文章编号: 1001-0920 (2012) 10-0000-00

Cubature 卡尔曼滤波-卡尔曼滤波算法

孙 枫, 唐 李 军

(哈尔滨工程大学 自动化学院，哈尔滨 150001)

摘 要: 针对条件线性高斯状态空间模型, 提出 cubature 卡尔曼滤波-卡尔曼滤波算法 (CKF-KF), 分别应用 CKF 和 KF 估计模型中的非线性和线性状态. 该算法对非线性与线性状态均进行 cubature 采样, 并将两种样本通过线性方程和量测方程进行传播, 以获得非线性状态估计. 机动目标跟踪仿真结果表明, CKF-KF 的估计精度比 Rao-Blackwellized 粒子滤波器 (RBPF) 略低,但算法运行时间不到其 1%; 与无迹卡尔曼滤波器 (UKF-KF) 相比, 估计精度相当, 但算法运行时间降低了 22%,有效地提高了实时性.

关键词: 条件线性高斯模型；cubature 卡尔曼滤波-卡尔曼滤波；RBPF；UKF-KF；实时性

中图分类号: U 249 **文献标志码:** A

Cubature Kalman filter-Kalman filter algorithm

SUN Feng, *TANG Li-jun*

(College of Automation，Harbin Engineering University，Harbin 150001，China．Correspondent：TANG Li-jun，E-mail：strapdown@163.com)

Abstract: A filtering algorithm, cubature Kalman filter-Kalman(CKF-KF) filter, is proposed for conditionally linear Gaussian state model, which respectively employs CKF and KF to estimate nonlinear state and linear state in the model. The above states are carried out cubature sampling, which are propagated through linear and observation equations to estimate nonlinear state. The maneuvering target tracking simulation results show that, compared to the Rao-Blackwellized particle filter(RBPF), the algorithm running time of CKF-KF is less than 1% of that with a slightly lower filtering performance loss, and the estimation accuracy of CKF-KF coincides with that of UKF-KF, whereas the algorithm running time reduces by 22% and effectively improves real-time.

Key words: conditionally linear Gaussian model；cubature Kalman filter-Kalman filter；RBPF；UKF-KF；real-time

1 引 言

条件线性高斯状态空间模型由参数满足马尔科夫过程的非线性状态及线性状态两部分组成[1], 其模型在许多实际领域[2-4]里得到了广泛应用. 条件线性高斯模型状态估计的常用方法是采用粒子滤波 (PF) 与卡尔曼滤波 (KF) 相结合的 Rao-Blackwellized particle filter(RBPF) 滤波器[3-7]. 因为该模型下的状态变量的线性部分在非线性部分的条件下的后验分布可用解析方法求得, 从而可使用 KF 估计线性状态, 使用 PF 估计非线性状态. 因为 RBPF 沿用了 PF 的框架, 所以也就面临了计算量大、难以满足实时性的问题. 解决 RBPF 计算量大和提高实时性的关键在于寻求新的非线性滤波方法取代 PF, 对条件线性高斯模型的非线性状态进行估计.基于上述思想, 文献 [8] 提出了无迹卡尔曼滤波器(UKF)[9]与 KF 相结合的 unscented 卡尔曼滤波-卡尔曼滤波算法. 该算法对模型的非线性部分采用计算量远小于 PF 的 UKF, 而线性部分依旧采用 KF. 与 RBPF 相比, 该算法在略失估计精度的情况下, 大幅度地降低了计算量、提高了实时性. 然而 UKF 有个缺点: 参数α, β, κ 在实际中需要合理地进行调节, 才能取得好的滤波效果,操作起来不方便, 且需要 $2n+1$ 个采样点. 为了避免 UKF 的上述不足, 减少采样点以及进一步提高模型状态估计的实时性, 本文沿用文献 [8] 的思想, 将最近出现的非线性滤波 cubature Kalman filter(CKF)[10]与 KF 结合, 提出 cubature 卡尔曼滤波-卡尔曼滤波算法 (CKF-KF). 机动目标仿真表明, CKF-KF 估计精度与 UKF-KF 相当, 但其算法运行时间降低了 22%.

收稿日期: 2011-04-08；**修回日期:** 2011-05-20.

基金项目: 国家自然科学基金项目 (60775001,60834005).

作者简介: 孙枫(1944–), 男, 教授, 博士生导师, 从事惯性导航、信息融合等研究；唐李军(1984–), 男, 博士生, 从事惯性导航、非线性滤波的研究.

2 条件线性高斯状态空间模型

条件线性高斯状态空间模型如下[1, 5]:

$$z_k \sim p(z_k/z_{k-1}), \tag{1}$$

$$x_k = A(z_k)x_{k-1} + B(z_k)w_k + F(z_k)u_k, \tag{2}$$

$$y_k = C(z_k)x_k + D(z_k)v_k + G(z_k)u_k. \tag{3}$$

其中: z_k 为非线性状态, 为马尔科夫过程, 其过程噪声 $r_k \sim N(0, Q_r)$; x_k 为线性状态; y_k 为观测量; u_k 为已知的控制输入; 高斯噪声 $w_k \sim N(0, Q_w)$; $v_k \sim N(0, R)$; A, B, C, D, F, G 分别为已知的状态矩阵、观测矩阵、噪声矩阵及输入输出矩阵. 为了完成对非线性状态 z_k 及线性状态 x_k 的状态估计, RBPF 借用了粒子滤波及卡尔曼滤波的框架[5]. RBPF 利用粒子滤波算法对 z_t^i 进行采样, 然后用卡尔曼滤波对 x_k 的均值 u_t^i 和方差 Σ_t^i 进行预测和更新. 但 RBPF 无法解决计算量大, 实时性差的问题, 因此需要寻求新的滤波途径来解决上述问题.

3 Cubature Kalman filter-Kalman filter 算法

3.1 Cubature 卡尔曼滤波

考虑如下形式的多维权重积分:

$$I(f) = \int_{R^n} f(x)\exp(-x^{\mathrm{T}}x)\mathrm{d}x. \tag{4}$$

其中: $f(x)$ 为任意函数, R^n 为积分区域. 一般情况下, 上述积分的解析值无法获得, 需通过数值积分方法来得到其近似值. 可选择一组具有权重值的点集 (ω_i, ξ_i) 来近似积分:

$$I(f) = \sum_{i=1}^{m} \omega_i f(\xi_i). \tag{5}$$

最近, 文献 [9] 提出一种基于 spherical-radial 准则的 CKF 非线性高斯滤波来选择上述点集. CKF 通过 spherical-radial 准则选取 $2n$(n为状态维数) 个具有相同权值的 cubature 点:

$$\begin{cases} \xi_i = \sqrt{\dfrac{2n}{2}}[1]_i, \\ \omega_i = \dfrac{1}{2n}, \\ i = 1, 2, \cdots, 2n. \end{cases} \tag{6}$$

其中: $[1]_i$ 表示集合 [1] 的第 i 列, 对于二维即 $n = 2$, 有

$$[1] = \left\{[1, 0]^{\mathrm{T}}, [-1, 0]^{\mathrm{T}}, [0, -1]^{\mathrm{T}}, [0, 1]^{\mathrm{T}}\right\},$$

计算出 cubature 点集 (ω_i, ξ_i) 后就可以通过时间更新和量测更新得到 CKF 滤波算法.

3.1.1 时间更新

CKF 算法时间更新过程方程分别如下:

$$X_{i,k-1|k-1} = \sqrt{P_{k-1|k-1}}\xi_i + \hat{x}_{k-1|k-1}, \tag{7}$$

$$X^*_{i,k|k-1} = f(X_{i,k-1|k-1}), \tag{8}$$

$$\hat{x}_{k|k-1} = \frac{1}{2n}\sum_{i=1}^{2n} X^*_{i,k|k-1}, \tag{9}$$

$$P_{k|k-1} = \frac{1}{2n}\sum_{i=1}^{2n} X^*_{i,k|k-1}X^{*\mathrm{T}}_{i,k|k-1} - \hat{x}_{k|k-1}\hat{x}^{\mathrm{T}}_{k|k-1} + Q_{k-1}. \tag{10}$$

3.1.2 量测更新

CKF 算法量测更新过程方程分别如下:

$$X_{i,k|k-1} = \sqrt{P_{k|k-1}}\xi_i + \hat{x}_{k|k-1}, \tag{11}$$

$$Z_{i,k|k-1} = h(X_{i,k|k-1}), \tag{12}$$

$$\hat{z}_{k|k-1} = \frac{1}{2n}\sum_{i=1}^{2n} Z_{i,k|k-1}, \tag{13}$$

$$P_{xz,k|k-1} = \sum_{i=1}^{2n} \omega_i X_{i,k|k-1}Z^{\mathrm{T}}_{i,k|k-1} - \hat{x}_{k|k-1}\hat{z}^{\mathrm{T}}_{k|k-1}, \tag{14}$$

$$P_{xz,k|k-1} = \sum_{i=1}^{2n} \omega_i X_{i,k|k-1}Z^{\mathrm{T}}_{i,k|k-1} - \hat{x}_{k|k-1}\hat{z}^{\mathrm{T}}_{k|k-1}, \tag{15}$$

$$W_k = P_{xz,k|k-1}P^{-1}_{zz,k|k-1}, \tag{16}$$

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + W_k(z_k - \hat{z}_{k|k-1}), \tag{17}$$

$$P_{k/k} = P_{k/k-1} - W_kP_{zz,k/k-1}W_k^{\mathrm{T}}. \tag{18}$$

3.2 Cubature Kalman filter- Kalman filter 算法

本文根据文献 [8] 的思想, 提出 CKF 与 KF 进行融合的 CKF-KF 算法, 并利用该算法对条件高斯状态模型进行状态估计, 其中: 使用 CKF 进行非线性状态 z_k 估计, 并使用 KF 进行线性状态 x_k 估计. CKF-KF 算法过程如下.

3.2.1 CKF 估计非线性状态 z_k

将式 (1) 改写为

$$z_k = f(z_{k-1}, r_{k-1}). \tag{19}$$

其中: $f(\cdot)$ 为非线性函数, $r_k \sim N(0, Q_r)$ 为噪声.

$k-1$ 时刻的状态 z 和 x 的估计值分别为 $\bar{z}_{k-1}$ 和 $\bar{x}_{k-1}$, 估计方差为 $P_{z,k-1}$ 和 $P_{x,k-1}$, 对 $\bar{z}_{k-1}$ 和 $P_{z,k-1}$ 进行状态扩维 (扩维后的状态表示为 z^a, 维数为 n)

$$\bar{z}^a_{k-1} = \begin{bmatrix} \bar{z}^{\mathrm{T}}_{k-1} & r^{\mathrm{T}}_{k-1} & v^{\mathrm{T}}_{k-1} \end{bmatrix}^{\mathrm{T}}, \tag{20}$$

$$P^a_{z,k-1} = \begin{bmatrix} P_{z,k-1} & 0 & 0 \\ 0 & Q_r & 0 \\ 0 & 0 & R \end{bmatrix}. \tag{21}$$

此时根据式 (7), 扩维后的状态 z^a 的 $2n$ 个 cubature 点可表示为

$$Z^a_{i,k-1} = \sqrt{P^a_{z,k-1}}\xi_i + \bar{z}^a_{k-1},\quad i = 1, 2, \cdots, 2n. \tag{22}$$

根据式 (19) 对 cubature 点 $Z^a_{i,k-1}$ 进行非线性传播, 即

$$Z^{*a}_{i,k/k-1} = f(Z^a_{i,k-1}), \tag{23}$$

从而得到状态 z^a 的预测值和方差为

$$\hat{z}^a_{k/k-1} = \frac{1}{2n}\sum_{i=1}^{2n} Z^{*a}_{i,k/k-1}, \tag{24}$$

$$P^a_{Z_{k/k-1}} = \frac{1}{2n}\sum_{i=1}^{2n} Z^{*a}_{i,k/k-1} Z^{*a\mathrm{T}}_{i,k/k-1} - \hat{z}^a_{k/k-1}\hat{z}^{a\mathrm{T}}_{k/k-1}. \tag{25}$$

根据式 (11) 和 (12) 可知, 观测量 y 的预测值需利用重新生成的状态 z^a 的 $i(i = 1, 2, \cdots, 2n)$ 个 cubature 点进行传播, 即

$$Z^a_{i,k/k-1} = \sqrt{P^a_{z,k/k-1}}\xi_i + \hat{z}^a_{k/k-1}. \tag{26}$$

结合式 (2), (3) 和 (12) 可知, CKF 算法决定了在量测更新阶段观测量 y 的预测值也需要通过量测方程 (3) 传播样本点形式的状态 x, 因此, 状态 x 需要进行 cubature 采样. 同时, 为了与状态 z^a 的 $i\,(i = 1, 2, \cdots, 2n)$ 个 cubature 点的数目进行匹配而共同实现 y 的预测, 状态 x 也需要 $i(i = 1, 2, \cdots, 2n)$ 个 cubature 点, 即

$$X_{i,k/k-1} = \sqrt{P_{x,k-1}}\xi_i + \bar{x}_{k-1}. \tag{27}$$

将式 (26), (27) 代入 (2) 有

$$\begin{aligned} x_{i,k/k-1} =& A(Z^a_{i,k/k-1})X_{i,k/k-1} + \\ & B(Z^a_{i,k/k-1})w_{i,k} + F(Z^a_{i,k/k-1})u_{i,k}. \end{aligned} \tag{28}$$

结合式 (26), (28) 和 (3), 有

$$\begin{aligned} y_{i,k/k-1} =& C(Z^a_{i,k/k-1})x_{i,k/k-1} + \\ & D(Z^a_{i,k/k-1})v_{i,k} + G(Z^a_{i,k/k-1})u_{i,k}. \end{aligned} \tag{29}$$

最后, 根据式 (13)~(18) 求取 k 时刻状态 z^a 的估计值 $\hat{z}^a_k$ 和方差估计值 $P^a_{z,k}$, 即

$$\hat{y}_{k|k-1} = \frac{1}{2n}\sum_{i=1}^{2n} y_{i,k|k-1}, \tag{30}$$

$$P_{yy,k/k-1} = \frac{1}{2n}\sum_{i=1}^{2n} y_{i,k/k-1}y^{\mathrm{T}}_{i,k/k-1} - \hat{y}_{k/k-1}\hat{y}^{\mathrm{T}}_{k/k-1}, \tag{31}$$

$$P_{zy,k/k-1} = \frac{1}{2n}\sum_{i=1}^{2n} Z^a_{i,k/k-1}y^{\mathrm{T}}_{i,k/k-1} - \hat{z}^a_{k/k-1}\hat{y}^{\mathrm{T}}_{k|k-1}, \tag{32}$$

$$W_k = P_{zy,k|k-1}P^{-1}_{yy,k|k-1}, \tag{33}$$

$$\hat{z}^a_k = \hat{z}^a_{k/k-1} + W_k(y_k - \hat{y}_{k|k-1}), \tag{34}$$

$$P^a_{z,k} = P^a_{z,k/k-1} - W_k P_{yy,k/k-1} W^{\mathrm{T}}_k. \tag{35}$$

注 1　状态 z 的估计值 $\hat{z}_k$ 和方差估计值 $P_{z,k}$ 是从 $\hat{z}^a_k$ 和 $P^a_{z,k}$ 中提取得到的.

3.2.2　KF 估计线性状态 x_k

通过 CKF 估计得到状态 z 的估计值 $\hat{z}_k$ 后, 式 (2) 和 (3) 就呈现为线性状态空间模型, 从而直接利用卡尔曼滤波估计线性状态 x_k 可得

$$x_{k/k-1} = A(\hat{z}_k)\bar{x}_{k-1} + F(\hat{z}_k)u_k, \tag{36}$$

$$y_{k/k-1} = C(\hat{z}_k)x_{k/k-1} + G(\hat{z}_k)u_k, \tag{37}$$

$$P_{x,k/k-1} = BQ_wB^{\mathrm{T}} + AP_{x,k-1}A^{\mathrm{T}}, \tag{38}$$

$$J_k = DRD^{\mathrm{T}} + CP_{x,k-1}C^{\mathrm{T}}, \tag{39}$$

$$K = P_{x,k/k-1}C^{\mathrm{T}}J^{-1}_k, \tag{40}$$

$$\bar{x}_k = x_{k/k-1} + K(y_k - y_{k/k-1}), \tag{41}$$

$$P_{x,k} = P_{x,k/k-1} - KCP_{x,k/k-1}. \tag{42}$$

4　仿真分析

下面通过一个具有代表性的例子考察 RBPF, UKF-KF, CKF-KF 三种滤波算法. 机动目标跟踪模型[7, 11]为

$$\begin{cases} z_k = \dfrac{z_{k-1}}{2} + \dfrac{25z_{k-1}}{1+z^2_{k-1}} + 8\cos(1.2(k-1)) + r_k, \\ x_k = Ax_{k-1} + Fz_k + w_k, \\ y_k = Cx_k + v_k. \end{cases} \tag{43}$$

其中: r, w, v 均为零均值高斯噪声, 方差分别为 Q_r, Q_w, R; z_k 为机动目标值, $x_k = [x, y, \dot{x}, \dot{y}]^{\mathrm{T}}$ 分别表示 k 时刻目标的 x 和 y 方向的位置和速度. 仿真参数如下:

$$A = \begin{bmatrix} 1 & 0 & 0.1 & 0 \\ 0 & 1 & 0 & 0.1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix},\ F = \begin{bmatrix} 1.25 \\ 1.25 \\ 0.25 \\ 0.25 \end{bmatrix},$$

$$Q_r = 1,\ C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix},$$

$$Q_w = 0.09I_{4\times4},\ R = 9I_{4\times4},$$

位置、速度的初始值真值为 $x_0 = [20, 30, 1.2, 1]^{\mathrm{T}}$, 初始方差为

$$P_{x,0} = \begin{bmatrix} 10 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

目标机动值的初始值为 $z_0 = 0$, 初始方差为 $P_{z,0} = 10$. RBPF 粒子数为 300; UKF-KF 中的参数 α, β, κ 分别取 1, 2, 0; 仿真时间为 200 s; 仿真初始值取: $x_{0/0} = x_0$, $z_{0/0} = z_0$, $P_{x/x,0} = P_{x,0}$, $P_{z/z,0} = P_{z,0}$. 3 种算法在相同条件下进行 100 次 Monte Carlo 仿真. 仿真软件为 Matlab 7.1(R14), 仿真计算机性能如下: 惠普 Pro 2 000, Intel 酷睿 2 双核 E7500, CPU 主频 2.93 GHz, 内存 2 G. 各滤波器的性能采用平均绝对

值误差进行比较. 第 j 个状态分量在 k 时刻的平均绝对值误差定义为[12]

$$\zeta_j(k)=\frac{1}{L}\sum_{l=1}^{L}|x_{j,k}^{l}-x_{j,k/k}^{l}|, \tag{44}$$

其中: L 为 Monte Carlo 仿真次数, $x_{j,k}^{l}$ 为第 l 次仿真时 k 时刻第 j 个状态分量的真值, $x_{j,k/k}^{l}$ 为估计值.

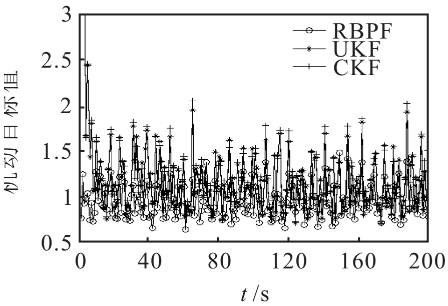

图 1 机动目标估计平均绝对值误差

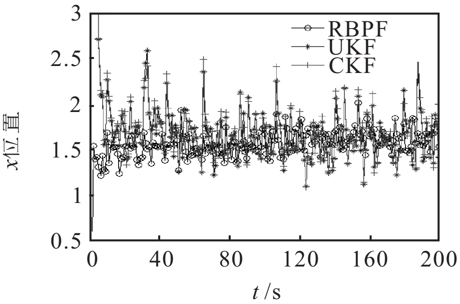

图 2 x 位置估计平均绝对值误差

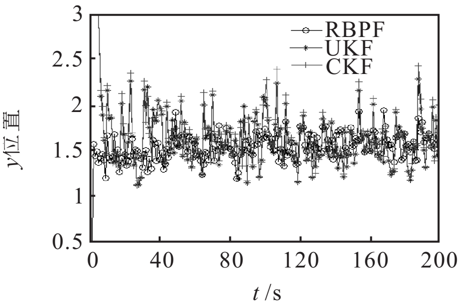

图 3 y 位置估计平均绝对值误差

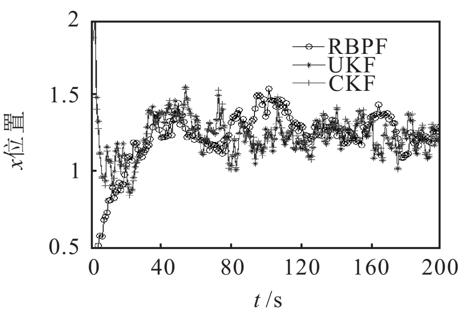

图 4 x 速度估计平均绝对值误差

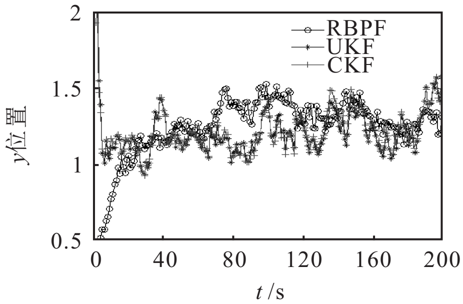

图 5 y 速度估计平均绝对值误差

表 1 状态估计平均绝对值误差的平均值

算法	RBPF	UKF-KF	CKF-KF
机动值	0.942 4	1.233	1.231
x 位置	1.561	1.812	1.818
y 位置	1.534	1.818	1.819
x 速度	1.229	1.244	1.248
y 速度	1.242	1.211	1.219

表 2 100 次 MC 仿真算法平均运行时间 s

算法	RBPF	UKF-KF	CKF-KF
运行时间	12.550 8	0.085 2	0.066 5(22%)

机动目标跟踪仿真结果如图 1 ∼图 5 所示. 状态估计平均绝对值误差的平均值如表 1 所示. 由图 1∼图 5 可以看出, CKF-KF 和 UKF-KF 的估计精度略低于 RBPF, 但算法运行时间不到 RBPF 的 1%, 实时性大幅度提高. 由于 RBPF 沿用了 PF 的框架, 需要大量的粒子对非线性状态后验概率密度进行逼近从而实现状态估计, 导致其算法计算量大, 耗时高. 此外, 在非线性系统估计中, 粒子滤波性能优于其他的高斯滤波器 (UKF, CKF 等). 因此, RBPF 精度高于 CKF-KF 和 UKF-KF, 然而, CKF-KF 和 UKF-KF 只需有限个点进行非线性传播, 且两者对均值和方差的估计都能达到泰勒展开式的三阶精度. 因此, 两者的估计精度相当 (在高斯条件下, UKF 通过调节参数 $\beta=2$ 保证了滤波估计方差的泰勒展开式的四阶项与真实值之间的相对误差最小[13], 而 CKF 无法通过参数调节来保证这种相对误差最小, 因而其四阶项相对误差高于 UKF, 从而本文仿真中 UKF-KF 估计精度稍高一些), 而算法运行时间比 RBPF 低. 同时, 对比 UKF 与 CKF 算法可以看出, UKF 需选择 $2n+1$ 个点进行非线性传播, 而 CKF 只需 $2n$ 个点, 因此 CKF-KF 算法实现时间要低于 UKF-KF, 其实时性更高. 由表 2 可知, 相对于 UKF-KF, CKF-KF 运行时间降低了 22%. 此外, UKF 的算法原理决定其必需调节各参数因子才能选择出有效的 Sigma 点及其权值, 而 CKF 只需通过式 (6) 就能方便地得 cubature 点及其权值, 这就使得 CKF-KF 算法设计更容易. 因此, 在条件线性高斯状态空间模型的估计中, 不论从算法

复杂度还是实时性上来讲, 相对于 RBPF 和 UKF-KF, CKF-KF 都是一种更佳的选择途径.

5 结 论

针对条件线性高斯状态空间模型, 本文提出了 CKF-KF 滤波算法. 该算法利用 CKF 估计模型中的非线性状态, 并利用 KF 估计线性状态. 为了有效融合 CKF 和 KF, 对 KF 估计的线性状态进行采样, 进而与 CKF 中的 cubature 点匹配实现非线性状态估计. 算法原理和机动目标跟踪仿真结果表明, 与 RBPF 相比, 在略失精度的情况下, CKF-KF 算法运行时间节约了 99%. 相对于 UKF-KF, CKF-KF 设计实现更简单, 实时性提高了 22%, 而精度与 UKF-KF 相当. 因此, CKF-KF 更适合处理目标跟踪等对实时性要求较高场合下的条件线性高斯状态估计问题.

参考文献(**References**)

[1] Doucet A, Godsill S, Andrieu C. On sequential Monte Carlo sampling methods for Bayesian filtering[J]. Statistics and Computing, 2000, 10(3): 197-208.

[2] Bar-Shalom Y, Li X R. Multitarget-Multisensor Tracking Principles and Techniques[M]. New Orleans: University of New Orleans, 1995: 40-60.

[3] Mustiere F, Bolic M, Bouchard M. Rao-Blackwellised particle filters: Examples of applications[C]. Canadian Conf on Electrical and Computer Engineering. Ottawa, 2006: 1196-1200.

[4] Matti Vihola. Rao-Blackwellised particle filtering in Random set multitarget tracking[J]. IEEE Trans on Aerospace and Electronic Systems, 2007, 43(2): 689-705.

[5] Nando de Freitas. Rao-Blackwellised particle filtering for fault diagnosis[C]. Proc of IEEE Aerospace Conf. Montana, 2002, 4: 1767-1772.

[6] Arnaud Doucet, Neil J Gordon, Vikram Krishnamurthy. Particle filters for state estimation of jump Markov linear systems[J]. IEEE Trans on Signal Processing, 2001, 49(3): 613-624.

[7] Mustiere F, Bolic M, Bouchard M. A modified Rao-Blackwellised particle filter[C]. IEEE Int Conf on Acoustics, Speech and Signal Processing. Toulouse, 2006, 3: 21-24.

[8] 尹建君, 张建秋, 林青. Unscented 卡尔曼滤波-卡尔曼滤波算法[J]. 系统工程与电子技术, 2008, 30(4): 617-620. (Yin J J, Zhang J Q, Lin Q. Unscented Kalman filter-Kalman filter algorithm[J]. Systems Engineering and Electronics, 2008, 30(4): 617-620.)

[9] Julier S, Uhlmann J, Durrant-Whyte H F. Anew method for the nonlinear transformation of means and covariances in filters and estimators[J]. IEEE Trans on Automatic Control, 2000, 45(3): 477-482.

[10] Arasaratnam I，Haykin S. Cubature Kalman Filter[J]. IEEE Trans on Automatic Control, 2009, 54(6): 1254-1269.

[11] Sanjeev Arulampalam M, Simon Maskell, Neil Gordon, et al. A tutorial on particle filters for online nonlinear/non-gaussian Bayesian tracking[J]. IEEE Trans on Signal Processing, 2002, 50(2): 174-188.

[12] Yuanxin Wu, Dewen Hu, Meiping Wu, et al. A numerical-integration perspective on Gaussian filters[J]. IEEE Trans on Signal Processing, 2006, 54(8): 2910-2921.

[13] Simon J Julier, Jeffrey K Uhlmann. Unscented filtering and nonlinear estimation[J]. Proc of the IEEE, 2004, 92(3): 401-422.