

二阶中心差分粒子滤波算法

石勇, 韩崇昭

(西安交通大学电子与信息工程学院, 710049, 西安)

摘要: 为了改善非线性系统状态估计问题中粒子滤波算法的估计精度, 提出采用二阶中心差分滤波方法来产生建议分布函数的新算法. 新算法对非线性系统方程作中心差分的二阶 Stirling 插值公式进行展开, 不需要计算雅可比矩阵, 易于实现, 并且采用 Cholesky 分解技术保证了协方差的正定性, 在一定程度上减小了局部线性化近似的截断误差, 并且在系统状态转移概率的基础上融合了最新的量测数据, 提高了建议分布对系统状态后验概率的逼近程度. 仿真实验表明, 与无迹粒子滤波算法相比, 新算法的计算量更小, 估计精度提高了 20% 以上.

关键词: 状态估计; 非线性系统; 粒子滤波; 中心差分

中图分类号: TP391 **文献标识码:** A **文章编号:** 0253-987X(2008)04-0409-05

Particle Filter Using Second-Order Central Difference

SHI Yong, HAN Chongzhao

(School of Electronics and Information Engineering, Xi'an Jiaotong University, Xi'an 710049, China)

Abstract: To improve the filtering precision when dealing with the state estimation problem of nonlinear systems, a new particle filter is proposed. The particle filter uses second-order central difference filter to generate the proposal distribution. The second-order central difference based on Stirling's interpolation formula is used to generate approximations to nonlinear dynamics, which avoids the evaluation of the Jacobian derivative matrix and is easy to implement. Cholesky factorization is employed to ensure the positive definiteness of the covariance matrix. The truncated errors of the local linearization are reduced in certain extent, and the latest measurements are integrated into the system state transition density so that the approximation to the system posterior density is improved. Simulation results indicate that the state estimation accuracy of new particle filter is improved more than 20% and the calculation cost is decreased, compared with the unscented particle filter.

Keywords: state estimation; nonlinear system; particle filter; central difference

粒子滤波(PF)通过非参数化的蒙特卡罗方法实现递推贝叶斯估计^[1], 由于不受非线性非高斯系统的限制, 目前被广泛应用于非线性、非高斯环境下的参数估计和状态估计^[2-3]. 贝叶斯估计要求从后验概率中抽取样本, 通常这样作是很困难的, 因此粒子滤波借助于一个容易采样的重要性密度函数(建议分布函数)来得到后验概率分布.

为了提高滤波的稳定性和精度, 必须要选择好

重要性密度函数. 近年来关于粒子滤波重要性函数选择方面的研究已经取得了很大进展, 文献[4]提出用扩展卡尔曼滤波(EKF)来产生重要性密度函数, 虽然基于EKF的粒子滤波算法估计性能有所改善, 但由于EKF采用泰勒展开近似, 又需要计算雅可比矩阵, 误差较大, 其改进效果不明显. 文献[5]提出用Unscented滤波来获得重要性密度函数的新的粒子滤波器(UPF), 其估计性能得到明显提高, 不过计

算量太大,实时性差。

本文采用二阶中心差分滤波方法来产生建议分布函数,提出一种二阶中心差分粒子滤波算法(DDPF),不需要计算微分,具有更好的数值特性。理论分析和仿真实验表明,本文算法的估计性能要优于现有其他粒子滤波算法。

1 粒子滤波基本算法

1.1 贝叶斯估计

考虑一般的非线性系统,状态方程和观测方程可以表示为

$$\mathbf{x}_k = f(\mathbf{x}_{k-1}, \mathbf{v}_{k-1}) \quad (1)$$

$$\mathbf{y}_k = h(\mathbf{x}_k, \mathbf{w}_k) \quad (2)$$

式中: $\mathbf{x}_k \in \mathbf{R}$ 为系统在 k 时刻的状态; $\mathbf{y}_k \in \mathbf{R}$ 为系统状态 $\mathbf{x}_k$ 的观测; $\mathbf{v}_{k-1}$ 为过程噪声; $\mathbf{w}_k$ 为观测噪声; $f(\cdot)$ 和 $h(\cdot)$ 为非线性函数。

我们的目的是在观测数据的基础上能够估计出系统状态的后验概率分布函数 $p(\mathbf{x}_{0:k} | \mathbf{y}_{1:k})$, 特别是边际概率分布函数 $p(\mathbf{x}_k | \mathbf{y}_{1:k})$, 即滤波概率分布函数, $\mathbf{x}_{1:k} := \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k\}$ 和 $\mathbf{y}_{1:k} := \{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_k\}$ 分别为到 k 时刻为止系统的累积状态和观测。如果状态的初始概率密度已知, 则状态预测方程为

$$p(\mathbf{x}_k | \mathbf{y}_{1:k-1}) = \int p(\mathbf{x}_k | \mathbf{x}_{k-1}) p(\mathbf{x}_{k-1} | \mathbf{y}_{1:k-1}) d\mathbf{x}_{k-1} \quad (3)$$

状态更新方程为

$$p(\mathbf{x}_k | \mathbf{y}_{1:k}) = \frac{p(\mathbf{y}_k | \mathbf{x}_k) p(\mathbf{x}_k | \mathbf{y}_{1:k-1})}{p(\mathbf{y}_k | \mathbf{y}_{1:k-1})} \quad (4)$$

式(3)、式(4)描述了最优贝叶斯估计的基本思想,但其解析解只对某些模型成立。

1.2 粒子滤波

粒子滤波是基于贝叶斯采样估计原理的序贯蒙特卡罗模拟方法,其核心思想是用一些随机样本的加权和表示后验概率,由此得到状态的估计值。当样本点很大时,粒子滤波估计接近于最优贝叶斯估计。

假设能够独立从滤波概率分布函数 $p(\mathbf{x}_k | \mathbf{y}_{1:k})$ 中抽取 N 个样本 $\{\mathbf{x}_k^i, i=1, 2, \dots, N\}$, 则状态概率分布函数可以用以下经验概率分布函数近似表述

$$p(\mathbf{x}_{0:k} | \mathbf{y}_{1:k}) \approx \frac{1}{N} \sum_{i=1}^N \delta(\mathbf{x}_{0:k} - \mathbf{x}_{0:k}^i) \quad (5)$$

通常不可能直接从状态的概率密度函数中采样,常用手法是从一个容易采样的重要性密度函数 $q(\mathbf{x}_{0:k} | \mathbf{y}_{1:k})$ 中独立抽取样本(重要性采样^[3]), 每个粒子的归一化权值取为 $\omega_k^i \propto \frac{p(\mathbf{x}_{0:k}^i | \mathbf{y}_{1:k})}{q(\mathbf{x}_{0:k}^i | \mathbf{y}_{1:k})}$, 根据前面

的贝叶斯估计原理,可以得到

$$\begin{aligned} p(\mathbf{x}_{0:k} | \mathbf{y}_{1:k}) &= \frac{p(\mathbf{y}_k | \mathbf{x}_k) p(\mathbf{x}_k | \mathbf{x}_{k-1})}{p(\mathbf{y}_k | \mathbf{y}_{1:k-1})} p(\mathbf{x}_{0:k-1} | \mathbf{y}_{1:k-1}) \propto \\ & p(\mathbf{y}_k | \mathbf{x}_k) p(\mathbf{x}_k | \mathbf{x}_{k-1}) p(\mathbf{x}_{0:k-1} | \mathbf{y}_{1:k-1}) \quad (6) \end{aligned}$$

对重要性密度函数,可以分解得到

$$q(\mathbf{x}_{0:k} | \mathbf{y}_{1:k}) = q(\mathbf{x}_k | \mathbf{x}_{0:k-1}, \mathbf{y}_{1:k}) q(\mathbf{x}_{0:k-1} | \mathbf{y}_{1:k-1}) \quad (7)$$

由前面几式,可以得到权值更新公式

$$\omega_k^i \propto \omega_{k-1}^i \frac{p(\mathbf{y}_k | \mathbf{x}_k^i) p(\mathbf{x}_k^i | \mathbf{x}_{k-1}^i)}{q(\mathbf{x}_k^i | \mathbf{x}_{k-1}^i, \mathbf{y}_k)} \quad (8)$$

可见粒子滤波是利用重要密度获取支撑点集,并随着测量值的依次到来迭代求得相应的权值,最终以加权和表征后验概率密度,得到状态的估计值。

2 二阶中心差分粒子滤波算法

重要性密度函数的选择是基于重要性抽样的粒子滤波器的关键技术之一。重要性权值的方差达到极小时的重要性密度函数为^[4]

$$q(\mathbf{x}_k | \mathbf{x}_{0:k-1}, \mathbf{y}_{1:k}) = p(\mathbf{x}_k | \mathbf{x}_{0:k-1}, \mathbf{y}_{1:k}) \quad (9)$$

在很多情况下,以 $p(\mathbf{x}_k | \mathbf{x}_{0:k-1}, \mathbf{y}_{1:k})$ 作为重要性密度函数还是很困难的,存在两个缺点:一是需要对其直接抽样,这是难以实现的;二是需要对新状态量进行积分计算,而且通常不存在解析解。实际应用中经常选择转移概率密度函数 $p(\mathbf{x}_k | \mathbf{x}_{k-1})$ 作为重要性密度函数,由于没有利用最新的系统观测,因此产生的样本与真实的后验概率产生的样本偏差较大,尤其当似然函数与转移概率密度对比过于尖锐时,或者似然函数位于系统状态转移概率密度函数的尾部,或者系统观测具有很高的精度时,偏差就很明显,容易导致滤波发散。为了提高粒子滤波的稳定性和滤波精度,首先要选择好的重要性函数。

2.1 二阶 Stirling 插值展开式

中心差分滤波器是利用 Stirling 插值公式将非线性模型按中心差分形式展开^[6-7],不需要计算函数的偏导数,甚至非线性函数不连续且存在奇异点也能进行状态估计。

设 $\mathbf{x} \in \mathbf{R}^n$ 为 n 维矢量, $\mathbf{y} = f(\mathbf{x})$, 则在 $\mathbf{x} = \bar{\mathbf{x}}$ 处用 Stirling 插值公式展开

$$\mathbf{y} \approx f(\bar{\mathbf{x}}) + \tilde{D}_{\Delta r} f(\bar{\mathbf{x}}) + \frac{1}{2!} \tilde{D}_{\Delta r}^2 f(\bar{\mathbf{x}}) \quad (10)$$

式中: $\tilde{D}_{\Delta r}$ 和 $\tilde{D}_{\Delta r}^2$ 分别为一阶和二阶差分算子。

$$\tilde{D}_{\Delta r} f(\bar{\mathbf{x}}) = \frac{1}{\lambda} \left(\sum_{p=1}^n \Delta \mathbf{x}_p \mu_p \delta_p \right) f(\bar{\mathbf{x}}) \quad (11)$$

$$\tilde{D}_{\Delta x}^2 f = \frac{1}{\lambda^2} \left(\sum_{p=1}^n \Delta x_p^2 \delta_p^2 + \sum_{p=1}^n \sum_{\substack{q=1 \\ q \neq p}}^n \Delta x_p \Delta x_q (\mu_p \delta_p) (\mu_q \delta_q) \right) f(\bar{\mathbf{x}}) \quad (12)$$

式中: δ_p 是偏微分算子; μ_p 为均值算子; λ 为给定的变量步长, 其最优选择是 $\lambda^2=3$.

2.2 二阶中心差分非线性滤波算法

二阶中心差分滤波器(DDF)不需要计算函数的偏导数, 可应用于任意函数, 滤波方程中采用了协方差矩阵的平方根形式, 保证了协方差矩阵在传播过程的半正定性, 具有更好的数值特性. 算法实现的复杂度和计算量都很小, 滤波精度高于 Unscented 滤波^[8].

对于一般的非线性系统, 系统状态方程和观测方程为式(1)、式(2), 且 $\mathbf{v}_k \sim N(\bar{\mathbf{v}}_k, \mathbf{Q})$, $\mathbf{w}_k \sim N(\bar{\mathbf{w}}_k, \mathbf{R})$. 采用 Cholesky 分解法引入 4 个平方根分解算子

$$\begin{aligned} \mathbf{Q} &= \mathbf{S}_v \mathbf{S}_v^T; \mathbf{R} = \mathbf{S}_w \mathbf{S}_w^T \\ \bar{\mathbf{P}} &= \bar{\mathbf{S}}_x \bar{\mathbf{S}}_x^T; \hat{\mathbf{P}} = \hat{\mathbf{S}}_x \hat{\mathbf{S}}_x^T \end{aligned} \quad (13)$$

过程噪声协方差矩阵 $\mathbf{Q}$ 和观测噪声协方差矩阵 $\mathbf{R}$ 的因子 $\mathbf{S}_v$ 与 $\mathbf{S}_w$ 可以预先得到, 预测协方差 $\bar{\mathbf{P}}$ 和估计协方差 $\hat{\mathbf{P}}$ 的因子 $\bar{\mathbf{S}}_x$ 和 $\hat{\mathbf{S}}_x$ 在滤波过程中不断更新修正.

基于插值近似公式, 定义以下一阶均差矩阵

$$\mathbf{S}_{xx}^{(1)}(k-1) = \{ (f_i(\hat{\mathbf{x}}_{k-1} + \lambda \hat{\mathbf{s}}_{x,j}, \bar{\mathbf{v}}_{k-1}) - f_i(\hat{\mathbf{x}}_{k-1} - \lambda \hat{\mathbf{s}}_{x,j}, \bar{\mathbf{v}}_{k-1})) / 2\lambda \} \quad (14)$$

$$\mathbf{S}_{xv}^{(1)}(k-1) = \{ (f_i(\hat{\mathbf{x}}_{k-1} + \lambda \hat{\mathbf{s}}_{x,j}, \bar{\mathbf{v}}_{k-1} + \lambda \hat{\mathbf{s}}_{v,j}) - f_i(\hat{\mathbf{x}}_{k-1} + \lambda \hat{\mathbf{s}}_{x,j}, \bar{\mathbf{v}}_{k-1} - \lambda \hat{\mathbf{s}}_{v,j})) / 2\lambda \} \quad (15)$$

$$\mathbf{S}_{yx}^{(1)}(k-1) = \{ (h_i(\bar{\mathbf{x}}_{k-1} + \lambda \bar{\mathbf{s}}_{x,j}, \bar{\mathbf{w}}_{k-1}) - h_i(\bar{\mathbf{x}}_{k-1} - \lambda \bar{\mathbf{s}}_{x,j}, \bar{\mathbf{w}}_{k-1})) / 2\lambda \} \quad (16)$$

$$\mathbf{S}_{yw}^{(1)}(k-1) = \{ (h_i(\bar{\mathbf{x}}_{k-1} + \lambda \bar{\mathbf{s}}_{x,j}, \bar{\mathbf{w}}_{k-1} + \lambda \hat{\mathbf{s}}_{w,j}) - h_i(\bar{\mathbf{x}}_{k-1} + \lambda \bar{\mathbf{s}}_{x,j}, \bar{\mathbf{w}}_{k-1} - \lambda \hat{\mathbf{s}}_{w,j})) / 2\lambda \} \quad (17)$$

式中: $\hat{\mathbf{s}}_{x,j}$ 、 $\bar{\mathbf{s}}_{x,j}$ 、 $\hat{\mathbf{s}}_{v,j}$ 、 $\hat{\mathbf{s}}_{w,j}$ 分别为矩阵 $\hat{\mathbf{S}}_x$ 、 $\bar{\mathbf{S}}_x$ 、 $\hat{\mathbf{S}}_v$ 、 $\hat{\mathbf{S}}_w$ 的第 j 列. $\hat{\mathbf{x}}_{k-1}$ 、 $\bar{\mathbf{x}}_{k-1}$ 分别为系统 $k-1$ 时刻系统的状态估计和预测.

一步预测

$$\begin{aligned} \bar{\mathbf{x}}_k &= \frac{\lambda^2 - n_x - n_v}{\lambda^2} f(\hat{\mathbf{x}}_{k-1}, \bar{\mathbf{v}}_{k-1}) + \\ &\frac{1}{2\lambda^2} \sum_{p=1}^{n_x} [f(\hat{\mathbf{x}}_{k-1} + \lambda \hat{\mathbf{s}}_{x,p}, \bar{\mathbf{v}}_{k-1}) + f(\hat{\mathbf{x}}_{k-1} - \lambda \hat{\mathbf{s}}_{x,p}, \bar{\mathbf{v}}_{k-1})] + \\ &\frac{1}{2\lambda^2} \sum_{p=1}^{n_v} [f(\hat{\mathbf{x}}_{k-1}, \bar{\mathbf{v}}_{k-1} + \lambda \hat{\mathbf{s}}_{v,p}) + f(\hat{\mathbf{x}}_{k-1}, \bar{\mathbf{v}}_{k-1} - \lambda \hat{\mathbf{s}}_{v,p})] \end{aligned} \quad (18)$$

式中: n_x 、 n_v 分别是状态向量和量测噪声向量的维数. 预测状态误差均方矩阵 Cholesky 因子需要通过复合矩阵的 Householder^[9] 三角化把它转换为方形 Cholesky 因子

$$\begin{aligned} \bar{\mathbf{S}}_x(k) &= \\ [\mathbf{S}_{xx}^{(1)}(k-1) \quad \mathbf{S}_{xv}^{(1)}(k-1) \quad \mathbf{S}_{xx}^{(2)}(k-1) \quad \mathbf{S}_{xv}^{(2)}(k-1)] \end{aligned} \quad (19)$$

预测误差协方差阵

$$\bar{\mathbf{P}}(k) = \bar{\mathbf{S}}_x(k) \bar{\mathbf{S}}_x(k)^T \quad (20)$$

滤波更新 计算基于插值近似公式定义的 4 个均差矩阵

$$\mathbf{S}_{xx}^{(2)}(k) = \left\{ \frac{(\lambda^2 - 1)^{1/2}}{2\lambda^2} (f_i(\hat{\mathbf{x}}_k + \lambda \hat{\mathbf{s}}_{x,j}, \bar{\mathbf{v}}_k) + f_i(\hat{\mathbf{x}}_k - \lambda \hat{\mathbf{s}}_{x,j}, \bar{\mathbf{v}}_k) - 2f_i(\hat{\mathbf{x}}_k, \bar{\mathbf{v}}_k)) \right\} \quad (21)$$

$$\mathbf{S}_{xv}^{(2)} = \left\{ \frac{(\lambda^2 - 1)^{1/2}}{2\lambda^2} (f_i(\hat{\mathbf{x}}_k, \bar{\mathbf{v}}_k + \lambda \hat{\mathbf{s}}_{v,j}) + f_i(\hat{\mathbf{x}}_k, \bar{\mathbf{v}}_k - \lambda \hat{\mathbf{s}}_{v,j}) - 2f_i(\hat{\mathbf{x}}_k, \bar{\mathbf{v}}_k)) \right\} \quad (22)$$

$$\mathbf{S}_{yx}^{(2)}(k) = \left\{ \frac{(\lambda^2 - 1)^{1/2}}{2\lambda^2} (h_i(\bar{\mathbf{x}}_k + \lambda \bar{\mathbf{s}}_{x,j}, \bar{\mathbf{w}}_k) + h_i(\bar{\mathbf{x}}_k - \lambda \bar{\mathbf{s}}_{x,j}, \bar{\mathbf{w}}_k) - 2h_i(\bar{\mathbf{x}}_k, \bar{\mathbf{w}}_k)) \right\} \quad (23)$$

$$\mathbf{S}_{yw}^{(2)}(k) = \left\{ \frac{(\lambda^2 - 1)^{1/2}}{2\lambda^2} (h_i(\bar{\mathbf{x}}_k, \bar{\mathbf{w}}_k + \lambda \hat{\mathbf{s}}_{w,j}) + h_i(\bar{\mathbf{x}}_k, \bar{\mathbf{w}}_k - \lambda \hat{\mathbf{s}}_{w,j}) - 2h_i(\bar{\mathbf{x}}_k, \bar{\mathbf{w}}_k)) \right\} \quad (24)$$

预测量测均方差矩阵的 Cholesky 因子可用下面的复合矩阵的 Householder 变换得到

$$\mathbf{S}_y(k) = [\mathbf{S}_{yx}^{(1)}(k) \quad \mathbf{S}_{yw}^{(1)}(k) \quad \mathbf{S}_{yx}^{(2)}(k) \quad \mathbf{S}_{yw}^{(2)}(k)] \quad (25)$$

预测量测方程

$$\begin{aligned} \bar{\mathbf{y}}_k &= \frac{\lambda^2 - n_x - n_w}{\lambda^2} h(\bar{\mathbf{x}}_k, \bar{\mathbf{w}}_k) + \\ &\frac{1}{2\lambda^2} \sum_{p=1}^{n_x} h(\bar{\mathbf{x}}_k - \lambda \bar{\mathbf{s}}_{x,p}, \bar{\mathbf{w}}_k) + h(\bar{\mathbf{x}}_k + \lambda \bar{\mathbf{s}}_{x,p}, \bar{\mathbf{w}}_k) + \\ &\frac{1}{2\lambda^2} \sum_{p=1}^{n_w} h(\bar{\mathbf{x}}_k, \bar{\mathbf{w}}_k + \lambda \hat{\mathbf{s}}_{w,p}) + h(\bar{\mathbf{x}}_k, \bar{\mathbf{w}}_k - \lambda \hat{\mathbf{s}}_{w,p}) \end{aligned} \quad (26)$$

式中: n_w 是量测噪声向量的维数.

预测误差交叉均方矩阵

$$\mathbf{P}_{xy}(k) = \bar{\mathbf{S}}_x(k) \mathbf{S}_{yx}^{(1)}(k)^T \quad (27)$$

Kalman 滤波增益

$$\mathbf{K}_k = \mathbf{P}_{xy}(k) [\mathbf{S}_y(k) \mathbf{S}_y(k)^T]^{-1} \quad (28)$$

滤波状态方程

$$\hat{\mathbf{x}}_k = \bar{\mathbf{x}}_k + \mathbf{K}_k (\mathbf{y}_k - \bar{\mathbf{y}}_k) \quad (29)$$

Cholesky 因子

$$\hat{\mathbf{S}}_x(k) = [\bar{\mathbf{S}}_x(k) - \mathbf{K}_k \mathbf{S}_{xy}^{(1)}(k) \quad \mathbf{K}_k \mathbf{S}_{yw}^{(1)}(k) \\ \mathbf{K}_k \mathbf{S}_{yx}^{(2)}(k) \quad \mathbf{K}_k \mathbf{S}_{yw}^{(2)}(k)] \quad (30)$$

滤波误差协方差阵

$$\hat{\mathbf{P}}(k) = \hat{\mathbf{S}}_x(k) \hat{\mathbf{S}}_x(k)^T \quad (31)$$

2.3 二阶中心差分粒子滤波算法

本文采用二阶中心差分滤波产生建议分布函数,并采取残差重抽样算法步骤等改进措施,构成了二阶中心差分粒子滤波算法,步骤如下。

(1) 初始化: 在时间 $k=0$

For $i=1, \dots, N$, 抽样粒子 $\mathbf{x}_0^{(i)} \sim p(\mathbf{x}_0)$, 则第 i 个粒子的预测为

$$\bar{\mathbf{x}}_0^{(i)} = E[\mathbf{x}_0^{(i)}] \quad (32)$$

$$\hat{\mathbf{P}}_0^{(i)} = E[(\mathbf{x}_0^{(i)} - \bar{\mathbf{x}}_0^{(i)})(\mathbf{x}_0^{(i)} - \bar{\mathbf{x}}_0^{(i)})^T] \quad (33)$$

$$\mathbf{Q} = \mathbf{S}_v(\mathbf{S}_v)^T; \quad \mathbf{R} = \mathbf{S}_w(\mathbf{S}_w)^T \quad (34)$$

(2) 在时间 $k=1, 2, \dots$,

计算每个粒子的一阶均差矩阵,在每一时刻,根据式(18)求得每个粒子的一步预测 $\bar{\mathbf{x}}_k^{(i)}$,用二阶插值滤波更新预测误差协方差

$$\bar{\mathbf{S}}_x^{(i)}(k) = [(\mathbf{S}_{xx}^{(1)}(k-1))^{(i)} \quad (\mathbf{S}_{xv}^{(1)}(k-1))^{(i)} \\ (\mathbf{S}_{vx}^{(2)}(k-1))^{(i)} \quad (\mathbf{S}_{vw}^{(2)}(k-1))^{(i)}] \quad (35)$$

$$\bar{\mathbf{P}}_k^{(i)} = \bar{\mathbf{S}}_x^{(i)}(\bar{\mathbf{S}}_x^{(i)})^T \quad (36)$$

再计算每个粒子的二阶均差矩阵,得到预测状态误差均方矩阵。

Cholesky 分解因子

$$\mathbf{S}_y^{(i)}(k) = [(\mathbf{S}_{yw}^{(1)}(k))^{(i)} \quad (\mathbf{S}_{yw}^{(1)}(k))^{(i)} \\ (\mathbf{S}_{yx}^{(2)}(k))^{(i)} \quad (\mathbf{S}_{yw}^{(2)}(k))^{(i)}] \quad (37)$$

根据式(29)更新每个粒子的状态估计 $\hat{\mathbf{x}}_k^{(i)}$,根据式(31)更新每个粒子的协方差估计 $\hat{\mathbf{P}}_k^{(i)}$ 。

(3) 抽样粒子 $\mathbf{x}_k^{(i)} \sim N(\mathbf{x}_k^{(i)}; \hat{\mathbf{x}}_k^{(i)}; \hat{\mathbf{P}}_k^{(i)})$ 。

(4) 根据式(8)计算每个粒子的权值 $w_k^{(i)}$,并且归一化 $\bar{w}_k^{(i)} = \frac{w_k^{(i)}}{\sum_j w_k^{(j)}}$ 。

(5) 残差重采样

如果 $(N_{\text{eff}} < N_{\text{th}})$, 将权值重新设置为 $\bar{w}_k^{(i)} = \frac{1}{N}$ 得到等权样本集 $\{\mathbf{x}_k^{(i)}, \bar{w}_k^{(i)}\}$ 。

输出: 经过重抽样,把原来的带权值样本集映射为等权值样本集,得到滤波分布的后验概率为

$$\hat{p}(\mathbf{x}_k | \mathbf{Y}_k) = \sum_{i=1}^N \bar{w}_k^{(i)} \delta(\mathbf{x}_k - \mathbf{x}_k^{(i)}) \quad (38)$$

$$\hat{\mathbf{x}}_k = \sum_{i=1}^N \bar{w}_k^{(i)} \mathbf{x}_k^{(i)} \quad (39)$$

$$\hat{\mathbf{P}}_k = \sum_{i=1}^N \bar{w}_k^{(i)} (\hat{\mathbf{x}}_k - \mathbf{x}_k^{(i)})(\hat{\mathbf{x}}_k - \mathbf{x}_k^{(i)})^T \quad (40)$$

3 仿真实验

3.1 仿真场景

为了验证本文改进算法的有效性,对采用转移概率分函数布作为建议分布函数的标准粒子滤波 PF、采用 EKF、UKF 作为建议分布函数的粒子滤波和改进粒子滤波算法(DDPF)进行了仿真实验,并对实验结果进行了分析比较。仿真实验的系统模型^[5]如下

$$x_k = 0.5x_{k-1} + \sin(0.04\pi k) + 0.5x_{k-1} + v_{k-1} \quad (41)$$

$$y_k = \begin{cases} 0.2x_k^2 + w_k & k \leq 30 \\ 0.5x_k - 2 + w_k & k > 30 \end{cases} \quad (42)$$

式中: v_k 是 $\zeta_a(3, 2)$ 分布的随机噪声矩阵; w_k 是协方差为 0.000 01 的高斯随机噪声。

3.2 仿真结果与分析

实验中,粒子数均为 $N=200$,在每个时间步长都采用残差重采样方法,跟踪时间 $T=60$ s,采用 100 次蒙特卡罗仿真。每次独立仿真实验的均方根误差定义为

$$E = \left(\frac{1}{T} \sum_{k=1}^T (\hat{x}_k - x_k)^2 \right)^{1/2} \quad (43)$$

在一次独立实验中,真实目标状态和不同粒子滤波器的系统状态估计如图 1 所示。图 2 为不同粒子滤波器的均方根误差。

由图 1 可以看出,标准粒子滤波由于采用转移概率作为建议分布函数,没有考虑最新量测信息,其状态估计会严重偏离真实状态,误差明显大于其他几种方法,而扩展卡尔曼粒子滤波采用 EKF 产生建议分布函数,EKF 在线性化和高斯假设中引入太多误差,估计性能一般。UPF 和本文提出的方法的状态估计比较接近于真实状态,本文给出的粒子滤波器状态估计效果最为理想。

粒子滤波算法几种算法单次蒙特卡罗仿真的运算时间分别为: PF, 15.357 s; PF-EKF, 31.206 s; PF-UKF, 52.309 s; PF-DDF, 37.193 s。

由图 1 可以看出,由于 DDPF 滤波器不需要计算微分,因此实现相对简单,运算时间要小于 UPF。从图 2 和表 1 可以看出,UPF 和 DDPF 的估计精度要比 PF-EKF 和 PF 好得多,与 UPF 相比,DDPF 的估计精度提高了 20% 以上。DDPF 滤波公式没有直接采用协方差矩阵,这样保证了滤波方差阵的正

定性,克服了由于舍入误差而引起的发散现象,另外也可以减少计算误差.由仿真结果可知,DDPF 的收敛性和稳定性最好,均方根误差也要小于其他几种算法.综合考虑,采用二阶中心差分滤波产生建议分布函数的粒子滤波器滤波估计性能优于 EKF-PF、UPF 和 PF.

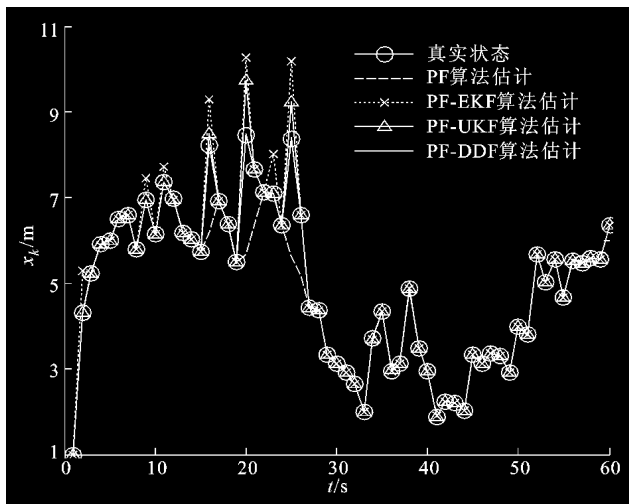


图1 不同粒子滤波器的状态估计

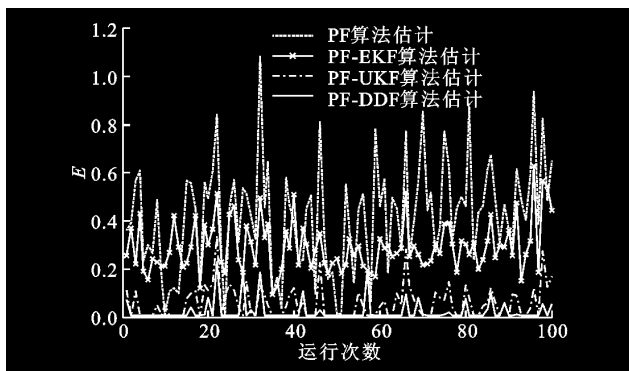


图2 不同粒子滤波器的均方根误差

4 结束语

本文采用二阶中心差分滤波产生建议分布函数,提出了一种新的粒子滤波算法 DDPF,并在理论上分析了改进算法相对于原来算法的优越性.仿真实验和统计数据表明,对于非线性、非高斯系统的状

态估计问题,二阶中心插值粒子滤波算法的滤波效果要好于其他几种粒子滤波算法,表现出更高的精度和更强的抗噪声性能.

参考文献:

- [1] GORDON N J, SALMOND D J, SMITH, A F M. Novel approach to nonlinear/non-Gaussian Bayesian state estimation [J]. IEE Proceedings-Radar, Sonar and Navigation, 1993, 140(2): 107-113.
- [2] AMLALAM S, MASKELL S, GORDON N J. A tutorial on particle filters for online non-linear/non-Gaussian bayesian tracking [J]. IEEE Transactions on Signal Processing, 2002, 50(2):174-188.
- [3] KOTECHA, J H, DJURIC, P M. Gaussian particle filtering [J]. IEEE Transactions on Signal Processing, 2003, 51(10):2592-2601.
- [4] DOUCET A, GODSIL S, ANDRIEU C. On sequential Monte Carlo sampling methods for Bayesian filtering [J]. Statistical Computing, 2000, 10(3):192-208.
- [5] van der MERWE R, DOUCET A, de FREITAS N, et al. The unscented particle filter [R]. Technical Report of the Cambridge University Engineering Department CUED/F-INFENG/TR, 380. Cambridge, UK: Cambridge University Press, 2000:1-45.
- [6] SCHEI, T S. A finite-difference method for linearization in nonlinear estimation algorithms [J]. Automatica, 2003, 51(10):2592-2601.
- [7] STEFFENSEN J F. Interpolation [M]. Baltimore, USA: Williams & Wilkins, 1927.
- [8] SAULSON B G, CHANG K C. Nonlinear estimation comparison for ballistic missile tracking [J]. Automatica, 2004, 43(6):1424-1438.
- [9] 邓建中,刘之行.计算方法[M].西安:西安交通大学出版社, 2001.

(编辑 刘杨)