

一种快速准蒙特卡罗粒子滤波算法

赵玲玲¹ 马培军¹ 苏小红¹

摘要 针对准蒙特卡罗 (Quasi-Monte Carlo, QMC) 方法应用于粒子滤波采样时计算复杂度高, 以及粒子滤波中重采样步骤引起样本枯竭的问题, 提出一种结合准蒙特卡罗方法的粒子滤波算法, 在重要性采样后, 将生成的随机化 QMC 序列分别映射到以大权重粒子为核心的独立子空间上, 避免了直接对采样空间进行预测, 同时又保持了样本多样性. 实验结果表明该方法可以有效抑制样本枯竭现象, 获得了高于蒙特卡罗 (Monte Carlo, MC) 方法的估计精度, 而计算效率与粒子滤波相近.

关键词 粒子滤波, 准蒙特卡罗, 样本枯竭, 重采样算法

DOI 10.3724/SP.J.1004.2010.01351

A Fast Quasi-Monte Carlo-based Particle Filter Algorithm

ZHAO Ling-Ling¹ MA Pei-Jun¹ SU Xiao-Hong¹

Abstract Particle filters have a high computational complexity when using quasi-Monte Carlo (QMC) methods, and are subject to the sample impoverishment caused by resampling step. To solve these problems, a new particle filter algorithm based on QMC method is proposed. It generates the randomized QMC points after sample importance, and then transforms them into some independent sub-spaces, whose kernels are the particles with heavy weights, to avoid predicting the sampling space and preserve the diversity of samples. The simulation results suggest that the algorithm can escape successfully from the sample impoverishment, provide more accurate estimators than the Monte Carlo (MC) method, meanwhile has a computational cost similar to the general particle filter.

Key words Particle filters, quasi-Monte Carlo (QMC), sample impoverishment, resampling algorithm

粒子滤波 (Particle filter, PF)^[1] 是一种基于重要性采样重采样的递归贝叶斯滤波技术, 由于其不受系统线性和高斯噪声的限制, 被广泛应用于定位导航、目标跟踪、模式识别等领域. 针对粒子滤波的权值退化问题, Gordon 等提出了重采样算法^[2], 但是, 重采样算法仅复制大权值样本, 可能导致采样枯竭, 当系统过程噪声较小时, 易造成样本的聚集. 为解决这一问题, 有学者提出添加马尔科夫链-蒙特卡罗 (Markov chain Monte Carlo)^[3] 移动步骤, 为每个样本构造一个迭代链来对样本进行移动, 但这增加了算法的计算负担. 另外也有学者提出了正则粒子滤波^[4], 但其在多维状态空间下构造平滑核比较困难.

准蒙特卡罗方法 (Quasi-Monte Carlo, QMC)^[5-7] 是一种确定性采样方法, 与蒙特卡罗 (Monte Carlo, MC) 方法不同, 该方法生成的样本在空间中分布更为均匀, 可以获得优于 MC 方法的估计精度^[5]. QMC 方法早期应用于金融领域的数理统计分析, 近年来被引入到滤波领域^[8-11]. Guo 等

提出序贯 QMC (Sequential quasi-Monte Carlo, SQMC) 算法^[9], 通过在采样空间中生成随机化 QMC 样本来获取更高的估计精度. 但是, 其计算复杂度为 $O(N^2)$ ^[8, 10], 而一般粒子滤波算法的计算复杂度为 $O(N)$ ^[8].

针对上述问题, 本文提出了一种新的基于 QMC 采样的粒子滤波算法, 在预测阶段, 从重要性分布中生成随机样本, 权值更新后, 根据权值分布确定样本子代个数, 然后在以大权重样本为核心的多个子空间内进行 QMC 采样. 该算法不必预测采样空间, 计算复杂度和一般粒子滤波相近, 低于 SQMC 算法, 另外用子空间内低差异性序列代替了重复样本, 提高了样本多样性, 解决了采样枯竭问题.

1 粒子滤波

离散的动力系统状态空间模型可描述为

$$x_t = f_t(x_{t-1}, \omega_t) \quad (1)$$

$$z_t = g_t(x_t, \nu_t) \quad (2)$$

其中, $x_t \in \mathbf{R}^n$ 为时刻 t 的状态向量, $z_t \in \mathbf{R}^m$ 表示传感器在时刻 t 得到的测量向量, $\omega_t \in \mathbf{R}^n$ 和 $\nu_t \in \mathbf{R}^m$ 为相互独立的过程噪声和观测噪声, $f_t: \mathbf{R}^n \times \mathbf{R}^n \rightarrow \mathbf{R}^n$ 是状态转移函数, $h_t: \mathbf{R}^n \times \mathbf{R}^m \rightarrow \mathbf{R}^m$ 为传感器量测函数.

粒子滤波通过寻找在状态空间 \mathbf{R}^n 中传播的加权随机样本 $\{x_t^i, w_t^i\}_{i=1}^N$ 来近似后验概率密度函数 $p(x_t|z_{1:t})$. 在算法迭代过程中, 退化现象不可避免, 为此引入了重采样步骤. 其思想是通过对本样本集再采样 N 次, 产生新的支撑点集 $\{x_t^{i*}\}_{i=1}^N$, 使 $p(x_t^{i*} = x_t^i) = w_t^i$. 由于权值被重设为 $w_t^i = 1/N$, 使得权重均分在所有样本上.

2 SQMC 算法

对积分 $I = \int_{[0,1]^d} f(x)dx$ 的估计问题, 常用的仿真方法是 MC 方法, 该方法在积分区域上随机采样, 利用式 (3) 对积分进行估计.

$$\hat{I} = \frac{1}{N} \sum_{i=1}^N f(x_i) \quad (3)$$

其中 $x_i \in [0, 1]^d$, 服从独立的 $U[0, 1]^d$ 分布.

2.1 QMC 方法

为改善 MC 采样造成的样本聚集和空隙现象, 有学者提出 QMC 方法, 它生成具有低差异性 (Low discrepancy) 的确定序列. 差异性可以认为是单位超立方空间的某子空间中样本序列的离散均匀分布与连续均匀分布的最大差异^[6], 差异性描述了样本在空间中分布的均匀程度, 差异性越低, 分布越均匀.

低差异性序列的估计精度高于 MC 样本, 研究表明 QMC 方法估计误差边界为 $1/N$, 而 MC 方法的误差边界为 $1/N^{1/2}$ ^[6], 独立于样本空间的维数.

典型的低差异性序列有 Halton 序列、Faure 序列、Sobol 序列、LHS 序列等. 其中 Halton 序列可以通过式 (4)^[6] 和 (5)^[6] 递推获得:

$$n = \sum_{j=0}^m a_j(n)b^j \quad (4)$$

$$H_b(n) = \sum_{j=0}^m \frac{a_j(n)}{b^{j+1}} \quad (5)$$

收稿日期 2009-03-24 录用日期 2009-12-31
Manuscript received March 24, 2009; accepted December 31, 2009
国家自然科学基金 (60773067) 资助
Supported by National Natural Science Foundation of China (60773067)
1. 哈尔滨工业大学计算机科学与技术学院 哈尔滨 150001
1. School of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001

其中, b 为 Halton 序列的基数, 给定一个随机整数 n , 可以构造一个关于 b 的多项式, 该多项式的值为 n , 项数为 m , 系数为 $a_j(n)$, $a_j(n) \in \{0, 1, \dots, b-1\}$, $j = 1, \dots, m$. 这样, 利用式 (5) 就可以计算得到与 n 对应的 Halton 序列中的一个元素 $H_b(n)$.

2.2 SQMC 算法

SQMC 算法^[9] 是用 QMC 方法代替 MC 方法来实现粒子滤波的采样过程, 由于 QMC 点集在空间分布上更为均匀, 因此可获得优于 MC 方法的估计精度, 保证了 SQMC 在滤波精度上优于一般的粒子滤波算法. 算法的基本步骤如下:

步骤 1. 初始化. 在区间 $[0, 1]^d$ 生成 QMC 点集 $\{u^j\}_{j=1}^N$, 并通过式 (6)^[9] 将其转换为 $[\alpha_0, \beta_0]$ 上的 $\{x_0^j\}_{j=1}^N$, 然后计算先验密度 $p(x_0^j)$.

$$x^j = [\alpha + (\beta - \alpha) \circ u^j] \quad (6)$$

其中符号 \circ 表示向量的点乘.

步骤 2. 预测. 预测 x_t 的支撑区间 $[\alpha_t, \beta_t]$, 将生成的 QMC 点集通过式 (6) 映射到该区间内. 根据式 (7)^[9] 计算预测密度 $p(x_t|z_{1:t-1})$.

$$p(x_t|z_{1:t-1}) \approx \sum_{j=1}^N w_{t-1}^j p(x_t|x_{t-1}^j) \quad (7)$$

步骤 3. 滤波. 通过式 (8)^[9] 计算每个样本的权值 w_t^j , $j = 1, \dots, N$.

$$w_t^j \approx p(x_t^j|z_{1:t}) \propto \frac{1}{\hat{C}_t} p(z_t|x_t^j) p(x_t^j|z_{1:t-1})$$

$$\hat{C}_t = \sum_{j=1}^N p(z_t|x_t^j) p(x_t^j|z_{1:t-1}) \quad (8)$$

由式 (7) 和 (8) 可知, 每个样本 x_t^j 对应的权值都依赖于前一时刻的所有样本分布, 因此步骤 2 的时间复杂度为 $O(N^2)$. 该算法的详细说明参见文献 [9].

3 基于 QMC 的粒子滤波算法

SQMC 算法面临两个主要问题: 1) 当状态变量分布空间变化时, 需要额外的步骤来对采样空间 $[\alpha_t, \beta_t]$ 进行估计, 算法精度受制于采样空间的选择, 如果状态的真值偏离预测的采样空间, 则可能导致滤波发散; 2) 计算复杂度高, 其原因是单个粒子权值的计算依赖于前一时刻所有粒子分布, 整个算法计算复杂度为 $O(N^2)$.

3.1 QMC 重采样算法

从上述问题出发, 本文将 QMC 方法应用于重采样步骤, 用在大权重粒子周围的子空间内生成随机化 QMC 序列代替一般粒子滤波复制大权重粒子的方法. 该方法的创新之处在于: 避免了 SQMC 中的预测采样空间的步骤, 保持了一般粒子滤波的计算复杂度. 同时, 支撑点集聚集在大权重粒子的邻域内, 在一定程度上保证了有效样本的比率, 同时又避免了一般 PF 算法中的样本过度重叠, 增加了样本多样性. 本文将上述算法称为 QMC 重采样 (Quasi-Monte Carlo-based resampling, QMCR) 算法, 步骤如下:

步骤 1. 确定子代个数. 根据时刻 t 更新后的样本权值 $\{\hat{w}_t^i\}_{i=1}^N$, 分配每个样本繁殖子代个数 n_t^i , $i = 1, \dots, N$.

步骤 2. 构造采样子空间. 将 $X_t = \{x_t^i\}_{i=1}^N$ 划分为两个集合 S 和 s , 其中, S 为子代集不为空的样本构成的集合, s 为子代集为空的样本构成的集合, 令 $N_S = |S|$, $N_s = |s|$. 为叙述方便, 将 S 中的元素记作 $S_j = (x_j, w_j, n_j)$, $j = 1, \dots, N_S$. w_j 代表 x_j 对应的权值, n_j 代表 x_j 的子代个数. 以 x_j 为核心, 构造 N_S 个 d 维子空间 $[\alpha_j, \beta_j]$;

步骤 3. QMC 采样. 生成随机化 QMC 点集 $Q = \{u_i\}_{i=1}^{n_j}$, 其中 $n = N - N_S$, 将其分别截取长度为 $n_j - 1$ 的 N_S 个子集 Q^j . 通过式 (6) 把 Q^j 投影到子空间 $[\alpha_j, \beta_j]$ 中, 得到 x_j 的 $n_j - 1$ 个子代 $\{x_j(i)\}_{i=1}^{n_j-1}$, 同时, 令 x_j 的第 n_j 子代 $x_j(n_j) = n_j \times x_j - \sum_{i=1}^{n_j-1} x_j(i)$.

步骤 4. 权值更新. 令 x_j 的所有子代均分 x_j 的权重 w_j , 即子代 $x_j(i)$ 对应的权值 $w_j(i) = w_j/n_j$. 需要说明的是, 首先, 子代个数的分配可以参照随机重采样等算法思想; 其次, 由于粒子的第 n_j 个子代 $x_j(n_j) = n_j \times x_j - \sum_{i=1}^{n_j-1} x_j(i)$, 因此:

$$\sum_{i=1}^{n_j} w_j(i) x_j(i) = \sum_{i=1}^{n_j-1} w_j(i) x_j(i) + w_j(n_j) x_j(n_j) =$$

$$\sum_{i=1}^{n_j-1} w_j(i) x_j(i) + w_j(n_j) (n_j \times x_j - \sum_{i=1}^{n_j-1} x_j(i)) =$$

$$\frac{w_j}{n_j} \times n_j \times x_j = w_j \times x_j \quad (9)$$

这样就保证了所有子代的加权估计与父代保持一致.

3.2 Halton 序列随机化

在 QMCR 算法的步骤 3 中, 采用 Halton 序列来实现确定性采样过程. 为了便于误差估计, 同时避免状态空间维数 $d > 1$ 时各维样本相同, 给出了一种序列间隔取值方法来实现 Halton 序列的随机化过程: 为状态的每一维生成一个随机整数 I_l , $l = 1, \dots, d$, $I_l \sim U(1, 100)$, 然后分别以 n_0 为起点, I_l 为采样间隔, N 为序列长度, 利用式 (4) 和式 (5) 为各维生成互相独立的 Halton 序列 H_b^l , 则随机化的 d 维 Halton 序列 $U = [H_b^1 \ \dots \ H_b^d]$. 上述随机化方法中产生的所有样本均取自于原 Halton 序列, 因此满足 Owen 提出的随机化准则^[12].

3.3 QPF 算法

将 QMCR 应用于粒子滤波中, 就得到了准蒙特卡罗粒子滤波算法 (Quasi-Monte Carlo-based particle filter, QPF), QPF 避开了 SQMC 中预测采样空间的困难, 同时由于采用了 SIR 框架, 计算复杂度低于 SQMC, 与一般粒子滤波相近, 基于 QMC 的粒子滤波算法的步骤如下:

步骤 1. 初始化. 初始化样本集 $\{x_0^i, w_0^i\}_{i=1}^N$.

步骤 2. 预测. 从提议分布 $\pi(x_t|z_t, x_{t-1}^i)$ 中进行采样, 得到新样本集 $\{x_t^i\}_{i=1}^N$.

步骤 3. 更新. 根据时刻 t 的测量 z_t , 更新每个样本的权值并归一化:

$$w_t^i \propto w_{t-1}^i \frac{p(z_t|x_t^i) p(x_t^i|x_{t-1}^i)}{\pi(x_t^i|z_t, x_{t-1}^i)} \quad (10)$$

$$\hat{w}_t^i = \frac{w_t^i}{\sum_{i=1}^N w_t^i} \quad (11)$$

步骤 4. QMC 重采样. 调用 QMCR 算法, 对 $\{x_t^i, \hat{w}_t^i\}_{i=1}^N$ 进行重采样, 得到新样本集 $\{x_t^{i*}, w_t^{i*}\}_{i=1}^N$.

上述算法中, 如果把采样子空间 $[\alpha_j, \beta_j]$ 的半径收缩为 0, 则 QPF 退化为一的粒子滤波.

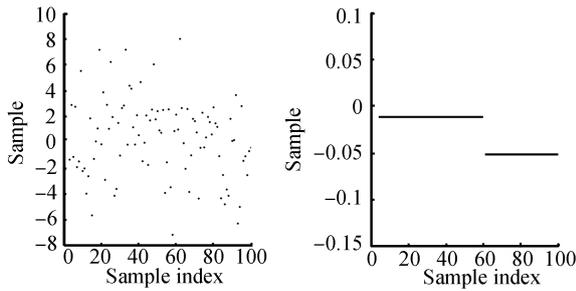
3.4 算法分析

3.4.1 样本空间分布特征

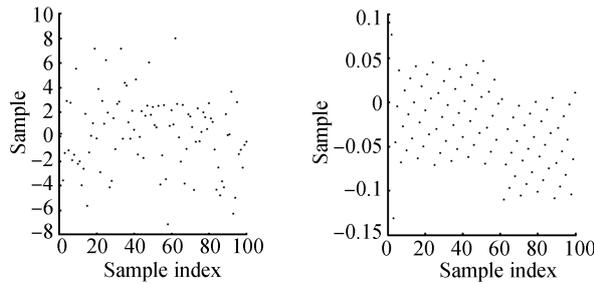
1) 在子空间 $[\alpha_j, \beta_j]$ 上, 样本由线性映射: $f: [0, 1]^d \rightarrow [\alpha_j, \beta_j]$ 通过式 (6) 生成, 因此, 在该空间上, 样本分布满足 QMC 点集的低差异性特征.

2) 整个样本空间上, 由于子空间之间可能存在交集和空隙, 在一定程度上影响了样本集的低差异性, 但可通过调节子空间半径来降低这种影响.

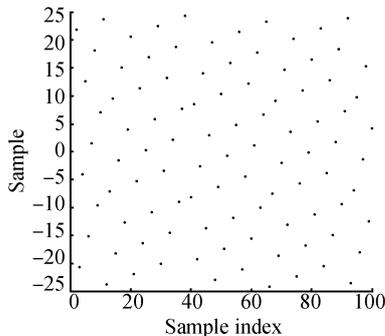
图 1 是 QPF、PF 重采样前后以及 SQMC 算法采样后的粒子分布情况, 通过对比可知, SQMC 粒子分布最为均匀, QPF 的样本在子空间内保持了低差异性特征, 子空间之间有少数粒子分散, PF 经过重采样过程, 样本出现了明显聚集.



(a) PF 重采样前后样本分布
(a) PF samples before and after resampling



(b) QPF 重采样前后样本分布
(b) QPF samples before and after resampling



(c) SQMC 样本分布
(c) SQMC samples

图 1 PF、QPF 和 SQMC 算法一个传播周期内的样本分布
Fig. 1 Samples in an iteration of PF, QPF, and SQMC

3.4.2 计算复杂度分析

QPF 算法原理如图 2 所示, 由 QPF 算法步骤可知, 算法计算复杂度与样本个数呈线性关系, 与 PF 算法的计算复杂度保持一致, 而一般的 SQMC 算法的计算复杂度为 $O(N^2)$.

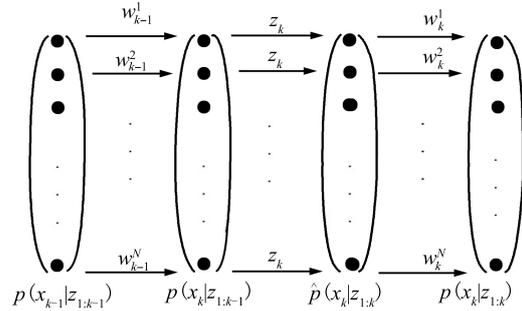


图 2 QPF 算法原理图

Fig. 2 Principle of QPF algorithm

4 仿真实验及分析

为验证 QPF 算法性能, 本文将 QPF 算法、PF 算法及 SQMC 算法应用于不同仿真条件下的三种动态状态空间模型, 分别比较了它们的估计误差、计算效率、收敛情况和跟踪稳定性等性能.

4.1 单变量非静态增长模型

单变量非静态增长模型 (Univariate nonstationary growth model, UNGM) 具有典型的非线性特征, 其状态空间方程为:

$$x_t = \alpha x_{t-1} + \beta \frac{x_{t-1}}{1 + x_{t-1}^2} + \gamma \cos(1.2t) + \omega_t \quad (12)$$

$$z_t = \frac{x_t^2}{20} + \nu_t \quad (13)$$

其中, $x_0 \sim N(0, \sigma_0^2)$, $\omega_t \sim N(0, \sigma_\omega^2)$, $\nu_t \sim N(0, \sigma_\nu^2)$. 令 $\alpha = 0.5$, $\beta = 25$, $\gamma = 8$, $\sigma_\nu = 1$. 数据样点数 $T = 60$, 仿真次数为 100 次. SQMC 算法的采样区间固定为 $[-25, 25]$, PF 与 QPF 算法均采用 $p(x_t|x_{t-1})$ 作为提议分布. 在粒子数 N 、过程噪声 σ_ω 不同的情况下比较三种算法后验加权估计的均方根误差 RMSE, 结果如表 1 所示. 其中 RMSE 的定义为: $RMSE(\hat{x}_t) = [1/T \sum_{i=1}^T (x_t - \hat{x}_t)^2]^{1/2}$. 从表 1 可知, QPF 算法精度优于 PF 算法, 与 SQMC 算法接近; 运行时间略高于 PF 算法, 明显低于 SQMC 算法, 验证了算法分析的结论.

表 1 粒子滤波和 QPF、SQMC 估计误差和运行时间比较
Table 1 Comparison of estimation errors and run times of PF, QPF, and SQMC

算法	RMSE (VAR)	Time (s)	
$N = 500, \sigma_\omega = 10$	PF	4.6913(0.9535)	1.8036
	SQMC	4.4460(0.8942)	5.1836
	QPF	4.2613(0.6435)	2.6999
$N = 100, \sigma_\omega = 10$	PF	6.3594(4.8031)	0.3953
	SQMC	4.6622(1.3812)	0.8583
	QPF	4.7994(1.8853)	0.5459
$N = 100, \sigma_\omega = 1$	PF	4.3184(1.0675)	0.4027
	SQMC	3.0500(0.8992)	0.8625
	QPF	3.0057(1.2259)	0.5465

图 3 为 $N = 100, \sigma_\omega = 10$ 的条件下的一次仿真过程. 从图 3 中可以看出, PF 在 $t = 2 \sim 9$ 时刻, 估计误差较大. 从图 4 中对应的样本有效容量可以看出, 粒子滤波发生了 3 次样本枯竭, 导致了目标丢失.

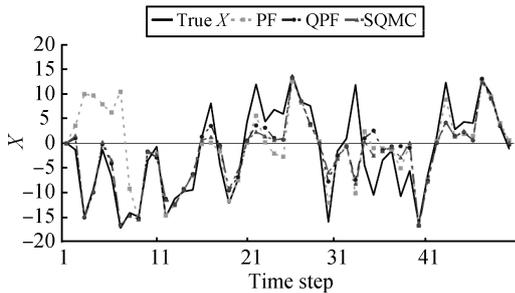


图 3 状态真值与 PF、QPF、SQMC 算法估计值

Fig. 3 True and estimated states of PF, QPF, and SQMC

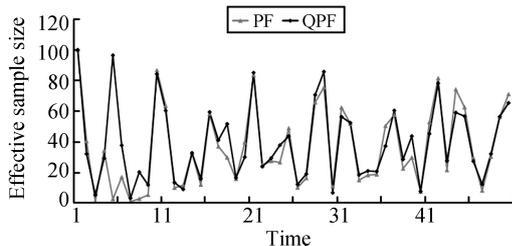


图 4 PF、QPF、SQMC 算法有效样本容量

Fig. 4 Effective sizes of samples of PF, QPF, and SQMC

图 5 给出了在不同粒子数条件下 PF 与 QPF 的误差变化情况. 从图 5 可以看出, 随着 N 的增大, QPF 的误差逐渐降低, 当 N 小于 100 时, 误差降低明显, 超过 100 后, 误差降低趋于平缓. 说明 QPF 算法随着 N 的增大而逐渐收敛, 但收敛速度并不均匀, 这与一般粒子滤波相似.

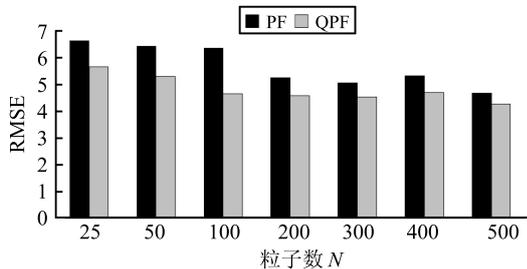


图 5 PF、QPF 算法收敛性

Fig. 5 Convergency of PF and QPF

4.2 非高斯模型

该状态空间模型为:

$$x_t = 1 + \sin(\alpha\pi t) + \phi x_{t-1} + \omega_t \quad (14)$$

$$z_t = \begin{cases} \phi x_t^2 + \nu_t, & t \leq 30 \\ \phi x_t - 2 + \nu_t, & t > 30 \end{cases} \quad (15)$$

其中, $\omega_t \sim \Gamma(3, 2), \alpha = 4E - 2, \phi = 0.5, \nu_t \sim N(0, R)$.

该模型系统噪声为非高斯分布, 测量方程具有一定的非线性特征. 把采用固定采样区间的 SQMC 算法记作 SQMC, 把用无迹卡尔曼滤波 (Unscented Kalman filter, UKF) 算法预测采样区间的 SQMC 记作 SQMC_UKF, 其中采样区间按文献 [11] 中的方法来确定. 设定粒子数 $N = 100$, 数据样点数 $T = 60$, 针对组 1 ($R = 0.1$) 和组 2 ($R = 0.0001$) 分别进行 100 次独立仿真, 实验结果如表 2 所示.

表 2 PF、QPF、UKF、SQMC、SQMC_UKF 估计误差和运行时间比较

Table 2 Comparison of estimation errors and run times of PF, QPF, UKF, SQMC, and SQMC_UKF

算法	RMSE (VAR)	Time (s)
PF	0.4161(0.0048)	1.6311
QPF	0.4030(0.0027)	1.9002
$R = 0.1$ UKF	0.5616(0.0011)	0.0993
SQMC	0.9388(0.0051)	2.4308
SQMC_UKF	0.4841(0.0046)	2.1936
PF	0.3812(0.0492)	1.7318
QPF	0.3737(0.0478)	1.9858
$R = 0.0001$ UKF	0.3017(0.0076)	0.1019
SQMC	0.4346(0.0055)	2.4573
SQMC_UKF	0.3434(0.0048)	2.5039

由表 2 可以看出: SQMC 的估计误差最高, 原因是此模型中状态空间发生了变化; SQMC_UKF 在仿真条件不同时, 其精度不稳定, 原因是其精度依赖于 UKF 预测采样区间的情况. 另外, 由于系统模型的线性特征和噪声分布特性使得粒子滤波的样本枯竭情况基本没有出现, 因此 PF 和 QPF 两者精度差异不大, 而 QPF 误差方差低于 PF, 说明 QPF 滤波稳定性能更好.

4.3 QPF 算法在目标跟踪中的应用

纯角度跟踪模型 (Bearing-only tracking, BOT) 是一种重要的目标跟踪模型, 一般假定目标在 2 维空间中运动, 传感器只能观测目标的方位角, 该模型测量方程具有典型的非线性特征.

$$x_t = \phi x_{t-1} + \Gamma \omega_t \quad (16)$$

$$z_t = \arctan \frac{y_t}{x_t} + \nu_t \quad (17)$$

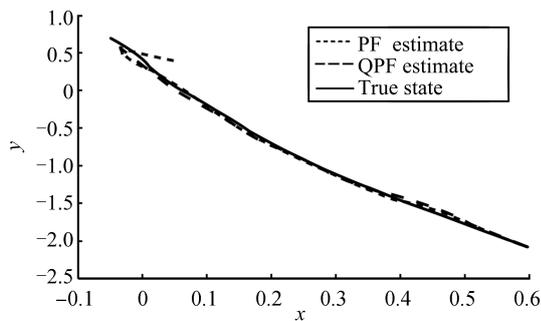
其中, $x_t = [x, \dot{x}, y, \dot{y}]^T, \omega_t = [\omega_x, \omega_y]^T, \omega_x, \omega_y$ 和 ν_t 服从零均值高斯分布, 标准差 $\sigma_\nu = 0.005, \sigma_x = \sigma_y = 0.001, \phi$ 和 Γ 的取值同文献 [2]. 设初始状态 $x_1 = [-0.05, 0.01, 0.7, -0.055]^T$.

由于实验 4.1 和实验 4.2 已将 SQMC 与 QPF 进行了对比, 因此此处只针对 PF 与 QPF 在目标跟踪中的性能进行了三组 100 次仿真实验, 结果如表 3 所示.

从表 3 中可以看出, QPF 的 RMSE 均值和方差均明显低于粒子滤波, 说明 QPF 有效地提高了目标跟踪的精度和稳定性, 而运行时间与 PF 相当. 图 6 给出了多种有代表性的单次仿真结果.

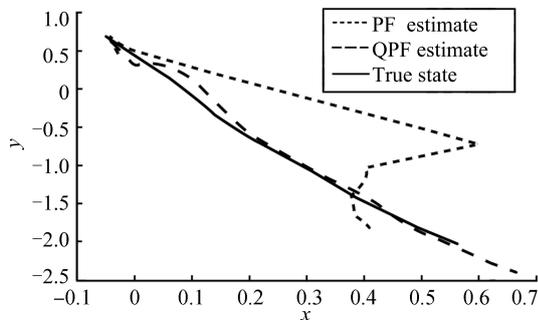
表 3 PF 算法和 QPF 算法跟踪性能比较
Table 3 Comparison of tracking performances of PF and QPF

算法	x 误差均值	x 误差方差	\dot{x} 误差均值	\dot{x} 误差方差	y 误差均值	y 误差方差	\dot{y} 误差均值	\dot{y} 误差方差	执行时间 (s)	
$N = 100$	PF	0.0858	0.0058	0.0054	8.09 E-6	0.2381	0.0200	0.0101	3.44 E-5	0.4385
	QPF	0.0651	0.0036	0.0043	5.57 E-6	0.1826	0.0090	0.0087	1.73 E-5	0.5328
$N = 500$	PF	0.0460	0.0010	0.0035	1.57 E-6	0.1835	0.0117	0.0085	3.13 E-5	2.1416
	QPF	0.0366	0.0009	0.0031	8.58 E-7	0.1561	0.0073	0.0072	1.92 E-5	2.5530
$N = 1000$	PF	0.0434	0.0012	0.0036	1.84 E-6	0.1959	0.0145	0.0094	4.09 E-5	4.3897
	QPF	0.0336	0.0003	0.0032	7.77 E-7	0.1453	0.0046	0.0069	1.03 E-5	5.4510



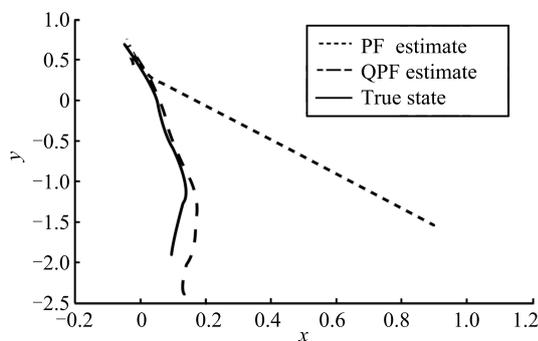
(a) 单次仿真结果 1

(a) Simulated tracking result 1



(b) 单次仿真结果 2

(b) Simulated tracking result 2



(c) 单次仿真结果 3

(c) Simulated tracking result 3

图 6 PF、QPF 算法目标跟踪仿真结果

Fig. 6 Estimated target trajectories of PF and QPF

图 6 (a) 中两种算法都进行了有效的目标跟踪, 估计精度相当; 图 6 (b) 中粒子滤波估计误差较大, QPF 跟踪效果良

好; 图 6 (c) 中展示了发散现象, 在 100 次仿真过程中, 共捕捉到粒子滤波运行中的 6 次滤波发散现象, 而 QPF 发生了 1 次滤波发散。

综合上述实验可知, QPF 算法在粒子数较少、系统噪声较小、系统具有较强的非线性特征等情况下, 具有更为明显的性能优势, 原因是这些条件更容易引发退化和样本枯竭。另外, 在系统状态空间不变的情况下, QPF 算法与 SQMC 算法精度相近, 算法时间复杂度明显低于 SQMC 算法。当状态空间变化时, UKF 对采样区间的预测效果直接影响了 SQMC 算法的滤波精度, QPF 算法不存在这一问题, 滤波稳定性更好。

5 结论

本文针对 SQMC 算法预测采样空间困难、计算复杂度高以及一般的粒子滤波重采样算法导致样本多样性损失的问题, 通过在随状态传播而变化的采样子空间上进行 QMC 再采样的方法, 来对样本进行繁殖, 改变了传统的对样本复制-抛弃的策略, 明显遏制了样本枯竭, 同时把 SQMC 算法中预测的整个采样区间分散为多个以大权值样本为核心的子空间, 降低了预测风险, 提高了滤波的鲁棒性和精度。另外, 本文中采样子空间的确定除了对父代粒子的空间位置的依赖之外, 子空间大小的选取仍然有待优化, 这将是本文下一步的研究方向。

References

- 1 Arulampalam M S, Maskell S, Gordon N, Clapp T. A tutorial on particle filters for on-line nonlinear/non-Gaussian Bayesian tracking. *IEEE Transactions on Signal Processing*, 2002, **50**(2): 174-188
- 2 Gordon N, Salmond D J, Smith A F M. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEEE Proceedings F: Radar and Signal Processing*, 1993, **140**(2): 107-113
- 3 Khan Z, Balch T, Dellaert F. MCMC-based particle filtering for tracking a variable number of interacting targets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2005, **27**(11): 1805-1819
- 4 Musso C, Oudjane N, LeGland F. Improving regularised particle filters. *Sequential Monte Carlo Methods in Practice*. New York: Springer-Verlag, 2001. 247-271
- 5 L'Ecuyer P. Quasi-Monte Carlo methods with applications in finance. *Finance and Stochastics*, 2009, **13**(3): 307-349

- 6 Boyle P P, Tan K S. Quasi-Monte Carlo methods. In: Proceedings of the 7th International AFIR Colloquium. Cairns, Australia: Society of Actuaries, 1997. 1–24
- 7 Joe S, Kuo F Y. Constructing Sobol' sequences with better two-dimensional projections. *SIAM Journal on Scientific Computing*, 2008, **30**(5): 2635–2654
- 8 Perez C J, Martin J, Rufo M J, Rojano C. Quasi-random sampling importance resampling. *Communications in Statistics: Simulation and Computation*, 2005, **34**(1): 97–112
- 9 Guo D, Wang X D. Quasi-Monte Carlo filtering in nonlinear dynamic systems. *IEEE Transactions on Signal Processing*, 2006, **54**(6): 2087–2098
- 10 Fearnhead P. Using random quasi-Monte-Carlo within particle filters, with application to financial time series. *Journal of Computational and Graphical Statistics*, 2005, **14**(4): 751–769
- 11 Huang D M, Pan Q. A new nonlinear filter algorithm based on QMC quadrature. In: Proceedings of the International Conference on Computer Science and Software Engineering. Wuhan, China: IEEE, 2008. 190–193
- 12 Owen A B. Latin supercube sampling for very high-dimensional simulations. *ACM Transactions on Modeling and Computer Simulation*, 1998, **8**(1): 71–102

赵玲玲 博士研究生。主要研究方向为信息融合和目标跟踪。本文通信作者。E-mail: zhaolinglinghit@126.com

(**ZHAO Ling-Ling** Ph.D. candidate. Her research interest covers information fusion and multi-target tracking. Corresponding author of this paper.)

马培军 博士, 教授。主要研究方向为信息融合、色彩匹配、图像处理和智能控制。E-mail: ma@hit.edu.cn

(**MA Pei-Jun** Ph.D., professor. His research interest covers information fusion, color matching, image processing, and intelligent control.)

苏小红 博士, 教授。主要研究方向为图像处理、模式识别、信息融合、软件工程和计算机图形学。E-mail: sxh@hit.edu.cn

(**SU Xiao-Hong** Ph.D., professor. Her research interest covers image processing, pattern recognition, information fusion, software engineering, and computer graphics.)
