

Optimization of the Particle Filter Algorithm*

Lei Liang¹, Jie Cao¹, Lili Zhao²

¹School of Computer and Communication, Lanzhou University of Technology, Lanzhou

²School of Economics and Management, Lanzhou University of Technology, Lanzhou

Email: llgyape@126.com; ll241gy@163.com

Received: Aug. 31st, 2011; revised: Sep. 27th, 2011; accepted: Oct. 19th, 2011.

Abstract: In this paper, two problems were explicated by the detailed presentation of basic concepts and principles of the particle filter algorithm. One is particle impoverishment which dues to re-sampling, and another is particle degeneracy. To overcome these problems, existed methods of particle filter optimization are analyzed. Finally, this paper presented a method that combine particle filter with intelligent algorithm, the combination algorithm can overcome the shortcomings of particle filter effectively. Simulation results show that it is superior to the traditional particle filter optimization algorithm.

Keywords: Particle Filter; Particle Swarm Optimization; Genetic Algorithm; Particle Degeneracy; Particle Impoverishment

粒子滤波算法的优化与改进*

梁磊¹, 曹洁¹, 赵丽丽²

¹兰州理工大学计算机与通信学院, 兰州

²兰州理工大学经济管理学院, 兰州

Email: llgyape@126.com; ll241gy@163.com

收稿日期: 2011年8月31日; 修回日期: 2011年9月27日; 录用日期: 2011年10月19日

摘要: 本文对粒子滤波的基本概念与算法原理进行了详细的介绍, 分析了粒子滤波所存在的粒子退化问题和重采样所导致的粒子匮乏问题, 以及目前针对这两个问题所提出的粒子滤波的优化算法。最后, 给出了粒子滤波与智能算法相结合的方法。通过对粒子滤波与智能算法的结合, 可以更有效的克服粒子滤波的缺点, 仿真结果表明, 通过智能算法优化的粒子滤波的滤波性能优于传统的粒子滤波优化算法。

关键词: 粒子滤波; 粒子群优化算法; 遗传算法; 粒子退化; 粒子贫化

1. 引言

粒子滤波(particle filter, PF)于上世纪五十年代被提出, 虽然在统计学和自动控制领域得到了应用, 但由于没有解决粒子退化问题和计算量过大的制约, 并未引起科研者们足够的重视。直到 1993 年, Gordon 等提出了自举粒子滤波算法^[1], 该算法在递推过程中引入了重新采样思想以克服粒子退化问题, 同时, 计算机运算能力的急剧增强也为粒子滤波提供了客观条件, 随后粒子滤波得到长足发展。

由于粒子滤波技术适用于任何能用状态空间模型

*资助信息: 甘省自然科学基金项目(1014ZSB064)。

表示的非高斯背景的非线性随机系统, 精度可以逼近最优估计, 更接近于工程现实。所以其应用极其广泛。在目标跟踪、故障诊断、金融等领域都体现出其先进的优越性。

2. 基本粒子滤波算法

粒子滤波是一种基于序贯蒙特卡罗方法和递推贝叶斯估计的统计滤波方法^[2]。其基本思想是: 首先依据系统状态向量的经验条件分布 $p(x_0)$, 在状态空间产生一组随机样本的集合 $\{x_{0:k}^i, w_k^i\}_{i=1}^{N_s}$, 称这些样本为粒子。然后根据量测不断调整粒子的权重和位置, 通

过调整后的粒子信息修正最初的经验条件分布。当样本容量很大时,可近似于状态变量真实的后验概率密度 $p(x_k | z_k)$ 。

2.1. 最优贝叶斯估计

设动态空间模型如下:

$$\begin{cases} x_k = f_k(x_{k-1}, v_{k-1}) \\ z_k = h_k(x_k, w_k) \end{cases} \quad (1)$$

若已知状态的初始概率密度函数为

$$p(x_0 | z_0) = p(x_0),$$

则状态预测方程为:

$$p(x_k | z_{1:k-1}) = \int p(x_k | x_{k-1}) p(x_{k-1} | z_{1:k-1}) dx_{k-1} \quad (2)$$

状态更新方程为:

$$p(x_k | z_{1:k}) = \frac{p(z_k | x_k) p(x_k | z_{1:k-1})}{p(z_k | z_{1:k-1})} \quad (3)$$

其中 $p(z_k | z_{1:k-1}) = \int p(z_k | x_k) p(x_k | z_{1:k-1}) dx_k$, 如果(2)、(3)表示的概率密度函数是线性高斯的,则其可由均值和方差来表示。此时,应用 Kalman 滤波所得结果可达到最优值。但如果是非线性非高斯模型,则很难得到完整的解析式来描述这样的概率密度函数

2.2. SIS 算法

采用蒙特卡罗方法可将积分运算转化为有限样本点的求和运算,如果能从 $p(x_{0:k} | z_{1:k})$ 中抽取 N 个独立的同分布的样本 $\{x_{0:k}^i, i=1, \dots, N\}$, 那么状态概率分布可用下面的经验概率分布近似表示:

$$\hat{p}(x_{0:k} | z_{1:k}) \approx \sum_{i=1}^N w_k^i \delta(x_{0:k} - x_{0:k}^i) \quad (4)$$

权值满足 $\sum_{i=1}^N w_k^i = 1$, 当 $N \rightarrow \infty$ 时, $\hat{p}(x_{0:k} | z_{1:k})$ 绝对收敛于 $p(x_{0:k} | z_{1:k})$ 。

在采样时,直接从状态的概率密度函数中取出样本是很困难的。但若从一个重要密度函数中采样则相对简单。可从 $q(x_{0:k} | z_{1:k-1})$ 中独立抽取 N 个样本 $\{x_{0:k}^i, i=1, \dots, N\}$, 状态的概率密度函数逼近为

$$\begin{cases} \hat{p}(x_{0:k} | z_{1:k}) = \sum_{i=1}^N \hat{w}_k^i \delta(x_{0:k} - x_{0:k}^i) \\ \hat{w}_k^i = w_k^i / \sum_{i=1}^N w_k^i \end{cases} \quad (5)$$

其中 $w_k^i \propto \frac{p(x_{0:k}^i | z_{1:k})}{q(x_{0:k}^i | z_{1:k})}$ 称为重要性权值。为了递推估计,先取重要性分布函数为:

$q(x_{0:k} | z_{1:k}) = q(x_k | x_{0:k-1}, z_{1:k}) q(x_{0:k-1} | z_{1:k-1})$ 可以得到新的粒子集 $\{x_{0:k}^i\}_{i=1}^{N_s}$, 重要性权值更新公式为:

$$w_k^i \propto w_{k-1}^i \frac{p(z_k | x_k^i) p(x_k^i | x_{k-1}^i)}{q(x_k^i | x_{k-1}^i, z_k)} \quad (6)$$

归一化权值,即可得到一组加权的粒子 $\{x_{0:k}^i, \hat{w}_k^i, i=1, \dots, N\}$ 。状态的概率密度函数可由(5)计算得出。

3. 粒子滤波算法存在的主要问题

当用重要性函数替代后验概率分布作为采样函数时,理想情况是重要性函数非常接近后验概率分布,也就是希望重要性函数的方差基本为零,即 $Var(w_k^i) = 0$ 。但由于没有考虑当前的量测值,从重要性概率密度中取样得到的样本与从真实后验概率密度采样得到的样本有很大偏差,特别是当似然函数呈尖峰状态或在系统状态转移概率密度的尾部时,偏差就变得更为明显。当重要性权重的方差随时间递增至,粒子的权重集中在少数粒子上,甚至在几步递归后,可能只有一个粒子有非零权值,从而使得大量的计算工作都被浪费在用来更新那些对后验概率分布 $p(x_k | z_{1:k})$ 的估计几乎不起作用的粒子上,结果粒子集无法表达实际的后验概率分布,这就是粒子滤波算法的退化问题^[3]。

3.1. 重采样

为避免退化现象的发生,引入了重采样技术。重采样的基本思想是通过后验概率

$p(x_k | z_{1:k}) \approx \sum_{i=1}^{N_s} w_k^i \cdot \delta(x_k - x_k^i)$ 重采样 N_s 次,产生新的粒子集 $\{x_k^{i*}\}_{i=1}^{N_s}$, 使得 $p(x_k^{i*} = x_k^j) = w_k^j$ 。由于重采样是独立同分布的,权值被重新设置为 $w_k^j = \frac{1}{N_s}$ 。判断是

否重采样的依据是粒子的退化程度 $N_{eff} = 1 / \sum_{i=1}^{N_s} (w_k^i)^2$, N_{eff} 越小,意味着退化现象越严重。首先设定一个有效样本数 $N_{threshold}$ 作为阈值,当 $N_{eff} = 1 / \sum_{i=1}^{N_s} (w_k^i)^2 < N_{threshold}$

时, 则进行重采样, 这样就无需在每个时刻都进行重采样, 从而能够自适应地根据样本情况决定是否要进行重采样, 可以在一定程度上降低算法复杂度。

但重采样也带来了新的问题, 即粒子匮乏。由于较大权值的粒子被多次选择, 采样结果中包含许多重复点, 降低了粒子的多样性^[4]。

3.2. 重要密度函数

选取好的重要密度函数也可以改善粒子退化现象。选取重要密度函数时以使重要密度函数容易采样和使得权系数的方差最小为准则。Doucet 等给出了在给定 $\{x_{0:k-1}^i\}_{i=1}^{N_s}$ 及 $z_{1:k}$ 的情形下使得权系数方差最小的最优重要密度函数是:

$$q(x_k | x_{k-1}^i, z_k) = p(x_k | x_{k-1}^i, z_k) = \frac{p(z_k | x_k, x_{k-1}^i) \cdot p(x_k | x_{k-1}^i)}{p(z_k | x_{k-1}^i)} \quad (7)$$

将(7)式代入(6)式得:

$$w_k^i \propto w_{k-1}^i \cdot p(z_k | x_{k-1}^i) = w_{k-1}^i \int p(z_k | x_k^i) \cdot p(x_k^i | x_{k-1}^i) dx_k^i \quad (8)$$

可见, 选取重要密度时, 需要从 $p(x_k | x_{k-1}^i, z_k)$ 抽取样本并估算积分 $\int p(z_k | x_k^i) \cdot p(x_k^i | x_{k-1}^i) dx_k^i$ 。当 x_k 为有限集或 $p(x_k | x_{k-1}^i, z_k)$ 为高斯函数时这是可行的, 但在一般情况下, 并不能直接解决这两个问题, 得到最优的重要密度函数的困难程度与直接从目标概率中抽取样本的困难程度相同, 但从最优重要密度函数的形式可以看出, 产生下一代预测样本依赖于已有的样本和量测数据。由此可见, 如何平衡效率和实现难易度之间的矛盾, 选择合适的重要密度函数是粒子滤波算法设计中关键的步骤。

4. 粒子滤波算法的优化方法

4.1. 基于重要密度函数的改进

基于重要密度函数选取问题, 目前较为热门的解决方法有: 高斯厄米粒子滤波(Gauss-hermite Particle Filter, GHPF)^[5], 它采用 GH 来产生重要密度函数, GH 积分公式是一种高斯型积分公式, 可以通过高斯点变换公式 $x_i = \left(\sqrt{p_{k|k-1}}\right)^T q_i + x_{k|k-1}$ 提高代数精度; 扩

展卡尔曼粒子滤波(Extended Kalman Particle Filter, EKPF)^[6], 它是通过 EKF 来更新粒子:

$$\left[\left\{ x_k^i, \tilde{w}_k^i \right\}_{i=1}^N \right] = \text{EKPF} \left[\left\{ x_{k-1}^i, \tilde{p}_{k-1}^i \right\}_{i=1}^N, z_k \right] \quad (9)$$

$$x_{k|k-1}^i = f(x_{k-1}^i) \quad (10)$$

$$P_{k|k-1}^i = F_k^i P_{k-1}^i (F_k^i)^T + F_k^i Q_k (F_k^i)^T \quad (11)$$

$$K_k = P_{k|k-1}^i H_k^i \left[U_k^i R_k (U_k^i)^T + H_k^i P_{k|k-1}^i (H_k^i)^T \right]^{-1} \quad (12)$$

$$\hat{x}_k^i = \hat{x}_{k|k-1}^i + K_k \left(z_k - h(\hat{x}_{k|k-1}^i) \right) \quad (13)$$

$$\hat{P}_k^i = P_{k|k-1}^i - K_k H_k^i P_{k|k-1}^i \quad (14)$$

EKF 是一种常用的非线性滤波方法, 主要通过对原来的非线性系统进行一阶泰勒近似, 将非线性系统方程和量测方程近似线性化, 然后用常规卡尔曼滤波方法进行状态估计。EKF 结合最新的量测值, 通过高斯近似不断更新后验分布来实现递推估计; 无味粒子滤波是利用无味卡尔曼滤波(Unscented Kalman Filter, UKF)^[7]方法来生成下一个预测粒子, 由于充分考虑了最新的观测值, 从而提高了估计精度。其基本思想是在使用 Unscented 变换的基础上, 加入最新的观测量并产生非线性粒子滤波的建议分布:

$$q(X_k/Z_k) = q(x_0) \prod_{j=1}^k q(x_j/X_{j-1}, Z_j) \quad (15)$$

式中, $X_k = \{x_1, x_2, \dots, x_k\}$ 和 $Z_k = \{z_1, z_2, \dots, z_k\}$ 分别表示 k 时刻所有状态序列和观测值序列, 由于状态序列服从一阶马尔可夫过程, 因而建议分布 $q(X_k/Z_k)$ 可分解成式(15)的形式。

4.2. 基于重采样的改进

基于重采样中粒子匮乏的问题, 目前较为热门的解决方法有: 重采样移动算法^[8], 在采样之后加上一步 MCMC(Markov chain Monte Carlo)移动处理使粒子集趋于平稳分布 $p(z)K(z, z') = p(z')K(z', z)$, 减弱粒子间的相关性; 而且 Markov 链能使粒子分布更加接近状态概率密度函数分布, 使样本分布更加合理; 辅助变量粒子滤波(Auxiliary Variable Particle Filter, AVPF)^[9], 在重采样前, AVPF 依据似然值的大小对

原粒子集中的各个权值进行修正, 得到粒子集为 $\left\{ \left(x_k^i, \xi_{k-1}^i, w_k^i \right)_{i=1}^N \sim q \left(x_k, \xi_{k-1}^i \mid z_{0:k} \right) \right\}$, 丢弃其中的辅助变量 ξ_{k-1}^i 可得到所需的粒子集 $\left\{ \left(x_k^i, w_k^i \right)_{i=1}^N \right\}$, 使得重采样后的粒子向似然函数的高值区移动。该算法从经过平滑的后验密度中采样, 更接近于状态的真值, 因此, 可以获得更小的权值方差。但是, 由于在 AVPF 算法的一次迭代中, 对于每个粒子需要计算两次似然函数和权值, 所以, 其计算量增大了。正则粒子滤波(Regularized Particle Filter, RPF)^[10], 正则化方法从后验概率 $p(x_k \mid z_{1:k})$ 的连续近似中进行重采样, $p(x_k \mid z_{1:k}) \approx \sum_{i=1}^N w_k^i K_h(x_k - x_k^i)$ 通过引入核密度函数和核带宽系数 $K_h(x) = \frac{1}{h^{n_x}} K\left(\frac{x}{h}\right)$ 以连续形式计算状态后验概率, 与标准粒子滤波的离散形式相比, 该算法可以有效缓解重采样过程造成的粒子退化问题, 在过程噪声较小时可获得较好的滤波精度。

5. 粒子滤波算法与其它智能算法的结合

近几年来, 科研者们更多的关注于粒子滤波与其它智能算法的结合。通过其它智能算法来优化粒子滤波, 改善了粒子退化与粒子匮乏现象。并取得了一定的成果。

5.1. 粒子群优化粒子滤波

粒子群优化^[11](Particle Swarm Optimization, PSO) 算法是由美国心理学家 Kennedy 和电气工程师 Eberhart 于 1995 年提出的。标准的 PSO 算法是一种有效的全局寻优算法, 它是基于群体智能理论的优化算法, 通过群体中粒子间的合作和竞争产生的群体指导优化搜索。PSO 算法基于种群的全局搜索策略, 将每个个体微粒看作是在 n 维搜索空间中的一个没有重量和体积的微粒, 并在搜索空间中以一定的速度飞行, 该飞行速度由个体的飞行经验和群体的飞行经验进行动态调整。在每次迭代中, 每个个体微粒根据下式来调整其飞行速度和位置:

$$v_{ij}(t+1) = wv_{ij}(t) + c_1r_{1j} [p_{ij}(t) - x_{ij}(t)] + c_2r_{2j} [g_{ij}(t) - x_{ij}(t)] \quad (16)$$

$$x_{ij}(t+1) = x_{ij}(t) + v_{ij}(t+1) \quad (17)$$

PSO 算法可以表述为: 随机初始化一个粒子群(数量为 m), 其中, 每 i 个粒子在 n 维空间的位置表示为 $X_i = (x_{i1}, x_{i2}, \dots, x_{in})$, 速度为 $V_i = (v_{i1}, v_{i2}, \dots, v_{in})$ 。每一次迭代, 粒子通过两个极值来更新自己的速度和位置: 一个是粒子本身从初始到当前迭代次数搜索产生的最优解, 称为个体极值 $P_i = (p_{i1}, p_{i2}, \dots, p_{in})$; 另一个是种群目前的最优解, 称为全局极值 $G = (g_1, g_2, \dots, g_n)$ 。在找到这两个最优值后, 每个粒子根据下式更新其速度和位置:

$$V_i = w * V_i + c_1 * \text{Rand}() * (P_i - X_i) + c_2 * \text{Rand}() * (G - X_i) \quad (18)$$

$$X = X_i + V_i \quad (19)$$

Rand 是介于(0, 1)区间的随机数; w 称为惯性系数; $c_1 = c_2 = 2$ 为学习因子。通过移动粒子群向最优粒子靠近, PSO 算法实际是驱动所有粒子向高似然区移动。选定一阈值 ε , 当粒子群的最优值与其符合时, 则说明粒子群分布在真实状态附近, 停止优化。再对粒子集进行权重更新和归一化处理, 为解决粒子退化问题, 需要选择和复制权重值较大的粒子, 即对粒子集进行重采样。在重采样之后, 真实状态附近的粒子权重值将会增大。通过上述优化过程, 使得粒子集在权重更新前更加趋向于高似然区域, 从而解决了粒子匮乏问题。

5.2. 遗传粒子滤波

遗传粒子滤波(GAPF)是遗传算法^[12]与粒子滤波的结合。遗传算法本身反映的就是一种进化思想, 针对解决标准粒子滤波算法中存在粒子退化及运算量过大的问题, 将遗传算法中的选择、交叉、变异操作引入到粒子滤波中, 代替传统的重采样方法。具体实现步骤如下: 1) 选择操作。对 k 时刻加权粒子集 $\{x_k^i, w_k^i\}_{i=1}^{N_s}$, w_k^i 视为此粒子群中对应粒子的适应度, 计算适应度的方差, 也就是粒子重要性权值的方差。由于粒子重要性权值的方差最小时, 状态估计最接近真实状态。因此, 根据适应度的方差大小来判断选择操作。若方差大小符合要求, 进行选择操作, 保留所有粒子, 而不再进行交叉、变异操作, 直接进入预测阶段; 若方差大小不符合要求, 跳过选择操作, 直接进行交叉、变异操作。2) 交叉操作。从粒子集中随机选取两个粒子 $(x_k^m, x_k^n)_{m,n=1}^{N_s}$, 按照

下面两式进行交叉操作:

$$\tilde{x}_k^m = \alpha x_k^m + (1-\alpha)x_k^n + \eta \quad (20)$$

$$\tilde{x}_k^n = \alpha x_k^n + (1-\alpha)x_k^m + \eta \quad (21)$$

式中 $\eta \sim N(0, \Sigma)$, $\alpha \sim U(0,1)$ 交叉准则为若 $p(z_k | \tilde{x}_k^m) > \max\{p(z_k | x_k^m), p(z_k | x_k^n)\}$, 则接受粒子 \tilde{x}_k^m ; 否则, 接受概率为 $p(z_k | \tilde{x}_k^m) / \max\{p(z_k | x_k^m), p(z_k | x_k^n)\}$ 的粒子。接受和放弃粒子 \tilde{x}_k^n 的方法与 \tilde{x}_k^m 相同。(3)变异操作。从粒子集中随机选择一个粒子 $(x_k^j)_{j=1}^{N_s}$, 按照下式进行变异操作: $\tilde{x}_k^j = x_k^j + \eta$, 变异准则为: 如果 $p(z_k | \tilde{x}_k^i) > p(z_k | \tilde{x}_k^j)$, 则接受粒子 \tilde{x}_k^i ; 否则, 接受粒子概率为 $p(z_k | \tilde{x}_k^i) / p(z_k | \tilde{x}_k^j)$ 的粒子。将上述遗传算法的选择、交叉与变异操作替换标准粒子滤波中的重采样, 可有效的改善粒子滤波算法中粒子退化与计算量过大的问题。

6. 仿真结果与实验分析

本文选取非静态增长模型(UNGM 模型), 应用 Matlab 软件进行仿真。系统过程模型与状态模型如下:

$$x(t) = 0.5x(t-1) + \frac{25x(t-1)}{1 + [x(t-1)]^2} + 8\cos(1.2(t-1)) + w(t) \quad (22)$$

$$z(t) = x(t)^2 / 20 + v(t) \quad (23)$$

本文选择在量测噪声方差 $R=1$, 过程噪声方差分别为 $Q=10$ 的噪声条件下, 来进行仿真实验。对粒子数分别取 $N=100$ 的 UPF、RPF、PSOPF 和 GPF 算法进行仿真比较。单次实验的均方根误差公式为:

$$RMSE = \left[\frac{1}{N} \sum_{i=1}^n (x_n - \hat{x}_n)^2 \right]^{1/2}$$

。平均有效样本数为 \bar{N}_{eff} ,

在同等情况下, 有效样本数越多, 估计精度越高。时间步长为 50, 仿真 50 次, 算法终止条件设为迭代次数 $n=50$ 次。下面分别给出 UPF 和 PSOPF、RPF 和 GPF 算法单次实验的估计仿真图 1~2:

本文实验通过对同是改进粒子滤波重要性密度函数部分的 UPF 和 PSOPF 算法的对比、同是改进粒子滤波重采样部分的 RPF 和 GPF 算法的对比, 说明各改进算法的滤波性能。图 1~2 中可以明显看出, PSOPF

算法和 GPF 算法的仿真估计曲线相比于 UPF 和 RPF 更接近真实值, 说明 PSOPF 和 GPF 算法的滤波精度高于 UPF 和 RPF 算法。表 1 所统计的实验平均数据可看出, 在相同噪声环境, 同等粒子数目条件下,

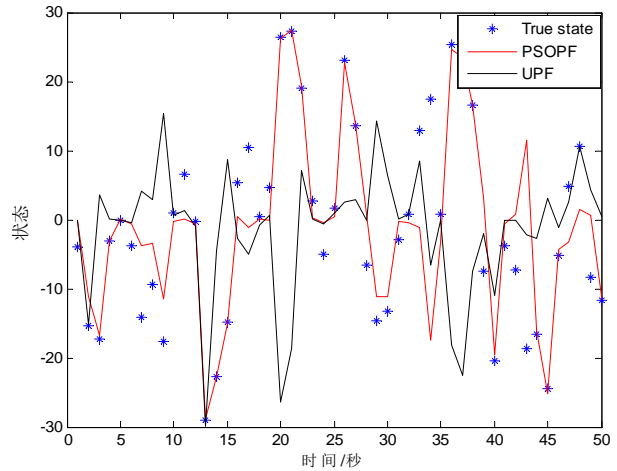


Figure 1. State estimation of UPF and PSOPF
图 1. UPF 与 PSOPF 算法仿真实验图

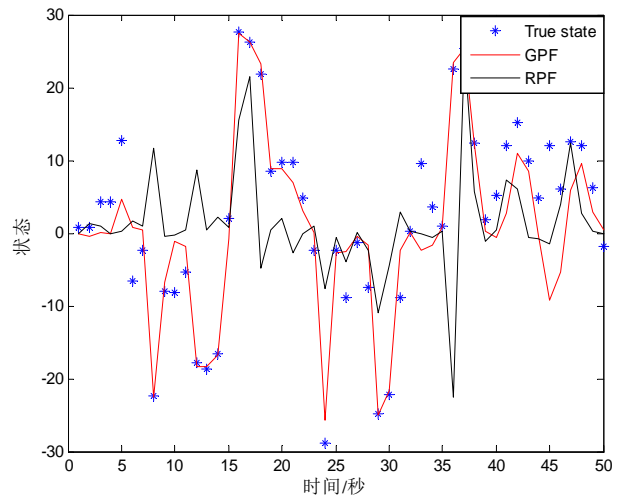


Figure 2. State estimation of RPF and GPF
图 2. RPF 与 GPF 算法仿真实验图

Table 1. Statistics of Filter performance
表 1. 滤波性能数据统计表

过程噪声方差为 10, 量测噪声方差为 1				
算法	粒子数 N	平均有效样本	RMSE	运行时间
UPF	100	15.452	2.4583	0.3259
PSOPF	100	18.816	1.8617	0.4369
RPF	100	16.547	2.2136	0.2495
GPF	100	20.189	1.6273	0.4172

PSOPF 和 GPF 的平均有效样本要多于 UPF 和 RPF, 说明它们有效地抑制了粒子退化现象, 提高了粒子多样性。且 PSOPF 和 GA 的 RMSE 也较小, 说明它们的滤波精度更高, 虽然在估计时间上有略有增长, 但并没超出实时性的要求。

7. 结束语

针对粒子滤波的退化与贫乏问题, 科研者们提出了许多改进的方法, 但仍未能在数学上解决算法收敛性的证明问题。若能有效解决收敛性问题, 则对粒子退化现象的抑制将有很大帮助。为了提高粒子滤波算法的运算速度和鲁棒性, 研究粒子滤波的硬件实现方法也尤为关键。目前, 粒子滤波算法在国外发展得很快, 并取得了许多研究成果。国内许多学者也开始积极探索粒子滤波的各种改进算法, 并成功地将粒子滤波算法应用到人脸识别、语音增强、目标跟踪、故障诊断等诸多领域, 促进了国内粒子滤波算法理论和应用研究的发展。

参考文献 (References)

- [1] N. J. Gordon, D. J. Salmond and A. F. M. Smith. A novel approach to nonlinear and non-Gaussian Bayesian state estimation. *IEEE Proceeding*, 1993, 140:107-113.
- [2] J. I. Ababneh, M. H. Batained. Linear phase FIR filter design using particle swarm optimization and genetic algorithms. *Digital Signal Processing*, 2008, 18(4): 657-668.
- [3] T. Bengtsson, P. Bickel. Curse-of-dimensionality revisited: Collapse of the particle filter in very large scale systems. *Institute of Mathematical Statistics Collections*, 2008, 2: 316-334.
- [4] A. Bain, D. Crisan. A continuous time particle filter. *Applications of Mathematics*, 2009, 60: 221-256.
- [5] 王宁. 基于高斯厄米粒子滤波的红外点目标跟踪算法研究[D]. 南京航空航天大学, 2007.
- [6] T.-Y. Xu, S.-L. Liu. Algorithm of 3-Dsingle observer passive location with the extended dalman particle filter. *IEEE Computer Society Proceeding*, 2009: 4704-4707.
- [7] J. K. Lee, C. Jekeli. Rao-blackwellized unscented particle filter for a handheld unexploded ordnance geolocation system using IMU/GPS. *Navigation*, 2011, 64(2): 327-340.
- [8] M. Johannes, N. Polson. Markov chain Monte Carlo. *Mathematics and Statistics*, 2009, part 7: 1001-1013.
- [9] W. Nick, S. Sumeetpal and G. Simon. Auxiliary particle implementation of probability hypothesis density filter. *Aerospce and Electronic Systems*, 2010, 46(3): 1437-1454.
- [10] S.-K. Park, J. P. Hwang and E. Kim. A new evolutionary particle filter for the prevention of sample impoverishment. *Evolutionary Computation*, 2009, 13(4): 801-809
- [11] Q.-K. Pan, M. F. Tasgetiren and Y.-C. Liang. A discrete particle swarm optimization for the no-wait flowshop scheduling problem. *Computers & Operations Research*, 2008, 35(9): 2807-2839.
- [12] U. Maulik, S. Bandyopadhyay and A. Mukhopadhyay. Multi-objective genetic algorithm-based fuzzy clustering. *Computer Science*, 2011: 89-121.