改进研究

张铠翔,姜文刚

(江苏科技大学 电子信息学院,江苏 镇江 212000)

摘要:针对机器人导航标准的快速同步定位与地图构建算法(FastSLAM)在重采样过程中存在采样粒子集的贫化以及粒子多样性的缺失导致机器人的定位与建图的精度下降的问题,提出一种基于改进的蝴蝶算法来优化 FastSLAM 中的粒子滤波部分。改进的算法将机器人的最新时刻的观测和状态信息融入到蝴蝶算法的香味公式中,并在蝴蝶位置更新的过程中加入自适应香味半径和自适应蝴蝶飞行调整步长因子,来减少算法的运算时间以及提高预测精度,同时引入偏差修正指数加权算法对粒子的权值进行优化组合,对组合后部分不稳定的粒子进行分布重采样,保证粒子的多样性。通过仿真验证了该算法在估计精度与稳定性方面优于 FastSLAM,因此在移动机器人运动模型的定位与建图中具有较高的定位精度与稳定性。

关键词:快速同步定位与地图构建算法;蝴蝶算法;粒子滤波;分布重采样;预测精度

中图分类号:TP242.6

文献标识码:A

文章编号:1673-629X(2021)04-0008-06

doi:10.3969/j.issn.1673-629X.2021.04.002

Study on Improvement of Bionic Optimization of FastSLAM Algorithm

ZHANG Kai-xiang, JIANG Wen-gang

(School of Electronic Information, Jiangsu University of Science and Technology, Zhenjiang 212000, China)

Abstract: In the process of resampling, the fast simultaneous localization and mapping algorithm of robot navigation standard (FastSLAM) has problems such as the dilution of sampling particle set and the lack of particle diversity, which leads to the decrease of robot localization and map construction accuracy. Therefore, we propose an improved butterfly algorithm to optimize the particle filtering in FastSLAM. The improved algorithm integrates the latest observation and state information of the robot into the fragrance formula of the butterfly algorithm, and adds the adaptive fragrance radius and the adaptive butterfly flight adjustment step factor in the process of updating the butterfly position to reduce the operation time of the algorithm and improve the pre-operation the measurement accuracy. At the same time, the deviation correction index weighting algorithm is introduced to optimize the combination of the weights of particles, and the distribution resampling of some unstable particles after the combination is carried out to ensure the diversity of particles. Simulation results show that the proposed algorithm is superior to FastSLAM in terms of estimation accuracy and stability, so it has higher positioning accuracy and stability in mobile robot motion model.

Key words: fast simultaneous localization and mapping algorithm; butterfly algorithm; particle filtering; distribution resampling; predictive accuracy

0 引言

同时定位与地图创建(simultaneous localization and mapping, SLAM)是探索未知环境的关键因素之一,尤其是在机器人自主导航领域更是扮演着至关重要的角色。SLAM的过程就是移动机器人在未知环境中进行探索,在不断移动的过程中通过自身携带的传感器获得的控制信息与观测信息进行环境地图的增量式创建,并在建造的地图中确定自身的位置。其中的控制信息和观测信息都会存在不确定的噪声,所以可以看出 SLAM 的本质就是概率估计的问题^[1]。

一般由概率方法来解决 SLAM 问题的经典算法分为基于扩展卡尔曼滤波 SLAM(EKF-SLAM)算法和快速 SLAM(FastSLAM)算法^[2-4]。但在实际的应用环境中,移动机器人的运动模型与观测模型一般是非线性的,EKF 算法主要是通过一阶泰勒展开式来近似非线性模型,所以避免不了引入线性化误差^[5],这便导致该算法无法在大范围的环境中运行。Murphy^[5]通过对全状态滤波器分解,提出了一种基于 Rao—Blackwellise 粒子滤波器(RBPF)的新方法来解决 SLAM 问题。随后 Montemerlo 等^[6]提出的 FastSLAM

算法,对解决 SLAM 问题起到了很大的推动作用。

FastSLAM 主要的思想是将机器人的运动状态与建立地图分化成对机器人的运动估计以及基于位姿的路标估计两个主要的过程。其中在机器人运动估计中使用的则是粒子滤波进行机器人位姿定位,地图上的每个路标估计则是使用的 EKF 估计^[1]。相比一些传统的 SLAM 算法,FastSLAM 算法的计算量大大降低,但在使用粒子滤波进行运动状态估计的过程中很容易出现粒子退化这一类问题^[7],此类问题一直是众多学者研究的热点,也涌现了大量的较为优秀的改进策略。目前在算法改进策略上有一大部分的研究者通过融入仿生算法对粒子滤波进行优化,因其具有很强的寻优性能,能够快速、智能地将粒子集中至高似然区域,有效地改善了样本贫化和粒子多样性丧失现象^[8-9],进而提高了滤波精度。像基于果蝇算法优化的粒子滤波算法^[8]、基于蝙蝠算法的粒子滤波算法^[9]等都是通过结合仿生算法进行粒子优化,其过程是初始化每个粒子最初时刻位置,并通过不断迭代发现局部最优值以及全局最优值,进而不断更新位置向粒子群中最优解的位置靠近。但这些结合仿生算法优化后的粒子一定程度上存在盲目搜索,导致出现寻优结果出现较多偏差、容易陷入早熟或收敛速度较慢的问题。

针对 FastSLAM 算法存在的缺陷,融合蝴蝶算法进行优化,主要的改进策略是将蝴蝶种群觅食的思想代入到 FastSLAM 的运动状态估计中,传统的蝴蝶算法中并没有使用当前时刻的测量值,从而使得算法结果存在较大的偏差。针对此问题,该文引入机器人运动过程中最新时刻的观测值和状态值,另外传统的蝴蝶算法在局部搜索飞行过程中,通过个体的香味进行互相吸引,但需要整个群体进行交互运算,这便产生较大计算量,导致较差的实时性。因此引入自适应香味半径和自适应调整飞行步长因子^[10],同时在重采样的过程中引入偏差修正指数加权平均(exponentially weighted average, EWA)算法^[11],分离出对估计精度影响较大的大权值粒子,对剩余部分的中小权值粒子进行分布采样,以提高粒子筛选的准确性,自适应调整重采样粒子数,增加粒子的多样性,减缓粒子退化。

1 机器人传统的 FastSLAM 算法

1.1 FastSLAM 的基本原理

在 SLAM 的问题上 FastSLAM 算法将其过程进行分解,分别是机器人在移动的过程中对其运动轨迹的估计和对周围环境的特征点的估计^[12]。该算法使用带有权重的粒子表示机器人的运动状态和地图特征,第 i 个粒子为 $s_k^i = \{x_k^i, m_k, w_k^i\}$,其中 w_k^i 为 k 时刻第 i 个粒子的权重, $i = 1, 2, \dots, N$ 。用公式表述如下:

$$p(X_{1:k}, m | Z_{1:k}, u_{1:k}, n_{1:k}) = p(X_{1:k} | Z_{1:k}, u_{1:k}, n_{1:k}) \prod_{i=1}^{n_k} p(m_i | X_{1:k}, Z_{1:k}, u_{1:k}, n_{1:k}) \quad (1)$$

其中, $X_{1:k}$ 为机器人的路径估计; m 为路标位置的集合; $Z_{1:k}$ 为机器人的观测量; $u_{1:k}$ 为机器人的控制输入量; $n_{1:k}$ 为数据关联表,每个 n_k 为 k 时刻观测点到的路标点。

1.2 FastSLAM 算法的实现

FastSLAM 算法基本上由五步构成:机器人状态预测、数据关联、重要性采样、地图更新、重采样。具体步骤如下:

(1) 机器人的下一时刻的位置预测。基于每一时刻输入的控制信息和机器人运动模型进行预测。

(2) 数据关联。将观测信息和各个粒子 t 时刻的路标特征依次进行数据关联。

(3) 重要性采样。重要性权值可以根据下面的公式递归计算求得。

$$w_k^i = w_{k-1}^i \frac{p(z_k | x_k^i) p(x_k^i | x_{k-1}^i, u_k)}{q(x_k^i | x_{k-1}^i, z_k, u_k)} \quad (2)$$

粒子的归一化权值为:

$$\tilde{w}_k(X_{1:k}^{(i)}) = \frac{w_k(X_{1:k}^{(i)})}{\sum_{i=1}^N w_k(X_{1:k}^{(i)})} \quad (3)$$

(4) 地图更新。采用 EKF 算法关联观测信息更新每个特征的估计。

(5) 重采样。设置阈值并计算有效粒子数 N_{eff} 。若 N_{eff} 小于阈值则进行重采样。

重采样过程能够将分散状态空间各处粒子较快地聚合到某些区域,但这样可能存在的问题是粒子过早地收敛陷入局部最优解导致机器人运动状态偏离真实状态概率分布,从而使得粒子的估计状态存在较大偏差。

因此,将改进后的蝴蝶算法融合到 FastSLAM 算法上可以让其在处理重要性采样的时候,能够充分考虑粒子间的作用关系,得到更加准确的粒子均值与方差,这样可以很好地保证整个系统在运行过程的效率,大大地提高算法性能,在重采样过程中,对自适应重采样进行改进,提高粒子多样性,获得更加准确的定位与地图。

2 基于改进的蝴蝶算法优化的 FastSLAM 算法

2.1 蝴蝶算法

蝴蝶算法(butterfly algorithm, BA)^[13]是一种基于蝴蝶觅食行为的优化算法。在收敛精度与速度上都要

高于其他算法。其思想是在每个蝴蝶搜寻食物时,自身会发出具有一定浓度的香味,并可以通过香味的浓度吸引周围的其他蝴蝶,其他蝴蝶则会朝着香味较多的个体进行移动^[14]。影响蝴蝶的香味最主要的因素是感知形态、刺激强度以及幂指数。用方程表示为:

$$F = cI^a \quad (4)$$

其中, F 是香味浓度, c 是感知形式, I 是刺激强度, a 是幂指数。

已知目标函数 $f(x)$, 蝴蝶算法的步骤如下:

(a) 初始化蝴蝶种群, 并通过目标函数 $f(x_i)$ 确定每只蝴蝶 x_i 的刺激强度 I_i 。

(b) 计算蝴蝶种群的个体适应度值, 根据这个适应度值来确立在所定蝴蝶种群中处于最佳性能的那只蝴蝶。

(c) 计算蝴蝶散发的香味。考虑到环境因素的干扰, 使用随机数 P 作为搜索方式的更换标志, 有利于后期的位置更新。

(d) 全局搜索。低香味的蝴蝶个体向全局适应度最高的蝴蝶飞行, 全局搜索的表达式为:

$$x_{t+1}^i = x_t^i + (g^* - x_t^i) \times \text{Lévy}(\lambda) \times F_i \quad (5)$$

其中, x_t^i 为第 i 只蝴蝶在第 t 次迭代的解向量; g^* 为当前所有蝴蝶中的最优解。

(e) 局部搜索, 蝴蝶个体进行随机飞行。局部搜索的表达式为:

$$x_{t+1}^i = x_t^i + (x_t^j - x_t^k) \times \text{Lévy}(\lambda) \times F_i \quad (6)$$

其中, x_t^i, x_t^j, x_t^k 分别为第 i 只、第 j 只、第 k 只蝴蝶在第 t 次迭代的解向量; x_t^j, x_t^k 为随机个体。

为了避免蝴蝶进入局部最优, Lévy 飞行被引入到算法中。

$$\text{Lévy} \sim u = t^{-\lambda}, 1 < \lambda \leq 3 \quad (7)$$

Lévy 飞行可以有效提高搜索效率。一般 λ 取 $(1, 3]$ 。

2.2 改进蝴蝶算法

改进的蝴蝶算法粒子滤波^[14]中提出引入粒子滤波中当前时刻的观测值, 进一步地提高算法的滤波精度。该文在此基础上进行优化, 采用最新时刻观测和状态值作为反馈信息对当前目标预估结果进行修正, 并重新定义蝴蝶算法的刺激强度公式表达:

$$I_i = \exp\left[-\frac{D_i^2}{2R}\right] \quad (8)$$

$$D_i = \text{pdist}(z_{\text{pre}}^i, z_{\text{new}}^i, x_{\text{pre}}^i, x_{\text{new}}^i)_{\text{euc}} \quad (9)$$

其中, $\text{pdist}(\cdot)_{\text{euc}}$ 是欧氏距离, $z_{\text{pre}}^i, z_{\text{new}}^i, x_{\text{pre}}^i, x_{\text{new}}^i$ 为第 i 粒子的预测观测值、最新观测值、预测状态值和最新状态值, R 为观测噪声的方差。

2.2.1 自适应香味半径

蝴蝶算法具有很强的全局搜索能力, 然而一旦个

体的数量不断增大, 计算复杂度也会相应增大。改进的蝴蝶算法粒子滤波^[14]提出的自适应半径, 可以有效将蝴蝶个体间吸引的范围限定在固定区域内, 进而可以减少粒子数目的使用, 将其引入到 FastSLAM 粒子滤波阶段可以很好地降低系统的计算量。香味半径的大小由香味浓度决定, 其香味半径与香味浓度成正比关系。对于两只香味不同的蝴蝶, 位置更新的条件则是任意两个蝴蝶间距小于等于这两个蝴蝶中香味浓度最大那个蝴蝶的香味半径, 否则不更新。该文定义的半径的公式为:

$$r = \eta * F_i \quad (10)$$

其中, r 为相对吸引半径; F_i 为种群个体的香味值; η 作为调节半径参数, 一般在 $(0, 1)$ 之间取值。加入自适应半径之后相邻粒子间吸引更加快速, 则引入参数后的算法, 在准确率与运行效率上都有所提高。

2.2.2 自适应调整飞行步长因子

模拟蝴蝶算法的实际运行初期, 每个预设的蝴蝶间距较大, 较高的全局寻优能力能够使蝴蝶粒子较晚地陷入局部极值。在算法不断迭代后期, 蝴蝶的个体可移动步长逐渐变小, 较强的局部寻优能力有利于蝴蝶算法后期的收敛, 不会因可移动间距过大而跳过全局最优解。因此固定的步长因子会降低滤波整体的性能, 导致预测精度低和收敛速度慢^[15]。

针对以上问题, 该文设计一种非线性衰减函数, 加入观测噪声的方差, 保证模型对于实际目标函数有较好的近似, 兼顾了收敛速度与稳定误差, 实现 α 值的自适应调整:

$$\alpha(i) = \exp\left[-R \frac{I_i}{I_j}\right] \quad (11)$$

其中, 令两个蝴蝶之间的相对香味刺激强度为 $f(I) = \frac{I_i}{I_j}$, 可见当 $f(I) \geq 1$ 时, $\alpha(i)$ 是收敛的, 则 $\alpha(i)$ 随 $f(I)$ 增大而减小, 则步长移动缩短, 局部搜索能更强, 收敛速度提高, 相反当 $f(I) < 1$ 时, α 是发散的, 步长移动增大, 全局的搜索最优能力增大。该文将位置更新公式设置为:

$$x_{t+1}^i = x_t^i + (x_t^j - x_t^i) \times |\text{Lévy}(\lambda)| \times F_i + \alpha(i) \times (\text{rand} \div 2) \quad (12)$$

其中, x_{t+1}^i 为个体 i 在 $t+1$ 时刻的解向量; x_t^i, x_t^j 为蝴蝶粒子个体 i, j 在 t 时刻的解向量。 $|\text{Lévy}|$ 为 Lévy 飞行的绝对值, 为了避免反方向飞行。

2.3 自适应部分分布重采样

FastSLAM 中粒子滤波的重采样一般采用的是自适应重采样, 其基本思想是: 若采样后的有效粒子数小于设定的阈值, 则进行重采样, 否则不重采样。此方法可以有效地缓解粒子退化、预估精度降低等问题。为

了进一步提高粒子的多样性,该文改进自适应重采样方法,并通过基于偏差修正指数加权平均算法优化粒子筛选,选出大权值的粒子,并对中小权值的粒子进行分布重采样,并设计如下策略:

计算粒子权重并按升序排列;

采用基于偏差修正指数加权平均算法得到粒子集的平均值分布曲线,其表达式如下:

$$\delta_i = \frac{\beta \delta_{i-1} + (1 - \beta) w_i^i}{1 - \beta^i} \quad (13)$$

其中, δ_i 、 δ_{i-1} 是前 i 、 $i-1$ 个粒子的平均权重, β 为超调参数, w_i^i 为第 i 个粒子的权重, $1 - \beta^i$ 为偏差修正项。

计算 δ_i 曲线平均值 δ 并判断粒子权重 w_i^i 与 δ 的大小,若 $w_i^i > \delta$,该粒子不参与重采样,若 $w_i^i < \delta$,该粒子加入重采样序列。

该算法中的偏差修正项可以降低前期加权平均值的误差,提供更加精确的平均值,进而更好区分出大权值,对中小权值进行采样,有效缓解了粒子退化,提高了粒子的多样性。

2.4 改进后的 FastSLAM 算法实现

结合上面引入的改进蝴蝶算法和自适应部分分布重采样的算法对 FastSLAM 进行优化,优化具体步骤如下:

预测:由提议分布对当前粒子集合进行预测采样,获得下一时刻的粒子集合, $s_k \sim p(s_k | s_{k-1}, u_k)$ 。

仿生算法优化粒子集合:将粒子集融合改进后的蝴蝶算法,粒子集合根据自身的吸引半径自适应地更新位置寻找到最优解,得到更加符合真实分布的后验概率密度。

重要性采样:计算出融合蝴蝶算法优化后粒子的后验位置和姿态建议分布,从建议分布中获得新的粒子集,并归一化权重。

地图更新:结合最新观测值更新机器人位姿和路标位置。

重采样:当有效粒子数小于设置阈值时,采用改进后的自适应部分分布重采样。

改进后的 FastSLAM 中通过对粒子的仿生优化使得粒子集更加符合真实分布,以及收敛速度上也有所提高,因此可以在下一时刻得到更好的初始值。

3 实验分析

该文利用 MATLAB2012a 仿真软件分别对传统 FastSLAM、BA-FastSLAM 与文中算法 IBA-FastSLAM 在稀疏路标环境下进行仿真实验并比较,以验证提出的算法的性能。

仿真中用到的运动模型与观测模型如式(14)和式(15)所示。

$$\begin{bmatrix} x_t \\ y_t \\ \varphi_t \end{bmatrix} = \begin{bmatrix} x_{t-1} + v_t \times \Delta t \times \cos(\varphi_{t-1} + \gamma) \\ y_{t-1} + v_t \times \Delta t \times \sin(\varphi_{t-1} + \gamma) \\ \varphi_{t-1} + \frac{v_t \times \Delta t}{B} \times \sin(\gamma) \end{bmatrix} + \delta_t \quad (14)$$

其中, (x_t, y_t, φ_t) 为机器人在 t 时刻的位姿; Δt 是 $t-1$ 时刻到 t 时刻的间隔; $u = [v \ \gamma]^T$ 为机器人的控制输入,其中分别是速度和方向角; δ_t 为机器人在运动的时候所产生的高斯噪声。

$$\begin{bmatrix} r_i \\ \theta_i \end{bmatrix} = \begin{bmatrix} \sqrt{(x_i - x_t)^2 + (y_i - y_t)^2} \\ \tan^{-1}(\frac{y_i - y_t}{x_i - x_t} - \varphi_t) \end{bmatrix} + \varepsilon_i \quad (15)$$

其中, (x_i, y_i) 为地图中第 i 存在的路标点; (r_i, θ_i) 为相对于机器人的位置传感器所检测到的路标点的距离以及方向夹角; ε_i 为观测噪声,其噪声服从高斯分布。

仿真实验的机器人参数配置如表1所示。

表1 FastSLAM 仿真实验运动参数配置

| 参数 | 数值 | 参数 | 数值 |
|---------|-------|--------|-------------|
| 机器人运动速度 | 4 m/s | 最大观测范围 | 30 m |
| 最大转向角 | 30° | 运动噪声 | 0.3 m/s, 3° |
| 车轮间距 | 4 m | 观测噪声 | 0.1 m/s, 1° |

仿真实验中采用 50 个粒子,并建立约为 200 m * 160 m 的定位与建图的模拟环境,此环境一共有 36 个路标点与 18 个航向点,在以上建立的环境下对 FastSLAM、BA-FastSLAM、IBA-FastSLAM 分别进行移动机器人定位与建图的仿真实验,得到的实验结果如图1~图3所示。

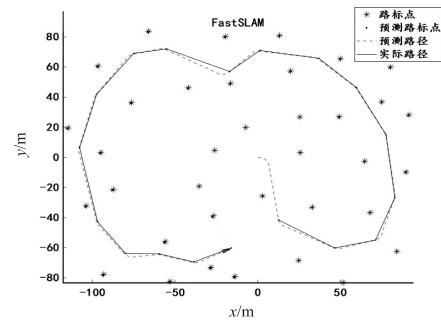


图1 FastSLAM

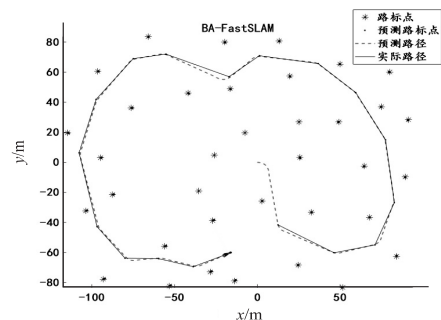


图2 BA-FastSLAM

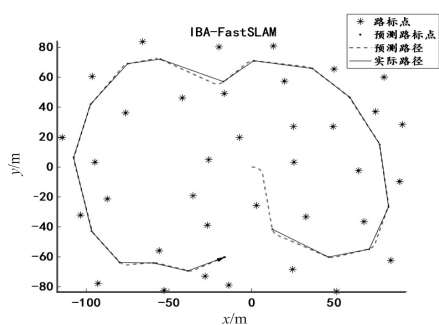
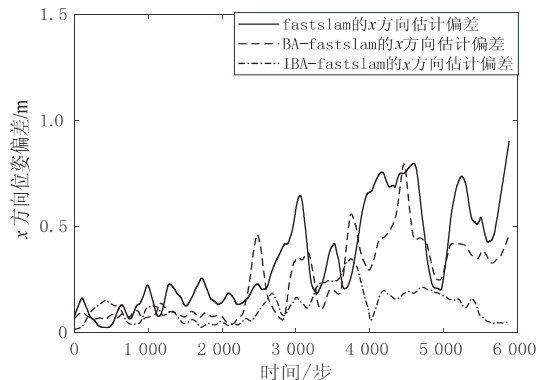
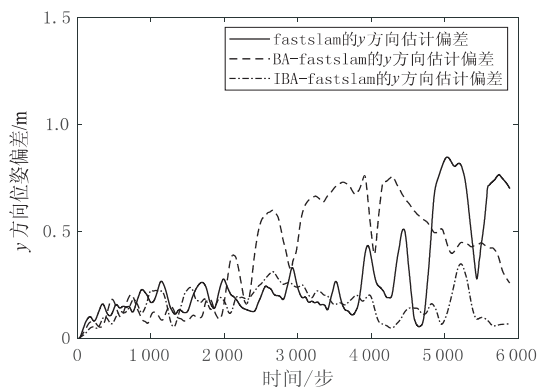


图 3 IBA-FastSLAM

由图可知,利用仿生寻优思想后一定程度上优化了粒子位置与路标位置,但依然存在较大的偏差,而文中在蝴蝶算法基础上进行改进的算法得到的估计路径更加贴合实际路径,路标的估计精度也相对提高。



(a) x 方向



(b) y 方向

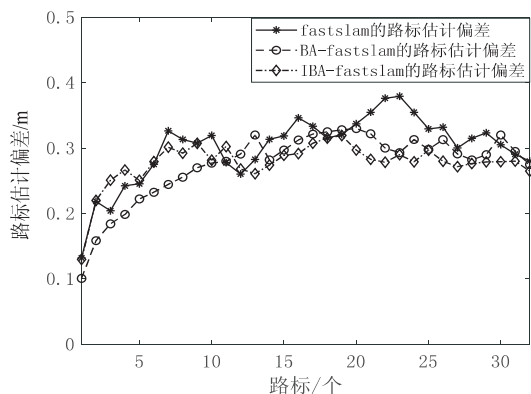
图 4 三种算法的 x 和 y 方向位姿估计误差比较

图 5 路标估计误差比较

对三种算法的性能进行对比,如图 4、图 5 所示。IBA-FastSLAM 在位姿估计偏差以及路标估计偏差上有少许部分稍微大于 BA-FastSLAM 和 FastSLAM,但整体上随着时间推移,IBA-FastSLAM 不管是位姿估计精度还是路标估计精度明显要比 BA-FastSLAM 和 FastSLAM 这两种算法高。

为了进一步验证算法的稳定性,分别利用 20, 40, 60 粒子数对两种算法进行仿真实验,计算两种算法在不同粒子数时,连续运行系统 20 次的位姿估计的均方根误差的均值和方差以作为评价算法性能的标准,如图 6 所示。

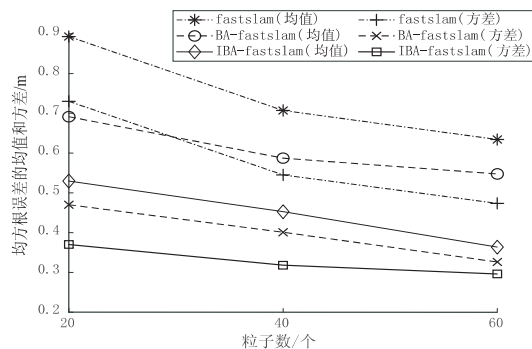


图 6 不同粒子数在位姿估计的均方根误差的均值和方差

通过图 6 所示,从整体上分析 IBA-FastSLAM 算法与原始的 BA-FastSLAM 在均方根误差的均值和方差都要高于 FastSLAM,这表明融合了蝴蝶算法后,粒子得到了优化,使得系统的精度以及稳定性都有明显的提升,一定程度上表明了仿生算法的引入,可以很好地提高原始 FastSLAM 的性能。而文中提出的 IBA-FastSLAM 算法比原始的 BA-FastSLAM 算法在均方根的均值和方差上具有更高的精度与稳定性,表明了未改进的蝴蝶算法存在一定程度的缺陷,满足不了系统的实际需求,进一步说明改进后的蝴蝶算法性能上远远要优于原始的蝴蝶算法。从不同粒子数的对比中可以看出,传统的 FastSLAM 随着粒子数的增加,误差也随之成非线性大幅度下降,可见,传统 FastSLAM 对于粒子数的依赖较大,而 BA-FastSLAM 相对于传统 FastSLAM 误差都有所降低,但对于粒子数的依赖还是较大,IBA-FastSLAM 均方根误差的均值在不同粒子数之间变化,相对于 BA-FastSLAM 与 FastSLAM 都要较平稳,其主要原因是改进后的重采样保证粒子多样性。从这一点可以看出,提出的 IBA-FastSLAM 算法对于通过增加粒子数来缓解粒子退化,提高粒子多样性以及滤波精度的需求相对降低,因此在选用粒子数量的时候可以适当减少采样粒子数,来降低计算量。

为了对三种算法在综合性能上进行比较,设定粒子数为 50,分别对三种算法连续运行 20 次,得出三种

算法在均方根的均值与方差以及运行时间对比,如表2所示。可以看出,BA-FastSLAM在预测精度与稳定性上分别比FastSLAM提高了17.10%和27.99%,表明融合蝴蝶算法后的FastSLAM在预测精度和稳定性上都有所提高,而运行时间却多于FastSLAM算法7.711秒,相当于收敛速度降低了7.9%,可见原始的BA-FastSLAM算法存在一定计算量较大的问题。而提出的IBA-FastSLAM均方根的均值和方差分别是0.379 5和0.231,在预测精度和稳定性上分别比FastSLAM提高了40.6%和41.22%,同时比改进前的BA-FastSLAM提高了28.34%和6.27%,表明在香味公式中融入最新观测值和状态值,以及引入香味半径后的蝴蝶算法提高了FastSLAM中粒子的滤波能力,减少了误差。IBA-FastSLAM在运行时间上是95.721秒,与FastSLAM的运行时间相比,缩短了3.876秒,而相对于BA-FastSLAM缩短了11.587秒,相当于收敛速度提高了10.93%,主要原因是提出的自适应蝴蝶步长因子加快了粒子向最优位置飞行速率。因此可以结合图6所得出的结论推出,IBA-FastSLAM算法可以在保持粒子数较小的情况下实现更加准确的预测,能从根本上改善粒子匮乏的现象以及粒子的多样性,从而使系统更加具有实时性,一定程度提升了系统运行的效率,因此能够使移动机器人在未知场景下进行精确定位与建图,从而合理地去决策与规划路径,更好地满足移动机器人的实际环境中的应用需求。

表2 三种算法的均方根的均值与方差以及运行时间对比

| 算法 | 均方根的均值 | 均方根的方差 | 运行时间/s |
|--------------|---------|--------|---------|
| FastSLAM | 0.638 9 | 0.393 | 98.197 |
| BA-FastSLAM | 0.529 6 | 0.283 | 105.908 |
| IBA-FastSLAM | 0.379 5 | 0.231 | 94.321 |

4 结束语

对FastSLAM的粒子滤波中的粒子集合融入了改进后的蝴蝶算法,对重要性采样过程进行优化,在粒子位置更新的过程中加入自适应香味半径和自适应蝴蝶飞行调制步长因子来控制蝴蝶种群寻优的搜索范围,能够有效保证算法的运行时间缩短和预测精度的提高,在重采样过程中加入偏差修正指数加权算法划分出稳定与不稳定的粒子集,并对划分后放入不稳定的粒子部分进行分布重采样,减少不稳定的粒子对重采样阶段的干扰,进而保证粒子的多样性。通过仿真实验结果表明,提出的IBA-FastSLAM算法可以在不需要增加过多粒子数量的情况下,既可以改善粒子的退化和多样性,又可以提升位姿与路标预测精度与稳定

性,以及算法的收敛速度,进而使机器人更实时地接近真实系统状态分布。上述方法为移动机器人的自主导航提供了一种精确的定位与建图的方法。

参考文献:

- [1] 吕太之,赵春霞.一种基于SR—UKF的FastSLAM算法[J].计算机应用研究,2012,29(10):3725-3727.
- [2] THRUN S,FOX D,BURGARD W. A probabilistic approach to concurrent mapping and localization for mobile robots[J]. Machine Learning,1998,31(1-3):29-53.
- [3] FOX D,BURGARD W,KRUPPA H,et al. A probabilistic approach to collaborative multi-robot localization[J]. Autonomous Robots,2000,8(3):325-344.
- [4] ZHOU Wu,ZHAO Chunxia. A new filter framework for FastSLAM algorithm[J]. Journal of Frontiers of Computer Science and Technology,2008,2(5):553-560.
- [5] MURPHY K P. Dynamic bayesian networks:representation,inference and learning[D]. Berkeley:University of California,2002.
- [6] MONTEMERL M,THRUN S,KOLLER D,et al. FastSLAM:a factored solution to the simultaneous localization and mapping problem[C]//Proceedings of the AAAI national conference on artificial intelligence. Edmonton, Canada: AAAI Press,2002:593-598.
- [7] 王田橙,蔡云飞,唐振民.基于区域粒子群优化和部分高斯重采样的SLAM方法[J].计算机工程,2017,43(11):310-316.
- [8] 韩 锐,张 赫.基于果蝇优化算法改进的粒子滤波及其在目标跟踪中的应用[J].湖南大学学报:自然科学版,2018,45(10):130-138.
- [9] 陈志敏,田梦楚,吴盘龙,等.基于蝙蝠算法的粒子滤波法研究[J].物理学报,2017,66(5):32-42.
- [10] 潘晓英,陈雪静,李昂儒,等.基于自适应步长的萤火虫划分聚类算法[J].计算机应用研究,2017,34(12):3576-3579.
- [11] WANG X,ZHANG C,LIU F,et al. Exponentially weighted particle filter for simultaneous localization and mapping based on magnetic field measurements[J]. IEEE Transactions on Instrumentation and Measurement,2017,66(7):1658-1667.
- [12] 于士友.移动机器人研制与基于粒子滤波的同时定位与地图创建研究[D].青岛:青岛大学,2008.
- [13] 刘云涛.基于蝴蝶优化的粒子滤波算法[J].信息技术与网络安全,2018,37(7):37-41.
- [14] 张威虎,郭明香,贺元恺,等.一种改进的蝴蝶算法优化粒子滤波算法[J].西安科技大学学报,2019,39(1):119-123.
- [15] 田梦楚,薄煜明,吴盘龙,等.基于萤火虫优化粒子滤波的新型机动目标跟踪算法[J].控制与决策,2017,32(10):1758-1766.