Particle Filtering With Soft State Constraints for Target Tracking

CUNJIA LIU, Member, IEEE
BAIBING LI, Senior Member, IEEE
WEN-HUA CHEN, Fellow, IEEE
Loughborough University, Loughborough, U.K.

In practice, additional knowledge about the target to be tracked, other than its fundamental dynamics, can often be modeled as a set of soft constraints and utilized in a filtering process to improve the tracking performance. This paper develops a general approach to the modeling of soft inequality constraints, and investigates particle filtering (PF) with soft state constraints for target tracking. We develop two PF algorithms with soft inequality constraints, i.e., a sequential-importance-resampling particle filter and an auxiliary sampling mechanism. The latter probabilistically selects the candidate particles from the soft inequality constraints of the state variables so that they are more likely to comply with the soft constraints. The performances of the proposed algorithms are evaluated using Monte Carlo simulations in a target tracking scenario.

Manuscript received June 22, 2017; revised December 22, 2017 and December 13, 2018; released for publication March 17, 2019. Date of publication March 29, 2019; date of current version December 5, 2019.

DOI. No. 10.1109/TAES.2019.2908292

Refereeing of this contribution was handled by S. Maskell.

This work was supported in part by the U.K. Engineering and Physical Sciences Research Council Autonomous and Intelligent Systems programme under the Grant EP/J011525/1 with BAE Systems as the leading industrial partner and in part by the EPSRC Grant EP/K014307/2 and by the U.K. MOD University Defence Research Collaboration in Signal Processing. The data used in this research are available from the Loughborough University data repository at: https://doi.org/10.17028/rd.lboro.7987403.

Authors' addresses: C. Liu and W.-H. Chen are with the Department of Aeronautical and Automotive Engineering, Loughborough University, Loughborough LE11 3TU, U.K., E-mail: (c.liu5@lboro.ac.uk; w.chen@lboro.ac.uk); B. Li is with the School of Business and Economics, Loughborough University, Loughborough LE11 3TU, U.K., E-mail: (b.li2@lboro.ac.uk). *(Corresponding author: Cunjia Liu.)*

I. INTRODUCTION

State estimation plays the central role in target tracking. The objective of state estimation is to draw statistical inference about the target status (e.g., position and velocity) by incorporating the sensor data, the dynamic model of the target, and possibly some additional external knowledge about the problem [1]. In the Bayesian framework, state estimation can be achieved by calculating or approximating the posterior probability density function (pdf) of the system's state vector after each new observation is acquired. Most of the classical approaches to Bayesian state estimation are based on the Kalman filter (KF) that is known to be optimal for linear systems with Gaussian noise [2]. To deal with nonlinear problems, the extended KF and some more sophisticated algorithms, such as unscented KF and Gaussian mixture filter, have been developed in the literature [2], [3]. These algorithms are usually based on various linearization and/or Gaussian approximation techniques. The particle filtering (PF) approach, on the other hand, uses a set of samples (particles) with associated weights to directly estimate the posterior distribution of the state vector. Hence, it is capable of dealing with highly nonlinear/non-Gaussian problems. General discussions about particle filters can be found in [4]–[6].

In many applications, external knowledge other than the state-space model can be utilized to provide additional information about the system of interest. Such context-based information, once being exploited appropriately through information fusion, may substantially improve statistical inference for the status of the system [7]. Take the road-constrained ground target tracking as an example: the target vehicle's position is constrained by the physical road network and its speed may also be restricted by the traffic rules [8], [9]. Hence, any relevant target vehicle information will reduce the degree of uncertainty about the vehicle being tracked. Recent development in maritime navigation [10], [11] has also demonstrated the benefits of using constraint-based filtering in terms of improving estimation accuracy, where the coastal map was converted into position constraints of the target ship. Another interesting work, by Sviestins and Pirard [12], tried to modify the target dynamics by embedding a simple guidance law that respects spatial constraints. In addition, state estimation using state constraints can be found in process control applications to large chemical plants as there are usually physical restrictions on certain quantities in chemical processes [13], [14]. An overview on state estimation with equality and/or inequality constraints can be found in [15]. Some recent advances for nonlinear/non-Gaussian systems with inequality state constraints can be found in [16]–[18].

In most of the studies in the existing literature (e.g., [15]–[18]), however, the filtering algorithms are designed for dealing with hard and deterministic constraints, i.e., the conditions that the state variables are required to satisfy; not much attention has been paid to the uncertainty of external knowledge when they are transformed into state constraints for estimation purposes. In practice, external knowledge in

some applications can naturally be subject to a great degree of uncertainty. Returning to the example of road-network-based tracking, a vehicle may be found being out of the road boundary due to inaccuracies of the roadmap or being temporarily beyond the permitted speed limit. Another example is aircraft approaching and landing. The rules of air near a terminal area of an airport provide general guidelines for aircraft flight patterns, but it cannot physically restrict aircraft within certain spatial volumes [19]–[21]. In these cases, simply using hard constraints may result in biased tracking. As Simon pointed out in [15]: "it can be argued that estimators for most practical engineering systems should be implemented with soft constraints rather than hard constraints." Clearly, there is a real need to develop new methods for state estimation with soft constraints to deal with this kind of problem.

In the literature, a soft constraint means that it is likely to be true; there however exists possibility that the system may violate such a state constraint sometimes. The pioneering work in [22] adopted a soft-constraint setting to enforce a slowly varying feature of a state when performing the health estimation of a turbofan engine. In a static parameter estimation problem [23], soft-constraints were used to impose a prior distribution of the parameters to be estimated. In addition, Papi *et al.* [10] considered occasional constraint violation in target tracking problems, but the soft-constraint was simply defined by a prefixed probability regardless of the actual state. Formal formulation of soft inequality constraints of states, however, has rarely been seen in the literature, other than a few early cases on equality constraints (e.g., [2], [24]) and a recent study by Palmer *et al.* [25]. In [25], Kalman filtering with soft inequality constraints for linear systems with Gaussian noise was investigated, where the uncertainty of knowledge was modeled as a number of soft linear constraints characterized with Gaussian noise.

This paper investigates Bayesian filtering with soft state constraints, aiming to extend the filtering method for linear systems with Gaussian noise and soft linear constraints in [25] to general nonlinear/non-Gaussian systems. Our contribution to the literature is twofold. First, rather than to use *ad hoc* methods to deal with soft inequality constraints, we propose a general formulation for soft inequality constraints of states that boils down soft inequality constraints to an additional pseudo measurement. Second, for general nonlinear/non-Gaussian systems with soft inequality constraints, we develop two PF algorithms in which the additional pseudo measurement characterizing the soft inequality constraints of states can nicely be incorporated into the filtering process via a modified likelihood function. More specifically, following the development of the soft-constrained PF (scPF), which is a natural extension of PF with soft state constraints incorporated into the modified likelihood function, we propose a novel auxiliary PF (termed soft-constrained APF (scAPF)) to improve the effectiveness of filtering. In the scAPF, we follow the original auxiliary particle filter (APF) [26] and the APF algorithm with hard inequality constraints [18], and investigate how to select particles such that those with higher likelihood of complying with the state constraints are more likely to propagate to the next time step.

The rest of this paper is organized as follows. In Section II, the problem of soft-constrained Bayesian state estimation is considered, with a proposed general formulation of soft inequality constraints. The importance-sampling-based particle filter with soft constraints, scPF algorithm, is developed in Section III. Section IV further improves state estimation by developing the scAPF algorithm. Numerical simulation study is carried out in Section V, which is followed by the final conclusions and discussion section.

II. PROBLEM FORMULATION

In this section, the problem of state estimation with soft inequality constraints is formulated within the Bayesian framework.

A. Statement of the Problem

Consider a dynamic system that is described by the following discrete-time state-space model:

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

$$\mathbf{z}_k = \mathbf{h}_k(\mathbf{x}_k) + \mathbf{v}_k \tag{2}$$

where $\mathbf{x}_k \in \mathbb{R}^{n_x}$ is the system state vector at time instant k, $\mathbf{z}_k \in \mathbb{R}^{n_z}$ the measurement vector, $\mathbf{w}_k$ the process noise vector, and $\mathbf{v}_k$ the measurement noise vector. $\mathbf{f}_k : \mathbb{R}^{n_x} \to \mathbb{R}^{n_x}$ and $\mathbf{h}_k : \mathbb{R}^{n_x} \to \mathbb{R}^{n_z}$ are possibly nonlinear vector-valued system functions and observation functions, respectively. The process and measurement noise vectors are assumed to be absolutely continuous random vectors with zero mean. They are mutually independent and follow pdfs $\mathbf{w}_k \sim p_w(\mathbf{w}_k)$ and $\mathbf{v}_k \sim p_v(\mathbf{v}_k)$, respectively. Let $\mathbf{x}_{0:k} = \{\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_k\}$ and $\mathbf{z}_{1:k} = \{\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_k\}$ denote the sets of all states and all measurements up to time instant k, respectively.

Clearly, the distribution of the state vector $\mathbf{x}_k$ conditioned on $\mathbf{x}_{k-1}$ can be derived from (1) and it is denoted by $p(\mathbf{x}_k|\mathbf{x}_{k-1})$. Likewise, the distribution of measurement $\mathbf{z}_k$ conditioned on $\mathbf{x}_k$, $p(\mathbf{z}_k|\mathbf{x}_k)$, can be obtained from (2); it is usually termed a *likelihood* function. The objective of state estimation is to recursively infer for the state vector $\mathbf{x}_k$, given the available information. In Bayesian sate estimation, this amounts to construct the posterior pdf $p(\mathbf{x}_{0:k}|\mathbf{z}_{1:k})$ given the observation sequence $\mathbf{z}_{1:k}$ (see [4] and [5]).

B. Modeling of Soft Inequality Constraints

The states of a system usually represent some physical quantities of interest. In practice, the state vector may be restricted to a certain feasible region. Mathematically, such constraints can usually be represented by a set of nonlinear inequality equations, $\mathbf{g}_k(\mathbf{x}_k) \leq 0$, where $\mathbf{g}_k = [g_{1,k}, \ldots, g_{n_c,k}]^T : \mathbb{R}^{n_x} \to \mathbb{R}^{n_c}$ is a vector of possibly nonlinear functions of $\mathbf{x}_k$. Although there are many studies

investigating filtering with hard constraints (e.g., [13], [15], [16], and [18]), the hard constraints provide no mechanism to tolerate any uncertainty in the external knowledge on the feasible region within which the state vector is supposed to evolve.

Inspired by [25], in this paper, we formulate soft inequality constraints by introducing a random vector for uncertainty, $\Gamma_k \in \mathbb{R}^{n_c}$, such that

$$\mathbf{g}_k(\mathbf{x}_k) - \Gamma_k \leq 0 \tag{3}$$

where $\Gamma_k = [\gamma_{1,k}, \ldots, \gamma_{n_c,k}]^T$ is an unknown vector of nonnegative random variables that follows a pdf $p_\gamma(\Gamma_k)$, where $p_\gamma(.)$ can be dependent on time but we suppress the time index to keep the notation simple. We assume that the time-varying random vector Γ_k is independent of state vector $\mathbf{x}_k$, as well as Γ_j for $j \neq k$.

Clearly the random vector Γ_k for uncertainty relaxes the requirement of the state vector fulfilling the constraints. In this way, the state constraints (3) become nondeterministic because the random vector Γ_k follows some certain probabilistic law. We also point out that the state constraints (3) include the soft constraints in [25] as a special case; the latter considers a set of linear inequality constraints.

Let Ω denote the space for the unknown random vector Γ_k. At time instant k, we define

$$\mathcal{C}_k := \{\Gamma_k \in \Omega \mid \mathbf{g}_k(\mathbf{x}_k) - \Gamma_k \leq 0\}. \tag{4}$$

The conditional probability of such an event can be written explicitly as

$$\begin{aligned} \Pr\{\mathcal{C}_k|\mathbf{x}_k\} &= \Pr\{\mathbf{g}_k(\mathbf{x}_k) - \Gamma_k \leq 0\} \\ &= \int_{\mathbf{g}_k(\mathbf{x}_k)-\Gamma_k \leq 0} p_\gamma(\Gamma_k)\, \mathrm{d}\Gamma_k. \end{aligned} \tag{5}$$

To keep the notation simple, we suppress the subscript k from Γ_k in the rest of this paper.

We note that for hard constraints without involving uncertainty, i.e., $\Gamma \equiv 0$, the above equation can be simplified as

$$\Pr\{\mathcal{C}_k|\mathbf{x}_k\} = \begin{cases} 1, & \text{if} \quad \mathbf{g}(\mathbf{x}_k) \leq 0; \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

which was examined in [18].

In the case that the elements γ_j ($j = 1, \ldots, n_c$) are independent of each other and each follows a pdf $\gamma_j \sim p_{\gamma_j}(\gamma_j)$, we have $p_\gamma(\Gamma) = \prod_{j=1}^{n_c} p_{\gamma_j}(\gamma_j)$. In practice, the distributions of the individual random variables for uncertainty, $p_{\gamma_i}(\gamma_i)$, $i = 1, \ldots, n_c$, can be specified from a wide range of distributions, such as the truncated Gaussian, exponential, truncated logistic, and gamma distributions. For example, we can consider the scenario that each γ_i follows a zero-mean Gaussian distribution truncated at zero, denoted as $\mathcal{TN}(\gamma; 0, \sigma^2)$

$$p(\gamma) = \begin{cases} 2(\sqrt{2\pi}\sigma)^{-1} \exp[-\gamma^2/(2\sigma^2)] & \text{if } \gamma \geq 0 \\ 0 & \text{if } \gamma < 0 \end{cases}.$$

Alternatively, the logistic distribution truncated at zero can be used to replace the above truncated Gaussian distribution.

Another simple choice is exponential distributions, i.e., $\gamma \sim \mathcal{E}(\gamma; \mu)$ with mean μ

$$p(\gamma) = \begin{cases} \mu^{-1} \exp(-\gamma/\mu) & \text{if } \gamma \geq 0 \\ 0 & \text{if } \gamma < 0 \end{cases}.$$

Gamma distributions are extensions of the above exponential distributions; they are more flexible to fit more complicated practical problems.

In practice, the parameters of the pdf $p_\gamma(\Gamma)$, e.g., σ and μ in the above equations, need to be specified prior to state estimation. We also point out that from a computational perspective, the exponential and truncated logistic distributions are less computationally expensive. Hence, where appropriate they can be used to fit soft-constraints in applications that need real-time solutions.

Before we conclude this section, we offer some further comments on the modeling of soft state constraints. In the literature, when the soft constraints of states are equality constraints, a commonly used approach is to add small nonzero measurement noise to the perfect constraint equations. This leads to a set of pseudo-measurements that can thus be incorporated into the likelihood function for Kalman filtering (see [15]). When the soft constraints of states are nonlinear, non-Gaussian, and inequality constraints, as specified in (3), the general formulation developed in this section boils down the problem into a pseudo-measurement in (5). As we will discuss in the following sections, this pseudo-measurement that characterizes the soft inequality constraints can be incorporated into the likelihood function for PF. This provides a unified approach to dealing with soft constraints by modifying the corresponding likelihood function, no matter whether it is equality/inequality, linear/nonlinear, Gaussian/non-Gaussian noise-based constraints. Clearly, similar tricks could be used with many other measurements. We also point out that this approach is closely linked to the imprecise likelihood approach in [27], where the observation equations have some parameters that are only partially known.

III. IMPORTANCE SAMPLING WITH SOFT CONSTRAINTS

In this section, we take into account the additional information about the state vector, which is modeled as a set of soft constraints and develop the scPF algorithm to estimate the state vector. More specifically, we will draw Bayesian inference about $\mathbf{x}_{0:k}$, given the information provided by the observations $\mathbf{z}_{1:k}$ and the soft-constraint information $\mathcal{C}_{1:k} \triangleq \{\mathcal{C}_1, \ldots, \mathcal{C}_k\}$ up to the present time k, using sequential importance-resampling (SIR).

For system (1), (2) with soft constraints (3), the joint posterior distribution of the state sequence $\mathbf{x}_{0:k}$ can be derived using the following Bayesian recursion (see,

e.g., [4]):

$$\begin{aligned} p(\mathbf{x}_{0:k}|\mathbf{z}_{1:k},\mathcal{C}_{1:k}) &= \frac{p(\mathbf{x}_{0:k},\mathbf{z}_{1:k},\mathcal{C}_{1:k})}{p(\mathbf{z}_{1:k},\mathcal{C}_{1:k})} \\ &= \frac{1}{c}p(\mathbf{z}_k|\mathbf{x}_k)\Pr\{\mathcal{C}_k|\mathbf{x}_k\}\; p(\mathbf{x}_k|\mathbf{x}_{k-1}) \\ &\times \quad p(\mathbf{x}_{0:k-1}|\mathbf{z}_{1:k-1},\mathcal{C}_{1:k-1}) \end{aligned} \tag{7}$$

where c is a normalization scalar defined as

$$c = p(\mathbf{z}_k,\mathcal{C}_k|\mathbf{z}_{1:k-1},\mathcal{C}_{1:k-1}). \tag{8}$$

We note that the recursion in (7) consists of four elements: the likelihood function $p(\mathbf{z}_k|\mathbf{x}_k)$, state transition distribution $p(\mathbf{x}_k|\mathbf{x}_{k-1})$, the posterior distribution at the previous time step $p(\mathbf{x}_{0:k-1}|\mathbf{z}_{1:k-1},\mathcal{C}_{1:k-1})$, and the soft-constraint-related probability $p(\mathcal{C}_k|\mathbf{x}_k)$. The first three elements are analogous to their counterparts in the standard PF problems [28]. The fourth element $\Pr\{\mathcal{C}_k|\mathbf{x}_k\}$, however, is related to the soft constraints; it will provide additional information for statistical inference [10].

In general, for a nonlinear/non-Gaussian system, there is no analytically tractable solution to the posterior pdf in (7); therefore, a numerical approach such as the Monte Carlo method based on importance sampling is often adopted. Specifically, suppose that a set of N particles $\{\mathbf{x}_{0:k}^i\}_{i=1}^N$ are drawn independently from a proposal distribution $\pi(\mathbf{x}_{0:k}|\mathbf{z}_{1:k},\mathcal{C}_{1:k})$. Then, the posterior distribution $p(\mathbf{x}_{0:k}|\mathbf{z}_{1:k},\mathcal{C}_{1:k})$ can be approximated as

$$p(\mathbf{x}_{0:k}|\mathbf{z}_{1:k},\mathcal{C}_{1:k}) \approx \sum_{i=1}^{N} w_k^i \delta(\mathbf{x}_{0:k}-\mathbf{x}_{0:k}^i) \tag{9}$$

where $\delta(\cdot)$ denotes the Dirac delta function and the importance weights w_k^i, $i=1,\ldots,N$ are calculated as

$$w_k^i \propto \frac{p(\mathbf{x}_{0:k}^i|\mathbf{z}_{1:k},\mathcal{C}_{1:k})}{\pi(\mathbf{x}_{0:k}^i|\mathbf{z}_{1:k},\mathcal{C}_{1:k})}, \quad i=1,\ldots,N \tag{10}$$

followed by the normalization such that $\sum_{i=1}^N w_k^i = 1$.

Usually the proposal density is chosen so that it can be written in a sequential manner as follows:

$$\begin{aligned} &\pi(\mathbf{x}_{0:k}|\mathbf{z}_{1:k},\mathcal{C}_{1:k}) \\ &\quad = \pi(\mathbf{x}_k|\mathbf{x}_{0:k-1},\mathbf{z}_{1:k},\mathcal{C}_{1:k})\pi(\mathbf{x}_{0:k-1}|\mathbf{z}_{1:k-1},\mathcal{C}_{1:k-1}). \end{aligned} \tag{11}$$

Substituting the proposal distribution (11) and the recursion (7) into the importance weights (10) yields the weight updating equation

$$w_k^i \propto w_{k-1}^i \frac{p(\mathbf{z}_k|\mathbf{x}_k^i)p(\mathcal{C}_k|\mathbf{x}_k^i)p(\mathbf{x}_k^i|\mathbf{x}_{k-1}^i)}{\pi(\mathbf{x}_k^i|\mathbf{x}_{0:k-1}^i,\mathbf{z}_{1:k},\mathcal{C}_{1:k})}. \tag{12}$$

In many applications, such as target tracking, it is usually the case that only the marginal posterior distribution $p(\mathbf{x}_k|\mathbf{z}_{1:k},\mathcal{C}_{1:k})$ is of interest, and hence the previous state trajectories $\mathbf{x}_{0:k-1}$ can be discarded. In this case, the proposal distribution can be chosen to have a simplified form of $\pi(\mathbf{x}_k|\mathbf{x}_{k-1},\mathbf{z}_k,\mathcal{C}_k)$. Moreover, using the most common choice for proposal distribution, i.e., the state transition distribution $\pi(\mathbf{x}_k|\mathbf{x}_{k-1},\mathbf{z}_k,\mathcal{C}_k) = p(\mathbf{x}_k|\mathbf{x}_{k-1})$, the weights can be updated as

$$w_k^i \propto w_{k-1}^i p(\mathbf{z}_k|\mathbf{x}_k^i)\Pr\{\mathcal{C}_k|\mathbf{x}_k^i\} \tag{13}$$

where $\Pr\{\mathcal{C}_k|\mathbf{x}_k\}$ comes from the pseudo-measurement that is related to the soft constraints. This piece of information contributes to the likelihood function for statistical inference in a similar way as the current observation $\mathbf{z}_k$. We therefore treat $p(\mathbf{z}_k|\mathbf{x}_k)p(\mathcal{C}_k|\mathbf{x}_k)$ as a generalized likelihood function.

Given $\Pr\{\mathcal{C}_k|\mathbf{x}_k\}$ defined in (5), we have

$$\begin{aligned} \Pr\{\mathcal{C}_k|\mathbf{x}_k^i\} &= \Pr\{\mathbf{g}_k(\mathbf{x}_k^i) - \Gamma \le 0\} \\ &= \int_{\mathbf{g}_k(\mathbf{x}_k^i)-\Gamma\le 0} p_\gamma(\Gamma)\,\mathrm{d}\Gamma. \end{aligned} \tag{14}$$

When the individual random variables γ_j are independent of each other and each follows $p_{\gamma_j}(\gamma_j)$, we obtain

$$\begin{aligned} &\Pr\{\mathcal{C}_k|\mathbf{x}_k^i\} \\ &\quad = \int\cdots\int_{g_{j,k}(\mathbf{x}_k^i)-\gamma_j\le 0} \prod_{j=1}^{n_c} p_{\gamma_j}(\gamma_j)\,\mathrm{d}\gamma_1\cdots\mathrm{d}\gamma_{n_c} \\ &\quad = \prod_{j=1}^{n_c}\int_{g_{j,k}(\mathbf{x}_k^i)-\gamma_j\le 0} p_{\gamma_j}(\gamma_j)\,\mathrm{d}\gamma_j = \prod_{j=1}^{n_c}\phi_{j,k}(\mathbf{x}_k^i) \end{aligned}$$

where

$$\phi_{j,k}(\mathbf{x}_k) = \int_{g_{j,k}(\mathbf{x}_k)-\gamma_j\le 0} p_{\gamma_j}(\gamma_j)\,\mathrm{d}\gamma_j. \tag{15}$$

Note that, in this case, the constraint information about the state vector can be evaluated by calculating the product of a series of cumulative distribution functions (cdfs), depending on the number of soft constraints.

We consider two important special cases for the soft constraint formulation. First, suppose that each of the random variables for uncertainty follows a zero-mean Gaussian distribution truncated at 0, $\mathcal{TN}(\gamma;0,\sigma^2)$. In this case, we have

$$\Pr\{\mathcal{C}_k|\mathbf{x}_k^i\} = 2\prod_{j=1}^{n_c}\left[1-\tilde{\Phi}\left(g_{j,k}(\mathbf{x}_k^i);0,\sigma_j^2\right)\right]$$

where $\tilde{\Phi}(u) = \Phi(u)$ if $u \ge 0$ and 0 otherwise. $\Phi(\cdot)$ is the cdf of the Gaussian distribution defined as

$$\Phi(y;\mu,\Sigma) = \int_{-\infty}^{y}\mathcal{N}(x;\mu,\Sigma)\,\mathrm{d}x.$$

Another important special case is the exponential distribution $\gamma_j \sim \mathcal{E}(\gamma_j;\mu_j)$ $(j=1,\ldots,n_c)$. In this case, we have

$$\begin{aligned} \Pr\{\mathcal{C}_k|\mathbf{x}_k^i\} &= \prod_{j=1}^{n_c}\exp\left(-g_{j,k}(\mathbf{x}_k^i)/\mu_j\right) \\ &= \exp\left(-\sum_{j=1}^{n_c} g_{j,k}(\mathbf{x}_k^i)/\mu_j\right). \end{aligned}$$

Clearly the exponential distribution has a much lower computational cost than the truncated normal distribution.

Algorithm 1: Soft-Constrained PF Algorithm.

Require: weighted samples: $\{\mathbf{x}_{k-1}^i, w_{k-1}^i\}_{i=1}^N$

1: **for** $i = 1 : N$ **do**
2: Draw a new particle $\mathbf{x}_k^i \sim p(\mathbf{x}_k|\mathbf{x}_{k-1}^i)$
3: Update weight w_k^i according to (13)
4: **end for**
5: Weight normalization such that $\sum_i^N w_k^i = 1$
6: Resampling

Ensure: new samples: $\{\mathbf{x}_k^i, w_k^i = \frac{1}{N}\}_{i=1}^N$

From the above discussion, a recursive estimation algorithm can be constructed by: 1) generating new particles $\{\mathbf{x}_k^i\}_{i=1}^N$ at k using the sample set $\{\mathbf{x}_{k-1}^i\}_{i=1}^N$ and the proposal distribution $\pi(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{z}_k, \mathcal{C}_k)$; and 2) updating weights using (13) after a new measurement $\mathbf{z}_k$ is obtained and the soft-constraint information $\mathcal{C}_k$ is exploited.

One important issue in PF is the degeneracy problem, since on average, the variance of the importance weights can only increase over time [28]. A useful measure on the degree of degeneracy is the effective sample size (ESS) [29]

$$N_{\text{ess}} = \frac{1}{\sum_{i=1}^N (w_k^i)^2} \tag{16}$$

which takes a value between 1 and N. To alleviate the degeneracy problem, a resampling procedure can be performed after the importance sampling to remove the particles with low weights and duplicate particles with higher weights [4].

The implementation of the SIR algorithm with soft state constraints, i.e., scPF, is provided in Algorithm 1.

IV. IMPROVED APF

The scPF described in Algorithm 1 provides a tool for drawing inference for nonlinear/non-Gaussian systems with soft constraints. When constructing the proposal distribution in the scPF, however, we incorporate a simple solution, i.e., using the state transition distribution $p(\mathbf{x_k}|\mathbf{x_{k-1}})$ as the proposal distribution, without considering the fact that the generated particles may lie outside of the soft-constrained area. When there are a substantial number of the generated particles lying outside the constraint region, the corresponding weights of these particles, as given by (12), are usually very low. This may in turn lead to a low PF efficiency as reflected by a deteriorated ESS.

Inspired by the APF and the recently developed APF algorithm with hard constraints in [18], we now extend the scPF algorithm and investigate a particle filter with an auxiliary structure to alleviate this problem and improve the efficiency of importance sampling.

A. Review of the APF

The standard APF intends to incorporate the knowledge of the newly obtained observation before the sampling stage so that the generated particles are more likely to be compatible with the latest observation [30], [31]. For this end, we first draw an auxiliary particle index from a designed distribution that weights each particle according to its compatibility to the new observation. For the ith particle, a suitable measure of compatibility is the approximation $\widehat{p}(\mathbf{z}_k|\mathbf{x}_{k-1}^i)$ of the predictive likelihood $p(\mathbf{z}_k|\mathbf{x}_{k-1}^i) = \int p(\mathbf{z}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{x}_{k-1}^i)\,\mathrm{d}\mathbf{x}_k$; the latter is usually difficult to calculate in practice [31]. In many applications, $\widehat{p}(\mathbf{z}_k|\mathbf{x}_{k-1}^i)$ is chosen as $p(\mathbf{z}_k|\lambda_k^i)$, where λ_k^i is the centre of $p(\mathbf{x}_k|\mathbf{x}_{k-1}^i)$ (see, e.g., [4]). More specifically, we have

$$\begin{aligned} p(\mathbf{x}_k|\mathbf{z}_{1:k}) &\propto \sum_{i=1}^N w_{k-1}^i p(\mathbf{z}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{x}_{k-1}^i) \\ &= \sum_{i=1}^N \underbrace{\frac{p(\mathbf{z}_k|\mathbf{x}_k)}{\widehat{p}(\mathbf{z}_k|\mathbf{x}_{k-1}^i)} \frac{p(\mathbf{x}_k|\mathbf{x}_{k-1}^i)}{\pi_k^i(\mathbf{x}_k)}}_{w_k^i(\mathbf{x}_k)} \underbrace{w_{k-1}^i \widehat{p}(\mathbf{z}_k|\mathbf{x}_{k-1}^i)}_{\alpha_k^i} \pi_k^i(\mathbf{x}_k) \end{aligned} \tag{17}$$

where $\pi_k^i(\mathbf{x}_k)$ is a proposal distribution corresponding to index i. The APF is implemented by the following steps.

1) Draw an index $I_k^i = j,\ j \in \{1, \ldots, N\}$, with the probability proportional to α_k^j;
2) Draw a particle $\mathbf{x}_k^i$ from the proposal distribution $\pi_k^{I_k^i}(\mathbf{x}_k)$;
3) Calculate the unnormalized weight as $w_k^i = w_k^{I_k^i}(\mathbf{x}_k^i)$.

B. Soft-Constraint-Based APF

In this section, we develop a soft-constraint-based APF algorithm. Following the discussion in the previous section, to improve the propagation of particles, we construct an approximate predictive likelihood that not only takes into consideration the information contained in the new observation, but also accounts for the compliance with the soft-constraints. Specifically, we choose $\widehat{p}(\mathbf{z}_k|\mathbf{x}_{k-1}^i)$ to be $p(\mathbf{z}_k|\lambda_k^i)p(\lambda_k^i|\mathbf{x}_{k-1}^i),\ i = 1, \ldots, N$, where λ_k^i denotes the mode of the soft-constrained transition distribution $\Pr\{\mathcal{C}_k|\mathbf{x}_k\}p(\mathbf{x}_k|\mathbf{x}_{k-1}^i)$, i.e.,

$$\lambda_k^i = \arg\max_{\mathbf{x}_k} \Pr\{\mathcal{C}_k|\mathbf{x}_k\}\, p(\mathbf{x}_k|\mathbf{x}_{k-1}^i). \tag{18}$$

The mode λ_k^i characterizes the most probable location of the predicted state vector $\mathbf{x}_k$ given the particle $\mathbf{x}_{k-1}^i$, $i = 1, \ldots, N$, in conjunction with the soft-constraint information. Therefore, for each particle i, $p(\mathbf{z}_k|\lambda_k^i)$ reflects the new-observation-based likelihood, whereas $p(\lambda_k^i|\mathbf{x}_{k-1}^i)$ measures the compliance of this particle with the soft constraints. This is illustrated by Fig. 1, where the state transition distribution at time k is assumed to be $p(x_k|x_{k-1}^i) = \mathcal{N}(x_k; 1, 1)$ (represented by the dotted line). The state soft-constraint is chosen as $g(x_k) - \gamma \le 0$, where $g(x_k) = 3 - x_k$ and the uncertainty variable $\gamma \sim \mathcal{TN}(\gamma; 0, 1)$. The soft constraint is therefore characterized by $\Pr(\mathcal{C}_k|x_k) = 2[1 - \tilde{\Phi}(3 - x_k; 0, 1)]$ (the broken line). The state transition distribution multiplied by the constraint-related probability, i.e., $\Pr\{\mathcal{C}_k|\mathbf{x}_k\}p(\mathbf{x}_k|\mathbf{x}_{k-1}^i)$, is depicted by the solid line with its mode indicated by a circle.

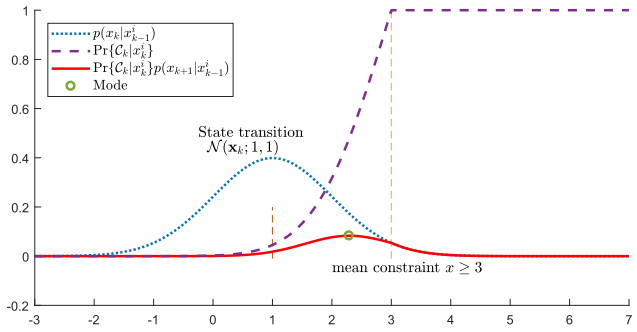

Fig. 1. Illustration of the mode location of the state transition distribution under soft constraints.

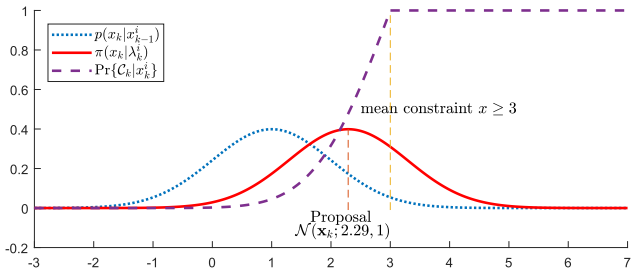

Fig. 2. Illustration of the new proposal distribution for a soft inequality constraint.

To enable the sampling with the proposed predictive likelihood, the mode λ_k^i needs to be calculated efficiently. We note that the mode defined in (18) is equivalent to

$$\lambda_k^i = \arg\min_{\mathbf{x}_k} -\log\left[p(\mathbf{x}_k|\mathbf{x}_{k-1}^i)\right] - \log\left[\Pr\{\mathcal{C}_k|\mathbf{x}_k\}\right] \quad (19)$$

where $\mathcal{C}_k$ defined in (4).

We follow [18] and solve the above optimization problem for each particle i. Specifically, we suppose that $p(\mathbf{x}_k|\mathbf{x}_{k-1}^i)$ is log-concave and we expand it about its mean $\bar{\mathbf{x}}_k^i = \mathbf{f}_{k-1}(\mathbf{x}_{k-1}^i)$ to the second order, i.e., $-\log\left[p(\mathbf{x}_k|\mathbf{x}_{k-1}^i)\right] \approx (\mathbf{x}_k - \bar{\mathbf{x}}_k^i)^T D_{k-1}^i + (\mathbf{x}_k - \bar{\mathbf{x}}_k^i)^T V_{k-1}^i (\mathbf{x}_k - \bar{\mathbf{x}}_k^i)/2 + \text{constant}$, where D_{k-1}^i and V_{k-1}^i denote the first- and second-order derivatives of $-\log[p(x_k|x_{k-1}^i)]$ evaluated at $\bar{\mathbf{x}}_k^i$. For the case where the process noise follows a Gaussian distribution $p_w(\mathbf{w}_k) = \mathcal{N}(\mathbf{w}_k; 0, Q_k)$ with covariance matrix Q_k, $\bar{\mathbf{x}}_k^i$ is the mean as well as the mode. Hence, we have $-\log\left[p(\mathbf{x}_k|\mathbf{x}_{k-1}^i)\right] = (\mathbf{x}_k - \bar{\mathbf{x}}_k^i)^T Q_{k-1}(\mathbf{x}_k - \bar{\mathbf{x}}_k^i)/2 + \text{constant}$.

Consequently, when the uncertainty variables are independent of each other, finding the mode for each particle i in (19) can be reformulated as an unconstrained nonlinear optimization

$$\begin{aligned}\min_{\mathbf{x}_k} J_k^i &= (\mathbf{x}_k - \bar{\mathbf{x}}_k^i)^T V_{k-1}^i (\mathbf{x}_k - \bar{\mathbf{x}}_k^i)/2 \\ &\quad + (\mathbf{x}_k - \bar{\mathbf{x}}_k^i)^T D_{k-1}^i - \sum_{j=1}^{n_c} \log \phi_{j,k}(\mathbf{x}_k)\end{aligned} \quad (20)$$

where $\phi_{j,k}(\mathbf{x}_k)$ is given by (15).

It can be observed that the first two terms on the right-hand side in (20) is a quadratic function that characterizes the deviation of a solution from the unconstrained centre of $p(\mathbf{x}_k|\mathbf{x}_{k-1}^i)$, i.e., $\bar{\mathbf{x}}_k^i$, whereas the second term is designed to drive the solution moving toward the constraint area.

We consider an important scenario where $g_{j,k}(\bar{\mathbf{x}}_k^i) \leq 0$ $(j = 1, \ldots, n_c)$ for a given particle $\mathbf{x}_{k-1}^i$. Since $\bar{\mathbf{x}}_k^i$ lies within the constraint area, we obtain $\log \phi_{j,k}(\mathbf{x}_k) = 0$ at $\mathbf{x}_k = \bar{\mathbf{x}}_k^i$. Consequently, we obtain $\lambda_k^i = \bar{\mathbf{x}}_k^i - [V_{k-1}^i]^{-1} D_{k-1}^i$ from (20). In particular, if the process noise follows the Gaussian distribution $p_w(\mathbf{w}_k) = \mathcal{N}(\mathbf{w}_k; 0, Q_k)$, we have $\lambda_k^i = \bar{\mathbf{x}}_k^i$.

In general, the optimization problem (20) is nonlinear and hence its exact global minimum is difficult to find. However, as argued in [18], the purpose of obtaining the constrained mode λ_k^i through the optimization is to replace the unconstrained center $\bar{\mathbf{x}}_k^i$ with an improved representation of the transition distribution that is subject to the soft constraints. The obtained mode will be used to select particles and to construct a proposal distribution. Consequently, there is no need to find the exact solution for each optimization problem; the approximation errors can statistically be corrected by using the corresponding weights (see (24) later) in the importance sampling stage. In the Appendix, we outline a fast algorithm for solving the optimization problem (20) with a very low computational cost.

In the rest of this section we discuss how to construct the proposal distributions from which particles will be drawn.

First, after an approximate mode λ_k^i is obtained, indexes I_k^i, $i = 1, \ldots, N$, with a probability proportional to $w_{k-1}^i p(\mathbf{z}_k|\lambda_k^i) p(\lambda_k^i|\mathbf{x}_{k-1}^i)$ can be drawn in the resampling stage.

Next, for each given I_k^i, we choose the proposal distribution $\pi(\mathbf{x}_k|\mathbf{z}_{1:k}, \mathcal{C}_{1:k})$ to be the transition distribution $p(\mathbf{x}_k|\mathbf{x}_{k-1}^{I_k^i})$ translated in a way such that it has its mode at $\lambda_k^{I_k^i}$, where the index I^i is obtained from the resampling stage. We denote this proposal distribution by $\pi(\mathbf{x}_k|\lambda_k^{I_k^i})$. Specifically, we choose

$$\pi(\mathbf{x}_k|\lambda_k^{I_k^i}) = p\left(\mathbf{x}_k - \lambda_k^{I_k^i} + \bar{\mathbf{x}}_k^{I_k^i}|\mathbf{x}_{k-1}^{I_k^i}\right). \quad (21)$$

As shown in (21), shifting the probability mass toward the constraint region results in the proposal density (21) closer to the optimal one with a heavier tail of the density, leading to an increase in the probability of drawing particles in the constraint region.

We then draw a particle from the above proposal distribution for each given I_k^i. In doing so, the particle drawn from this proposal distribution has a much high probability to comply with the constraints because the computation of $\lambda_k^{I_k^i}$ has taken into account the soft constraints.

To illustrate this sampling process, we return to the previous example where the state soft-constraint is defined as $g(x_k) - \gamma \leq 0$ with $g(x_k) = 3 - x_k$ and $\gamma \sim \mathcal{TN}(\gamma; 0, 1)$, and the state transition distribution at time k is $p(x_k|x_{k-1}^i) = \mathcal{N}(x_k; 1, 1)$. Fig. 2 displays two distributions, i.e., the transition distribution $p(x_k|x_{k-1}^i)$ and $\pi(x_k|\lambda_k^i)$ in (21). It can be seen from Fig. 2 that, if the transition distribution $p(x_k|x_{k-1}^i)$

(represented by the dotted line with the mode of 1) is used as the proposal distribution, then it has a very low probability of complying with the constraint $g(x_k) = 3 - x_k \leq 0$. The new proposal distribution in (21), $\pi(x_k|\lambda_k^i)$ (the solid line), has the model of around 2.29. As a result, the chance that the particles drawn from this new proposal distribution comply with the soft-constraint is substantially increased, as illustrated in Fig. 2.

Finally, we note that, comparing to the direct sampling from the transition distribution $p(\mathbf{x}_k|\mathbf{x}_{k-1}^{I_k^i})$, the importance weights need to be adjusted by a factor

$$\rho_k^i = \frac{p\left(\mathbf{x}_k|\mathbf{x}_{k-1}^{I_k^i}\right)}{\pi\left(\mathbf{x}_k|\lambda_k^{I_k^i}\right)}. \tag{22}$$

We summarize the overall filtering process as follows. First, with the particles $\{\mathbf{x}_{k-1}^i; w_{k-1}^i\}_{i=1}^N$, we can rewrite the posterior pdf to be

$$\begin{aligned} &p(\mathbf{x}_k|\mathbf{z}_{1:k}, \mathcal{C}_{1:k}) \\ &\quad \propto \sum_{i=1}^N w_{k-1}^i p(\mathbf{z}_k|\mathbf{x}_k) p(\mathcal{C}_k|\mathbf{x}_k) p(\mathbf{x}_k|\mathbf{x}_{k-1}^i) \\ &\quad = \sum_{i=1}^N \underbrace{\frac{p(\mathbf{z}_k|\mathbf{x}_k) p(\mathcal{C}_k|\mathbf{x}_k) p(\mathbf{x}_k|\mathbf{x}_{k-1}^i)}{p(\mathbf{z}_k|\lambda_k^i) p(\lambda_k^i|\mathbf{x}_{k-1}^i) \pi_k^i(\mathbf{x}_k)}}_{\tilde{w}_k^i(\mathbf{x}_k)} \\ &\qquad \times \underbrace{w_{k-1}^i p(\mathbf{z}_k|\lambda_k^i) p(\lambda_k^i|\mathbf{x}_{k-1}^i)}_{\alpha_k^i} \pi_k^i(\mathbf{x}_k). \end{aligned} \tag{23}$$

At the resampling stage, the particles from time $k-1$ are selected with the probabilities proportional to α_k^i for propagation. This results in a set of indexes I_k^i $(i = 1, \ldots, N)$. Next, each new particle $\mathbf{x}_k^i$ at time k is drawn from the proposal distribution $\pi_k^i(\mathbf{x}_k) = \pi(\mathbf{x}_k|\lambda_k^{I_k^i})$. Correspondingly, the weight $\tilde{w}_k^i(\mathbf{x}_k^i)$ is updated as

$$\tilde{w}_k^i(\mathbf{x}_k^i) = \frac{p(\mathbf{z}_k|\mathbf{x}_k^i)\, p(\mathcal{C}_k|\mathbf{x}_k^i)\, p\left(\mathbf{x}_k^i|\mathbf{x}_{k-1}^{I_k^i}\right)}{p\left(\mathbf{z}_k|\lambda_k^{I_k^i}\right)\, p\left(\lambda_k^{I_k^i}|\mathbf{x}_{k-1}^{I_k^i}\right)\, \pi\left(\mathbf{x}_k^i|\lambda_k^{I_k^i}\right)} \tag{24}$$

subject to the normalization. Note that, from (23), the weights, as defined in (24), ensure that the obtained particles constitute a representative sample of the true posterior distribution. The essentials of the scAPF at time k are outlined in Algorithm 2.

V. NUMERICAL SIMULATION

A. Simulation Scenario

In this section, the proposed soft-constrained particle filters are evaluated and compared using an airborne target tracking example. The scenario employs an unmanned aerial vehicle equipped with a gimballed camera to track a ground vehicle maneuvering on a road section (see Fig. 3). The camera can be considered as a bearing-only sensor, which provides the azimuth angle (ζ) and elevation angle

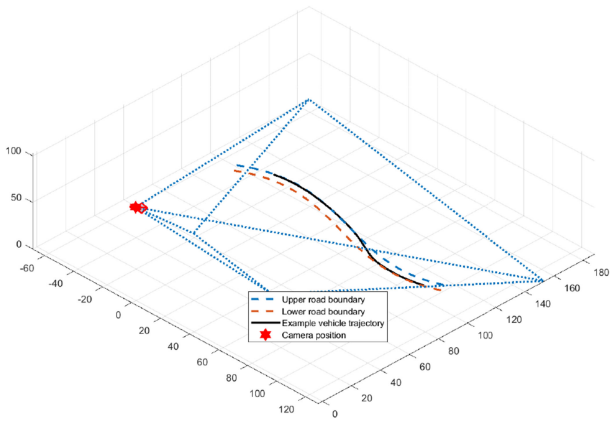

Fig. 3. Airborne target tracking scenario.

Algorithm 2: Soft-Constrained APF Algorithm.

Require: weighted samples: $\{\mathbf{x}_{k-1}^i, w_{k-1}^i\}_{i=1}^N$
for $i = 1 : N$ **do**
  if $g_{j,k}(\bar{\mathbf{x}}_k^i) \leq 0, \forall j \in \{1, \ldots, n_c\}$ **then**
    Set $\lambda_k^i = \bar{\mathbf{x}}_k^i = \mathbf{f}_{k-1}(\mathbf{x}_{k-1}^i)$
  else
    Calculate mode λ_k^i by solving (20)
  end if
  Calculate auxiliary weight α_k^i defined in (23)
end for
Ensure: Samples with auxiliary weights: $\{\mathbf{x}_{k-1}^i, \alpha_k^i\}_{i=1}^N$
for $i = 1 : N$ **do**
  Draw an index $I_k^i = j$, $j \in \{1, \ldots, N\}$, according to probability proportional to α_k^j
  Draw a new particle $\mathbf{x}_k^i$ from $\pi(\mathbf{x}_k|\lambda_k^{I_k^i})$ in (21)
  Update weight $\tilde{w}_k^i$ according to (24)
end for
Weight normalization such that $w_k^i = \frac{\tilde{w}_k^i}{\sum_{i=1}^N \tilde{w}_k^i}$
Ensure: new samples: $\{\mathbf{x}_k^i, w_k^i\}_{i=1}^N$

(η) to the target with respect to the sensor platform [32]. Without loss of generality, this paper assumes a flat ground and a known sensor position at the altitude $z^s = 100\,\text{m}$ above the origin of the local coordinate. Thus, the observation model can be simplified as

$$\mathbf{z}_k = h(\mathbf{x}_k) = \begin{bmatrix} \zeta_k \\ \eta_k \end{bmatrix} = \begin{bmatrix} \arctan_2(y_k, x_k) \\ \arctan_2\left(z^s, \sqrt{x_k^2 + y_k^2}\right) \end{bmatrix} + \mathbf{v}_k \tag{25}$$

where $\mathbf{x}_k = [\, x_k \;\; y_k \;\; \dot{x}_k \;\; \dot{y}_k \,]^T$ is the target's state vector consisting of the vehicle position and velocity components in the x and y directions, and the sensor noise $\mathbf{v}_k$ is modeled as a two-dimensional Gaussian zero-mean vector with covariance $R = \text{diag}\{2 \times 10^{-4}\ \text{rad}^2,\ 2 \times 10^{-4}\ \text{rad}^2\}$.

We assume that the target vehicle travels on the road section as displayed in Fig. 3. The dynamics of the target vehicle is described by a nearly constant-velocity model

that is widely used in the literature [33]

$$\mathbf{x}_{k+1} = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{x}_k + \mathbf{w}_k \tag{26}$$

where $T = 0.2$ s is the sampling interval, and $\mathbf{w}_k$ is a Gaussian process noise with zero mean and covariance matrix Q defined as

$$Q = \begin{bmatrix} \frac{T^3}{3}q_1 & 0 & \frac{T^2}{2}q_1 & 0 \\ 0 & \frac{T^3}{3}q_2 & 0 & \frac{T^2}{2}q_2 \\ \frac{T^2}{2}q_1 & 0 & Tq_1 & 0 \\ 0 & \frac{T^2}{2}q_2 & 0 & Tq_2 \end{bmatrix}. \tag{27}$$

The process noise intensity q_1 and q_2 are chosen as $q_1 = q_2 = 0.8\,\mathrm{m}^2/\mathrm{s}^3$.

In addition to the dynamic model, other information about the vehicle's behavior is also exploited in the tracking process. First, the vehicle position is constrained by the road boundary defined by the following two polynomial curves:

$$g_{1,k} = y_k - (b_3 x_k^3 + b_2 x_k^2 + b_1 x_k + b_0 + b_w) \tag{28}$$

$$g_{2,k} = (b_3 x_k^3 + b_2 x_k^2 + b_1 x_k + b_0 - b_w) - y_k \tag{29}$$

where $b_3 = 5 \times 10^5$, $b_2 = -0.004$, $b_1 = -0.2$, and $b_0 = 125$ are coefficients of the polynomials and $b_w = 2.5$ is the distance from the central line to the road boundary. It also assumes that the speed limit for this road section is $\bar{V} = 12.5$ m/s so that the speed constraint can be defined as $g_{3,k} = (\dot{x}_k^2 + \dot{y}_k^2)^{\frac{1}{2}} - \bar{V}$. The uncertainties about the road boundaries and the possibility of violating the speed limit are characterized by the random variables that follow an exponential distribution $\gamma \sim \mathcal{E}(\gamma; \mu)$, where μ is the mean of the distribution. Consequently, the soft state constraints can be formulated as

$$\begin{bmatrix} g_{1,k}(\mathbf{x}_k) \\ g_{2,k}(\mathbf{x}_k) \\ g_{3,k}(\mathbf{x}_k) \end{bmatrix} - \begin{bmatrix} \gamma_1 \\ \gamma_2 \\ \gamma_3 \end{bmatrix} \leq 0 \tag{30}$$

where we choose $\gamma_1 \sim \mathcal{E}(\gamma_1; 0.25)$, $\gamma_2 \sim \mathcal{E}(\gamma_2; 0.25)$, and $\gamma_3 \sim \mathcal{E}(\gamma_3; 1)$.

The example target trajectory is depicted in Fig. 4, where the vehicle initial position is (90, 109). It can be seen that the vehicle first travels close to the lower boundary and then moves to the other side of the road. During the transition, the vehicle temporarily drives outside of the nominal boundary. The vehicle speed profile against the speed limit is also illustrated in Fig. 5.

For this target tracking scenario, simulation experiments were carried out on a PC with 3.4 GHz CPU. For each simulation experiment, 100 Monte Carlo runs were performed where different realizations of the measurement noises were generated based on the example trajectory. In the state estimation, the initial prior distribution was chosen as a Gaussian distribution with mean $\hat{\mathbf{x}}_0 = [85, 119, -14, -2]^T$ and covariance matrix $P_0 = \mathrm{diag}\{10, 10, 2.5, 2.5\}$. The performances of the proposed filters are evaluated using the following criteria: 1) estimation accuracy as measured by mean square error (MSE) of the position-related states; 2) the ratio of the ESS to the total number of particles N in percentage term (PESS) as a measure of particle quality; and 3) the mean computation time (CT) for one time step in the simulation experiment. The values of MSE, PESS, and CT averaged over the 100 MC runs are reported in this paper, together with the corresponding standard deviation (SD) of the position MSE.

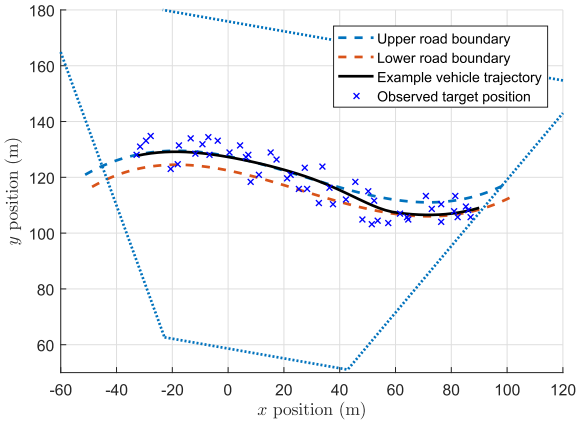

Fig. 4. Example target trajectory.

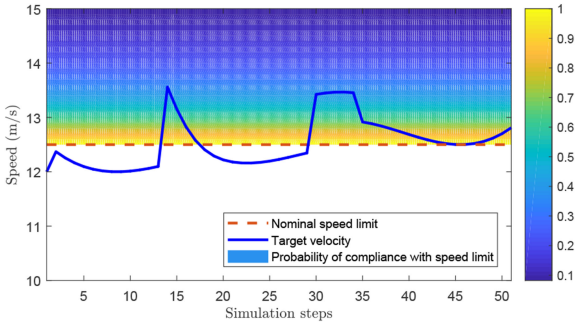

Fig. 5. Target vehicle speed profile.

B. Impact of Different Optimization Strategies

The scAPF developed in this paper involves solving a number of optimization problems. To make this algorithm computationally feasible in practice, a fast algorithm is proposed in the Appendix, which produces suboptimal solutions. This fast algorithm is based on the quasi-Newton method, with 1-iteration and 2-iterations, respectively, to approximate the exact solution; the latter is obtained using the MATLAB *fminunc* function in the following experiments. The approximations will then be rectified statistically by using the weight update function (24) in the scAPF. The first part of our simulation experiments aims to investigate the estimation performances when using the different optimization strategies.

The simulation results of the scAPF algorithm with different optimization methods based on $N = 500$ particles are displayed in Table I.

TABLE I
Performance of scAPF With Different Optimization Strategies Over 100 MC Runs

N	Method	MSE	SD	PESS (%)	CT (s)
500	1-Iter.	3.99	0.75	70.5	0.028
500	2-Iter.	3.90	0.79	70.6	0.041
500	fminunc	3.94	0.77	70.6	1.204

TABLE II
Performance of Different PF Strategies Averaged Over 100 MC Runs

Filter	N	MSE	SD	PESS (%)	CT (s)
SIR	250	22.45	15.22	61.1	0.004
APF	250	16.32	9.99	91.5	0.006
scPF	250	5.15	1.08	54.1	0.007
scAPF	250	4.45	1.13	68.8	0.016
CAPFA	250	7.51	3.60	72.7	0.019
SIR	500	18.25	12.55	61.2	0.007
APF	500	12.50	7.50	92.6	0.011
scPF	500	4.53	0.99	55.5	0.013
scAPF	500	3.99	0.75	70.5	0.028
CAPFA	500	6.92	4.45	73.6	0.035
SIR	1000	13.81	7.72	62.0	0.015
APF	1000	10.29	4.29	93.2	0.022
scPF	1000	4.05	0.88	56.5	0.027
scAPF	1000	3.58	0.80	72.0	0.054
CAPFA	1000	5.72	1.63	74.4	0.065

Table I shows that, overall, the scAPF with the proposed fast algorithm and the scAPF with the standard optimization technique, i.e., MATLAB *fminunc*, have similar performances in terms of average MSE, SD, and PESS. The main drawback of using MATLAB *fminunc* is the high computational load, which may be problematic for some real-time applications. In the following simulation experiments, the scAPF with the quasi-Newton method using only 1-iteration is used for comparison purposes.

C. Performances of Different Particles Filters in the Presence of Soft State Constraints

As shown in the literature (e.g., [9], [11], [34]), appropriately incorporating extra knowledge about the target may significantly improve the tracking performance. To demonstrate the benefits of accounting for soft-constraints in the filtering process, several different particle filters are implemented and compared in this section. First, the SIR algorithm and the standard APF are implemented where only sensor information is used for tracking. Next, the proposed soft-constrained PFs are tested, including both the scPF and scAPF. In addition, a recently proposed hard-constrained PF, i.e. Constrained Auxiliary PF Algorithm (CAPFA) [18], is tested in this tracking scenario. This hard-constrained PF is able to exploit the constraint information but cannot deal with any uncertainty about noncompliance with the constraints.

The simulation experiment results with different particle numbers are summarized in Table II.

It can be seen from Table II that there are several factors influencing the filtering performances. A general trend that can be observed is that a larger number of particles usually results in a better estimation performance at the expense of a higher CT. Other major factors include the exploitation of constraint information and the effectiveness of filtering algorithms.

When the constraint information is not taken into account, the standard SIR delivers much worse estimation accuracy even if a large sample size $N = 1000$ is used, whereas the APF shows some improvements due to the more effective resampling mechanism in this case. In contrast, both soft-constrained filters (scPF and scAPF) demonstrate a significantly improved performance in terms of MSE and the associated SD. If the hard-constraint information is embedded into the filtering process, the resulting filter CAPFA can produce smaller MSEs comparing to their unconstrained counterparts (i.e., SIR and APF). However, because the hard-constrained PF only exploits the particles within the road boundary and within the speed limit, its robustness may be problematic when the sample size N is small and the initial guess is poor. Moreover, if the target exhibits any temporary violation of those constraints, the performance of the hard-constrained PF will be degraded comparing to the results from the soft-constrained PFs.

Next, we focus on the two proposed soft-constrained PFs. The simulation results show that the scAPF algorithm outperforms the scPF algorithm in terms of MSE and PESS. This is not surprising. The particles generated by the proposal distribution $p(\mathbf{x}_k|\mathbf{x}_{k-1})$ in the scPF scheme may stay far away from the constraint area, which in turn may have less weights due to the soft-constraints. The scAPF, on the other hand, is designed to improve this situation by preselecting particles that are more compliant with the constraints. At the expense of a manageable computational load, the resulting PESS is much higher than those for the scPF, suggesting that the particles can better represent the posterior distribution [35]. This is not only important for a point estimate represented by the mean or mode of the posterior distribution, but also provides a better foundation for calculating a region in which the target likely lies. Therefore, depending on application scenarios, one may choose to use the scPF with a less computational load or the scAPF with a better particle quality and more robust performance.

Finally, we closely look into the proposed soft-constrained PF algorithms in comparison with the soft pseudo-measurement (SPM) PF in [10]. The pioneering work in [10] developed a particle filter following an SIR structure but also adopting a modified likelihood function to incorporate soft-constraints, such that

$$\Pr\{\mathcal{C}_k|\mathbf{x}_k\} = \begin{cases} 1, & \text{if} \quad \mathbf{g}(\mathbf{x}_k) \le 0; \\ \alpha, & \text{otherwise} \end{cases} \tag{31}$$

where $0 \le \alpha \le 1$ is a constant to be chosen based on the frequency at which the constraint has been violated. If α is set to 0, the above constraints become hard constraints, whereas if $\alpha = 1$ it reduces to the ordinary SIR PF without constraints. The MC simulation results with different α settings are given in Table III. It can be seen that the best

TABLE III
Comparison With SPM Particle Filters Over 100 MC Runs

Particle Filter	N	MSE	SD	PESS (%)	CT (s)
SPM ($\alpha = 0.1$)	500	7.60	7.22	52.4	0.008
SPM ($\alpha = 0.2$)	500	9.18	8.44	55.3	0.008
SPM ($\alpha = 0.4$)	500	11.11	9.57	59.4	0.008
SPM ($\alpha = 0.6$)	500	13.18	10.64	61.4	0.008
SPM ($\alpha = 0.8$)	500	16.30	13.18	61.5	0.008
SIR	500	18.25	12.55	61.2	0.007
scPF	500	4.53	0.99	55.5	0.013
SPM ($\alpha = 0.1$)	1000	6.13	4.65	53.3	0.016
SPM ($\alpha = 0.2$)	1000	6.47	4.34	55.8	0.016
SPM ($\alpha = 0.4$)	1000	7.96	6.05	60.0	0.016
SPM ($\alpha = 0.6$)	1000	9.92	6.31	62.0	0.016
SPM ($\alpha = 0.8$)	1000	12.50	6.83	62.2	0.016
SIR	1000	13.81	7.72	62.0	0.015
scPF	1000	4.05	0.88	56.5	0.027

MSE performance of the SPM PF is greatly dependent on the appropriate value of α in the MC simulation runs. Overall, however, because α is set to be a constant value rather than a state-dependent likelihood function as defined in (5), usually the soft constraints (31) cannot accurately reflect the nature of the constraints, and consequently its numerical performance is not ideal.

VI. CONCLUSION

This paper investigates PF for nonlinear/non-Gaussian systems with soft inequality constraints. We have extended the existing formulation in the literature in twofold: 1) the constraint formulation $\mathbf{g}_k(\mathbf{x}_k)$ is generalized from linear to nonlinear, and 2) the probabilistic measure for uncertainties is extended from Gaussian to non-Gaussian.

With the proposal distribution chosen as the state transition distribution, we have proposed the scPF algorithm to deal with soft constraints. In addition, to further improve the performance and circumvent the difficulty of a relatively low ESS, we have developed a novel scAPF algorithm. The proposed scAPF fully exploits the advantages of the traditional APF that probabilistically selects particles for propagation by using the likelihood information. Specifically, the scAPF utilizes the soft-constraint information to define the auxiliary variable for particle propagation in the resampling stage and develops a new proposal distribution in the importance sampling stage so that the generated samples are more likely to comply with the soft-constraints.

A Monte Carlo simulation study is carried out to evaluate the performances of the scPF and scAPF algorithms. The numerical results show that the proposed fast optimization method spends only a fraction of time used by the standard "fminunc" method in MATLAB with little price of performance degradation. The simulation study also shows the advantages of incorporating soft state constraints in particle sampling: the accuracy as measured by MSE is considerably improved in comparison with the standard APF for which the soft-constraint information is not utilized. The proposed scPF and scAPF also outperform the particle filters with hard constraints in terms of accuracy as measured by MSE.

The developed algorithms in this paper can be applied to a wide range of areas where the systems under investigation are nonlinear and/or non-Gaussian, and the state vectors are subject to soft constraints. Examples may include ground vehicle tracking, air traffic monitoring, maritime navigation, and the other areas beyond target tracking such as networked systems [36] and source term estimation [37]. Finally, we point out that, for those applications where a continuous-time dynamic system is discretized, the information provided by soft state constraints can be applied in a higher sampling rate than the sensor measurements, and hence having a potential to further increase the filtering performance.

APPENDIX
A FAST ALGORITHM FOR OPTIMIZATION PROBLEMS

In this appendix, we investigate a fast algorithm for the optimization problems in Section IV-B based on the quasi-Newton method, which only takes one or two iterations.

Suppose that the starting point of the ith optimization problem (20) at time step k is chosen as $\mathbf{x}_{(0)} = \bar{\mathbf{x}}_k^i$, where superscript i for particle index and the subscript k for time step are suppressed for the sake of simplicity. The quasi-Newton method uses the iteration $\mathbf{x}_{(m+1)} = \mathbf{x}_{(m)} + \tau_{(m)}\mathbf{d}_{(m)}$ to approach the optimal solution, where the subscript m denotes the number of iteration, $\tau_{(m)}$ is the step length, and $\mathbf{d}_{(m)}$ is the searching direction. The step length $\tau_{(m)}$ can be determined by an inexact line search satisfying the Wolfe condition. The search direction is defined as $\mathbf{d}_{(m)} = H^{-1}(\mathbf{x})\nabla J(\mathbf{x})|_{\mathbf{x}=\mathbf{x}_{(m)}}$, where ∇J and H are the gradient and Hessian matrix of the cost function, respectively, both evaluated at $\mathbf{x}_{(m)}$. By recalling the cost function (20), the gradient can be derived as

$$\nabla J(\mathbf{x}_{(m)}) = V_{k-1}^i(\mathbf{x}_{(m)} - \bar{\mathbf{x}}_k^i) + D_{k-1}^i + \nabla G_k(\mathbf{x}_{(m)}) \quad (32)$$

where

$$\begin{aligned}\nabla G_k(\mathbf{x}) &= -\sum_{j=1}^{n_c} \frac{\nabla \phi_j(\mathbf{x}_{(m)})}{\phi_j(\mathbf{x}_{(m)})} \\ &= \sum_{j=1}^{n_c} \frac{p_{\gamma_j}(g_{j,k}(\mathbf{x}_{(m)}))}{\phi_j(\mathbf{x}_{(m)})} \nabla g_{j,k}(\mathbf{x}_{(m)}).\end{aligned} \quad (33)$$

As the Hessian matrix is usually complicated and the cost for calculating its inverse can be expensive, the quasi-Newton method is adopted to approximate Hessian inverse using BFGS algorithm. Specifically we

1) Set $\mathbf{s}_{(m)} = \mathbf{x}_{(m+1)} - \mathbf{x}_{(m)}$ and $\mathbf{y}_{(m)} = \nabla J_{(m+1)} - \nabla J_{(m)}$;
2) Update Hessian inverse $H_{(m+1)}^{-1} = (I - \frac{\mathbf{s}_{(m)}\mathbf{y}_{(m)}^T}{\mathbf{y}_{(m)}^T\mathbf{s}_{(m)}})H_{(m)}^{-1}$ $(I - \frac{\mathbf{y}_{(m)}\mathbf{s}_{(m)}^T}{\mathbf{y}_{(m)}^T\mathbf{s}_{(m)}}) + \frac{\mathbf{s}_{(m)}\mathbf{s}_{(m)}^T}{\mathbf{y}_{(m)}^T\mathbf{s}_{(m)}}$.

Since this optimization problem is embedded in the PF framework, solving the optimization problem does not need to be accurate but fast. Thus, only one or two iterations will be used depending on the settings of the original Bayesian estimation problem.

For the special case where truncated Gaussian distributions $\mathcal{TN}(\gamma_j; 0, \sigma_j^2)$ are used, the gradient in (33) can be rewritten as

$$\nabla G_k(\mathbf{x}_{(m)}) = \sum_{j=1}^{n_c} \frac{\mathcal{TN}(g_{j,k}(\mathbf{x}_{(m)}); 0, \sigma_j^2) \nabla g_{j,k}(\mathbf{x}_{(m)})}{2 \prod_{j=1}^{n_c} [1 - \tilde{\Phi}(g_{j,k}(\mathbf{x}_{(m)}); 0, \sigma_j^2)]}. \quad (34)$$

Note that when the probability of satisfying jth constraint $\phi_{j,k}(\mathbf{x}_{(m)})$ is close to zero, the inequality relations of Mill's ratio for normal distributions [38] can be used to find an approximation of (34) in order to avoid any numerical issue.

On the other hand, if the exponential distribution $\mathcal{E}(\gamma_j; \mu_j)$ is used in (33), a much simpler expression can be derived, such that $\nabla G_k(\mathbf{x}_{(m)}) = \sum_{j=1}^{n_c} \mu_j^{-1} \nabla g_{j,k}(\mathbf{x}_{(m)})$.

REFERENCES

[1] Y. Bar-Shalom, X. Li, and T. Kirubarajan
Estimation with Applications to Tracking and Navigation: Theory Algorithms and Software. New York, NY, USA: Wiley, 2004.

[2] D. Simon
Optimal State Estimation: Kalman, H-Infinity, and Nonlinear Approaches. Hoboken, NJ, USA: Wiley, 2006.

[3] B. Ristic, S. Arulampalam, and N. Gordon
Beyond the Kalman Filter: Particle Filters for Tracking Applications Artech House radar library. London, U.K.: Artech House, 2004.

[4] M. Arulampalam, S. Maskell, N. Gordon, and T. Clapp
A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking
IEEE Trans. Signal Process., vol. 50, no. 2, pp. 174–188, Feb. 2002.

[5] A. Doucet and A. M. Johansen
A tutorial on particle filtering and smoothing: Fifteen years later
In *The Oxford Handbook of Nonlinear Filtering*, D. Crisan and B. Rozovsky, Eds. Oxford, U.K.: Oxford Univ. Press, 2011.

[6] F. Gustafsson
Particle filter theory and practice with positioning applications
IEEE Aerosp. Electron. Syst. Mag., vol. 25, no. 7, pp. 53–82, Jul. 2010.

[7] L. Snidaro, J. Garca, and J. Llinas
Context-based information fusion: A survey and discussion
Inf. Fusion, vol. 25, pp. 16–31, 2015.

[8] M. Ulmke and W. Koch
Road-map assisted ground moving target tracking
IEEE Trans. Aerosp. Electron. Syst., vol. 42, no. 4, pp. 1264–1274, Oct. 2006.

[9] Y. Cheng and T. Singh
Efficient particle filtering for road-constrained target tracking
IEEE Trans. Aerosp. Electron. Syst., vol. 43, no. 4, pp. 1454–1469, Oct. 2007.

[10] F. Papi, M. Podt, Y. Boers, G. Battistello, and M. Ulmke
On constraints exploitation for particle filtering based target tracking
In *Proc. 15th Int. Conf. Inf. Fusion*, Jul. 2012, pp. 455–462.

[11] M. Bocquel, F. Papi, M. Podt, and H. Driessen
Multitarget tracking with multiscan knowledge exploitation using sequential MCMC sampling
IEEE J. Sel. Topics Signal Process., vol. 7, no. 3, pp. 532–542, Jun. 2013.

[12] E. Sviestins and V. Pirard
Constraint aware dynamics in target tracking
In *Proc. 16th Int. Conf. Inf. Fusion*, Jul. 2013, pp. 572–579.

[13] X. Shao, B. Huang, and J. M. Lee
Constrained Bayesian state estimation: A comparative study and a new particle filter based approach
J. Process Control, vol. 20, no. 2, pp. 143–157, 2010.

[14] J. Prakash, S. C. Patwardhan, and S. L. Shah
On the choice of importance distributions for unconstrained and constrained state estimation using particle filter
J. Process Control, vol. 21, no. 1, pp. 3–16, 2011.

[15] D. Simon
Kalman filtering with state constraints: A survey of linear and nonlinear algorithms
IET Control Theory Appl., vol. 4, no. 8, pp. 1303–1318, 2010.

[16] O. Straka, J. Dunik, and M. Simandl
Truncation nonlinear filters for state estimation with nonlinear inequality constraints
Automatica, vol. 48, no. 2, pp. 273–286, 2012.

[17] Z. Zhao, B. Huang, and F. Liu
Constrained particle filtering methods for state estimation of nonlinear process
AIChE J., vol. 60, no. 6, pp. 2072–2082, 2014.

[18] B. Li, C. Liu, and W. Chen
An auxiliary particle filtering algorithm with inequality constraints
IEEE Trans. Autom. Control, vol. 62, no. 9, pp. 4639–4646, Sep. 2017.

[19] C. Seah and I. Hwang
Stochastic linear hybrid systems modeling, estimation, and application in air traffic control
IEEE Trans. Control Syst. Technol., vol. 17, no. 3, pp. 563–575, May 2009.

[20] O. McAree and W.-H. Chen
Artificial situation awareness for increased autonomy of unmanned aerial systems in the terminal area
J. Intell. Robot. Syst., vol. 70, nos. 1–4, pp. 545–555, 2013.

[21] C. Liu, M. Coombes, B. Li, and W.-H. Chen
Enhanced situation awareness for unmanned aerial vehicle operating in terminal areas with circuit flight rules
Proc. IMechE, Part G: J. Aerosp. Eng., vol. 230, no. 9, pp. 1683–1693, 2016.

[22] D. Simon and D. Simon
Kalman filtering with inequality constraints for turbofan engine health estimation
IEE Proc.—Control Theory Appl., vol. 153, no. 3, pp. 371–378, May 2006.

[23] A. Benavoli, L. Chisci, A. Farina, L. Ortenzi, and G. Zappa
Hard-constrained versus soft-constrained parameter estimation
IEEE Trans. Aerosp. Electron. Syst., vol. 42, no. 4, pp. 1224–1239, Oct. 2006.

[24] Y. Hel-Or , A. Rappoport, and M. Werman
Relaxed parametric design with probabilistic constraints
In *Proc. 2nd ACM Symp. Solid Model. Appl.*, 1993, pp. 261–270.

[25] A. Palmer, A. Hill, and S. Scheding
Applying Gaussian distributed constraints to Gaussian distributed variables
Inf. Fusion, vol. 32, pp. 1–11, 2016.

[26] M. K. Pitt and N. Shephard
Auxiliary variable based particle filters
In *Sequential Monte Carlo Methods in Practice*, A. Doucet, N. Freitas, and N. Gordon, Eds. New York, NY, USA: Springer, 2001.

[27] B. Ristic
Bayesian estimation with imprecise likelihoods: Random set approach
IEEE Signal Process. Lett., vol. 18, no. 7, pp. 395–398, Jul. 2011.

[28] A. Doucet, S. Godsill, and C. Andrieu
On sequential Monte Carlo sampling methods for Bayesian filtering
Statist. Comput., vol. 10, no. 3, pp. 197–208, 2000.

[29] A. Kong, J. S. Liu, and W. H. Wong
Sequential imputations and Bayesian missing data problems
J. Am. Stat. Assoc., vol. 89, no. 425, pp. 278–288, 1994.

[30] M. K. Pitt and N. Shephard
Filtering via simulation: Auxiliary particle filters
J. Am. Stat. Assoc., vol. 94, no. 446, pp. 590–599, 1999.

[31] A. M. Johansen and A. Doucet
A note on auxiliary particle filters
Statist. Probab. Lett., vol. 78, no. 12, pp. 1498–1504, 2008.

[32] P. Skoglar, U. Orguner, D. Tornqvist, and F. Gustafsson
Road target search and tracking with gimballed vision sensor on an unmanned aerial vehicle
Remote Sens., vol. 4, no. 7, pp. 2076–2111, 2012.

[33] X. Li and V. Jilkov
Survey of maneuvering target tracking: Part I. Dynamic models
IEEE Trans. Aerosp. Electron. Syst., vol. 39, no. 4, pp. 1333–1364, Oct. 2003.

[34] L. Xu, X. Li, Z. Duan, and J. Lan
Modeling and state estimation for dynamic systems with linear equality constraints
IEEE Trans. Signal Process., vol. 61, no. 11, pp. 2927–2939, Jun. 2013.

[35] L. Martino, V. Elvira, and F. Louzada
Effective sample size for importance sampling based on discrepancy measures
Signal Process., vol. 131, pp. 386–401, 2017.

[36] W. Li, Z. Wang, Y. Yuan, and L. Guo
Particle filtering with applications in networked systems: a survey
Complex Intell. Syst., vol. 2, no. 4, pp. 293–315, Dec. 2016.

[37] M. Hutchinson, C. Liu, and W.-H. Chen
Source term estimation of a hazardous airborne release using an unmanned aerial vehicle
J. Field Robot., pp. 1–21, 2018, doi: 10.1002/rob.21844.

[38] J. Patel and C. Read
Handbook of the Normal Distribution Statistics: A Series of Textbooks and Monographs. New York, NY, USA: Taylor & Francis, 1996.

Cunjia Liu (M'16) received the B.Eng. and M.Sc. degrees in guidance, navigation, and control from Beihang University, Beijing, China, in 2005 and 2008, respectively. In 2011, he received the Ph.D. degree in autonomous vehicle control from Loughborough University, Loughborough, Leicestershire, U.K.

He was a Research Associate with the Department of Aeronautical and Automotive Engineering, Loughborough University, where he was appointed as a Lecturer and then a Senior Lecturer in unmanned vehicles in 2013 and 2018, respectively. His current research interests include optimization-based control, disturbance-observer-based control, Bayesian information fusion and their applications to autonomous vehicles for flight control, path planning, decision making, and situation awareness.

Baibing Li (M'13–SM'19) received the B.Sc. degree from Yunnan University, Kunming, China, the M.Sc. degree from Shanghai Jiao Tong University, Shanghai, China, and the M.Sc. degree from Vrije Universiteit Brussel, Brussels, Belgium. In 1991, he received the Ph.D. degree from the Management School, Shanghai Jiao Tong University.

He was a Postdoctoral Research Fellow with Katholieke Universiteit Leuven, Leuven, Belgium, and a Research Associate with Newcastle University, Newcastle upon Tyne, U.K. In 2001, he was appointed as a Lecturer with Newcastle University. In 2004, he moved to the School of Business and Economics, Loughborough University, Loughborough, U.K., as a Lecturer, where he was subsequently appointed as a Reader in 2007 and a Professor in 2011. His current research interests include Bayesian statistical modeling and forecasting for Gaussian and non-Gaussian dynamic problems in various management areas. In recent years, much of his work has also involved transport and traffic management such as transportation demand analysis, travel behavior modeling, and intelligent transportation systems.

Dr. Li is a Member of the Royal Statistical Society.

Wen-Hua Chen (M'00–SM'06–F'18) received the M.Sc. and Ph.D. degrees from the Department of Automatic Control, Northeast University, Shenyang, China, in 1989 and 1991, respectively.

He is currently a Professor in autonomous vehicles with the Department of Aeronautical and Automotive Engineering, Loughborough University, U.K. From 1991 to 1996, he was a Lecturer with the Department of Automatic Control, Nanjing University of Aeronautics and Astronautics, Nanjing, China. He held a research position and then a Lectureship in control engineering with the Centre for Systems and Control, University of Glasgow, Glasgow, U.K., from 1997 to 2000. He has published 3 books and 250 papers in journals and conferences. His research interests include the development of advanced control strategies and their applications in aerospace engineering, particularly in unmanned aircraft systems.

Dr. Chen is a Fellow of the Institution of Engineering and Technology and the Institution of Mechanical Engineers.