Approximate Estimators for Linear Systems With Additive Cauchy Noises

Robert Fonod* and Moshe Idan†
Technion - Israel Institute of Technology, Haifa 3200003, Israel
Jason L. Speyer‡
University of California, Los Angeles, California 90095, USA

The recently published optimal Cauchy estimator poses practical implementation challenges due to its time-growing complexity. Alternatively, addressing impulsive measurement and process noises while using common estimation approaches requires heuristic schemes. Approximate methods, such as particle and Gaussian sum filters, were suggested to tackle the estimation problem in heavy-tailed noise environment when constraining the computational load. In this paper, the performance of a particle filter and a Gaussian sum filter, designed for a linear system with specified Cauchy noise parameters, are compared numerically to a Cauchy filter-based approximation showing the advantages of the latter.

I. Introduction

Impulsive processes appear naturally in a variety of practical problems that range from engineering and science to economics and finance. In many applications the underlying random processes or noises are better described by heavy-tailed non-Gaussian densities [1], for example by the Cauchy probability density function (pdf) [2]. Traditional filtering techniques often rely on the Gaussian assumption mainly because modern methods and algorithms are able to handle such systems very efficiently [3], yielding, e.g., the Kalman filter[a] [4]. However, in the presence of significantly non-Gaussian, heavy-tailed noises, and particularly in the presence of outliers, the performance of the Kalman filter degrades severely [5].

Impulsive measurement and process noises in stochastic state estimators have typically been handled by heuristic schemes that augment the estimation process. Recently, an analytical recursive non-linear estimation scheme, i.e., the Idan/Speyer Cauchy Estimator (ISCE), for linear scalar systems driven by Cauchy distributed process and measurement noises has been developed using pdf [6] and characteristic function (cf) [7, 8] approaches. Specifically, the latter recursively generates, in closed-form, the *characteristic function* of the unnormalized conditional pdf (ucpdf) of the state given the measurement history. The number of terms in the sum that expresses this cf grows with each measurement update.

Although the ISCE provides the exact minimum conditional variance estimates of the system's states given a measurement sequence, its computational complexity and memory burden becomes very high, requiring an approximation in its implementation. One such approximation was suggested in [6, 7] for the scalar and one in [9, 10] for the two-state ISCE, respectively. For both cases, a fixed sliding window of the most recent measurements was considered to attain a near optimal estimate, leading to an estimator with a finite computational burden.

Alternatively, in this study we wish to evaluate other general estimation algorithms in addressing impulsive noises [11–14]. Although those approaches are suboptimal they may offer reasonable approximations when tuned for the heavy-tailed, Cauchy noise environment. Two of the most popular approximations are

*Postdoctoral Fellow, Faculty of Aerospace Engineering, robert.fonod@technion.ac.il.
†Associate Professor, Faculty of Aerospace Engineering, moshe.idan@technion.ac.il. Associate Fellow AIAA.
‡Ronald and Valerie Sugar Distinguished Professor in Engineering, Department of Mechanical and Aerospace Engineering, speyer@ucla.edu. Fellow AIAA.

[a]The Kalman filter is the minimum variance filter if the additive noise is Gaussian and the best linear minimum variance filter if the noise is non-Gaussian, but the second order statistics for the additive noise exists.

the particle filter (PF) and the Gaussian sum filter (GSF) in that they were shown to converge to the correct conditional density of the state as the number of terms in their implementation tends to infinity. Therefore, for real-time applications, they are implemented with some degree of approximation, producing a tradeoff between numerical efficiency and the estimation performance in constructing the conditional pdf of the state given the measurement history and the resulting conditional mean and variance.

Using statistical Monte Carlo (MC) simulations and a judiciously chosen measure to evaluate the estimation performance of a heavy-tailed process, we extend the sample run results presented in [15]. Our main objective is to compare the efficiency of the PF and GSF to that of the scalar and two-state ISCE-based approximations, hence providing guidance for a practical implementation of an estimator for a heavy-tail noise environment.

II. Problem Formulation

Consider a single-input-single-output, multivariate, discrete-time and time-invariant linear dynamic system described by

$$\boldsymbol{x}_{k+1} = \boldsymbol{\Phi}\boldsymbol{x}_k + \boldsymbol{\Gamma}w_k, \tag{1}$$

$$z_k = \boldsymbol{H}\boldsymbol{x}_k + v_k, \tag{2}$$

with state vector $\boldsymbol{x}_k \in \mathbb{R}^n$, scalar measurement $z_k \in \mathbb{R}$, and known matrices $\boldsymbol{\Phi} \in \mathbb{R}^{n\times n}$, $\boldsymbol{\Gamma} \in \mathbb{R}^{n\times 1}$, and $\boldsymbol{H} \in \mathbb{R}^{1\times n}$. The sequence $\boldsymbol{x}_k$ for $k = 1, 2, \ldots$, is a discrete-time Markov process. The scalar noise inputs w_k and v_k are independent Cauchy distributed random variables with zero median and scaling parameters $\beta > 0$ and $\gamma > 0$, respectively. Their pdf-s and cf-s are denoted p and ϕ, respectively. They are assumed to be time independent and given by

$$p_W(w_k) = \frac{\beta/\pi}{w_k^2 + \beta^2} \quad \Rightarrow \quad \phi_W(\bar{\nu}) = e^{-\beta|\bar{\nu}|}, \tag{3}$$

$$p_V(v_k) = \frac{\gamma/\pi}{v_k^2 + \gamma^2} \quad \Rightarrow \quad \phi_V(\bar{\nu}) = e^{-\gamma|\bar{\nu}|}, \tag{4}$$

where $\bar{\nu}$ is a scalar spectral variable.

The initial conditions at $k = 1$ are also assumed to be independent Cauchy distributed random variables, i.e., each element $x_{1;i}$ of the initial state $\boldsymbol{x}_1$ has a Cauchy pdf with a zero median and a scaling parameter $\alpha_i > 0$, $i = 1, \ldots, n$. The joint pdf of the initial conditions and its cf are given by

$$p_{X_1}(\boldsymbol{x}_1) = \prod_{i=1}^{n} \frac{\alpha_i/\pi}{x_{1;i}^2 + \alpha_i^2} \quad \Rightarrow \quad \phi_{X_1}(\boldsymbol{\nu}) = \prod_{i=1}^{n} e^{-\alpha_i|\nu_i|}, \tag{5}$$

where ν_i is an element of the n-dimensional spectral variable $\boldsymbol{\nu} \in \mathbb{R}^n$.

The measurement history used in the estimation problem formulation is defined as $\boldsymbol{z}_{1:k} \triangleq \{z_1, \ldots, z_k\}$. The objective is to examine the practically computable minimum conditional variance estimates of $\boldsymbol{x}_k$ given the measurement history $\boldsymbol{z}_{1:k}$.

III. Bayesian Approach

From a Bayesian perspective, the filtering problem is solved by constructing the posterior density $p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k})$[b] of the state $\boldsymbol{x}_k$ at time k given all the available information $\boldsymbol{z}_{1:k}$. The pdf $p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k})$ contains all available statistical information, and thus is the complete solution to the estimation problem [3]. In principle, an optimal (with respect to any criterion) estimate of the state may be obtained from this pdf.

Given the posterior pdf $p(\boldsymbol{x}_{k-1}|\boldsymbol{z}_{1:k-1})$ at time $k-1$, the posterior pdf $p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k})$ at time k can be computed using a two stage approach of prediction and update. For square systems (i.e., $\text{rank}(\boldsymbol{\Gamma}) = n$), the prediction stage can be obtained via the Chapman-Kolmogorov equation

$$p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k-1}) = \int p(\boldsymbol{x}_{k-1}|\boldsymbol{z}_{1:k-1}) p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}) \mathrm{d}\boldsymbol{x}_{k-1}, \tag{6}$$

[b]For simplicity, $p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k})$ denotes $p_{X_k|Z_{1:k}}(\boldsymbol{x}_k|\boldsymbol{z}_{1:k})$. This notation simplification is used hereafter to avoid extensive notation, whenever the context is clear.

where $p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1})$ is the state transition probability density determined by the state equation (1) and the known statistics of the process noise w_{k-1}, i.e.,

$$p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}) = p_W\left(\boldsymbol{\Gamma}^{-1}(\boldsymbol{x}_k - \boldsymbol{\Phi}\boldsymbol{x}_{k-1})\right). \tag{7}$$

For non-square (i.e., $\text{rank}(\boldsymbol{\Gamma}) \neq n$) and non-singular systems (i.e., $\text{rank}(\boldsymbol{\Phi}) = n$), the prediction stage (6) can be computed via [8]

$$p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k-1}) = \left|\boldsymbol{\Phi}^{-1}\right| \int p_{X_{k-1}|Z_{1:k-1}}\left(\boldsymbol{\Phi}^{-1}\boldsymbol{x}_k - \boldsymbol{\Phi}^{-1}\boldsymbol{\Gamma}w_{k-1} \,\middle|\, \boldsymbol{z}_{1:k-1}\right) p_W(w_{k-1})\mathrm{d}w_{k-1}, \tag{8}$$

where $|\cdot|$ stands for determinant of a matrix.

If at time k a measurement z_k becomes available, then z_k can be used to update the prior pdf $p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k-1})$ via Bayes' rule[c]

$$p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k}) = \frac{p(z_k|\boldsymbol{x}_k)p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k-1})}{\int p(z_k|\boldsymbol{x}_k)p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k-1})\mathrm{d}\boldsymbol{x}_k}, \tag{9}$$

where $p(z_k|\boldsymbol{x}_k)$ is the measurement likelihood defined by the measurement model (2) and the known statistics of the measurement noise v_k, i.e.,

$$p(z_k|\boldsymbol{x}_k) = p_V\left(z_k - \boldsymbol{H}\boldsymbol{x}_k\right). \tag{10}$$

Given $p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k})$, the minimum conditional variance estimate $\hat{\boldsymbol{x}}_k$ and the second conditional moment of the state are given by

$$\hat{\boldsymbol{x}}_k = \mathrm{E}[\boldsymbol{x}_k|\boldsymbol{z}_{1:k}] = \int \boldsymbol{x}_k p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k})\mathrm{d}\boldsymbol{x}_k, \tag{11}$$

$$\mathrm{E}[\boldsymbol{x}_k\boldsymbol{x}_k^T|\boldsymbol{z}_{1:k}] = \int \boldsymbol{x}_k\boldsymbol{x}_k^T p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k})\mathrm{d}\boldsymbol{x}_k. \tag{12}$$

The minimal conditional variance of the estimation error defined as $\tilde{\boldsymbol{x}}_k \triangleq \boldsymbol{x}_k - \hat{\boldsymbol{x}}_k$ is then determined by

$$\boldsymbol{P}_k = \mathrm{E}[\tilde{\boldsymbol{x}}_k\tilde{\boldsymbol{x}}_k^T|\boldsymbol{z}_{1:k}] = \mathrm{E}[\boldsymbol{x}_k\boldsymbol{x}_k^T|\boldsymbol{z}_{1:k}] - \hat{\boldsymbol{x}}_k\hat{\boldsymbol{x}}_k^T. \tag{13}$$

IV. Optimal Solution

Here we present a brief overview of the analytical estimation solution to the Cauchy problem, the ISCE.

IV.A. Scalar ISCE: pdf Approach

The pioneering work of Idan and Speyer[6] derived the ISCE for scalar-state systems by solving, in closed-form, the integrals of the Bayesian update rule involved in constructing the posterior pdf $p(x_k|\boldsymbol{z}_{1:k})$. It was shown that under mild conditions on the system and noise parameters (see Assumption 4.1 in [6]), $p(x_k|\boldsymbol{z}_{1:k})$ can be expressed as

$$p(x_k|\boldsymbol{z}_{1:k}) = \sum_{i=1}^{k+2} \frac{a_{k|k}^i x_k + b_{k|k}^i}{(x_k - \sigma_{k|k}^i)^2 + (\omega_{k|k}^i)^2}. \tag{14}$$

Initialization and update rules for the series coefficients $a_{k|k}^i$, $b_{k|k}^i$, $\sigma_{k|k}^i$ and $\omega_{k|k}^i$ can be found in [6]. All but $\omega_{k|k}^i$ are functions of the measurements. The series coefficients of the above pdf are used to determine $\hat{x}_k$ and P_k as

$$\hat{x}_k = \pi \sum_{i=1}^{k+2} \frac{a_{k|k}^i[(\sigma_{k|k}^i)^2 - (\omega_{k|k}^i)^2] + b_{k|k}^i\sigma_{k|k}^i}{\omega_{k|k}^i}, \tag{15}$$

$$P_k = \pi \sum_{i=1}^{k+2} \frac{h_{k|k}^i - 2a_{k|k}^i\sigma_{k|k}^i(\omega_{k|k}^i)^2}{\omega_{k|k}^i} - \hat{x}_k^2, \tag{16}$$

where $h_{k|k}^i = (a_{k|k}^i\sigma_{k|k}^i + b_{k|k}^i)((\sigma_{k|k}^i)^2 - (\omega_{k|k}^i)^2)$.

The approach above provides a closed-form expression for $p(x_k|\boldsymbol{z}_{1:k})$, which can be examined for its shape and additional properties, as will be carried out in this study.

[c] At time $k = 1$, the prior pdf is defined as: $p(\boldsymbol{x}_1|\boldsymbol{z}_{1:0}) \triangleq p_{X_1}(\boldsymbol{x}_1)$.

IV.B. Multivariate ISCE: cf Approach

Since the pdf approach did not extend to multivariate systems, an alternative derivation that utilizes the characteristic function of the ucpdf was proposed. The ISCE for scalar-state systems was re-derived first using this approach in [7]. Subsequently, it was extended to multivariate systems in [8].

Here, we propagate the cf of $p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k})$ given by

$$\phi_{\boldsymbol{x}_k|\boldsymbol{z}_{1:k}}(\boldsymbol{\nu}) = \int p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k}) e^{j\boldsymbol{\nu}^T\boldsymbol{x}_k} \mathrm{d}\boldsymbol{x}_k. \tag{17}$$

Moreover, for computational simplicity, the normalization by $p(z_k|\boldsymbol{z}_{1:k-1})$ when computing $p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k})$ can be postponed, thus propagating the cf of the ucpdf

$$\bar{\phi}_{\boldsymbol{x}_k|\boldsymbol{z}_{1:k}}(\boldsymbol{\nu}) = \int p(\boldsymbol{x}_k, \boldsymbol{z}_{1:k}) e^{j\boldsymbol{\nu}^T\boldsymbol{x}_k} \mathrm{d}\boldsymbol{x}_k. \tag{18}$$

From the Bayesian update rule, $p(\boldsymbol{x}_k, \boldsymbol{z}_{1:k})$ is the ucpdf of the state, while the normalization factor to obtain $p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k})$ is given by $\bar{\phi}_{\boldsymbol{x}_k|\boldsymbol{z}_{1:k}}(\boldsymbol{0})$.

In [8], it was shown that $\bar{\phi}_{\boldsymbol{x}_k|\boldsymbol{z}_{1:k}}(\boldsymbol{\nu})$ at the update time k can be expressed as a growing sum of $n_t^{k|k}$ terms

$$\bar{\phi}_{\boldsymbol{x}_k|\boldsymbol{z}_{1:k}}(\boldsymbol{\nu}) = \sum_{i=1}^{n_t^{k|k}} g_i^{k|k}\left(y_{gi}^{k|k}(\boldsymbol{\nu})\right) \exp\left(y_{ei}^{k|k}(\boldsymbol{\nu})\right), \tag{19}$$

i.e., a sum of exponential terms multiplied by a coefficient function, $g_i^{k|k}(\cdot)$, that is a complex, nonlinear function of the measurements. The argument of this coefficient function, $y_{gi}^{k|k}(\boldsymbol{\nu})$, is real and is expressed as a sum of sign functions of $\boldsymbol{\nu}$ with known parameters. The real part of the argument of the exponents, $y_{ei}^{k|k}(\boldsymbol{\nu})$, is the absolute value of a function of the spectral vector $\boldsymbol{\nu}$, and its imaginary part is a linear function of the measurements. The details of the various parameters and functions of the above expression can be found in [8]. Since $\bar{\phi}_{\boldsymbol{x}_k|\boldsymbol{z}_{1:k}}(\boldsymbol{\nu})$ is twice continuously differentiable, the minimum conditional variance estimate of the state can be obtained by [8]

$$\hat{\boldsymbol{x}}_k = \frac{1}{jp_{Z_{1:k}}(\boldsymbol{z}_{1:k})} \sum_{i=1}^{n_t^{k|k}} g_i^{k|k}\left(y_{gi}^{k|k}(\hat{\boldsymbol{\nu}})\right) \bar{y}_{ei}^{k|k}(\hat{\boldsymbol{\nu}}), \tag{20}$$

where j is the imaginary unit, $\hat{\boldsymbol{\nu}}$ is a fixed direction in the $\boldsymbol{\nu}$ domain, $\bar{y}_{ei}^{k|k}(\hat{\boldsymbol{\nu}})$ is a n-dimensional vector which relates to $y_{ei}^{k|k}(\hat{\boldsymbol{\nu}})$ by $\langle \bar{y}_{ei}^{k|k}(\hat{\boldsymbol{\nu}}), \hat{\boldsymbol{\nu}}\rangle = y_{ei}^{k|k}(\hat{\boldsymbol{\nu}})$, and $p_{Z_{1:k}}(\boldsymbol{z}_{1:k})$ is

$$p_{Z_{1:k}}(\boldsymbol{z}_{1:k}) = \bar{\phi}_{\boldsymbol{x}_k|\boldsymbol{z}_{1:k}}(\epsilon\hat{\boldsymbol{\nu}})\big|_{\epsilon=0} = \sum_{i=1}^{n_t^{k|k}} g_i^{k|k}\left(y_{gi}^{k|k}(\hat{\boldsymbol{\nu}})\right). \tag{21}$$

The estimation error covariance matrix is obtained by

$$\boldsymbol{P}_k = \frac{1}{j^2 p_{Z_{1:k}}(\boldsymbol{z}_{1:k})} \sum_{i=1}^{n_t^{k|k}} g_i^{k|k}\left(y_{gi}^{k|k}(\hat{\boldsymbol{\nu}})\right)\left(\bar{y}_{ei}^{k|k}(\hat{\boldsymbol{\nu}})\right)\left(\bar{y}_{ei}^{k|k}(\hat{\boldsymbol{\nu}})\right)^T - \hat{\boldsymbol{x}}_k\hat{\boldsymbol{x}}_k^T. \tag{22}$$

V. Suboptimal Solutions

In this section, the PF and GSF are designed to approximate the posterior density $p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k})$ for the system described by (1) and (2), in which the noise sequences w_k and v_k and the initial state $\boldsymbol{x}_1$ are Cauchy distributed. As was observed above, the exact posterior pdf (14) or its cf (19) are expressed in the optimal ISCE as a series with a growing number of terms. To avoid the associated computational burden, [6, 7, 9, 10] suggest a truncation procedure that limits the number of terms in these series to a prescribed fixed sliding window of the n_s most recent measurements. The validity of these approximations was demonstrated even when using only around twenty terms for the scalar case and a window of eight, i.e., around 3000 terms, for the two-state ISCE. Consequently, only approximate ISCE implementations were considered in this study when comparing the performance with the proposed PF and GSF.

V.A. Particle Filter

The particle filter, also known as the sequential Monte Carlo method, is a set of algorithms implementing recursive Bayesian estimation based on point mass representations of probability densities. For good surveys, see [16, 17].

The main idea of the PF is to represent the posterior density $p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k})$ using a set of random samples with associated weights. Let $\{\boldsymbol{x}_k^i, \mu_k^i\}_{i=1}^{n_p}$ be such a representation that characterizes the posterior $p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k})$ at time k, where $\{\boldsymbol{x}_k^i\}_{i=1}^{n_p}$ is a set of n_p support points (particles) with associated nonnegative weights $\{\mu_k^i\}_{i=1}^{n_p}$. The weights are normalized such that $\sum_{i=1}^{n_p} \mu_k^i = 1$.

V.A.1. Sequential Importance Sampling

Most PFs are based on the algorithm known as sequential importance sampling (SIS), also known as the *bootstrap filter* [11], which is a MC technique for solving the Bayesian problem. Given $\{\boldsymbol{x}_{k-1}^i, \mu_{k-1}^i\}_{i=1}^{n_p}$ and z_k, the posterior density $p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k})$ at time k can be approximated using the principle of importance sampling, Markov property, and Bayes's rule as follows [17]

$$p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k}) \approx \sum_{i=1}^{n_p} \mu_k^i \delta(\boldsymbol{x}_k - \boldsymbol{x}_k^i), \tag{23}$$

where $\delta(\cdot)$ stands for the Dirac delta density. In (23), the ith particle $\boldsymbol{x}_k^i$ is sampled from the chosen importance density $q(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}^i, z_k)$, also known as the proposal density. The ith importance weight μ_k^i associated with $\boldsymbol{x}_k^i$ is updated as

$$\mu_k^i \propto \mu_{k-1}^i \frac{p(z_k|\boldsymbol{x}_k^i)p(\boldsymbol{x}_k^i|\boldsymbol{x}_{k-1}^i)}{q(\boldsymbol{x}_k^i|\boldsymbol{x}_{k-1}^i, z_k)}, \quad i = 1, 2, \ldots, n_p, \tag{24}$$

where $p(z_k|\boldsymbol{x}_k^i)$ is the measurement likelihood given by (10), $p(\boldsymbol{x}_k^i|\boldsymbol{x}_{k-1}^i)$ is the state transition density given by (7), and the symbol $\propto$ signifies "proportional to".

The SIS algorithm thus consists of recursive propagation of the weights and particles as each measurement is received sequentially. Based on the strong law of large numbers, as $n_p \to \infty$, the approximated posterior of (23) approaches the true posterior $p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k})$ [17]. The numerical approximation to (11) and (13), respectively, is computed as

$$\hat{\boldsymbol{x}}_k \approx \sum_{i=1}^{n_p} \mu_k^i \boldsymbol{x}_k^i, \tag{25}$$

$$\boldsymbol{P}_k \approx \sum_{i=1}^{n_p} \mu_k^i (\boldsymbol{x}_k^i - \hat{\boldsymbol{x}}_k)(\boldsymbol{x}_k^i - \hat{\boldsymbol{x}}_k)^T. \tag{26}$$

V.A.2. Choice of the Importance Density

It has been shown in [12] that the optimal choice for the importance density that minimises the variance of the importance weight μ_k^i is

$$q(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}^i, z_k) = p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}^i, z_k) = \frac{p(z_k|\boldsymbol{x}_k, \boldsymbol{x}_{k-1}^i)p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}^i)}{p(z_k|\boldsymbol{x}_{k-1}^i)}. \tag{27}$$

However, this importance density is not always available and can be used only in special cases, e.g., for a class of models for which $p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}^i, z_k)$ is Gaussian [12]. Hence, the most widely used importance density is the prior pdf [17], i.e.

$$q(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}^i, z_k) = p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}^i). \tag{28}$$

This choice of importance density means that for $k \geq 2$, we need to sample particles from $p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}^i)$. A sample $\boldsymbol{x}_k^i \sim p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}^i)$ can be obtained by first generating a process noise sample $w_{k-1} \sim p_W(w_{k-1})$ and setting $\boldsymbol{x}_k^i = \boldsymbol{\Phi}\boldsymbol{x}_{k-1}^i + \boldsymbol{\Gamma}w_{k-1}^i$, where $p_W(w_{k-1})$ is the Cauchy process noise pdf defined in (3). For $k = 1$,

the particles are generated from the initial density, i.e., $\boldsymbol{x}_1^i \sim p_{X_1}(\boldsymbol{x}_1)$, where $p_{X_1}(\boldsymbol{x}_1)$ is the initial state pdf defined in (5). Additionally, the weight update formula (24) reduces to

$$\mu_k^i \propto \mu_{k-1}^i p(z_k|\boldsymbol{x}_k^i) = \mu_{k-1}^i p_V(z_k - \boldsymbol{H}\boldsymbol{x}_k^i), \quad i = 1, 2, \ldots, n_p, \tag{29}$$

where $p_V(\cdot)$ is the Cauchy measurement noise pdf defined in (4).

Notice that the importance density of (28) is independent of the measurement z_k and hence the state space is explored without knowledge of the actual observation. This choice of $q(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}^i, z_k)$ may fail if new measurements appear in the tail of the prior or if the measurement likelihood is too peaked in comparison to the prior. This strategy promotes a well known problem of the SIS algorithm, known as the degeneracy (or sample impoverishment) problem [17]. On the other hand, the advantage of the SIS algorithm is that its computational burden is constant at each time step.

V.A.3. Degeneracy Problem

The degeneracy problem arises when after a few iterations of the SIS algorithm, only a few of the particles have significant weights while the other particles have negligible weights. This yields a very poor approximation of $p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k})$, and may lead to a breakdown of the algorithm. Note, this phenomenon occurs even if the optimal importance density (27) is used, but is more severe when using $p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}^i)$.

A suitable measure to assess the degeneracy of the SIS algorithm is the effective sample size estimate [17]

$$\hat{n}_p^{eff} = 1/\sum_{i=1}^{n_p} (\mu_k^i)^2. \tag{30}$$

Here, $1 \leq \hat{n}_p^{eff} \leq n_p$, where the upper bound is attained when all particles have the same weight, and the lower bound when the entire probability mass is at one particle. Small $\hat{n}_p^{eff}$ indicates severe degeneracy.

V.A.4. Resampling

The most common solution to tackle the degeneracy problem is the use of resampling. It discards particles that have low importance weights, as they do not contribute to the approximation, and replaces them with particles in the vicinity of those with high importance weights [11]. To prevent degeneracy when $\hat{n}_p^{eff}$ is below a fixed threshold n_p^t, an appropriate resampling procedure is utilized.

Several resampling schemes exist. The choice of the particular resampling scheme affects the computational load as well as the approximation error, see the discussion and classification in [18]. In our study, only the *systematic resampling* strategy [19] was used as it was shown empirically to outperform other methods for the Cauchy case.

V.B. Gaussian Sum Filter

In this subsection, the problem of determining a posterior density $p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k})$ is treated using the Gaussian sum approximation. The presented filtering scheme is an adaptation of the well-known GSF algorithm of Sorenson and Alspach [13] to the system described by (1) and (2), in which the noise sequences w_k and v_k and the initial state $\boldsymbol{x}_1$ are Cauchy distributed.

V.B.1. Gaussian Mixture Model

Based on the Wiener approximation theory [20], any pdf can be expressed, or approximated with a given level of accuracy, using a weighted sum of Gaussian densities, what is also known as a Gaussian mixture model (GMM) given by

$$p_X^{\mathcal{G}}(\boldsymbol{x}) = \sum_{i=1}^{n_x} \mu_x^i \, \mathcal{N}\left(\boldsymbol{x}; \bar{\boldsymbol{x}}_x^i, \boldsymbol{P}_x^i\right). \tag{31}$$

Here n_x is a positive integer indicating the number of Gaussian components (terms) in the GMM, $\mu_x^i \geq 0$, $i = 1, 2, \ldots, n_x$ are scalar weighting factors satisfying $\sum_{i=1}^{n_x} \mu_x^i = 1$, and $\mathcal{N}\left(\boldsymbol{x}; \bar{\boldsymbol{x}}_x^i, \boldsymbol{P}_x^i\right)$ denotes a multivariate Gaussian density function with argument $\boldsymbol{x} \in \mathbb{R}^m$, mean $\bar{\boldsymbol{x}}_x^i \in \mathbb{R}^m$, and covariance matrix $\boldsymbol{P}_x^i \in \mathbb{R}^{m \times m}$. It can be shown that $p_X^{\mathcal{G}}(\boldsymbol{x})$ is a valid density function and converges uniformly to any density of practical concern by letting n_x increase and each elemental covariance approach the zero matrix.

V.B.2. Fitting a GMM to a Cauchy Density

To obtain the recursive Bayesian filter in the GMM framework, first the stationary Cauchy densities given in (3), (4), and (5) need to be fitted (approximated) by a GMM. The fitting can be done in various ways. In this paper, this fitting is formulated as the following constrained optimization problem: Given the desired number of Gaussian components ($n_x \geq 1$), find $\{\mu_x^i, \bar{\boldsymbol{x}}_x^i, \boldsymbol{P}_x^i\}_{i=1}^{n_x}$ which minimizes the integral square difference (ISD) between a particular Cauchy density of interest and a GMM, i.e.,

$$\underset{\{\mu_x^i, \bar{\boldsymbol{x}}_x^i, \boldsymbol{P}_x^i\}_{i=1}^{n_x}}{\operatorname{argmin}} J_f = \int \left[p_X^{\mathcal{C}}(\boldsymbol{x}) - p_X^{\mathcal{G}}(\boldsymbol{x})\right]^2 \mathrm{d}\boldsymbol{x}, \tag{32a}$$

such that

$$\sum_{i=1}^{n_x} \mu_x^i = 1, \quad \mu_x^i \geq 0, \quad \boldsymbol{P}_x^i = (\boldsymbol{P}_x^i)^T > \boldsymbol{0}, \quad i = 1, 2, \ldots, n_x. \tag{32b}$$

Here $p_X^{\mathcal{G}}(\boldsymbol{x})$ is a GMM defined in (31) and $p_X^{\mathcal{C}}(\boldsymbol{x}) = \prod_{i=1}^{m} \frac{\delta_i/\pi}{x_i^2+\delta_i^2}$ is a zero median multivariate Cauchy density function with argument $\boldsymbol{x} = [x_1, x_2, \ldots, x_m]^T$ and scaling parameters $\delta_i > 0$, $i = 1, 2, \ldots, m$. The complex minimization problem (32) can be solved, e.g., numerically by standard constrained optimization tools.

Figure 1 illustrates an actual fitting of a standard scalar Cauchy pdf ($m = 1$ and $\delta_1 = 1$) with a GMM having different numbers of Gaussian components n_x. For $n_x = 3$, $n_x = 5$, and $n_x = 7$ the resulting ISD, computed numerically, is approximately equal to $J_f = 2.4 \times 10^{-3}$, $J_f = 9.79 \times 10^{-5}$, and $J_f = 6.01 \times 10^{-5}$, respectively.

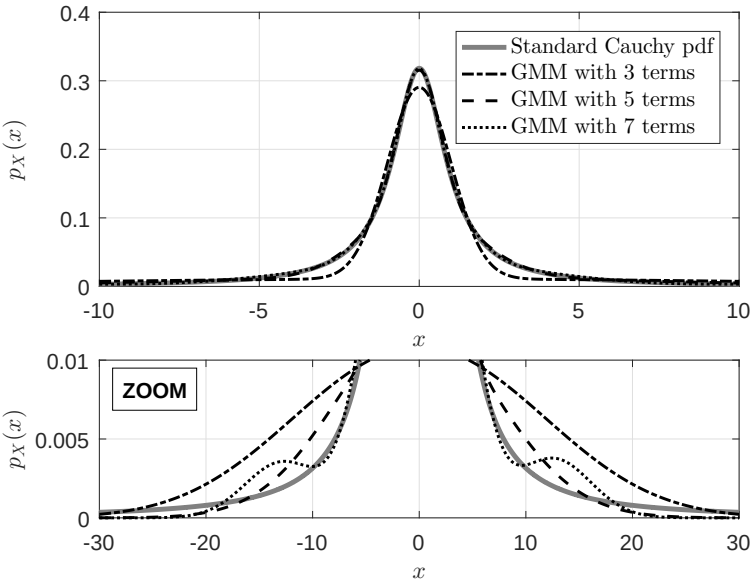

Figure 1. Fitting a Cauchy PDF with a GMM having different numbers of terms.

To proceed with the GSF algorithm, assume that the Cauchy densities given in (3), (4), and (5) are all fitted by a GMM in the same way as $p_X^{\mathcal{G}}$ was fitted to $p_X^{\mathcal{C}}$ in (32), i.e.,

$$p_W(w_k) \approx \sum_{i=1}^{n_w} \mu_w^i \mathcal{N}\left(w_k; \bar{x}_w^i, P_w^i\right), \tag{33a}$$

$$p_V(v_k) \approx \sum_{i=1}^{n_v} \mu_v^i \mathcal{N}\left(v_k; \bar{x}_v^i, P_v^i\right), \tag{33b}$$

$$p_{X_1}(\boldsymbol{x}_1) \approx \sum_{i=1}^{n_{1|0}} \mu_{1|0}^i \mathcal{N}\left(\boldsymbol{x}_1; \bar{\boldsymbol{x}}_{1|0}^i, \boldsymbol{P}_{1|0}^i\right). \tag{33c}$$

V.B.3. Time Propagation

Suppose that at time $k-1$, the posterior density $p(\boldsymbol{x}_{k-1}|\boldsymbol{z}_{1:k-1})$ is approximated by a weighted sum of $n_{k-1|k-1}$ Gaussian densities

$$p(\boldsymbol{x}_{k-1}|\boldsymbol{z}_{1:k-1}) \approx \sum_{i=1}^{n_{k-1|k-1}} \mu_{k-1|k-1}^{i} \mathcal{N}\left(\boldsymbol{x}_{k-1}; \bar{\boldsymbol{x}}_{k-1|k-1}^{i}, \boldsymbol{P}_{k-1|k-1}^{i}\right). \tag{34}$$

Then, the approximation of the a priori density $p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k-1})$ at time k is obtained in the GSF sense as

$$p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k-1}) \approx \sum_{i=1}^{n_{k-1|k-1}} \sum_{j=1}^{n_w} \tilde{\mu}_{k|k-1}^{ij} \mathcal{N}\left(\boldsymbol{x}_k; \bar{\boldsymbol{m}}_{k|k-1}^{ij}, \boldsymbol{M}_{k|k-1}^{ij}\right), \tag{35}$$

where $\bar{\boldsymbol{m}}_{k|k-1}^{ij}$ and $\boldsymbol{M}_{k|k-1}^{ij}$ are computed using Kalman-like equations, i.e., for all $i = 1, \ldots, n_{k-1|k-1}$ and $j = 1, \ldots, n_w$ we have

$$\bar{\boldsymbol{m}}_{k|k-1}^{ij} = \boldsymbol{\Phi}\bar{\boldsymbol{x}}_{k-1|k-1}^{i} + \boldsymbol{\Gamma}\bar{x}_w^j, \tag{36a}$$

$$\boldsymbol{M}_{k|k-1}^{ij} = \boldsymbol{\Phi}\boldsymbol{P}_{k-1|k-1}^{i}\boldsymbol{\Phi}^T + \boldsymbol{\Gamma}P_w^j\boldsymbol{\Gamma}^T. \tag{36b}$$

The weighting factors $\tilde{\mu}_{k|k-1}^{ij}$ are updated for all $i = 1, \ldots, n_{k-1|k-1}$ and $j = 1, \ldots, n_w$ as

$$\tilde{\mu}_{k|k-1}^{ij} = \mu_{k-1|k-1}^{i}\mu_w^j. \tag{37}$$

For notation convenience, the double summation in (35) can be restated as

$$p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k-1}) \approx \sum_{i=1}^{n_{k|k-1}} \mu_{k|k-1}^{i} \mathcal{N}\left(\boldsymbol{x}_k; \bar{\boldsymbol{x}}_{k|k-1}^{i}, \boldsymbol{P}_{k|k-1}^{i}\right), \tag{38}$$

where $n_{k|k-1} = (n_{k-1|k-1})(n_w)$, and $\mu_{k|k-1}^{i}$, $\bar{\boldsymbol{x}}_{k|k-1}^{i}$, and $\boldsymbol{P}_{k|k-1}^{i}$ are formed in an obvious fashion from $\tilde{\mu}_{k|k-1}^{ij}$, $\bar{\boldsymbol{m}}_{k|k-1}^{ij}$, and $\boldsymbol{M}_{k|k-1}^{ij}$, respectively.

V.B.4. Measurement Update

Suppose that at time k, the a priori density $p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k-1})$ is expressed as in (38)[d]. Then, using the measurement z_k, the a posterior density $p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k})$ at time k is approximated in the GSF sense as

$$p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k}) \approx \sum_{i=1}^{n_{k|k-1}} \sum_{j=1}^{n_v} \tilde{\mu}_{k|k}^{ij} \mathcal{N}\left(\boldsymbol{x}_k; \bar{\boldsymbol{m}}_{k|k}^{ij}, \boldsymbol{M}_{k|k}^{ij}\right), \tag{39}$$

where $\bar{\boldsymbol{m}}_{k|k}^{ij}$ and $\boldsymbol{M}_{k|k}^{ij}$ are computed for all $i = 1, \ldots, n_{k|k-1}$ and $j = 1, \ldots, n_v$ as

$$\hat{z}_k^{ij} = \boldsymbol{H}\bar{\boldsymbol{x}}_{k|k-1}^{i} + \bar{x}_v^j, \tag{40a}$$

$$S_k^{ij} = \boldsymbol{H}\boldsymbol{P}_{k|k-1}^{i}\boldsymbol{H}^T + P_v^j, \tag{40b}$$

$$\boldsymbol{K}_k^{ij} = \boldsymbol{P}_{k|k-1}^{i}\boldsymbol{H}^T(S_k^{ij})^{-1}, \tag{40c}$$

$$\bar{\boldsymbol{m}}_{k|k}^{ij} = \bar{\boldsymbol{x}}_{k|k-1}^{i} + \boldsymbol{K}_k^{ij}(z_k - \hat{z}_k^{ij}), \tag{40d}$$

$$\boldsymbol{M}_{k|k}^{ij} = \boldsymbol{P}_{k|k-1}^{i} - \boldsymbol{K}_k^{ij}S_k^{ij}(\boldsymbol{K}_k^{ij})^T. \tag{40e}$$

The weighting factors $\tilde{\mu}_{k|k}^{ij}$ are updated for all $i = 1, \ldots, n_{k|k-1}$ and $j = 1, \ldots, n_v$ using the following rule

$$\tilde{\mu}_{k|k}^{ij} = \frac{\mu_{k|k-1}^{i}\mu_v^j\mathcal{N}(z_k; \hat{z}_k^{ij}, S_k^{ij})}{\sum_{l=1}^{n_{k|k-1}}\sum_{m=1}^{n_v}\mu_{k|k-1}^{l}\mu_v^m\mathcal{N}(z_k; \hat{z}_k^{lm}, S_k^{lm})}. \tag{41}$$

[d]Note that at time $k = 1$, the a priori density corresponds to the GMM representation of the initial state density given in (33c), which has the same form as (38).

For convenience, one can rewrite (39) as:

$$p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k}) \approx \sum_{i=1}^{n_{k|k}} \mu_{k|k}^i \mathcal{N}\left(\boldsymbol{x}_k; \bar{\boldsymbol{x}}_{k|k}^i, \boldsymbol{P}_{k|k}^i\right), \tag{42}$$

where $n_{k|k} = (n_{k|k-1})(n_v)$ and $\mu_{k|k}^i$, $\bar{\boldsymbol{x}}_{k|k}^i$, and $\boldsymbol{P}_{k|k}^i$ are again formed from $\tilde{\mu}_{k|k}^{ij}$, $\bar{\boldsymbol{m}}_{k|k}^{ij}$, and $\boldsymbol{M}_{k|k}^{ij}$, respectively. The weighting factors $\mu_{k|k}^i$ satisfy $\mu_{k|k}^i \geq 0$ and $\sum_{i=1}^{n_{k|k}} \mu_{k|k}^i = 1$, thus generating a proper pdf stated in (42). Note that for $n_w = n_v = n_{1|0} = 1$, the above GSF equations reduce to the standard Kalman filter equations.

V.B.5. Conditional Mean and Estimation Error Variance

Using the posterior density $p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k})$ at time k, as given in (42), the conditional mean (11) and the estimation error covariance (13) can be approximated in the GSF sense as

$$\hat{\boldsymbol{x}}_k \approx \sum_{i=1}^{n_{k|k}} \mu_{k|k}^i \bar{\boldsymbol{x}}_{k|k}^i, \tag{43}$$

$$\boldsymbol{P}_k \approx \sum_{i=1}^{n_{k|k}} \mu_{k|k}^i \left(\boldsymbol{P}_{k|k}^i + (\bar{\boldsymbol{x}}_{k|k}^i - \hat{\boldsymbol{x}}_k)(\bar{\boldsymbol{x}}_{k|k}^i - \hat{\boldsymbol{x}}_k)^T\right). \tag{44}$$

The major disadvantage of the GSF algorithm is that the number of terms in the Gaussian sum increases exponentially in time. The number of terms in $p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k})$ at step k can be expressed explicitly as $n_{k|k} = n_{1|0} \times (n_w \times n_v)^k / n_w$. This is obviously a heavy computational burden for real-time implementation. It is normally addressed by the Gaussian Sum re-approximation discussed next.

V.B.6. Gaussian Sum Re-approximation

Several heuristic approaches have been proposed in the literature to avoid the exponential growth of terms $n_{k|k}$ in the Gaussian sum (42), see for instance [13, 21] and references therein. A seemingly tempting method of adopting Gaussian components with largest weights was found to be inefficient [22]. This is mainly due to the fact that even if the weight of a Gaussian term is very small at a certain point, it might become large at the next time step. This is often the case for the Cauchy distributed environment, when an outlier is very likely to occur. Ignoring such a component might have severe consequences.

In this paper, we suggest to re-approximate the posteriori density estimate (42) of the GSF by a new reduced-order GMM with pre-fixed number of Gaussian components. This reduction in terms is motivated by the observation that a relatively small number of weighted Gaussian components can approximate a large class of densities [23], as well as by our aim to confine the computational time of the GSF in order to compare its performance to the approximate ISCE and PF, both having a bounded computational burden.

For ease of notation, assume that at a given time step the measurement updated Gaussian sum density (42) is denoted by $p_a(\boldsymbol{x})$ and has n_a terms, i.e.,

$$p_a(\boldsymbol{x}) = \sum_{i=1}^{n_a} \mu_a^i \mathcal{N}\left(\boldsymbol{x}; \bar{\boldsymbol{x}}_a^i, \boldsymbol{P}_a^i\right). \tag{45}$$

After evaluating (43) and (44) using $p_a(\boldsymbol{x})$, the objective is to approximate $p_a(\boldsymbol{x})$ by another Gaussian sum density,

$$p_b(\boldsymbol{x}) = \sum_{i=1}^{n_b} \mu_b^i \mathcal{N}\left(\boldsymbol{x}; \bar{\boldsymbol{x}}_b^i, \boldsymbol{P}_b^i\right), \tag{46}$$

which has a constant pre-fixed number of terms n_b. Obviously, if $n_a \leq n_b$, then there is no need for re-approximation and $p_b(\boldsymbol{x}) = p_a(\boldsymbol{x})$ is considered. If $n_a > n_b$, then the task of the suggested re-approximation scheme is to determine $\{\mu_b^i,\ \bar{\boldsymbol{x}}_b^i,\ \boldsymbol{P}_b^i\}_{i=1}^{n_b}$ such that the mean and covariance of the new Gaussian mixture $p_b(\boldsymbol{x}_k)$ match exactly those of $p_a(\boldsymbol{x}_k)$, while also minimizing the ISD between $p_a(\boldsymbol{x}_k)$ and $p_b(\boldsymbol{x}_k)$. Given $n_b \geq 1$, this task can be formulated as a constrained optimization problem

$$\underset{\{\mu_b^i, \bar{\boldsymbol{x}}_b^i, \boldsymbol{P}_b^i\}_{i=1}^{n_b}}{\operatorname{argmin}} \quad J = \int \left[p_a(\boldsymbol{x}) - p_b(\boldsymbol{x})\right]^2 \mathrm{d}\boldsymbol{x}, \tag{47a}$$

such that

$$\sum_{i=1}^{n_b} \mu_b^i = 1, \quad \mu_b^i \geq 0, \quad \boldsymbol{P}_b^i = (\boldsymbol{P}_b^i)^T > \boldsymbol{0}, \quad i = 1, 2, \ldots, n_b, \tag{47b}$$

$$\sum_{i=1}^{n_a} \mu_a^i \bar{\boldsymbol{x}}_a^i = \sum_{i=1}^{n_b} \mu_b^i \bar{\boldsymbol{x}}_b^i, \qquad \sum_{i=1}^{n_a} \mu_a^i \left(\boldsymbol{P}_a^i + \bar{\boldsymbol{x}}_a^i (\bar{\boldsymbol{x}}_a^i)^T \right) = \sum_{i=1}^{n_b} \mu_b^i \left(\boldsymbol{P}_b^i + \bar{\boldsymbol{x}}_b^i (\bar{\boldsymbol{x}}_b^i)^T \right). \tag{47c}$$

The cost function (47a) can be expanded and rewritten as

$$J = \int p_a^2(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} - 2 \int p_a(\boldsymbol{x}) p_b(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} + \int p_b^2(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \triangleq J_{aa} - 2J_{ab} + J_{bb}, \tag{48a}$$

where the particular integrals J_{aa}, J_{ab}, and J_{bb} were solved, in closed-form, by Williams and Maybeck [24], to yield

$$J_{aa} = \sum_{i=1}^{n_a} \sum_{j=1}^{n_a} \mu_a^i \mu_a^j \mathcal{N}\left(\bar{\boldsymbol{x}}_a^i; \bar{\boldsymbol{x}}_a^j, \boldsymbol{P}_a^i + \boldsymbol{P}_a^j \right), \tag{48b}$$

$$J_{ab} = \sum_{i=1}^{n_a} \sum_{j=1}^{n_b} \mu_a^i \mu_b^j \mathcal{N}\left(\bar{\boldsymbol{x}}_a^i; \bar{\boldsymbol{x}}_b^j, \boldsymbol{P}_a^i + \boldsymbol{P}_b^j \right), \tag{48c}$$

$$J_{bb} = \sum_{i=1}^{n_b} \sum_{j=1}^{n_b} \mu_b^i \mu_b^j \mathcal{N}\left(\bar{\boldsymbol{x}}_b^i; \bar{\boldsymbol{x}}_b^j, \boldsymbol{P}_b^i + \boldsymbol{P}_b^j \right). \tag{48d}$$

A small value of J indicates, in the ISD sense, that $p_b(\boldsymbol{x})$ is a good approximation of $p_a(\boldsymbol{x})$. However, there is no guarantee that the re-approximated density also preserves the higher order moments of the original one.

Note that solving the above constrained minimization problem generally involves computationally costly nonlinear optimization with respect to $n_b \times (n^2/2 + 3n/2 + 1)$ independent variables, where n is the dimension of the system state vector $\boldsymbol{x}$. Obviously, the computational burden of the GSF will also be affected by the numerical procedure used to solve the above constrained optimization problem. In this paper, we assume that the computational time of the GSF algorithm is dictated solely by n and n_b. This can be achieved by using numerical solvers with fixed number of iterations.

VI. Numerical Study

In this section, the performance of the PF and GSF is analyzed and numerically compared to the scalar and two-state ISCE. All simulations were performed in MATLAB (R2016a) on a desktop computer with an 8-core Intel Xeon processor at 2.90 GHz and 128 GB of RAM.

The PF was implemented using the systematic resampling technique [19], with a threshold parameter $n_p^t = (2/3)n_p$. The GSF-related constrained optimization problems, defined in (32) and (47), were solved with the *interior-point* algorithm using *fmincon* function from the Optimization Toolbox [25] of MATLAB.

Note that the fitting problem defined in (32) is performed off-line (filter design phase) and it does not affect the on-line computation load (testing phase) of the GSF. On the other hand, the GMM re-approximation procedure defined in (47) has to be solved almost after each measurement update. For this task, the *fmincon* function is constrained to a maximum of 100 iterations. This leads to a GSF implementation which has a limited computational burden.

VI.A. Scalar Case Example

For the scalar case example, the following system parameters were considered: $\Phi = 0.75$, $\Gamma = 1$, $H = 2$, $\beta = 0.1$, $\gamma = 0.2$, and $\alpha = 0.5$.

VI.A.1. Sample Run

Before turning to a statistical MC evaluation, we compare the accuracy of the PF and GSF on a sample run scenario driven by the noise sequences depicted in Fig. 2. For clarity of the presentation, we omit the

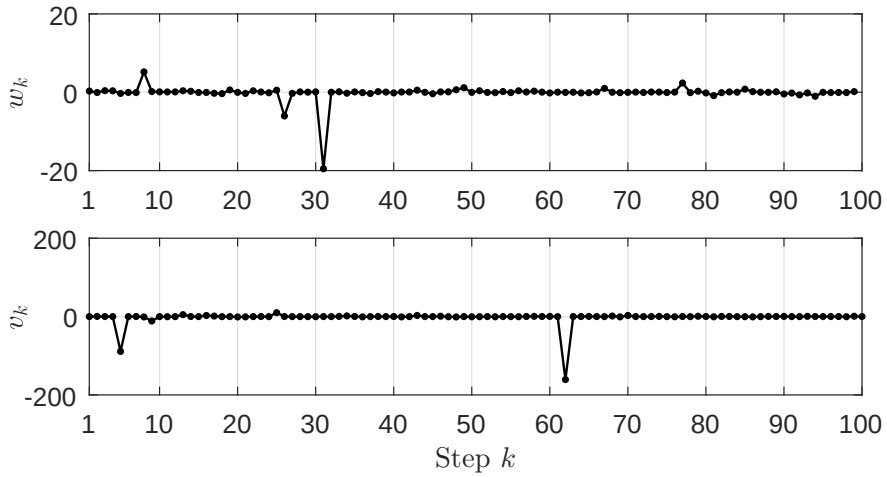

Figure 2. Cauchy distributed process and measurement noise sample sequences.

approximate implementation of the scalar ISCE from the sample run comparison as the results for a sliding window of $n_s \geq 10$ are indistinguishable from the optimal ISCE.

Based on the discussion presented in Section V.B.2, the process noise, measurement noise, and initial state pdf of the GSF was fitted to the Cauchy pdf with a weighted Gaussian sum of $n_w = 7$, $n_v = 7$, and $n_{1|0} = 9$ components, respectively. Two different number of particles (n_p) and Gaussian components kept at each step (n_b) are considered for the PF and for the GSF, respectively, to demonstrate their effect on the accuracy of those approximations.

We first assess the performance of the PF and GSF through their approximation of the true conditional pdf $p(x_k|z_{1:k})$ obtained from the optimal ISCE as shown in Fig. 3. The conditional pdf at time step 8 is considered as it yields a refined bimodal distribution. Fig. 3 clearly demonstrates that only the PF with $100,000$ particles approximates reasonably well the true conditional pdf. However, the computational burden when compared to the ISCE is quite high. The average computation time of the PF with 50 particles is 4.5 times and of the PF with $100,000$ particles is $7,000$ times higher than the average computation time of the optimal ISCE while processing 100 data steps, and when carried out on the same computer.

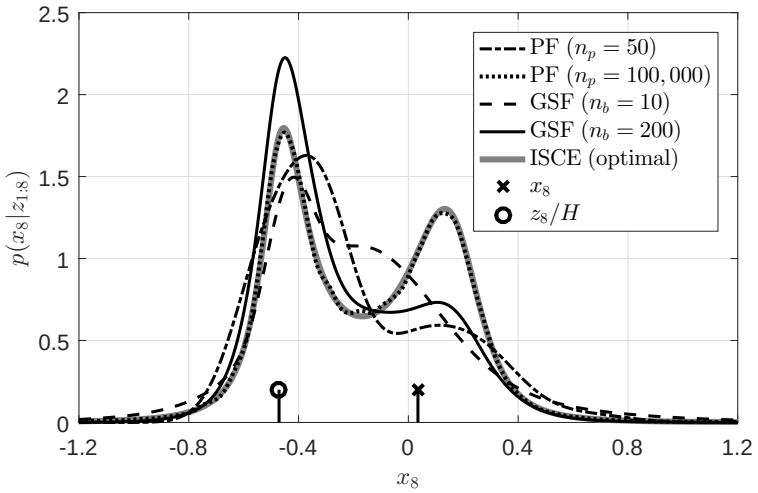

Figure 3. Comparison of PF and GSF approximations of the true density at k=8.

On the other hand, the GSF-based approximation of the conditional pdf is very poor even when 200 Gaussian components[e] are kept at each step, as shown in Fig. 3. The bimodal shape of the true conditional pdf is barely preserved. The computational burden of the GSF with 10 components is approximately $1,000$

[e]Note that without engaging the re-approximation procedure of (47), the number of Gaussian components in (42) at time $k = 8$ would reach a number greater than 4×10^{13}.

times and with 200 components is approximately 25,000 times higher than the computational burden of the ISCE, respectively.

Next, estimation results for a 100 step sample run are presented. The system is again driven by the noise sequences depicted in Fig. 2. The upper subplot in Fig. 4 shows the difference between the exact minimum variance state estimate ($\hat{x}_k^*$) and its approximation ($\hat{x}_k$) computed by the PF or the GSF. The bottom subplot presents the difference between the exact standard deviation of the estimation error (σ_k^*) and its approximation (σ_k) computed by the PF or GSF. Note that the exact values of $\hat{x}_k^*$ and σ_k^* are computed by the optimal ISCE, see (15) and (16), respectively. The approximate implementation of the scalar ISCE is omitted here as the results are indistinguishable from the optimal ISCE.

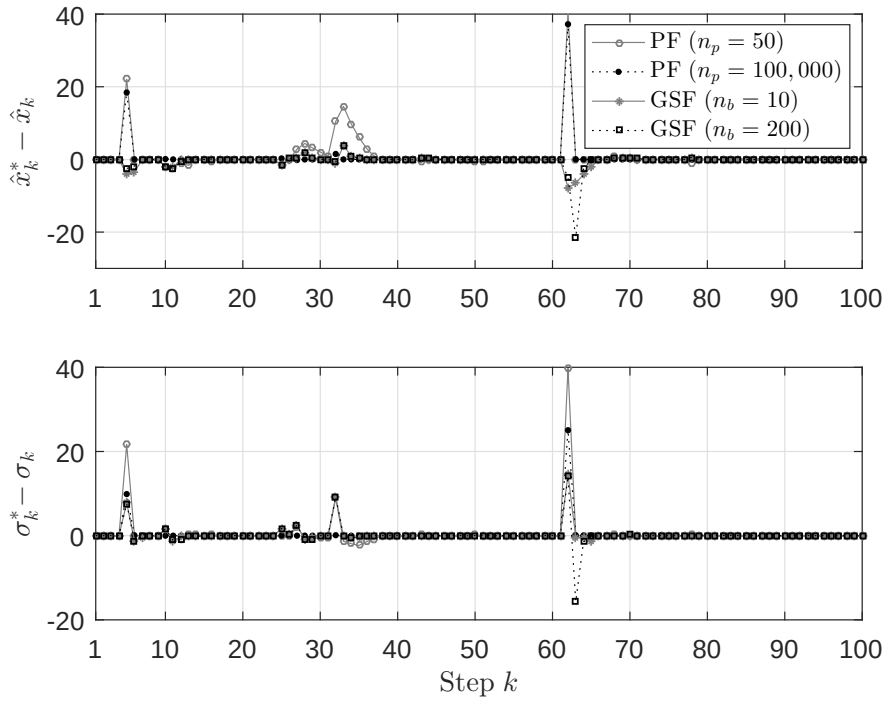

Figure 4. PF and GSF approximation error statistics compared to the optimal ISCE values.

Fig. 4 shows that the PF with both 50 and 100,000 particles disregards the measurement outlier at steps 5 and 62. On the other hand, process noise outliers of steps 26 and 31 cause a slight divergence of the PF with 50 particles, but vanishing after few steps. Such behavior is observed when the number of particles is insufficient to capture properly the heavy-tail characteristics of the Cauchy noise environment. At the cost of increased computational burden, and except the two measurement outliers discussed earlier, the performance of the PF with 100,000 particles is comparable to that of the ISCE.

Similar conclusions can be drawn when examining the performance of the GSF. In this case, the measurement noise outliers at steps 5 and 62 yield to a slight divergence, especially at step 62.

VI.A.2. Monte Carlo Analysis

The presented sample run results suggest that, at the cost of a significantly higher computational burden, both the PF and the GSF perform comparably to the ISCE when considering the scalar-state problem. Now we will consider a Monte Carlo simulation-based evaluation of the PF and GSF performance. Additionally, to allow a fair comparison of the PF and GSF with the ISCE, the number of particles n_p of the PF and the number of components (n_b, n_w, n_v, $n_{1|0}$) of the GSF are selected such that the average computational burden of the PF and GSF is similar to that of the approximate ISCE implementation with a window of 20 steps, i.e., $n_s = 20$. Consequently, n_p was set to 12 while the parameters of the GSF are set to be as minimalist as possible, i.e., $n_b = 3$, $n_w = 3$, $n_v = 3$, and $n_{1|0} = 3$. This resulted in a computational burden of the GSF to be approximately 30 times larger that the ISCE and PF.

Conventional evaluation of a MC-based ensemble mean and variance of the estimation error cannot be performed for the studied problem as both the system state and the measurements are Cauchy distributed.

As a consequence, the estimation errors (no matter whether computed by ISCE, PF, or GSF) are also heavy-tailed, leading to infinite variance when computed via the conventional MC averaging method. Therefore, in this paper, we suggest to evaluate the estimation performance using the log of the geometric mean square, i.e.,

$$\tilde{\sigma}_k^2 \triangleq \frac{1}{n_{mc}} \sum_{i=1}^{n_{mc}} \log\left(\left(x_k^{(i)} - \hat{x}_k^{(i)} \right)^2 \right), \tag{49}$$

where i indicates the ith MC realization, n_{mc} is the total number of MC runs. Since the log is monotonic, but also suppresses the large deviations caused by the Cauchy impulsive uncertainty, $\tilde{\sigma}_k^2$ appears to allow an ensemble measure of the heavy-tailed mean-square estimation error deviations.

Figure 5 presents the results for the suggested measure $\tilde{\sigma}_k^2$ based on a set of $n_{mc} = 10,000$ MC runs. For consistency, we also depict the obtained results for the optimal ISCE. As the PF implementation is nondeterministic, for each MC realization a set of 100 *inner MC runs* were considered to obtain an averaged state estimate $\hat{x}_k^{(i)}$ for the PF.

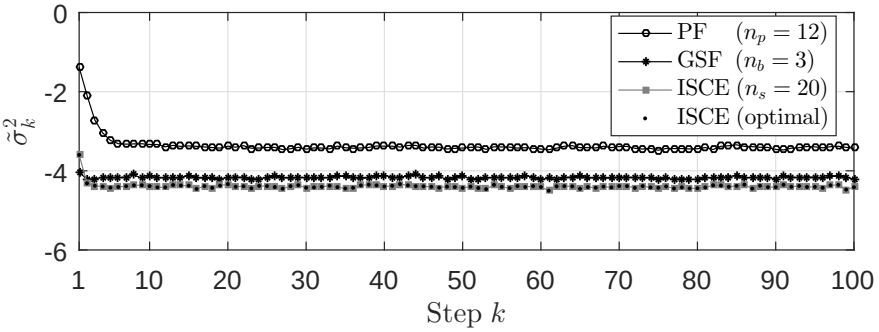

Figure 5. Log of the geometric mean square of the scalar estimation error.

It can be observed from Fig. 5 that the performance of the PF degrades severely when constraining the computational load to be comparable to the ISCE with $n_s = 20$. Despite the fact that the computational burden of the GSF is much higher, its performance is inferior to the ISCE.

VI.B. Two-state Case Example - Monte Carlo Analysis

For the two-state case example, the following system parameters were chosen: $\boldsymbol{H} = [\,1 \;\; 2\,]$,

$$\boldsymbol{\Phi} = \begin{bmatrix} 0.9 & 0.1 \\ -0.2 & 1.0 \end{bmatrix}, \quad \boldsymbol{\Gamma} = \begin{bmatrix} 1.0 \\ 0.3 \end{bmatrix}, \quad \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} = \begin{bmatrix} 0.5 \\ 0.3 \end{bmatrix}.$$

The process noise, $\beta = 0.1$, and measurement noise, $\gamma = 0.2$, parameters are the same as in the scalar case example. Sample run results comparing the performance of the PF and GSF with the two-state ISCE, implemented both with a six-step and a eight-step sliding window approximation, are presented in [15]. In the current study, the performance of the six-step window implementation, more suitable for real time applications, will be shown to be statistically comparable to the eight-step approximation. Hence the six-window ISCE will serve as the baseline for selecting the number of particles n_p of the PF and the number of components (n_b, n_w, n_v, $n_{1|0}$) of the GSF. Consequently, n_p was set to $4,500$ while $n_b = 3$, $n_w = 3$, $n_v = 3$, and $n_{1|0} = 9$.

Figure 6 shows the obtained results based on a set of 1,000 MC simulations. Again, 100 inner MC runs were considered for the PF. Additionally to the six-window ISCE approximation, we depict also the eight-window case to demonstrate their consistency. Figure 6 clearly demonstrates that $4,500$ particles, which result in a comparable computing time to that of the six-step ISCE approximation, are not enough to properly estimate the system states. Similar conclusion can be drawn when examining the performance of the GSF depicted in Fig. 6. In this case, to have a comparable computing time, the GSF is constrained to keep only 3 Gaussian components. This figure clearly shows that the GSF performs very poorly, worse than the respective PF approximation. It demonstrates that the heavy-tail characteristics of the Cauchy noise environment cannot be captured well enough by a limited number of Gaussian pdf's.

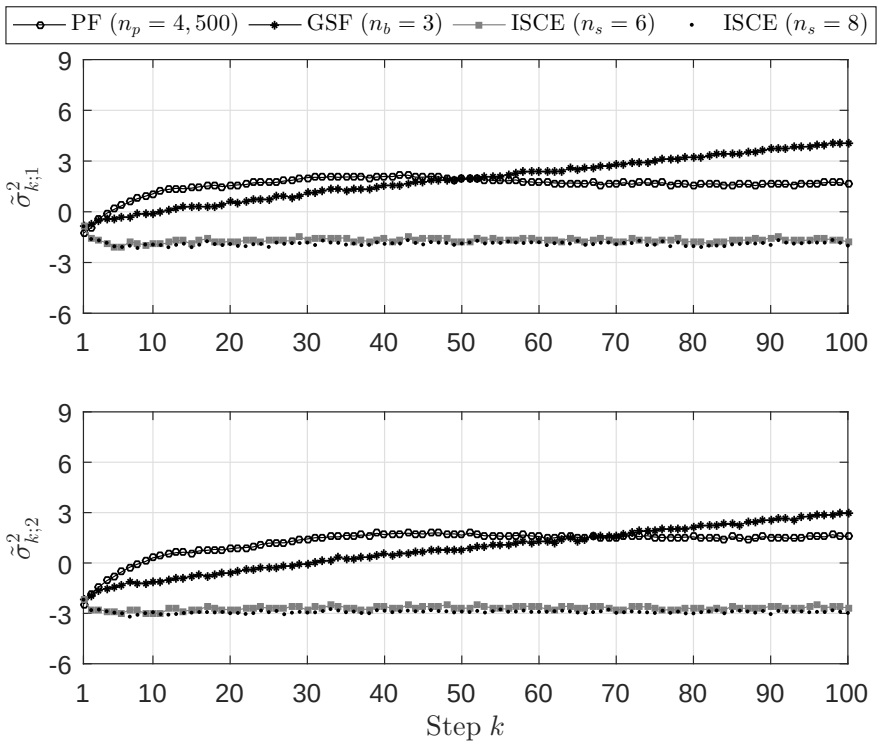

Figure 6. Log of the geometric mean square of the two-state estimation error.

VII. Conclusion

The estimation performance of two popular approximate filtering algorithms have been numerically compared with the approximate scalar and two-state ISCE, for a linear discrete-time dynamic system with additive Cauchy measurement and process noises. Despite the fact that both the PF and GSF were designed based on the same a priori information as the ISCE, sample run results shows that in the scalar case, the PF tends to converge to the optimal solution only when using a very large number of particles, while the GSF demonstrated modest convergence and a large discrepancy when the pdf approximation is of concern. Monte Carlo simulation results for the scalar and two-state case revealed that both the PF and GSF perform poorly and even diverge for a computation time consistent with that of the approximate ISCE. Hence the PF and GSF do not provide a practical alternative to the approximation that is based on the optimal solution. Consequently, for real-time implementation of filtering problems in the impulsive noise environment represented here as heavy-tailed Cauchy noises, the approximate scalar and two-state ISCE with a bounded computational burden is clearly the superior solution.

Acknowledgments

This work was supported by the National Science Foundation (NSF) under Grant No. 1607502, the United StatesŰIsrael Binational Science Foundation (BSF) under Grant No. 2012122, and the joint NSF-BSF ECCS program under Grant No. 2015702.

References

[1] Taleb, N. N., *The Black Swan: The Impact of the Highly Improbable*, Random House, New York, 2007.

[2] Carpenter, J. R. and Mashiku, A. K., "Cauchy Drag Estimation For Low Earth Orbiters," in "AAS/AIAA Space Flight Mechanics Meeting," Williamsburg, VA; United States, 2015, pp. 2731–2746.

[3] Speyer, J. L. and Chung, W. H., *Stochastic Processes, Estimation, and Control*, SIAM, Philadelphia,

2008.

[4] Kalman, R. E., "A new approach to linear filtering and prediction problems," *Journal of basic Engineering*, Vol. 82, No. 1, 1960, pp. 35–45, doi:10.1115/1.3662552.

[5] Schick, I. C. and Mitter, S. K., "Robust recursive estimation in the presence of heavy-tailed observation noise," *The Annals of Statistics*, Vol. 22, No. 2, 1994, pp. 1045–1080, doi:10.1214/aos/1176325511.

[6] Idan, M. and Speyer, J. L., "Cauchy Estimation for Linear Scalar Systems," *IEEE Transactions on Automatic Control*, Vol. 55, No. 6, 2010, pp. 1329–1342, doi:10.1109/TAC.2010.2042009.

[7] Idan, M. and Speyer, J. L., "State Estimation for Linear Scalar Dynamic Systems with Additive Cauchy Noises: Characteristic Function Approach," *SIAM Journal on Control and Optimization*, Vol. 50, No. 4, 2012, pp. 1971–1994, doi:10.1137/110831362.

[8] Idan, M. and Speyer, J. L., "Multivariate Cauchy Estimator with Scalar Measurement and Process Noises," *SIAM Journal on Control and Optimization*, Vol. 52, No. 2, 2014, pp. 1108–1141, doi:10.1137/120891897.

[9] Fernandez, J. H., *Methods for Estimation and Control of Linear Systems Driven by Cauchy Noises*, Ph.D. thesis, UCLA: Mechanical Engineering 0330, 2013.

[10] Fernandez, J. H., Speyer, J. L., and Idan, M., "Stochastic Estimation for Two-State Linear Dynamic Systems with Additive Cauchy Noises," *IEEE Transactions on Automatic Control*, Vol. 60, No. 12, 2015, pp. 3367–3372, doi:10.1109/TAC.2015.2422478.

[11] Gordon, N. J., Salmond, D. J., and Smith, A. F. M., "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," in "IEE Proceedings F - Radar and Signal Processing," IET, Vol. 140, 1993, pp. 107–113, doi:10.1049/ip-f-2.1993.0015.

[12] Doucet, A., Godsill, S., and Andrieu, C., "On sequential Monte Carlo sampling methods for Bayesian filtering," *Statistics and Computing*, Vol. 10, No. 3, 2000, pp. 197–208, doi:10.1023/A:1008935410038.

[13] Sorenson, H. W. and Alspach, D. L., "Recursive Bayesian Estimation Using Gaussian Sums," *Automatica*, Vol. 7, No. 4, 1971, pp. 465–479, doi:10.1016/0005-1098(71)90097-5.

[14] Alspach, D. L. and Sorenson, H. W., "Nonlinear Bayesian estimation using Gaussian sum approximations," *IEEE Transactions on Automatic Control*, Vol. 17, No. 4, 1972, pp. 439–448, doi:10.1109/TAC.1972.1100034.

[15] Fonod, R., Idan, M., and Speyer, J. L., "State Estimation for Linear Systems with Additive Cauchy Noises: Optimal and Suboptimal Approaches," in "Proceedings of European Control Conference," IEEE, Piscataway, NJ, 2016, pp. 1434–1439, doi:10.1109/ECC.2016.7810491.

[16] Doucet, A., de Freitas, N., and Gordon, N., *Sequential Monte-Carlo Methods in Practice*, Springer-Verlag, New York, 2001.

[17] Arulampalam, M. S., Maskell, S., Gordon, N., and Clapp, T., "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, Vol. 50, No. 2, 2002, pp. 174–188, doi:10.1109/78.978374.

[18] Li, T., Bolic, M., and Djuric, P. M., "Resampling Methods for Particle Filtering: Classification, implementation, and strategies," *IEEE Signal Processing Magazine*, Vol. 32, No. 3, 2015, pp. 70–86, doi:10.1109/MSP.2014.2330626.

[19] Kitagawa, G., "Monte Carlo filter and smoother for non-Gaussian nonlinear state space models," *Journal of computational and graphical statistics*, Vol. 5, No. 1, 1996, pp. 1–25, doi:10.2307/1390750.

[20] Achieser, N. I., *Theory of Approximation*, Dover Publications, New York, 1992. Chap. 6.

[21] Psiaki, M. L., Schoenberg, J. R., and Miller, I. T., "Gaussian Sum Reapproximation for Use in a Nonlinear Filter," *Journal of Guidance, Control, and Dynamics*, Vol. 38, No. 2, 2015, pp. 292–303, doi:10.2514/1.G000541.

[22] Kitagawa, G., "The two-filter formula for smoothing and an implementation of the Gaussian-sum smoother," *Annals of the Institute of Statistical Mathematics*, Vol. 46, No. 4, 1994, pp. 605–623, doi:10.1007/BF00773470.

[23] Kitagawa, G., "Non-Gaussian seasonal adjustment," *Computers & Mathematics with Applications*, Vol. 18, No. 6, 1989, pp. 503–514, doi:10.1016/0898-1221(89)90103-X.

[24] Williams, J. L. and Maybeck, P. S., "Cost-function-based Gaussian mixture reduction for target tracking," in "Proceedings of the sixth international conference of information fusion," Vol. 2, 2003, pp. 1047–1054, doi:10.1109/ICIF.2003.177354.

[25] Venkataraman, P., *Applied optimization with MATLAB programming*, John Wiley & Sons, New York, 2002.