

容积粒子滤波算法及其应用

穆静^{1,2}, 蔡远利¹, 张俊敏¹

(1. 西安交通大学电子与信息工程学院, 710049, 西安; 2. 西安工业大学计算机科学与工程学院, 710032, 西安)

摘要: 针对使用现有粒子滤波算法对非线性/非高斯离散时间系统的状态估计精度较低的问题, 提出了一种新的粒子滤波算法——容积粒子滤波(CPF)算法. 新算法使用容积数值积分原则直接计算非线性随机函数的均值和方差, 产生粒子滤波算法的建议性密度函数, 获得所需要的带权粒子, 进而通过计算粒子均值, 获得系统状态的最小均方误差估计. CPF算法由于产生粒子时使用了最新的测量信息, 因而提高了对系统状态后验概率的逼近程度. 仿真实验结果表明, CPF算法的估计误差约是标准粒子滤波算法和扩展粒子滤波算法误差的1/5和1/3, 是无味粒子滤波(UPF)算法的估计误差的1/2, 且运行时间只有UPF算法的1/3.

关键词: 粒子滤波; 容积原则; 建议性密度函数

中图分类号: TP391 **文献标志码:** A **文章编号:** 0253-987X(2011)08-0013-05

Cubature Particle Filter and Its Application

MU Jing^{1,2}, CAI Yuanli¹, ZHANG Junmin¹

(1. School of Electronic and Information Engineering, Xi'an Jiaotong University, Xi'an 710049, China; 2. School of Computer Science and Engineering, Xi'an Technological University, Xi'an 710032, China)

Abstract: A new particle filter named cubature particle filter (CPF) is proposed to improve the low state estimation accuracy of existing particle filters for nonlinear/non-Gaussian discrete time systems. The CPF directly uses the cubature rule based numerical integration method to calculate the mean and covariance, to generate the proposal density function for the particle filter, and to obtain the required particles with weights. Then the minimum square error state estimation is obtained based on these particles and weights. The particles generated using the CPF algorithm involves the use of the latest measurements so that the approximation to the system posterior density is improved. Simulation results show that the estimation error of the CPF algorithm is about one-fifth of that of the generic particle filter and one-third of that of the extended particle filter, respectively, and half of that of unscented particle filter (UPF), while the run time of CPF is only one third of that of UPF.

Keywords: state estimation; particle filter; cubature rule; proposal density function

在现代信号处理、图像处理、雷达跟踪及自动控制等领域中存在着大量的非线性/非高斯系统的状态估计问题. 粒子滤波(PF)算法是解决非线性/非高斯滤波问题的一种非常有效的方法. 粒子滤波算法是基于贝叶斯采样估计中的顺序重要性采样

(SIS)^[1-2]的滤波思想发展起来的, 其实现过程是首先选择建议性密度函数, 通过建议性密度函数获取带权随机粒子, 进而产生近似状态后验概率密度函数, 并以粒子均值代替积分运算, 从而获得状态的最小方差估计的过程. 因此, 建议性密度函数的选取是

粒子滤波算法实现中的一个关键问题。

标准粒子滤波(GPF)算法实现时,选取系统状态的转移概率函数作为建议性密度函数,即使用状态转移概率分布来产生预测样本,随着迭代次数的增加,GPF 算法会出现粒子退化和贫化现象^[3]。为了解决粒子退化和贫化问题,国内外学者选取不同的建议密度函数,提出了许多改进的粒子滤波算法^[4-7],但这些算法的滤波精度的提高是以运算量大为代价的,且运行时间较长。文献[8-9]提出的辅助粒子滤波算法和正则化粒子滤波算法考虑了当前时刻的测量信息,过程噪声较小时,滤波精度比标准粒子滤波算法精度高,但过程噪声较大时,滤波精度比标准粒子滤波算法还要低,并且正则化粒子滤波算法在非高斯情况下,核函数和核带宽系数选择不能达到最优。另外,文献[10-11]提出的扩展粒子滤波(EPF)算法和无迹粒子滤波(UPF)算法,分别使用扩展卡尔曼滤波(EKF)算法和无迹卡尔曼滤波(UKF)算法产生粒子滤波算法的建议密度函数。EPF 在估计性能上有所改善,虽然考虑到当前时刻的测量值,但使用 EKF 作为建议性密度函数在模型线性化和高斯假设中引入过多误差,其改进效果有限。由于 UKF 的滤波性能优于 EKF 算法,UPF 算法的状态估计的精度高于 EPF 算法,但 UPF 算法实现时,需产生较多的 Sigma 点,以致于算法的运行速度较慢。文献[12]提出的粒子滤波算法使用容积卡尔曼滤波(CKF)算法作为粒子滤波的建议密度函数,仅适用于高斯连续时间系统的状态估计,而不适合非高斯离散时间系统的状态估计。

容积卡尔曼滤波(CKF)算法是一种免微分的、滤波精度高的滤波算法^[13],该算法实现时使用的容积点的个数较少,运行时间短。本文使用容积卡尔曼滤波算法产生粒子滤波算法的建议性密度函数,形成一种适合于非线性/非高斯系统的粒子滤波算法——容积粒子滤波(CPF)算法。采用 CPF 算法对非线性/非高斯的离散时间系统的状态进行估计,蒙特卡罗仿真实验表明,此算法能提高系统状态的估计精度,实现简单且运行速度快,是一种有效的状态估计方法。

1 粒子滤波问题

考虑如下的非线性/非高斯离散时间动态系统

$$\left. \begin{aligned} \mathbf{x}_k &= \mathbf{f}(\mathbf{x}_{k-1}) + \mathbf{w}_{k-1} \\ \mathbf{z}_k &= \mathbf{h}(\mathbf{x}_k) + \mathbf{v}_k \end{aligned} \right\} \quad (1)$$

式中: $\mathbf{x}_k$ 和 $\mathbf{z}_k$ 为 k 时刻的系统状态向量和测量值,

且已知初始状态的概率分布密度 $p(\mathbf{x}_0)$; $\mathbf{f}(\cdot)$ 和 $\mathbf{h}(\cdot)$ 分别为系统状态转移模型函数和测量模型函数; $\mathbf{w}_k$ 和 $\mathbf{v}_k$ 分别是过程噪声和观测噪声。

标准粒子滤波算法的实现是基于序贯性重要重采样(SIS)算法^[2],其过程可描述如下。

(1)从重要性密度函数 $q(\mathbf{x}_k | \mathbf{x}_{1:k-1}, \mathbf{z}_k)$ 中随机抽取 N 个有限样本。

(2)计算每个样本 i 的权值

$$\omega_k^{(i)} = \omega_k^{(i-1)} \frac{p(\mathbf{z}_k | \mathbf{x}_k^{(i)}) p(\mathbf{x}_k^{(i)} | \mathbf{x}_k^{(i-1)})}{q(\mathbf{x}_k^{(i)} | \mathbf{x}_k^{(i-1)}, \mathbf{z}_k)} \quad (2)$$

式中: $p(\mathbf{z}_k | \mathbf{x}_k)$ 是量测 $\mathbf{z}_k$ 的似然函数。对权值 $\omega_k^{(i)}$ 进行归一化处理

$$\bar{\omega}_k^{(i)} = \frac{\omega_k^{(i)}}{\left(\sum_{j=1}^N \omega_k^{(j)} \right)} \quad (3)$$

(3) 根据每个粒子的归一化权值对样本重新采样,即保留或复制权值较大的粒子,删除权值较小的粒子,得到等权值粒子集 $\{\mathbf{x}_k^{(i)}, N^{-1}, i=1, \dots, N\}$ 。因此可获得如下的经验后验分布密度函数

$$\hat{p}(\mathbf{x}_k | \mathbf{z}_{1:k}) = \frac{1}{N} \sum_{i=1}^N \delta(\mathbf{x}_k - \mathbf{x}_k^{(i)}) \quad (4)$$

式中: $\delta(\cdot)$ 是狄拉克 delta 函数。基于最小均方误差准则意义下的状态最优估计及其方差为

$$\hat{\mathbf{x}}_k = \frac{1}{N} \sum_{i=1}^N \mathbf{x}_k^{(i)} \quad (5)$$

$$\hat{\mathbf{P}}_k = \frac{1}{N} \sum_{i=1}^N \mathbf{x}_k^{(i)} \mathbf{x}_k^{(i)T} - \hat{\mathbf{x}}_k \hat{\mathbf{x}}_k^T \quad (6)$$

2 容积卡尔曼滤波

容积卡尔曼滤波算法使用基于容积原则的数值积分方法直接计算状态的均值和方差。若 $k-1$ 时刻的后验概率为 $p(\mathbf{x}_{k-1} | \mathbf{z}_{1:k-1}) \sim N(\mathbf{x}_{k-1}; \hat{\mathbf{x}}_{k-1}, \hat{\mathbf{P}}_{k-1})$, 且获得方差 $\hat{\mathbf{P}}_{k-1}$ 的一个平方根为 $\mathbf{S}_{k-1}$, 即 $\hat{\mathbf{P}}_{k-1} = \mathbf{S}_{k-1} \mathbf{S}_{k-1}^T$, 则平方根容积卡尔曼滤波算法的步骤如下^[13]。

(1) 计算基本的容积点和对应的权值。使用三阶容积原则获得如下的基本容积点和相应权值

$$\xi_j = \left(\frac{m}{2} \right)^{1/2} [1]_j \quad (7)$$

$$\omega_j = \frac{1}{m} \quad (8)$$

式中: $j=1, \dots, m$; m 表示容积点总数,使用三阶容积原则,容积点总数是状态维数的 2 倍,即 $m=2n_x$;

$[1]_j$ 表示点集中在 $[1]$ 的第 j 个点,其中符号 $[1]$ 为完整全对称点集,表示对 n_x 维单位向量 $\mathbf{e}=[1,0,\dots,0]^T$ 的元素进行全排列和改变元素符号所产生的点集。

(2) 时间更新. 计算容积点

$$\mathbf{X}_{j,k-1} = \mathbf{S}_{k-1}\hat{\boldsymbol{\xi}}_j + \hat{\mathbf{x}}_{k-1} \quad (9)$$

计算通过状态方程传播的容积点

$$\mathbf{X}_{j,k}^* = \mathbf{f}(\mathbf{X}_{j,k-1}) \quad (10)$$

计算状态预测和方差预测的平方根

$$\bar{\mathbf{x}}_k = \sum_{i=1}^m \omega_i \mathbf{X}_{i,k}^* \quad (11)$$

$$\bar{\mathbf{S}}_k = \text{tria}([\boldsymbol{\chi}_k^*, \mathbf{S}_{Q,k-1}]) \quad (12)$$

式中:tria()表示对矩阵进行三角化. 矩阵 $\boldsymbol{\chi}_k^*$ 定义为

$$\boldsymbol{\chi}_k^* = m^{-1/2}[\mathbf{X}_{1,k}^* - \bar{\mathbf{x}}_k, \mathbf{X}_{2,k}^* - \bar{\mathbf{x}}_k, \dots, \mathbf{X}_{m,k}^* - \bar{\mathbf{x}}_k] \quad (13)$$

(3) 测量更. 计算容积点

$$\mathbf{X}_{j,k} = \bar{\mathbf{S}}_k \hat{\boldsymbol{\xi}}_j + \bar{\mathbf{x}}_k \quad (14)$$

计算通过测量方程传播的容积点和测量预测

$$\mathbf{Z}_{j,k} = \mathbf{h}(\mathbf{X}_{j,k}) \quad (15)$$

$$\bar{\mathbf{z}}_k = \sum_{i=1}^m \omega_i \mathbf{Z}_{i,k} \quad (16)$$

计算新息方差的平方根和协方差

$$\mathbf{S}_{\Sigma,k} = \text{tria}([\mathbf{A}_k, \mathbf{S}_{R,k}]) \quad (17)$$

$$\mathbf{P}_{\Sigma,k|k-1} = \boldsymbol{\chi}_k \mathbf{A}_k^T \quad (18)$$

式中:矩阵 $\boldsymbol{\chi}_k, \mathbf{A}_k$ 定义为

$$\boldsymbol{\chi}_k = m^{-1/2}[\mathbf{X}_{1,k} - \bar{\mathbf{x}}_k, \mathbf{X}_{2,k} - \bar{\mathbf{x}}_k, \dots, \mathbf{X}_{m,k} - \bar{\mathbf{x}}_k] \quad (19)$$

$$\mathbf{A}_k = m^{-1/2}[\mathbf{Z}_{1,k} - \bar{\mathbf{z}}_k, \mathbf{Z}_{2,k} - \bar{\mathbf{z}}_k, \dots, \mathbf{Z}_{m,k} - \bar{\mathbf{z}}_k] \quad (20)$$

计算增益、 k 时刻的状态和方差的平方根估计

$$\mathbf{W}_k = (\mathbf{P}_{\Sigma,k} / \mathbf{S}_{\Sigma,k}^T) / \mathbf{S}_{\Sigma,k} \quad (21)$$

$$\hat{\mathbf{x}}_k = \bar{\mathbf{x}}_k + \mathbf{W}_k(\bar{\mathbf{z}}_k - \bar{\mathbf{z}}_k) \quad (22)$$

$$\mathbf{S}_k = \text{tria}([\boldsymbol{\chi}_k - \mathbf{W}_k \mathbf{A}_k, \mathbf{W}_k \mathbf{S}_{R,k}]) \quad (23)$$

式中:符号/为矩阵的右除运算符。

3 容积粒子滤波算法

实现粒子滤波算法时,重要性密度函数的选择是关键问题之一,本文采用比EKF和UKF算法性能更好的容积滤波(CKF)算法来产生重要性密度函数,获得新的粒子滤波算法——CPF算法. 容积粒子滤波算法描述如下。

(1) 初始化($k=0$),以获得初始粒子 $\mathbf{x}_0^{(i)}$ 和对

应的方差 $\hat{\mathbf{P}}_0^{(i)}$,即从初始状态概率分布密度 $p(\mathbf{x}_0)$ 抽取粒子 $\mathbf{x}_0^{(i)} (i=1, \dots, N)$,且

$$\bar{\mathbf{x}}_0^{(i)} = E[\mathbf{x}_0^{(i)}] \quad (24)$$

$$\hat{\mathbf{P}}_0^{(i)} = E[(\mathbf{x}_0^{(i)} - \bar{\mathbf{x}}_0^{(i)})(\mathbf{x}_0^{(i)} - \bar{\mathbf{x}}_0^{(i)})^T] \quad (25)$$

(2) 当时刻 $k=1, 2, \dots$,进行如下步骤:①

若 $k-1$ 时刻具有等权值 $1/N$ 的粒子集为 $\{\mathbf{x}_{k-1}^{(i)}, \hat{\mathbf{P}}_{k-1}^{(i)}, i=1, \dots, N\}$,对粒子进行容积卡尔曼滤波,得到粒子集及协方差为 $(\hat{\mathbf{x}}_k^{(i)}, \hat{\mathbf{P}}_k^{(i)})$;②选择的重要性密度函数是以 $\hat{\mathbf{x}}_k^{(i)}$ 为均值、 $\hat{\mathbf{P}}_k^{(i)}$ 为方差的高斯分布函数,即

$$q(\mathbf{x}_k^{(i)} | \mathbf{x}_{k-1}^{(i)}, \mathbf{z}_{1:k}) = N(\hat{\mathbf{x}}_k^{(i)}, \hat{\mathbf{P}}_k^{(i)}) \quad (26)$$

使用式(26)定义的重要性密度函数来产生预测粒子集及对应的方差 $\{\hat{\mathbf{x}}_k^{(i)}, \hat{\mathbf{P}}_k^{(i)}, i=1, \dots, N\}$;③使用式(2)计算每一个粒子的权值 $\omega_k^{(i)}$,且使用式(3)获得粒子归一化的权值 $\bar{\omega}_k^{(i)}$;④根据归一化的权值 $\bar{\omega}_k^{(i)}$ 对粒子集 $\{\hat{\mathbf{x}}_k^{(i)}, \hat{\mathbf{P}}_k^{(i)}, i=1, \dots, N\}$ 进行重采样,获得等权值为 $1/N$ 的粒子集 $\{\mathbf{x}_k^{(i)}, \hat{\mathbf{P}}_k^{(i)}, i=1, \dots, N\}$;⑤使用式(5)和式(6)输出 k 时刻的状态和方差估计 $\hat{\mathbf{x}}_k$ 和 $\hat{\mathbf{P}}_k$ 。

4 实验及分析

将本文提出的容积粒子滤波(CPF)算法与标准粒子滤波(GPF)、扩展粒子滤波(EPF)和无迹粒子滤波(UKF)算法进行比较. 实验所用的系统模型和量测模型为^[4]

$$x_{k+1} = 1 + \sin(\omega \pi k) + \varphi_1 x_k + v_k \quad (27)$$

$$y_k = \begin{cases} \varphi_2 x_k^2 + n_k, & k \leq 30 \\ \varphi_3 x_k - 2 + n_k, & k > 30 \end{cases} \quad (28)$$

式中: $\omega=4 \times 10^{-2}$; $\varphi_1=0.5$; $\varphi_2=0.2$; $\varphi_3=0.5$; v_k 是服从 $\Gamma(3,2)$ 分布的随机变量;观测噪声 n_k 是服从 $N(0,0.0001)$ 分布的高斯随机变量. 系统状态的观测使用2个不同阶段的观测模型. 实验中,粒子数选取为 $N=200$,采用残差重采样方法,时间步长选取为 $T=60$ s,蒙特卡罗实验仿真次数 $M=100$. 使用均方根误差和累加均方根误差进行算法比较,状态 x 的均方根误差(RMSE)定义为

$$E_{x, \text{RMSE}} = \left(\frac{1}{M} \sum_{j=1}^M (\hat{x}_k^{(j)} - x_k^{(j)})^2 \right)^{1/2} \quad (29)$$

每一次独立实验的累加均方根误差(ARMSE)定义为

$$E_{x,ARMSE} = \left(\frac{1}{T} \sum_{i=1}^T (\hat{x}_i - x_i)^2 \right)^{1/2} \quad (30)$$

100次独立实验后获得的4种粒子滤波算法对系统的状态估计如图1所示。图2为100次独立实验后获得的4种粒子滤波算法状态估计的均方根误差曲线。各滤波算法的累加均方根误差的均值和方差如表1所示。表2列出了4种算法的100次独立实验的平均运行时间。

从图1可以看出,标准粒子滤波(GPF)算法和扩展粒子滤波(EPF)算法的状态估计有时会严重偏离真实状态,而无迹粒子滤波(UPF)与容积粒子滤波(CPF)算法的估计状态与真实状态比较吻合。

由图2可知,GPF和EPF算法的均方根误差较大,UPF和CPF算法的均方根误差总体要比GPF和EPF算法的小。

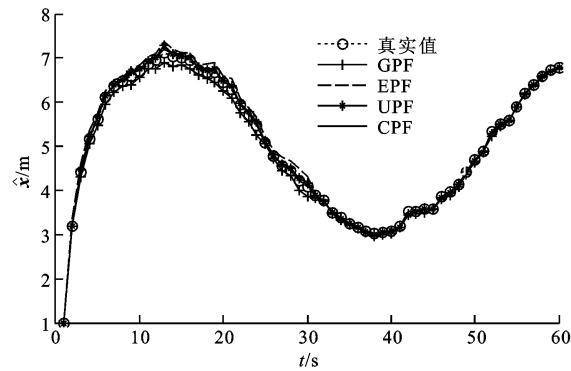


图1 4种粒子滤波算法的状态估计

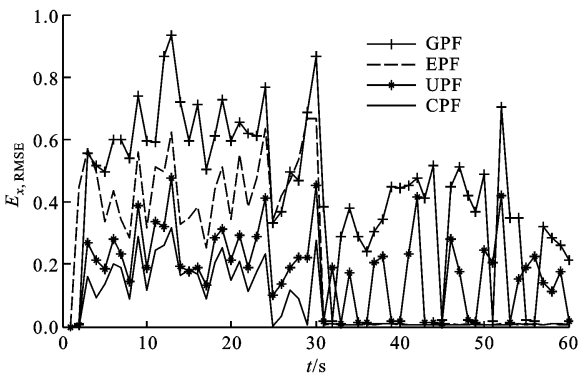


图2 4种算法的均方根误差

从表1同样可以看出,GPF和EPF算法的累加均方根误差的均值和方差较大,而UPF和CPF算法的累加均方根误差的均值和方差大大降低。这是由于使用UKF和CKF算法作为重要性密度函数,更能逼近最优的重要性密度分布,且使用最新的测量信息产生预测粒子集,因此UPF和CPF算法对

状态的估计精度要明显高于GPF和EPF算法。从图1、图2和表1可知,CPF算法的误差约是GPF和EPF算法误差的1/5和1/3,是UPF算法误差的1/2,因此CPF算法对状态的估计精度更高。

从表2可以看出,CPF算法的运行时间比EPF算法的稍短,远远短于UPF算法的运行时间,大约只有UPF算法的1/3。

表1 运行100次的累加均方根误差的均值和方差

滤波算法	$E_{x,ARMSE}$ 的均值/m	$E_{x,ARMSE}$ 的方差/ m^2
GPF	0.442 31	0.060 673
EPF	0.302 49	0.0147 34
UPF	0.182 24	0.016 541
CPF	0.091 50	0.007 249

表2 4种算法的平均运行时间

滤波算法	平均运行时间/s
GPF	0.521 6
EPF	3.755 9
UPF	7.221 6
CPF	2.987 9

5 结 论

通过以容积卡尔曼滤波器为建议性密度函数的容积粒子滤波算法(CPF)对非线性/非高斯系统的状态进行估计,算法实现简单,状态估计精度较高,且运行速度快。仿真结果表明,CPF算法明显优于标准粒子滤波算法、扩展粒子滤波算法,优于无迹粒子滤波算法,并且运行速度大大快于无迹粒子滤波算法。

参考文献:

[1] LIU Jun, CHEN Rong. Sequential Monte Carlo methods for dynamical systems [J]. Journal of American Statistical Association, 1998, 93(443):1032-1044.

[2] DOUCET A, DE FREITAS J F G, GORDON N J. An introduction to sequential Monte Carlo methods [M]// Doucet A, Gordon N. Sequential Monte Carlo Methods in Practice. New York, USA: Springer-Verlag, 2001:3-14.

[3] 胡士强, 敬中良. 粒子滤波算法综述[J]. 控制与决策, 2005, 20(4):361-365.

HU Shiqiang, JING Zhongliang. Overview of particle

- filter algorithm [J]. Control and Decision, 2005, 20(4):361-365.
- [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-Hermit particle filter [J]. Acta Electronic Sinica, 2003, 31(7):970-973.
- [5] 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.
- [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 filter [J]. Journal of Xi'an Jiaotong University, 2009, 43(2):25-28.
- [7] KABA OGLU N. Target tracking using particle filters with support vector regression [J]. IEEE Trans on Vehicular Technology, 2009, 58(5): 2569-2573.
- [8] PITT M K, SHEPHARD N. Filtering via simulation: auxiliary particle filters [J]. Journal of the American Statistical Association, 1999, 94 (2):590-599.
- [9] MUSSO C, OUDJANE N, LEGLAND F. Improving regularized particle filters [M] // Doucet A, Gordon N. Sequential Monte Carlo Methods in Practice. New York, USA: Springer-Verlag, 2001:247-271.
- [10] DE FREITAS J F G, NIRANJAN M, GEE A H, et al. Sequential Monte Carlo methods to train neural network models [J]. Neural Computation, 2000, 12(4): 955-993.
- [11] VAN DER MERWE R, DOUCET A, DE FREITAS N, et al. The unscented particle filter, CUED/F-INFEN G/TR, 380 [R]. Cambridge, UK: Cambridge University Press, 2000:1-45.
- [12] HAVLICEK M, JAN J, BRAZDIL M, et al. Nonlinear estimation of BOLD signal based on cubature particle filter [EB/OL]. (2010-06-27) [2010-11-01]. <http://www.biosignal.cz/bs2010/papers/1078.pdf>.
- [13] ARASARATNAM I, HAYKIN S. Cubature Kalman filters [J]. IEEE Trans on Automatic Control, 2009, 54(6): 1254-1269.

[本刊相关文献链接]

- 引入模糊推理与强跟踪滤波技术的变结构多模型估计. 西安交通大学学报,2011,45(6):27-33.
- 观测域求积分卡尔曼滤波的机载无源定位算法. 西安交通大学学报,2011,45(5):137-142.
- 广义证据理论中的基本概率指派生成方法. 西安交通大学学报,2011,45(2):34-38.
- 频谱包络滤波器及其应用. 西安交通大学学报,2010,44(8):48-52.
- 高斯衍生粒子滤波器. 西安交通大学学报,2010,44(6):72-77.
- 阈值去噪下的改进粒子滤波算法. 西安交通大学学报,2010,44(2):31-34.
- 广义证据理论的基本框架. 西安交通大学学报,2010,44(12):119-124.
- 合成孔径雷达图像的最小均方误差线性最优滤波. 西安交通大学学报,2009,43(12):6-10.
- 一种自相似业务量预测的卡尔曼滤波算法. 西安交通大学学报,2009,43(4):57-61.
- 求积分卡尔曼粒子滤波算法. 西安交通大学学报,2009,43(2):25-28.
- 一种多线索融合的均值偏移跟踪算法. 西安交通大学学报,2009,43(10):42-46.
- 法向约束的多幅点云数据融合算法. 西安交通大学学报,2009,43(5):71-75.
- 一种 Log-Gabor 滤波器结合多分辨率分析的虹膜识别方法. 西安交通大学学报,2009,43(4):31-33.
- 一种基于分辨函数的属性约简算法及其应用. 西安交通大学学报,2008,42(12):1455-1458.

(编辑 刘杨)