

# DOA Tracking via Simultaneous Angle-Source Update (SASU)

Xunzhang Gao\*, Jason Filos<sup>†</sup> and Wei Dai<sup>†</sup>

\*College of Electronic Science and Engineering, National University of Defense Technology, P.R.China

<sup>†</sup>Department of Electrical and Electronic Engineering, Imperial College London

\*gaoxunzhang@gmail.com; <sup>†</sup>{jason.filos03, wei.dai1}@imperial.ac.uk

**Abstract**—This paper focuses on the DOA (direction of arrival) tracking problem with uniform linear arrays. In the literature, there are two common approaches: one is to first use MUSIC based algorithms to estimate the signal subspace and then calculate the arrival angles from it; the other is to use Kalman filtering (KF) techniques to track the angles directly. This paper emphasizes on the fast movement case and hence follows the second approach. The key feature of this work is to simultaneously update the arrival angles and the source signals in the KF step by treating the source signals as a function of the arrival angles. Hence the term Simultaneous Angle-Source Update (SASU). On the theoretical front, we show the deep connections between SASU and the two common approaches in the literature. On the more practical side, a Newton method is developed to solve the optimization problem in the KF step. Simulations demonstrate that SASU is more robust and accurate than the benchmark algorithms, especially when uncertain and fast moving sources are concerned.

**Index Terms**—DOA tracking, Kalman filtering, Newton method.

## III. INTRODUCTION

Direction of Arrival (DOA) tracking using sensor arrays has been a main topic in array signal processing due to its wide applications in radar, sonar, navigation, communication, and remote sensing. Various DOA tracking algorithms have been developed in the literature in the past decades.

There are two popular approaches to track arrival angles. The first approach is characterized by a two-step procedure that estimates first the signal subspace and then the arrival angles. This approach is partly motivated by the huge success of MUSIC [1], ESPRIT [2], and other similar methods that separate the signal and noise subspaces. Since arrival angles somewhat determine the signal subspace, DOA estimation can be achieved by using the estimated signal subspace. The advantage of this approach is the minimum requirement in prior information, for example, the number of source signals and the noise variance can be estimated concurrently. However, multiple measurements from different time instants are grouped together for subspace estimation. When the arrival angles are not static, the estimated subspaces may exhibit severe smearing. Particularly designed for tracking non-static subspaces, the projection approximation subspace tracking (PAST) algorithm [3] and various similar methods [4], [5] have been proposed. Nevertheless, the smearing phenomenon

exists especially when the angular speeds of the sources are large compared to the sampling rate. Furthermore, in certain circumstances, suboptimal mechanisms for angle estimation (from the subspace) and data association have to be performed for real applications. These extra steps introduce unnecessary computational complexity and performance loss.

The other approach is marked by applying tracking algorithms, for example, Kalman filtering (KF), directly to arrival angles [6], [7], [8], [9], [10], [11], [12]. This approach often requires more prior information, including proper statistical modeling, knowledge about the number of sources, and reasonable initial estimates of arrival angles (which could be obtained from MUSIC-based algorithms). On the other hand, this approach efficiently uses statistical information about the arrival angles and angular speeds, and is therefore more suitable for tracking fast-moving sources. Since the measurement vector is a nonlinear function of the arrival angles, nonlinear Kalman filters, for example extended Kalman filters (EKF) [9], [10], unscented Kalman filters (UKF) and particle Kalman filters (PKF), have been proposed. The idea of EKF is to use the first order Taylor expansion to linearize the nonlinear system [9], [10]. In contrast, UKF and PKF work with the nonlinear system directly but generate multiple point estimates which are combined to give the final estimate [12]. All these techniques are often employed in practice.

This paper emphasizes on the fast movement case<sup>1</sup> and hence follows the Kalman approach. The key feature of this work is to simultaneously update the arrival angles and the source signals in the KF step by treating the source signals as a function of the arrival angles: when the arrival angles are updated, the source signals are automatically updated as well. Hence the term Simultaneous Angle-Source Update (SASU). With this treatment, the cost function in the KF step involves only the arrival angles. A Newton method is designed to efficiently minimize the cost function. Simulations demonstrate that SASU is more robust and accurate than the benchmark algorithms, especially when fast movements and uncertain sources (i.e. the variances of the source signals are large) are concerned. Another unique feature is the explicit inclusion of the source signals in the Kalman state. The effects of this statistical modeling are two folded. Firstly, it connects the standard formulations for signal subspace tracking [3]

This work is partly supported by the University Defence Research Centre (UDRC), UK.

<sup>1</sup>The technique developed here can be applied to slow moving or static sources as well.

and EKF [9], [10] together into one framework. Secondly, it actually motivates the SASU algorithm design.

The remainder of this paper is organized as follows. Section IV introduces the system model and the standard Kalman formulation. The proposed SASU method is detailed in Section V, where the relation between SASU and other commonly used methods is briefly discussed. Numerical comparisons are presented in Section VI to demonstrate the performance improvement of SASU.

#### IV. SYSTEM MODEL AND STATE EQUATION

We consider the DOA tracking problem with a uniform linear array (ULA) of  $m$  antennas assuming far-field and narrow-band source signals. For any given source signal  $s \in \mathbb{C}$  at an arrival angle  $\theta \in [-\pi/2, \pi/2]$ , the measurement vector  $\mathbf{y} \in \mathbb{C}^m$  can be written as [13]

$$\mathbf{y} = \mathbf{a}(\theta) s + \mathbf{w},$$

where the array direction vector  $\mathbf{a}(\theta)$  is given by

$$\mathbf{a}(\theta) = \left[ 1, e^{-j2\pi(d/\lambda)\sin\theta}, \dots, e^{-j2\pi(d/\lambda)(m-1)\sin\theta} \right]^T, \quad (1)$$

$d$  is the distance between adjacent antennas in the ULA,  $\lambda$  is the wavelength, the superscript  $T$  denotes a transpose, and  $\mathbf{w} \in \mathbb{C}^m$  is the complex Gaussian noise vector. Suppose that there are  $n$  sources present, then the measurement vector is given by

$$\mathbf{y} = \mathbf{A}(\boldsymbol{\theta}) \mathbf{s} + \mathbf{w}, \quad (2)$$

where  $\mathbf{s} = [s_1, \dots, s_n]^T$  and  $\boldsymbol{\theta} = [\theta_1, \dots, \theta_n]^T$  represent the source signals and the arrival angles respectively, and

$$\mathbf{A}(\boldsymbol{\theta}) = [\mathbf{a}(\theta_1), \dots, \mathbf{a}(\theta_n)] \in \mathbb{C}^{m \times n} \quad (3)$$

is the array direction matrix. It is worth to note that the model and the method developed in this paper can be extended to general one-dimensional or two-dimensional antenna arrays by specifying the array direction vector  $\mathbf{a}(\theta)$  according to the particular structure of the antenna arrays.

For the DOA tracking problem, both arrival angles and source signals are dynamic. At the time instant  $t$ , one has  $\mathbf{y}(t) = \mathbf{A}(\boldsymbol{\theta}(t)) \mathbf{s}(t) + \mathbf{w}(t)$ , which gives the observation equation in the Kalman filter. There are multiple choices for the state equation in the Kalman filter, depending on the targeted applications. One popular choice is to include not only the arrival angles but also the angular speeds into the state equation. More specifically, let  $\dot{\theta}_i(t)$  be the angular speed for the  $i^{\text{th}}$  source signal, and  $\dot{\boldsymbol{\theta}}(t) = [\dot{\theta}_1(t), \dots, \dot{\theta}_n(t)]^T$ . The state equation can be written as

$$\begin{bmatrix} \boldsymbol{\theta}(t) \\ \dot{\boldsymbol{\theta}}(t) \end{bmatrix} = \underbrace{\begin{bmatrix} \mathbf{I}_n & T_s \mathbf{I}_n \\ \mathbf{0} & \mathbf{I}_n \end{bmatrix}}_F \begin{bmatrix} \boldsymbol{\theta}(t-1) \\ \dot{\boldsymbol{\theta}}(t-1) \end{bmatrix} + \mathbf{u}_\theta(t), \quad (4)$$

where  $\mathbf{I}_n$  is the  $n \times n$  identity matrix,  $T_s$  is the inverse of the sampling frequency,<sup>2</sup> and  $\mathbf{u}_\theta \in \mathbb{R}^{2n}$  is the innovation vector,

<sup>2</sup>That is, the time difference between the discrete time instants  $t-1$  and  $t$  is  $T_s$ .

which is assumed to have zero-mean and covariance matrix [13], [12]

$$\boldsymbol{\Sigma}_\theta = \sigma_\theta^2 \begin{bmatrix} \frac{T_s^3}{3} \mathbf{I}_n & \frac{T_s^2}{2} \mathbf{I}_n \\ \frac{T_s^2}{2} \mathbf{I}_n & T_s \mathbf{I}_n \end{bmatrix}. \quad (5)$$

For notational convenience, we use  $\bar{\boldsymbol{\theta}}(t)$  to denote the vector  $[\boldsymbol{\theta}^T(t), \dot{\boldsymbol{\theta}}^T(t)]^T$ . The array direction matrix can be written as a function of  $\bar{\boldsymbol{\theta}}$ , i.e.,  $\mathbf{A}(\bar{\boldsymbol{\theta}})$ , as well.

*Remark 1.* Note that the system model here can be easily extended to the cases  $\bar{\boldsymbol{\theta}}(t) := \boldsymbol{\theta}(t)$  and  $\bar{\boldsymbol{\theta}}(t) = [\boldsymbol{\theta}(t), \dot{\boldsymbol{\theta}}(t), \ddot{\boldsymbol{\theta}}(t)]$  where  $\ddot{\boldsymbol{\theta}}(t)$  is the angular acceleration vector at the time instant  $t$ .

It is clear that the measurement vector  $\mathbf{y}(t)$  is not a linear function of the state vector  $\bar{\boldsymbol{\theta}}(t)$ . One way to apply Kalman filtering is a linearization via Taylor expansion. More specifically, suppose that the estimate  $\hat{\boldsymbol{\theta}}(t-1)$  at time instant  $t-1$  is available. From the state equation (4), a prediction  $\hat{\boldsymbol{\theta}}(t|t-1)$  can be obtained via  $\hat{\boldsymbol{\theta}}(t|t-1) = \mathbf{F} \hat{\boldsymbol{\theta}}(t-1)$ . An estimate of the source signal is given by  $\hat{\mathbf{s}}(t|t-1) = \mathbf{A}^\dagger(\hat{\boldsymbol{\theta}}(t|t-1)) \mathbf{y}(t)$  where the superscript  $\dagger$  denotes the pseudo-inverse. Define

$$\mathbf{G} = \left. \frac{\partial}{\partial \bar{\boldsymbol{\theta}}} (\mathbf{A}(\bar{\boldsymbol{\theta}}) \mathbf{s}) \right|_{\hat{\boldsymbol{\theta}}(t|t-1), \hat{\mathbf{s}}(t|t-1)}.$$

Then the observation equation can be approximated by the linear form

$$\mathbf{y}(t) = \mathbf{A}(\hat{\boldsymbol{\theta}}(t|t-1)) \hat{\mathbf{s}}(t) + \mathbf{G} \cdot (\bar{\boldsymbol{\theta}}(t) - \hat{\boldsymbol{\theta}}(t-1)) + \mathbf{w}(t).$$

From it, one can directly apply the Kalman update equation to estimate  $\bar{\boldsymbol{\theta}}(t)$ .

#### V. SIMULTANEOUS ANGLE-SOURCE UPDATE

##### A. The optimization framework

While a system with observation and state equations (2) and (4) is commonly considered in the literature, it is reasonable to add source signals

$$\mathbf{s}(t) = \mathbf{s}(t-1) + \mathbf{u}_s(t), \quad (6)$$

where  $\mathbf{u}_s(t)$  is a complex Gaussian innovation vector, into the state equation. The motivation of this model is to consider the uncertainty of the source signals due to scattering, shielding, and reflection changes, where the level of uncertainty is described by the covariance matrix of  $\mathbf{u}_s(t)$ . Note that  $\mathbf{s}(t)$  and  $\mathbf{u}_s(t)$  are complex vectors. It is convenient to rewrite the dynamic system in terms of real vectors. Define

$$\begin{aligned} \bar{\mathbf{s}}(t) &= \begin{bmatrix} \text{Re}(\mathbf{s}(t)) \\ \text{Im}(\mathbf{s}(t)) \end{bmatrix}, \quad \bar{\mathbf{u}}_s(t) = \begin{bmatrix} \text{Re}(\mathbf{u}_s(t)) \\ \text{Im}(\mathbf{u}_s(t)) \end{bmatrix}, \\ \bar{\mathbf{y}}(t) &= \begin{bmatrix} \text{Re}(\mathbf{y}(t)) \\ \text{Im}(\mathbf{y}(t)) \end{bmatrix}, \quad \bar{\mathbf{w}}(t) = \begin{bmatrix} \text{Re}(\mathbf{w}(t)) \\ \text{Im}(\mathbf{w}(t)) \end{bmatrix}, \quad \text{and} \\ \bar{\mathbf{A}}(\bar{\boldsymbol{\theta}}(t)) &= \begin{bmatrix} \text{Re}(\mathbf{A}(\bar{\boldsymbol{\theta}}(t))) & -\text{Im}(\mathbf{A}(\bar{\boldsymbol{\theta}}(t))) \\ \text{Im}(\mathbf{A}(\bar{\boldsymbol{\theta}}(t))) & \text{Re}(\mathbf{A}(\bar{\boldsymbol{\theta}}(t))) \end{bmatrix}. \end{aligned}$$

Then one has

$$\begin{aligned}\bar{\mathbf{s}}(t) &= \bar{\mathbf{s}}(t-1) + \bar{\mathbf{u}}_s(t), \\ \bar{\boldsymbol{\theta}}(t) &= \mathbf{F}\bar{\boldsymbol{\theta}}(t-1) + \mathbf{u}_\theta(t), \text{ and} \\ \bar{\mathbf{y}}(t) &= \bar{\mathbf{A}}(\bar{\boldsymbol{\theta}}(t))\bar{\mathbf{s}}(t) + \bar{\mathbf{w}}(t).\end{aligned}\quad (7)$$

We assume that  $\bar{\mathbf{u}}_s(t) \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_s)$  where  $\boldsymbol{\Sigma}_s = \sigma_s^2 \mathbf{I}_{2n}$ ,  $\mathbf{u}_\theta(t) \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_\theta)$  where  $\boldsymbol{\Sigma}_\theta$  is given in (5), and  $\bar{\mathbf{w}}(t) \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_w)$  where  $\boldsymbol{\Sigma}_w = \sigma_w^2 \mathbf{I}_{2m}$ .

Given the dynamic system in (7), let us discuss the Kalman update step. That is, given  $\bar{\mathbf{y}}(t)$ , the predictions  $\hat{\mathbf{s}}(t|t-1) = \hat{\mathbf{s}}(t-1)$ , and  $\hat{\boldsymbol{\theta}}(t|t-1) = \mathbf{F}\hat{\boldsymbol{\theta}}(t-1)$ , one needs to estimate  $\hat{\mathbf{s}}(t)$  and  $\hat{\boldsymbol{\theta}}(t)$ . Motivated by the well known fact that the standard Kalman estimate coincides with the maximum a posteriori (MAP) estimate, we choose the cost function as the posteriori probability, i.e.,

$$\begin{aligned}& (\hat{\mathbf{s}}(t), \hat{\boldsymbol{\theta}}(t)) \\ &= \arg \max_{\bar{\mathbf{s}}, \bar{\boldsymbol{\theta}}} \Pr(\bar{\mathbf{s}}, \bar{\boldsymbol{\theta}} | \bar{\mathbf{y}}(t), \hat{\mathbf{s}}(t|t-1), \hat{\boldsymbol{\theta}}(t|t-1)).\end{aligned}$$

To simplify the notations, let  $\bar{\mathbf{s}}^0 = \hat{\mathbf{s}}(t|t-1)$  and  $\bar{\boldsymbol{\theta}}^0 = \hat{\boldsymbol{\theta}}(t|t-1)$ . For any vector  $\mathbf{x}$  and a positive definite covariance matrix  $\boldsymbol{\Sigma}$ , define  $\|\mathbf{x}\|_{\boldsymbol{\Sigma}^{-1}}^2 = \mathbf{x}^T \boldsymbol{\Sigma}^{-1} \mathbf{x}$ . Then by the Gaussian assumption, one has

$$\begin{aligned}& -\log \Pr(\bar{\mathbf{s}}, \bar{\boldsymbol{\theta}} | \bar{\mathbf{y}}(t), \hat{\mathbf{s}}(t|t-1), \hat{\boldsymbol{\theta}}(t|t-1)) \\ &= c + \frac{1}{2} \|\bar{\mathbf{y}}(t) - \bar{\mathbf{A}}(\bar{\boldsymbol{\theta}})\bar{\mathbf{s}}\|_{\boldsymbol{\Sigma}_w^{-1}}^2 \\ &+ \frac{1}{2} \|\bar{\boldsymbol{\theta}} - \bar{\boldsymbol{\theta}}^0\|_{\boldsymbol{\Sigma}_{\theta(t)}^{-1}}^2 + \frac{1}{2} \|\bar{\mathbf{s}} - \bar{\mathbf{s}}^0\|_{\boldsymbol{\Sigma}_{s(t)}^{-1}}^2,\end{aligned}\quad (8)$$

where  $c$  is a constant independent of  $\bar{\boldsymbol{\theta}}$  and  $\bar{\mathbf{s}}$ , and  $\boldsymbol{\Sigma}_{\theta(t)}$  and  $\boldsymbol{\Sigma}_{s(t)}$  are the covariance matrices for  $\boldsymbol{\theta}(t)$  and  $\mathbf{s}(t)$  respectively. The details on updating  $\boldsymbol{\Sigma}_{\theta(t)}$  and  $\boldsymbol{\Sigma}_{s(t)}$  will be discussed in Section V-C. Maximizing the posteriori probability is equivalent to minimizing (8).

This is a joint optimization problem with respect to  $\bar{\mathbf{s}}$  and  $\bar{\boldsymbol{\theta}}$ . The key of SASU is to treat the source signals  $\bar{\mathbf{s}}$  as a function of the arrival angles  $\bar{\boldsymbol{\theta}}$ . This is based on the observation that for a given  $\bar{\boldsymbol{\theta}}$ , the optimal  $\bar{\mathbf{s}}$  to minimize (8) is unique: take the partial derivative of (8) with respect to  $\bar{\mathbf{s}}$  and set it to zero; it can be verified that the optimal  $\bar{\mathbf{s}}$  is given by

$$\bar{\mathbf{s}}(\bar{\boldsymbol{\theta}}) = \bar{\mathbf{s}}^0 + (\bar{\mathbf{A}}^T \boldsymbol{\Sigma}_w^{-1} \bar{\mathbf{A}} + \boldsymbol{\Sigma}_s^{-1})^{-1} \bar{\mathbf{A}}^T \boldsymbol{\Sigma}_w^{-1} (\bar{\mathbf{y}}(t) - \bar{\mathbf{A}}\bar{\mathbf{s}}^0), \quad (9)$$

where we use the notation  $\bar{\mathbf{s}}(\bar{\boldsymbol{\theta}})$  to emphasize that the optimal  $\bar{\mathbf{s}}$  is determined by  $\bar{\boldsymbol{\theta}}$ . As a result, the cost function (8) can be written as

$$\begin{aligned}f(\bar{\boldsymbol{\theta}}) &= \|\bar{\mathbf{y}}(t) - \bar{\mathbf{A}}(\bar{\boldsymbol{\theta}})\bar{\mathbf{s}}(\bar{\boldsymbol{\theta}})\|_{\boldsymbol{\Sigma}_w^{-1}}^2 \\ &+ \|\bar{\boldsymbol{\theta}} - \bar{\boldsymbol{\theta}}^0\|_{\boldsymbol{\Sigma}_{\theta(t)}^{-1}}^2 + \|\bar{\mathbf{s}}(\bar{\boldsymbol{\theta}}) - \bar{\mathbf{s}}^0\|_{\boldsymbol{\Sigma}_{s(t)}^{-1}}^2.\end{aligned}\quad (10)$$

It is worth to note that the optimization is with respect to the arrival angles  $\bar{\boldsymbol{\theta}}$  only and that the source signals  $\bar{\mathbf{s}}$  will be automatically updated when  $\bar{\boldsymbol{\theta}}$  is updated.

## B. A Newton Method for State Update

The focus of this subsection is on how to efficiently minimize the cost function (10). From standard optimization theory, first order methods, such as the gradient descent method, have a slow convergence rate. A second order method, i.e., a Newton method, is preferred when applicable. Note that in order to apply a Newton method, exact or approximated second order derivatives are often required.

We design a Newton method in which the second order derivative of  $f$  is not exactly computed. The basic idea is sketched as follows. At a given point, say  $\bar{\boldsymbol{\theta}}^k$ , we approximate the cost function  $f$  by a quadratic function  $q$ . The optimal  $\bar{\boldsymbol{\theta}}$  that minimizes  $q$  admits a closed form, denoted by  $\bar{\boldsymbol{\theta}}^*$ . It is clear that  $\bar{\boldsymbol{\theta}}^*$  may not be the minimizer of  $f$ . However,  $q$  is designed so that  $\bar{\boldsymbol{\theta}}^* - \bar{\boldsymbol{\theta}}^k$  always gives a descent direction of  $f$ . Hence, one can search along the line  $\bar{\boldsymbol{\theta}}(\alpha) = \bar{\boldsymbol{\theta}}^k + \alpha(\bar{\boldsymbol{\theta}}^* - \bar{\boldsymbol{\theta}}^k)$  to find an  $\alpha \in [0, 1]$  such that  $f(\bar{\boldsymbol{\theta}}(\alpha)) < f(\bar{\boldsymbol{\theta}}^k)$ . Set  $\bar{\boldsymbol{\theta}}^{k+1} = \bar{\boldsymbol{\theta}}(\alpha)$ . Now one has an iterative procedure to minimize the cost function  $f$ .

The quadratic approximation  $q$  is defined as follows. For a given point  $\bar{\boldsymbol{\theta}}^k$ , define

$$\mathbf{G}^k = \nabla(\bar{\mathbf{A}}(\bar{\boldsymbol{\theta}})\bar{\mathbf{s}}(\bar{\boldsymbol{\theta}}^k)), \quad (11)$$

where the gradient is with respect to  $\bar{\boldsymbol{\theta}}$ . Note that in the approximation  $q$ ,  $\bar{\mathbf{s}}$  is fixed to  $\bar{\mathbf{s}}(\bar{\boldsymbol{\theta}}^k)$ . Define  $\boldsymbol{\delta}^k = \bar{\boldsymbol{\theta}} - \bar{\boldsymbol{\theta}}^k$  and  $\bar{\mathbf{y}}_r(t) = \bar{\mathbf{y}}(t) - \bar{\mathbf{A}}(\bar{\boldsymbol{\theta}}^k)\bar{\mathbf{s}}(\bar{\boldsymbol{\theta}}^k)$ . Then at a neighborhood of  $\bar{\boldsymbol{\theta}}^k$ ,

$$\begin{aligned}\bar{\mathbf{y}}(t) - \bar{\mathbf{A}}(\bar{\boldsymbol{\theta}})\bar{\mathbf{s}}(\bar{\boldsymbol{\theta}}) &\approx \bar{\mathbf{y}}(t) - \bar{\mathbf{A}}(\bar{\boldsymbol{\theta}})\bar{\mathbf{s}}(\bar{\boldsymbol{\theta}}^k) \\ &\approx \bar{\mathbf{y}}(t) - \bar{\mathbf{A}}(\bar{\boldsymbol{\theta}}^k)\bar{\mathbf{s}}(\bar{\boldsymbol{\theta}}^k) - \mathbf{G}^k(\bar{\boldsymbol{\theta}} - \bar{\boldsymbol{\theta}}^k) \\ &= \bar{\mathbf{y}}_r(t) - \mathbf{G}^k \boldsymbol{\delta}^k,\end{aligned}$$

and the cost function (10) can be approximated by

$$\begin{aligned}& \|\bar{\mathbf{y}}_r(t) - \mathbf{G}^k \boldsymbol{\delta}^k\|_{\boldsymbol{\Sigma}_w^{-1}}^2 + \|\bar{\boldsymbol{\theta}} - \bar{\boldsymbol{\theta}}^0\|_{\boldsymbol{\Sigma}_{\theta(t)}^{-1}}^2 + \|\bar{\mathbf{s}}(\bar{\boldsymbol{\theta}}^k) - \bar{\mathbf{s}}^0\|_{\boldsymbol{\Sigma}_{s(t)}^{-1}}^2 \\ &\approx \underbrace{\|\bar{\mathbf{y}}_r(t) - \mathbf{G}^k \boldsymbol{\delta}^k\|_{\boldsymbol{\Sigma}_w^{-1}}^2 + \|\boldsymbol{\delta}^k + \bar{\boldsymbol{\theta}}^k - \bar{\boldsymbol{\theta}}^0\|_{\boldsymbol{\Sigma}_{\theta(t)}^{-1}}^2}_{q(\boldsymbol{\delta}^k) \text{ or equivalently } q(\bar{\boldsymbol{\theta}})} + c,\end{aligned}$$

where  $c = \|\bar{\mathbf{s}}(\bar{\boldsymbol{\theta}}^k) - \bar{\mathbf{s}}^0\|_{\boldsymbol{\Sigma}_{s(t)}^{-1}}^2$  is a constant independent of  $\bar{\boldsymbol{\theta}}$ . It is clear that  $q$  is a quadratic function of  $\boldsymbol{\delta}^k$ . When  $\boldsymbol{\Sigma}_w$  and  $\boldsymbol{\Sigma}_{\theta(t)}$  are invertible, the approximation  $q$  admits a unique minimizer given by

$$\begin{aligned}\boldsymbol{\delta}^* &= \left(\mathbf{G}^{kT} \boldsymbol{\Sigma}_w^{-1} \mathbf{G}^k + \boldsymbol{\Sigma}_{\theta(t)}^{-1}\right)^{-1} \\ &\cdot \left(\mathbf{G}^{kT} \boldsymbol{\Sigma}_w^{-1} \bar{\mathbf{y}}_r - \boldsymbol{\Sigma}_{\theta(t)}^{-1}(\bar{\boldsymbol{\theta}}^k - \bar{\boldsymbol{\theta}}^0)\right).\end{aligned}\quad (12)$$

Besides the closed form for the minimizer, another important property of  $q$  is the following proposition.

**Proposition 2.** *Suppose that the vector  $\boldsymbol{\delta}^*$  in (12) is nonzero. Then there exists a strictly positive constant  $\alpha > 0$  such that  $f(\bar{\boldsymbol{\theta}}^k + \alpha \boldsymbol{\delta}^*) < f(\bar{\boldsymbol{\theta}}^k)$ .*

Due to space constraints, the proof is postponed to the journal version of this paper [14]. The significance of this property is that  $\boldsymbol{\delta}_k^*$  gives a descent direction of  $f$ . Hence,

an iterative procedure to monotonically decrease  $f$  can be designed. The algorithm we use is detailed in Algorithm 1.

---

**Algorithm 1** An Iterative Procedure to Minimize  $f$

---

**Input:**  $\bar{\mathbf{y}}(t)$ ,  $\boldsymbol{\theta}^0 = \hat{\boldsymbol{\theta}}(t|t-1)$ ,  $\bar{\mathbf{s}}^0 = \hat{\mathbf{s}}(t|t-1)$ ,  $\boldsymbol{\Sigma}_w$ ,  $\boldsymbol{\Sigma}_{\bar{\boldsymbol{\theta}}(t)}$ ,  $\boldsymbol{\Sigma}_{\bar{\mathbf{s}}(t)}$ , and error tolerance constants  $\epsilon_1, \epsilon_2 \in \mathbb{R}^+$ .

**Output:**  $\hat{\boldsymbol{\theta}}(t|t)$  and  $\hat{\mathbf{s}}(t|t)$ .

Iteratively perform the following:  $k = 0, 1, \dots$

- 1) Compute  $\boldsymbol{\delta}^*$  according to (12).
  - 2) Find  $\bar{\boldsymbol{\theta}}^{k+1} = \bar{\boldsymbol{\theta}}^k + \alpha \boldsymbol{\delta}^*$  by perform the following.
    - a) If  $\|\boldsymbol{\delta}^*\|_2 < \epsilon_1$ , set  $\hat{\boldsymbol{\theta}}(t|t) = \bar{\boldsymbol{\theta}}^k$  and  $\hat{\mathbf{s}}(t|t) = \bar{\mathbf{s}}(\bar{\boldsymbol{\theta}}^k)$ , and quit this algorithm.
    - b) Set  $\alpha = 1$ . If  $f(\bar{\boldsymbol{\theta}}^k + \alpha \boldsymbol{\delta}^*) \geq f(\bar{\boldsymbol{\theta}}^k)$ , then decrease  $\alpha$  via  $\alpha = \alpha/2$  until  $f(\bar{\boldsymbol{\theta}}^k + \alpha \boldsymbol{\delta}^*) < f(\bar{\boldsymbol{\theta}}^k)$ .
    - c) Set  $\bar{\boldsymbol{\theta}}^{k+1} = \bar{\boldsymbol{\theta}}^k + \alpha \boldsymbol{\delta}^*$ .
    - d) If  $|f(\bar{\boldsymbol{\theta}}^{k+1}) - f(\bar{\boldsymbol{\theta}}^k)| < \epsilon_2 f(\bar{\boldsymbol{\theta}}^k)$ , then set  $\hat{\boldsymbol{\theta}}(t|t) = \bar{\boldsymbol{\theta}}^{k+1}$  and  $\hat{\mathbf{s}}(t|t) = \bar{\mathbf{s}}(\bar{\boldsymbol{\theta}}^{k+1})$ , and quit this algorithm. Otherwise, start a new iteration.
- 

Finally, we explicitly compute the gradient  $\mathbf{G}^k$  defined in (11). Since  $\bar{\mathbf{A}}(\bar{\boldsymbol{\theta}}) \bar{\mathbf{s}}(\bar{\boldsymbol{\theta}}^k) \in \mathbb{R}^{2m}$  and  $\bar{\boldsymbol{\theta}} \in \mathbb{R}^{2n}$ , one has  $\mathbf{G}^k \in \mathbb{R}^{2m \times 2n}$ . From the definition of  $\bar{\mathbf{A}}(\bar{\boldsymbol{\theta}})$  in (3), it can be verified that the  $(i, \ell)^{th}$  entry of  $\mathbf{G}^k$ , denoted by  $\mathbf{G}_{i,\ell}^k$ ,  $1 \leq i \leq 2m$ ,  $1 \leq \ell \leq 2n$ , is given by

$$\mathbf{G}_{i,\ell}^k = \frac{\partial (\bar{\mathbf{A}}(\bar{\boldsymbol{\theta}}) \bar{\mathbf{s}}(\bar{\boldsymbol{\theta}}^k))_i}{\partial \bar{\theta}_\ell} = \begin{cases} -2\pi \frac{d}{\lambda} (i-1) \cos \theta_\ell \sin(2\pi \frac{d}{\lambda} (i-1) \sin \theta_\ell) \bar{s}_\ell(\bar{\boldsymbol{\theta}}) & \text{if } i \leq m, \text{ and } \ell \leq n, \\ -2\pi \frac{d}{\lambda} (i-1) \cos \theta_\ell \cos(2\pi \frac{d}{\lambda} (i-1) \sin \theta_\ell) \bar{s}_\ell(\bar{\boldsymbol{\theta}}) & \text{if } i > m, \text{ and } \ell \leq n, \\ 0 & \text{if } \ell > n. \end{cases}$$

### C. Covariance Update

In Kalman filtering, one needs to update both the state and the covariance matrix of the state. That is, estimate  $\boldsymbol{\Sigma}_{\bar{\mathbf{s}}(t)}$  and  $\boldsymbol{\Sigma}_{\bar{\boldsymbol{\theta}}(t)}$  from  $\boldsymbol{\Sigma}_{\bar{\mathbf{s}}(t-1)}$  and  $\boldsymbol{\Sigma}_{\bar{\boldsymbol{\theta}}(t-1)}$ . For the purpose of covariance update, we linearize the system at the point  $\hat{\mathbf{s}}(t|t)$  and  $\hat{\boldsymbol{\theta}}(t|t)$  (to simplify the notations, we use  $\hat{\mathbf{s}}$  and  $\hat{\boldsymbol{\theta}}$  henceforth). With respect to  $\bar{\mathbf{s}}$ , the system can be approximated by  $\bar{\mathbf{y}}(t) = \bar{\mathbf{A}}(\hat{\boldsymbol{\theta}}) \bar{\mathbf{s}} + \bar{\mathbf{w}}$ . Following from the covariance update step in the standard Kalman filter and Woodbury matrix inverse identity, one has

$$\boldsymbol{\Sigma}_{\bar{\mathbf{s}}(t|t-1)}^{-1} = \boldsymbol{\Sigma}_s^{-1} - \boldsymbol{\Sigma}_s^{-1} \left( \boldsymbol{\Sigma}_{\bar{\mathbf{s}}(t-1)}^{-1} + \boldsymbol{\Sigma}_s^{-1} \right) \boldsymbol{\Sigma}_s^{-1},$$

$$\boldsymbol{\Sigma}_{\bar{\mathbf{s}}(t)}^{-1} = \boldsymbol{\Sigma}_{\bar{\mathbf{s}}(t|t-1)}^{-1} + \bar{\mathbf{A}}^T(\hat{\boldsymbol{\theta}}) \boldsymbol{\Sigma}_w^{-1} \bar{\mathbf{A}}(\hat{\boldsymbol{\theta}}),$$

where the matrix  $\boldsymbol{\Sigma}_s$  is defined at the beginning of Section V. With respect to  $\bar{\boldsymbol{\theta}}$ , the system can be approximated by  $\bar{\mathbf{y}}(t) = \bar{\mathbf{A}}(\hat{\boldsymbol{\theta}}) \hat{\mathbf{s}} + \mathbf{G}(\bar{\boldsymbol{\theta}} - \hat{\boldsymbol{\theta}}) + \bar{\mathbf{w}}$ , where  $\mathbf{G} = \nabla(\bar{\mathbf{A}}(\bar{\boldsymbol{\theta}}) \hat{\mathbf{s}})$  and the differentiation is performed at the point  $\bar{\boldsymbol{\theta}} = \hat{\boldsymbol{\theta}}$ . Based

on this linear approximation, one has

$$\boldsymbol{\Sigma}_{\bar{\boldsymbol{\theta}}(t|t-1)}^{-1} = \boldsymbol{\Sigma}_\theta^{-1} - \boldsymbol{\Sigma}_\theta^{-1} \mathbf{F}$$

$$\cdot \left( \boldsymbol{\Sigma}_{\bar{\boldsymbol{\theta}}(t-1)}^{-1} + \mathbf{F}^T \boldsymbol{\Sigma}_\theta^{-1} \mathbf{F} \right)^{-1} \mathbf{F}^T \boldsymbol{\Sigma}_\theta^{-1},$$

$$\boldsymbol{\Sigma}_{\bar{\boldsymbol{\theta}}(t)}^{-1} = \boldsymbol{\Sigma}_{\bar{\boldsymbol{\theta}}(t|t-1)}^{-1} + \mathbf{G}^T \boldsymbol{\Sigma}_w^{-1} \mathbf{G},$$

where the constant matrix  $\boldsymbol{\Sigma}_\theta$  is defined at the beginning of Section V. It is worth to note that instead of updating  $\boldsymbol{\Sigma}_{\bar{\mathbf{s}}(t)}$  and  $\boldsymbol{\Sigma}_{\bar{\boldsymbol{\theta}}(t)}$  directly, we choose to update  $\boldsymbol{\Sigma}_{\bar{\mathbf{s}}(t)}^{-1}$  and  $\boldsymbol{\Sigma}_{\bar{\boldsymbol{\theta}}(t)}^{-1}$ . Note that  $\boldsymbol{\Sigma}_{\bar{\mathbf{s}}(t)}^{-1}$  and  $\boldsymbol{\Sigma}_{\bar{\boldsymbol{\theta}}(t)}^{-1}$  are used in the objective function  $f$ . This update not only reduces computational complexity but also improves numerical stability according to our experience in simulations.

### D. Relation to Other Works

We briefly comment on the relation between SASU and two popular methods in the literature. A detailed justification of the comments is postponed to the journal version of this paper [14].

The PAST algorithm was proposed by Yang in [3] and has been widely adopted as a benchmark algorithm to track the signal subspace. There, a matrix  $\mathbf{W}(t)$  that spans the signal subspace was used to present the signal subspace. For a given measurement vector  $\mathbf{y}(t)$ , the projection of  $\mathbf{y}(t)$  on the signal subspace can be calculated via  $\mathbf{W}(t) \mathbf{W}^\dagger(t) \mathbf{y}(t)$ . When  $\mathbf{W}(t)$  comprises orthonormal columns, then  $\mathbf{W}(t) \mathbf{W}^\dagger(t) \mathbf{y}(t) = \mathbf{W}(t) \mathbf{W}^H(t) \mathbf{y}(t)$  where  $\mathbf{W}^H(t)$  denotes the Hermitian of  $\mathbf{W}(t)$ . To track the subspace, it was proposed in [3] to minimize the cost function  $\|\mathbf{y}(t) - \mathbf{W}(t) \mathbf{W}^H(t) \mathbf{y}(t)\|_2^2$ .

In the SASU framework, with special choices of parameters, the cost function is equivalent to that used in the PAST algorithm. More specifically, recall that  $\boldsymbol{\Sigma}_s = \sigma_s^2 \mathbf{I}$  and  $\boldsymbol{\Sigma}_\theta = \sigma_\theta^2 \mathbf{I}$ . Let  $\sigma_s^2 \rightarrow +\infty$  and  $\sigma_\theta^2 \rightarrow +\infty$ . It can be proved that  $\bar{\mathbf{s}}(\bar{\boldsymbol{\theta}}) = \bar{\mathbf{A}}^\dagger(\bar{\boldsymbol{\theta}}) \bar{\mathbf{y}}(t)$ , and the last two terms in the cost function (10) approach zero. Assume that the measurement noise is white, i.e.,  $\boldsymbol{\Sigma}_w = \sigma_w^2 \mathbf{I}$ . Then the cost function (10) is equivalent to  $\|\bar{\mathbf{y}}(t) - \bar{\mathbf{A}}(\bar{\boldsymbol{\theta}}) \bar{\mathbf{A}}^\dagger(\bar{\boldsymbol{\theta}}) \bar{\mathbf{y}}(t)\|_2^2$ . Essentially, only the subspace spanned by the array direction matrix matters.

Regarding the standard EKF, the linearization is based on the Taylor expansion at  $\hat{\mathbf{s}}(t|t-1) = \bar{\mathbf{A}}^\dagger(\hat{\boldsymbol{\theta}}(t|t-1)) \bar{\mathbf{y}}(t)$ . This particular choice of  $\hat{\mathbf{s}}(t|t-1)$  is the special case of the SASU framework when  $\sigma_s^2 \rightarrow +\infty$ . The state update rule of EKF is in fact the first iteration in Algorithm 1.

## VI. SIMULATIONS

In this Section, we highlight the smearing phenomenon of MUSIC-based methods, introduced by large angular speeds, by numerical evaluation. This justifies the Kalman approach adopted in this paper. The numerical comparison of several Kalman-based methods is conducted to demonstrate the performance improvement of SASU. Due to the space constraint, the illustration of the smearing phenomenon is postponed to the journal version [14]. To compare Kalman-based methods, two scenarios are considered: a single target and multiple targets. For both scenarios, we assume a ULA with 10 antennas

and that the distance between adjacent antennas is given by  $d = \lambda/2$ . The total tracking period is 10 seconds while the sampling interval is 0.1 second. We choose a low sampling frequency as we are particularly interested in tracking fast moving targets (with respect to the sampling frequency).

In our simulations, we focus on a comparison between SASU, EKF and UKF. To compare these tracking algorithms, 200 random experiments are performed where the target movement is deterministic (and priorly designed) but both the source signals and the measurement noise are random. The value of  $\sigma_\theta^2$  used in (5) is estimated from the deterministic movement of the targets.

We first look at the case of single target tracking. Assume that  $\sigma_s^2 = 1$  and  $\sigma_w^2 = 0.1$ . Further assume that  $E[\bar{s}(t)] = \mathbf{0}$ , which corresponds to the challenging case that the source signal may not always be present. Define the tracking bias and variance by  $\frac{1}{200n} \sum_{i=1}^n \left| \sum_{k=1}^{200} \hat{\theta}_i(t) - \theta_i(t) \right|$ , and  $\frac{1}{200n} \sum_{i=1}^n \sum_{k=1}^{200} \left( \hat{\theta}_i(t) - \theta_i(t) \right)^2$  respectively, where the number 200 indicates the amount of random experiments performed. Figure 1 compares the tracking performance in terms of tracking trajectory, bias and variance. It is clear that EKF and UKF do not accurately track the true trajectory while SASU tracks the DOA almost perfectly.

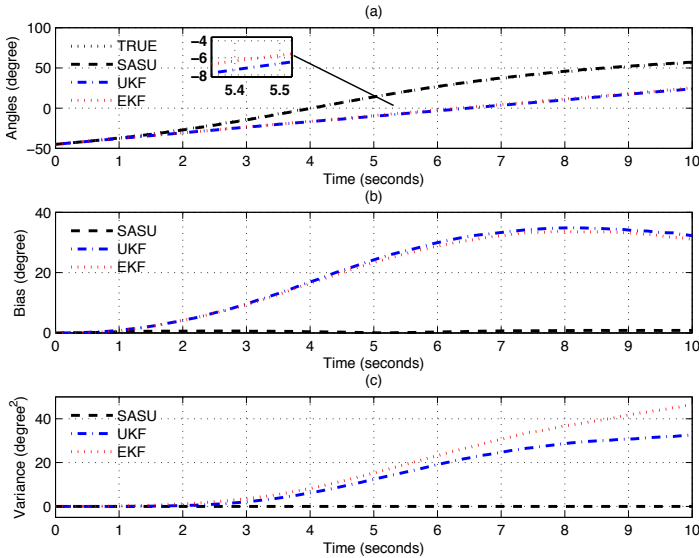


Figure 1. Tracking performance comparison: Single target tracking.

For multiple target tracking, we use the same setup as that for single target tracking except  $E[\bar{s}(t)] = \mathbf{1}$ . We are interested in this case as we would like to see the performance difference when all methods under test can roughly track the DOAs. The results are presented in Figure 2. Again, SASU clearly outperforms the other methods.

To test the robustness of SASU, we also numerically compare the performance when  $\sigma_w^2$  is fixed and  $\sigma_s^2$  varies, and when  $\sigma_s^2$  is fixed and  $\sigma_w^2$  varies. The detailed results are postponed for the journal version of this paper [14] due to space constraints. The results verify that SASU is much more robust than EKF and UKF especially when the uncertainty of

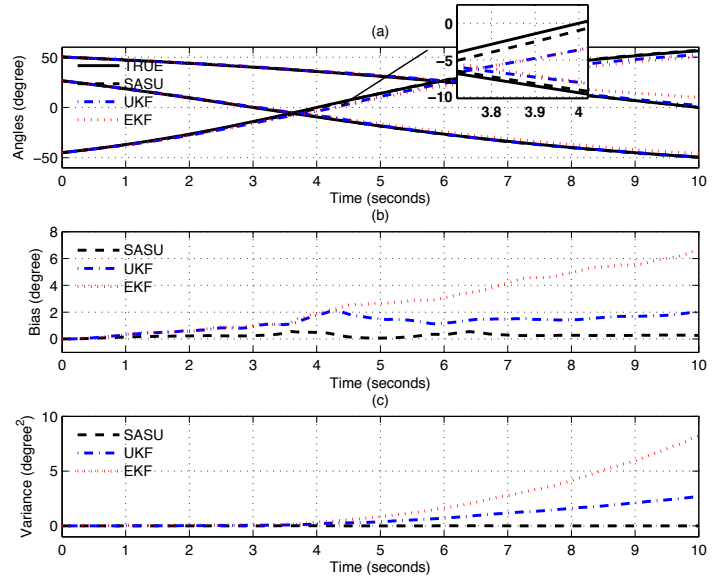


Figure 2. Tracking performance comparison: Three targets tracking.

the source signals is large.

## REFERENCES

- [1] R. Schmidt, "Multiple emitter location and signal parameter estimation," *Antennas and Propagation, IEEE Transactions on*, vol. 34, no. 3, p. 276–280, 1986.
- [2] R. Roy and T. Kailath, "ESPRIT-estimation of signal parameters via rotational invariance techniques," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 37, no. 7, p. 984–995, 1989.
- [3] B. Yang, "Projection approximation subspace tracking," *Signal Processing, IEEE Transactions on*, vol. 43, no. 1, p. 95–107, 1995.
- [4] J. F. Yang and M. Kaveh, "Adaptive eigensubspace algorithms for direction or frequency estimation and tracking," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 36, no. 2, p. 241–251, 1988.
- [5] S. Javier and S. Marcos, "An efficient PASTd-algorithm implementation for multiple direction of arrival tracking," *Signal Processing, IEEE Transactions on*, vol. 47, no. 8, p. 2321–2324, 1999.
- [6] K. W. Lo and C. K. Li, "An improved multiple target angle tracking algorithm," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 28, no. 3, p. 797–805, 1992.
- [7] S. B. Park, C. S. Ryu, and K. K. Lee, "Multiple target angle tracking algorithm using predicted angles," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 30, no. 2, p. 643–648, 1994.
- [8] C. R. Rao, C. R. Sastry, and B. Zhou, "Tracking the direction of arrival of multiple moving targets," *Signal Processing, IEEE Transactions on*, vol. 42, no. 5, p. 1133–1144, 1994.
- [9] K. Reif and R. Unbehauen, "The extended Kalman filter as an exponential observer for nonlinear systems," *Signal Processing, IEEE Transactions on*, vol. 47, no. 8, p. 2324–2328, 1999.
- [10] D. Kong and J. Chun, "A fast DOA tracking algorithm based on the extended Kalman filter," in *National Aerospace and Electronics Conference, 2000. NAECON 2000. Proceedings of the IEEE 2000*, p. 235–238, 2000.
- [11] C. S. Ryu, J. S. Lee, and K. K. Lee, "Multiple target angle-tracking algorithm with efficient equation for angular innovation," *Electronics Letters*, vol. 38, no. 10, p. 483–484, 2002.
- [12] S. Y. Hou, H. S. Hung, S. H. Chang, J. C. Liu, et al., "Novel algorithms for tracking multiple targets," *Journal of Marine Science and Technology*, vol. 18, no. 2, p. 259–267, 2010.
- [13] Y. Bar-Shalom, X. R. Li, T. Kirubarajan, and J. Wiley, *Estimation with applications to tracking and navigation*. Wiley Online Library, 2001.
- [14] X. Gao, J. Filos, and W. Dai, "DOA tracking: Kalman filtering with simultaneous angle-source update (SASU)," *In preparation*.