

高斯衍生粒子滤波器

宁小磊¹, 王宏力¹, 宁宇琪², 张琪¹

(1. 第二炮兵工程学院测控系, 710025, 西安; 2. 西安交通大学经济与金融学院, 710061, 西安)

摘要: 针对基于高斯滤波的重要性采样方法运算量的明显增加主要集中在使用高斯滤波生成更好的重要性密度函数的问题,提出了一种新的高斯衍生粒子滤波算法(GDPF). 该算法将一种类似光子衍射的粒子衍生重要性采样方法与现有的高斯辅助粒子滤波算法(GAPF)相结合,通过粒子的扩张与收缩,在保证不减少参与状态估计的粒子数的条件下减少更新粒子数,根据粒子权值大小自适应地调整衍生粒子数,能很好地缓解精度与运算量之间的矛盾,抑制粒子退化等问题. 对衍生粒子进行理论分析,证明了其与高斯采样粒子的等效性. 仿真结果表明,当选取了相同的参与状态估计的粒子数时,所提算法保持了与原算法相当的估计精度,同时运算量大大降低.

关键词: 粒子滤波;高斯滤波;粒子退化;粒子衍生

中图分类号: TP391 **文献标志码:** A **文章编号:** 0253-987X(2010)06-0072-06

Gaussian Diffracted Particle Filter

NING Xiaolei¹, WANG Hongli¹, NING Yuqi², ZHANG Qi¹

(1. Controlling and Testing Department, The Second Artillery Engineering College, Xi'an 710025, China;

2. School of Economics and Finance, Xi'an Jiaotong University, Xi'an 710061, China)

Abstract: The particle filter combined with Gaussian filter can restrict particle degeneracy with a secondary result that the new particle filter has a high calculation cost. In order to reduce the expensive calculation cost in the Gaussian aided particle filter (GAPF), a Gaussian diffracted particle filter (GDPF) is proposed by introducing a light-diffracting-like particle diffracting sampling method into the current GAPF. The proposed method predicts fewer particles, and keeps more particles to be re-sampled from each Gaussian importance density function, so that the overall particles in the estimation of the system state are maintained the same by extending and contracting of particles. The number of particles is also adjusted according to particle weights. Therefore the calculation cost in GDPF is significantly reduced when the accuracy is required the same as the GAPF method, and the sample degeneracy problem is successfully improved. Theoretical analysis indicates that the efficiencies of both GDPF and GAPF are the same. The results of Monte Carlo simulations with the same number of particles in state estimation show that the improved particle filter can preserve the same accuracy of estimation while computation burden is greatly reduced.

Keywords: particle filter; Gaussian filter; particle degeneracy; particle diffracting

粒子滤波(PF)通过将具有强大计算能力的 Monte Carlo 采样方法和 Bayes 推断结合起来,从而实现了参数和状态的在线估计. 由于不受非线性、非高斯系统的限制,目前被广泛应用于目标跟踪、故障诊断、信号处理和自动控制等领域^[1-5]. 然而,在标准粒子滤波器的设计过程中,普遍存在运算量较大和粒子退化等问题^[3-10],严重限制了其在工程领域中的应用. 针对粒子退化问题,学者们提出了多种改进

方法,其中最著名的是基于高斯滤波重要性采样的粒子滤波改进算法,如扩展卡尔曼粒子滤波算法(EPF)、基于样本 Unscented 变换的卡尔曼粒子滤波算法(UPF)和高斯-厄米特粒子滤波算法(GH-PF)等^[2-10]. 该方法使用高斯滤波对每个粒子生成一个高斯型重要性密度函数,然后再从该高斯型重要性密度函数中抽取粒子. 由于考虑了系统最新观测信息,该方法能够有效地抑制粒子退化问题,达到改善粒子滤波性能的目的,称这类混合滤波器为高斯辅助粒子滤波器(GAPF). 然而,从 GAPF 的流程看,GAPF 是在标准粒子滤波的基础上,增加高斯滤波对每个粒子进行更新的步骤. 因此,在选定相同粒子数参与状态估计的情况下,GAPF 在滤波精度上有所提高,同时运算量也明显增加了.

本文为了提高 GAPF 的实时性,提出一种类似光子衍射的粒子衍生过程,并将该过程与已有的 GAPF 相结合,得到一种新的高斯衍生粒子滤波器(GDPF).

1 高斯衍生粒子滤波算法

分析已有 GAPF 的算法流程^[2-10],其运算量增加主要集中在使用高斯滤波生成更好的重要性密度函数,并从中抽取样本. 为了解决 GAPF 在滤波精度提高的同时运算量明显增加的问题,一个最直接的方法是减少高斯滤波更新粒子的时间,但对 GAPF 而言,减少粒子更新时间也就是要相应地减少更新的粒子数,这将会导致粒子滤波精度的下降. 针对以上矛盾,提出通过减少高斯滤波更新粒子的运行时间,但同时并不减少参与状态估计的粒子数,从而实现在保持一定滤波精度的前提下提高滤波的实时性.

1.1 算法的总体思路

GDPF 的基本思路是减少高斯滤波更新的粒子数,相应地增加从高斯型重要性密度函数中抽取的粒子,多抽取的粒子数至少等于或多于高斯滤波更新减少的粒子数. 该算法在保证不减少参与状态估计粒子数的条件下减少了粒子更新时间,但同时增加了粒子抽样时间. 由于更新 1 个粒子花费的时间比从高斯型重要性密度函数中抽取 1 个粒子花费的时间多,因此 GDPF 能够有效地缓解粒子滤波精度和实时性之间的矛盾. 下面给出实现 GDPF 的关键技术.

(1)更新粒子. 根据实际应用的需要,确定参与状态估计的粒子数 N ,并确定需要更新的粒子数

N' . 此时,应有 $N > N'$,这样才能体现 GDPF 的优势. 其中, N' 个粒子用于生成下一步采样所需的一簇高斯型重要性密度函数,即采样后的粒子由高斯滤波进行更新,所得的均值和方差作为新采样用的高斯型重要性密度函数的均值和方差. N 和 N' 的具体关系一般要根据具体的实际应用背景确定.

(2)衍生粒子. 从高斯滤波生成的每个重要性密度函数中抽取 M^i 个粒子,同时将对应的粒子权值进行分裂,分配给抽取的 M^i 个粒子,得到新粒子集 $\{x_k^j, w_k^j/M^i; i=1, 2, \dots, N', j=1, 2, \dots, M^i\}$. 这个粒子抽取过程类似于光子的衍射,因此形象地称其为“粒子衍生”过程,这也是 GDPF 名称的由来.

(3)输出结果. 对粒子集 $\{x_k^j, w_k^j/M^i; i=1, 2, \dots, N', j=1, 2, \dots, M^i\}$ 进行权值更新、归一化后,输出估计结果(均值、方差和后验分布). 这时,参与状态估计的粒子明显多于更新粒子,可保证一定的滤波精度. 为了继续迭代运算,还需要解决的问题是如何使衍生后的粒子数 $\sum_{i=1}^N M^i$ 恢复到粒子更新时的粒子数 N' .

(4)权值恢复. 按粒子-权值递增的顺序对粒子集 $\{x_k^j, \tilde{w}_k^j; i=1, 2, \dots, N', j=1, 2, \dots, M^i\}$ 排队,参照初始预测样本容量保留前面的 N' 个高权值粒子,抛弃剩余的低权值粒子. 此时,留下的粒子数比参与估计过程的粒子数少,即 $\sum_{i=1}^N M^i = N > N'$,它们已经不满足归一化权值的关系式 $\sum_{i=1}^{N'} \tilde{w}_k^i = 1$,因此需要对保留下来的这些粒子重新进行权值归一化处理,称这个过程为“权值恢复”. 其具体做法是对保留下来的每个粒子的权值乘以对应的衍生粒子数 M^i ,然后对处理后的粒子集权值重新归一化,得到新粒子集 $\{x_k^i, w_k^i; i=1, 2, \dots, N\}$,为下一时刻的迭代做准备. 最后,计算有效粒子数,如果满足重采样条件,对粒子集进行重采样. 其实在权值恢复过程中,由于保留的都是高权值粒子,所以权值的方差减小,权值退化减弱. 同时,使用高斯滤波器生成了更好的重要性函数,从中可以抽取大量粒子,所以粒子集的多样性增强,避免了粒子枯竭问题. 综上,整个算法可具体归纳为“预测-衍生-估计-权值恢复”几个步骤,其进一步的阐释如图 1 所示.

对使用高斯滤波器更新得到的粒子进行分析,其综合了转移先验、似然及最近的观测数据,而从高斯型重要性函数抽样的粒子也综合了转移先验、似然及最近的观测数据这些信息. 经过简单的分析可

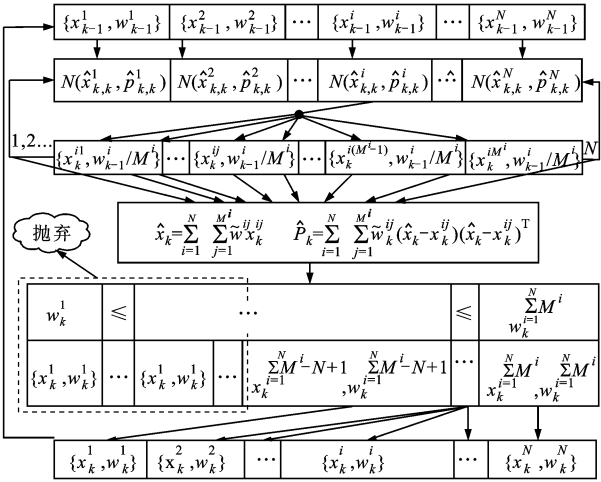


图1 粒子衍生过程图

知,这两种支持粒子集对粒子滤波器来讲本质上是一样的,因为它们综合了相同的系统信息.

1.2 算法流程

综上所述,基本的高斯衍射粒子滤波算法总体流程如图2所示.其具体步骤可归纳如下.

步骤1 衍生重要性抽样.在 $k-1$ 时刻,利用最新的系统观测对粒子集 $\{x_{k-1}^i, 1/N; i=1, 2, \dots, N\}$ 中的每个粒子进行高斯滤波,得到每个粒子的均值和方差的估计 $\hat{x}_{k,k}^i$ 和 $\hat{p}_{k,k}^i$,从而得到以 $\hat{x}_{k,k}^i$ 为均值、 $\hat{p}_{k,k}^i$ 为方差的高斯型重要性密度函数,即

$$q(x_k/x_{0:k-1}, y_{1:k}) = N(x_k^i; \hat{x}_k^i, \hat{P}_k^i) \quad (1)$$

从式(1)定义的重要性密度函数中产生预测样本,即从上述得到的一簇高斯分布中的每个高斯分布中抽取粒子 $\{x_k^i, i=1, 2, \dots, M^i\} \sim N(x_k^i; \hat{x}_k^i, \hat{P}_k^i)$,并对 $k-1$ 时刻的每个粒子的权值 $\{w_{k-1}^i, i=1, 2, \dots, N\}$ 进行分裂 $\{w_{k-1}^i/M^i, i=1, 2, \dots, N\} = \{w_{k-1}^{ij}, i=1, 2, \dots, N, j=1, 2, \dots, M^i\}$,这样得到预测样本集 $\{x_k^{ij}, (w_{k-1}^i/M^i); i=1, 2, \dots, N, j=1, 2, \dots, M^i\} \sim N(x_k^i; \hat{x}_k^i, \hat{P}_k^i)$.由于充分利用了最新的系统观测信息,因而新样本集可以改进粒子抽样的精度.

步骤2 计算权值

$$w_k^{ij} = w_{k-1}^{ij} \frac{p(y_k/x_k^{ij})p(x_k^{ij}/x_{k-1}^{ij})}{q(x_k^i/x_{k-1}^i, y_k)} \quad (2)$$

式中: $p(y_k/x_k^{ij})$ 是 y_k 的似然函数.

步骤3 归一化权值

$$\tilde{w}_k^{ij} = w_k^{ij} / \sum_{i=1}^N \sum_{j=1}^{M^i} w_k^{ij} \quad (3)$$

步骤4 输出结果.

步骤5 权值恢复.挑出优选的 N' 个粒子所对应的

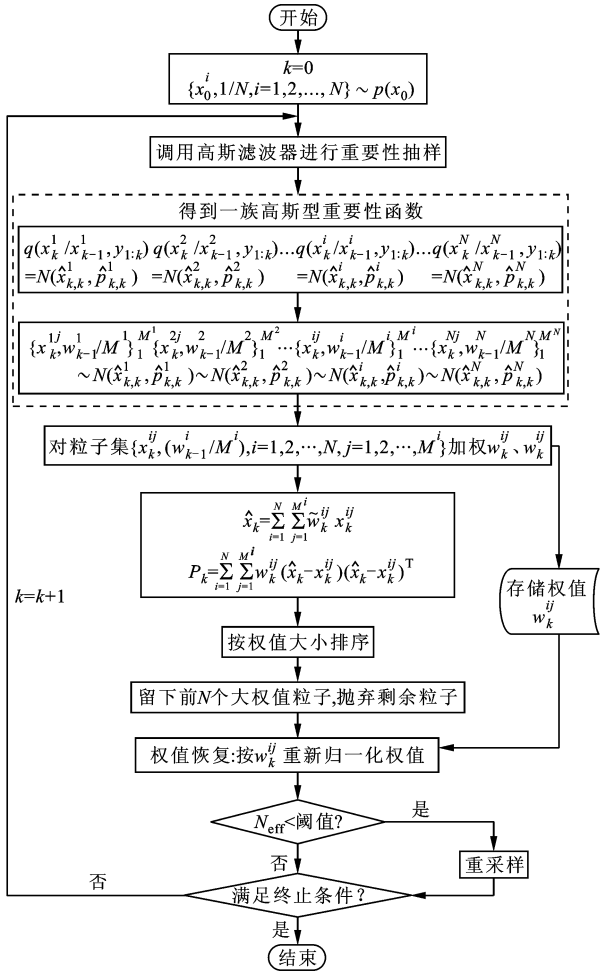


图2 高斯衍生粒子滤波算法流程图

衍生前由步骤3计算出的粒子权值,分别乘以 M^i 后,重新归一化

$$\tilde{w}_k^i = w_k^i / \sum_{i=1}^N w_k^i \quad (4)$$

步骤6 令 $k=k+1$,从步骤2重复上述过程.

1.3 衍生粒子有效性分析

由上述分析知,GAPF与GDPF的抽样方法是不同的.GAPF的所有粒子都参与了粒子更新并生成一簇重要性密度函数,然后从这一簇重要性密度函数中再抽取参与状态估计的粒子;GDPF的新采样粒子参与了粒子更新并生成一簇重要性密度函数,然后从这一簇重要性密度函数中抽取比新采样粒子数更多的粒子参与状态估计.二者的不同在于GAPF是从高斯型重要性密度函数中仅抽取一个粒子,而GDPF从高斯型重要性密度函数中抽取的粒子不止一个.2种抽样方法的直观比较如图3所示.

增加粒子数可以提高粒子滤波的性能,从这个角度可以对本文所提算法的有效性进行分析,即需

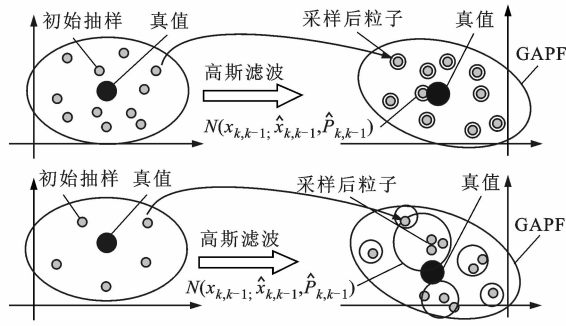


图 3 2 种抽样方法的比较

要证明 2 种方法产生的粒子是等效的,也就是证明这 2 种抽样方法没有本质上的不同。

定理 1 当 GAPF 和 GDPF 选取的更新粒子数相同时,得到的抽样粒子集等效。

证明 把 2 种方式的更新粒子数都选为 N ,经相同的高斯滤波更新后,分别得到 N 个高斯重要性函数 $\{N(x_{k,k-1}; \hat{x}_{k,k-1}^i, \hat{P}_{k,k-1}^i), i = 1, 2, \dots, N\}$ 。这时,这两簇高斯型重要性函数是一样的,不一样的只是从这两簇高斯型重要性函数中抽取的粒子数。GAPF 算法从每个重要性函数中只抽取一个粒子,得到粒子集 $\{x_k^i, w_{k-1}^i, i = 1, 2, \dots, N\} \sim N(x_k^i; \hat{x}_k^i, \hat{P}_k^i)$, GDPF 算法则根据每个粒子的权值抽取 M^i 个粒子,得到粒子集 $\{x_k^{ij}, (w_{k-1}^i/M^i), i = 1, 2, \dots, N, j = 1, 2, \dots, M^i\} \sim N(x_k^i; \hat{x}_k^i, \hat{P}_k^i)$,对 GAPF 生成的粒子集求加权均值

$$\bar{x}_k = (\sum_{i=1}^N \tilde{w}_{k-1}^i x_k^i) / N \quad (5)$$

对 GDPF 生成的粒子集求加权均值

$$\bar{x}'_k = (\sum_{i=1}^N \sum_{j=1}^{M^i} \tilde{w}_{k-1}^i x_k^{ij}) / \sum_{i=1}^N M^i \quad (6)$$

对式(6)作进一步处理

$$\begin{aligned} \bar{x}'_k &= (\sum_{i=1}^N \sum_{j=1}^{M^i} \tilde{w}_{k-1}^i x_k^{ij}) / \sum_{i=1}^N M^i = \\ &= (\sum_{j=1}^{M^1} \tilde{w}_{k-1}^{1j} x_k^{1j} + \sum_{j=1}^{M^2} \tilde{w}_{k-1}^{2j} x_k^{2j} + \dots + \sum_{j=1}^{M^N} \tilde{w}_{k-1}^{Nj} x_k^{Nj}) / \\ &= (\sum_{i=1}^N \sum_{j=1}^{M^i} \tilde{w}_{k-1}^i x_k^{ij}) / \sum_{i=1}^N M^i = \\ &= (\tilde{w}_{k-1}^1 x_k^1 + \tilde{w}_{k-1}^2 x_k^2 + \dots + \tilde{w}_{k-1}^N x_k^N) / N = \bar{x}_k \end{aligned} \quad (7)$$

即 2 种方法得到的粒子集均值是相等的。

同理可证它们的方差是相等的,即 $\bar{P}_k = \bar{P}'_k$ 。

综合可得,2 个粒子集中的每个粒子服从同一个正态分布 $N(x_k^i; \bar{x}_k^i, \bar{P}_k^i)$,2 种方法抽取的粒子没有本质上的区别,它们是等效的。证毕。

定理 2 当 GAPF 和 GDPF 选取的更新粒子数不同时,得到的抽样粒子集等效。

证明 为了对比,可以选择相同的粒子数参与状态估计,即 $N = \sum_{i=1}^{N'} M^i$,其中 N 为 GAPF 选取的更新粒子数, N' 为 GDPF 选取的更新粒子数。由图 3 所示的抽样过程可见,2 种方法得到的参与状态估计的粒子都融入了状态转移 $p(x_k/x_{k-1})$ 和最新观测信息,从这方面说它们并没有不同,都比直接从状态转移 $p(x_k/x_{k-1})$ 抽取的粒子更接近直接从真实后验分布抽取的粒子,不同的是部分粒子来源的重要性函数不同。这可以解释为粒子更新前的新采样抽取了不同的粒子,那么经高斯滤波更新后得到的均值和方差一般也不同,但差别不会太大。这种解释其实是合理的,因为在抽取粒子时是随机的。这时就可以把 GDPF 参与状态估计的衍生粒子等效为与 GAPF 在新采样时刻抽取了不同的样本,也就是说它们都是经过了高斯滤波器的更新,然后从生成的高斯重要性函数中抽取的,即这部分粒子来自新采样时刻抽取了不同的样本所得到的不同的重要性函数,因此它们不会影响滤波结果。

当然,选择不同的粒子数参与状态估计,即 $N \neq \sum_{i=1}^{N'} M^i$,上述结论也成立。

综上所述,2 种抽样方法抽取的参与状态估计的粒子是等效的。证毕。

1.4 算法有效性分析

由于 GDPF 使用高斯滤波生成的重要性密度函数融入了最新系统观测,提高了其对系统状态后验概率的逼近程度,所以该算法在一定程度上缓解了粒子退化问题。同时,GDPF 不减少参与状态估计的粒子数,且每个粒子都融入了最新观测信息,因此在选取相同粒子数参与状态估计的情况下,GDPF 获得了滤波精度和算法实时性的双重提高。在一定运行速度的要求下,GDPF 可以抽取更多的粒子参与状态估计,因此可以进一步提高滤波精度。此外,GDPF 可以根据粒子权值的大小控制从每个重要性函数中抽取的粒子数,也就意味着它可以有效减少无效粒子,增加有效粒子,从而更好地抑制了粒子滤波退化和枯竭问题。

下面再从产生一个粒子所需要的时间角度来分析 GDPF 的有效性。对 GAPF 来说,产生一个粒子

需要迭代一次高斯滤波方程,然后再从生成的重要性函数中抽取粒子集.高斯滤波方程中不仅涉及矩阵乘法,甚至还涉及矩阵求逆和计算雅克比矩阵等,而 GDPF 产生一个粒子仅需从重要性函数中直接抽取.当然,GDPF 中参与生成重要性函数的那部分粒子所用的时间并没有缩短,与 GAPF 相同.

2 高斯滤波的一种快速采样算法

为了进一步减小采样过程对算法实时性的影响,提出了一种快速的高斯采样算法,其基本思想是对初始粒子集进行线性变换完成高斯采样.具体做法如下.

初始粒子集为 $\{a_i, i=1,2,\cdots,N\}$,其中每个粒子均为从标准高斯分布中相互独立地采样得到.对每个粒子进行线性变换得

$$b_i = \alpha a_i + \mu, \quad 0 \leq i \leq N \tag{8}$$

则线性变换后得到的粒子集为 $\{\sigma_i + \mu, i=1,2,\cdots,N\}$,至此完成采样过程.

下面对该算法的有效性进行分析.从高斯分布中抽取 N 个粒子,线性变换需 N 次乘法和加法,而以往的高斯分布采样法,例如目前较常用且快速的 Ziggurat 算法,需 $4N$ 次乘法和加法.可见,生成同样数量的粒子集,线性变换法的时间复杂度为 $O(N)$,而高斯分布采样法的时间复杂度至少为 $O(4N)$,线性变换法所需时间相对较少,因此用线性变换代替高斯采样可以提高采样速度.

3 仿真实验与结果分析

单变量非平稳增长模型(UNGM)的状态方程和观测方程分别为^[7]

$$\left. \begin{aligned} x_k &= \alpha x_{k-1} + \beta \frac{x_{k-1}}{1+x_{k-1}^2} + \gamma \cos(1.2k) + v_k \\ y_k &= \frac{x_k^2}{20} + w_k, \quad k=1,2,\cdots,M \end{aligned} \right\} \tag{9}$$

式中: $v_k \sim N(v_k; 0, \sigma_v^2), \sigma_v^2=1; w_k \sim N(w_k; 0, \sigma_w^2), \sigma_w^2=1; \alpha=0.5; \beta=25; \gamma=8; p(x_0) \sim N(0,1), x_0=0.1$.

一般用均方根误差 RMSE(设为 R)来度量粒子滤波算法的误差,单次运行的 R 按下式计算

$$R = \left[\frac{1}{N'} \sum_{k=1}^{N'} (x_k^* - x_k)^2 \right]^{1/2} \tag{10}$$

式中: N' 为迭代步数; x_k 为第 k 步的状态真实值; x_k^* 为第 k 步的状态估计值.

为了比较各算法的滤波精度,图 4 给出了采用 EKF、EPF、EDPF、DDPF 等方法进行 50 次独立实验后,各种非线性滤波器状态估计的 R 随仿真次数变化的曲线.其中,DDPF 为二阶中心差分粒子滤波改进算法^[9-10],DDDPF 为使用 DD 衍生粒子的二阶中心差分衍生粒子滤波改进算法,EPF、DDPF 的粒子数和 EDPF、DDDPF 的初始粒子数均为 100 个,每个粒子衍生 2 个粒子,即从生成的每个重要性密度函数中抽取 2 个粒子,这时 EDPF 参与状态估计的粒子数为 200 个.

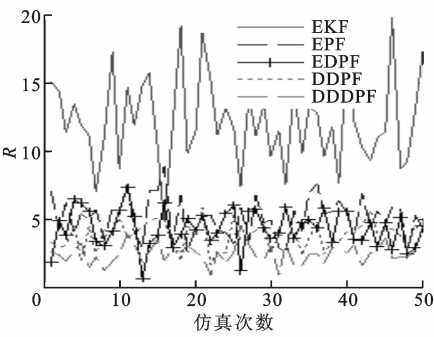


图 4 各种非线性滤波算法的均为根误差

表 1 给出了采用 EKF、EPF、EDPF 独立运行 10 次状态估计的 R .其中,EPF 的粒子数取 100 个,EDPF 的初始粒子数取 50 个,每个粒子衍生 2 个粒子,即从生成的每个重要性密度函数中抽取 2 个粒子,这时 EDPF 参与状态估计的粒子数为 100 个.

表 1 3 种非线性滤波算法 R 的比较

运行序号	R		
	EKF	EPF	EDPF
1	17.56	4.376	5.150
2	19.25	4.127	4.620
3	16.21	5.247	4.981
4	15.82	4.210	4.260
5	16.80	3.841	4.440
6	16.85	4.140	5.020
7	17.36	5.854	4.810
8	18.01	4.587	5.330
9	17.65	3.750	4.261
10	16.99	4.215	4.460

由表 1 可见:采用 EKF 时 R 的均值为 17.25、方差为 0.966 2;采用 EPF 时 R 的均值为 4.434 7、方差为 0.65;采用 EDPF 时 R 的均值为 4.733 2、方差为 0.38.

通过分析仿真结果,可以得到以下结论.

(1)EDPF 和 DDDPF 的 R 较小,对状态的估计精度要明显高于 EPF 和 DDPF,这是因为这 2 种改

进算法参与状态估计的粒子数明显多于 EPF 和 DDPF.

(2)当高斯衍生粒子滤波算法参与状态估计的粒子数与高斯辅助粒子滤波算法相等,而更新粒子数少于高斯辅助粒子滤波算法时,这2种非线性滤波算法的滤波精度不分伯仲,说明就滤波精度而言,高斯辅助粒子滤波与高斯衍生粒子滤波是相当的.

4 结 论

针对基于高斯滤波重要性采样过程的粒子滤波改进算法在解决粒子退化的同时带来了运算量增加的问题,提出了一种类似光子衍射的粒子衍生过程,并将该过程与 GAPF 结合,形成了一种新的粒子滤波算法.该算法在保证参与状态估计粒子数不变的同时,减少了高斯滤波需要更新的粒子数,从而在保证一定精度的前提下达到了减少算法运算量的目的,使已有的高斯辅助粒子滤波更具实用性.同时,该算法根据粒子权值大小自适应地调整衍生粒子数,更增加了有效粒子数.这样,所提算法在减少运算量的同时,更加有效地抑制了粒子退化,提高了滤波估计精度.

参考文献:

- [1] 武元新. 对偶四元数导航算法与非线性高斯滤波研究 [D]. 长沙: 国防科技大学机电工程与自动化学院, 2005.
- [2] DE FREITAS J F G. Sequential Monte Carlo methods to train neural network models [J]. Neural Computation, 2000, 12(4): 955-993.
- [3] VAN DER MERWE R, DE FREITAS N, DOUCET A, et al. The unscented particle filter [R]. Cambridge, UK: Cambridge Univ., 2000.
- [4] 袁泽剑, 郑南宁, 贾新春. 高斯-厄米特粒子滤波器 [J]. 电子学报, 2003, 31(7): 970-973.
YUAN Zejian, ZHENG Nanning, JIA Xinchun. The Gauss-Hermite Particle Filter [J]. Acta Electronica Sinica, 2003, 31(7): 970-973.
- [5] 李良群, 姬红兵, 罗军辉. 迭代扩展卡尔曼粒子滤波器 [J]. 西安电子科技大学学报, 2007, 34(2): 233-238.
LI Liangqun, JI Hongbing, LUO Junhui. Iterated extended Kalman particle filtering [J]. Journal of Xidian University, 2007, 34(2): 233-238.
- [6] 巫春玲, 韩崇昭. 求积分卡尔曼粒子滤波算法 [J]. 西安交通大学学报, 2009, 43(2): 25-28.
WU Chunling, HAN Chongzhao. Quadrature Kalman particle filter [J]. Journal of Xi'an Jiaotong University, 2009, 43(2): 25-28.
- [7] 胡昌华, 张琪, 乔玉坤. 强跟踪粒子滤波算法及其在故障预报中的应用 [J]. 自动化学报, 2008, 34(12): 1522-1528.
HU Changhua, ZHANG Qi, QIAO Yukun. A strong tracking particle filter with application to fault prediction [J]. Acta Electronica Sinica, 2008, 34(12): 1522-1528.
- [8] 李甫, 齐飞, 石光明, 等. 一种补偿的扩展 KALMAN 粒子滤波 [J]. 系统仿真学报, 2009, 21(15): 4752-4758.
LI Fu, QI Fei, SHI Guangming, et al. Compensated extended Kalman particle filter [J]. Journal of System Simulation, 2009, 43(2): 25-28.
- [9] 石勇, 韩崇昭. 二阶中心差分粒子滤波算法 [J]. 西安交通大学学报, 2008, 42(4): 409-413.
SHI Yong, HAN Chongzhao. Particle filter using second-order central difference [J]. Journal of Xi'an Jiaotong University, 2008, 42(4): 409-413.
- [10] 熊剑, 刘建业, 赖际舟, 等. 基于二阶插值滤波的粒子滤波改进算法研究 [J]. 控制与决策, 2009, 24(6): 907-910.
XIONG Jian, LIU Jianye, LAI Jizhou, et al. Improved particle filtering algorithm based on 2-order interpolation filtering [J]. Control and Decision, 2009, 24(6): 907-910.

(编辑 武红江 苗凌)