

## 一种简化的强跟踪容积卡尔曼滤波算法

蔡宗平, 张雪影, 牛创, 卫浩  
(火箭军工程大学自动化系, 西安 710025)

**摘要:** 针对强跟踪容积卡尔曼滤波(STCKF)算法因引入渐消因子而导致计算量增加、实时性变差的问题,提出一种简化的STCKF算法。通过证明STCKF算法的时间更新环节与KF算法的一步预测过程相一致,推导出简化的STCKF算法,并进行了算法复杂度分析。仿真结果表明,简化后的STCKF算法在保证滤波精度不变的情况下,有效提高了算法实时性。

**关键词:** 目标跟踪; 强跟踪; 容积卡尔曼滤波; 实时性

**中图分类号:** TP391 **文献标志码:** A **文章编号:** 1671-637X(2017)01-0006-03

## A Simplified Strong Tracking Cubature Kalman Filtering Algorithm

CAI Zong-ping, ZHANG Xue-ying, NIU Chuang, WEI Hao  
(Department of Automation, Rocket Force University of Engineering, Xi'an 710025, China)

**Abstract:** To deal with the problems of the increased calculation complexity and decreased real-time performance due to the introduction of the fading factor in Strong Tracking Cubature Kalman Filter (STCKF) algorithm, a simplified STCKF algorithm is proposed. By proving that the time update of STCKF algorithm is consistent with one-step prediction process of Kalman Filter (KF) algorithm, the simplified STCKF algorithm is derived and the complexity of the algorithm is analyzed. The simulation results show that the simplified STCKF algorithm can effectively improve the real-time performance of the algorithm while keeping the filtering accuracy.

**Key words:** target tracking; strong tracking; Cubature Kalman Filter (CKF); real-time performance

### 0 引言

为解决目标跟踪过程中目标运动模型不确定等问题,文献[1]在扩展卡尔曼滤波(Extended Kalman Filter, EKF)的基础上引入渐消因子,提出了强跟踪滤波器(STF)算法,但同时也使得 STF 算法继承了 EKF 算法精度偏低、计算量大的缺点;文献[2]中利用 Unscented 变换(UT)取代 STF 算法中的雅克比矩阵计算,提出一种基于 UT 的强跟踪滤波器(UTSTF)算法,在一定程度上提高了计算效率;文献[3]提出了一种基于强跟踪 UKF 算法的自适应算法,虽然克服了 STF 算法一些固有的局限性,但 UT 本身对参数化要求高,不利于计算;针对 UKF 算法的不足,ARASARATNAM 等

人<sup>[4]</sup>在三阶-球面径向容积准则基础上提出的容积卡尔曼滤波(Cubature Kalman Filter, CKF)算法,计算量及复杂度较小,且能保证滤波精度和时效性<sup>[5]</sup>;文献[6-7]中考虑将 STF 算法与 CKF 算法相结合,提出了强跟踪 CKF(STCKF)算法,通过引入时变渐消因子来提高算法自适应跟踪的能力,并取得了较好的效果。

由于引入了渐消自适应因子,算法的复杂度有所增加,导致实时性下降。对此,本文在系统状态方程为线性的前提下,对强跟踪 CKF 的时间更新环节进行简化,推导出一种简化的强跟踪 CKF 算法,并对简化后的算法复杂度进行了分析,最后在主动雷达跟踪模型中仿真验证算法的有效性。

### 1 强跟踪 CKF 算法

#### 1.1 强跟踪滤波器

考虑非线性系统为

$$\begin{cases} \mathbf{X}_k = f(\mathbf{X}_{k-1}, u_{k-1}) + \Gamma \mathbf{V}_{k-1} \\ \mathbf{z}_k = h(\mathbf{X}_k, u_k) + \mathbf{W}_k \end{cases} \quad (1)$$

收稿日期:2015-12-29

修回日期:2016-09-29

基金项目:国家自然科学基金(61174207)

作者简介:蔡宗平(1964—),男,陕西岐山人,教授,硕导,研究方向为导航制导与控制。

式中:  $\mathbf{X}_k$  表示状态向量;  $\mathbf{z}_k$  表示量测向量;  $\mathbf{u}_{k-1}$  表示输入控制量;  $f(\cdot)$  和  $h(\cdot)$  分别表示系统的状态函数与量测函数;  $\mathbf{V}_{k-1}$  与  $\mathbf{W}_k$  表示均值为零的高斯白噪声, 且相互独立<sup>[8]</sup>。

强跟踪滤波是一种带有次优渐消因子的拓展卡尔曼滤波<sup>[1]</sup>, 其主要思想是将渐消因子  $\lambda_{k+1}$  代入状态预测协方差阵, 在线调节增益矩阵  $\mathbf{K}_{k+1}$ , 并满足

$$E((\mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1})(\mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1})^T) = \min \quad (2)$$

$$E(\gamma_{k+1+j} \gamma_{k+1}^T) = 0 \quad k=0, 1, \dots, j=1, 2, \dots \quad (3)$$

渐消因子  $\lambda_{k+1}$  按照文献[1]中的算法进行计算, 即

$$\lambda_{k+1} = \begin{cases} \lambda_0 & \lambda_0 \geq 1 \\ 1 & \lambda_0 < 1 \end{cases} \quad (4)$$

式中

$$\lambda_0 = \text{tr}(\mathbf{N}_{k+1}) / \text{tr}(\mathbf{M}_{k+1}) \quad (5)$$

$$\mathbf{N}_{k+1} = \mathbf{V}_{k+1} - \mathbf{H}_{k+1} \mathbf{Q}_{k+1} \mathbf{H}_{k+1}^T - \beta \mathbf{R}_{k+1} \quad (6)$$

$$\mathbf{M}_{k+1} = \mathbf{H}_{k+1} \mathbf{\Phi}_{k+1} \mathbf{P}_k \mathbf{\Phi}_{k+1}^T \mathbf{H}_{k+1}^T \quad (7)$$

式中:  $\beta$  为弱化因子;  $\text{tr}(\cdot)$  为矩阵求迹运算;  $\mathbf{V}_{k+1}$  为输出残差序列协方差阵, 即

$$\mathbf{V}_{k+1} = \begin{cases} \gamma_1 \gamma_1^T & k=0 \\ \rho \mathbf{V}_k + \gamma_k \gamma_k^T & k \geq 1 \end{cases} \quad (8)$$

式中: 输出残差序列为  $\gamma_{k+1} = \mathbf{z}_{k+1} - \hat{\mathbf{z}}_{k+1}$ ;  $0 < \rho \leq 1$  表示遗忘因子, 它能够强化残差序列的影响, 提升滤波算法跟踪能力, 一般取  $\rho = 0.95$ 。

## 1.2 强跟踪 CKF 算法

$\mathbf{N}_{k+1}$  与  $\mathbf{M}_{k+1}$  的等价表达式为

$$\mathbf{N}_{k+1} = \mathbf{V}_{k+1} - \mathbf{H}_{k+1} \mathbf{Q}_k \mathbf{H}_{k+1}^T - \beta \mathbf{R}_{k+1} = \mathbf{V}_{k+1} - \mathbf{P}_{zz}^T (\mathbf{P}_{k+1k})^{-1} \mathbf{Q}_k (\mathbf{P}_{k+1k})^{-1} \mathbf{P}_{zz} - \beta \mathbf{R}_{k+1} \quad (9)$$

$$\begin{aligned} \mathbf{M}_{k+1} &= \mathbf{H}_{k+1} \mathbf{\Phi}_{k+1} \mathbf{P}_k \mathbf{\Phi}_{k+1}^T \mathbf{H}_{k+1}^T = \mathbf{H}_{k+1} (\mathbf{P}_{k+1k} - \mathbf{Q}_k) \cdot \\ \mathbf{H}_{k+1}^T &= \mathbf{H}_{k+1} \mathbf{P}_{k+1k} \mathbf{H}_{k+1}^T - \mathbf{H}_{k+1} \mathbf{Q}_k \mathbf{H}_{k+1}^T = \mathbf{P}_{zz} - \mathbf{V}_{k+1} + \mathbf{N}_{k+1} \end{aligned} \quad (10)$$

将 STF 思想引入 CKF 算法得到强跟踪 CKF 算法, 基本流程如下所述。

1) 时间更新。

$$\hat{\mathbf{x}}_{k|k-1} = \frac{1}{2n} \sum_{i=1}^{2n} \mathbf{X}_{k-1|k-1}^{i*} \quad (11)$$

$$\mathbf{P}_{k|k-1} = \frac{1}{2n} \sum_{i=1}^{2n} \mathbf{X}_{k-1|k-1}^{i*} \mathbf{X}_{k-1|k-1}^{i*\top} - \hat{\mathbf{x}}_{k|k-1} \hat{\mathbf{x}}_{k|k-1}^T + \mathbf{Q}_{k-1} \quad (12)$$

2) 量测更新。

$$\hat{\mathbf{z}}_{k|k-1} = \frac{1}{2n} \sum_{i=1}^{2n} \mathbf{Z}_{k|k-1}^i \quad (13)$$

$$\mathbf{P}_{zz} = \frac{1}{2n} \sum_{i=1}^{2n} \mathbf{Z}_{k|k-1}^i \mathbf{Z}_{k|k-1}^{i\top} - \hat{\mathbf{z}}_{k|k-1} \hat{\mathbf{z}}_{k|k-1}^T + \mathbf{R}_k \quad (14)$$

$$\mathbf{P}_{zx} = \frac{1}{2n} \sum_{i=1}^{2n} \mathbf{X}_{k|k-1}^i \mathbf{Z}_{k|k-1}^{i\top} - \hat{\mathbf{x}}_{k|k-1} \hat{\mathbf{z}}_{k|k-1}^T \quad (15)$$

算得  $\mathbf{P}_{k+1k}$ ,  $\mathbf{P}_{zz}$ ,  $\mathbf{P}_{zx}$  后, 将这些值分别代入式(4)、式(5)、式(8)、式(9)和式(10)可得渐消因子  $\lambda_{k+1}$ , 再将渐消因子代入  $\mathbf{P}_{k+1k}$ , 即

$$\mathbf{P}_{k+1k} = \lambda_{k+1} \mathbf{\Phi} \mathbf{P}_{k|k} \mathbf{\Phi}^T + \mathbf{Q}_k \quad (16)$$

将状态一步预测值  $\hat{\mathbf{x}}_{k+1|k}$  和引入渐消因子后的  $\mathbf{P}_{k+1k}$  代入量测更新环节, 计算加入渐消因子的观测一步预测值  $\hat{\mathbf{z}}_{k+1|k}$ 、自相关协方差阵  $\mathbf{P}_{zz}$  和互相关协方差阵  $\mathbf{P}_{zx}$ , 获得最新的量测数据后更新滤波。

滤波增益

$$\mathbf{W}_k = \mathbf{P}_{zx} \mathbf{P}_{zz}^{-1} \quad (17)$$

状态估计值

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{W}_k (\mathbf{z}_k - \hat{\mathbf{z}}_{k|k-1}) \quad (18)$$

状态协方差估计值

$$\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} - \mathbf{W}_k \mathbf{P}_{zz} \mathbf{W}_k^T \quad (19)$$

## 2 简化的强跟踪 CKF 算法

通常情况下, 目标的动态模型是在笛卡尔坐标系下建模, 状态方程一般为线性<sup>[9]</sup>。本文的简化思路正是基于系统状态方程为线性这个前提展开的。

首先, 对比 STCKF 算法的时间更新环节与 KF 的状态一步预测过程。在 STCKF 算法时间更新环节中, 计算状态一步预测值方程为

$$\begin{aligned} \hat{\mathbf{x}}_{k|k-1} &= \frac{1}{2n} \sum_{i=1}^{2n} \mathbf{X}_{k-1|k-1}^{i*} = \frac{1}{2n} \sum_{i=1}^{2n} \mathbf{\Phi} \mathbf{X}_{k-1|k-1}^i = \\ &\mathbf{\Phi} \hat{\mathbf{x}}_{k-1} + \frac{1}{2n} \mathbf{\Phi} \mathbf{S}_k \sum_{i=1}^{2n} \xi_i \end{aligned} \quad (20)$$

式中,  $\xi = \sqrt{n}[1] = \sqrt{n}[I - I]$ , 可知,  $\sum_{i=1}^{2n} \xi_i = 0$ 。

由此可得

$$\hat{\mathbf{x}}_{k|k-1} = \frac{1}{2n} \sum_{i=1}^{2n} \mathbf{X}_{k-1|k-1}^{i*} = \mathbf{\Phi} \hat{\mathbf{x}}_{k-1} \quad (21)$$

在 KF 算法中, 计算状态一步预测值

$$\hat{\mathbf{X}}_{k|k-1} = \mathbf{\Phi} \hat{\mathbf{X}}_{k-1} \quad (22)$$

对比式(21)和式(22)可以看出, 当状态方程为线性时, STCKF 算法的时间更新环节状态一步预测过程与 KF 算法的状态一步预测过程一致。

然后, 对比误差协方差一步预测过程, STCKF 算法的误差协方差一步预测过程为

$$\begin{aligned} \mathbf{P}_{k|k-1} &= \frac{1}{2n} \sum_{i=1}^{2n} (\mathbf{\Phi} \mathbf{x}_{k|k-1}^i) (\mathbf{\Phi} \mathbf{x}_{k|k-1}^i)^T - \bar{\mathbf{x}}_k \bar{\mathbf{x}}_k^T + \mathbf{Q}_{k-1} = \\ &\frac{1}{2n} \sum_{i=1}^{2n} \mathbf{\Phi} \mathbf{x}_{k|k-1}^i \mathbf{x}_{k|k-1}^{i\top} \mathbf{\Phi}^T - \bar{\mathbf{x}}_k \bar{\mathbf{x}}_k^T + \mathbf{Q}_{k-1} \end{aligned} \quad (23)$$

由式(21)可知,  $[\mathbf{\Phi} \hat{\mathbf{x}}_{k-1}][\mathbf{\Phi} \hat{\mathbf{x}}_{k-1}]^T = \hat{\mathbf{x}}_{k|k-1} \hat{\mathbf{x}}_{k|k-1}^T$  及  $\sum_{i=1}^{2n} \xi_i = 0$ , 代入式(23)并化简可得

$$\mathbf{P}_{k|k-1} = \mathbf{\Phi} \mathbf{S}_k \mathbf{S}_k^T \mathbf{\Phi}^T + \mathbf{Q}_{k-1} \quad (24)$$

在 KF 算法里,状态误差协方差一步预测求解方程为

$$P_{k|k-1} = \Phi P_{k-1|k-1} \Phi^T + Q_{k-1} \quad (25)$$

可以看出,式(24)与式(25)完全一样。根据式(20)~式(25)的推导过程可知,当系统状态方程是线性时,STCKF 算法中时间更新过程的状态协方差一步预测方程与 KF 算法相同。

考虑到 KF 算法的计算量较小,当系统满足状态方程是线性时,可将 STCKF 算法时间更新过程里的一步预测公式与 KF 算法中的一步预测公式相替换。

由此可以得出,简化后的 STCKF 算法基本过程如下所述。

1) 时间更新。计算状态一步预测值

$$\hat{x}_{k+1|k} = \Phi \hat{x}_{k|k} \quad (26)$$

状态误差协方差一步预测值

$$P_{k+1|k} = \Phi P_{k|k} \Phi^T + Q_k \quad (27)$$

2) 量测更新。计算式(13)~式(19)。

### 3 算法复杂度分析

对于  $n$  维状态变量的系统,分别用  $c_1, c_2, c_3$  表示数乘运算、加减运算和矩阵分解运算的计算量,忽略矩阵的转置运算时间,考虑到简化后算法的量测更新环节并未发生变化,本文只对 STCKF 算法与简化 STCKF 算法的时间更新环节运算成本进行比较,结果如表 1 所示。

表 1 算法运算成本比较

Table 1 Comparison of calculating complexity

算法	运算成本
STCKF	$(4n^3 + 3n^2 + n)c_1 + (6n^3 + 6n^2)c_2 + 2nc_3$
简化 STCKF	$(2n^3 + n^2)c_1 + (2n^3 + 2n^2)c_2$

根据表 1 的比较结果能够看出,由于简化后的 STCKF 算法无需对容积点进行计算,因此运算成本要低于 STCKF 算法。同时,在时间更新环节计算状态误差协方差的步骤中,STCKF 算法需要做乔列斯基分解运算,但简化后的 STCKF 算法无需做矩阵分解,因而计算效率提高。又由于只是对时间更新环节的计算步骤做了合理简化,在状态方程为线性的前提下,省略了非必要的过程,因此,在理论上其滤波精度将不会发生变化。

### 4 仿真结果与分析

为验证简化 STCKF 算法的滤波效果,采用主动雷达跟踪模型进行仿真,状态初值  $x_0 = (2000, 50, 0, 1500, 50, 0)^T$ , 误差协方差初值  $P_{00} = \text{diag}(10^2, 3^2, 0.5^2, 10^2, 3^2, 0.5^2)$ , 系统噪声方差  $Q = \text{diag}(1^2, 1^2)$ , 量测噪声方差  $R_k = \text{diag}(\sigma_r^2, \sigma_\theta^2)$ ,  $\sigma_r = 10 \text{ m}$ ,  $\sigma_\theta = \sqrt{10} \text{ mrad}$ 。

假设目标在 0~30 s 保持匀速运动,在 30~40 s 给目

标一个 X 方向的加速度,  $(a_x, a_y)$  为  $(15 \text{ m/s}^2, 0 \text{ m/s}^2)$ , 在 40~60 s 恢复匀速运动,在 60~70 s 给目标一个 Y 方向的加速度,  $(a_x, a_y)$  为  $(0 \text{ m/s}^2, 15 \text{ m/s}^2)$ , 70 s 以后恢复匀速运动。遗忘因子  $\rho$  为 0.95, 弱化因子  $\beta$  为 1.5。

分别用 CKF 算法、STCKF 算法与简化的 STCKF 算法进行对比仿真,采用均方根误差评价算法性能,进行 100 次蒙特卡罗仿真实验。

为了观察方便起见,同时给出第二次 Y 方向加速时的局部轨迹图,其跟踪轨迹及 RMSE 如图 1~图 3 所示。

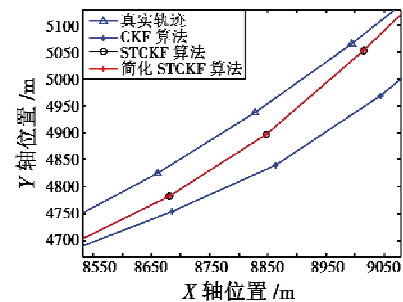


图 1 跟踪轨迹局部图

Fig. 1 Local image of tracking trajectory

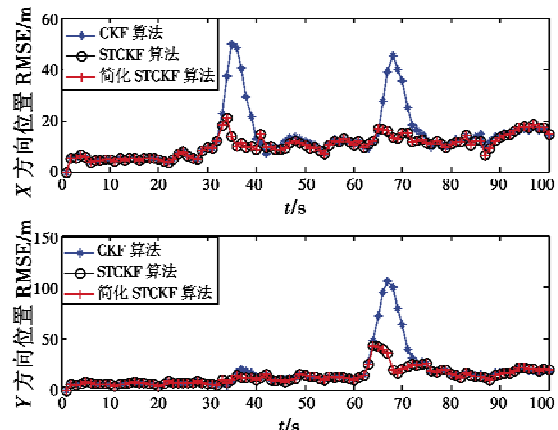


图 2 位置 RMSE

Fig. 2 RMSE of position

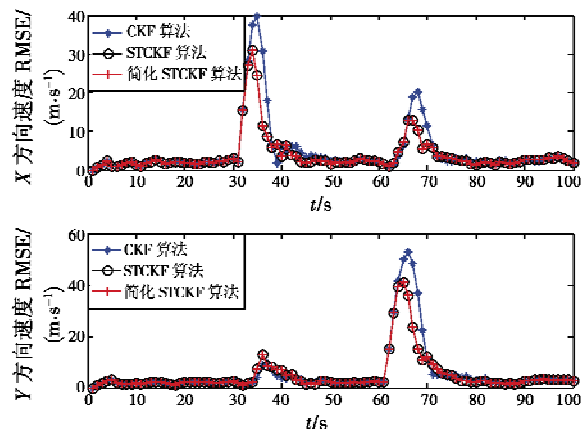


图 3 速度 RMSE

Fig. 3 RMSE of velocity

- compared to state-of-the-art superpixel methods [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2012, 34(11):2274-2282.
- [5] RADHAKRISHNA A, SHAJI A, SMITH K, et al. Slic superpixels[R]. Technical Report 149300, EPFL, 2010.
- [6] YANG F, LU H C, YANG M H. Robust superpixel tracking [J]. IEEE Transactions on Image Processing, 2014, 23(4):1639-1651.
- [7] KWON J, JUNHA R, LEE K M, et al. Robust visual tracking with double bounding box model [C]//European Conference on Computer Vision, 2014:377-392.
- [8] NING J, ZHANG L, ZHANG D, et al. Robust mean shift tracking with corrected background-weighted histogram [J]. IET Computer Vision, 2012, 6(1):62-69.
- [9] DIETTERICH T G, LATHROP R H, LOZANO-PEREZ T. Solving the multiple instance problem with axis-parallel rectangles [J]. Artificial Intelligence, 1997, 89(1):31-71.
- [10] BABENKO B, BELONGIE S, YANG M H. Visual tracking with online multiple instance learning [C]//IEEE Conference on Computer Vision and Pattern Recognition, 2009:1003-1010.
- [11] CHEN D C, ZHU M, GAO W, et al. Real-time object tracking via online weighted multiple instance learning [J]. Optics and Precision Engineering, 2014, 22(6):1661-1667.
- [12] WU J J, ZHAO Y B, ZHU J Y, et al. MILCut: a sweeping line multiple instance learning paradigm for interactive image segmentation [C]//IEEE Conference on Computer Vision and Pattern Recognition, 2014:256-263.
- [13] VIOLA P A, PLATT J C, ZHANG C. Multiple instance boosting for object detection [C]//Advances in Neural Information Processing Systems, 2005:1417-1426.
- [14] GRABNER H, GRABNER M, BISCHOF H. Real-time tracking via on-line boosting [C]//British Machine Vision Conference, 2006:45-56.
- [15] DANELLJAN M, KHAN F S, FELSBERG M, et al. Adaptive color attributes for real-time visual tracking [C]//IEEE Conference on Computer Vision and Pattern Recognition, 2014:65-72.

(上接第 8 页)

各算法的 RMSE 计算结果如表 2 所示。

表 2 RMSE 比较 (取自然对数)

Table 2 Performance comparison (natural logarithm)

统计特性	CKF		STCKF		简化 STCKF	
	位置/ m	速度/ (m·s <sup>-1</sup> )	位置/ m	速度/ (m·s <sup>-1</sup> )	位置/ m	速度/ (m·s <sup>-1</sup> )
均值	2.72	1.65	2.46	1.38	2.46	1.38
方差	5.26	4.28	3.51	3.61	3.51	3.61

3 种算法的运行时间如表 3 所示。

表 3 运行时间比较

Table 3 Running time comparison ms

	CKF	STCKF	简化 STCKF
运行时间	426.1	526.2	457.8

对仿真结果进行分析:

1) 从图 1~图 3 及表 2 可知,在目标发生突发机动的情况下,两种滤波算法的精度及稳定性均明显优于 CKF 算法,简化的 STCKF 算法与 STCKF 算法滤波效果一致,表明对 STCKF 算法的简化没有影响原算法的滤波效果;

2) 从表 3 中看出,由于引入了渐消因子,强跟踪 CKF 算法运行时间高于 CKF 算法,而由于简化的 STCKF 算法对时间更新环节进行了简化,算法运行时间明显减少,在保证算法精度及稳定性的同时提高了实时性。

## 5 结束语

针对强跟踪 CKF 算法由于引入渐消因子的计算环

节而导致算法实时性下降的问题,提出了一种简化方法。首先研究了强跟踪 CKF 算法,在系统状态方程通常为线性的前提下,证明了强跟踪 CKF 算法的时间更新环节与 KF 算法的一步预测过程相一致,以此为基础,推导出一种简化的强跟踪 CKF 算法,分析了算法复杂度,进行了实验仿真。仿真结果表明,简化后的算法在保持滤波精度与稳定性的同时,实时性有了明显改善。

## 参考文献

- [1] 周东华,席裕庚,张钟俊.一种带多重次优渐消因子的扩展卡尔曼滤波器[J].自动化学报,1991,17(6):689-695.
- [2] 王小旭,赵琳,夏全喜,等.基于 Unscented 变换的强跟踪滤波器[J].控制与决策,2010,25(7):1063-1068.
- [3] 张文玲,朱明清,陈宗海.基于强跟踪 UKF 的自适应 SLAM 算法[J].机器人,2010,32(2):190-195.
- [4] ARASARATNAM I, HAYKIN S. Cubature Kalman filters [J]. IEEE Transactions on Automatic Control, 2009, 54(6):1254-1269.
- [5] 张鑫春,郭承军.均方根嵌入式容积卡尔曼滤波[J].控制理论与应用,2013,30(9):1116-1121.
- [6] 霍光,李冬海,李晶.基于强跟踪容积卡尔曼滤波的单站无源跟踪算法[J].现代雷达,2013,35(11):52-57.
- [7] 刘万利,张秋昭.基于 Cubature 卡尔曼滤波的强跟踪滤波算法[J].系统仿真学报,2014,26(5):1102-1107.
- [8] 何友,修建娟,张晶炜,等.雷达数据处理及应用[M].北京:电子工业出版社,2006.
- [9] 蔡宗平,牛创,戴定成.基于简化 CKF 的 IMM 算法[J].现代防御技术,2015,43(6):111-115.