

Article

A Two-Filter Approach for State Estimation Utilizing Quantized Output Data

Angel L. Cedeño ^{1,2,*} , Ricardo Albornoz ^{1,2} , Rodrigo Carvajal ³ , Boris I. Godoy ⁴  and Juan C. Agüero ^{1,2} 

¹ Departamento Electrónica, Universidad Técnica Federico Santa María (UTFSM), Av. España 1680, Valparaíso 2390123, Chile; ricardo.albornoz.13@sansano.usm.cl (R.A.); juan.aguero@usm.cl (J.C.A.)

² Advanced Center for Electrical and Electronic Engineering, AC3E, Gral. Bari 699, Valparaíso 2390136, Chile

³ Escuela de Ingeniería Eléctrica, Pontificia Universidad Católica de Valparaíso, Av. Brasil 2147, Valparaíso 2374631, Chile; rodrigo.carvajal@pucv.cl

⁴ Department of Mechanical Engineering, Boston University, Boston, MA 02215, USA; bgodoy@bu.edu

* Correspondence: angel.cedeno@sansano.usm.cl

Abstract: Filtering and smoothing algorithms are key tools to develop decision-making strategies and parameter identification techniques in different areas of research, such as economics, financial data analysis, communications, and control systems. These algorithms are used to obtain an estimation of the system state based on the sequentially available noisy measurements of the system output. In a real-world system, the noisy measurements can suffer a significant loss of information due to (among others): (i) a reduced resolution of cost-effective sensors typically used in practice or (ii) a digitalization process for storing or transmitting the measurements through a communication channel using a minimum amount of resources. Thus, obtaining suitable state estimates in this context is essential. In this paper, Gaussian sum filtering and smoothing algorithms are developed in order to deal with noisy measurements that are also subject to quantization. In this approach, the probability mass function of the quantized output given the state is characterized by an integral equation. This integral was approximated by using a Gauss–Legendre quadrature; hence, a model with a Gaussian mixture structure was obtained. This model was used to develop filtering and smoothing algorithms. The benefits of this proposal, in terms of accuracy of the estimation and computational cost, are illustrated via numerical simulations.

Keywords: state estimation; quantized data; Gaussian sum filtering; Gaussian sum smoothing; Gauss–Legendre quadrature



Citation: Cedeño, A.L.; Albornoz, R.; Carvajal, R.; Godoy, B.I.; Agüero, J.C. A Two-Filter Approach for State Estimation Utilizing Quantized Output Data. *Sensors* **2021**, *21*, 7675. <https://doi.org/10.3390/s21227675>

Academic Editor: Andrey V. Savkin

Received: 27 September 2021

Accepted: 16 November 2021

Published: 18 November 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

It is well known that discrete-time dynamical systems can be described as first-order difference equations relating internal variables called states [1]. State estimation is a scientific discipline that studies methodologies and algorithms for estimating the state of dynamical systems from input–output measurements [2,3]. There are a variety of applications that use state estimation, such as control [4–7], parameter identification [8–10], power systems [11,12], fault detection [13–17], prognosis [18,19], cyber–physical systems [20], hydrologic and geophysical data assimilation [21,22], maritime tracking [23], consensus-based state estimation using wireless sensor networks [24–26], navigation systems [27], and transportation and highway traffic management [28–30], to mention a few. Depending on the measurements that are used, two algorithms of state estimation can be distinguished: filtering and smoothing. Filtering algorithms estimate the current state using measurements up to the current instant, and smoothing algorithms estimate the state at some time in the past using measurement up to the current instant [23,31].

In general, the experimental noisy data can suffer a significant loss of information introduced by low-resolution and cost-effective sensors [32] in the digitalization process [33].

Typically, the digitalization process encompasses a process known as *quantization*. Quantization is a nonlinear map that partitions the whole signal space and represents all of the values in each subspace by a single one [32]. In spite of the loss of information, the benefits of quantization have led to a number of applications in which quantized measurements arise. This occurs due to fundamental limitations on measuring equipment and bandwidth resources [34], digital and analog converters [35], and experimental designs where it is necessary to quantize the data in order to store it or minimize communication resource utilization [36]. In particular, estimation problems utilizing quantized measurements arise in networked control over limited-/finite-capacity communication channels, where usually, encoder–decoder state estimation schemes are used [37–40]. In addition, in order to deal with uncertain dynamic systems, robust estimation algorithms have also been developed; see, e.g., [37,38,41].

Currently, state estimation from quantized data has gained significant attention in a growing number of applications such as fault detection [42,43], networked control [42,44,45], and system identification [35,46–48]. For instance, in [49,50], Kalman-based state estimation algorithms were developed using multiple sensors for distributed systems. In [24], a Kalman smoothing algorithm was developed for any-time minimum-mean-squared error optimal-consensus-based state estimation using wireless sensor networks. In [27], an online smoothing algorithm was developed to estimate the positional and orientation parameters of integrated navigation systems utilizing a low-cost microelectromechanical system inertial sensor in near-real time.

In Figure 1, a usual representation of a process is shown, which is defined by the interconnection of three blocks: (1) an actuator, (2) a process, and (3) a sensor. This representation includes a link that can be a communication channel and a base station. The actuator input usually comes from a control system, and the output of the process is measured with noise by a sensor. The sensor introduces quantization to the noisy measurements. This representation has been used for state estimation and control in a microgrid with multiple distributed energy resources, where a dequantizer is used to reconstruct the received signal and then perform the standard Kalman filter [51]. In [42], a fault isolation filter for a discrete-time networked control system with multiple faults was developed using the Kalman filter, where the sensor measurements were transmitted only when the output signal was greater than a threshold. The authors in [52] dealt with a similar structure to the one in Figure 1 to estimate the vehicle sideslip angle of an in-vehicle networked system with sensor failure, dynamic quantization, and data dropouts. For more examples, see, e.g., [15,53].

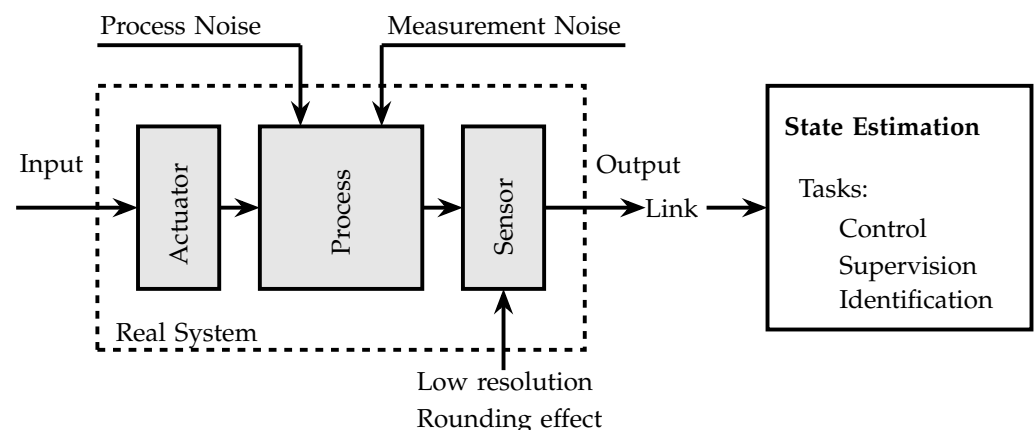


Figure 1. Diagram of a real dynamic system with quantized data.

For linear systems and Gaussian noises (without quantization), the optimal estimator of the system state is the celebrated Kalman filter [1] and smoother [22]. However, in the most general case, i.e., nonlinear systems and non-Gaussian noises, it is not possible to obtain an optimal estimator because the computation of some integrals in the filtering

and smoothing equations is difficult or the integrals are just intractable. As mentioned above, the quantization process is a nonlinear map that results in a significant loss of information on the system dynamics, which produces a biased state estimation and incorrect characterization of the filtering and smoothing probability density functions (PDFs). In this context, several suboptimal filtering and smoothing algorithms for state-space systems with quantized data have been developed, for instance, standard- [49,54], unscented- [55], and extended- [36] Kalman filters for quantized data, in which some structural elements of the state-space models and the quantizer are exploited. Sequential Monte Carlo methods have been also used for filtering and smoothing with quantized data, where complex integrals are approximated by a set of weighted samples called particles [31], which define (approximately) a desired PDF. However, the most common difficulty in these methods is dealing with the quantizer model. Some approaches have been proposed for this purpose such as gradient-based approximation of the quantizer [56] or modeling the quantizer as uniformly distributed additive noise [32,57]. In [58], an approximation of the integral in the non-Gaussian probability mass function (PMF) of the quantized output given the state was proposed by using Gaussian quadrature rules in order to deal with binary data. This approximation naturally yields an explicit Gaussian mixture model (GMM) form for the PMF of the quantized output. In this paper, the approximation of this integral was extended by considering a more general (finite- and infinite-level) quantizer, and it was used to develop Gaussian sum filtering and smoothing algorithms.

Main Contributions

The main contributions of this work are:

1. Developing an explicit model (of the GMM form) for the PMF of the quantized output considering a finite- and infinite-level quantizer, to solve in closed-form filtering and smoothing recursions in a Bayesian framework;
2. Designing Gaussian sum filtering and smoothing algorithms to deal with quantized data, providing closed expressions for the state estimates and for the filtering and smoothing PDFs.

The filtering algorithm for quantized data presented in this paper includes, as a particular case, the filtering algorithm presented in [58], where only the case of binary data was considered. Additionally, the smoothing algorithm presented here, based on the approximation of the PMF of the quantized output given the state, is completely novel. The remainder of the paper is as follows: In Section 2, the problem of interest is defined. In Section 3, the characterization and approximation of the PMF of the quantized data given the state are presented, and using this explicit model, the Gaussian sum filtering and smoothing algorithms are developed. In Section 4, some examples are presented to show the benefits of this proposal in terms of accuracy and computational cost. Finally, in Section 5, conclusions are presented.

2. Statement of the Problem

2.1. System Model

Consider the state-space model for a discrete-time linear-time-invariant system with quantized output (see Figure 2):

$$x_{t+1} = Ax_t + Bu_t + w_t, \quad (1)$$

$$z_t = Cx_t + Du_t + v_t, \quad (2)$$

$$y_t = q\{z_t\}, \quad (3)$$

where $x_t \in \mathbb{R}^n$ is the state vector, $z_t \in \mathbb{R}$ is the nonquantized output, $y_t \in \mathbb{R}$ is the quantized output, and $u_t \in \mathbb{R}^m$ is the input of the system. The matrices are $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, $C \in \mathbb{R}^{1 \times n}$, and $D \in \mathbb{R}^{1 \times m}$. The nonlinear map $q\{\cdot\}$ is the quantizer. The state noise $w_t \in \mathbb{R}^n$ and the output noise $v_t \in \mathbb{R}$ are zero-mean white Gaussian noises with covariance matrix

Q and R , respectively. The system in (1)–(2) can be described using the state transition and conditional nonquantized output probability distributions:

$$p(x_1) = \mathcal{N}_{x_1}(\mu_1, P_1), \quad (4)$$

$$p(x_{t+1}|x_t) = \mathcal{N}_{x_{t+1}}(Ax_t + Bu_t, Q), \quad (5)$$

$$p(z_t|x_t) = \mathcal{N}_{z_t}(Cx_t + Du_t, R), \quad (6)$$

where $\mathcal{N}_x(\mu, P)$ represents a PDF corresponding to a Gaussian distribution with mean μ and the covariance matrix P of the variable x . The initial condition x_1 , the model noise w_t , and the measurement noise v_t are statistically independent random variables.

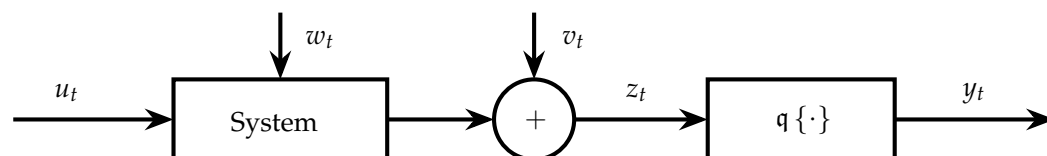


Figure 2. State-space model with quantized output.

2.2. Quantizer Model

Let the quantizer $q\{\cdot\} : \mathbb{R} \rightarrow \mathcal{V}$ be a map from the real line defined by the set of intervals $\{\mathcal{J}_i \subset \mathbb{R} : i \in \mathcal{I}\}$ to the finite or countable infinite set $\mathcal{V} = \{\beta_i \in \mathbb{R} : i \in \mathcal{I}\}$ [59], where \mathcal{I} is a set of indices defined by the quantizer type. In this work, two quantizers were considered. The first an infinite-level quantizer, in which the output of the quantizer has infinite (countable) levels of quantization corresponding to the indices' set:

$$\mathcal{I} = \{\dots, 1, 2, \dots, L, \dots\}. \quad (7)$$

The definition of the infinite-level quantizer is as follows (see Figure 3 (left)):

$$q\{z_t\} = \beta_i \quad \text{if} \quad z_t \in \mathcal{J}_i, \quad i \in \mathcal{I}, \quad (8)$$

where the sets $\mathcal{J}_i = \{z_t : q_{i-1} \leq z_t < q_i\}$ are disjoint intervals and each β_i is the value that the quantizer takes in the region \mathcal{J}_i . The second is a finite-level quantizer, in which the output of the quantizer is limited to a minimum and maximum values (saturated quantizer) corresponding to the indices' set:

$$\mathcal{I} = \{1, 2, \dots, L-1, L\}. \quad (9)$$

The definition of the finite-level quantizer (see Figure 3 (right)) is similar to (8), where \mathcal{I} is defined in (9), $\mathcal{J}_1 = \{z_t : z_t < q_1\}$, $\mathcal{J}_L = \{z_t : q_{L-1} \leq z_t\}$, and $\mathcal{J}_i = \{z_t : q_{i-1} \leq z_t < q_i\}$ with $i = 2, \dots, L-1$.

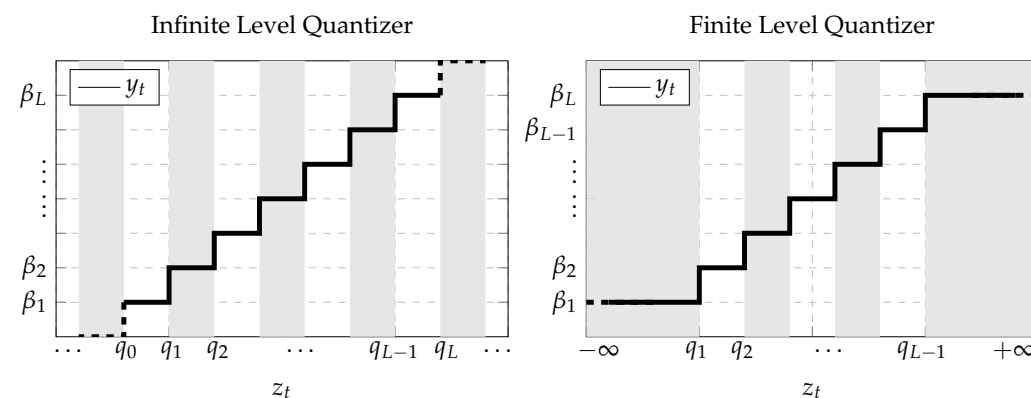


Figure 3. Representation of the (left) uniform infinite- and (right) finite-level quantizers defined in terms of the quantized values β_i and the intervals \mathcal{J}_i .

2.3. Problem Definition

The problem of interest can be defined as follows: Given the available data $u_{1:N} = \{u_1, u_2, \dots, u_N\}$ and $y_{1:N} = \{y_1, y_2, \dots, y_N\}$, where N is the data length, obtain the filtering and smoothing PDFs of the state given the quantized measurements, $p(x_t|y_{1:t})$ and $p(x_t|y_{1:N})$, respectively the state estimators:

$$\hat{x}_{t|t} = \mathbb{E}\{x_t|y_{1:t}\}, \quad (10)$$

$$\hat{x}_{t|N} = \mathbb{E}\{x_t|y_{1:N}\}, \quad (11)$$

and the corresponding covariance matrices of the estimation error:

$$\Sigma_{t|t} = \mathbb{E}\{(x_t - \hat{x}_{t|t})(x_t - \hat{x}_{t|t})^T|y_{1:t}\}, \quad (12)$$

$$\Sigma_{t|N} = \mathbb{E}\{(x_t - \hat{x}_{t|N})(x_t - \hat{x}_{t|N})^T|y_{1:N}\}, \quad (13)$$

where $t \leq N$ and $\mathbb{E}\{x|y\}$ denotes the conditional expectation of x given y .

3. Gaussian Sum Filtering and Smoothing for Quantized Data

Here, the Gaussian sum filtering and smoothing algorithms for quantized output data are explained in detail.

3.1. Gaussian Mixture Models

Gaussian mixture models refer to a convex combination of Gaussian densities corresponding to a random variable $\zeta \in \mathbb{R}^n$. Then, the PDF can be written as [60]: $p(\zeta) = \sum_{i=1}^K \varphi_i \mathcal{N}_{\zeta}(v_i, \Gamma_i)$ subject to $\varphi_i > 0$ and $\sum_{i=1}^K \varphi_i = 1$, where φ_i is the i th mixing weight, $v_i \in \mathbb{R}^n$ is the i th mean, and $\Gamma_i \in \mathbb{R}^{n \times n}$ is the i th covariance matrix. GMMs are used in a variety of applications to approximate non-Gaussian densities [61–63] and filtering and smoothing [64,65], to mention a few.

3.2. General Bayesian Framework

The well-known equations for Bayesian filtering (see, e.g., [31]) are given by:

$$p(x_t|y_{1:t}) = \frac{p(y_t|x_t)p(x_t|y_{1:t-1})}{p(y_t|y_{1:t-1})}, \quad (14)$$

$$p(x_{t+1}|y_{1:t}) = \int p(x_{t+1}|x_t)p(x_t|y_{1:t})dx_t, \quad (15)$$

where $p(x_t|y_{1:t})$ and $p(x_{t+1}|y_{1:t})$ are the measurement and time-update equations, respectively. The PDF $p(x_{t+1}|x_t)$ is obtained from the model in (5), and $p(y_t|y_{1:t-1})$ is a normalization constant. On the other hand, the well-known formula for Bayesian smoothing (see, e.g., [31]) is defined by:

$$p(x_t|y_{1:N}) = p(x_t|y_{1:t}) \int \frac{p(x_{t+1}|y_{1:N})p(x_{t+1}|x_t)}{p(x_{t+1}|y_{1:t})} dx_{t+1}. \quad (16)$$

However, the PDF in (16) is difficult to compute because of the division by a non-Gaussian density. This difficulty was overcome in [64] where the smoothing equations were separated in a two-fold formula filter: (i) the first formula, called the *backward filter*, defined by the following recursion:

$$p(y_{t+1:N}|x_t) = \int p(y_{t+1:N}|x_{t+1})p(x_{t+1}|x_t)dx_{t+1}, \quad (17)$$

$$p(y_{t:N}|x_t) = p(y_t|x_t)p(y_{t+1:N}|x_t), \quad (18)$$

where $p(y_{t+1:N}|x_t)$ and $p(y_{t:N}|x_t)$ are the backward prediction and the backward-measurement-update equations, respectively, and (ii) the second formula, given by:

$$p(x_t|y_{1:N}) = \frac{p(x_t|y_{1:t-1})p(y_{t:N}|x_t)}{p(y_{t:N}|y_{1:t-1})}, \quad (19)$$

where $p(x_t|y_{1:t-1})$ is the time-update equation from the filtering step, $p(y_{t:N}|y_{1:t-1})$ is a normalization constant, and $p(y_{t:N}|x_t)$ is obtained using the backward recursion given in (17) and (18).

Due to the non-Gaussianity of $p(y_t|x_t)$, the integrals in both the filtering and backward-filtering algorithms in (15) and (17), respectively, are difficult to compute or intractable. Widely used methods to deal with these integrals are Monte-Carlo-based algorithms, such as particle filtering and smoothing; see, e.g., [31,66]. These methods represent the posterior distributions $p(x_t|y_{1:t})$ and $p(x_t|y_{1:N})$ by a set of weighted random samples. However, in general, they exhibit a growing computational complexity as the model dimension increases. Here, an alternative method to compute the PMF $p(y_t|x_t)$ using Gauss–Legendre quadrature is proposed. This procedure results in a GMM form. Hence, the integrals in both the filtering and backward filtering in (15) and (17), respectively, can be computed in closed form.

Remark 1. Notice that since y_t is a discrete random variable, the measurement-update equation in (14) and the backward-measurement-update equation in (18) comprise PDFs and a PMF. Hence, generalized probability density functions are used here; see, e.g., [67].

3.3. Computing an Explicit Model of $p(y_t|x_t)$

From (8), it is observed that the output $z_t \in \mathcal{J}_i$ is mapped to a single value β_i . Then, the probability that y_t takes the value β_i is the same as the probability of z_t belonging to set \mathcal{J}_i , as shown in Figure 4. Notice that the quantizer regions \mathcal{J}_i include finite and semi-infinite intervals.

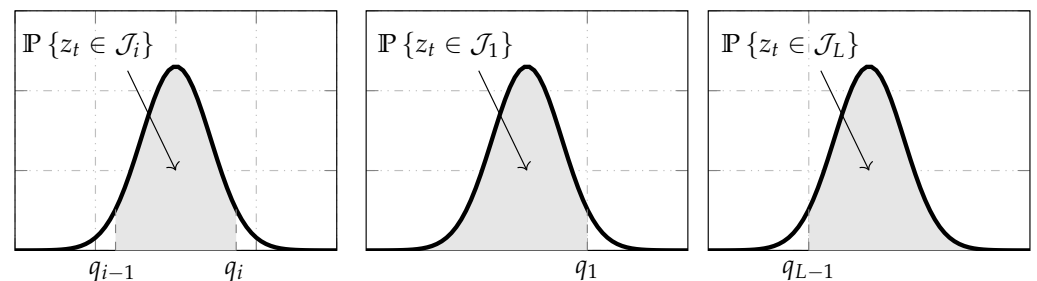


Figure 4. The shaded area represents the probability of y_t taking a value β_i that is equal to the probability of z_t belonging to set $\in \mathcal{J}_i$. Here, $\mathbb{P}\{\cdot\}$ denotes probability.

In the following theorem, the characterization of $p(y_t|x_t)$ is formalized via the computation of the probability $\mathbb{P}\{z_t \in \mathcal{J}_i\}$ shown in Figure 4 through the integral definition of probabilities [68]. Therefore, the approximation of this integral by using the Gauss–Legendre quadrature rule is presented (see, e.g., [69]).

Theorem 1. Consider the system (1)–(3), the infinite- and finite-level quantizers defined in (8). Then, the PMF of the discrete random variable y_t given the state x_t is given by:

$$p(y_t|x_t) = \int_{a_t}^{b_t} \mathcal{N}_{v_t}(0, R) dv_t, \quad (20)$$

where a_t and b_t are functions of the boundary values of each region of the quantizers and are defined in Table 1. In addition, the integral in (20) can be approximated using the Gauss–Legendre

quadrature rule, yielding:

$$p(y_t|x_t) \approx \sum_{\tau=1}^K \zeta_t^\tau \mathcal{N}_{\eta_t^\tau}(Cx_t + Du_t + \mu_t^\tau, R), \quad (21)$$

where K is the number of points from the Gauss–Legendre quadrature rule (defined by the user), ζ_t^τ , η_t^τ , and μ_t^τ are defined in Table 1, and ω_τ and ψ_τ are weights and points defined by the quadrature rule, given in, e.g., [69].

Table 1. Integral limits and parameters of Theorem 1. ILQ and FLQ mean infinite- and finite-level quantizers, respectively.

	FLQ: $y_t = \beta_1$	ILQ: $y_t = \beta_i$ with $i \in \mathcal{I}$ in (7) FLQ: $y_t = \beta_i$ with $i = 2, \dots, L - 1$	FLQ: $y_t = \beta_L$
a_t	$-\infty$	$q_{i-1} - Cx_t - Du_t$	$q_{L-1} - Cx_t - Du_t$
b_t	$q_1 - Cx_t - Du_t$	$q_i - Cx_t - Du_t$	∞
ζ_t^τ	$2\omega_\tau / (1 + \psi_\tau)^2$	$\omega_\tau (q_i - q_{i-1}) / 2$	$2\omega_\tau / (1 + \psi_\tau)^2$
η_t^τ	$-(1 - \psi_\tau) / (1 + \psi_\tau)$	$\psi_\tau (q_i - q_{i-1}) / 2$	$(1 - \psi_\tau) / (1 + \psi_\tau)$
μ_t^τ	$-q_1$	$-(q_i + q_{i-1}) / 2$	$-q_{L-1}$

Proof. From the infinite- and finite-level quantizers, it is observed that the random variable y_t can only take the values β_i with i in the indices' sets given in (7) and (9). Then, the probability of $y_t = \dots, \beta_1, \beta_2, \dots, \beta_L, \dots$ or $y_t = \beta_1, \beta_2, \dots, \beta_{L-1}, \beta_L$ is the same as the probability that the random variable z_t belongs to the sets \mathcal{J}_i . This probability can be obtained from the distribution function as follows:

$$\mathbb{P}\{y_t = \beta_i|x_t\} = \mathbb{P}\{z_t \in \mathcal{J}_i|x_t\}, \quad (22)$$

where $\mathbb{P}\{\cdot\}$ denotes probability. Considering the infinite-level quantizer and the output equation in (2), the following expressions are obtained for $y_t = \beta_i$ with $i = \dots, 1, 2, \dots, L, \dots$:

$$\begin{aligned} \mathbb{P}\{y_t = \beta_i|x_t\} &= \mathbb{P}\{q_{i-1} \leq z_t < q_i|x_t\}, \\ &= \mathbb{P}\{a_t \leq v_t < b_t|x_t\}, \end{aligned} \quad (23)$$

where $a_t = q_{i-1} - Cx_t - Du_t$ and $b_t = q_i - Cx_t - Du_t$. Additionally, for the finite-level quantizer, (23) holds for $y_t = \beta_i$ with $i = 2, \dots, L - 1$, and for $y_t = \beta_1$ and $y_t = \beta_L$, the following holds:

$$\mathbb{P}\{v_t < b_t|x_t\} \quad \text{if } y_t = \beta_1, \quad (24)$$

$$\mathbb{P}\{v_t \geq a_t|x_t\} \quad \text{if } y_t = \beta_L, \quad (25)$$

where $b_t = q_1 - Cx_t - Du_t$ and $a_t = q_{L-1} - Cx_t - Du_t$. Then, using the fact that $p(v_t) = \mathcal{N}_{v_t}(0, R)$ and using (23)–(25), the integral in (20) can be obtained, where the integral limits are given in Table 1.

On the other hand, the Gauss–Legendre quadrature for approximating a definite integral over the interval $[-1, 1]$ is given by:

$$\int_{-1}^1 f(r)dr \approx \sum_{\tau=1}^K \omega_\tau f(\psi_\tau), \quad (26)$$

where ω_τ and ψ_τ are the quadrature weights and the roots of the order K Legendre polynomial, respectively; see, e.g., [69]. Notice that the integral over $[a_t, b_t]$ can be mapped onto the interval $[-1, 1]$ using:

$$\int_{a_t}^{b_t} f(r)dr = \frac{b_t - a_t}{2} \int_{-1}^1 f\left(\frac{b_t - a_t}{2}r + \frac{b_t + a_t}{2}\right)dr, \quad (27)$$

and using the definition in (26), this integral is approximated by:

$$\int_{a_t}^{b_t} f(r)dr \approx \frac{b_t - a_t}{2} \sum_{\tau=1}^K \omega_\tau f\left(\frac{b_t - a_t}{2}\psi_\tau + \frac{b_t + a_t}{2}\right). \quad (28)$$

Defining $a_t = q_{i-1} - Cx_t - Du_t$, $b_t = q_i - Cx_t - Du_t$, and $f(v_t) = \mathcal{N}_{v_t}(0, R)$ with $i = \dots, 1, 2, \dots, L, \dots$, the approximation of $p(y_t|x_t)$ given in (21) for the infinite-level quantizer is derived.

For the finite-level quantizer, (28) holds for $y_t = \beta_i$ with $i = 2, \dots, L - 1$. Then, the approximation of $p(y_t|x_t)$ for $y_t = \beta_1$ and $y_t = \beta_L$ is defined. First, it is observed that the integral over the semi-infinite interval $[a_t, \infty)$ can be mapped onto the interval $(0, 1]$ using $r = a_t + (1 - \xi)/\xi$, so that:

$$\int_{a_t}^{\infty} f(r)dr = \int_0^1 f\left(a_t + \frac{1 - \xi}{\xi}\right) \frac{d\xi}{\xi^2}. \quad (29)$$

Then, using an appropriate change of variable, it can be mapped onto the interval $(-1, 1]$, yielding:

$$\int_{a_t}^{\infty} f(r)dr = \int_{-1}^1 f\left(a_t + \frac{1 - s}{1 + s}\right) \frac{2ds}{(1 + s)^2}. \quad (30)$$

Using the Gauss–Legendre approximation given in (26), the approximation of $p(y_t|x_t)$ for $y_t = \beta_L$ is obtained as follows:

$$\int_{a_t}^{\infty} f(r)dr \approx \sum_{\tau=1}^K \omega_\tau f\left(a_t + \frac{1 - \psi_\tau}{1 + \psi_\tau}\right) \frac{2}{(1 + \psi_\tau)^2}. \quad (31)$$

Defining $a_t = q_{L-1} - Cx_t - Du_t$, $b = \infty$, and $f(v_t) = \mathcal{N}_{v_t}(0, R)$, (21) is obtained. A similar procedure for the integral over the semi-infinite interval $(-\infty, b_t]$ can be applied using $r = b_t - (1 - \xi)/\xi$ to find the approximation of $p(y_t|x_t)$ for $y_t = \beta_1$, as follows:

$$\int_{-\infty}^{b_t} f(r)dr \approx \sum_{\tau=1}^K \omega_\tau f\left(b_t - \frac{1 - \psi_\tau}{1 + \psi_\tau}\right) \frac{2}{(1 + \psi_\tau)^2}. \quad (32)$$

Defining $a_t = -\infty$, $b_t = q_1 - Cx_t - Du_t$, and $f(v_t) = \mathcal{N}_{v_t}(0, R)$, (21) is obtained. This completes the proof. \square

Remark 2. Notice that any quadrature rule, such as Newton–Cotes, Gauss–Laguerre, and Gauss–Hermite (see, e.g., [69,70]), used to approximate the integral in (20), yields a weighted sum of Gaussian distributions evaluated as a linear function of the values q_i and the state x_t . Furthermore, it is possible to interpret $p(y_t|x_t)$ as a weighted sum of Gaussian distributions in the random variable x_t . This weighted sum is denoted as the GMM structure. Thus, this structure is considered for developing the Gaussian sum filtering and smoothing algorithms that deal with quantized data.

Remark 3. Notice that in [70], a suboptimal filtering algorithm called the quadrature Kalman filter, where a linear regression is used to linearize the nonlinear process and measurement functions using the Gauss–Hermite quadrature rule, was developed. This approach is not directly applicable to the problem of interest in this paper, so that the quantizer is a nondifferentiable nonlinearity.

On the other hand, the cubature Kalman filter [70] and smoother [71] are approaches that use the spherical–radial cubature rule to approximate the n -dimensional integrals when computing the expected values in (10)–(13) in a nonlinear state-space model. These integral approximations are obtained under the assumption that $p(x_t|y_{1:t})$ and $p(x_t|y_{1:N})$ are Gaussian distributed. The difference between these approaches and the proposed method in this paper is that the Gauss quadrature rule was used to approximate the integral in the probabilistic model of $p(y_t|x_t)$. It is clear that in this paper, the Gaussian assumption in the filtering and smoothing densities was not used. In fact, it is shown that $p(x_t|y_{1:t})$ and $p(x_t|y_{1:N})$ are non-Gaussian PDFs.

3.4. Understanding the Gaussian Sum Filtering and Smoothing Algorithms

The general Bayesian framework for filtering leads to the product $p(y_t|x_t)$ and $p(x_t|y_{1:t-1})$, as shown in (14). From the definition of $p(y_t|x_t)$ in (21), it is observed that $p(x_t|y_{1:t})$ results in the product of two GMMs. This, in turn, results in a GMM with more components than the individual factors $p(y_t|x_t)$ and $p(x_t|y_{1:t-1})$. This implies that the time-update equation $p(x_{t+1}|y_{1:t})$ in (15) is also a GMM distribution. A similar situation occurs in the backward-measurement-update equation $p(y_{t:N}|x_t)$ in (18), which is the product of two GMM structures $p(y_t|x_t)$ and $p(y_{t+1:N}|x_t)$. This yields another GMM structure with more components than the individual factors, which implies that the backward-prediction equation in (17) is also a GMM structure. For the clarity of the presentation, reducing the product of two GMMs (structures) into one is necessary.

In order to understand the transformation of the product of two summations into another summation, the following sums are considered: $g = \sum_{\tau=1}^K g_\tau$ and $f = \sum_{\ell=1}^M f_\ell$. Then, for each two-tuple (τ, ℓ) where $\tau = 1, \dots, K$ and $\ell = 1, \dots, M$, the product $h = fg$ is another summation and has KM terms indexed by $k = (\ell - 1)K + \tau$. Then, reordering these terms, the following sum is obtained: $h = \sum_{k=1}^{KM} h_k$, where $h_k = f_\ell g_\tau$.

3.5. Gaussian Sum Filtering for Quantized Data

Using the approximation of the PMF $p(y_t|x_t)$ defined in Theorem 1, the Gaussian sum filtering algorithm can be derived using (14) and (15) as follows:

Theorem 2 (Gaussian sum filter). Consider the system in (4)–(6) and the approximation of $p(y_t|x_t)$ in (21). The filtering equations for state-space systems with quantized output data are defined as follows:

Initialization: For $t = 1$, the predicted distribution is $p(x_1) = \mathcal{N}_{x_1}(\mu_1, P_1)$, where $p(x_1)$ is the prior distribution of the initial state. Then, for $t = 1, \dots, N$, the measurement-update and the time-update equations for the Gaussian sum filtering are defined as follows:

Measurement update: The PDF of the current state x_t given the current and previous measurements, that is y_1, \dots, y_t , is the GMM given by:

$$p(x_t|y_{1:t}) = \sum_{k=1}^{M_{t|t}} \gamma_{t|t}^k \mathcal{N}_{x_t}(\hat{x}_{t|t}^k, \Sigma_{t|t}^k), \quad (33)$$

where $\gamma_{t|t}^k$, $\hat{x}_{t|t}^k$, and $\Sigma_{t|t}^k$ are given in Appendix B, $M_{t|t} = KM_{t|t-1}$, and $M_{t|t-1}$ is the number of Gaussian components in the time update step. The initial values satisfy that $M_{1|0} = 1$, $\gamma_{1|0} = 1$, $\hat{x}_{1|0} = \mu_1$, and $\Sigma_{1|0} = P_1$.

Time update: The PDF of the future state x_{t+1} , one-step-ahead prediction given the measurements y_1, \dots, y_t , is the GMM given by:

$$p(x_{t+1}|y_{1:t}) = \sum_{k=1}^{M_{t+1|t}} \gamma_{t+1|t}^k \mathcal{N}_{x_{t+1}}(\hat{x}_{t+1|t}^k, \Sigma_{t+1|t}^k), \quad (34)$$

where $M_{t+1|t} = M_{t|t}$, $\gamma_{t+1|t}^k$, $\hat{x}_{t+1|t}^k$, and $\Sigma_{t+1|t}^k$ are given in Appendix B.

Proof. Consider the recursion in (14) and (15). The required PDF $p(x_{t+1}|x_t)$ and PMF $p(y_t|x_t)$ are obtained from (5) and (21), respectively. Then, using the measurement-update equations in (14) and Lemma 2 in [65] (p. 86), the following expression is obtained:

$$p(x_t|y_{1:t}) \propto \sum_{\tau=1}^K \sum_{\ell=1}^{M_{t|t-1}} \zeta_t^\tau \gamma_{t|t-1}^\ell \mathcal{N}_{\eta_t^\tau}(\kappa_t^{\ell\tau}, V_t^\ell) \mathcal{N}_{x_t}(\hat{x}_{t|t}^k, \Sigma_{t|t}^k) \quad (35)$$

where $\kappa_t^{\ell\tau}$, V_t^ℓ , $\hat{x}_{t|t}^k$, and $\Sigma_{t|t}^k$ are defined in (A11), (A12), (A7), and (A8), respectively. Notice that $\mathcal{N}_{\eta_t^\tau}(\kappa_t^{\ell\tau}, V_t^\ell)$ is a coefficient since the measurement y_t is available. Then, rewriting the double summation in (35) as a single one with a new index $k = (\ell - 1)K + \tau$, the measurement-update equation in (33) is derived defining $M_{t|t} = KM_{t|t-1}$, the weights $\bar{\gamma}_{t|t}^k$ as in (A9), and computing the corresponding normalization constant. On the other hand, using the time-update equation in (15) and Lemma 2 in [65] (p. 86), the following equation is obtained:

$$p(x_{t+1}|y_{1:t}) = \sum_{k=1}^{M_{t|t}} \gamma_{t|t}^k \int \mathcal{N}_{x_t}(m_t^k, S_t^k) \mathcal{N}_{x_{t+1}}(\hat{x}_{t+1|t}^k, \Sigma_{t+1|t}^k) dx_t, \quad (36)$$

where $\hat{x}_{t+1|t}^k$ and $\Sigma_{t+1|t}^k$ are defined in (A14), and (A15), respectively. Then, solving this integral, the time-update equation in (34) can be derived defining $M_{t+1|t} = M_{t|t}$, the weights $\gamma_{t+1|t}^k$ as in (A13), and computing the corresponding normalization constant. This completes the proof. \square

3.6. Computing the State Estimator $\hat{x}_{t|t}$ from a GMM

Provided $p(x_t|y_{1:t})$ in (33) as a GMM, the state estimator given in (10) and the covariance matrix of the estimation error in (12) can be computed as follows:

$$\hat{x}_{t|t} = \sum_{k=1}^{M_{t|t}} \gamma_{t|t}^k \hat{x}_{t|t}^k, \quad (37)$$

$$\Sigma_{t|t} = \sum_{k=1}^{M_{t|t}} \gamma_{t|t}^k \left[\Sigma_{t|t}^k + (\hat{x}_{t|t}^k - \hat{x}_{t|t})(\hat{x}_{t|t}^k - \hat{x}_{t|t})^T \right]. \quad (38)$$

3.7. Backward Filtering for Quantized Data

Using the approximation of the PMF $p(y_t|x_t)$ defined in Theorem 1, the backward filter recursion can be derived as follows:

Theorem 3 (Backward filtering). Consider the system in (4)–(6) and the approximation of $p(y_t|x_t)$ in (21). Then, the backward filter for state-space systems with quantized output data is:

Initialization: For $t = N$, the backward measurement update is given by:

$$p(y_N|x_N) = \sum_{k=1}^{S_{N|N}} \epsilon_{N|N}^k \lambda_{N|N}^k \exp \left\{ -\frac{1}{2} \left(x_N^T F_{N|N}^k x_N - 2G_{N|N}^{kT} x_N + H_{N|N}^k \right) \right\}, \quad (39)$$

where $S_{N|N} = K$, and:

$$\begin{aligned} \epsilon_{N|N}^k &= \zeta_N^k, & \lambda_{N|N}^k &= (\det\{2\pi R\})^{-1/2} & \theta_N^k &= \eta_N^k - Du_N - \mu_N^k, \\ F_{N|N}^k &= C^T R^{-1} C, & G_{N|N}^{kT} &= \theta_N^{kT} R^{-1} C, & H_{N|N}^k &= \theta_N^{kT} R^{-1} \theta_N^k, \end{aligned} \quad (40)$$

where ζ_N^k , η_N^k , and μ_N^k are defined in Table 1. Then, for $t = N - 1, \dots, 1$, the backward prediction and the backward-measurement-update equations are defined as follows:

Backward predictions: The backward-prediction equation is given by:

$$p(y_{t+1:N}|x_t) = \sum_{k=1}^{S_{t|t+1}} \epsilon_{t|t+1}^k \lambda_{t|t+1}^k \exp\left\{-\frac{1}{2}\left(x_t^T F_{t|t+1}^k x_t - 2G_{t|t+1}^{kT} x_t + H_{t|t+1}^k\right)\right\}, \quad (41)$$

where $S_{t|t+1} = S_{t+1|t+1}$, $S_{t+1|t+1}$ is the number of components in the backward-measurement-update step, and $\epsilon_{t|t+1}^k, \lambda_{t|t+1}^k, F_{t|t+1}^k, G_{t|t+1}^{kT}, H_{t|t+1}^k$ are given in Appendix C.

Backward-measurement update: The distribution of $y_{t:N}|x_t$ evaluated at y_t, \dots, y_N is:

$$p(y_{t:N}|x_t) = \sum_{k=1}^{S_{t|t}} \epsilon_{t|t}^k \lambda_{t|t}^k \exp\left\{-\frac{1}{2}\left(x_t^T F_{t|t}^k x_t - 2G_{t|t}^{kT} x_t + H_{t|t}^k\right)\right\}, \quad (42)$$

where $S_{t|t} = KS_{t|t+1}$, $S_{t|t+1}$ is the number of components in the backward-prediction step, and $\epsilon_{t|t}^k, \lambda_{t|t}^k, F_{t|t}^k, G_{t|t}^{kT}, H_{t|t}^k$ are given in Appendix C.

Proof. Consider the recursion in (17) and (18). The required PDF $p(x_{t+1}|x_t)$ and PMF $p(y_t|x_t)$ are given by Equations (5) and (21), respectively. The proof is carried out by induction in reverse time. First, it is verified that it holds for $t = N - 1$, then it is assumed to be true for $t = s + 1$, and finally, it is verified that it holds for $t = s$. Notice that the recursion starts in $t = N$ with $p(y_{N:N}|x_N) = p(y_N|x_N)$, which is defined in (39). From (17), at time $t = N - 1$, the backward-prediction step is defined as:

$$p(y_{N:N}|x_{N-1}) = \int p(y_{N:N}|x_N)p(x_N|x_{N-1})dx_N, \quad (43)$$

where $p(y_{N:N}|x_N)$ is given by (39) and $p(x_N|x_{N-1})$ is defined by the system model in (5). From the definition given in (43), the equation below is obtained:

$$p(y_{N:N}|x_{N-1}) = \sum_{k=1}^{S_{N|N}} \frac{\epsilon_{N|N}^k \lambda_{N|N}^k}{\sqrt{\det\{2\pi Q\}}} \int \exp\left\{-\frac{1}{2}\left[x_N^T \bar{P}_N^k x_N - 2\bar{V}_N^{kT} x_N + \bar{S}_N^k\right]\right\} dx_N, \quad (44)$$

where $\bar{P}_N^k = (F_{N|N}^k + Q^{-1})$, $\bar{V}_N^k = (G_{N|N}^k + J_{N-1})$, $\bar{S}_N^k = (H_{N|N}^k + L_{N-1})$, $J_{N-1}^T = (Ax_{N-1} + Bu_{N-1})^T Q^{-1}$, and $L_{N-1} = (Ax_{N-1} + Bu_{N-1})^T Q^{-1}(Ax_{N-1} + Bu_{N-1})$. Then, completing the square and solving the integral in (44), it is obtained:

$$p(y_{N:N}|x_{N-1}) = \sum_{k=1}^{S_{N-1|N}} \epsilon_{N-1|N}^k \lambda_{N-1|N}^k \times \exp\left\{-\frac{1}{2}\left(x_{N-1}^T F_{N-1|N}^k x_{N-1} - 2G_{N-1|N}^{kT} x_{N-1} + H_{N-1|N}^k\right)\right\}, \quad (45)$$

where $S_{N-1|N} = S_{N|N}$ and the remaining terms in (45) are defined in (A22)–(A26), but evaluated at $t = N - 1$. The backward-measurement-update step in (18) is as follows:

$$p(y_{N-1:N}|x_{N-1}) = p(y_{N-1}|x_{N-1})p(y_{N:N}|x_{N-1}). \quad (46)$$

Thus, using (45) in the definition given in (46), it is obtained:

$$\begin{aligned}
p(y_{N-1:N}|x_{N-1}) &= \sum_{\tau=1}^K \sum_{\ell=1}^{S_{N-1|N}} \frac{\varsigma_{N-1}^{\tau} \epsilon_{N-1|N}^{\ell} \lambda_{N-1|N}^{\ell}}{\sqrt{\det\{2\pi R\}}} \\
&\times \exp\left\{-\frac{1}{2}\left(x_{N-1}^T \tilde{F}_{N-1} x_{N-1} - 2\tilde{G}_{N-1}^{\tau T} x_{N-1} + \tilde{H}_{N-1}^{\tau}\right)\right\} \\
&\times \exp\left\{-\frac{1}{2}\left(x_{N-1}^T F_{N-1|N}^{\ell} x_{N-1} - 2G_{N-1|N}^{\ell T} x_{N-1} + H_{N-1|N}^{\ell}\right)\right\},
\end{aligned} \tag{47}$$

where $\tilde{F}_{N-1} = C^T R^{-1} C$, $\tilde{G}_{N-1}^{\tau T} = \theta_{N-1}^{\tau T} R^{-1} C$, and $\tilde{H}_{N-1}^{\tau} = \theta_{N-1}^{\tau T} R^{-1} \theta_{N-1}^{\tau}$ with $\theta_{N-1}^{\tau} = \eta_{N-1}^{\tau} - Du_{N-1} - \mu_{N-1}^{\tau}$. Finally, rewriting the double summation in the last equation into a single one with a new index $k = (\ell - 1)K + \tau$ results in:

$$\begin{aligned}
p(y_{N-1:N}|x_{N-1}) &= \sum_{k=1}^{S_{N-1|N-1}} \epsilon_{N-1|N-1}^k \lambda_{N-1|N-1}^k \\
&\times \exp\left\{-\frac{1}{2}\left[x_{N-1}^T F_{N-1|N-1}^k x_{N-1} - 2G_{N-1|N-1}^{kT} x_{N-1} + H_{N-1|N-1}^k\right]\right\},
\end{aligned} \tag{48}$$

where $S_{N-1|N-1} = KS_{N-1|N}$ and the remaining terms in (48) are the same as the ones shown in (A16)–(A21), but evaluated at $t = N - 1$. Thus, it is concluded that it holds for $t = N - 1$. A similar procedure was applied to find that, for both the backward-prediction and backward-measurement-update steps in the backward filter, it yields the same expressions in (A22)–(A26) and (A16)–(A21), but evaluated at $t = s$. Thus, it is concluded that Theorem 3 holds for all t . \square

From Theorems 2 and 3, it is clear that the number of elements in the mixture grows exponentially with every iteration, making the algorithm computationally intractable after a few iterations. In addition, it would be necessary to save and manage a large amount of information in every iteration of the Gaussian sum filtering and in the backward recursion. Therefore, an algorithm that reduces the number of GMM components should be implemented in every iteration of these two algorithms in order to keep the number of components bounded. Different methods have been proposed to perform this kind of procedure, termed Gaussian sum reduction, such as pruning, joining, and integral-squared-error-based methods; see [70]. In this work, the Kullback–Leibler approach for Gaussian sum reduction proposed in [64,72] was used. The idea behind the Gaussian sum reduction is to transform the GMM $\{\varphi_i, v_i, \Gamma_i\}_{i=1}^J$ into a GMM $\{\varphi_i, v_i, \Gamma_i\}_{i=1}^S$, where $1 \leq S \leq J$. In [64], it was suggested to use a measure of dissimilarity between two components and pooling the pair of components that minimize this measure. Then, based on this idea, in [72], the Kullback–Leibler information number was used as the measure of dissimilarity. The author in [72] provided an algorithm to merge two components so that the merged component preserves the first- and second-order moments of the original two components, which is given by:

$$(\varphi_{ij}, v_{ij}, \Gamma_{ij}) = \mathcal{M}\{(\varphi_i, v_i, \Gamma_i), (\varphi_j, v_j, \Gamma_j)\}, \tag{49}$$

where $\mathcal{M}\{\cdot, \cdot\}$ is a merging function such that:

$$\varphi_{ij} = \varphi_i + \varphi_j, \tag{50}$$

$$v_{ij} = \bar{\varphi}_i v_i + \bar{\varphi}_j v_j, \tag{51}$$

$$\Gamma_{ij} = \bar{\varphi}_i \Gamma_i + \bar{\varphi}_j \Gamma_j + \bar{\varphi}_i \bar{\varphi}_j (v_i - v_j)(v_i - v_j)^T, \tag{52}$$

where $\bar{\varphi}_i = \varphi_i / (\varphi_i + \varphi_j)$ and $\bar{\varphi}_j = \varphi_j / (\varphi_i + \varphi_j)$. The merging function applied to two components i and j ($i \neq j$) minimizes the dissimilarity measure $\mathcal{D}(i, j)$, defined as:

$$\mathcal{D}(i, j) = \frac{1}{2} [\varphi_{ij} \log \det\{\Gamma_{ij}\} - \varphi_i \log \det\{\Gamma_i\} - \varphi_j \log \det\{\Gamma_j\}], \quad (53)$$

The function $\mathcal{D}(i, j)$ satisfies $\mathcal{D}(i, j) = \mathcal{D}(j, i)$ and $\mathcal{D}(i, i) = 0$. This implies that the total number of combinations to merge is reduced to $0.5J(J - 1)$. The authors in [73] used Runnalls' algorithm in a Bayesian filtering environment, and they modified it to include a user-defined threshold for the number of components after reduction and a user-defined threshold ε that satisfies $\mathcal{D}(i, j) < \varepsilon$. In Algorithm 1, the steps for implementing the Gaussian sum filtering are summarized.

Algorithm 1: Gaussian sum filter algorithm for quantized output data

```

1 Input: The PDF of the initial state  $p(x_1)$ , e.g.,  $M_{1|0} = 1$ ,  $\gamma_{1|0} = 1$ ,  $\hat{x}_{1|0} = \mu_1$ ,
    $\Sigma_{1|0} = P_1$ . The points of the Gauss–Legendre quadrature  $\{\omega_\tau, \psi_\tau\}_{\tau=1}^K$ .
2 for  $t = 1$  to  $N$  do
3   Compute and store  $\zeta_t^\tau$ ,  $\eta_t^\tau$ , and  $\mu_t^\tau$  according to Theorem 1.
4   Measurement Update:
5   Set  $M_{t|t} = KM_{t|t-1}$ .
6   for  $\ell = 1$  to  $M_{t|t-1}$  do
7     for  $\tau = 1$  to  $K$  do
8       Compute the index  $k = (\ell - 1)K + \tau$ .
9       Compute and store  $\gamma_{t|t}^k$ ,  $\hat{x}_{t|t}^k$ , and  $\Sigma_{t|t}^k$  according to Theorem 2.
10    end
11  end
12  Compute the state estimation in (10) and the covariance matrix of the
   estimation error in (12) according to (37) and (38).
13  Perform the Gaussian sum reduction algorithm according to [73] to obtain the
   reduced GMM of  $p(x_t|y_{1:t})$ .
14  Time Update
15  Set  $M_{t+1|t} = M_{t|t}$ .
16  for  $k = 1$  to  $M_{t+1|t}$  do
17    Compute and store  $\gamma_{t+1|t}^k$ ,  $\hat{x}_{t+1|t}^k$ , and  $\Sigma_{t+1|t}^k$  according to Theorem 2.
18  end
19 end
20 Output: The state estimation in (10), the covariance matrix of the estimation error
   in (12), the filtering PDFs  $p(x_t|y_{1:t})$ , the predictive PDFs  $p(x_{t+1}|y_{1:t})$ , and the set
    $\{\zeta_t^\tau, \eta_t^\tau, \mu_t^\tau\}$ , for  $t = 1, \dots, N$ .

```

The backward recursion in Theorem 3 is used to obtain the smoothing PDF in (19). For this purpose, $p(y_{t:N}|x_t)$ is converted into a GMM structure of the random variable x_t . Then, the Gaussian sum reduction algorithm is applied to the GMM structure of $p(y_{t:N}|x_t)$ to obtain:

$$p(y_{t:N}|x_t) = \sum_{k=1}^{S_{\text{red}}} \delta_{t|t}^k \mathcal{N}_{x_t} \left(z_{t|t}^k, U_{t|t}^k \right), \quad (54)$$

where S_{red} is the number of Gaussian components kept after the Gaussian reduction procedure and $\delta_{t|t}^k$, $z_{t|t}^k$, and $U_{t|t}^k$ are the corresponding weight, mean, and covariance matrix. This reduced GMM structure is used to obtain the smoothing PDFs. However, for the next recursion in the backward filter, reconverting the reduced GMM structure into the backward filter form to obtain the reduced version of the backward-measurement-update step in (42) is required. This conversion process between the backward filter and GMM

structure is summarized in Lemma A3 in Appendix A. In Algorithm 2, the steps for implementing the backward filter are summarized.

Algorithm 2: Backward-filtering algorithm for quantized output data

```

1 Input: The initial backward measurement  $p(y_N|x_N)$  given in (39), e.g.,  $S_{N|N}$ ,  $\epsilon_{N|N}^k$ ,
 $\lambda_{N|N}^k$ ,  $F_{N|N}^k$ ,  $G_{N|N}^{kT}$ , and  $H_{N|N}^k$ . The set  $\{\zeta_t^\tau, \eta_t^\tau, \mu_t^\tau\}$  for  $t = 1, \dots, N$  computed in
Algorithm 1.
2 for  $t = N - 1$  to 1 do
3   Backward Prediction
4   Set  $S_{t|t+1} = S_{t+1|t+1}$ .
5   for  $k = 1$  to  $S_{t|t+1}$  do
6     Compute and store  $\epsilon_{t|t+1}^k$ ,  $\lambda_{t|t+1}^k$ ,  $F_{t|t+1}^k$ ,  $G_{t|t+1}^{kT}$ , and  $H_{t|t+1}^k$  according to
     Theorem 3.
7   end
8   Backward Measurement Update:
9   Set  $S_{t|t} = KS_{t|t+1}$ .
10  for  $\ell = 1$  to  $S_{t|t+1}$  do
11    for  $\tau = 1$  to  $K$  do
12      Compute the index  $k = (\ell - 1)K + \tau$ .
13      Compute and store  $\epsilon_{t|t}^k$ ,  $\lambda_{t|t}^k$ ,  $F_{t|t}^k$ ,  $G_{t|t}^{kT}$ , and  $H_{t|t}^k$  according to Theorem 3
14    end
15  end
16  Compute the GMM structure of  $p(y_{t:N}|x_t)$  using Lemma A3 in Appendix A.
  Perform the Gaussian sum reduction algorithm according to [73] to obtain the
  reduced GMM structure of  $p(y_{t:N}|x_t)$  given in (54).
17  Compute and store the backward filter form of the reduced version of
 $p(y_{t:N}|x_t)$  using Lemma A3 in Appendix A.
18 end
19 Output: The backward prediction  $p(y_{t+1:N}|x_t)$  and the backward measurement
update  $p(y_{t:N}|x_t)$  for  $t = N, \dots, 1$ .

```

3.8. Smoothing Algorithm with Quantized Data

In order to obtain the smoothing PDF in (19), the GMM structure $p(y_{t:N}|x_t)$ in the random variable x_t given in (54) is used. This GMM structure is multiplied by the time-update equation $p(x_t|y_{1:t-1})$ of the filtering algorithm given in (34). Then, the smoothing PDF is obtained from the following:

Theorem 4 (Gaussian sum smoothing). Consider the system in (4)–(6). Given $p(x_t|y_{1:t-1})$, $p(x_N|y_{1:N})$ and $p(y_{t:N}|x_t)$, then the smoothing PDF at time $t = N$ is given by $p(x_N|y_{1:N})$, and for $t = N - 1, \dots, 1$, the PDF $p(x_t|y_{1:N})$ is a GMM given by:

$$p(x_t|y_{1:N}) = \sum_{k=1}^{S_{t|N}} \epsilon_{t|N}^k \mathcal{N}_{x_t}(\hat{x}_{t|N}^k, \Sigma_{t|N}^k), \quad (55)$$

where $S_{t|N} = M_{t|t-1} S_{red}$, $M_{t|t-1}$ and S_{red} are given in (34) and (54), respectively, and $\epsilon_{t|N}^k$, $\hat{x}_{t|N}^k$, and $\Sigma_{t|N}^k$ are given in Appendix D.

Proof. Consider the definition of Bayesian smoothing given in (19), the time-update equation $p(x_t|y_{1:t-1})$ obtained from (34), and the reduced version of the measurement-update

step in the backward filter $p(y_{t:N}|x_t)$ defined in (54). For $t = N - 1, N - 2, \dots, 1$, it is obtained:

$$p(x_t|y_{1:N}) \propto \sum_{\tau=1}^{M_{t|t-1}} \sum_{\ell=1}^{S_{\text{red}}} \gamma_{t|t-1}^{\tau} \delta_{t|t}^{\ell} \mathcal{N}_{x_t}(z_{t|t}^{\ell}, U_{t|t}^{\ell}) \mathcal{N}_{x_t}(\hat{x}_{t|t-1}^{\tau}, \Sigma_{t|t-1}^{\tau}). \quad (56)$$

Defining $g(x_t) = \mathcal{N}_{x_t}(z_{t|t}^{\ell}, U_{t|t}^{\ell}) \mathcal{N}_{x_t}(\hat{x}_{t|t-1}^{\tau}, \Sigma_{t|t-1}^{\tau})$, the following equation is derived:

$$g(x_t) = \frac{\exp\left\{-\frac{1}{2}\left[\phi_{1t}^{\ell} + \phi_{2t}^{\tau} - \phi_{3t}^{\ell\tau}\right]\right\} \mathcal{N}_{x_t}\left(\left(L_t^{\ell\tau}\right)^{-1} \rho_t^{\ell\tau}, \left(L_t^{\ell\tau}\right)^{-1}\right)}{(2\pi)^{\frac{n}{2}} \sqrt{\det\{L_t^{\ell\tau}\} \det\{U_{t|t}^{\ell}\} \det\{\Sigma_{t|t-1}^{\tau}\}}}, \quad (57)$$

where $L_t^{\ell\tau}$, $\rho_t^{\ell\tau}$, ϕ_{1t}^{ℓ} , ϕ_{2t}^{τ} , and $\phi_{3t}^{\ell\tau}$ are defined in (A32), (A31), (A34), (A35), and (A33), respectively. Next, expressing the double summation in (56) as a single one with a new index $k = (\ell - 1)M_{t|t-1} + \tau$, it is obtained:

$$p(x_t|y_{1:N}) \propto \sum_{k=1}^{S_{t|N}} \bar{\epsilon}_{t|N}^k \mathcal{N}_{x_t}(\hat{x}_{t|N}^k, \Sigma_{t|N}^k), \quad (58)$$

where $S_{t|N} = M_{t|t-1}S_{\text{red}}$ and $\bar{\epsilon}_{t|N}^k$, $\Sigma_{t|N}^k$, and $\hat{x}_{t|N}^k$ are given in (A30), (A28), and (A29), respectively. Finally, the smoothing PDF in (55) is obtained by computing the normalized weights as (A27), and this completes the proof. \square

Provided $p(x_t|y_{1:N})$ in (55) as a GMM, the state estimator given in (11) and the covariance matrix of the estimation error in (13) can be computed as follows:

$$\hat{x}_{t|N} = \sum_{k=1}^{S_{t|N}} \epsilon_{t|N}^k \hat{x}_{t|N}^k, \quad (59)$$

$$\Sigma_{t|N} = \sum_{k=1}^{S_{t|N}} \epsilon_{t|N}^k \left[\Sigma_{t|N}^k + (\hat{x}_{t|N}^k - \hat{x}_{t|N})(\hat{x}_{t|N}^k - \hat{x}_{t|N})^T \right]. \quad (60)$$

In Algorithm 3, the steps to implement the Gaussian sum smoothing are summarized.

Algorithm 3: Gaussian sum smoothing algorithm for quantized output data

- 1 **Input:** The PDFs $p(x_t|y_{1:t-1})$ and $p(x_N|y_{1:N})$ obtained from Algorithm 1 and $p(y_{t:N}|x_t)$ obtained from Algorithm 2.
 - 2 Save the PDF $p(x_N|y_{1:N})$.
 - 3 **for** $t = N - 1$ **to** 1 **do**
 - 4 Set $S_{t|N} = M_{t|t-1}S_{\text{red}}$.
 - 5 **for** $\ell = 1$ **to** S_{red} **do**
 - 6 **for** $\tau = 1$ **to** $M_{t|t-1}$ **do**
 - 7 Compute the index $k = (\ell - 1)M_{t|t-1} + \tau$.
 - 8 Compute and store $\epsilon_{t|N}^k$, $\hat{x}_{t|N}^k$, and $\Sigma_{t|N}^k$ according to Theorem 4.
 - 9 **end**
 - 10 **end**
 - 11 Compute the state estimation (11) and the covariance matrix of the estimation error in (13) according to (59) and (60).
 - 12 **end**
 - 13 **Output:** The state estimation in (11), the covariance matrix of the estimation error in (13), and the smoothing PDFs $p(x_t|y_{1:N})$, for $t = 1, \dots, N$.
-

3.9. Computing the Smoothing Joint PDF $p(x_{t+1}, x_t | y_{1:N})$

In Theorems 2–4, the filtering and smoothing problems for quantized data are solved. However, many strategies for the identification of state-space systems [8,9] require the joint PDF $p(x_{t+1}, x_t | y_{1:N})$, which for the quantized output data case is given by the following:

Theorem 5. Consider the system in (4)–(6), the PDF $p(x_t | y_{1:t})$ and the backward prediction equation $p(y_{t+1:N} | x_{t+1})$ given in Theorems 2 and 3, respectively, and the PDF $p(x_{t+1} | x_t)$ given in Equation (5). Then, for $t = N - 1, \dots, 1$, the joint PDF $p(x_{t+1}, x_t | y_{1:N})$ is the GMM given by:

$$p(x_{t+1}, x_t | y_{1:N}) = \sum_{k=1}^{S_{t,t+1}} \alpha_k \mathcal{N}_{\chi_t}(\hat{\chi}_{t|N}^k, E_{t|N}^k), \quad (61)$$

where $S_{t,t+1} = M_{t|t} S_{t+1|t+1}$, $M_{t|t}$ and $S_{t+1|t+1}$ are given in (33) and (42), respectively, α_k , $\hat{\chi}_{t|N}^k$ and $E_{t|N}^k$ are given in Appendix E, and χ_t is the extended vector given by:

$$\chi_t^T = [x_{t+1}^T \quad x_t^T]^T. \quad (62)$$

At time $t = N$, the joint PDF $p(x_{N+1}, x_N | y_{1:N})$ is given by (61) with:

$$\begin{aligned} S_{N+1|N+1} &= 1, & \epsilon_{N+1|N+1}^\ell &= 1, & \lambda_{N+1|N+1}^\ell &= 1, \\ F_{N+1|N+1}^\ell &= 0, & G_{N+1|N+1}^{\ell T} &= 0, & H_{N+1|N+1}^\ell &= 0. \end{aligned} \quad (63)$$

Proof. Using Bayes' theorem, the joint PDF $p(x_{t+1}, x_t | y_{1:N})$ can be obtained as:

$$p(x_{t+1}, x_t | y_{1:N}) = \frac{p(y_{t+1:N} | x_{t+1}) p(x_{t+1} | x_t) p(x_t | y_{1:t})}{p(y_{t+1:N} | y_{1:t})}, \quad (64)$$

where $p(y_{t+1:N} | y_{1:t})$ is a normalization constant. The required PDFs $p(x_t | y_{1:t})$ and $p(x_{t+1} | x_t)$ and backward-prediction equation $p(y_{t+1:N} | x_{t+1})$ are given in (33), (5) and (42), respectively. Using Lemma A2, $p(x_t | y_{1:t})$ in (33) can be rewritten as:

$$p(x_t | y_{1:t}) = \sum_{\tau=1}^{M_{t|t}} \frac{\gamma_{t|t}^\tau}{\sqrt{\det\{2\pi\Sigma_{t|t}^\tau\}}} \exp\left\{-\frac{1}{2}\left(x_t^T \left(\Sigma_{t|t}^\tau\right)^{-1} x_t - 2J_t^{\tau T} x_t + L_t^\tau\right)\right\}, \quad (65)$$

where $J_t^{\tau T}$ and L_t^τ are defined in (A44) and (A45), respectively. Considering $p(y_{t+1:N} | x_{t+1})$, $p(x_t | y_{1:t})$, $p(x_{t+1} | x_t)$, and the extended vector of the state given in (62), the argument of these three functions can be written as follows:

$$A_1 = \chi_t^T \begin{bmatrix} 0 & 0 \\ 0 & (\Sigma_{t|t}^\tau)^{-1} \end{bmatrix} \chi_t - 2[0 \quad J_t^{\tau T}] \chi_t + L_t^\tau, \quad (66)$$

$$A_2 = \chi_t^T \begin{bmatrix} F_{t+1|t+1}^\ell & 0 \\ 0 & 0 \end{bmatrix} \chi_t - 2[G_{t+1|t+1}^{\ell T} \quad 0] \chi_t + H_{t+1|t+1}^\ell, \quad (67)$$

$$\begin{aligned} A_3 &= \chi_t^T \begin{bmatrix} Q^{-1} & -Q^{-1}A \\ -A^T Q^{-1} & A^T Q^{-1}A \end{bmatrix} \chi_t \\ &\quad - 2[u_t^T B^T Q^{-1} \quad -u_t^T B^T Q^{-1}A] \chi_t + u_t^T B^T Q^{-1} B u_t, \end{aligned} \quad (68)$$

Then, adding these expressions, the following equation is derived:

$$p(x_{t+1}, x_t | y_{1:N}) \propto \sum_{\tau=1}^{M_{t|t}} \sum_{\ell=1}^{S_{t+1|t+1}} \frac{\gamma_{t|t}^{\tau} \epsilon_{t+1|t+1}^{\ell} \lambda_{t+1|t+1}^{\ell}}{(2\pi)^n \sqrt{\det\{Q\}} \det\{\Sigma_{t|t}^{\tau}\}} \quad (69)$$

$$\times \exp\left\{-\frac{1}{2}\left(\chi_t^T \mathcal{F}_t^{\ell\tau} \chi_t - 2\mathcal{G}_t^{\ell\tau T} \chi_t + \mathcal{H}_t^{\ell\tau}\right)\right\},$$

where $\mathcal{F}_t^{\ell\tau}$, $\mathcal{G}_t^{\ell\tau T}$, and $\mathcal{H}_t^{\ell\tau}$ are defined in (A41), (A42), and (A43), respectively. Completing the square and expressing the double summation as a single one with a new index $k = (\ell - 1)M_{t|t} + \tau$, it is obtained:

$$p(x_{t+1}, x_t | y_{1:N}) \propto \sum_{k=1}^{S_{t,t+1}} \bar{\alpha}_k \mathcal{N}_{\chi_t}(\hat{\chi}_{t|N}^k, E_{t|N}^k), \quad (70)$$

where $S_{t,t+1} = M_{t|t}S_{t+1|t+1}$, $\bar{\alpha}_k$ is defined in (A39), and $\hat{\chi}_{t|N}^k$ and $E_{t|N}^k$ are defined in (A37) and (A38), respectively. Finally, the smoothing joint distribution in (64) for $t = N - 1, \dots, 1$ is derived by computing the normalized weights as in (A36). Notice that, for $t = N$, the smoothing joint PDF $p(x_{N+1}, x_N | y_{1:N}) = p(x_{N+1} | x_N) p(x_N | y_{1:N})$ is obtained as (61) considering the expressions given in (63). Then, the proof is finished. \square

In Algorithm 4, the steps to implement the Gaussian sum smoothing for computing the joint PDF $p(x_{t+1}, x_t | y_{1:N})$ are summarized.

Algorithm 4: Gaussian sum smoother to compute $p(x_{t+1}, x_t | y_{1:N})$ for quantized output data

- 1 **Input:** The PDF $p(x_t | y_{1:t})$ obtained in Algorithm 1, $p(y_{t+1:N} | x_{t+1})$ computed in Algorithm 2, and the PDF $p(x_{t+1} | x_t)$ given in Equation (5).
 - 2 **for** $t = 1$ **to** $N - 1$ **do**
 - 3 Set $S_{t,t+1} = M_{t|t}S_{t+1|t+1}$.
 - 4 **for** $\ell = 1$ **to** $S_{t+1|t+1}$ **do**
 - 5 **for** $\tau = 1$ **to** $M_{t|t}$ **do**
 - 6 Compute the index $k = (\ell - 1)M_{t|t} + \tau$.
 - 7 Compute and store α_k , $\hat{\chi}_{t|N}^k$, and $E_{t|N}^k$ according to Theorem 5.
 - 8 **end**
 - 9 **end**
 - 10 **end**
 - 11 Compute and store $p(x_{N+1}, x_N | y_{1:N})$ according to Theorem 5 with $S_{N+1|N+1}$, $\epsilon_{N+1|N+1}^{\ell}$, $\lambda_{N+1|N+1}^{\ell}$, $F_{N+1|N+1}^{\ell}$, $G_{N+1|N+1}^{\ell T}$, and $H_{N+1|N+1}^{\ell}$ given in (63).
 - 12 **Output:** The smoothing PDFs $p(x_{t+1}, x_t | y_{1:N})$, for $t = 1, \dots, N$.
-

4. Numerical Example

In this section, a numerical example to illustrate the benefits of this paper proposal is presented. Furthermore, a practical simulation example is studied: the problem of estimating the tank liquid level in a two-tank system is addressed. Typically, a numerical simulation approach is used for testing new algorithms and designs in applications of state estimation [74], control [75], and system identification [63], among others. This approach is used to evaluate the performance of the estimation algorithms, in order to avoid safety problems that can occur in a real-world processes.

To evaluate the performance of the proposed filtering and smoothing methods for quantized data, a comparison with three classical techniques is presented: standard Kalman filtering and smoothing [76], quantized Kalman filtering and smoothing [49,77], and particle filtering and smoothing [31]. Notice that it is also possible to implement a version

of particle filtering and smoothing algorithms using the approximation of $p(y_t|x_t)$ given in (21). However, in the numerical examples run for filtering, the computation time was high. In order to validate the approximation in Theorem 1, the standard particle filtering and smoothing algorithms with a large number of particles were used, where $p(y_t|x_t)$ was computed using the integral in (20) from the MATLAB function *mvncdf*, which computes the multivariate normal cumulative distribution function. The *true* filtering and smoothing PDFs were considered to be provided by the particle filter and smoother with 20,000 particles, which was defined as the ground truth.

In the following examples, the discrete-time system in state-space form given in (1)–(3) is used with:

$$y_t = \Delta_q \text{round}\{z_t/\Delta_q\}, \quad (71)$$

where the quantizer is defined in terms of the *round* function in MATLAB and the quantization step Δ_q . The infimum and supremum values of the sets \mathcal{J}_i defined in (8), q_{i-1} and q_i , can be calculated for the infinite-level quantizer as $q_{i-1} = y_t - 0.5\Delta_q$ and $q_i = y_t + 0.5\Delta_q$ for $i = \dots, 1, 2, \dots, L, \dots$.

The experiments were carried out on a computer with the following specifications: processor: Intel(R) Core(TM) i5-8300H CPU @ 2.30 GHz, RAM memory: 8.00 GB, operating system: Windows 10 with MATLAB 2020b.

4.1. Example 1: First-Order System

In this example, the following state-space system was considered:

$$x_{t+1} = 0.9x_t + 1.0u_t + w_t, \quad (72)$$

$$z_t = 2.0x_t + 0.5u_t + v_t, \quad (73)$$

where $w_t \sim \mathcal{N}_{w_t}(0,1)$ and $v_t \sim \mathcal{N}_{v_t}(0,0.5)$. Furthermore, the input was considered to be drawn from $\mathcal{N}_{u_t}(0,2)$ and $x_1 \sim \mathcal{N}_{x_1}(1,1)$. In Figure 5, the filtering PDFs for some time instants are shown. The quantization step used in this example and the number of Gaussian components to approximate $p(y_t|x_t)$ were considered to be $\Delta_q = 7$ and $K = 10$, respectively. Furthermore, $S_{\text{red}} = K$.

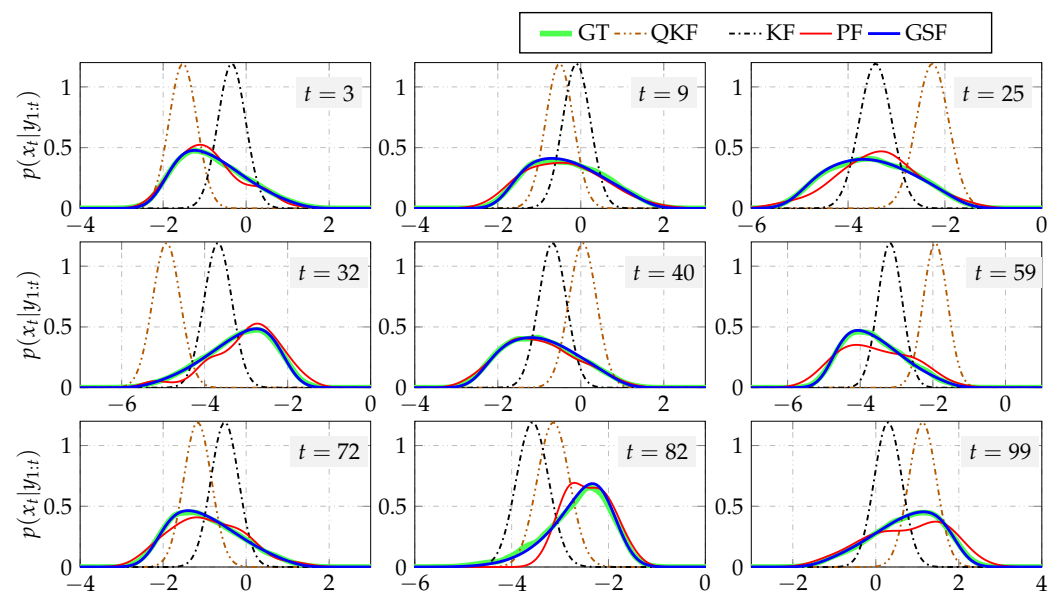


Figure 5. Example 1. Filtering PDFs at some instants of time for $\Delta_q = 7$, $K = 10$, and 100 particles. GT stands for ground truth. QKF, KF, PF, and GSF stand for quantized Kalman filter, Kalman filter, particle filter, and Gaussian sum filter, respectively.

Figure 5 shows that the filtering PDFs obtained with our proposal, the Gaussian sum filter, were that best fit to the ground truth. In contrast, the PDFs obtained with the Kalman

filter, quantized Kalman filter, and particle filter were different from the ground truth. Notice that the results obtained using the particle filter with 100 particles were good at $t = 3, 9, 40$, whilst the resulting PDFs for $t = 25, 32, 59, 72, 82, 99$ differed slightly more from the ground truth. However, the performance of particle filtering can be improved by increasing the number of particles, which produces an increment in the execution time.

To compare the accuracy of the state estimation, 100 Monte Carlo trials were run, and the mean-squared error between the true state and the estimation obtained by the Kalman filter, quantized Kalman filter, Gaussian sum filter, and particle filter is computed as follows:

$$\text{MSE} = \frac{1}{\mathcal{R}} \sum_{k=1}^{\mathcal{R}} \|x_t - \hat{x}_{t|t}\|_2^2, \quad (74)$$

where x_t is the true state (which in a real-world system is not available, but in simulations, it can be used to analyze the performance of the estimation techniques), $\hat{x}_{t|t}$ is the estimation of the state system, and $\|\cdot\|_2^2$ denotes the squared Euclidean norm. In Figure 6, the box plots corresponding to 100 Monte Carlo runs for $\Delta_q = \{3, 5, 7\}$ are shown.

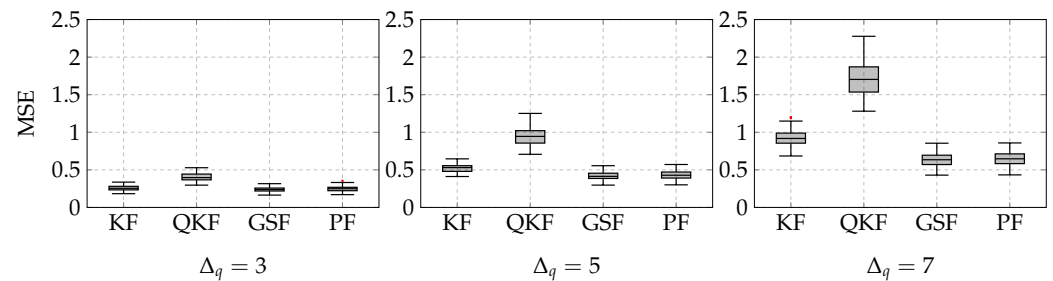


Figure 6. MSE between the true and estimated state for $\Delta_q = \{3, 5, 7\}$. KF, QKF, GSF, and PF stand for the Kalman filter, quantized Kalman filter, Gaussian sum filter, and particle filter, respectively.

Figure 6 shows that the MSE between the true state and the estimation obtained with the Kalman filter and quantized Kalman filter increased fast as Δ_q increased. However, the MSE increased slowly for the state estimation obtained with the Gaussian sum filter and particle filter.

In Figure 7, the smoothing PDFs at time $t = 100$ are shown. The quantization step was considered to be chosen from $\Delta_q = \{3, 5, 7\}$, and the number of Gaussian components to approximate $p(y_t|x_t)$ was chosen from $K = \{6, 8, 10\}$. Furthermore, the number of Gaussian components kept after the reduction procedure was considered as $S_{\text{red}} = K$. In order to obtain the adequate number of particles to compare the particle smoother with the Gaussian sum smoother, 100 Monte Carlo simulations were carried out to obtain the number of particles that yielded smoothing PDFs that were as close to the true PDFs as the Gaussian sum smoother using $K = \{6, 8, 10\}$. For comparison purposes, the particle smoother execution time corresponding to the time required to implement the particle smoother algorithm using the number of particles that produces a similar result to the Gaussian sum smoother with K components is defined as $Par(K)$. The L_2 -norm of the difference between the true and the estimated PDFs as the measure of similarity was used:

$$\|q - \hat{q}\|_2 = \left[\sum_{k=1}^M |q - \hat{q}|^2 \right]^{1/2}, \quad (75)$$

where q represents the true PDF and \hat{q} represents the estimated PDF, which was chosen so that $\|q - \hat{q}\|_2 < 1 \times 10^{-6}$. The approximated number of particles (labeled in each PDF in Figure 7) was used to compare the execution time of both algorithms, the Gaussian sum smoother and particle smoother, and the results are shown in Table 2. Figure 7 shows that the smoothing PDFs obtained with our proposal using a small number of Gaussian components and the PDFs obtained using the particle smoother with a large number of

particles fit the ground truth. In contrast, the PDFs obtained with the Kalman smoother and quantized Kalman smoother were different from the ground truth. Furthermore, the execution time in Table 2 shows that the required time to perform the Gaussian sum smoother was less compared to the time to perform the particle smoother, which needs a large number of particles to produce a result similar to the Gaussian sum smoother.

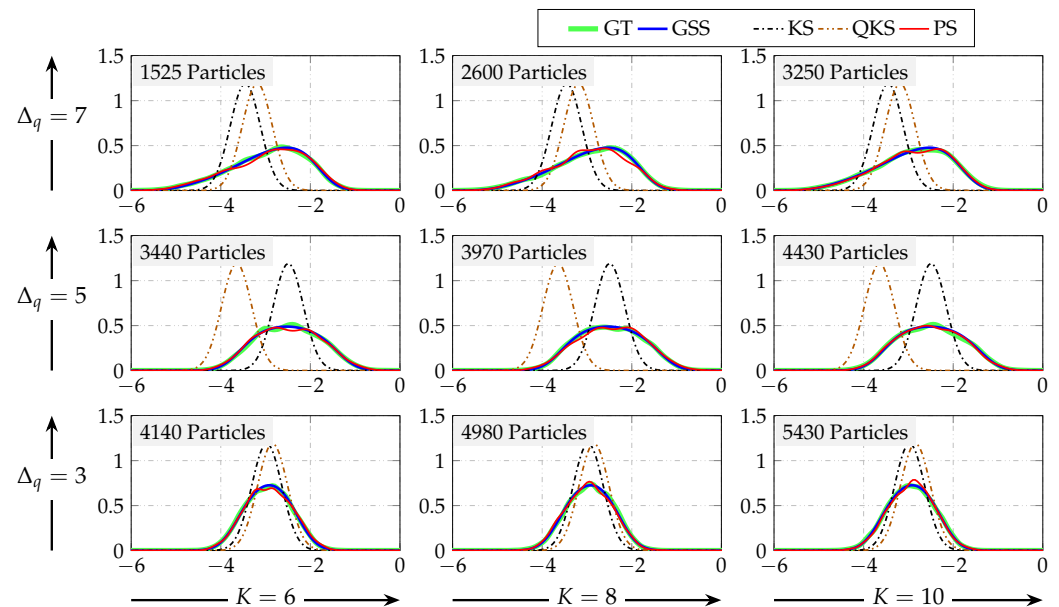


Figure 7. Smoothing PDF at time $t = 100$, for $K = \{6, 8, 10\}$, where K increases to the right, and for $\Delta_q = \{3, 5, 7\}$, where Δ_q increases upwards. GT stands for the ground truth. GSS, KS, QKS, and PS stand for the Gaussian sum smoother, Kalman smoother, quantized Kalman smoother, and particle smoother, respectively.

Table 2. Time in seconds required to perform the smoothing algorithm for the scalar system. $Par(K)$ represents the number of particles (labeled in Figure 7) that produce a similar result to the Gaussian sum smoother with K components. KS, QKS, GSS, and PS stand for the Kalman smoother, quantized Kalman smoother, Gaussian sum smoother, and particle smoother, respectively.

	KS	QKS	GSS			PS		
Δ_q	-	-	$K = 6$	$K = 8$	$K = 10$	Par(6)	Par(8)	Par(10)
7	0.0379	0.1494	0.4158	0.4221	0.4735	2.0420	3.3538	4.0462
5	0.0048	0.0044	0.2695	0.3706	0.4588	3.0074	3.5665	4.0192
3	0.0033	0.0043	0.2609	0.3491	0.4618	4.2471	5.5501	6.3408

From the results shown in Figures 5–7 and Table 2, it can be concluded that:

- The filtering and smoothing PDFs are non-Gaussian, although the process and output noises in (1) and (2) are Gaussian distributed;
- The accuracy of the standard and quantized Kalman filtering and smoothing decreased as the quantization step increased;
- The state estimates obtained with particle filter and smoother were similar to the results obtained using the Gaussian sum filter and smoother. However, the characterization of the filtering and smoothing PDFs using the Gaussian sum filter and smoother were better than the PDF obtained by the particle filter and smoother. Notice that a correct characterization of a PDF is important when high-order moments need to be computed, especially in system identification tasks;
- In order to implement the Gaussian sum filter and smoother, the parameters K (the number that defines the quality of the $p(y_t|x_t)$ approximation) and S_{red} (the Gaussian

components kept after the Gaussian sum reduction algorithm) need to be chosen by the user. These parameters can be found in a simulation study by the trial and error approach and should be set by a trade-off between the time complexity and the accuracy of the estimation. A large value of K produces an accurate estimate, but a high computational load;

- The larger the quantization step is, the larger the number of Gaussian components, K , needed to approximate $p(y_t|x_t)$ in order to obtain an accurate approximation. However, for a large quantization step, the number K needed to obtain a good approximation of the filtering and smoothing PDFs is relatively small compared to the number of particles required to obtain similar results using the particle filter and smoother;
- The maximum number of Gaussian components kept after the Gaussian reduction procedure is important for the accuracy of the approximation. In the simulations, $S_{\text{red}} = K$ was used. Furthermore, it was noticed that once an adequate S_{red} was defined, incrementing this value did not produce a significant improvement in the estimation. However, this increment in S_{red} was really critical for the resulting numerical complexity of the algorithm (and hence, the execution time), which increased since the Gaussian sum reduction procedure (e.g., Kullback–Leibler reduction) utilized more time to reduce a large amount of Gaussian components;
- The Gaussian sum smoother execution time for all values of Δ_q was small. This occurred because in each case, a relatively small number of Gaussian components to approximate $p(y_t|x_t)$ were used. However, the particle smoother execution time is variable for different values of Δ_q . As Δ_q decreased, the L_2 -norm between the Gaussian sum smoother and the ground truth decreased, and a larger number of particles to obtain a comparable L_2 -norm between the particle smoother and the ground truth were required.

4.2. Real-World Application: Tank Liquid Level

In this example, the problem of estimating the tank liquid level in a two-tank system by using the measurements taken by a low-cost sensor based on a variable resistor that is attached to an arm with a floater is considered; see Figure 8.

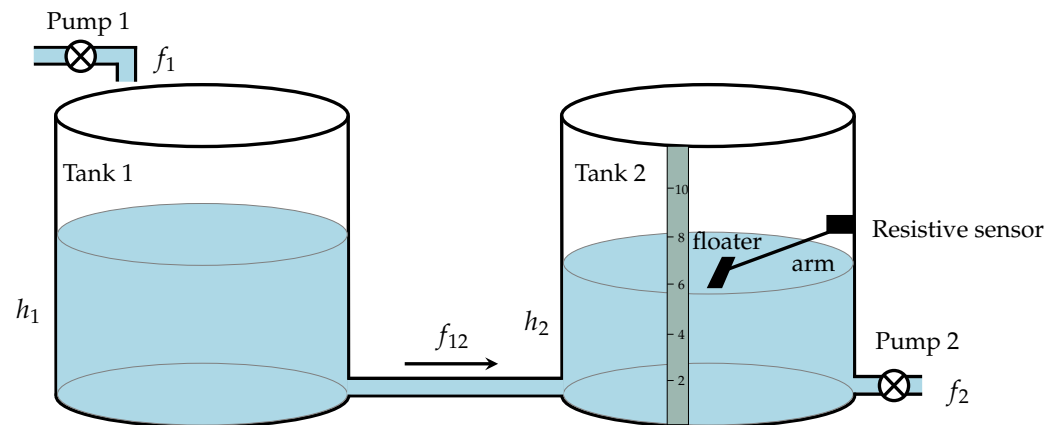


Figure 8. Two-tank system. h_1 and h_2 denote the liquid level in Tank 1 and Tank 2, respectively. The liquid flows into Tank 1 at a rate f_1 and out of Tank 2 at a rate f_2 . The quantizer has minimum and maximum values $\beta_1 = 2$ and $\beta_L = 10$.

A linearized model of this system can be found in [78]. Here, it was assumed that h_2 can be measured and h_1 cannot. The discrete-time version of the model in [78] with sample time 0.1 s was considered:

$$x_{t+1} = \begin{bmatrix} 0.9959 & 0.0041 \\ 0.0041 & 0.9959 \end{bmatrix} x_t + \begin{bmatrix} 0.0998 & -0.0002 \\ 0.0002 & -0.0998 \end{bmatrix} u_t + w_t, \quad (76)$$

$$z_t = \begin{bmatrix} 0 & 1.0 \end{bmatrix} x_t + v_t, \quad (77)$$

where $x_t = [h_1 \ h_2]^T$ and $u_t = [f_1 - k \ f_2 + k]^T$ with $k = 0.4111$. The sensor measures h_2 vary its resistance in discrete steps, with minimum and maximum values $\beta_1 = 2$ and $\beta_L = 10$. To simulate this system, it was considered that: $w_t \sim \mathcal{N}_{w_t}(0, 0.001I_2)$, $v_t \sim \mathcal{N}_{v_t}(0, 0.0001)$; the input $[f_1 \ f_2]^T$ was drawn from $\mathcal{N}([10 \ 2]^T, 10I_2)$; the initial condition $x_1 \sim \mathcal{N}_{x_1}([10 \ 5]^T, 0.01I_2)$. For this example, 100 Monte Carlo runs were simulated. In Figure 9 (left), the output z_t that corresponds to the values of h_2 that are nonquantized and the output y_t that corresponds to the measurements given by the sensor (for one of the Monte Carlo runs) are shown. In this figure, the level of quantization in the measurements can be observed. In Figure 9 (right), the MSE between the true and estimated state is shown. It was observed that the proposal presented in this paper, the Gaussian sum filter, yielded the most accurate estimation of h_1 , followed by the particle filter, Kalman filter, and quantized Kalman filter. In this example, $K = 50$ and 1000 particles were used to implement the Gaussian sum filter and particle filter, respectively.

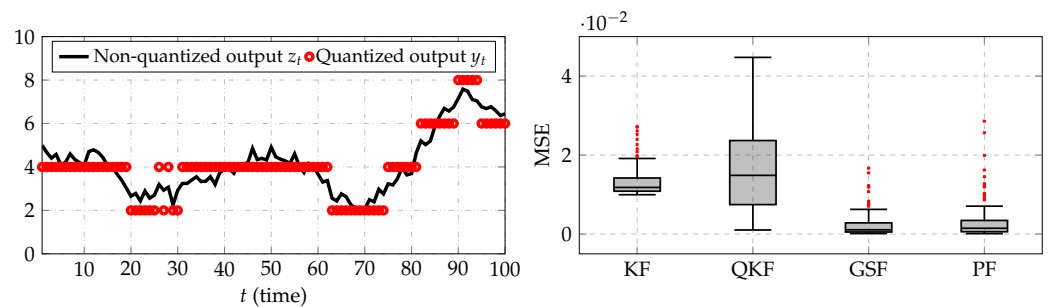


Figure 9. (Left) The quantized and nonquantized measurements; (right) the MSE between the true and estimated filtered state. KF, QKF, GSF, and PF stand for the Kalman filter, quantized Kalman filter, Gaussian sum filter, and particle filter, respectively.

In this example, a relatively high number of Gaussian components ($K = 50$) were required to obtain a good estimation of the filtering and smoothing distributions, and hence of the state. This produced an increment in the execution time since the Gaussian sum reduction algorithm needed more time to deal with a high number of Gaussian components in every iteration. This resulted in similar execution times for our proposed algorithm and the traditional particle filter. However, the execution time of the Gaussian sum filter and smoother was smaller than the execution time of the ground truth (particle filter and smoother with 20,000 particles).

5. Conclusions

In this paper, Gaussian sum filtering and smoothing algorithms for linear-time-invariant state-space systems with quantized output data were developed. An approximation of the integral equation that defines the probability mass function of the quantized data given the current state, $p(y_t|x_t)$, using Gaussian quadrature was considered. This approximation naturally yielded an explicit mathematical model with a GMM structure for this probability function. Using the approximation of $p(y_t|x_t)$ summarized in Theorem 1, it was possible to solve in closed form the general equations of filtering and smoothing to deal with quantized data. This fact allowed for a closed-form expression of the system state estimators given the quantized data $\hat{x}_{t|t}$ and $\hat{x}_{t|N}$. Via numerical simulations, it was shown that approximating $p(y_t|x_t)$ with a small number of Gaussian components was adequate, yielding an approximation comparable to the true filtering and smoothing PDFs given by the particle approach (using a large number of particles, namely 20,000 particles). This reduced number of Gaussian components allowed for a low computational load, especially when the system order increased. In addition, our results showed overall less computational load for our proposed techniques since the number of Gaussian components was considerably less than the number of particles used in particle filtering and smoothing.

The proposed Gaussian sum filtering and smoothing algorithms can be utilized, in principle, to develop system identification algorithms and control strategies having quantized output measurements.

Author Contributions: Conceptualization, A.L.C., R.A. and J.C.A.; methodology, A.L.C. and J.C.A.; software, A.L.C.; validation, R.C. and B.I.G.; formal analysis, A.L.C., R.A., R.C., B.I.G. and J.C.A.; investigation, A.L.C., R.A. and J.C.A.; resources, J.C.A.; writing—original draft preparation, A.L.C., R.A., R.C., B.I.G. and J.C.A.; writing—review and editing, A.L.C., R.C., B.I.G. and J.C.A.; visualization, R.C. and B.I.G.; supervision, J.C.A. All authors have read and agreed to the published version of the manuscript.

Funding: PIIC program of DGP at Universidad Técnica Federico Santa María No. 062/2018 and 035/2021. Grants ANID-Fondecyt 1211630 and 11201187, ANID-Basal Project FB0008 (AC3E). Chilean National Agency for Research and Development (ANID) Scholarship Program/Doctorado Nacional/2020-21202410.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not applicable to this article.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

PDF/PMF:	Probability density/mass function
GMM:	Gaussian mixture model
GSF/GSS:	Gaussian sum filter/smoothing
PF/PS:	Particle filter/smoothing
KF/KS:	Kalman filter/smoothing
QKF/QKS:	Quantized Kalman filter/smoothing
FLQ/ILQ:	Finite-/infinite-level quantizer
MSE:	Mean squared error
S_{red} :	Number of Gaussian components kept after the reduction procedure
GT:	Ground truth obtain by using the particle filter/smoothing with a large number of particles
K:	Order of the polynomials in the Gauss–Legendre quadrature

Appendix A. Technical Lemmata

Lemma A1. The PDF $\mathcal{N}_y(Cx + \mu, R)$ of the random variable $Y \in \mathbb{R}^p$ can be rewritten as follows:

$$\mathcal{N}_y(Cx + \mu, R) = \frac{1}{\sqrt{\det\{2\pi R\}}} \exp\left\{-\frac{1}{2}\left(x^T F x - 2G^T x + H\right)\right\}, \quad (\text{A1})$$

where $F = C^T R^{-1} C$, $G^T = (y - \mu)^T R^{-1} C$, and $H = (y - \mu)^T R^{-1} (y - \mu)$.

Proof. Directly expand the exponential argument and reorder the terms in the variable x . \square

Lemma A2. The PDF $\mathcal{N}_x(Aw + v, Q)$ of the random variable $X \in \mathbb{R}^n$ can be rewritten as follows:

$$\mathcal{N}_x(Aw + v, Q) = \frac{1}{\sqrt{\det\{2\pi Q\}}} \exp\left\{-\frac{1}{2}\left(x^T Q^{-1} x - 2J^T x + L\right)\right\}, \quad (\text{A2})$$

where $J^T = (Aw + v)^T Q^{-1}$ and $L = (Aw + v)^T Q^{-1} (Aw + v)$.

Proof. Directly expand the exponential argument and reorder the terms in the variable x . \square

Lemma A3. Consider the backward filter function $p(y_{t:N}|x_t)$ given in (42). Its GMM structure is given by:

$$p(y_{t:N}|x_t) = \sum_{k=1}^{S_{t|t}} \sigma_{t|t}^k \mathcal{N}_{x_t} \left(v_{t|t}^k, \Omega_{t|t}^k \right), \quad (\text{A3})$$

where $\sigma_{t|t}^k = \bar{\sigma}_{t|t}^k / \sum_{s=1}^{S_{t|t}} \bar{\sigma}_{t|t}^s$ is the normalized mixing weight and $v_{t|t}^k = (F_{t|t}^k)^{-1} G_{t|t}^k$ and $\Omega_{t|t}^k = (F_{t|t}^k)^{-1}$, are the mean vector and the covariance matrix, respectively, with:

$$\bar{\sigma}_{t|t}^k = \frac{\epsilon_{t|t}^k \lambda_{t|t}^k \exp \left\{ -\frac{1}{2} \left(H_{t|t}^k - G_{t|t}^{kT} (F_{t|t}^k)^{-1} G_{t|t}^k \right) \right\}}{(2\pi)^{-\frac{n}{2}} \sqrt{\det \{ F_{t|t}^k \}}}. \quad (\text{A4})$$

On the other hand, consider the GMM structure given by (54). Then, the backward filter form in (42) is obtained using the following:

$$\begin{aligned} S_{t|t} &= S_{red}, & \epsilon_{t|t}^k &= \delta_{t|t}^k, & \lambda_{t|t}^k &= \left(\det \{ 2\pi U_{t|t}^k \} \right)^{-1/2}, \\ F_{t|t}^k &= \left(U_{t|t}^k \right)^{-1}, & G_{t|t}^{kT} &= \left(z_{t|t}^k \right)^T \left(U_{t|t}^k \right)^{-1}, & H_{t|t}^k &= \left(z_{t|t}^k \right)^T \left(U_{t|t}^k \right)^{-1} \left(z_{t|t}^k \right). \end{aligned} \quad (\text{A5})$$

Proof. Directly by using Lemma A1. \square

Appendix B. Quantities to Perform the Gaussian Sum Filter

For each two-tuple (τ, ℓ) , where $\tau = 1, \dots, K$ and $\ell = 1, \dots, M_{t|t-1}$, let k be a new index, so that $k = (\ell - 1)K + \tau$. Then:

$$\gamma_{t|t}^k = \frac{\bar{\gamma}_{t|t}^k}{\sum_{s=1}^{M_{t|t}} \bar{\gamma}_{t|t}^s}, \quad (\text{A6})$$

$$\hat{x}_{t|t}^k = \hat{x}_{t|t-1}^\ell + K_t^\ell (\eta_t^\tau - \kappa_t^{\ell\tau}), \quad (\text{A7})$$

$$\Sigma_{t|t}^k = (I - K_t^\ell C) \Sigma_{t|t-1}^\ell, \quad (\text{A8})$$

are the weights, means, and covariance matrices of the measurement-update equation in the Gaussian sum filter, with the following definitions:

$$\bar{\gamma}_{t|t}^k = \zeta_t^\tau \gamma_{t|t-1}^\ell \mathcal{N}_{\eta_t^\tau} \left(\kappa_t^{\ell\tau}, V_t^\ell \right), \quad (\text{A9})$$

$$K_t^\ell = \Sigma_{t|t-1}^\ell C^T \left(V_t^\ell \right)^{-1}, \quad (\text{A10})$$

$$\kappa_t^{\ell\tau} = C \hat{x}_{t|t-1}^\ell + D u_t + \mu_t^\tau, \quad (\text{A11})$$

$$V_t^\ell = R + C \Sigma_{t|t-1}^\ell C^T, \quad (\text{A12})$$

where ζ_t^τ , η_t^τ , and μ_t^τ are defined in Table 1, and:

$$\gamma_{t+1|t}^k = \gamma_{t|t}^k, \quad (\text{A13})$$

$$\hat{x}_{t+1|t}^k = A \hat{x}_{t|t}^k + B u_t, \quad (\text{A14})$$

$$\Sigma_{t+1|t}^k = Q + A \Sigma_{t|t}^k A^T, \quad (\text{A15})$$

are the weights, means, and covariance matrices of the time-update equation in the Gaussian sum filter.

Appendix C. Quantities to Perform the Backward Filter

For each two-tuple (τ, ℓ) , where $\tau = 1, \dots, K$ and $\ell = 1, \dots, S_{t|t+1}$, let k be a new index, so that $k = (\ell - 1)K + \tau$. Then:

$$\epsilon_{t|t}^k = \zeta_t^\tau \epsilon_{t|t+1}^\ell, \quad (\text{A16})$$

$$\lambda_{t|t}^k = (\det\{2\pi R\})^{-1/2} \lambda_{t|t+1}^\ell, \quad (\text{A17})$$

$$\theta_t^\tau = \eta_t^\tau - Du_t - \mu_t^\tau, \quad (\text{A18})$$

$$F_{t|t}^k = F_{t|t+1}^\ell + C^T R^{-1} C, \quad (\text{A19})$$

$$G_{t|t}^{kT} = G_{t|t+1}^{\ell T} + \theta_t^{\tau T} R^{-1} C, \quad (\text{A20})$$

$$H_{t|t}^k = H_{t|t+1}^\ell + \theta_t^{\tau T} R^{-1} \theta_t^\tau. \quad (\text{A21})$$

are the quantities needed to compute the backward-measurement-update equation in the backward filter, where ζ_t^τ , η_t^τ , and μ_t^τ are defined in Table 1, and:

$$\epsilon_{t|t+1}^k = \epsilon_{t+1|t+1}^k, \quad (\text{A22})$$

$$\lambda_{t|t+1}^k = \left(\det\{Q\} \det\{F_{qk}\} \right)^{-1/2} \lambda_{t+1|t+1}^k, \quad (\text{A23})$$

$$F_{t|t+1}^k = A^T M_{qk} A, \quad (\text{A24})$$

$$G_{t|t+1}^{kT} = G_{t+1|t+1}^{kT} F_{qk}^{-1} Q^{-1} A - u_t^T B^T M_{qk} A, \quad (\text{A25})$$

$$H_{t|t+1}^k = H_{t+1|t+1}^k - G_{t+1|t+1}^{kT} F_{qk}^{-1} G_{t+1|t+1}^k + u_t^T B^T M_{qk} B u_t - 2u_t^T B^T Q^{-1} F_{qk}^{-1} G_{t+1|t+1}^k, \quad (\text{A26})$$

are the quantities needed to compute the backward prediction equation in the backward filter, where $F_{qk} = F_{t+1|t+1}^k + Q^{-1}$ and $M_{qk} = Q^{-1} - Q^{-1} F_{qk}^{-1} Q^{-1}$.

Appendix D. Quantities to Perform the Gaussian Sum Smoother

For each two-tuple (τ, ℓ) , where $\tau = 1, \dots, M_{t|t-1}$ and $\ell = 1, \dots, S_{\text{red}}$, let k be a new index, so that $k = (\ell - 1)M_{t|t-1} + \tau$. Then:

$$\epsilon_{t|N}^k = \frac{\bar{\epsilon}_{t|N}^k}{\sum_{s=1}^{S_{t|N}} \bar{\epsilon}_{t|N}^s}, \quad (\text{A27})$$

$$\hat{x}_{t|N}^k = (L_t^{\ell\tau})^{-1} \rho_t^{\ell\tau}, \quad (\text{A28})$$

$$\Sigma_{t|N}^k = (L_t^{\ell\tau})^{-1}, \quad (\text{A29})$$

are the weights, means, and covariance matrices of the smoothing PDF $p(x_t | y_{1:N})$, where:

$$\bar{\epsilon}_{t|N}^k = \frac{\gamma_{t|t-1}^\tau \delta_{t|t}^\ell \exp\left\{-\frac{1}{2}[\phi_{1t}^\ell + \phi_{2t}^\tau - \phi_{3t}^{\ell\tau}]\right\}}{(2\pi)^{\frac{n}{2}} \sqrt{\det\{L_t^{\ell\tau}\} \det\{U_{t|t}^\ell\} \det\{\Sigma_{t|t-1}^\tau\}}}, \quad (\text{A30})$$

$$\rho_t^{\ell\tau} = \left[\left(U_{t|t}^\ell \right)^{-1} z_{t|t}^\ell + \left(\Sigma_{t|t-1}^\tau \right)^{-1} \hat{x}_{t|t-1}^\tau \right], \quad (\text{A31})$$

$$L_t^{\ell\tau} = \left[\left(U_{t|t}^\ell \right)^{-1} + \left(\Sigma_{t|t-1}^\tau \right)^{-1} \right], \quad (\text{A32})$$

$$\phi_{3t}^{\ell\tau} = \left(\rho_t^{\ell\tau} \right)^T \left(L_t^{\ell\tau} \right)^{-1} \left(\rho_t^{\ell\tau} \right), \quad (\text{A33})$$

$$\phi_{1t}^\ell = \left(z_{t|t}^\ell \right)^T \left(U_{t|t}^\ell \right)^{-1} \left(z_{t|t}^\ell \right), \quad (\text{A34})$$

$$\phi_{2t}^\tau = \left(\hat{x}_{t|t-1}^\tau \right)^T \left(\Sigma_{t|t-1}^\tau \right)^{-1} \left(\hat{x}_{t|t-1}^\tau \right), \quad (\text{A35})$$

$\gamma_{t|t-1}^\tau$, $\hat{x}_{t|t-1}^\tau$ and $\Sigma_{t|t-1}^\tau$ are obtained from the time-update step of Theorem 2 (34), and $\delta_{t|t}^\ell$, $z_{t|t}^\ell$, and $U_{t|t}^\ell$ are obtained from the reduced measurement-update step of the backward-filtering algorithm in (54).

Appendix E. Quantities to Compute $p(x_{t+1}, x_t | y_{1:N})$

For each two-tuple (τ, ℓ) , where $\tau = 1, \dots, M_{t|t}$ and $\ell = 1, \dots, S_{t+1|t+1}$, let k be a new index, so that $k = (\ell - 1)M_{t|t} + \tau$. Then:

$$\alpha_k = \frac{\bar{\alpha}_k}{\sum_{s=1}^{S_{t+1|t+1}} \bar{\alpha}_s}, \quad (\text{A36})$$

$$\hat{\lambda}_{t|N}^k = (\mathcal{F}_t^{\ell\tau})^{-1} \mathcal{G}_t^{\ell\tau}, \quad (\text{A37})$$

$$E_{t|N}^k = (\mathcal{F}_t^{\ell\tau})^{-1}, \quad (\text{A38})$$

are the weights, means, and covariance matrices of the joint PDF $p(x_{t+1}, x_t | y_{1:N})$, where:

$$\bar{\alpha}_k = \frac{\gamma_{t|t}^\tau \epsilon_{t+1|t+1}^\ell \lambda_{t+1|t+1}^\ell S_t^{\ell\tau}}{\sqrt{\det\{Q\} \det\{\mathcal{F}_t^{\ell\tau}\} \det\{\Sigma_{t|t}^\tau\}}}, \quad (\text{A39})$$

$$S_t^{\ell\tau} = \exp\left\{-\frac{1}{2} \left(\mathcal{H}_t^{\ell\tau} - \mathcal{G}_t^{\ell\tau T} \left(\mathcal{F}_t^{\ell\tau} \right)^{-1} \mathcal{G}_t^{\ell\tau} \right)\right\}, \quad (\text{A40})$$

$$\mathcal{F}_t^{\ell\tau} = \begin{bmatrix} Q^{-1} + F_{t+1|t+1}^\ell & -Q^{-1}A \\ -A^T Q^{-1} & \left(\Sigma_{t|t}^\tau \right)^{-1} + A^T Q^{-1}A \end{bmatrix}, \quad (\text{A41})$$

$$\mathcal{G}_t^{\ell\tau T} = \left[G_{t+1|t+1}^{\ell T} + u_t^T B^T Q^{-1} \quad J_t^{\tau T} - u_t^T B^T Q^{-1}A \right], \quad (\text{A42})$$

$$\mathcal{H}_t^{\ell\tau} = H_{t+1|t+1}^\ell + L_t^\tau + u_t^T B^T Q^{-1} B u_t, \quad (\text{A43})$$

with:

$$J_t^{\tau T} = \left(\hat{x}_{t|t}^\tau \right)^T \left(\Sigma_{t|t}^\tau \right)^{-1}, \quad (\text{A44})$$

$$L_t^\tau = \left(\hat{x}_{t|t}^\tau \right)^T \left(\Sigma_{t|t}^\tau \right)^{-1} \left(\hat{x}_{t|t}^\tau \right), \quad (\text{A45})$$

$\gamma_{t|t}^\tau$, $\hat{x}_{t|t}^\tau$, and $\Sigma_{t|t}^\tau$ are obtained from the measurement-update step in Theorem 2, and

$\epsilon_{t+1|t+1}^\ell$, $\lambda_{t+1|t+1}^\ell$, $F_{t+1|t+1}^\ell$, $G_{t+1|t+1}^{\ell T}$ and $H_{t+1|t+1}^\ell$ are obtained from the reduced backward-measurement-update step in Theorem 3.

References

1. Kalman, R.E. A New Approach to Linear Filtering and Prediction Problems. *Trans. ASME-J. Basic Eng.* **1960**, *82*, 35–45. [[CrossRef](#)]
2. Kamen, E.W.; Su, J.K. *Introduction to Optimal Estimation*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 1999.
3. Anderson, B.D.O.; Moore, J.B. *Optimal Filtering*; Prentice-Hall, Inc.: Hoboken, NJ, USA, 1979.
4. Leong, A.S.; Dey, S.; Quevedo, D.E. Transmission scheduling for remote state estimation and control with an energy harvesting sensor. *Automatica* **2018**, *91*, 54–60. [[CrossRef](#)]
5. Liang, H.; Guo, X.; Pan, Y.; Huang, T. Event-Triggered Fuzzy Bipartite Tracking Control for Network Systems Based on Distributed Reduced-Order Observers. *IEEE Trans. Fuzzy Syst.* **2021**, *29*, 1601–1614. [[CrossRef](#)]
6. Liu, L.; Gao, T.; Liu, Y.J.; Tong, S.; Chen, C.L.P.; Ma, L. Time-varying IBLFs-based adaptive control of uncertain nonlinear systems with full state constraints. *Automatica* **2021**, *129*, 109595. [[CrossRef](#)]
7. Liu, L.; Liu, Y.J.; Chen, A.; Tong, S.; Chen, C.L.P. Integral Barrier Lyapunov function-based adaptive control for switched nonlinear systems. *Sci. China Inf. Sci.* **2020**, *63*, 132203. [[CrossRef](#)]
8. Gibson, S.; Ninness, B. Robust maximum-likelihood estimation of multivariable dynamic systems. *Automatica* **2005**, *41*, 1667–1682. [[CrossRef](#)]
9. Agüero, J.C.; Tang, W.; Yuz, J.I.; Delgado, R.; Goodwin, G.C. Dual time–frequency domain system identification. *Automatica* **2012**, *48*, 3031–3041. [[CrossRef](#)]
10. Kaltiokallio, O.; Hostettler, R.; Yiğitler, H.; Valkama, M. Unsupervised Learning in RSS-Based DFLT Using an EM Algorithm. *Sensors* **2021**, *21*, 5549. [[CrossRef](#)] [[PubMed](#)]
11. Zhao, J.; Netto, M.; Huang, Z.; Yu, S.S.; Gómez-Expósito, A.; Wang, S.; Kamwa, I.; Akhlaghi, S.; Mili, L.; Terzija, V.; et al. Roles of Dynamic State Estimation in Power System Modeling, Monitoring and Operation. *IEEE Trans. Power Syst.* **2021**, *36*, 2462–2472. [[CrossRef](#)]
12. Ji, X.; Yin, Z.; Zhang, Y.; Wang, M.; Zhang, X.; Zhang, C.; Wang, D. Real-time robust forecasting-aided state estimation of power system based on data-driven models. *Int. J. Electr. Power Energy Syst.* **2021**, *125*, 106412. [[CrossRef](#)]
13. Bonvini, M.; Sohn, M.D.; Granderson, J.; Wetter, M.; Piette, M.A. Robust on-line fault detection diagnosis for HVAC components based on nonlinear state estimation techniques. *Appl. Energy* **2014**, *124*, 156–166. [[CrossRef](#)]
14. Nemati, F.; Safavi Hamami, S.M.; Zemouche, A. A nonlinear observer-based approach to fault detection, isolation and estimation for satellite formation flight application. *Automatica* **2019**, *107*, 474–482. [[CrossRef](#)]
15. Jeong, H.; Park, B.; Park, S.; Min, H.; Lee, S. Fault detection and identification method using observer-based residuals. *Reliab. Eng. Syst. Saf.* **2019**, *184*, 27–40. [[CrossRef](#)]
16. Noshad, Z.; Javaid, N.; Saba, T.; Wadud, Z.; Saleem, M.Q.; Alzahrani, M.E.; Sheta, O.E. Fault Detection in Wireless Sensor Networks through the Random Forest Classifier. *Sensors* **2019**, *19*, 1568. [[CrossRef](#)]
17. Huang, C.; Shen, B.; Zou, L.; Shen, Y. Event-Triggering State and Fault Estimation for a Class of Nonlinear Systems Subject to Sensor Saturations. *Sensors* **2021**, *21*, 1242. [[CrossRef](#)]
18. Corbetta, M.; Sbarufatti, C.; Giglio, M.; Todd, M.D. Optimization of nonlinear, non-Gaussian Bayesian filtering for diagnosis and prognosis of monotonic degradation processes. *Mech. Syst. Signal Process.* **2018**, *104*, 305–322. [[CrossRef](#)]
19. Ahwiadi, M.; Wang, W. An Adaptive Particle Filter Technique for System State Estimation and Prognosis. *IEEE Trans. Instrum. Meas.* **2020**, *69*, 6756–6765. [[CrossRef](#)]
20. Ding, D.; Han, Q.L.; Ge, X.; Wang, J. Secure State Estimation and Control of Cyber-Physical Systems: A Survey. *IEEE Trans. Syst. Man Cybern. Syst.* **2021**, *51*, 176–190. [[CrossRef](#)]
21. McLaughlin, D. An integrated approach to hydrologic data assimilation: Interpolation, smoothing, and filtering. *Adv. Water Resour.* **2002**, *25*, 1275–1286. [[CrossRef](#)]
22. Cosme, E.; Verron, J.; Bresseur, P.; Blum, J.; Auroux, D. Smoothing problems in a Bayesian framework and their linear Gaussian solutions. *Mon. Weather Rev.* **2012**, *140*, 683–695. [[CrossRef](#)]
23. Stone, L.D.; Streit, R.L.; Corwin, T.L.; Bell, K.L. *Bayesian Multiple Target Tracking*; Artech House: Norwood, MA, USA, 2013.
24. Schizas, I.D.; Giannakis, G.B.; Roulmetiotis, S.I.; Ribeiro, A. Consensus in Ad Hoc WSNs With Noisy Links—Part II: Distributed Estimation and Smoothing of Random Signals. *IEEE Trans. Signal Process.* **2008**, *56*, 1650–1666. [[CrossRef](#)]
25. Liu, J.; Liu, Y.; Dong, K.; Ding, Z.; He, Y. A Novel Distributed State Estimation Algorithm with Consensus Strategy. *Sensors* **2019**, *19*, 2134. [[CrossRef](#)] [[PubMed](#)]
26. Han, Y.; Cui, M.; Liu, S. Optimal Sensor and Relay Nodes Power Scheduling for Remote State Estimation with Energy Constraint. *Sensors* **2020**, *20*, 1073. [[CrossRef](#)] [[PubMed](#)]
27. Chiang, K.W.; Duong, T.T.; Liao, J.K.; Lai, Y.C.; Chang, C.C.; Cai, J.M.; Huang, S.C. On-Line Smoothing for an Integrated Navigation System with Low-Cost MEMS Inertial Sensors. *Sensors* **2012**, *12*, 17372–17389. [[CrossRef](#)] [[PubMed](#)]
28. Rostami Shahrbabaki, M.; Safavi, A.A.; Papageorgiou, M.; Papamichail, I. A data fusion approach for real-time traffic state estimation in urban signalized links. *Transp. Res. Part C Emerg. Technol.* **2018**, *92*, 525–548. [[CrossRef](#)]
29. Ahmed, A.; Naqvi, S.A.A.; Watling, D.; Ngoduy, D. Real-Time Dynamic Traffic Control Based on Traffic-State Estimation. *Transp. Res. Rec.* **2019**, *2673*, 584–595. [[CrossRef](#)]

30. Schreiter, T.; van Lint, H.; Treiber, M.; Hoogendoorn, S. Two fast implementations of the Adaptive Smoothing Method used in highway traffic state estimation. In Proceedings of the 13th International IEEE Conference on Intelligent Transportation Systems, Funchal, Portugal, 19–22 September 2010; pp. 1202–1208.
31. Särkkä, S. *Bayesian Filtering and Smoothing*; Cambridge University Press: Cambridge, UK, 2013; Volume 3.
32. Widrow, B.; Kollár, I. *Quantization Noise: Roundoff Error in Digital Computation, Signal Processing, Control, and Communications*; Cambridge University Press: Cambridge, UK, 2008.
33. Msechu, E.J.; Giannakis, G.B. Sensor-Centric Data Reduction for Estimation With WSNs via Censoring and Quantization. *IEEE Trans. Signal Process.* **2012**, *60*, 400–414. [[CrossRef](#)]
34. Wang, L.Y.; Yin, G.G.; Zhang, J. System identification using binary sensors. *IEEE Trans. Automat. Contr.* **2003**, *48*, 1892–1907. [[CrossRef](#)]
35. Curry, R.E. *Estimation and Control with Quantized Measurements*; MIT Press: Cambridge, MA, USA, 1970.
36. Liu, S.; Wang, Z.; Hu, J.; Wei, G. Protocol-based extended Kalman filtering with quantization effects: The Round-Robin case. *Int. J. Robust Nonlinear Control* **2020**, *30*, 7927–7946. [[CrossRef](#)]
37. Malyavej, V.; Savkin, A.V. The problem of optimal robust Kalman state estimation via limited capacity digital communication channels. *Syst. Control Lett.* **2005**, *54*, 283–292. [[CrossRef](#)]
38. Farhadi, A.; Charalambous, C.D. Stability and reliable data reconstruction of uncertain dynamic systems over finite capacity channels. *Automatica* **2010**, *46*, 889–896. [[CrossRef](#)]
39. Silva, E.I.; Agüero, J.C.; Goodwin, G.C.; Lau, K.; Wang, M. The SNR Approach to Networked Control. In *The Control Handbook: Control System Applications*; Levine, W.S., Ed.; CRC Press: Boca Raton, FL, USA, 2011; Chapter 25, pp. 1–26.
40. Godoy, B.I.; Agüero, J.C.; Carvajal, R.; Goodwin, G.C.; Yuz, J.I. Identification of sparse FIR systems using a general quantisation scheme. *Int. J. Control* **2014**, *87*, 874–886. [[CrossRef](#)]
41. Li, Z.M.; Chang, X.H.; Yu, L. Robust quantized H_∞ filtering for discrete-time uncertain systems with packet dropouts. *Appl. Math. Comput.* **2016**, *275*, 361–371. [[CrossRef](#)]
42. Li, S.; Sauter, D.; Xu, B. Fault isolation filter for networked control system with event-triggered sampling scheme. *Sensors* **2011**, *11*, 557–572. [[CrossRef](#)] [[PubMed](#)]
43. Zhang, L.; Liang, H.; Sun, Y.; Ahn, C.K. Adaptive Event-Triggered Fault Detection Scheme for Semi-Markovian Jump Systems With Output Quantization. *IEEE Trans. Syst. Man, Cybern. Syst.* **2021**, *51*, 2370–2381. [[CrossRef](#)]
44. Zhang, X.; Han, Q.; Ge, X.; Ding, D.; Ding, L.; Yue, D.; Peng, C. Networked control systems: A survey of trends and techniques. *IEEE/CAA J. Autom. Sin.* **2020**, *7*, 1–17. [[CrossRef](#)]
45. Goodwin, G.C.; Haimovich, H.; Quevedo, D.E.; Welsh, J.S. A moving horizon approach to Networked Control system design. *IEEE Trans. Autom. Control* **2004**, *49*, 1427–1445. [[CrossRef](#)]
46. Gustafsson, F.; Karlsson, R. Statistical results for system identification based on quantized observations. *Automatica* **2009**, *45*, 2794–2801. [[CrossRef](#)]
47. Wang, L.Y.; Yin, G.G.; Zhang, J.; Zhao, Y. *System Identification with Quantized Observations*; Springer: Berlin/Heidelberg, Germany, 2010.
48. Marelli, D.E.; Godoy, B.I.; Goodwin, G.C. A scenario-based approach to parameter estimation in state-space models having quantized output data. In Proceedings of the 49th IEEE Conference on Decision and Control (CDC), Atlanta, GA, USA, 15–17 December 2010; pp. 2011–2016.
49. Leong, A.S.; Dey, S.; Nair, G.N. Quantized Filtering Schemes for Multi-Sensor Linear State Estimation: Stability and Performance Under High Rate Quantization. *IEEE Trans. Signal Process.* **2013**, *61*, 3852–3865. [[CrossRef](#)]
50. Li, D.; Kar, S.; Alsaadi, F.E.; Dobaie, A.M.; Cui, S. Distributed Kalman Filtering with Quantized Sensing State. *IEEE Trans. Signal Process.* **2015**, *63*, 5180–5193. [[CrossRef](#)]
51. Rana, M.M.; Li, L. An Overview of Distributed Microgrid State Estimation and Control for Smart Grids. *Sensors* **2015**, *15*, 4302–4325. [[CrossRef](#)]
52. Chang, X.H.; Liu, Y. Robust H_∞ Filtering for Vehicle Sideslip Angle With Quantization and Data Dropouts. *IEEE Trans. Veh. Technol.* **2020**, *69*, 10435–10445. [[CrossRef](#)]
53. Curiaç, D. Towards wireless sensor, actuator and robot networks: Conceptual framework, challenges and perspectives. *J. Netw. Comput. Appl.* **2016**, *63*, 14–23. [[CrossRef](#)]
54. Allik, B.; Piovoso, M.J.; Zurakowski, R. Recursive estimation with quantized and censored measurements. In Proceedings of the 2016 American Control Conference (ACC), Boston, MA, USA, 6–8 July 2016; pp. 5130–5135.
55. Zhou, Y.; Li, J.; Wang, D. Unscented Kalman Filtering based quantized innovation fusion for target tracking in WSN with feedback. In Proceedings of the 2009 International Conference on Machine Learning and Cybernetics, Baoding, China, 12–15 July 2009; Volume 3, pp. 1457–1463.
56. Wigren, T. Approximate gradients, convergence and positive realness in recursive identification of a class of non-linear systems. *Int. J. Adapt. Control Signal Process.* **1995**, *9*, 325–354. [[CrossRef](#)]
57. Gustafsson, F.; Karlsson, R. Estimation based on Quantized Observations. *IFAC Proc. Vol.* **2009**, *42*, 78–83. [[CrossRef](#)]
58. Cedeño, A.L.; Alborno, R.; Carvajal, R.; Godoy, B.I.; Agüero, J.C. On Filtering Methods for State-Space Systems having Binary Output Measurements. *IFAC-PapersOnLine* **2021**, *54*, 815–820. [[CrossRef](#)]

59. Gersho, A.; Gray, R.M. *Vector Quantization and Signal Compression*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2012; Volume 159.
60. Frühwirth, S.; Celeux, G.; Robert, C.P. *Handbook of Mixture Analysis*; CRC Press: Boca Raton, FL, USA, 2019.
61. Lo, J. Finite-dimensional sensor orbits and optimal nonlinear filtering. *IEEE Trans. Inf. Theory* **1972**, *18*, 583–588. [[CrossRef](#)]
62. Cedeño, A.L.; Orellana, R.; Carvajal, R.; Agüero, J.C. EM-based identification of static errors-in-variables systems utilizing Gaussian Mixture models. *IFAC-PapersOnLine* **2020**, *53*, 863–868. [[CrossRef](#)]
63. Orellana, R.; Carvajal, R.; Escárate, P.; Agüero, J.C. On the Uncertainty Identification for Linear Dynamic Systems Using Stochastic Embedding Approach with Gaussian Mixture Models. *Sensors* **2021**, *21*, 3837. [[CrossRef](#)] [[PubMed](#)]
64. Kitagawa, G. The two-filter formula for smoothing and an implementation of the Gaussian-sum smoother. *Ann. Inst. Stat. Math.* **1994**, *46*, 605–623. [[CrossRef](#)]
65. Kitagawa, G.; Gersch, W. *Smoothness Priors Analysis of Time Series*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 1996; Volume 116.
66. Karlsson, R.; Gustafsson, F. Particle filtering for quantized sensor information. In Proceedings of the 2005 13th European Signal Processing Conference, Antalya, Turkey, 4–8 September 2005; pp. 1–4.
67. DeGroot, M.H. *Optimal Statistical Decisions*; Wiley Classics Library, Wiley: Hoboken, NJ, USA, 2005.
68. Borovkov, A.A. *Probability Theory*; Springer: Berlin/Heidelberg, Germany, 2013.
69. Cohen, H. *Numerical Approximation Methods*; Springer: Berlin/Heidelberg, Germany, 2011.
70. Arasaratnam, I.; Haykin, S.; Elliott, R.J. Discrete-time nonlinear filtering algorithms using Gauss–Hermite quadrature. *Proc. IEEE* **2007**, *95*, 953–977. [[CrossRef](#)]
71. Arasaratnam, I.; Haykin, S. Cubature kalman smoothers. *Automatica* **2011**, *47*, 2245–2250. [[CrossRef](#)]
72. Runnalls, A.R. Kullback–Leibler approach to Gaussian mixture reduction. *IEEE Trans. Aerosp. Electron. Syst.* **2007**, *43*, 989–999. [[CrossRef](#)]
73. Balenzuela, M.P.; Dahlin, J.; Bartlett, N.; Wills, A.G.; Renton, C.; Ninness, B. Accurate Gaussian Mixture Model Smoothing using a Two-Filter Approach. In Proceedings of the 2018 IEEE Conference on Decision and Control (CDC), Miami, FL, USA, 17–19 December 2018; pp. 694–699.
74. Reddy, H.P.; Narasimhan, S.; Bhallamudi, S.M. Simulation and state estimation of transient flow in gas pipeline networks using a transfer function model. *Ind. Eng. Chem. Res.* **2006**, *45*, 3853–3863. [[CrossRef](#)]
75. Chang, S.I.; Wang, S.J.; Lin, M.S. The Role of Simulation in Control System Design/Modification. In *Nuclear Simulation*; Springer: Berlin/Heidelberg, Germany, 1987; pp. 205–222.
76. Anderson, B.D.O.; Moore, J.B. *Optimal Control: Linear Quadratic Methods*; Courier Corporation: Massachusetts, MA, USA, 2007.
77. Gómez, J.C.; Sad, G.D. A State Observer from Multilevel Quantized Outputs. In Proceedings of the 2020 Argentine Conference on Automatic Control (AADECA), Buenos Aires, Argentina, 28–30 October 2020; pp. 1–6.
78. Goodwin, G.C.; Graebe, S.F.; Salgado, M.E. *Control System Design*; Prentice Hall: Hoboken, NJ, USA, 2001; Volume 240.