# Iterated Unscented Kalman Filter for Passive Target Tracking

**It is of great importance to develop a robust and fast tracking algorithm in passive localization and tracking system because of its inherent disadvantages such as weak observability and large initial errors. In this correspondence, a new algorithm referred to as the iterated unscented Kalman filter (IUKF) is proposed based on the analysis and comparison of conventional nonlinear tracking problem. The algorithm is developed from UKF but it can obtain more accurate state and covariance estimation. Compared with the traditional approaches (e.g., extended Kalman filter (EKF) and UKF) used in passive localization, the proposed method has potential advantages in robustness, convergence speed, and tracking accuracy. The correctness as well as validity of the algorithm is demonstrated through numerical simulation and experiment results.**

## I. INTRODUCTION

Passive localization and tracking is usually a problem of nonlinear filtering, and the general purpose is to estimate the time-varying state of the target based on the noisy nonlinear measurement. The classical and widely used algorithm for this purpose is the extended Kalman filter (EKF) [1, 2], under which framework, the state distribution is approximated by a Gaussian random variable (GRV), which is then propagated analytically through the first-order linearization of the nonlinear system. However, this method of linearization may introduce large errors in the true posterior mean and covariance of the transformed GRV, leading to sub-optimal performance or sometimes divergence of the filter [3, 4]. To reduce the effect introduced by linearization, Song and Speyer proposed a modified gain extended Kalman filter (MGEKF) [5] in an attempt to obtain improved performance by calculating a modified Kalman gain. A basic assumption in MGEKF is that the nonlinear system function is modifiable.[1] Under the condition of bearings-only (BO) location, Galkowski and Islam [6] developed a simple method to modify the nonlinear function, and they confirmed that the erratic behavior of EKF could be eliminated by MGEKF.

However, it is not always the case that the modifiable condition can be satisfied, which greatly limits the application of this method. In [7], Guo Fucheng et al. put forward a modified covariance EKF (MVEKF) algorithm in the belief that state filtering was closer to the true value than state prediction, so after obtaining the filtering estimate, they relinearized the nonlinear measurement equation at the filtered state and then used the recomputed Jacobian matrix to calculate the modified Kalman gain and covariance matrices. Simulation results indicated that this method exhibited the same performance as MGEKF but without the limitation of modifiable condition.

In light of the intuition that to approximate a probability distribution is easier than to approximate an arbitrary nonlinear transformation, Julier and Uhlmann presented a novel filter called the unscented Kalman filter (UKF) in [8], [9]. Unlike the EKF which approximates the nonlinear state and measurement equations using linearization, the UKF uses a set of carefully chosen sample (or sigma) points to represent the state distribution. These sample points can completely capture the true mean and covariance of the GRV, and when propagated through the true nonlinear system, capture the posterior mean and covariance accurately to the 3rd-order Taylor series expansion for any nonlinearity. Besides its higher approximation accuracy, this approach can avoid the cumbersome evaluation of Jacobian and Hessian matrices, making the algorithm easier to implement. For these advantages, the UKF has received increasing attraction [10–13].

The UKF methods have proven to be far superior to standard EKF in a wide range of applications [4, 11, 13]. In the application of passive target tracking, however, because of the large initial error and weak observability of the system, the standard UKF also shows its weakness in robustness, convergence speed, and tracking accuracy. In this correspondence, we propose an iterated UKF to address these problems. We also provide the numerical simulation and experiment results to show the improved performance achieved.

Manuscript received June 20, 2005; revised April 10, 2006 and January 10, January 24, and January 28, 2007; released for publication June 1, 2007.

IEEE Log No. T-AES/43/3/908421.

Refereeing of this contribution was handled by D. J. Salmond.

[1] The term modifiable is defined as follows. Given the time-varying function $h_k : R^n \rightarrow R^m$, there exists a matrix $G_k(\cdot) \in R^{m\times n}$, so that for any $\mathbf{x}$, $\bar{\mathbf{x}} \in R^n$, we have $h_k(\mathbf{x}) - h_k(\bar{\mathbf{x}}) = G_k(\mathbf{z}_k, \bar{\mathbf{x}})(\mathbf{x} - \bar{\mathbf{x}})$, where $\mathbf{z}_k = h_k(\mathbf{x}_k)$. For any nonlinear function $h_k$, if the modifiable condition is met, then the Jacobian matrix $H_k = \nabla_{\mathbf{x}} h(\mathbf{x})|_{\mathbf{x}=\bar{\mathbf{x}}}$ can be replaced with the modifying matrix $G_k(\mathbf{z}_k, \bar{\mathbf{x}})$ in calculating Kalman gain and covariance matrix. Accordingly, the resulted EKF is referred to as MGEKF.

## II. DEVELOPMENT OF THE ITERATED UNSCENTED KALMAN FILTER

### A. Unscented Kalman Filter

Suppose that a state variable $\mathbf{x}$ with mean $\bar{\mathbf{x}}$ and covariance $P$ is propagated through the following nonlinear state equation

$$\mathbf{x}_k = f(\mathbf{x}_{k-1}) + \mathbf{v}_{k-1} \tag{1}$$

and the measurement equation is given as

$$\mathbf{y}_k = h(\mathbf{x}_k) + \mathbf{n}_k \tag{2}$$

where $\mathbf{v}_k$ and $\mathbf{n}_k$ denote the process and measurement noise with corresponding covariance matrices $Q_k$ and $R_k$, respectively.

Under the general UKF framework, the recursive estimate for $\mathbf{x}_k$, based on (1)–(2), can be obtained from the following steps.

*Step* 1 Calculate Sigma Points.

At instant $k-1$, a set of deterministic sample points with associated weights are generated as

$$\begin{aligned}
\mathcal{X}_{0,k-1} &= \bar{\mathbf{x}}_{k-1}\\
\mathcal{X}_{i,k-1} &= \bar{\mathbf{x}}_{k-1} + \left(\sqrt{(L+\gamma)P_{k-1}}\right)_i, \qquad i = 1,2,\ldots,L\\
\mathcal{X}_{i,k-1} &= \bar{\mathbf{x}}_{k-1} - \left(\sqrt{(L+\gamma)P_{k-1}}\right)_{i-L}, \qquad i = L+1,\ldots,2L
\end{aligned} \tag{3}$$

$$\begin{aligned}
w_0^{(m)} &= \gamma/(L+\gamma)\\
w_0^{(c)} &= \gamma/(L+\gamma) + (1-\alpha^2+\beta)\\
w_i^{(m)} &= w_i^{(c)} = 1/\{2(L+\gamma)\}, \qquad i = 1,2,\ldots,2L
\end{aligned} \tag{4}$$

where $\mathcal{X}_i$ and $w_i$ represent sigma point and corresponding weight, respectively; $L$ is the dimension of $\mathbf{x}$; $\gamma = \alpha^2(L+\kappa) - L$ is a scaling parameter; $\alpha$ determines the spread of the sigma points around $\bar{\mathbf{x}}$, and is usually a small positive value (1e-4 $\leq \alpha \leq 1$); $\kappa$ is a secondary scaling parameter which is usually set to 0; $\beta$ is used to incorporate the prior distribution of $\mathbf{x}$ (for a Gaussian prior the optimal choice [14] is $\beta = 2$); $(\sqrt{P})_i$ denotes the $i$th row of the matrix square root.

*Step* 2 Time Update.

After the sample points are propagated through the nonlinear equations, the predicted mean and covariance are computed as

$$\mathcal{X}_{i,k|k-1} = f(\mathcal{X}_{i,k-1}) \tag{5}$$

$$\hat{\mathbf{x}}_k^- = \sum_{i=0}^{2L} w_i^{(m)} \mathcal{X}_{i,k|k-1} \tag{6}$$

$$P_k^- = \sum_{i=0}^{2L} w_i^{(c)} [\mathcal{X}_{i,k|k-1} - \hat{\mathbf{x}}_k^-][\mathcal{X}_{i,k|k-1} - \hat{\mathbf{x}}_k^-]^T + Q_k \tag{7}$$

$$\mathcal{X}^*_{i,k|k-1} = \left[\mathcal{X}_{0:2L,k|k-1} \quad \mathcal{X}_{0,k|k-1} + \nu\sqrt{Q_k} \quad \mathcal{X}_{0,k|k-1} - \nu\sqrt{Q_k}\right]_i \tag{8}$$

$$\mathcal{Y}^*_{i,k|k-1} = h(\mathcal{X}^*_{i,k|k-1}) \tag{9}$$

$$\hat{\mathbf{y}}_k^- = \sum_{i=0}^{2L^a} w_i^{*(m)} \mathcal{Y}^*_{i,k|k-1} \tag{10}$$

$$P_{\mathbf{yy},k} = \sum_{i=0}^{2L^a} w_i^{*(c)} [\mathcal{Y}^*_{i,k|k-1} - \hat{\mathbf{y}}_k^-][\mathcal{Y}^*_{i,k|k-1} - \hat{\mathbf{y}}_k^-]^T + R_k \tag{11}$$

$$P_{\mathbf{xy},k} = \sum_{i=0}^{2L^a} w_i^{*(c)} [\mathcal{X}^*_{i,k|k-1} - \hat{\mathbf{x}}_k^-][\mathcal{Y}^*_{i,k|k-1} - \hat{\mathbf{y}}_k^-]^T \tag{12}$$

where $L^a = 2L$, $\nu = \sqrt{L+\gamma}$; $w_i^{*(m)}$ and $w_i^{*(c)}$ are calculated in the same way as (4) with the replacement of $L$ by $L^a$. Note that in (8), the sigma points are augmented with additional points derived from the matrix square root of the process noise covariance $Q$. The main purpose is to incorporate the effect of the process noise on the observed sigma points $\mathcal{Y}$; see [4] for details.

*Step* 3 Measurement Update.

The Kalman gain is calculated to update the state and covariance.

$$\mathcal{K}_k = P_{\mathbf{xy},k} P_{\mathbf{yy},k}^{-1} \tag{13}$$

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + \mathcal{K}_k(\mathbf{y}_k - \hat{\mathbf{y}}_k^-) \tag{14}$$

$$P_k = P_k^- - \mathcal{K}_k P_{\mathbf{yy},k} \mathcal{K}_k^T. \tag{15}$$

The above three steps provide a general summary of the UKF algorithm. Given the initial condition $\bar{\mathbf{x}}_0 = E[\mathbf{x}_0]$ and $P_0 = E[(\mathbf{x}_0 - \bar{\mathbf{x}}_0)(\mathbf{x}_0 - \bar{\mathbf{x}}_0)^T]$, the filtering procedure can be recursively implemented.

### B. Iterated Extended Kalman Filter

Before coming to the proposed algorithm, we first recall another variation of EKF called the iterated EKF (IEKF) [1, 2]. The main difference between IEKF and EKF lies in the step of measurement update. For the IEKF, once the state prediction $\hat{\mathbf{x}}_k^-$ and corresponding covariance $P_k^-$ are obtained, the following iterates will be recursively carried out

$$\hat{\mathbf{x}}_{k,0} = \hat{\mathbf{x}}_k^-, \qquad P_{k,0} = P_k^- \tag{16}$$

$$\hat{\mathbf{x}}_{k,j+1} = \hat{\mathbf{x}}_k^- + K_{k,j}[\mathbf{y}_k - h(\hat{\mathbf{x}}_{k,j}) - H_{k,j}(\hat{\mathbf{x}}_k^- - \hat{\mathbf{x}}_{k,j})] \tag{17}$$

$$P_{k,j} = (I - K_{k,j}H_{k,j})P_k^- \tag{18}$$

where

$$\begin{aligned}
H_{k,j} &= \left.\frac{\partial h(\mathbf{x})}{\partial \mathbf{x}}\right|_{\mathbf{x}=\hat{\mathbf{x}}_{k,j}}\\
K_{k,j} &= P_k^- H_{k,j}^T (H_{k,j} P_k^- H_{k,j}^T + R_k)^{-1}.
\end{aligned} \tag{19}$$

The iterative process will not be stopped until a certain termination condition is met. The termination condition may be varied in different cases, but a commonly used criterion to terminate the iterate is that the inequality $\|\hat{\mathbf{x}}_{k,j+1} - \hat{\mathbf{x}}_{k,j}\| \le V_{\text{th}}$ can be satisfied, where $V_{\text{th}}$ is the predetermined threshold. For the iterates number $j = 1,2,\ldots,N$, the ultimate outputs of the state estimate and corresponding covariance matrix are

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_{k,N}, \qquad P_k = P_{k,N}. \tag{20}$$

It was pointed out in [15] that the sequence of iterates generated by the IEKF with initial value $\hat{\mathbf{x}}_{k,0}$ and that generated by the Gauss-Newton method were identical, thus global convergence of the iterates was guaranteed.

Theoretically, the IEKF is surely superior to EKF and MVEKF; however, as indicated in [7], [16], this is not always true. One reason is that the conclusion in [15] was drawn under the assumption that the local linearization condition is unconditionally met, i.e., the state estimate is close enough to the true value. In many applications (especially in our case), however, this assumption does not always hold because the initial estimate errors may be very large. The other reason is that the measurement error cannot be ideally low. From the update equations, it is clear that the state correction in each iterate is realized through measurement; accordingly, the convergence property of iterates depends on the measurement precision. As confirmed in [7], [16], the IEKF is very sensitive to measurement errors. So if the state correction is completely dependent upon the measurement during the process of iterate, there is probably no improved performance obtained, instead, the performance may be greatly degraded. Another reason is that the Gauss-Newton method is not guaranteed to go up the likelihood surface [17], though it is guaranteed to be globally convergent. Additionally, the threshold $V_{\text{th}}$ is crucial to successfully using the iterated algorithm, but a proper choice of $V_{\text{th}}$ is not easy.

### C. Iterated Unscented Kalman Filter

Enlightened by the development of IEKF as well as the superiority of UKF, a natural idea is that improved performance may be expected if the iterates are implemented in UKF. But in view of the potential problems exhibited by the IEKF, special steps should be taken to make the iterated filter perform as is possible. In what follows, an iterated unscented Kalman filter (IUKF) will be developed to address this problem, using a different iteration strategy. The pseudocode for the IUKF can be summarized as follows.

*Step* 1 For each instant $k$ ($k \ge 1$), evaluate the state estimate $\hat{\mathbf{x}}_k$ and corresponding covariance matrix $P_k$ through (3)–(15).

*Step* 2 Let $\hat{\mathbf{x}}_{k,0} = \hat{\mathbf{x}}_k^-$, $P_{k,0} = P_k^-$ and $\hat{\mathbf{x}}_{k,1} = \hat{\mathbf{x}}_k$, $P_{k,1} = P_k$. Also let $g = 1$, $j = 2$.

*Step* 3 Generate new sigma points in the same way as (3)

$$\mathcal{X}_{i,j} = \left[\hat{\mathbf{x}}_{k,j-1} \quad \hat{\mathbf{x}}_{k,j-1} \pm \sqrt{(L+\gamma)P_{k,j-1}}\right]_i \tag{21}$$

*Step* 4 Recalculate (6)–(15) as follows

$$\hat{\mathbf{x}}_{k,j}^- = \sum_{i=0}^{2L} w_i^{(m)} \mathcal{X}_{i,j} \tag{22}$$

$$\mathcal{Y}_{i,j} = h(\mathcal{X}_{i,j}) \tag{23}$$

$$\hat{\mathbf{y}}_{k,j}^- = \sum_{i=0}^{2L} w_i^{(m)} \mathcal{Y}_{i,j} \tag{24}$$

$$P_{\mathbf{yy},k,j} = \sum_{i=0}^{2L} w_i^{(c)} [\mathcal{Y}_{i,j} - \hat{\mathbf{y}}_{k,j}^-][\mathcal{Y}_{i,j} - \hat{\mathbf{y}}_{k,j}^-]^T + R_k \tag{25}$$

$$P_{\mathbf{xy},k,j} = \sum_{i=0}^{2L} w_i^{(c)} [\mathcal{X}_{i,j} - \hat{\mathbf{x}}_{k,j}^-][\mathcal{Y}_{i,j} - \hat{\mathbf{y}}_{k,j}^-]^T \tag{26}$$

$$\mathcal{K}_{k,j} = P_{\mathbf{xy},k,j} P_{\mathbf{yy},k,j}^{-1} \tag{27}$$

$$\hat{\mathbf{x}}_{k,j} = \hat{\mathbf{x}}_{k,j}^- + g \cdot \mathcal{K}_{k,j}(\mathbf{y}_k - \hat{\mathbf{y}}_{k,j}^-) \tag{28}$$

$$P_{k,j} = P_{k,j-1} - \mathcal{K}_{k,j} P_{\mathbf{yy},k,j} \mathcal{K}_{k,j}^T \tag{29}$$

where the subscript $j$ denotes the $j$th iterate; $\mathcal{Y}_{i,j}$ represents the $i$th component of $\mathcal{Y}_j$. It is worth noting that for the 0th iterate of the IUKF, the sigma points have been augmented (the entire set of sigma points have been drawn with $Q$ included), i.e., the effect of the process noise has been incorporate into the observed sigma points, so there is no need to augment the sigma points for the remaining iterates.

*Step* 5 Define the following three equations

(1) $\hat{\mathbf{y}}_{k,j} = h(\hat{\mathbf{x}}_{k,j})$

(2) $\tilde{\mathbf{x}}_{k,j} = \hat{\mathbf{x}}_{k,j} - \hat{\mathbf{x}}_{k,j-1}$

(3) $\tilde{\mathbf{y}}_{k,j} = \mathbf{y}_k - \hat{\mathbf{y}}_{k,j}$.

*Step* 6 If the following inequality holds

$$\tilde{\mathbf{x}}_{k,j}^T P_{k,j-1}^{-1} \tilde{\mathbf{x}}_{k,j} + \tilde{\mathbf{y}}_{k,j}^T R_k^{-1} \tilde{\mathbf{y}}_{k,j} < \tilde{\mathbf{y}}_{k,j-1}^T R_k^{-1} \tilde{\mathbf{y}}_{k,j-1} \tag{30}$$

and $j \le N$, then set $g = \eta \cdot g$, $j = j + 1$ and return to Step 3; otherwise, continue to Step 7.

*Step* 7 Stop if the inequality (30) is not satisfied or if $j$ is too large ($j > N$) and set $\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_{k,j}$, $P_k = P_{k,j}$.

To explain why the iterates converge to a solution, we first come back to (29). For the positive definite matrices $P_{k,j}$, $P_{k,j-1}$, and $P_{\mathbf{yy},k,j}$, assume that $\lim_{j\to\infty} \mathcal{K}_{k,j} \neq \mathbf{0}$, then according to (29) we have $P_{k,j} < P_{k,j-1}$[2] for any $j = 1,2,\cdots < \infty$. Based on

[2] For positive definite matrices $A,B \in R^{n\times n}$, by denoting $A < B$ we mean that for any vector $\mathbf{a} \in R^n$ and $\mathbf{a} \neq \mathbf{0}$ we have $\mathbf{a}^T A\mathbf{a} < \mathbf{a}^T B\mathbf{a}$.

the fact that each element of the matrix $P_{k,j}$ is bounded, it is easy to know (see Appendix A for details) that

$$\lim_{j\to\infty} P_{k,j} = \lim_{j\to\infty} P_{k,j-1}. \tag{31}$$

With this premise, it can be inferred from (29) that $\lim_{j\to\infty} \mathcal{K}_{k,j} = \mathbf{0}$, which violates the assumption that $\lim_{j\to\infty} \mathcal{K}_{k,j} \neq \mathbf{0}$. Obviously, the assumption does not hold and the only possibility is that $\lim_{j\to\infty} \mathcal{K}_{k,j} = \mathbf{0}$. Now suppose that $\mathcal{K}_{k,j} \to \mathbf{0}$ when $j > N$, then from (28) and (29) we have $\hat{\mathbf{x}}_{k,j} \to \hat{\mathbf{x}}_{k,j}^{-} = \hat{\mathbf{x}}_{k,j-1}$ and $P_{k,j} \to P_{k,j-1}$, which means that convergence is guaranteed as the iterate proceeds.

According to (28), the result of $N$th iterate is $\hat{\mathbf{x}}_{k,N} = \hat{\mathbf{x}}_{k,N}^{-} + \eta^{N-1} \cdot \mathcal{K}_{k,N}(\mathbf{y}_k - \hat{\mathbf{y}}_{k,N}^{-})$. Now if the decaying factor $\eta$ is chosen as $0 < \eta < 1$ and $N$ is large enough, then we have $\hat{\mathbf{x}}_{k,N} \to \hat{\mathbf{x}}_{k,N}^{-} = \hat{\mathbf{x}}_{k,N-1}$ because $\eta^{N-1} \to 0$ for a large $N$. From this assumption, it can also be concluded that the iterates will converge to a solution; meanwhile, the convergence speed is affected by the factor $\eta$.

In comparison with traditional IEKF, the proposed algorithm differs mainly in three aspects. One is the termination criteria in Step 6, the inequality in (30) is guaranteed to go up the likelihood surface in the process of iterates (see Appendix B for a brief justification), i.e., the iterates are guarded to move towards maximum likelihood solution (optimal solution). The other difference is the step length adjustment, the decaying factor $\eta$ $(0 < \eta \leq 1)$ is used to weaken the effect of the latest correction on the prediction state $(\hat{\mathbf{x}}_{k,j}^{-})$, which makes two successive iterates $(\hat{\mathbf{x}}_{k,j}$ and $\hat{\mathbf{x}}_{k,j-1})$ become closer as the iterate proceeds, thus speeding up the iterative convergence. Another difference is that in each iterate, the new sigma points are generated according to (21), which means that the samples generation is guided by the latest covariance matrix, making the sigma points converge at the true state step by step.

Compared with the standard UKF, the IUKF can adjust the state estimate to adaptively approach the true value through corrections of the measurement, so after the iterate terminates, a lower state error can be expected. In addition, the proposed filter can respond to new measurement as quickly as possible with the adjustment of state and covariance matrix, making a faster convergence speed possible in situations where the initial error is large.

Here we should point out that while this paper was under review, a similar algorithm referred to as iterated sigma point Kalman filter (ISPKF) has been published in [18]. However, there are significant differences. The ISPKF and IEKF use the same iterate method (Gauss-Newton method), and they differ mainly in error propagation. For the conventional IEKF the linear error is propagated using Taylor series expansion, while the statistically linear error of ISPKF is propagated through unscented transformation (for details see [18]). Conversely, the proposed IUKF employs a step length-varying strategy to guard the iterates towards the maximum likelihood solution.

For the problem of state estimation, the computational complexity of UKF is comparable to that of EKF [19] and is in general $\mathbf{O}(L^3)$. Similarly, for each instant $k$, if the iterate is repeated $N$ times, then the IEKF and IUKF will have an equivalent computational cost of $\mathbf{O}(NL^3)$. On average, the calculational burden of IUKF and UKF is within the same order of magnitude if $N$ is not too large.

## III. NUMERICAL SIMULATION AND EXPERIMENT

To demonstrate the performance of the proposed algorithm, the problem of passive target tracking in x-y plane is considered in the following subsections. The state equation is linear and can be represented as

$$\mathbf{x}_{k+1} = F\mathbf{x}_k + G\mathbf{v}_k \tag{32}$$

where

$$F = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$G = \begin{bmatrix} 0.5T^2 & 0 \\ T & 0 \\ 0 & 0.5T^2 \\ 0 & T \end{bmatrix}$$

$$\mathbf{x}_k = [x_k \ \dot{x}_k \ y_k \ \dot{y}_k]^T.$$

$\mathbf{v}_k = [v_{x_k} \ v_{y_k}]^T$ is modeled as zero-mean process noise with covariance matrix $Q_k$. Throughout this paper, the time interval $T$ is 0.5 s.

The measurements consist of bearing $\theta$, bearing rate $\dot{\theta}$, and Doppler rate $\dot{f}_d$, so the measurement equation can be written as [16]

$$\mathbf{z}_k = \begin{bmatrix} \theta_k \\ \dot{\theta}_k \\ \dot{f}_{dk} \end{bmatrix} = \begin{bmatrix} \tan^{-1}(y_k/x_k) \\ (\dot{y}_k x_k - \dot{x}_k y_k)/r_k^2 \\ -(\dot{y}_k x_k - \dot{x}_k y_k)^2/(\lambda r_k^3) \end{bmatrix} + \begin{bmatrix} n_{\theta_k} \\ n_{\dot{\theta}_k} \\ n_{\dot{f}_{dk}} \end{bmatrix}$$

$$\triangleq h(\mathbf{x}_k) + \mathbf{n}_k \tag{33}$$

where $r_k = \sqrt{x_k^2 + y_k^2}$, $\lambda$ denotes the wavelength of received signal, $n_{\theta_k}$, $n_{\dot{\theta}_k}$, and $n_{\dot{f}_{dk}}$ are mutually independent Gaussian noise with covariance matrix $R_k = \mathrm{diag}[\sigma_{n\theta}^2, \sigma_{n_{\dot{\theta}}}^2, \sigma_{n_{\dot{f}_d}}^2]$.

### A. Numerical Simulation

Assume the observer is located at the origin, and the target moves at a nearly constant velocity in the

TABLE I
Robustness Comparison about Several Algorithms

| Percent Convergent Runs, Out of 100 Trials | | | |
|---|---|---|---|
| Algorithms | Simulation 1 | Simulation 2 | Simulation 3 |
| EKF | 100 | 76 | 43 |
| IEKF$_1$ | 100 | 73 | 38 |
| IEKF$_2$ | 100 | 88 | 68 |
| MVEKF | 100 | 85 | 49 |
| UKF | 100 | 91 | 74 |
| IUKF | 100 | 100 | 88 |

*Notes*:

1) In Simulation 1–Simulation 3, the measurement errors are [2e-3 rad, 1e-4 rad/s, 0.5 Hz/s], [5e-3 rad, 2e-4 rad/s, 1 Hz/s] and [1e-2 rad, 5e-4 rad/s, 2 Hz/s], respectively.

2) In each run, the tracking time is 120 s.

3) A run is considered to be convergent only if the RPE (defined in (36)) $< 15\%$ at the end of 120 s.

4) Both IUKF and IEKF$_2$ use the proposed iteration strategy with a maximum iteration number $N = 5$ and $\eta = 0.85$; while IEKF$_1$ uses the traditional iteration strategy with a threshold $V_{\text{th}} = 100$.

2-D plane with initial state

$$\mathbf{x}_0 = [120 \text{ km} \ \ -250 \text{ m/s} \ \ 80 \text{ km} \ \ 100 \text{ m/s}]^T. \tag{34}$$

The initial position estimate of the target can be obtained using the ranging formula

$$r = \lambda\dot{f}_d/\dot{\theta}^2. \tag{35}$$

In the simulations, the process noise covariance matrix is assumed to be $Q_k = \text{diag}[(3 \text{ m/s}^2)^2, (2 \text{ m/s}^2)^2]$. Different measurement errors ([2e-3 rad, 1e-4 rad/s, 0.5 Hz/s]; [5e-3 rad, 2e-4 rad/s, 1 Hz/s]; [0.01 rad, 5e-4 rad/s, 2 Hz/s]) are given to test the algorithm. The evaluation metrics of interest here are the convergence rate and the relative position error (RPE) which is defined as

$$\text{RPE} = \frac{\sqrt{(x_{\text{true}} - \hat{x})^2 + (y_{\text{true}} - \hat{y})^2}}{\sqrt{x_{\text{true}}^2 + y_{\text{true}}^2}} \times 100\% \tag{36}$$

where $(x_{\text{true}}, y_{\text{true}})$ denotes the true position of the target, and $(\hat{x}, \hat{y})$ represents the position estimation.

In each of the three simulations, 100 Monte-Carlo runs are carried out and the results are summarized in Table I.

As shown in Table I, all the filters converge well under high measurement precision; however, with the increase of measurement error, they exhibit a different degree of performance loss. By comparison, the filters under EKF framework are more sensitive to measurement error than those under UKF framework because of linearization effect. As can be seen from the table, the iterated EKF (IEKF$_1$ and IEKF$_2$) is greatly affected by different iteration strategy. Given the proper termination criteria, IEKF$_2$ can usually obtain better performance than EKF and MVEKF, which is in accord with the theoretical analysis because in essence, EKF and MVEKF are two special forms of the iterated EKF, with no iteration and just a single iteration, respectively. However, an improper choice of the threshold, say IEKF$_1$, may lead to severe performance degradation.

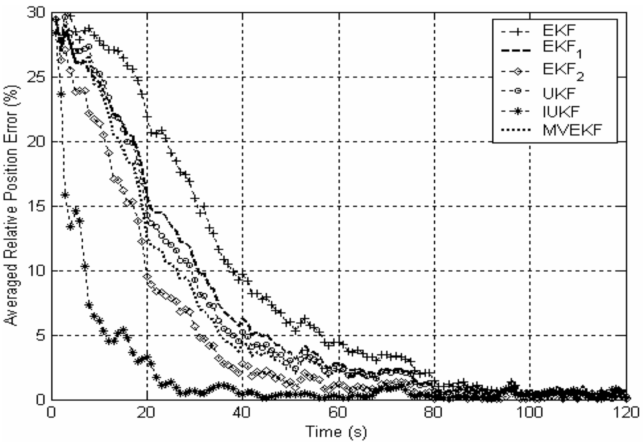

Fig. 1. Comparison of tracking results for different algorithms with measurement precision [5e-3 rad, 2e-4 rad/s, 1 Hz/s].

In the simulations, the termination criteria can generally be met only after 2–3 iterates, meaning that the computational cost of the proposed IUKF is about three times as much as that of UKF on average.

The optimal setting of the scaling parameters ($\alpha$, $\beta$ and $\kappa$) in UKF is problem specific, but usually $\kappa$ will be chosen as $L + \kappa > 0$ to guarantee positive semidefiniteness of the covariance matrix and nonnegativity of the weights. To avoid sampling nonlocal effects, the parameter $\alpha$ should be chosen as a small value when the nonlinearities are strong. A detailed discussion on this problem can be found in [4], [14]. In our simulations, the parameters are typically chosen as $\alpha = 0.2$, $\beta = 2$, and $\kappa = 0$. Nevertheless, it is found that the algorithm is not sensitive to the exact values chosen for these parameters as long as they result in a numerically well-behaved set of sigma points and weights. It is also found in the simulations that the choice of $\eta$ is not crucial to the iterative convergence when $0.5 < \eta \leq 1$; however, an improper choice of $\eta$ ($\eta > 1$) may lead to instability of the filter.

To provide more intuitionistic performance comparison of different algorithms, the averaged RPE of simulation 2 is plotted in Fig. 1. It should be noted that Fig. 1 reflects the statistical performance of Monte-Carlo trials, and only those convergent trials (with RPE $<$ 15%) are counted in.

As illustrated in Fig. 1, the UKF is superior to the EKF both in convergence time and tracking error. In the sense of statistical average, it seems that MVEKF has a lower tracking error than UKF, but this does not mean that MVEKF has the same or even better performance because more divergent trials of MVEKF are excluded in the simulation. In comparison with their noniterated counterparts, the iterated filters

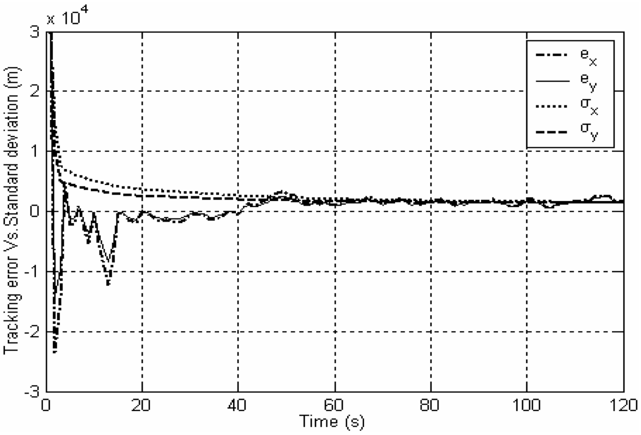

Fig. 2. Comparison between standard deviations and tracking errors.

usually need shorter convergence time, which makes the iterated filters more appealing especially when the initial error is very large. Clearly, IUKF outperforms all the other filters both in convergence speed and tracking accuracy, though those divergent tracks (of EKF, IEKF$_1$, IEKF$_2$, UKF, and MVEKF) are excluded.

In practical use, more attention may be paid to the robustness of the algorithm from the viewpoint of each single trial. In this case, the advantage of IUKF is more significant.

To investigate the stability of the proposed IUKF, the state standard deviations and the tracking errors of an illustrative run are shown in Fig. 2. The tracking errors along different coordinate axes are defined as

$$e_x = x_{\text{true}} - \hat{x}, \qquad e_y = y_{\text{true}} - \hat{y} \tag{37}$$

while the standard deviations are given as

$$\sigma_x = \sqrt{p_{11}}, \qquad \sigma_y = \sqrt{p_{33}} \tag{38}$$

where $p_{ii}$ denotes the diagonal element of the covariance matrix $P$.

As shown in Fig. 2, the tracking errors are very close to the standard deviations after the filter converges, meaning that the tracking errors can be accurately described by the standard deviations. From these results it can be inferred that once the standard deviations reach a stable status, so do the tracking errors. By comparison, both the estimate error and the standard deviation along x-axis are larger than those along y-axis during the initial part of track, which can be interpreted from the fact that the tracking precision is closely related to the distance (between the target and the observer); the longer the range is, the higher the tracking errors are under the same condition.

### B. Outfield Experiment

To further validate its performance, the proposed algorithm is applied to the outfield experiment. The experiment scenario is depicted as follows. A car carrying a radar emitter (considered as the target which is being tracked) runs on the airport course (seen as the trajectory) from one end to the other end, as illustrated in Fig. 3. The observer is located at the origin, and the vertical range $d$ from the observer to the trajectory is about 520 m. The bearing and its change rate are obtained via radar receiver and tracking television, respectively; while the Doppler rate is estimated indirectly from the received signal. Specifically, the true position of the target is provided by the differential GPS.

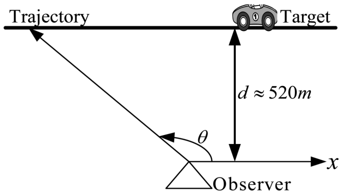

Fig. 3. Relative geometrical position between target and observer.

In the experiment, the emitter moves at a velocity of about 15 m/s from the right to the left of the trajectory, and the statistical results indicate that the standard deviations of the measurements are 0.36 mrad, 2.5 mrad/s, and 2.8 Hz/s, respectively. All six filters[3] (EKF, IEKF$_1$, IEKF$_2$, UKF, IUKF, and MVEKF) are used to track the target, and the results are shown in Fig. 4 and Fig. 5.

From Fig. 4 and Fig. 5, it is clearly seen that the iterated filters (IEKF$_1$, IEKF$_2$, and IUKF) respond to new measurement more quickly than their noniterated counterparts (EKF and UKF); that is why an iterated filter has a faster convergent speed if the measurements are accurate enough. It is also seen from the figures that both UKF and MVEKF perform better than the conventional EKF because these two filters have more accurate approximation to the first two order moments (of the state to be estimated). As the experiment indicates, for the iterated EKF, iteration strategy may have a great effect on the results, so a proper choice of the termination threshold is crucial to improve the estimate performance.

Obviously, the IUKF has a faster convergence speed and lower tracking error than all the other filters, which is consistent with the results in the previous simulations. The distinct advantages of IUKF come mainly from two aspects: one is that the standard UKF is superior to EKF in estimate accuracy, so the former exhibits better performance in both simulation and experiment. The other reason is that the iterated UKF has a quicker response than UKF, which leads to a faster convergence speed.

[3] In the experiment, IUKF and IEKF$_2$ use the same iteration strategy as that in the simulations with $N = 5$ and $\eta = 0.8$; while IEKF$_1$ uses the general iteration strategy with $V_{\text{th}} = 50$.

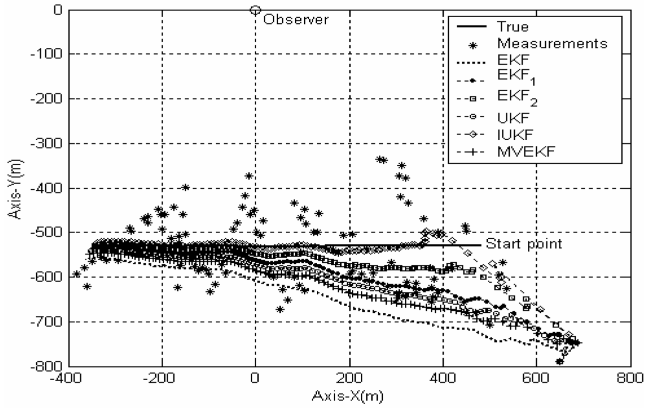

Fig. 4. Comparison of tracking results, measurements plotted with asterisks are obtained directly from $(x,y) = (r\cos(\theta), r\sin(\theta))$.

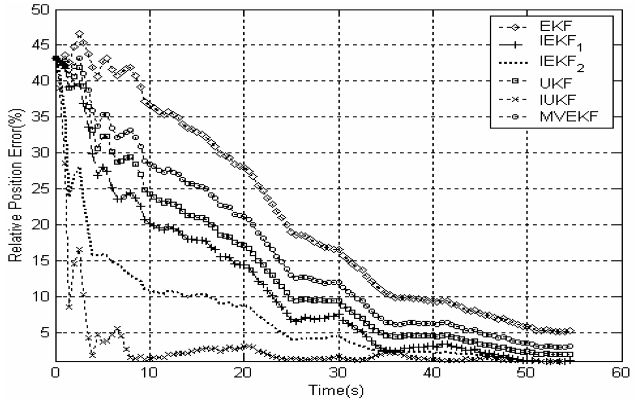

Fig. 5. Comparison of relative position error.

## IV. CONCLUSION

Comparisons of the conventional algorithms used in passive target tracking have been made in the presented study, on which basis an IUKF was developed to improve the tracking performance. The IUKF adopts a step length-varying iteration strategy to estimate the filtering state and covariance matrix, so the termination condition can be readily satisfied within a few iterates and the computational complexity is close to that of UKF. Numerical simulation and experiment results show that the IUKF is much more robust than traditional approaches except for its shorter convergence time and lower tracking error, indicating that the IUKF is very suitable for practical application. The presented method can also be extended to other problems of nonlinear filtering.

## APPENDIX A. DERIVATION OF EQUATION (31)

For a positive definite covariance matrix sequence $\{P_j\}$, where $P_j = [p_{m,n}^{(j)}] \in R^{L\times L}$, if the inequality $P_{j+1} < P_j$ holds for any $j = 1,2,\ldots$, it can be proved that the limit of the matrix sequence $\{P_j\}$ exists. The proof is given in two steps. 1) The limit of the diagonal element $p_{m,m}^{(j)}$ $(m = 1,2,\ldots,L)$ exists. 2) The limit of arbitrary element $p_{m,n}^{(j)}$ $(m,n = 1,2,\ldots,L)$ exists.

PROOF OF 1 Let $\mathbf{e}_m = [0,\ldots,0,1,0,\ldots 0]^T$ be an $L$-dimensional unity vector with 1 in the $m$th dimension, then based on the given condition $P_{j+1} < P_j$, we have $\mathbf{e}_m^T P_{j+1}\mathbf{e}_m < \mathbf{e}_m^T P_j \mathbf{e}_m$, i.e.,

$$p_{m,m}^{(j+1)} < p_{m,m}^{(j)}.$$

From the above inequality, we know that $\{p_{m,m}^{(j)}\}$ is a monotonous decreasing sequence and it is also bounded by $0 \le p_{m,m}^{(j)} \le p_{m,m}^{(1)}$, so the limit of the sequence exists, i.e., $\lim_{j\to\infty} p_{m,m}^{(j)} = c_{m,m}$ (where $c_{m,m}$ is a constant).

PROOF OF 2 For any unity vectors $\mathbf{e}_m$ and $\mathbf{e}_n$, if $P_{j+1} < P_j$, then we have

$$(\mathbf{e}_m^T + \mathbf{e}_n^T)P_{j+1}(\mathbf{e}_m + \mathbf{e}_n) < (\mathbf{e}_m^T + \mathbf{e}_n^T)P_j(\mathbf{e}_m + \mathbf{e}_n).$$

Also note that $P_{j+1}$ and $P_j$ are symmetric matrices, hence

$$p_{m,m}^{(j+1)} + p_{n,n}^{(j+1)} + 2p_{m,n}^{(j+1)} < p_{m,m}^{(j)} + p_{n,n}^{(j)} + 2p_{m,n}^{(j)}.$$

Using the results in 1), we immediately obtain $\lim_{j\to\infty} p_{m,n}^{(j)} = c_{m,n}$.

From 1) and 2), we have $\lim_{j\to\infty} P_j = \lim_{j\to\infty}[p_{m,n}^{(j)}] = [c_{m,n}] = C$.

## APPENDIX B. DERIVATION OF INEQUALITY

For the problem of filtering estimate, the purpose of measurement update is to find a better state estimate and corresponding covariance using the given information. Consider $\hat{\mathbf{x}}_j$ and $\mathbf{y}$ as realization of independent random vectors with multivariate normal distributions, so we have

$$\hat{\mathbf{x}}_j \sim \mathcal{N}(\mathbf{x}, P_j), \qquad \mathbf{y} \sim \mathcal{N}(h(\mathbf{x}), R) \qquad (39)$$

where $\mathbf{x}$ and $\hat{\mathbf{x}}_j$ denote the current state and its estimate, respectively; $\mathbf{y}$ represents the observation, $h(\mathbf{x})$ models a noiseless measurement.

For convenience, the two vectors are formed to a single augmented one

$$\mathbf{z} = \begin{bmatrix} \mathbf{y} \\ \hat{\mathbf{x}}_j \end{bmatrix}, \qquad g(\mathbf{x}) = \begin{bmatrix} h(\mathbf{x}) \\ \mathbf{x} \end{bmatrix}. \qquad (40)$$

It follows from (39) and the independent assumption that

$$\mathbf{z} \sim \mathcal{N}(g(\mathbf{x}), S), \qquad S = \begin{bmatrix} R & \mathbf{0} \\ \mathbf{0} & P_j \end{bmatrix}. \qquad (41)$$

The update problem becomes the one that computing a state estimation $\mathbf{x}^*$ and corresponding covariance given $\mathbf{z}$, $g(\mathbf{x})$, and $S$.

If $\mathbf{x}$ is replaced by a free variable $\xi$, we get the following likelihood function

$$\Lambda(\xi) = \text{const} \cdot \exp[-\tfrac{1}{2}(\mathbf{z} - g(\xi))^T S^{-1}(\mathbf{z} - g(\xi))]. \tag{42}$$

The maximum likelihood estimate (optimal estimate) for $\mathbf{x}$ are given by

$$\mathbf{x}^* = \arg\max[\Lambda(\xi)]. \tag{43}$$

The equivalent formulation of (43) is

$$\mathbf{x}^* = \arg\min[q(\xi)] \tag{44}$$

where

$$q(\xi) = \tfrac{1}{2}(\mathbf{z} - g(\xi))^T S^{-1}(\mathbf{z} - g(\xi)). \tag{45}$$

It is not easy to obtain the exact $\xi^*$ to satisfy (44), but if we have

$$q(\xi_{j+1}) < q(\xi_j) \tag{46}$$

we say that $q(\xi_{j+1})$ is closer to the maximum likelihood surface than $q(\xi_j)$, equivalently $\xi_{j+1}$ has a more accurate approximation than $\xi_j$ to the optimal solution.

According to (45), we have the following inequality

$$\begin{aligned}&[\mathbf{y} - h(\xi_{j+1})]^T R^{-1}[\mathbf{y} - h(\xi_{j+1})] + (\hat{\mathbf{x}}_j - \xi_{j+1})^T P_j^{-1}(\hat{\mathbf{x}}_j - \xi_{j+1})\\ &\quad < [\mathbf{y} - h(\xi_j)]^T R^{-1}[\mathbf{y} - h(\xi_j)] + (\hat{\mathbf{x}}_j - \xi_j)^T P_j^{-1}(\hat{\mathbf{x}}_j - \xi_j).\end{aligned} \tag{47}$$

If the sequence $\{\xi_j\}$ is generated from the iterates, then we can immediately obtain (30).

**RONGHUI ZHAN**
**JIANWEI WAN**
**Laboratory of Signal Processing**
**School of Electronic Science and Engineering**
**National University of Defense Technology**
**Changsha, Hunan**
**P.R. China, 410073**
**E-mail: (zhanrh@nudt.edu.cn)**

## REFERENCES

[1] Gelb, A.
*Applied Optimal Estimation.*
Cambridge, MA: M.I.T. Press, 1974.

[2] Bar-Shalom, Y., Li, X. R., and Kirubarajan, T.
*Estimation with Application to Tracking and Navigation: Theory, Algorithm, and Software.*
New York: Wiley, 2001.

[3] Wan, E. A., and van der Merwe, R.
The unscented Kalman filter for nonlinear estimation.
In *Proceedings of IEEE Symposium 2000* (AS-SPCC), Lake Louise, Alberta, Canada, Oct. 2000, 153–158.

[4] van der Merwe, R.
Sigma-point Kalman filters for probabilistic inference in dynamic state-space models.
Ph.D. dissertation, Oregon Health Sciences University, Portland, OR, 2004.

[5] Song, T. L., and Speyer, J.
A stochastic analysis of a modified gain extended Kalman filter with applications to estimation with bearings only measurements.
*IEEE Transactions on Automatic Control*, **AC-30**, 10 (Oct. 1985), 940–949.

[6] Galkowski, P. J., and Islam, M. A.
An alternative derivation of modified gain function of Song and Speyer.
*IEEE Transactions on Automatic Control*, **36**, 11 (Nov. 1991), 1322–1326.

[7] Guo, F., Sun, Z., and Huangfu, K.
A modified covariance extended Kalman filtering algorithm in passive location.
In *Proceedings of the IEEE International Conference on Robotics, Intelligent Systems and Signal Processing*, Oct. 2003, 307–311.

[8] Julier, S. J., and Uhlmann, J. K.
A new extension of the Kalman filter to nonlinear systems.
In *Proceedings of AeroSense: The 11th International Symposium on Aerospace/Defence Sensing, Simulation and Controls*, Orlando, FL, 1997.

[9] Julier, S. J., and Uhlmann, J. K.
A new method for the nonlinear transformation of means and covariances in filters and estimators.
*IEEE Transactions on Automatic Control*, **45**, 3 (Mar. 2000), 477–482.

[10] Vijayakumar, C., and Rajagopal, R.
Passive target tracking by unscented filters.
In *Proceedings of the IEEE International Conference on Industrial Technology*, Jan. 2000, 129–133.

[11] Julier, S. J., and Uhlmann, J. K.
Unscented filtering and nonlinear estimation.
*Proceedings of the IEEE*, **92**, 3 (Mar. 2004), 401–422.

[12] Yi X., and Li, L.
Single observer bearings-only tracking with the unscented Kalman filter.
In *Proceedings of the IEEE International Conference on Communications, Circuits and Systems*, vol. 2, June 2004, 901–905.

[13] van der Merwe, R., Wan, E. A., and Julier, S. J.
Sigma-point Kalman filters for nonlinear estimation and sensor-fusion: Applications to integrated navigation.
In *Proceedings of AIAA Guidance Navigation and Controls Conference*, Providence, RI, 2004.

[14] Julier, S. J.
The scaled unscented transformation.
In *Proceedings of the American Control Conference*, May 2002, 4555–4559.

[15] Bell, B. M., and Cathey, F. W.
The iterated Kalman filter update as a Gauss-Newton method.
*IEEE Transactions on Automatic Control*, **38**, 2 (Feb. 1993), 294–297.

[16] Gong, X.
The study of challenge technology of single observer passive location and tracking using frequency rate and differential direction of arrival.
Ph.D. dissertation, National University of Defense Technology, Changsha, P.R. China, 2004.

[17] Johnston, L. A., and Krishnamurthy, V.
Derivation of a sawtooth iterated extended Kalman smoother via the AECM algorithm.
*IEEE Transactions on Signal Processing*, **49**, 9 (Sept. 2001), 1899–1909.

[18] Sibley, G., Sukhatme, G., and Matthies, L.
The iterated sigma point Kalman filter with applications to long range stereo. In *Proceedings of Robotics: Science and Systems*, Philadelphia, PA, Aug. 2006.

[19] van der Merwe, R., and Wan, E. A.
The square-root unscented Kalman filter for state and parameter-estimation.
In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing* (ICASSP), Salt Lake City, UT, May 2001, 3461–3464.

# Localization and Observability of Aircraft via Doppler Shifts

**The linear trajectory of an airplane with constant velocity is obtained from Doppler shifts at stationary sensors, each yielding a corresponding time of closest approach, closest distance, and speed [3, 8]. Based on these inferences and on the sensors' locations, three sensors yield two candidate trajectories, and four sensors yield one—provided that the locations span $\mathbb{R}^2$ and $\mathbb{R}^3$, respectively: constituting observability for the problem at hand.**

## I. INTRODUCTION

The bombilation of airplanes resolves into spectral bands, whose frequencies manifest Doppler shifts [4, (11.1.1)]. Such shifts allow inference of the velocities of stars. For vicinal objects, however, they yield supplemental insights [3, 8]. Note that the results of this manuscript are applicable both to acoustic and to radar modalities [8].

For an airplane with constant velocity, by analogy to the "point-slope formula" its (linear) trajectory may be written, in vector notation, as

$$(A_1 t + B_1, A_2 t + B_2, A_3 t + B_3) \tag{1}$$

with the $A$s and $B$s denoting constants and with $t$ denoting time; the airplane's speed equals $\sqrt{A_1^2 + A_2^2 + A_3^2}$.

For such trajectories, Doppler shifts at a sensor yield the following conclusiunculae: the distance, or range $\tilde{r}$ of closest approach, the corresponding time of closest approach $\tilde{t}$, and the airplane's speed $\tilde{s}$ [3, 8]. In [8] $\tilde{r}$, $\tilde{s}$, and $\tilde{t}$ are found from the roots of a cubic equation, based on the frequencies at three equally spaced time points and on the asymptotic frequencies. In [3], $\tilde{r}$, $\tilde{s}$, and $\tilde{t}$ are obtained by least-square fitting to a parameterized curve, involving of the order of 100 frequency measurements.

In this manner, one sensor delimits the linear candidate trajectories to those of speed $\tilde{s}$ tangent, at time $\tilde{t}$, to a sphere of radius $\tilde{r}$ centered on the sensor. A method is developed herein for fully resolving the trajectory (1), based on the $\tilde{r}$s, $\tilde{s}$s, $\tilde{t}$s, and sensor coordinates from small numbers of sensors. Horizontality of the trajectory is not assumed [1, 3, 5]. The intrinsic advantages of statistical solutions notwithstanding [3, 7], the present approach, the like time-difference-of-arrival (TDOA) method [7, Appendix A], is geometrical. Errors in $\tilde{r}$, $\tilde{s}$, $\tilde{t}$ and in the sensor coordinates are assumed to be negligible, and, accordingly, all sensors' estimates for $\tilde{s}$ are taken to be equal. Sensors are also assumed to share a common clock.

The natural question of observability is "What is the minimum number of sensors capable, in this context, of identifying the trajectory?" Fig. 1 illustrates the capabilities of two sensors; corresponding spheres touching the trajectory are depicted. The points of closest approach, $\tilde{\mathbf{p}}_1$ and $\tilde{\mathbf{p}}_2$ are part and parcel of the trajectory. However, even if the latter were known, rotations about the axis connecting the locations of the sensors would plainly generate additional solutions, with the closest-approach points tracing circles on the spheres (Fig. 1). Thus, observability demands additional sensors.

In Section IIIA two trajectories are obtained from the $\tilde{r}$s, $\tilde{s}$, $\tilde{t}$s, and coordinates of three noncollinear sensors. This ambivalence is consistent with the observation that the reflection of an acceptable trajectory in the plane of the sensors evidently constitutes another acceptable trajectory. In Section IIIB a unique trajectory is obtained from four sensors in general position. Formulae for these trajectories are derived in Section III.

Section II introduces four parameters which specify lines in $\mathbb{R}^3$. The current solutions evaluate these parameters, based on sensors' data, yielding the line containing the trajectory. Section II also illustrates the facile reintroduction of time, completing the identification.

## II. LINES IN $\mathbb{R}^3$

A line in $\mathbb{R}^3$ is an elementary affine variety $\vec{A}\sigma + \vec{B}$, where $\vec{A}$ and $\vec{B}$ are arbitrary 3-vectors (or triples of real numbers) and where $\sigma \in \mathbb{R}$. Apart from $\sigma$, this

Manuscript received April 4, 2005; revised December 20, 2006; released for publication May 12, 2007.

IEEE Log No. T-AES/43/3/908422.

Refereeing of this contribution was handled by C. Jauffret.

This work was supported by the U.S.D.O.E. under Contract W-7405-ENG-36, to the University of California and by an NNSA Grant "Distributed Sensor Networks with Collective Computation," A. Mielke, PI.

0018-9251/07/$25.00 © 2007 IEEE