Nonlinear Estimation Techniques Applied on Target Tracking Problems

Andrew Gadsden
Department of Mechanical Engineering,
McMaster University,
1280 Main Street West,
Hamilton, ON, L8S 4L7, Canada
e-mail: gadsden@mcmaster.ca

Saeid Habibi
Department of Electrical and Computer Engineering,
McMaster University
1280 Main Street West,
Hamilton, ON, L8S 4L7, Canada
e-mail: habibi@mcmaster.ca

Darcy Dunne
Department of Mechanical Engineering,
McMaster University,
1280 Main Street West,
Hamilton, ON, L8S 4L7, Canada
e-mail: ddunne@mcmaster.ca

T. Kirubarajan
Department of Electrical and Computer Engineering,
McMaster University,
1280 Main Street West,
Hamilton, ON, L8S 4L7, Canada
e-mail: kiruba@mcmaster.ca

This paper discusses the application of four nonlinear estimation techniques on two benchmark target tracking problems. The first problem is a generic air traffic control (ATC) scenario, which involves nonlinear system equations with linear measurements. The second study is a classical ground surveillance problem, where a moving airborne platform with a sensor is used to track a moving target. The tracking scenario is set in two dimensions, with the measurement providing nonlinear bearing-only observations. These two target tracking problems provide a good benchmark for comparing the following nonlinear estimation techniques: the common extended and unscented Kalman filters (EKF/UKF), the particle filter (PF), and the relatively new smooth variable structure filter (SVSF). The results of applying the SVSF on the two target tracking problems demonstrate its stability and robustness. Both of these attributes make use of the SVSF advantageous over other popular methods. The filters performances are quantified in terms of robustness, resilience to poor initial conditions and measurement outliers, and tracking accuracy and computational complexity. The purpose of this paper is to demonstrate the effectiveness of applying the SVSF on nonlinear target tracking problems, which in the past have typically been solved by Kalman or particle filters.
[DOI: 10.1115/1.4006374]

Keywords: estimation, target tracking, Kalman filter, particle filter, smooth variable structure filter, sliding mode estimation

1 Introduction

In the estimation world, even after 50 years, the Kalman filter (KF) method remains the most studied and one of the most popular tools used to estimate states from systems [1–3]. It may be applied on linear dynamic systems in the presence of Gaussian white noise, and it provides an elegant and statistically optimal solution by minimizing the mean-squared estimation error.

In practice, all systems in nature are in fact nonlinear, such that linear estimation techniques may not be used to provide optimal solutions. However, suboptimal techniques may be applied to handle the nonlinearities. Such techniques include the extended and unscented Kalman filters, the particle filter, and the smooth variable structure filter. The EKF is a popular extension of the KF and is commonly used in target tracking [4]. It uses partial derivatives of the nonlinearities in the state dynamic and measurement models, such that linearized approximations are obtained and then used in the estimation process [2,4]. The UKF is different from the EKF in the sense that it does not approximate any of the nonlinear functions [4]. The UKF approximates the posterior distribution of the states, using a set of deterministically chosen sample points, which, after a transformation, capture the true mean and covariance up to the second order of nonlinearity [4]. The PF takes the Bayesian approach to dynamic state estimation, in which one attempts to accurately represent the probability distribution function of the values of interest [4]. The SVSF is a relatively new predictor–corrector method based on sliding mode concepts used for state and parameter estimation [5,6].

In target tracking applications, one may be concerned with surveillance, guidance, obstacle avoidance, or tracking a target given some measurements [4]. In a typical scenario, sensors provide a signal that is processed and output as a measurement. These measurements are related to the target state, and are typically noise-corrupted observations [4]. The target state usually consists of kinematic information such as position, velocity, and acceleration. The measurements are processed in order to form and maintain tracks, which are a sequence of target state estimates that vary with time [4]. Gating and data association techniques help classify the source of measurements, and help associate measurements to the appropriate track [4]. Typically, these gating techniques help to avoid extraneous measurements which would otherwise cause the estimation process to go unstable and fail. A tracking filter is used in a recursive manner to carry out the target state estimation.

2 State Estimation

State and parameter estimation techniques are quite useful for systems when not all of the dynamics are known. Estimation theory involves finding a value of some parameter of interest, which affects the output of the system, often in the presence of inaccurate or uncertain observations [3]. States are representative of the dynamics of a system. For example, for space vehicles, inertial measuring units may be used to calculate the acceleration. However, since their alignment deteriorates over time, calculating the acceleration by other means (i.e., state estimation) may be desirable [7].

The purpose of estimation, as described by Bar-Shalom et al. [3], can be one of many reasons: determination of planet orbit parameters, statistical inference, aircraft traffic control system (i.e., tracking), use in control plants with uncertainties (i.e., parameter identification or state estimation), determination of model parameters (i.e., system identification), message retrieval from noisy signals (i.e., communication theory), and also signal and image processing. A filter may be used to estimate the state of a dynamic system, whether linear or nonlinear. The word filter is used because when finding the best estimate, one has to filter out the noisy signals or uncertain observations [3]. In this paper, four filters (the commonly used EKF, UKF, PF, and the relatively new SVSF) are applied on two target tracking problems, and the performances in terms of robustness, stability, and accuracy are compared.

2.1 Kalman and Extended Kalman Filters. As previously mentioned, the KF may be applied on linear dynamic systems in the

Contributed by the Dynamic Systems Division of ASME for publication in the Journal of Dynamic Systems, Measurement, and Control. Manuscript received October 5, 2009; final manuscript received March 4, 2012; published online June 5, 2012. Assoc. Editor: Swaroop Darbha.

presence of Gaussian white noise, and provides an elegant and statistically optimal solution by minimizing the mean-squared estimation error. However, in the presence of nonlinearities, one may implement the extended form (i.e., EKF). Conceptually, the EKF is very similar to the standard KF. The nonlinear system and measurement functions (f and h, respectively) are used to predict the state estimates and predicted measurements. However, it is not possible to use these functions to calculate the predicted and updated state error covariance matrices. The EKF requires that the functions f and h be linearized (as per its Jacobian). Although this allows the KF to handle mildly nonlinear estimation problems, it introduces a number of instabilities [8]. For example, the linearization process may overlook unmodeled nonlinear modeling uncertainties, which may cause the estimate to go unstable [2]. Furthermore, the calculation of the Jacobian increases the computational complexity of the filter. The partial derivatives are used to compute linearized system and measurement matrices F and H, respectively, found as follows [2]:

$$F_k = \left.\frac{\partial f(x)}{\partial x}\right|_{x=\hat{x}_{k|k},u_k} \tag{2.1.1}$$

$$H_{k+1} = \left.\frac{\partial h(x)}{\partial x}\right|_{x=\hat{x}_{k+1|k}} \tag{2.1.2}$$

Equations (2.1.1) and (2.1.2) essentially linearize the nonlinear system or measurement functions around the current state estimate [9]. This comes at a loss of optimality, as the KF gain is no longer considered to be the best solution to the estimation problem [10]. The EKF process may be summarized by Eqs. (2.1.3)–(2.1.9). The state estimate $\hat{x}_{k+1|k}$ is predicted using the nonlinear system model (2.1.3), and the corresponding state error covariance matrix $P_{k+1|k}$ is found in Eq. (2.1.4).

$$\hat{x}_{k+1|k} = f\left(\hat{x}_{k|k}, u_k\right) \tag{2.1.3}$$

$$P_{k+1|k} = F_k P_{k|k} F_k^T + Q_k \tag{2.1.4}$$

The measurement error (or innovation) $\tilde{y}_{k+1}$ is then found in Eq. (2.1.5), based on the nonlinear measurement model h, followed by the measurement error (innovation) covariance matrix S_{k+1} (2.1.6).

$$\tilde{y}_{k+1} = z_{k+1} - h\left(\hat{x}_{k+1|k}\right) \tag{2.1.5}$$

$$S_{k+1} = H_{k+1} P_{k+1|k} H_{k+1}^T + R_{k+1} \tag{2.1.6}$$

The near-optimal KF gain K_{k+1} is calculated by Eq. (2.1.7). This gain is then used in conjunction with the predicted state estimate $\hat{x}_{k+1}$ and the measurement error $\tilde{y}_{k+1}$ to update the state estimate (2.1.8).

$$K_{k+1} = P_{k+1|k} H_{k+1}^T S_{k+1}^{-1} \tag{2.1.7}$$

$$\hat{x}_{k+1|k+1} = \hat{x}_{k+1|k} + K_{k+1}\tilde{y}_{k+1} \tag{2.1.8}$$

Finally, the state error covariance matrix is updated as per Eq. (2.1.9).

$$P_{k+1|k+1} = (I - K_{k+1}H_{k+1})P_{k+1|k} \tag{2.1.9}$$

Equations (2.1.1)–(2.1.9) form the EKF estimation process. The linearization process of Eqs. (2.1.1) and (2.1.2) introduces uncertainties that can sometimes cause the filter to go unstable [11]. However, for mildly nonlinear systems, the EKF provides a very good estimate of the states, and is relatively easy to implement [12].

2.2 Unscented Kalman Filter. The next major progression of KF theory involved the introduction of the sigma point Kalman filter (SPKF) [12]. The SPKF is based on a weighted statistical linear regression strategy which linearizes the nonlinear model statistically [13,14]. Essentially, SPKF methods draw a certain number of points (referred to as sigma points) from the projected probability distribution of the states [15]. These points are then projected using the nonlinear system model, in an effort to obtain an a posteriori estimate for the probability distribution. Note that this strategy avoids the requirement of linearization, which generally leads to a more accurate estimation strategy since it avoids the calculation of Jacobian matrices [16,17]. The most popular type of SPKF is the UKF [18,19]. A number of different forms exist, and include [15] the unscented [12,19], general unscented [12,16], simplex unscented [12,17,20,21], and spherical unscented [12,21]. The standard UKF method will be presented and discussed in this paper [19]. The UKF strategy makes use of a deterministic sampling technique referred to as the unscented transform. It is well established in literature that this method provides a more accurate estimate of the state mean and covariance than the EKF [2]. As shown in Fig. 1, a finite number of weighted sample points (selected about the mean) are propagated through the nonlinear functions, which create an approximate solution to the mean and covariance of the desired estimate [18,19].

The following equations help summarize the UKF estimation method [19]. The first step to applying the UKF is to generate the sigma points. The n-dimensional random variable x_k with mean $\hat{x}_{k|k}$ and covariance $P_{k|k}$ may be approximated by $(2n+1)$ sigma points. The initial sigma points (corresponding sample and weight) may be calculated as follows:

$$X_{0,k|k} = \hat{x}_{k|k} \tag{2.2.1}$$

$$W_0 = k/(n+k) \tag{2.2.2}$$

The next n number of sigma points may be calculated as follows:

$$X_{i,k|k} = \hat{x}_{k|k} + \left(\sqrt{(n+k)P_{k|k}}\right)_i \tag{2.2.3}$$

$$W_i = 1/[2(n+k)] \tag{2.2.4}$$

Likewise, the remaining n number of sigma points may be found as

$$X_{i+n,k|k} = \hat{x}_{k|k} - \left(\sqrt{(n+k)P_{k|k}}\right)_i \tag{2.2.5}$$

$$W_{i+n} = 1/[2(n+k)] \tag{2.2.6}$$

The parameter k is a design value (typically a small value, significantly less than 1), $\left(\sqrt{(n+k)P_{k|k}}\right)_i$ is the ith row or column of the matrix square root of $((n+k)P_{k|k})$, and W_i is the weight that is associated with the ith sample point [18]. The sigma points are then propagated through the nonlinear system model (2.2.7), and then are used with their corresponding weights to calculate the predicted state estimate (2.2.8).

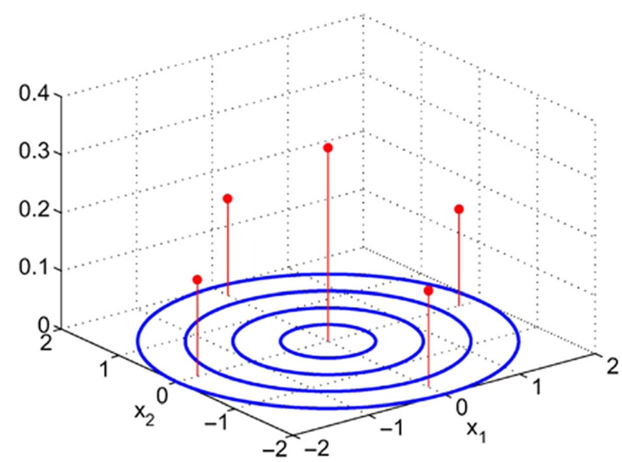

Fig. 1 Distribution of sigma point set for the UKF in 2D space [22]

$$\hat{X}_{i,k+1|k} = f\left(X_{i,k|k}, u_k\right) \tag{2.2.7}$$

$$\hat{x}_{k+1|k} = \sum_{i=0}^{2n} W_i \hat{X}_{i,k+1|k} \tag{2.2.8}$$

From Eqs. (2.2.7) and (2.2.8), it is possible to calculate the predicted state error covariance as follows:

$$P_{k+1|k} = \sum_{i=0}^{2n} W_i \left(\hat{X}_{i,k+1|k} - \hat{x}_{k+1|k}\right)\left(\hat{X}_{i,k+1|k} - \hat{x}_{k+1|k}\right)^T \tag{2.2.9}$$

Next, the sigma points are propagated through the nonlinear measurement model (2.2.10), and the predicted measurement is calculated by Eq. (2.2.11).

$$\hat{Z}_{i,k+1|k} = h\left(\hat{X}_{i,k+1|k}, u_k\right) \tag{2.2.10}$$

$$\hat{z}_{k+1|k} = \sum_{i=0}^{2n} W_i \hat{Z}_{i,k+1|k} \tag{2.2.11}$$

The measurement (or innovation) covariance is then calculated as follows:

$$P_{zz,k+1|k} = \sum_{i=0}^{2n} W_i \left(\hat{Z}_{i,k+1|k} - \hat{z}_{k+1|k}\right)\left(\hat{Z}_{i,k+1|k} - \hat{z}_{k+1|k}\right)^T \tag{2.2.12}$$

Likewise, the cross-covariance (between the state and the measurement) is then calculated as follows:

$$P_{xz,k+1|k} = \sum_{i=0}^{2n} W_i \left(\hat{X}_{i,k+1|k} - \hat{x}_{k+1|k}\right)\left(\hat{Z}_{i,k+1|k} - \hat{z}_{k+1|k}\right)^T \tag{2.2.13}$$

From Eqs. (2.2.12) and (2.2.13), the Kalman gain K_{k+1} may be calculated by Eq. (2.2.14).

$$K_{k+1} = P_{xz,k+1|k} P_{zz,k+1|k}^{-1} \tag{2.2.14}$$

The remaining UKF process is conceptually similar to the standard KF or EKF. The updated (or a posteriori) state estimate may be calculated by Eq. (2.2.15), and the updated state error covariance may be found in Eq. (2.2.16).

$$\hat{x}_{k+1|k+1} = \hat{x}_{k+1|k} + K_{k+1}\left(z_{k+1} - \hat{z}_{k+1|k}\right) \tag{2.2.15}$$

$$P_{k+1|k+1} = P_{k+1|k} - K_{k+1} P_{zz,k+1|k} K_{k+1}^T \tag{2.2.16}$$

By the nature of its derivation, the UKF may appear to be more computationally demanding than the EKF. However, both methods are roughly the same, as the UKF does not require linearization of the nonlinear functions [2]. For mildly nonlinear estimation problems, both the EKF and the UKF will yield the same solution [23]. However, the UKF becomes more advantageous when the nonlinearities are increased.

2.3 Particle Filter.

The PF has many names: Monte Carlo filters, interacting particle approximations [24], bootstrap filters [25], condensation algorithm [26], and survival of the fittest [27], to name a few. Particle filters perform sequential Monte Carlo (SMC) estimation based on weighted particles [4]. As mentioned in Ref. [4], the basic concepts of SMC were introduced in statistics literature in the 1950s [28]. The earliest implementations of the PF were formed on the principles of sequential importance sampling (SIS) [29–31]. The main idea behind the SIS strategy is to represent the desired posterior density function by a number of weighted samples, and then compute estimates based on these samples [4]. The larger the number of samples, the more accurate the representation; however, this comes at the cost of computational demand and time.

The following equations help to describe the SIS strategy. The expectation of some function $f(\cdot)$ can be approximated as a weighted average, as follows:

$$\int f(x_k) p(x_k|z_0, \ldots, z_k) \approx \sum_{i=1}^{P} \omega^i f(x_k^i) \tag{2.3.1}$$

where $p(x_k|z_0, \ldots, z_k)$ refers to the desired distribution, ω^i refers to the weight of the ith particle, x_k^i refers to the ith particle at time k, and P is the total number of particles. The first step of the process involves drawing the particles from the proposal distribution q (i.e., typically the system function f) given the current measurements and the particles from the previous time step (or initialization)

$$x_k^i \sim q\left(x_k|x_{k-1}^i, z_k\right) \tag{2.3.2}$$

The next step involves calculating the corresponding particle weights, which are then used to approximate the desired distribution. The weights may be calculated using the likelihood function $p\left(z_k|x_k^i\right)$, which is defined by the measurement model h and the measurement covariance R_k.

$$\widehat{\omega}_k^i \sim p\left(z_k|x_k^i\right) \cdot \omega_{k-1}^i \tag{2.3.3}$$

The particle weights are then normalized as follows, such that the sum of the particle weights is equal to unity (i.e., $\sum_i^P \omega_k^i = 1$).

$$\omega_k^i = \frac{\widehat{\omega}_k^i}{\sum_{j=1}^{P} \widehat{\omega}_k^j} \tag{2.3.4}$$

Finally, the state(s) may be estimated by the weighted summation of the particles, as follows:

$$\hat{x}_k = \sum_{i=1}^{P} \omega_k^i x_k^i \tag{2.3.5}$$

Equations (2.3.2)–(2.3.5) summarize the SIS strategy, which is the basic form of the PF [4]. This method is relatively straightforward and yields good results. However, it was discovered that the SIS implementation suffers from a numerical phenomenon referred to as degeneration [25,32]. The degeneracy problem refers to only a few particles having significant importance weights after a large number of recursions. An important step referred to as resampling was added after Eq. (2.3.4), and helps to avoid the degeneracy problem [25]. This step eliminates the particles with low weights and multiplies with high weights [4]. The concept of effective sample size N_{eff} was introduced to help measure the amount of degeneracy present in the algorithm [33]. Should a significant amount of degeneracy be detected (below a designer threshold) as per Eq. (2.3.6), then the particles should be resampled.

$$N_{\text{eff}} = \frac{1}{\sum_{i=1}^{P} \left(\omega_k^i\right)^2} \tag{2.3.6}$$

Although resampling helps to avoid the degeneracy problem, it also decreases the amount of particle diversity (i.e., particles with significant weights will be resampled) [4]. This is referred to as sample impoverishment. However, resampling is known to increase the overall accuracy of the estimated state(s) [25]. The aforementioned strategy is referred to as sampling importance resampling, and forms the standard PF used in literature.

2.4 The Smooth Variable Structure Filter.

In 2002, the variable structure filter (VSF) was introduced as a new predictor–corrector method used for state and parameter estimation [5,6]. It is a type of sliding mode estimator, where gain switching is used to ensure that the estimates converge to true state values. An internal model of the system, either linear or nonlinear, is used to predict an a priori state estimate. A corrective term is then applied to calculate the a posteriori state estimate, and the estimation process is repeated iteratively. The SVSF was later derived from the VSF, and used a simpler and less complex gain calculation [34]. In its present form, the SVSF is stable and robust to modeling uncertainties and noise, given an upper bound [34]. The basic concept of the SVSF estimation strategy is shown in Fig. 2.

Assume that the solid line in Fig. 2 is a trajectory of some state (amplitude versus time). An initial value is selected for the state estimate. The estimated state is pushed toward the true value. Once the value enters the existence subspace, the estimated state is forced into switching along the system state trajectory [12]. Consider the following process for the SVSF estimation strategy, as applied to a nonlinear system with a linear measurement equation. The predicted state estimates $\hat{x}_{k+1|k}$ are first calculated as follows:

$$\hat{x}_{k+1|k} = \hat{f}\left(\hat{x}_{k|k}, u_k\right) \tag{2.4.1}$$

Utilizing the predicted state estimates $\hat{x}_{k+1|k}$, the corresponding predicted measurements $\hat{z}_{k+1|k}$ and the measurement error vector $e_{z,k+1|k}$ may be calculated

$$\hat{z}_{k+1|k} = C\hat{x}_{k+1|k} \tag{2.4.2}$$

$$e_{z,k+1|k} = z_{k+1} - \hat{z}_{k+1|k} \tag{2.4.3}$$

Next, the SVSF gain K_{k+1} is calculated as follows [34]:

$$K_{k+1} = C^{+}\left(\left|e_{z,k+1|k}\right|_{\text{Abs}} + \gamma\left|e_{z,k|k}\right|_{\text{Abs}}\right) \circ \text{sat}\left(\frac{e_{z,k+1|k}}{\psi}\right) \tag{2.4.4}$$

The SVSF gain is a function of the a priori and a posteriori measurement error vectors $e_{z,k+1|k}$ and $e_{z,k|k}$, the smoothing boundary layer widths ψ, the "SVSF" memory or convergence rate γ with elements $0 < \gamma_{ii} \leq 1$, and the linear measurement matrix C. The SVSF gain is used to refine the state estimates as follows:

$$\hat{x}_{k+1|k+1} = \hat{x}_{k+1|k} + K_{k+1} \tag{2.4.5}$$

The updated measurement estimates $\hat{z}_{k+1|k+1}$ and corresponding errors $e_{z,\,k+1|k+1}$ are then calculated

$$\hat{z}_{k+1|k+1} = C\hat{x}_{k+1|k+1} \tag{2.4.6}$$

$$e_{z,k+1|k+1} = z_{k+1} - \hat{z}_{k+1|k+1} \tag{2.4.7}$$

The SVSF process may be summarized by Eqs. (2.4.1)–(2.4.7), and is repeated iteratively. According to Ref. [34], the estimation process is stable and converges to the existence subspace if the following condition is satisfied

$$\left|e_{k|k}\right|_{\text{Abs}} > \left|e_{k+1|k+1}\right|_{\text{Abs}} \tag{2.4.8}$$

Note that $|e|_{\text{Abs}}$ is the absolute of the vector e, and is equal to $|e|_{\text{Abs}} = e \cdot \text{sign}(e)$. The proof, as described in Refs. [15] and [34], yields the derivation of the SVSF gain from Eq. (2.4.8).

3 Nonlinear Target Tracking Problems

This section describes two target tracking problems that were used as benchmarks for comparing four nonlinear estimation techniques.

3.1 Air Traffic Control Scenario. The first target tracking problem is based on a generic ATC scenario found in Ref. [3]. Radar stationed at the origin provides direct position only measurements, with a very large standard deviation of 1000 m in each coordinate. As shown in Fig. 3, an aircraft starts from an initial position of (25,000 m, 10,000 m) at time $t = 0$ s, and flies westward at 120 m/s for125 s. The aircraft then begins a coordinated turn for a period of 90 s at a rate of 1 deg/s. It then flies southward at 120 m/s for125 s, followed by another coordinated turn for 30 s at 3 deg/s. Finally, it continues to fly westward.

In ATC scenarios, the behavior of civilian aircraft may be modeled by two different modes: uniform motion (UM), which involves a straight flight path with a constant speed and course, and maneuvering which includes turning or climbing and descending [3]. In this case, maneuvering will refer to a coordinated turn (CT) model, where a turn is made at a constant turn rate and speed. The uniform motion model used for this target tracking problem is given by Eq. (3.1.1) [3,35]

$$x_{k+1} = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} x_k + \begin{bmatrix} \frac{1}{2}T^2 & 0 \\ 0 & \frac{1}{2}T^2 \\ T & 0 \\ 0 & T \end{bmatrix} w_k \tag{3.1.1}$$

The state vector of the aircraft may be defined as follows:

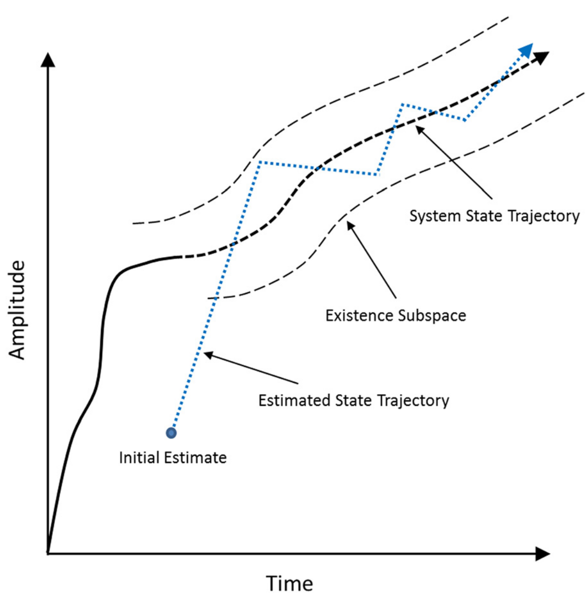

Fig. 2 SVSF estimation concept [8]

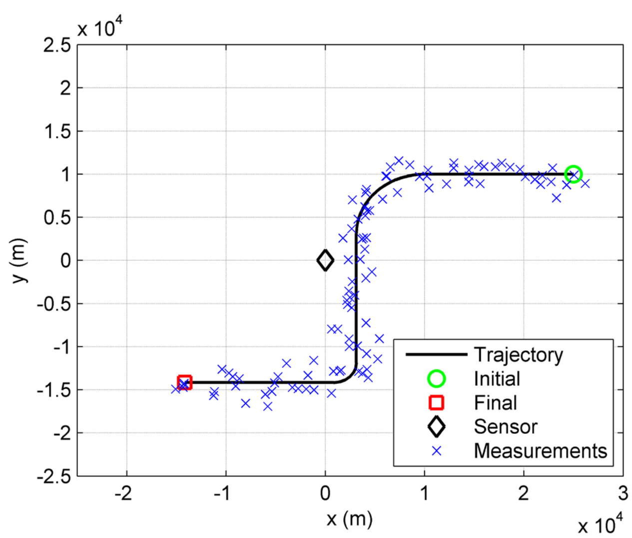

Fig. 3 Aircraft trajectory

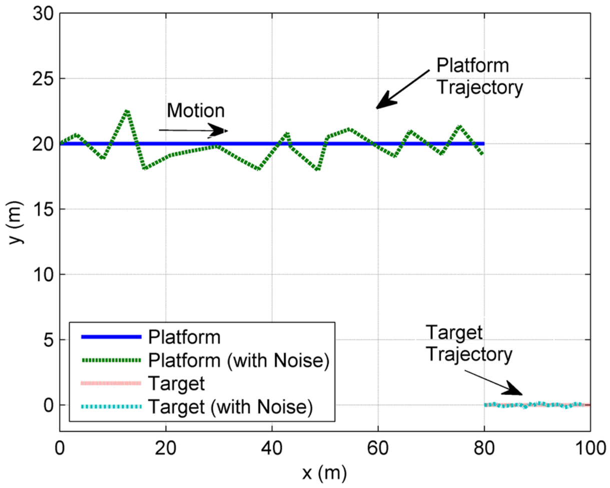

Fig. 4 Platform and target trajectory

$$x_k = \begin{bmatrix} \xi_k & \eta_k & \dot{\xi}_k & \dot{\eta}_k \end{bmatrix}^T \tag{3.1.2}$$

The first two states refer to the position along the x-axis and y-axis, respectively, and the last two states refer to the velocity along the x-axis and y-axis, respectively. The sampling time used in this simulation was 5 s. When using the CT model, the state vector needs to be augmented to include the turn rate, as shown in Eq. (3.1.3) [3]. The CT model may be considered nonlinear if the turn rate of the aircraft is not known. Note that a left turn corresponds to a positive turn rate, and a right turn has a negative turn rate. This sign convention follows the commonly used trigonometric convention (the opposite is true for navigation convention) [3]. As per Refs. [3,35], the CT model is given by Eq. (3.1.4)

$$x_k = \begin{bmatrix} \xi_k & \eta_k & \dot{\xi}_k & \dot{\eta}_k & \omega_k \end{bmatrix}^T \tag{3.1.3}$$

$$x_{k+1} = \begin{bmatrix} 1 & 0 & \dfrac{\sin \omega_k T}{\omega_k} & -\dfrac{1-\cos \omega_k T}{\omega_k} & 0 \\ 0 & 1 & \dfrac{1-\cos \omega_k T}{\omega_k} & \dfrac{\sin \omega_k T}{\omega_k} & 0 \\ 0 & 0 & \cos \omega_k T & -\sin \omega_k T & 0 \\ 0 & 0 & \sin \omega_k T & \cos \omega_k T & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} x_k + \begin{bmatrix} \frac{1}{2}T^2 & 0 & 0 \\ 0 & \frac{1}{2}T^2 & 0 \\ T & 0 & 0 \\ 0 & T & 0 \\ 0 & 0 & T \end{bmatrix} w_k \tag{3.1.4}$$

Note that the measurements of the state vector (3.1.3) are m, m, m/s, m/s, and rad/s, respectively. Since the radar stationed at the origin provides direct position measurements only, the measurement equation may be formed linearly as follows:

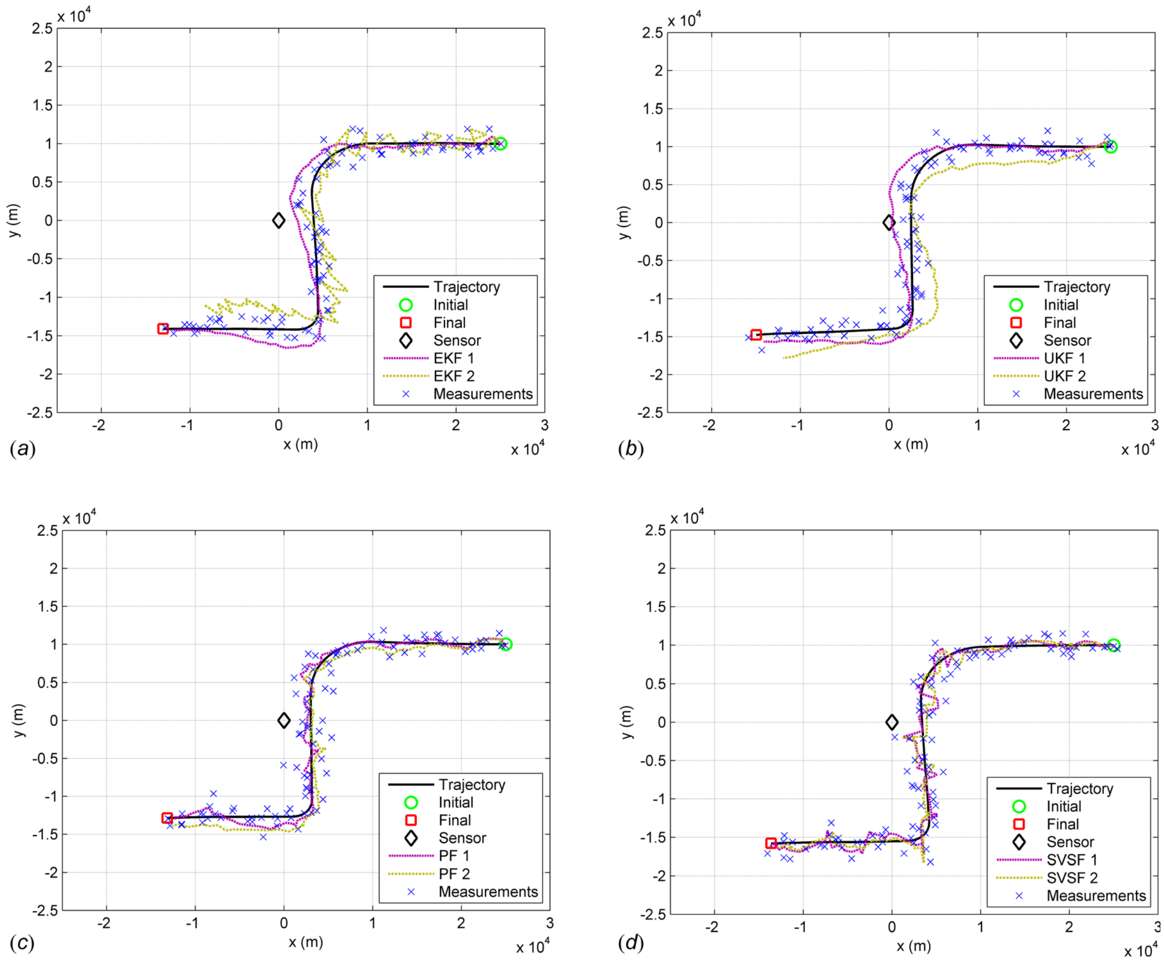

Fig. 5 Results for Sec. 4.1.1, normal conditions: (a) EKF results, (b) UKF results, (c) PF results, and (d) SVSF results

Table 1 Summary of RMSE for estimation strategies (normal conditions, UM model)

RMSE per state	EKF	UKF	PF	SVSF
Position (states 1, 2), m	1518	1299	702	667
Velocity (states 3, 4), m/s	53.6	45.7	36.7	43.7

$$z_k = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix} x_k + v_k \tag{3.1.5}$$

Equations (3.1.1)–(3.1.5) were used to generate the true state values of the trajectory and the radar measurements for this target tracking scenario.

3.2 Bearing-Only Tracking Scenario. The benchmark problem that is studied here is shown in Fig. 4. This problem is described in Refs. [3,36], and will be presented as such. An elevated platform with a sensor travels according to the following equations:

$$x_{p,k} = \bar{x}_{p,k} + \Delta x_{p,k} \tag{3.2.1}$$

$$y_{p,k} = \bar{y}_{p,k} + \Delta y_{p,k} \tag{3.2.2}$$

where $x_{p,k}$ and $y_{p,k}$ are the horizontal and vertical position coordinates, respectively. The first term on the right-hand side of the Eqs. (3.2.1 and 3.2.2) refers to the average platform position coordinates. The last term represents perturbations (i.e., random wind disturbances), and are assumed to be zero-mean Gaussian and independent with variances of $R_x = 1\ \mathrm{m}^2$ and $R_y = 1\ \mathrm{m}^2$, respectively. Note that k represents the discrete-time sequence (from 0 to 20 s).

The average platform motion is assumed to be horizontal with constant velocity, and may be described by the following two equations [36]:

$$\bar{x}_{p,k} = 4k \tag{3.2.3}$$

$$\bar{y}_{p,k} = 20 \tag{3.2.4}$$

The system equation (for the target) is defined according to the following:

$$x_{k+1} = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} x_k + \begin{bmatrix} T^2/2 \\ T \end{bmatrix} w_k \tag{3.2.5}$$

The state vector is defined by the position (m) and velocity (m/s) of the target. The sampling period used in this simulation

Table 2 Summary of RMSE for estimation strategies (normal conditions, CT model)

RMSE per state	EKF	UKF	PF	SVSF
Position (states 1, 2), m	1906 m	2033	1144	781
Velocity (states 3, 4), m/s	11,948	76.7	49.4	126.5
Omega (state 5), rad/s	1.96	1.84	1.79	1.84

was 1 s. The system noise described by w_k is zero-mean Gaussian with a variance of $Q = 10^{-2}\ \mathrm{m}^2/s^4$. For the normal case, the initial position of the target was set to 80 m, and the initial velocity was set to 1 m/s. For case of poor initial estimates, the position of the target was set to 40 m. The nonlinear measurement (sensor) equation is defined by:

$$z_{k+1} = \tan^{-1} \frac{y_{p,k+1}}{x_{1,k+1} - x_{p,k+1}} + v_{k+1} \tag{3.2.6}$$

The first term on the right-hand side of Eq. (3.2.6) is the measured bearing between the horizontal and the line-of-sight from the sensor to the target [36]. The measurement noise v_{k+1} is defined as zero-mean Gaussian with a variance of $R_s = (3°)^2$.

4 Computer Experiments

This section describes the results of applying the four nonlinear estimation techniques on the two target tracking problems. Further to the calculation of the root mean square error (RMSE), the Cramér-Rao lower bound (CRLB) is used as an indicator of the performance of each filter. The CRLB is defined as the inverse of the Fisher information matrix (FIM), which quantifies the available information found in the observations about a state [3]. The CRLB provides a lower bound on the achievable variance in the estimation of a parameter. A derivation that can be used for discrete-time nonlinear filtering is the posterior form (PCRLB) [37–39]. This allows meaningful evaluations of estimation techniques, such that the RMSE for each filter can be determined and compared with the PCRLB. Ideally, one would want the RMSE to reach the PCRLB, or be as close as possible. The CRLB of the error covariance matrix is defined as the inverse of the FIM [3]

$$C = E\{[\hat{x} - x][\hat{x} - x]^T\} \geq J^{-1} \tag{4.1}$$

The inverse of the PCRLB may be calculated recursively as follows [37]:

$$J_{k+1} = \left(F_k J_k^{-1} F_k^T + Q_k\right)^{-1} + H(x_k)^T R_k^{-1} H(x_k) \tag{4.2}$$

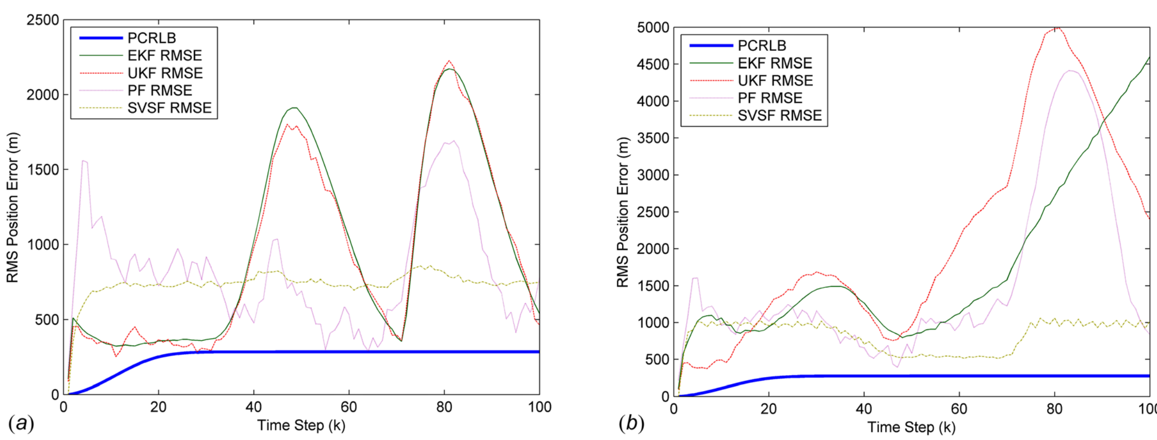

Fig. 6 Results for Sec. 4.1.1, PCRLB and RMSE results: (*a*) uniform motion model and (*b*) coordinated turn model

4.1 Results of the Air Traffic Control Scenario. Both the UM and the CT models were used by each filter. Note that in the following figures, "Filter" 1 refers to a filter being applied to the UM model, and "Filter" 2 refers to a filter being applied to the CT model. For each simulation, a total of 500 Monte Carlo runs were generated to obtain the results.

As previously mentioned, the EKF uses a linearized form of the system and measurement matrices. In this case, the system defined in Eq. (3.1.4) is nonlinear, such that the Jacobian of it yields a linearized form as shown in Eq. (4.1.1). The terms in the last column of Eq. (4.1.1) are correspondingly defined in Eq. (4.1.2).

$$\left[\nabla_x F_{k,x}^T\right]^T\Big|_{x_k=\hat{x}_k} = \begin{bmatrix} 1 & 0 & \dfrac{\sin\hat{\omega}_k T}{\hat{\omega}_k} & -\dfrac{1-\cos\hat{\omega}_k T}{\omega_k} & F_{\hat{\omega}1} \\ 0 & 1 & \dfrac{1-\cos\hat{\omega}_k T}{\hat{\omega}_k} & \dfrac{\sin\hat{\omega}_k T}{\hat{\omega}_k} & F_{\hat{\omega}2} \\ 0 & 0 & \cos\hat{\omega}_k T & -\sin\hat{\omega}_k T & F_{\hat{\omega}3} \\ 0 & 0 & \sin\hat{\omega}_k T & \cos\hat{\omega}_k T & F_{\hat{\omega}4} \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.1.1}$$

$$\begin{bmatrix} F_{\hat{\omega}1} \\ F_{\hat{\omega}2} \\ F_{\hat{\omega}3} \\ F_{\hat{\omega}4} \end{bmatrix} = \begin{bmatrix} \dfrac{(\cos\hat{\omega}_k T)T}{\hat{\omega}_k}\hat{\dot{\xi}}_k - \dfrac{(\sin\hat{\omega}_k T)}{\hat{\omega}_k^2}\hat{\dot{\xi}}_k - \dfrac{(\sin\hat{\omega}_k T)T}{\hat{\omega}_k}\hat{\dot{\eta}}_k - \dfrac{(-1+\cos\hat{\omega}_k T)}{\hat{\omega}_k^2}\hat{\dot{\eta}}_k \\ \dfrac{(\sin\hat{\omega}_k T)T}{\hat{\omega}_k}\hat{\dot{\xi}}_k - \dfrac{(1-\cos\hat{\omega}_k T)}{\hat{\omega}_k^2}\hat{\dot{\xi}}_k - \dfrac{(\cos\hat{\omega}_k T)T}{\hat{\omega}_k}\hat{\dot{\eta}}_k - \dfrac{(\sin\hat{\omega}_k T)}{\hat{\omega}_k^2}\hat{\dot{\eta}}_k \\ -(\sin\hat{\omega}_k T)T\hat{\dot{\xi}}_k - (\cos\hat{\omega}_k T)T\hat{\dot{\eta}}_k \\ (\cos\hat{\omega}_k T)T\hat{\dot{\xi}}_k - (\sin\hat{\omega}_k T)T\hat{\dot{\eta}}_k \end{bmatrix} \tag{4.1.2}$$

To generate the results for this section, the following values were used for the initial state error covariance matrix $P_{0|0}$, the system noise matrix Q, and the measurement noise matrix R.

$$P_{0|0} = \begin{bmatrix} R_{11} & 0 & 0 & 0 & 0 \\ 0 & R_{22} & 0 & 0 & 0 \\ 0 & 0 & 100 & 0 & 0 \\ 0 & 0 & 0 & 100 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.1.3}$$

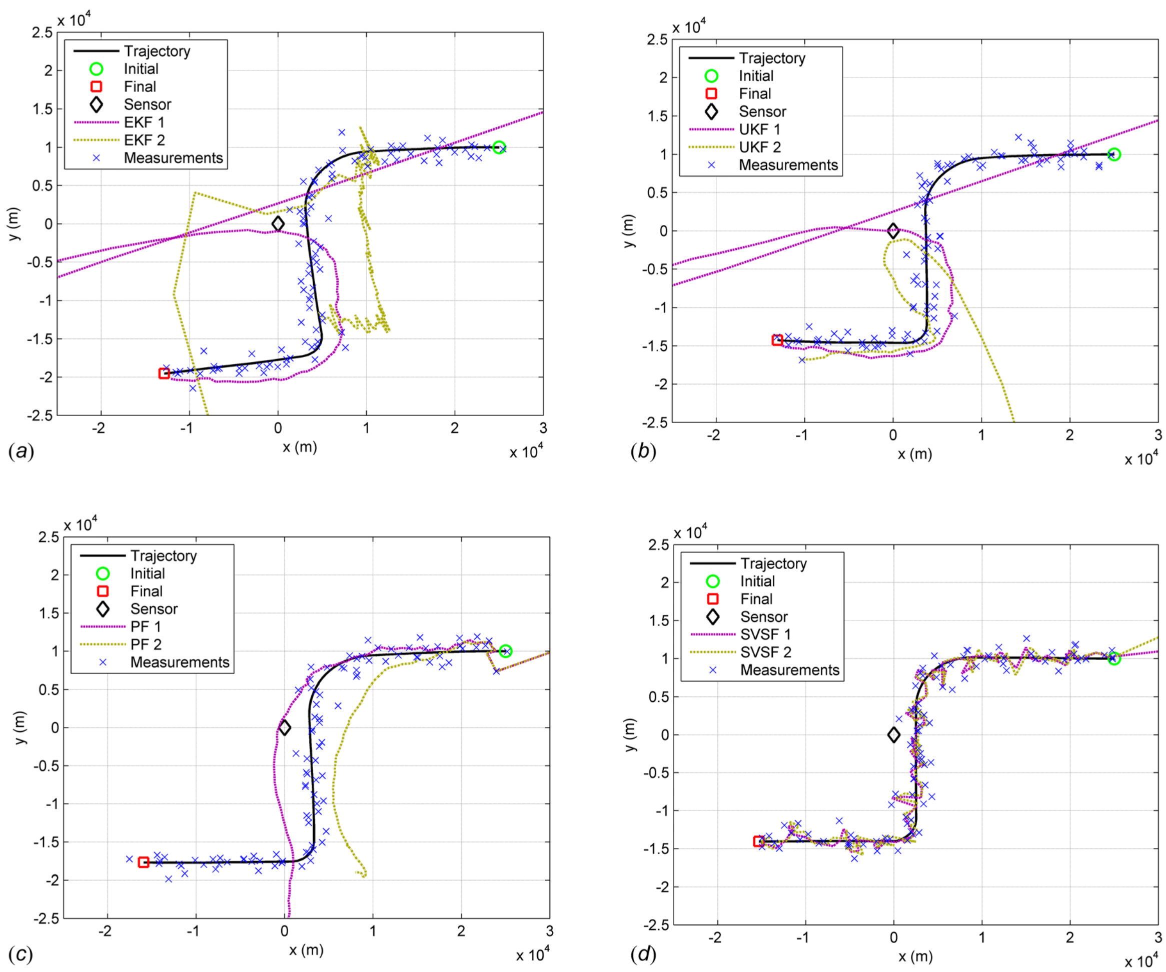

Fig. 7 Results for Sec. 4.1.2, poor initial conditions: (*a*) EKF results, (*b*) UKF results, (*c*) PF results, and (*d*) SVSF results

$$Q = L_1 \begin{bmatrix} \frac{T^3}{3} & 0 & \frac{T^2}{2} & 0 & 0 \\ 0 & \frac{T^3}{3} & 0 & \frac{T^2}{2} & 0 \\ \frac{T^2}{2} & 0 & T & 0 & 0 \\ 0 & \frac{T^2}{2} & 0 & T & 0 \\ 0 & 0 & 0 & 0 & \frac{L_2}{L_1}T \end{bmatrix} \tag{4.1.4}$$

$$R = 1,000^2 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{4.1.5}$$

Note that L_1 and L_2 are referred to as power spectral densities, and were defined as 0.16 and 0.01, respectively [35]. The system and measurement noises (w_k and v_k) were generated using their respective covariance values (Q and R). Also, when using the UM model, the fifth row and column of Eqs. (4.1.3) and (4.1.4) were truncated.

For the UKF estimation process the design parameter k was set to 0.001. For the standalone SVSF estimation process, the limit on the smoothing boundary layer widths were defined as $\psi = [500 \quad 500 \quad 1,000 \quad 1,000 \quad 1]^T$, and the SVSF "memory" or convergence rate was set to $\gamma = 0.1$. These parameters were tuned based on some knowledge of the uncertainties (i.e., magnitude of noise) and with the goal of decreasing the estimation error. A large number of particles were used (5000) to implement the PF, with an effective number of particles set to $N_{\text{eff}} = 0.8$. Furthermore, note that the initial state estimates $\hat{x}_{0|0}$ are set to the initial measurements z_0.

4.1.1 Normal Conditions. This case involved normal conditions, without poor initial conditions and the presence of outliers. The initial state estimates were set to the true initial state values. The PF and the SVSF performed better than the two Kalman-based filters for both models. The simulation results obtained using the EKF are shown in Fig. 5(*a*). While the EKF was using the UM model, the trajectory was not tracked well. After the first turn, the EKF was unable to recover and performed poorly for the remainder of the tracking. The EKF did perform better using the second model; however, there was a significant amount of chattering across the target trajectory resulting in a higher RMSE. The simulation results obtained using the UKF are shown in Fig. 5(*b*). The UKF results obtained using the UM model were similar to the EKF. After the first turn, the UKF was unable to recover and performed poorly for the remainder of the tracking. The UKF did not perform very well using the second model; however, compared with the EKF the estimates were smoother.

The simulation results obtained using the PF are shown in Fig. 5(*c*). Note how the PF was able to follow the trajectory fairly well, using both models. The simulation results obtained using the SVSF are shown in Fig. 5(*d*). Note how the SVSF was also able to track the trajectory and measurements relatively well with each model. The SVSF appeared to be impartial to both models, with the exception of a higher velocity RMSE for the second model. Tables 1 and 2 summarize the RMSE) for the filtering strategies. Table 1 refers to when the filters were implemented using the uniform motion model. Table 2 refers to the results obtained using the coordinated turn model. Note how the EKF was unable to accurately track the velocity estimates. The following is the list of filters, ordered in terms of RMSE accuracy: SVSF, PF, UKF, and

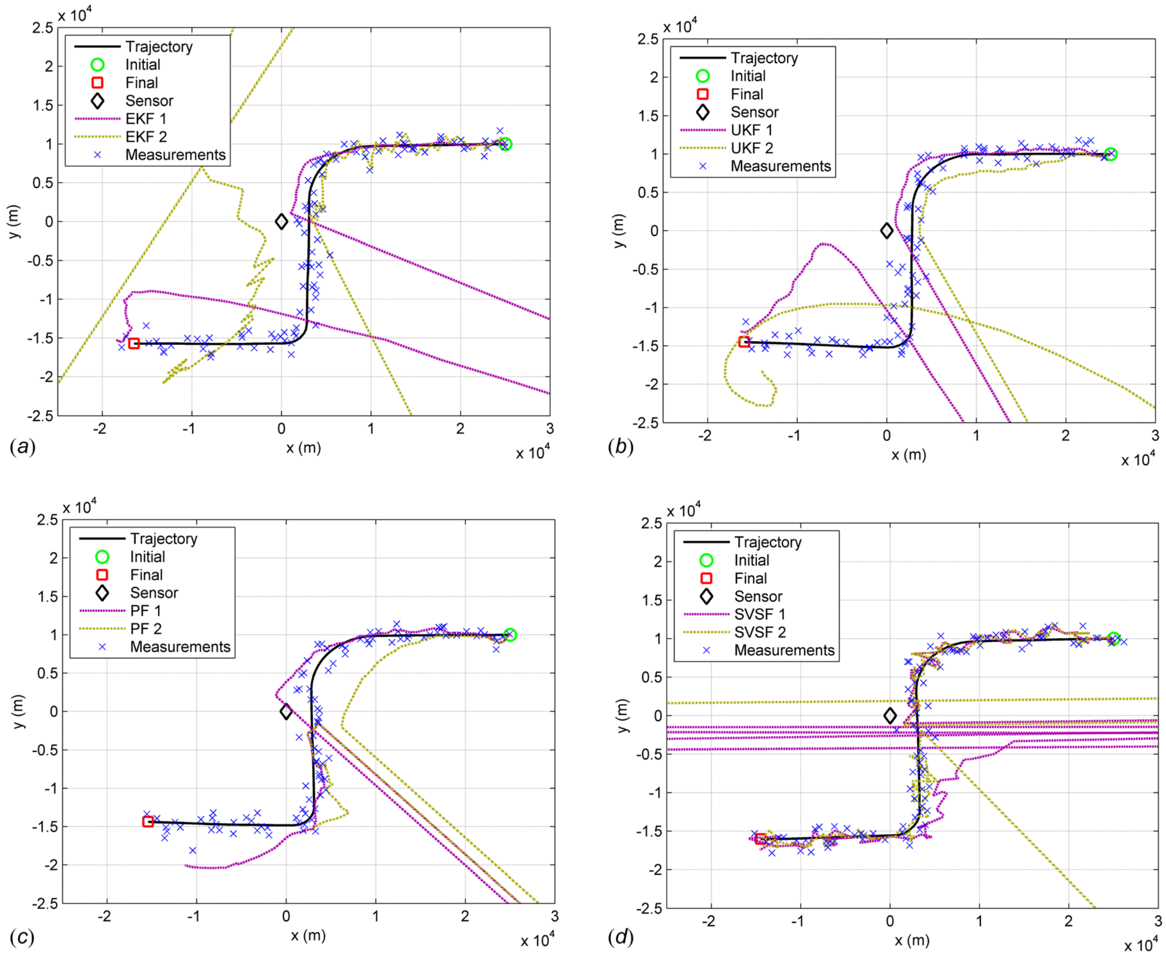

Fig. 8 Results for Sec. 4.1.3, presence of a measurement outlier: (*a*) EKF results, (*b*) UKF results, (*c*) PF results, and (*d*) SVSF results

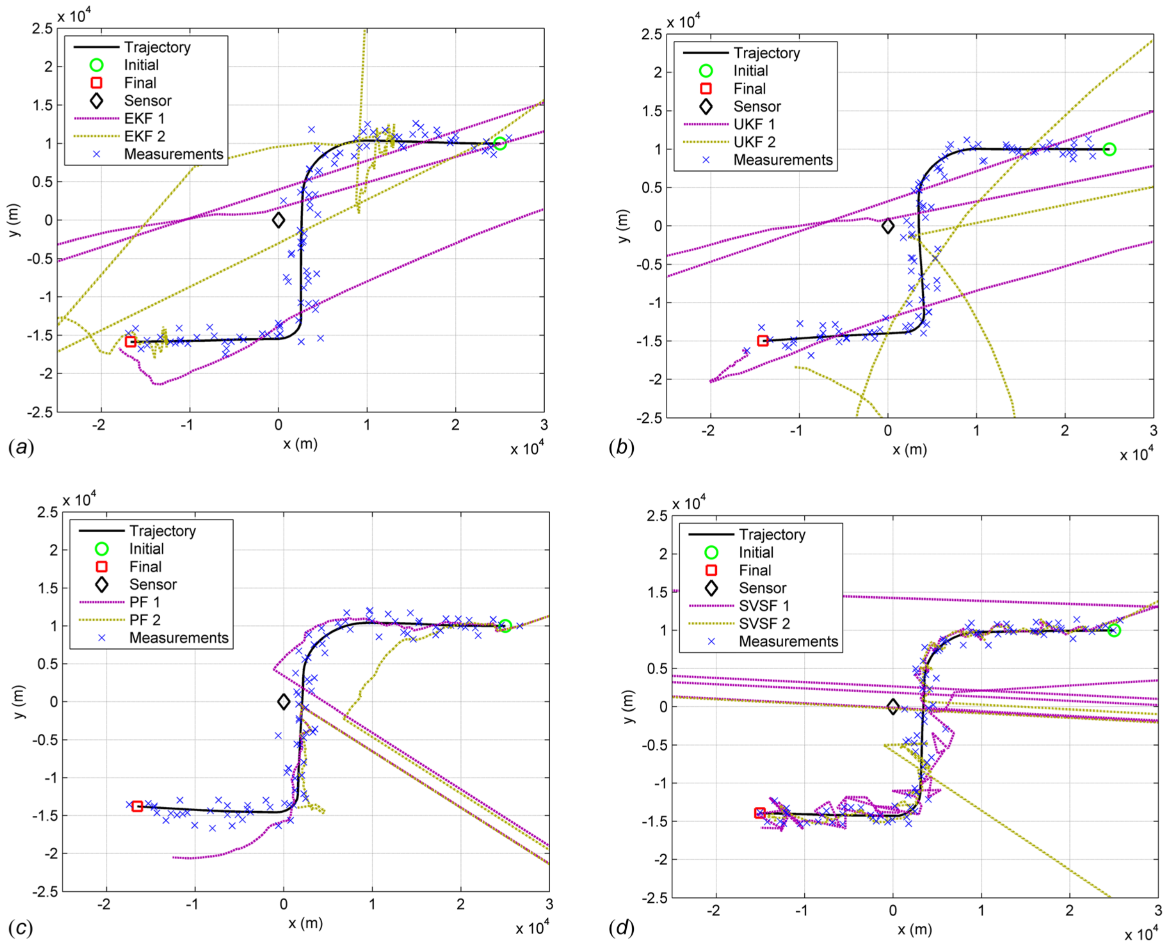

Fig. 9 Results for Sec. 4.1.4, poor initial conditions and outlier: (*a*) EKF results, (*b*) UKF results, (*c*) PF results, and (*d*) SVSF results

EKF. Note how the SVSF performed relatively the same, regardless of which model was implemented.

The PCRLB and root mean square (RMS) position errors for 500 Monte Carlo runs are shown in Fig. 6. Figure 6(*a*) shows the results when the filters were implemented using the uniform motion model, whereas Fig. 6(*b*) is for the coordinated turn model. When studying these figures, it was found that the EKF and UKF performed similarly for the uniform motion case. Quite surprisingly, the EKF performed better than the UKF for the coordinated turn model. It was expected that the UKF would provide a more accurate estimate based on the fact that no linearization of the coordinated turn model was required. For this estimation problem, it was found that the PF was able to obtain a good track of the target. However, overall the SVSF was found to perform the best in terms of estimation accuracy and filter stability, as the RMS position errors were low and bounded fairly close to the PCRLB.

4.1.2 Poor Initial Conditions. This case involved poor initial conditions (the starting estimates were increased by a factor of 100), such that $100 \times \hat{x}_{0|0}$ and $100 \times P_{0|0}$. Changing the initial estimates by a factor of 100 greatly affected the quality of the results obtained by the EKF, as demonstrated in Fig. 7(*a*). The EKF was unable to recover using any of the two models, thus resulting in an unstable estimate. The UKF, as demonstrated in Fig. 7(*b*), was unable to yield a stable estimate. The PF was unable to recover using any of the two models, thus resulting in an unstable estimate, as shown in Fig. 7(*c*). Changing the initial conditions did not greatly affect the behavior of the SVSF, as shown in Fig. 7(*d*). The SVSF recovered after only a few time steps, and was stable for the remainder of the simulation.

Table 3 Values for the SVSF variable boundary layer

State	$1/\psi_1$	$1/\psi_2$	Error range
Position (state 1), m	0.1	200	$>\|2.0\|$
Velocity (state 2), m/s	80	50,000	$>\|0.4\|$

4.1.3 Presence of a Measurement Outlier. This case involved the presence of an outlier among the measurements (the middle measurement—scan 50 of 100—was multiplied by a factor of 500), without poor initial conditions. The presence of an outlier greatly affected the results of the EKF, as shown in Fig. 8(*a*). After the onset of the outlier (about half-way through tracking the target), the EKF became unstable and was unable to accurately continue with the estimation. The UKF yielded a similar result, as shown in Fig. 8(*b*). As shown in Fig. 8(*c*), the PF was slightly less affected when compared with the EKF and UKF. The presence of an outlier did have some effect on the SVSF. As shown in Fig. 8(*d*), chattering began at the onset of the outlier. However, the presence of the chatter was beneficial as it allowed the SVSF to remain stable and bounded to within the target trajectory. The remainder of the estimation continued as in the normal case shown in Sec. 4.1.1.

4.1.4 Poor Initial Conditions and the Presence of a Measurement Outlier. This case involved both the poor initial conditions and the presence of an outlier among the measurements. In this case, the EKF did not perform well at all. In fact, as shown in Fig. 9(*a*), the EKF was completely unstable and

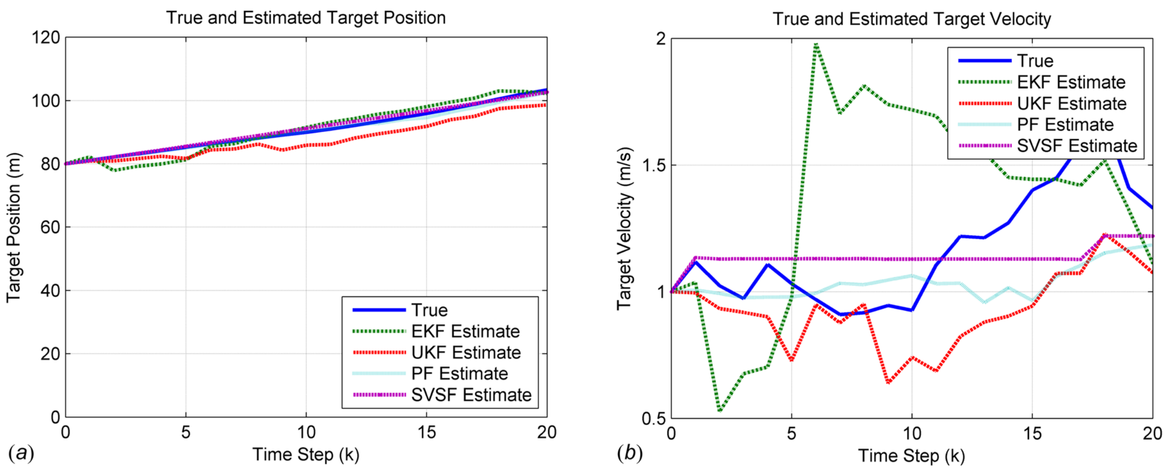

Fig. 10 Results for Sec. 4.2.1, normal conditions: (*a*) estimated position of target and (*b*) estimated velocity of target

was unable to provide any sort of estimate. Likewise, the UKF was unable to yield a stable estimate, as shown in Fig. 9(*b*). Note that the PF performed similarly to the previous case. Unlike the EKF, UKF, and PF, the SVSF was able to overcome the poor initial conditions and additional presence of an outlier. Figure 9(*d*) clearly demonstrates the stability and robustness of this filter.

4.2 Results of the Bearing-Only Tracking Scenario. The EKF was implemented in what is referred to as mixed coordinates. The measurement was left in polar coordinates (bearing-only), while the states of the target were in Cartesian coordinates. The system matrix in this case is already linear; however, the measurement matrix is nonlinear. Taking the partial derivative of the nonlinear component in Eq. (3.2.6) with respect to the first state yields

$$\frac{\partial h}{\partial x_1} = -\frac{y_p}{(x_1 - x_p)^2 + y_p^2} \tag{4.2.1}$$

Furthermore, note that only the knowledge of the average platform positions are used such that the linearized form of the nonlinear measurement matrix becomes

$$H = \begin{bmatrix} -\frac{\bar{y}_p}{(\hat{x}_1 - \bar{x}_p)^2 + \bar{y}_p^2} & 0 \end{bmatrix} \tag{4.2.2}$$

Two cases were studied: normal conditions and poor initial conditions. Under the first scenario, the initial position used by the filters was set to the true value (80 m) and the initial velocity was set to 1 m/s. During the second scenario, the initial position estimate was set to 40 m. The initial covariance matrix used by the EKF was defined as follows:

$$P_{0|0} = \begin{bmatrix} 30 & 0 \\ 0 & 1 \end{bmatrix} \tag{4.2.3}$$

The nonlinear measurement matrix was linearized as per Eq. (4.2.2). There are two main SVSF design parameters. The first parameter (set to $\gamma = 0.2$) controls the speed of convergence, whereas the second (ψ) refers to the boundary layer width which is used to smooth out the switching action. These parameters were tuned by trial-and-error, based on minimizing the estimation error.

To increase the quality of the estimate (in terms of convergence speed and estimation accuracy), the boundary layer was made to change with time. Essentially, the variable boundary layer allows the state estimate to approach the true value as quickly as possible

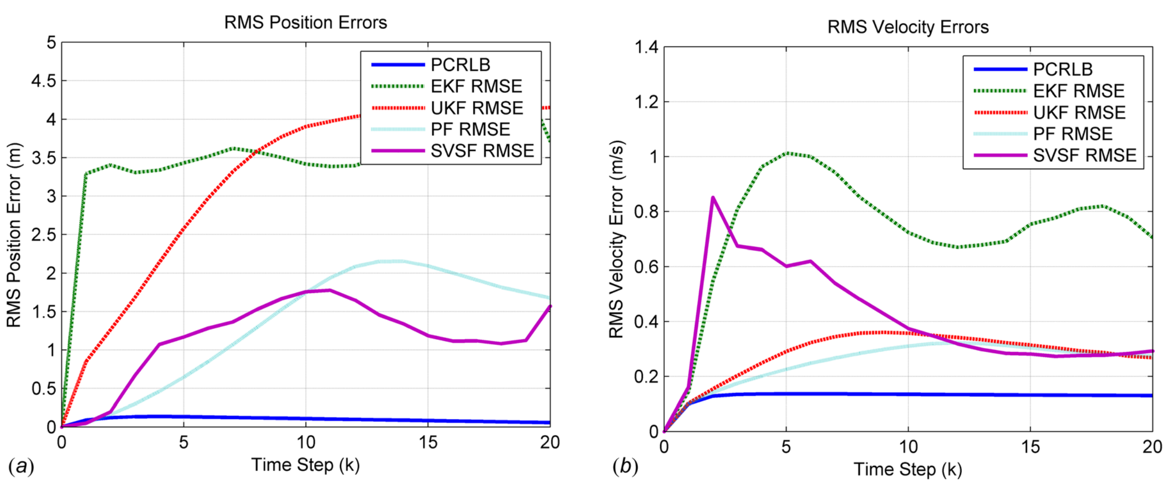

Fig. 11 Results for Sec. 4.2.1, normal conditions: (*a*) RMS position error and (*b*) RMS velocity error

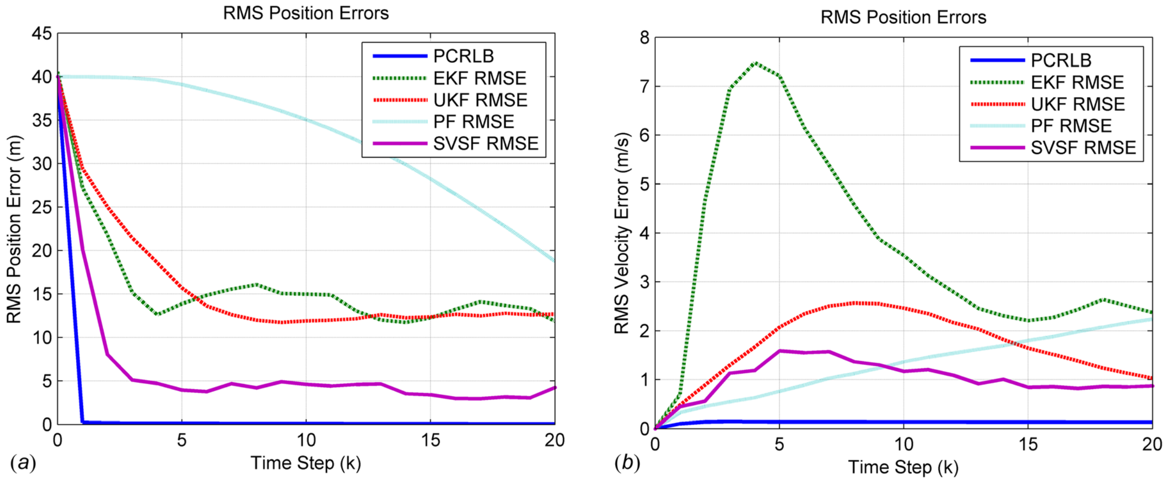

Fig. 12 Results for Sec. 4.2.2, poor initial conditions: (*a*) RMS position error and (*b*) RMS velocity error

by using a large boundary layer width. Once the state estimate is within an acceptable range of the true value, the layer width is decreased, and the estimate is smoothed out. A simple two-stage approach was used, as shown in Table 3, where the values were determined by trial-and-error.

For example, if the absolute position error (between the estimate and the true value) was greater than 2.0, a value of 10 was set for the boundary layer width. Once the absolute position error was less than 2.0, the boundary layer was reduced to a smaller value (5×10^{-3}). Implementing this type of variable boundary layer yielded a more accurate result when compared to a fixed boundary layer width.

Furthermore, since there is no velocity measurement available, the SVSF algorithm described in Sec. 2.4 needs to be slightly modified. The measurement function described by Eq. (3.2.6) needs to be augmented such that a position estimate is formed based on the horizontal and vertical platform positions and the measurement as follows:

$$\hat{\sigma}_{1,k+1} = \frac{\bar{y}_{p,k+1}}{\tan(z_{k+1})} + \bar{x}_{p,k+1} \tag{4.2.4}$$

Furthermore, an estimate of the velocity based on Eq. (4.2.4) may be defined by Eq. (4.2.5). The output errors used in Eqs. (4.2.4) and (4.2.5) are then determined by Eq. (4.2.6), where the second term is found by Eqs. (2.4.1) and (4.2.5)

$$\hat{\sigma}_{2,k+1} = \frac{\hat{\sigma}_{1,k+1} - \hat{\sigma}_{1,k}}{T} \tag{4.2.5}$$

$$e_{\sigma,k+1|k} = \hat{\sigma}_{2,k+1} - \hat{x}_{2,k+1|k} \tag{4.2.6}$$

Furthermore, the PF was implemented using 5000 particles with an effective number of particles set to $N_{\text{eff}} = 0.8$. The code was initialized by sampling from the distribution used to initialize the EKF.

4.2.1 Normal Conditions. An example of the position and velocity target estimates (single run) is shown in Fig. 10. Under normal conditions, when compared with the EKF and UKF, the position was estimated more accurately using the PF and SVSF. Since there was no measurement directly associated with the velocity, the performance of the four filters was found to be significantly worse when attempting to estimate the velocity. However, the filters remained relatively stable, with the EKF providing a slightly worse estimate for the velocity.

The PCRLB and RMS position and velocity errors for 10,000 Monte Carlo runs are shown in Figs. 11 and 12. For the normal conditions case, it was found that the PF and SVSF performed significantly better than the EKF and UKF, in terms of estimation error. For the RMS position error, the PF and SVSF yielded relatively the same results. However, when estimating the velocity,

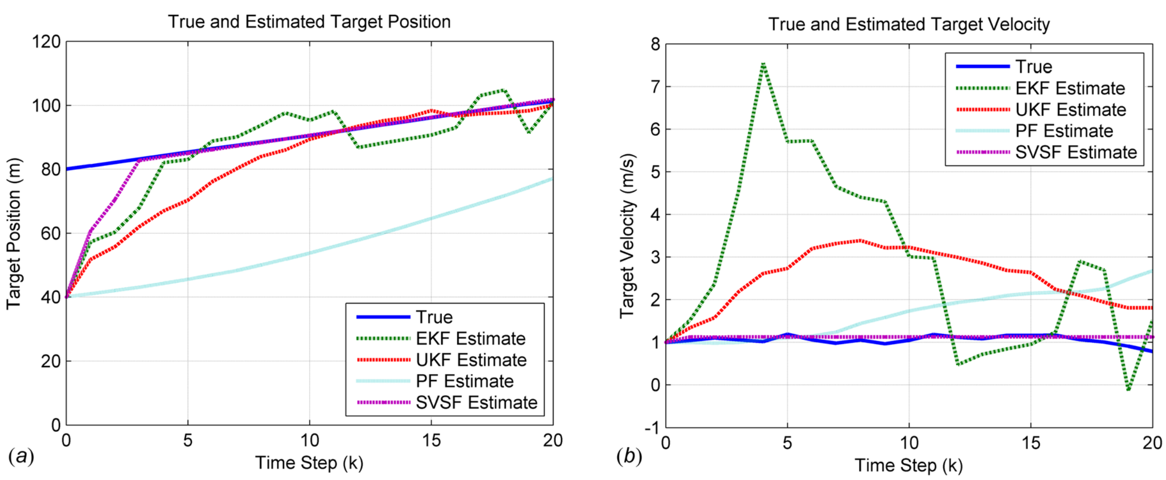

Fig. 13 Results for Sec. 4.2.2, poor initial conditions: (*a*) estimated position of target and (*b*) estimated velocity of target

Table 4 Performance ranking of the estimation strategies

Performance characteristic	EKF	UKF	PF	SVSF
Robustness	4	3	2	1
Stability	4	3	2	1
Accuracy	4	3	2	1
Complexity	1	3	4	2

the SVSF initially had difficulty due to the lack of a velocity measurement. The velocity estimate had to be extracted from the estimated position, as described earlier by Eqs. (4.2.4)–(4.2.6). This estimation process was sensitive to error due to the relatively large sampling time. If a smaller sampling time was used, it is expected that the SVSF would yield a more accurate velocity estimate. The SVSF was able to overcome the lack of information after a few time steps, and provide a relatively stable estimate.

4.2.2 Poor Initial Conditions. Under poor initial conditions, as shown in Fig. 13, the SVSF converged toward the true position value faster than the other three filters. Notice how the EKF overshot the estimate, whereas the SVSF did not. The SVSF appears to be more robust, mainly due to the inherent switching function shown in Eq. (2.4.4) that allows the estimate to stay within a close proximity of the true value. Once within an accurate range of the true value trajectory, the EKF, UKF, and SVSF yielded relatively the same performance as in the normal case. However, the initial estimation errors (about 50%) were too significant, as the PF was unable to maintain the target track and provide an accurate estimate. Note that the EKF and UKF were found to be more sensitive than the SVSF to the influence of the poor initial condition, as shown in the estimate of the target velocity.

The main difference between the filters becomes apparent in the case of the poor initial conditions. The PF was unable to overcome the large initial estimate error; however, the estimates were approaching the correct values and most likely would have reached them given enough time. The PF yields better results if one were to increase the number of particles from 5000 to 20,000. However, this increases the computational time required for the estimation process. Since the PF was already running the slowest, it was not desirable to increase the number of particles. The EKF had difficulty with estimating the velocity but was able to yield a relatively stable estimate after about 10–12 time steps. The overall RMSE was significantly lower for the SVSF, thus suggesting that the SVSF is more robust to handling initial errors in the estimation process. As already mentioned, this is most likely due to the inherent switching found within the SVSF gain.

5 Summary of Computer Experiments

In the first target tracking problem, a generic air traffic control scenario was studied under four different cases: normal, poor initial conditions, presence of an outlier, and a combination of the latter two. It was demonstrated throughout that the EKF was less robust to modeling uncertainties, poor initial conditions, and the presence of outliers. The chattering that is present in the SVSF, caused by the gain switching, brings an inherent amount of stability and robustness to the filter. This is clearly demonstrated in the third case, where the EKF failed due to the outlier. However, due to the switching function of the SVSF gain, the estimates were forced to chatter about the target trajectory, which helped to maintain the target track and yield an accurate estimate.

The performance of these algorithms was ranked in terms of robustness, resilience to poor initial conditions and measurement outliers, and tracking accuracy and computational complexity. Table 4 shows the ranking of the estimation strategies for the results of the simulation. The SVSF is thought to be slightly more complex (computationally) when compared with the EKF. However, the SVSF was demonstrated to be more robust, stable, and accurate.

In the second target tracking problem, a nonlinear bearing-only target tracking scenario was studied using the four estimation strategies, and their performances were compared. This scenario was based on a classical ground surveillance problem, which appears to be deceptively simple. The nonlinear position measurement and lack of a velocity measurement created an interesting estimation problem. For the normal conditions case, the simulation shows that both the PF and the SVSF performed relatively the same in terms of estimation accuracy. The EKF and UKF yielded less accurate state estimates. In the poor initial conditions case, the SVSF yielded more accurate results suggesting that the SVSF is more robust to handling initial errors in the estimation process. In its current form, the SVSF offers two very important advantages: robustness to modeling errors and uncertainties, as well as estimation stability. Currently, the main disadvantage of the SVSF point is the tuning process in determining the SVSF parameters.

It is important to remind the reader that gating and data association techniques help classify the source of measurements, and help associate measurements to the appropriate track [3]. Typically these gating techniques help to avoid extraneous measurements which would otherwise cause the estimation process to go unstable and fail. Obviously the above simulations were designed to test the robustness and stability of the aforementioned estimation techniques. The standalone filters were tested, without any measurement data verification techniques. Furthermore, it is expected that the four filters would work significantly better if multiple-model techniques were implemented [3,8]. Multiple-model techniques assume that the system behaves according to one of a finite number of models. The estimation process would switch between whichever model yields the smallest covariance, or estimated state error [3]. Overall, this would increase the accuracy of the estimation. However, for the purpose of this paper, the filters themselves were tested without the addition of multiple-model techniques.

6 Conclusions

The results of applying the SVSF on two common target tracking problems demonstrate its stability and robustness. It is shown in the above scenarios that the EKF and the UKF perform poorly in the presence of bad initial conditions and measurement outliers. Likewise, the PF appears to have some difficulty when presented with poor initial conditions or uncertain models. However, the SVSF is able to overcome these difficulties, and provide a stable estimate of the states. Furthermore, the EKF appears to be sensitive to model mismatch, as demonstrated in the ATC scenario by the different estimates of the same target, which was calculated using two different target motion models. The SVSF was not as affected, and yielded relatively the same estimate for both models, and is further demonstrated based on the RMSE calculation. Its stability to model mismatch and robustness to poor initial conditions and outliers make using the SVSF advantageous over the well-known extended and unscented Kalman filters. In the past, target tracking problems have typically been solved by Kalman or particle filters. However, the above computer experiment results demonstrate that the SVSF may also be an effective method for handling these types of nonlinear estimation problems.

Nomenclature

C = Cramér-Rao lower bound (CRLB)
e = state estimation error vector
f, F = nonlinear, linear system function or matrix
h, H = nonlinear, linear output or measurement function or matrix
J = Fisher information matrix (FIM)
k = time step index
K = gain value (EKF, UKF, or SVSF)
m = number of measurements

$n =$ number of states
$N_{\text{eff}} =$ effective number of particles
$P =$ error covariance matrix
$q =$ probability or proposal distribution
$Q =$ system noise covariance matrix
$R =$ measurement noise covariance matrix
Sat $=$ saturation function
$t, T =$ simulation, sample time
$u =$ input to the system
$v =$ measurement noise
$w =$ system noise
$W =$ sample weight (UKF)
$x =$ system states
$X =$ sample point (UKF)
$z =$ measurement output
$\xi =$ Cartesian coordinate (position) along the x-axis
$\gamma =$ constant diagonal gain matrix with elements having values between 0 and 1
$\eta =$ Cartesian coordinate (position) along the y-axis
$\Omega =$ turn rate of the target
$\omega =$ particle weight
$\psi =$ smoothing boundary layer
$\hat{} =$ denotes an estimated value
$\sim =$ denotes an error value
$\cdot =$ on top of a parameter denotes a time derivative

Furthermore, note that subscript $(k+1|k)$ refers to an a priori time step and the subscript $(k+1|k+1)$ refers an a posteriori time step. A superscript of T denotes a matrix transpose.

References

[1] Kalman, R. E., 1960, "A New Approach to Linear Filtering and Prediction Problems," Trans. ASME J. Basic Eng., **82**, pp. 35–45.
[2] Grewal, M. S., and Andrews, A. P., 2008, *Kalman Filtering: Theory and Practice Using MATLAB*, 3rd ed., Wiley, New York.
[3] Bar-Shalom, Y., Rong Li, X., and Kirubarajan, T., 2001, *Estimation With Applications to Tracking and Navigation*, Wiley, New York.
[4] Ristic, B., Arulampalam, S., and Gordon, N., 2004, *Beyond the Kalman Filter: Particle Filters for Tracking Applications*, Artech House, Boston, MA.
[5] Habibi, S. R., Burton, R., and Chinniah, Y., 2002, "Estimation Using a New Variable Structure Filter," *American Control Conference*, Anchorage, AK, pp. 2937–2942.
[6] Habibi, S. R., and Burton, R., 2003, "The Variable Structure Filter," ASME J. Dyn. Syst. Meas. Control, **125**, pp. 287–293.
[7] Nise, N., 2011, *Control Systems Engineering*, 6th ed., Wiley, New York.
[8] Gadsden, S. A., 2011, "Smooth Variable Structure Filtering: Theory and Applications," Ph.D. thesis, Department of Mechanical Engineering, McMaster University, Hamilton, ON.
[9] Anderson, B. D. O., and Moore, J. B., 1979, *Optimal Filtering*, Prentice-Hall, Englewood Cliffs, NJ.
[10] Gelb, A., 1974, *Applied Optimal Estimation*, MIT Press, Cambridge, MA.
[11] Welch, G., and Bishop, G., 2006, "An Introduction to the Kalman Filter," Department of Computer Science, University of North Carolina, Chapel Hill, NC.
[12] Simon, D., 2006, *Optimal State Estimation: Kalman, H-Infinity, and Nonlinear Approaches*, Wiley-Interscience, Hoboken, NJ.
[13] Van der Merwe, R., and Wan, E., 2004, "Sigma-Point Kalman Filters for Integrated Navigation," *Proceedings of the Institute of Navigation Annual Meeting*, Dayton, OH, pp. 641–654.
[14] Van der Merwe, R., 2004, "Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State Space Models," Ph.D. thesis, OGI School of Science and Engineering, Oregon Health and Science University, Portland, OR.
[15] Al-Shabi, M., 2011, "The General Toeplitz/Observability SVSF,", Ph.D. thesis, Department of Mechanical Engineering, McMaster University, Hamilton.
[16] Ambadan, J., and Tang, Y., 2009, "Sigma-Point Kalman Filter Data Assimilation Methods for Strongly Nonlinear Systems," J. Atmos. Sci., **66**(2), pp. 261–285.
[17] Tang, X., Zhao, X., and Zhang, X., 2008, "The Square-Root Spherical Simplex Unscented Kalman Filter for State and Parameter Estimation," *Proceedings of the International Conference on Signal Processing*, Beijing, pp. 260–263.
[18] Julier, S. J., Ulhmann, J. K., and Durrant-Whyte, H. F., 2000, "A New Method for Nonlinear Transformation of Means and Covariances in Filters and Estimators," IEEE Trans. Autom. Control, **45**, pp. 472–482.
[19] Julier, S. J., and Uhlmann, J. K., 2004, "Unscented Filtering and Nonlinear Estimation," Proc. IEEE, **92**(3), pp. 401–422.
[20] Kim, S., and Han, W., 2005, "Joint Estimation of Time Delay and Channel Amplitude by Simplex Unscented Filter Without Assisted Pilot in CDMA Systems," *The 7th International Conference on Advanced Communications Technology*, Phoenix Park, South Korea, pp. 233–240.
[21] Julier, S., 2003, "The Spherical Simplex Unscented Transformation," *Proceedings of the American Conference Conference*, Denver, pp. 2430–2434.
[22] Haykin, S., 2001, *Kalman Filtering and Neural Networks*, Wiley, New York.
[23] Moral, D., 1998, "Measure Valued Processes and Interacting Particle Systems: Application to Non-Linear Filtering Problems," Ann. Appl. Prob., **8**(2), pp. 438–495.
[24] Gordon, N. J., Salmond, D. J., and Smith, A. F. M., 1993, "Novel Approach to Nonlinear/Non-Gaussian Bayesian State Estimation," IEEE Proc.-F, **140**(2), pp. 107–113.
[25] MacCormick, J., and Blake, A., 1999, "A Probabilistic Exclusion Principle for Tracking Multiple Objects," *Proceedings of International Conference on Computer Vision*, Kerkyra, Greece, pp. 572–578.
[26] Kanazawa, K., Koller, D., and Russell, S. J., 1995, "Stochastic Simulation Algorithms for Dynamic Probabilistic Networks," *Proceedings of the Eleventh Annual Conference on Uncertainty in Artificial Intelligence*, Montreal, pp. 346–351.
[27] Hammersley, J. M., and Morton, K. W., 1954, "Poor Man's Monte Carlo," J. Roy. Stat. Soc. B, **16**, pp. 23–28.
[28] Handschin, J. E., and Mayne, D. Q., 1969, "Monte Carlo Techniques to Estimate the Conditional Expectation in Multi-Stage Non-Linear Filtering," Int. J. Control, **9**(5), pp. 547–559.
[29] Akashi, H., and Kumamoto, H., 1977, "Random Sampling Approach to State Estimation in Switching Environments," Automatica, **13**, pp. 429–434.
[30] Doucet, A., de Freitas, N., and Gordon, N. J., 2001, *Sequential Monte Carlo Methods in Practice*, Springer, New York.
[31] Djuric, P. M., and Goodsill, S. J., 2002, "Guest Editorial Special Issue on Monte Carlo Methods for Statistical Signal Processing," IEEE Trans. Signal Process., **50**(2), p. 173.
[32] Kong, A., Liu, J. S., and Wong, W. H., 1994, "Sequential Imputations and Bayesian Missing Data Problems," J. Am. Stat. Assoc., **89**(425), pp. 278–288.
[33] Habibi, S. R., 2007, "The Smooth Variable Structure Filter," Proc. IEEE, **95**(5), pp. 1026–1059.
[34] Wang, H., Kirubarajan, T., and Bar-Shalom, Y., 1999, "Precision Large Scale Air Traffic Surveillance Using an IMM Estimator With Assignment," Trans. IEEE Aerosp. Electron. Syst. J., **35**(1), pp. 255–266.
[35] Lin, X., Kirubarajan, T., Bar-Shalom, Y., and Maskell, S., 2002, "Comparison of EKF, Pseudomeasurement Filter and Particle Filter for a Bearings-only Target Tracking Problem," Proceedings of the SPIE Conference of Signal and Data Processing of Small Targets, Orlando, FL.
[36] Hernandez, M. L., Kirubarajan, T., and Bar-Shalom, Y., 2004, "Multisensor Resource Management in the Presence of Clutter Using Cramér-Rao Lower Bounds," IEEE Trans. Aerosp. Electron. Syst., **40**(2), pp. 399–416.
[37] Zhang, X., and Willett, P., 2001, "Cramér-Rao Bounds for Discrete-Time Linear Filtering With Measurement Origin Uncertainties," Proceedings of the Workshop on Estimation, Tracking and Fusion: A Tribute to Yaakov Bar-Shalom, CA.
[38] Tichavsky, P., 1998, "Posterior Cramér-Rao Bounds for Discrete-Time Nonlinear Filtering," IEEE Trans. Signal Process., **46**(5), pp. 1386–1396.
[39] Arasaratnam, I., Haykin, S., and Elliott, R. J., 2007, "Discrete-Time Nonlinear Filtering Algorithms Using Gauss-Hermite Quadrature," Proc. IEEE., **95**(5), pp. 953–977.