On Entropy Approximation for Gaussian Mixture Random Vectors

Marco F. Huber, Tim Bailey, Hugh Durrant-Whyte, and Uwe D. Hanebeck

Abstract— **For many practical probability density representations such as for the widely used Gaussian mixture densities, an analytic evaluation of the differential entropy is not possible and thus, approximate calculations are inevitable. For this purpose, the first contribution of this paper deals with a novel entropy approximation method for Gaussian mixture random vectors, which is based on a component-wise Taylor-series expansion of the logarithm of a Gaussian mixture and on a splitting method of Gaussian mixture components. The employed order of the Taylor-series expansion and the number of components used for splitting allows balancing between accuracy and computational demand. The second contribution is the determination of meaningful and efficiently to calculate lower and upper bounds of the entropy, which can be also used for approximation purposes. In addition, a refinement method for the more important upper bound is proposed in order to approach the true entropy value.**

I. Introduction

Differential entropy as a measure of uncertainty extends the classical entropy introduced by Shannon [1] to continuous random vectors. It has a value within $[-\infty, +\infty]$ and achieves a minimum when the random vector comprises no uncertainty (i.e., the density function is a Dirac delta distribution) and approaches a maximum as the random vector becomes uniformly distributed. This entropy measure has applications in its own right, such as for parameter estimation [2], but it is also central to the computation of other measures, such as mutual information, which is a measure of dependency between two random variables. Mutual information is also widely used, with applications including capacity of communication channels [3], sensor management for multitarget tracking [4], image registration [5], and many others.

For Gaussian density functions, entropy has an analytic solution proportional to the determinant of the covariance matrix. The Gaussian function also has many other computational advantages, which is one of the reasons for its dominance in data fusion applications. However, many real-world systems possess strongly non-Gaussian uncertainties, and a Gaussian approximation is sometimes inadequate for highly skewed or multimodal distributions. Gaussian mixture densities are a popular representation of non-Gaussian densities. They are a universal function approximator in that, given a sufficient number of components, they can approximate any smooth function to arbitrary accuracy [6]. They also tend to be a useful representation in practice for multivariate densities.

However, there is no known closed-form solution to differential entropy for Gaussian mixtures. There exist several approximations in the literature, including loose upper and lower bounds, but the only existent approximation that can be demonstrated to converge to the true entropy relies on expensive random sampling methods. Other approximations offer either very loose bounds or can be shown to deviate from the true entropy in an arbitrary fashion and hence, are of limited usefulness.

In this paper, we present a novel approximation to differential entropy for Gaussian mixture random vectors based on Taylor-series expansions. For each Gaussian mixture component, a Taylor-series expansion of the logarithm of the Gaussian mixture is determined as described in Section IV, which facilitates an analytical evaluation of the entropy measure. Additionally, a splitting technique for Gaussian densities is employed in Section IV-B in order to avoid Gaussian components with high variance, which would require computationally expensive higher order expansion terms. Through the use of higher-order terms or component splitting it is possible to obtain an entropy approximation, which is of practical usefulness and versatile applicability as it

- permits a tradeoff between computational demand and accuracy
- is deterministically evaluable,
- makes no assumptions or restrictions on the structure of the Gaussian mixture.

This is an improvement over existing fixed-cost approximations that can be found in the literature (see Section III) as none fulfills all these requirements.

We also discuss cheap calculation of lower and upper bounds for differential entropy (see Section V). Besides the application of such bounds to entropy approximation and optimization, they can be also used in combination with the proposed Taylor-series expansion and splitting based approximation scheme in order to ensure meaningful approximation results. Especially for the theoretically important upper bound, a refinement algorithm based on component clustering is proposed in Section V-C, which allows calculating significantly tighter upper bounds. The performance of the proposed methods is demonstrated by means of simulations in Section VI, while in Section VII conclusions and an outlook to future work are given.

M. F. Huber and U. D. Hanebeck are with the Intelligent Sensor-Actuator-Systems Laboratory (ISAS), Institute of Computer Science and Engineering, Universität Karlsruhe (TH), Germany {marco.huber|uwe.hanebeck}@ieee.org

T. Bailey and H. Durrant-Whyte are with ARC Centre of Excellence in Autonomous Systems (CAS), The University of Sydney, Australia {tbailey|hugh}@acfr.usyd.edu.au

This work was partially supported by the German Research Foundation (DFG) within the Research Training Group GRK 1194 "Self-organizing Sensor-Actuator-Networks".

II. Problem Formulation

For a continuous-valued random vector $\underline{\boldsymbol{x}} \in \mathbb{R}^N$ with probability density function $f(\underline{x})$, the differential entropy is defined as

$$H(\underline{\boldsymbol{x}}) = \mathrm{E}\{-\log f(\underline{\boldsymbol{x}})\} = -\int_{\mathbb{R}^N} f(\underline{x}) \cdot \log f(\underline{x}) \, \mathrm{d}\underline{x} \ . \quad (1)$$

As the entropy is a measure of the degree of uncertainty the random vector $\underline{\boldsymbol{x}}$ comprises, it is utilized in many engineering applications from the field of nonlinear estimation, fusion, and control. Especially in planning tasks like sensor placement and scheduling for sensor networks, optimizing objective functions based on the entropy is crucial.

Thanks to their universal approximation property [6], Gaussian mixtures are a very common representation of the density function $f(\underline{x})$ in those tasks, i.e., $f(\underline{x})$ is given by the Gaussian mixture

$$f(\underline{x}) = \sum_{i=1}^{L} \omega_i \cdot \mathcal{N}(\underline{x}; \underline{\mu}_i, \mathbf{C}_i) \ ,$$

where ω_i are non-negative weighting coefficients with $\sum_i \omega_i = 1$ and $\mathcal{N}(\underline{x}; \underline{\mu}, \mathbf{C})$ is a Gaussian density with mean vector $\underline{\mu}$ and covariance matrix $\mathbf{C}$.

However, the entropy generally cannot be calculated in closed form for Gaussian mixtures due to the logarithm of a sum of exponential functions. Except for the special case of a single Gaussian density, where the entropy is

$$H(\underline{\boldsymbol{x}}) = \frac{1}{2} \log\left((2\pi e)^N |\mathbf{C}|\right) \ , \quad (2)$$

an approximate solution for (1) has to be applied. It is worth mentioning that (2) provides an upper bound for all Gaussian mixture random vectors with the same covariance $\mathbf{C}$ as in (2).

The following sections are concerned with the derivation of a novel entropy approximation scheme. In order to provide a practical and versatile entropy approximation, the novel scheme has to satisfy the requirements mentioned in Section I. Subsequently, novel computationally cheap lower and upper bounds of the true entropy values are determined, which also can be utilized as approximate entropy values.

In order to ease the discussion, in the following sections the notion

$$H(\underline{\boldsymbol{x}}) = -\int_{\mathbb{R}^N} f(\underline{x}) \cdot \log g(\underline{x}) \, \mathrm{d}\underline{x} \quad (3)$$

is used for the entropy, with $f(\underline{x}) = g(\underline{x})$. This allows differentiating between the Gaussian mixture $g(\underline{x})$ that is affected by the logarithm and the Gaussian mixture $f(\underline{x})$ that is not argument of the logarithm.

III. Prior Work

The literature provides many methods for an approximate calculation of the entropy for Gaussian mixture random vectors. As we will point out in this section, those methods typically do not (completely) satisfy the aforementioned requirements.

One of the most straightforward ways to approximate (3) results from employing the closed-form solution for a single Gaussian [7]. Here, $g(\underline{x})$ is replaced by the Gaussian density that exactly captures the first two moments of $f(\underline{x})$. Although this method is very efficient, it does not converge to the exact solution. However, since approximating a Gaussian mixture by a single Gaussian exactly capturing the first two moments [8], this method provides an (albeit very loose) upper bound approximation to the entropy.

The only entropy approximation method so far that generally converges to the true entropy is given by Monte Carlo sampling. Here, the Gaussian mixture $f(\underline{x})$ in (3) is represented by a set of samples drawn i.i.d. from $f(\underline{x})$, which allows a point-wise evaluation of the logarithm term in (3). According to the law of large numbers, this approximation converges to the true entropy value as the number of samples goes to infinity. However, a relatively large number of samples has to be used in order to obtain a good approximation, which in turn is computationally demanding. Since randomization is used, no deterministic approximation is provided, which complicates comparison and precludes classical optimization techniques like gradient descents for entropy minimization.

Deterministic sampling instead allows the use of far less sample points for a specific approximation quality. This is the idea the entropy approximation proposed in [9] is based on. By employing the unscented transform (see e.g. [10]) each Gaussian component of $f(\underline{x})$ in (3) is replaced by a set of so-called sigma points. This allows a point-wise evaluation of the logarithm, which is computationally efficient. However, in contrast to Monte Carlo sampling, convergence is no longer guaranteed as the number of sigma-points is constant.

In [11], [12], a deterministic approximation is developed by replacing (1) with the squared integral difference between $f(\underline{x})$ and a uniform density. This method can be regarded as linearizing (1) with a second-order Taylor-series expansion around the uniform density. This method turns out to be computationally demanding and often inaccurate. Furthermore, it is only applicable for the special case of Gaussian mixtures with axis-aligned components.

IV. Novel Entropy Approximation

The critical part of the entropy calculation is the logarithm of the Gaussian mixture $g(\underline{x})$ in (3). To obtain an accurate and versatile entropy approximation that can be evaluated analytically, the key idea of the proposed approach is to use an appropriate approximation of the logarithm of $g(\underline{x})$ for each Gaussian component of $f(\underline{x})$.

A. *Component-wise Taylor-series Expansion*

By replacing the logarithm with a multivariate Taylor-series expansion, the resulting integral can be solved in closed form. Therefore, the logarithm is expanded around the mean vector $\underline{\mu}_i$ of each Gaussian component of $f(\underline{x})$, which leads to

$$\log g(\underline{x}) = \sum_{k=0}^{R} \frac{1}{k!} \left((\underline{x} - \underline{\mu}_i) \odot \nabla\right)^k \log g(\underline{x}) \Big|_{\underline{x}=\underline{\mu}_i} + O_R$$

for $i = 1, 2, \ldots, L$, where ∇ is the gradient with respect to $\underline{x}$, O_R is the remainder term, and $\odot$ is the so-called matrix contradiction operator

$$c = \mathbf{A} \odot \mathbf{B} = \sum_i \sum_j A_{ij} \cdot B_{ij} \ ,$$

for two matrices $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{N \times M}$, which consists of an element-wise matrix multiplication and a subsequent summation of all matrix elements.

It is obvious that the Taylor-series expansion is of infinite order since derivatives of a sum of exponential functions are considered. Thus, the expansion has to be truncated in order to maintain a practical solution. Truncating at order R yields the approximation

$$\begin{aligned} H(\boldsymbol{x}) \approx -\sum_{i=1}^{L} \int_{\mathbb{R}^N} \omega_i \cdot \mathcal{N}(\underline{x}; \underline{\mu}_i, \mathbf{C}_i) \cdot \\ \left(\sum_{k=0}^{R} \frac{1}{k!} \left((\underline{x} - \underline{\mu}_k) \odot \nabla \right)^k \log g(\underline{x}) \Big|_{\underline{x}=\underline{\mu}_i} \right) \mathrm{d}\underline{x} \end{aligned} \tag{4}$$

of the entropy.

This approximation can be evaluated analytically, as solving the integral for component i corresponds[1] to determining the first R central moments of a Gaussian density. These moments can be calculated on the basis of the mean $\underline{\mu}_i$ and the elements of the covariance matrix $\mathbf{C}_i$.

Furthermore, the derivatives of the logarithm of $g(\underline{x})$ always exist. In order to avoid the complex representation of for example a cubix and quadrix for third-order and forth-order derivatives, respectively, Kronecker algebra can be employed for a compact representation. This is demonstrated in the following example.

Example 1 (Derivatives of a Gaussian)
Essential parts of the proposed entropy approximation are the derivatives of a Gaussian density

$$g(\underline{x}) := \mathcal{N}(\underline{x}; \underline{\mu}, \mathbf{C}) = \tfrac{1}{\sqrt{|2\pi\mathbf{C}|}} e^{-\frac{1}{2}(\underline{x}-\underline{\mu})^{\mathrm{T}} \mathbf{C}^{-1} (\underline{x}-\underline{\mu})} \tag{5}$$

with respect to the vector $\underline{x}$. In the following, the first-order up to the third-order derivatives are given. Employing $\frac{\partial}{\partial \underline{\mu}} (\underline{x} - \underline{\mu})^{\mathrm{T}} \mathbf{C}^{-1} (\underline{x} - \underline{\mu}) = -2\mathbf{C}^{-1}(\underline{x} - \underline{\mu})$, the first-order derivative is (see [13] for comparison)

$$\nabla g(\underline{x}) = \frac{\partial}{\partial \underline{x}} \mathcal{N}(\underline{x}; \underline{\mu}, \mathbf{C}) = \mathbf{C}^{-1}(\underline{x} - \underline{\mu}) \cdot g(\underline{x}) \ .$$

The second-order derivative or Hessian of (5) is given by

$$\begin{aligned} \mathbf{H}(\underline{x}) &= \frac{\partial^2}{\partial \underline{x} \partial \underline{x}^{\mathrm{T}}} \mathcal{N}(\underline{x}; \underline{\mu}, \mathbf{C}) \\ &= \mathbf{C}^{-1}(\underline{x} - \underline{\mu}) \left(\nabla g(\underline{x}) \right)^{\mathrm{T}} - \mathbf{C}^{-1} g(\underline{x}) \ . \end{aligned} \tag{6}$$

Using the Kronecker product $\otimes$ and matrix-vectorization $\#$, the third-order derivative can be written as

$$\frac{\partial^3}{\partial \underline{x} \partial \underline{x}^{\mathrm{T}} \partial \underline{x}} \mathcal{N}(\underline{x}; \underline{\mu}, \mathbf{C}) =$$
$$\mathbf{H}(\underline{x}) \otimes \left(\mathbf{C}^{-1} (\underline{x} - \underline{\mu}) \right) - \nabla g(\underline{x}) \otimes \mathbf{C}^{-1} - \# \left(\mathbf{C}^{-1} \right) \cdot \left(\nabla g(\underline{x}) \right)^{\mathrm{T}} \ .$$

It can be easily seen, that each derivative uses the results of lower-order derivatives and thus, together with the use of Kronecker algebra, calculating higher-order derivatives can be carried out efficiently.

Based on these results, in Appendix A and B the zeroth-order and the second-order component-wise Taylor-series expansions are derived.

B. Variance Reduction via Splitting

As the logarithm term of the entropy is expanded component-wise around the mean vector of the individual Gaussian components of $f(\underline{x})$, there is a strong dependency between the variance of the Gaussians and the employed order of the Taylor-series expansion. Thus, for a Gaussian with large variance, more terms should be used in order to reduce the approximation error, while for very small variances[2], one or two expansion terms are sufficient.

1) Basic Idea: Besides exploiting higher-order terms of the Taylor-series expansion for gaining more accuracy, another key idea of the proposed entropy approximation is to exploit splitting of the Gaussian components in order to reduce their variance. Several possibilities arise for performing a split, where simply reproducing the original component is not sufficient since the symmetry has to be broken to facilitate approximating the logarithm in different ways [14].

A very straightforward way for splitting is to use two Gaussians for replacing the original component, since for two Gaussians a replacement that preserves mean and covariance can be easily guaranteed [15]. For entropy approximation, it is also important that splitting preserves the shape of the original Gaussian. If repeated splitting of the components of $f(\underline{x})$ captures the original shape exactly, the approximation converges to the true entropy value.

Unfortunately, exactly representing a Gaussian by several ones is not possible[3]. Thus, by every split a small error is introduced. In order to keep this error as small as possible and simultaneously to keep the computational demand bounded, we use a library for splitting the standard one-dimensional Gaussian density into a Gaussian mixture consisting of a constant number of components as proposed in [16]. Throughout this paper, splitting into four components is employed, where the corresponding parameters of this mixture are listed in Table I and the resulting approximation is depicted in Fig. 1. This library can be applied to splitting arbitrary multivariate Gaussian densities by means of covariance diagonalization, shifting, and scaling.

TABLE I
SPLITTING LIBRARY

i	$\tilde{\omega}_i$	$\tilde{\mu}_i$	$\tilde{\sigma}_i$
1	0.12738084098	-1.4131205233	0.51751260421
2	0.37261915901	-0.44973059608	0.51751260421
3	0.37261915901	0.44973059608	0.51751260421
4	0.12738084098	1.4131205233	0.51751260421

[1] up to a constant factor

[2] Consider the border case of a Gaussian with zero variance, where even the zeroth-order expansion is exact.

[3] Exact representation requires a infinite number of Gaussians.

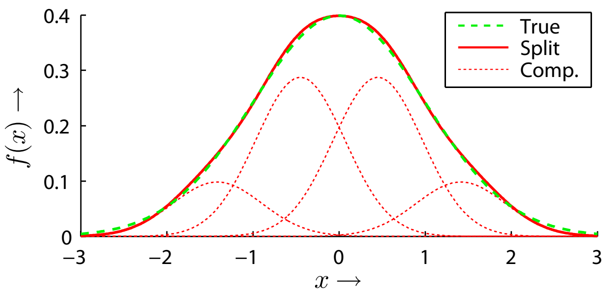

Fig. 1. Approximation of a standard Gaussian density by means of a Gaussian mixture consisting of four components.

Remark 1 (Affected Gaussian Mixture) *It is important to note that splitting only affects the Gaussian mixture $f(\underline{x})$ in* (3). *The mixture $g(\underline{x})$ remains unchanged, since splitting $g(\underline{x})$ does not improve the approximation accuracy.*

2) Performing the Splitting: For refining the approximation by employing the splitting library, the following steps have to be performed repeatedly.

a) Component Selection: At first, a Gaussian component of $f(\underline{x})$ is selected for splitting. The simplest but also computationally most demanding way is to select every component. Obviously, a lot of computation time is wasted for splitting components, which already have small variances.

Instead, we select at each splitting round the component $\omega \cdot \mathcal{N}(\underline{x}; \underline{\mu}, \mathbf{C})$, whose covariance ellipsoid has the largest principal axis. The length of a principal axis coincides with the largest eigenvalue v_d of the covariance matrix $\mathbf{C}$, where the index d indicates the number of the largest principal axis.

b) Component Replacement: Splitting is then performed along the largest principal axis. To replace $\omega \cdot \mathcal{N}(\underline{x}; \underline{\mu}, \mathbf{C})$ by the Gaussian mixture

$$\omega \cdot \mathcal{N}(\underline{x}; \underline{\mu}, \mathbf{C}) \approx \sum_{i=1}^{M} \omega_i \cdot \mathcal{N}(\underline{x}; \underline{\mu}_i, \mathbf{C}_i) \ , \tag{7}$$

the covariance matrix $\mathbf{C}$ is diagonalized using eigenvalue decomposition. This results in the axis-aligned weighted Gaussian $\omega \cdot \mathcal{N}(\underline{x}; \underline{\mu}, \mathbf{D})$, where $\mathbf{D} = \mathrm{diag}(v_1, v_2, \ldots, v_N)$ is the diagonal matrix of eigenvalues of $\mathbf{C}$. This Gaussian can be easily replaced by the mixture $\sum_{i=1}^{M} \omega_i \cdot \mathcal{N}(\underline{x}; \underline{\mu}_i, \mathbf{D}_i)$. The parameters of the mixture are determined according to

$$\begin{aligned} \omega_i &= \tilde{\omega}_i \cdot \omega \ , \\ \underline{\mu}_i &= \underline{\mu} + \sqrt{v_d} \cdot \tilde{\mu}_i \cdot \underline{e}_d \ , \\ \mathbf{D}_i &= \mathrm{diag}(v_1, \ldots, v_{d-1}, \tilde{\sigma}_i^2 \cdot v_d, v_{d+1}, \ldots, v_N) \ , \end{aligned}$$

for $i = 1, \ldots, M$, where $\underline{e}_d = [0, \ldots, 0, 1, 0, \ldots, 0]^{\mathrm{T}}$ is the canonical unit vector, where only element d is one. For $M = 4$, the parameters $\tilde{\omega}_i$, $\tilde{\mu}_i$, and $\tilde{\sigma}_i$ accord to the library values in Table I.

Finally, the diagonal covariance matrices $\mathbf{D}_i$ are rotated back according to

$$\mathbf{C}_i = \mathbf{V} \mathbf{D}_i \mathbf{V}^{\mathrm{T}} \ ,$$

where $\mathbf{V}$ is the matrix of eigenvectors of the covariance matrix $\mathbf{C}$, which yields the desired replacement (7).

C. Properties

The proposed approach satisfies all requirements on an entropy approximation. Obviously, it is deterministic and makes no assumption on the structure of the Gaussian mixture. Furthermore, the following theorem allows demonstrating that the remaining requirement regarding the trade-off between accuracy and computational demand is also fulfilled.

Theorem 1 (Component-wise Entropy Approximation) *The component-wise Taylor-series expansion* (4) *of the entropy employing component splitting* (7) *has following properties:*

1) *The approximation is exact for $R \rightarrow \infty$ and $M \rightarrow \infty$.*
2) *The approximation is exact for $f(\underline{x})$ being a Gaussian density, $R \geq 2$, and $M = 1$, i.e., without splitting.*
3) *All odd order terms of the expansion are zero.*

PROOF.
1) Follows directly from the properties of the Taylor-series expansion and the universal approximation property of a Gaussian mixture.
2) With $R = 2$, $M = 1$, $f(\underline{x}) = \mathcal{N}(\underline{x}; \underline{\mu}, \mathbf{C})$ and the result of Appendix B for $L = 1$, the entropy approximation yields

$$\begin{aligned} H(\underline{\boldsymbol{x}}) &= -\log \mathcal{N}(\underline{\mu}; \underline{\mu}, \mathbf{C}) + \tfrac{1}{2} \underbrace{\mathbf{C}^{-1} \odot \mathbf{C}}_{=\dim(\underline{x})=N} \\ &= -\log \tfrac{1}{\sqrt{|2\pi\mathbf{C}|}} + \tfrac{1}{2} \log e^N \\ &= \frac{1}{2} \log |2\pi e \mathbf{C}| \ , \end{aligned}$$

which coincides with (2).
3) As stated in Section IV-A, evaluating (4) depends on determining the central moments of the Gaussian components of $f(\underline{x})$. Since all odd central moments of a Gaussian density are zero, the odd-order terms of the expansion are also zero. □

Thus, by the choice of the order of expansion R and the number of components used for splitting M, the user can obtain arbitrarily accurate approximations at the expense of computational resources. Thanks to the Taylor-series expansion and the universal approximation property of a Gaussian mixture, the approximation converges to the exact entropy value for an increasing R and M. This allows providing a deterministic alternative to the Monte Carlo approach, which so far was the only converging approximation.

V. Entropy Bounds

Independent of the used entropy approximation method, it is generally difficult or even impossible to quantify the deviation between the true entropy value and its approximation. In some situations, the approximation may be arbitrarily wrong. By providing a close lower and upper bound of the entropy value of a Gaussian mixture random vector, it is possible to decide whether the approximation is meaningful or not. Furthermore, as we will show, both bounds can be calculated in closed form. Thus, the bounds themselves can be used for efficiently approximating the true entropy value.

A. Lower Bound

A lower bound of (3) can be obtained by employing an upper bound of the Kullback-Leibler divergence (see [17]) between two Gaussian mixtures, which is derived in [7]. In doing so, the logarithm is moved outside the integral and only an integral of a product of two Gaussian densities remains, which has a well-known closed-form solution.

Theorem 2 (Lower Bound)
A lower bound $H_l(\underline{\boldsymbol{x}})$ of (3) *is given by*

$$H_l(\underline{\boldsymbol{x}}) = -\sum_{i=1}^{L} \omega_i \cdot \log\left(\sum_{j=1}^{L} \omega_j \cdot z_{i,j}\right) \ ,$$

with $z_{i,j} = \mathcal{N}\left(\underline{\mu}_i; \underline{\mu}_j, \mathbf{C}_i + \mathbf{C}_j\right)$.

PROOF. Since $-\log x$ is concave in x, Jensen's inequality (see e.g. [3]) can be employed. Thus, with $-\log \mathrm{E}\{\underline{\boldsymbol{x}}\} \leq \mathrm{E}\{-\log \underline{\boldsymbol{x}}\}$ we obtain a lower bound of (3) according to

$$\begin{aligned} H(\underline{\boldsymbol{x}}) &= -\sum_{i=1}^{L} \omega_i \cdot \int_{\mathbb{R}^N} \mathcal{N}(\underline{x}; \underline{\mu}_i, \mathbf{C}_i) \cdot \log g(\underline{x}) \, \mathrm{d}\underline{x} \\ &\geq -\sum_{i=1}^{L} \omega_i \cdot \log\left(\int_{\mathbb{R}^N} \mathcal{N}(\underline{x}; \underline{\mu}_i, \mathbf{C}_i) \cdot g(\underline{x}) \, \mathrm{d}\underline{x}\right) \\ &= -\sum_{i=1}^{L} \omega_i \cdot \log\left(\sum_{j=1}^{L} \omega_j \cdot z_{i,j}\right) \ , \end{aligned}$$

with the constant

$$\begin{aligned} z_{i,j} &= \int_{\mathbb{R}^N} \mathcal{N}(\underline{x}; \underline{\mu}_i, \mathbf{C}_i) \cdot \mathcal{N}(\underline{x}; \underline{\mu}_j, \mathbf{C}_j) \, \mathrm{d}\underline{x} \\ &= \mathcal{N}\left(\underline{\mu}_i; \underline{\mu}_j, \mathbf{C}_i + \mathbf{C}_j\right) \ . \end{aligned}$$

□

B. Upper Bound

Besides the aforementioned usefulness of bounding values, a cheap calculation of the upper bound is of additional importance. In optimization problems like sensor scheduling, directly minimizing the entropy of Gaussian mixture random vectors can be avoided by minimizing its upper bound. This procedure is reasonable, if the computational demand of calculating the upper bound is significantly lower as the computational demand for approximating the entropy value. The upper bound derived next fulfills this condition, as it consists of a weighted sum of the individual entropies of the Gaussian components (2).

Theorem 3 (Basic Upper Bound)
An upper bound $H_u(\underline{\boldsymbol{x}})$ of (3) *is given by*

$$H_u(\underline{\boldsymbol{x}}) = \sum_{i=1}^{L} \omega_i \cdot \left(-\log\omega_i + \tfrac{1}{2}\log\left((2\pi e)^N |\mathbf{C}_i|\right)\right) \ . \tag{8}$$

PROOF. By separating the i-th component of $g(\underline{x})$, the entropy for $\underline{\boldsymbol{x}}$ can be written as

$$\begin{aligned} H(\underline{\boldsymbol{x}}) &= -\int_{\mathbb{R}^N} \sum_{i=1}^{L} \omega_i \cdot \mathcal{N}(\underline{x}; \underline{\mu}_i, \mathbf{C}_i) \cdot \\ &\qquad \log\left(\sum_{j=1}^{L} \omega_j \cdot \mathcal{N}(\underline{x}; \underline{\mu}_j, \mathbf{C}_j)\right) \mathrm{d}\underline{x} \\ &= -\sum_{i=1}^{L} \omega_i \int_{\mathbb{R}^N} \mathcal{N}(\underline{x}; \underline{\mu}_i, \mathbf{C}_i) \cdot \\ &\qquad \log\Big(\omega_i \cdot \mathcal{N}(\underline{x}; \underline{\mu}_i, \mathbf{C}_i) \cdot (1 + \epsilon_i)\Big) \, \mathrm{d}\underline{x} \\ &= -\sum_{i=1}^{L} \omega_i \int_{\mathbb{R}^N} \mathcal{N}(\underline{x}; \underline{\mu}_i, \mathbf{C}_i) \cdot \\ &\qquad \Big(\log\Big(\omega_i \cdot \mathcal{N}(\underline{x}; \underline{\mu}_i, \mathbf{C}_i)\Big) + \log(1 + \epsilon_i)\Big) \, \mathrm{d}\underline{x} \ , \end{aligned} \tag{9}$$

where

$$\epsilon_i = \frac{\sum_{i \neq j=1}^{L} \omega_j \cdot \mathcal{N}(\underline{x}; \underline{\mu}_j, \mathbf{C}_j)}{\omega_i \cdot \mathcal{N}(\underline{x}; \underline{\mu}_i, \mathbf{C}_i)} \ . \tag{10}$$

Since $\log(1 + \epsilon_i)$ in (9) is always non-negative, neglecting it yields the desired upper bound. □

Typically, the upper bound (8) is significantly closer to the true entropy value than the well-known bound given by a single Gaussian matching the first two moments of $f(\underline{x})$. Furthermore, the bound is exact for the single Gaussian case and in cases, where the Gaussian components of $f(\underline{x})$ are separated, i.e., the shared support of the components in $f(\underline{x})$ becomes negligible[4].

C. Upper Bound Refinement

Even if $f(\underline{x})$ has a large number of components, the shape of the Gaussian mixture is often not that complex. For example, a mode of $f(\underline{x})$ is represented by several Gaussians, whereas a single component would be adequate for approximately representing the mode. Another very common example are clustered Gaussian mixtures, where the mixture consists of (almost) separated clusters of Gaussian components, where each cluster can be adequately represented by a single Gaussian. As shown in the following, merging several components of $f(\underline{x})$ to a single Gaussian allows to calculate a further upper bound of the entropy.

Theorem 4 (Upper Bound by Merging Gaussians)
Given a Gaussian mixture random vector $\underline{\boldsymbol{x}} \sim f(\underline{x})$, where the mixture $f(\underline{x})$ is divided into two mixtures according to $f(\underline{x}) = f_1(\underline{x}) + f_2(\underline{x})$. Replacing $f_1(\underline{x})$ by a weighted Gaussian $\tilde{f}_1(\underline{x}) := \omega \cdot \mathcal{N}(\underline{x}; \underline{\mu}_1, \mathbf{C}_1)$ that matches the first two moments of $f_1(\underline{x})$, where $\omega = \int f_1(\underline{x}) \, \mathrm{d}x$, yields a new mixture $\tilde{f}(\underline{x}) = \tilde{f}_1(\underline{x}) + f_2(\underline{x})$ with

$$H(\underline{\boldsymbol{x}}) \leq -\int_{\mathbb{R}^N} \tilde{f}_1(\underline{x}) \cdot \log \tilde{f}_1(\underline{x}) + f_2(\underline{x}) \cdot \log f_2(\underline{x}) \, \mathrm{d}\underline{x} \tag{11}$$

[4] This corresponds to the case, where ϵ_i in (10) approaches zero.

and thus, applying Theorem 3 on (11) *provides an easily computable upper bound for the entropy of* $\underline{\boldsymbol{x}}$.

PROOF. The entropy for $\underline{\boldsymbol{x}}$ can be written as

$$H(\underline{\boldsymbol{x}}) = -\int_{\mathbb{R}^N} f_1(\underline{x}) \cdot \log f(\underline{x}) + f_2(\underline{x}) \cdot \log f(\underline{x}) \, \mathrm{d}\underline{x} \ .$$

Separating $f_1(\underline{x})$ and $f_2(\underline{x})$ from $f(\underline{x})$ in the logarithm terms similarly to the proof of Theorem 3 leads to

$$H(\underline{\boldsymbol{x}}) \leq -\int_{\mathbb{R}^N} f_1(\underline{x}) \cdot \log f_1(\underline{x}) + f_2(\underline{x}) \cdot \log f_2(\underline{x}) \, \mathrm{d}\underline{x} \ .$$

By exploiting the fact that the entropy of $\tilde{f}_1(\underline{x})$ upper bounds the entropy of $f_1(\underline{x})$ results in (11). Evaluating (8) separately for $\tilde{f}_1(\underline{x})$ and $f_2(\underline{x})$ automatically provides the easily computable upper bound. □

It is important to note that Theorem 4 and (11), respectively, provide a family of upper bounds: All possible combinations of merged and unmerged Gaussian components give an upper bound. Obviously, the better the merged components can be represented by a single Gaussian, the tighter the upper bound provided by Theorem 4 is. A lowest upper bound will be one that merges clusters of Gaussians that are approximately Gaussian-shaped and does not merge well-separated components. In this case, the entropy value contribution of a merged cluster to the bound (8) is close to the entropy of the original (unmerged) mixture and thus potentially lower than the contribution of the individual Gaussians of the original mixture to the bound.

Instead of evaluating the whole family of bounds in a brute-force fashion for obtaining the lowest upper bound, a more efficient algorithm is proposed. According to Algorithm 1, Gaussian components of the mixture $f(\underline{x})$ are successively merged in order to identify Gaussian-shaped clusters (line 3). Afterwards, the corresponding upper bound is calculated (line 4) and compared with the currently lowest upper bound (line 5).

Methods that can be used for merging in line 3 are manifold. They differ in the distance measure used for identifying similar Gaussian components in $f(\underline{x})$ and the number of components merged in one step. Merging-based Gaussian mixture reduction methods like Salmond's clustering algorithm [18] or Runnall's reduction method [15] typically provide a good trade-off between computational complexity and accuracy in identifying Gaussian-shaped clusters. In this paper, we employ Runnall's method, where at each step two components are merged. The distance measure of this method is based on the Kullback-Leibler divergence, which is scale invariant and thus is the ideal measure for Gaussian mixture reduction purposes.

Example 2 (Upper Bound Refinement)
In this example, the functionality of Algorithm 1 is demonstrated by applying it to a two-dimensional random vector $\underline{\boldsymbol{x}}$ represented by the six component Gaussian mixture depicted in Fig. 2. The sequence of upper bounds generated by Algorithm 1 is listed in the following table.

Algorithm 1 $H_u(\underline{\boldsymbol{x}}) \leftarrow \mathrm{Refine}(f(\underline{x}))$

```
1: $H_u(\underline{\boldsymbol{x}}) \leftarrow \mathrm{UpperBound}(f(\underline{x}))$        // According to (8)
2: while Number of components of $f(\underline{x}) > 1$ do
3:     $\tilde{f}(\underline{x}) \leftarrow \mathrm{Merge}(f(\underline{x}))$
4:     $H_{\mathrm{tmp}} \leftarrow \mathrm{UpperBound}(\tilde{f}(\underline{x}))$        // According to (8)
5:     $H_u(\underline{\boldsymbol{x}}) \leftarrow \min\{H_u(\underline{\boldsymbol{x}}), H_{\mathrm{tmp}}\}$
6:     $f(\underline{x}) \leftarrow \tilde{f}(\underline{x})$
7: end while
```

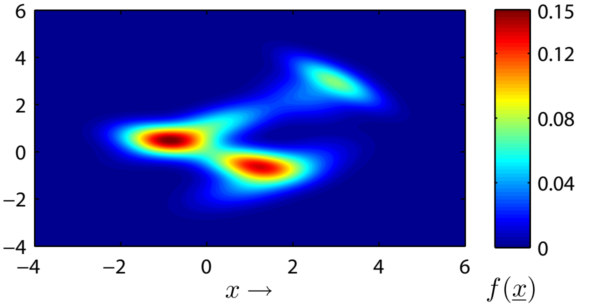

Fig. 2. Top view on a Gaussian mixture consisting of six components with modes at $(-0.79, 0.49)$, $(1.25, 0.63)$, and $(2.91, 3.07)$.

step	1	2	3	4	5	6
$H_u(\underline{\boldsymbol{x}})$	3.671	3.601	3.463	3.329	3.467	3.594
# Gaussians	6	5	4	3	2	1

Thus, the lowest upper bound is calculated at step four, which is also the return value of Algorithm 1. This bound corresponds to a Gaussian mixture reduced to three components, which is sufficient for representing the three modes of the original mixture (see Fig. 2).

VI. SIMULATION RESULTS

For demonstrating the effectiveness of the proposed entropy approximation method, the first simulation of this section is concerned with a parameter identification problem. A multivariate Gaussian mixture is utilized in the second simulation for demonstrating the usefulness of the proposed bounds.

A. *Application: Parameter Identification*

This simulation is concerned with the practical application of identifying the unknown parameter a in the linear function

$$\boldsymbol{y} = a \cdot \boldsymbol{x} + \boldsymbol{w} \ . \tag{12}$$

Therefore, samples are drawn randomly from the Gaussian mixture density function characterizing the random variable $\boldsymbol{x}$ and are propagated through the function (12), which results in a sample representation for the density of $\boldsymbol{y}$. Then, the samples for $\boldsymbol{y}$ are used for determining the estimate $\tilde{\boldsymbol{y}}$ by propagating them through

$$\tilde{\boldsymbol{y}} = \tilde{a} \cdot \boldsymbol{y} \ ,$$

with $\tilde{a}$ being a scaling parameter. This allows identifying the unknown parameter a by minimizing the entropy $H(\boldsymbol{e})$ of the deviation

$$\boldsymbol{e} = \boldsymbol{y} - \tilde{\boldsymbol{y}} \ ,$$

where the entropy is minimized for $\tilde{a} = \frac{1}{a}$.

Specifically, the parameter in (12) is $a = 2$, the Gaussian mixture for $\boldsymbol{x}$ is given by

$$f(x) = 0.4 \cdot \mathcal{N}(x; -1, .025) + 0.6 \cdot \mathcal{N}(x; 1, 1) \ ,$$

and $\boldsymbol{w}$ represents Gaussian noise with zero mean and variance $\sigma_w^2 = 0.04$. Furthermore, 100 samples are drawn from $\boldsymbol{x}$ as well as from $\boldsymbol{w}$ and a Parzen density estimator [19] is employed for determining a Gaussian mixture approximation of the density of $\boldsymbol{e}$.

For minimizing the entropy $H(\boldsymbol{e})$, classical optimization methods like gradient descent could by applied, which excludes the use of Monte Carlo sampling for approximating $H(\boldsymbol{e})$. However, in this simulation the concrete optimization method is not relevant. Instead, we simply vary the scaling parameter $\tilde{a}$ in a brute force fashion from -2 to 6 in order to demonstrate to accuracy provided by the proposed entropy approximation. The resulting entropy approximation for several $\tilde{a}$ and for a zeroth-order Taylor-series expansion as well as a second-order Taylor-series expansion are depicted in Fig. 3.

Even the results of the zeroth-order expansion are quite close to true entropy values, which are calculated by means of numerical integration. Thus, identifying the unknown parameter a is possible, since the minimum of $H(\boldsymbol{e})$ for the inverse parameter $\tilde{a} = 0.5$ can be correctly determined. By spending additional effort for utilizing the second-order expansion, an entropy approximation is provided that coincides with the ground truth.

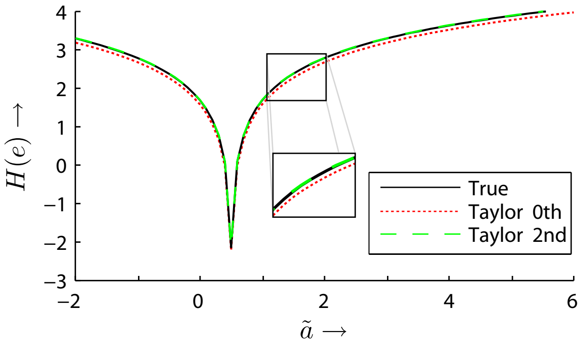

Fig. 3. Entropy approximation for parameter identification by employing zeroth-order (red, dotted line) and second-order Taylor-series expansion (green, dashed).

B. Multivariate Gaussian Mixture

In the next simulation, the two-dimensional random vector $\underline{\boldsymbol{x}}$ is characterized by a Gaussian mixture consisting of $L = 5$ components with parameters

$$\begin{aligned} &\omega_i = 0.2 \ , \ \text{for } i = 1, \dots, 5 \ , \\ &\underline{\mu}_1 = [0, 0]^\mathrm{T} \ , \ \underline{\mu}_2 = [3, 2]^\mathrm{T} \ , \ \underline{\mu}_3 = [1, -0.5]^\mathrm{T} \ , \\ &\underline{\mu}_4 = [2.5, 1.5]^\mathrm{T} \ , \ \underline{\mu}_5 = c \cdot [1, 1]^\mathrm{T} \ , \\ &\mathbf{C}_1 = \mathrm{diag}(0.16, 1) \ , \ \mathbf{C}_2 = \mathrm{diag}(1, 0.16) \ , \\ &\mathbf{C}_3 = \mathbf{C}_4 = \mathbf{C}_5 = \mathrm{diag}(0.5, 0.5) \ . \end{aligned}$$

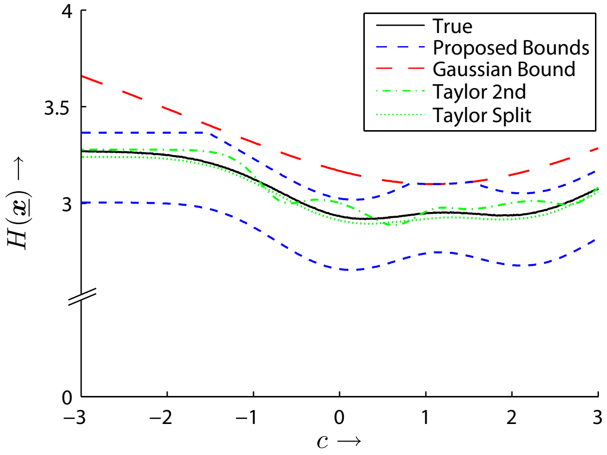

Fig. 4. True entropy values (black, solid line) for a five component Gaussian mixture, where the value of c is used for varying the position of the fifth component, compared to the approximate values resulting from the proposed bounds (blue, dashed), the single Gaussian bound (red, long dashed) as well as the proposed component-wise second-order Taylor-series expansion with (green, dash-dotted) and without splitting (green, dotted).

The position $\underline{\mu}_5$ of the fifth component is varied by the scalar parameter $c \in [-3, 3]$. Various methods are utilized for approximating the entropy value: Lower and upper bounds are calculated according to Theorem 2 and Algorithm 1, respectively. For comparison, the upper bound given by a single Gaussian matching mean and covariance of the mixture is also calculated. Furthermore, a component-wise second-order Taylor-series expansion is used with and without splitting as described in Section IV. To keep the computational demand bounded, a maximum of 20 splitting operations are permitted. Again, the true entropy value is determined by numerical integration.

The resulting entropy values for different c are depicted in Fig. 4. It can be clearly seen that the proposed lower and upper bound are close to the true entropy value, while the well-known single Gaussian bound is significantly less tight. It is worth mentioning that for $c \in [0.8, 1.5]$ both upper bounds coincide as the fifth Gaussian component is in between the remaining components and thus, a single Gaussian representation of the mixture is appropriate. The constant upper bound values for $c \in [-3, -1.5]$ are based on the fact that the Gaussian mixture consists of three well-separated cluster of components.

The component-wise second-order Taylor-series expansion without splitting already provides good approximate entropy values, which are always between the lower and upper bound. However, by additionally employing the proposed splitting method, an almost exact approximation is obtained. This improvement in accuracy is at the expense of an increased computation time, which is one order of magnitude larger than the time consumed by the Taylor-series expansion without splitting.

VII. Conclusions and Future Work

In this paper, a novel method for an approximate entropy calculation of Gaussian mixture random vectors was introduced. In contrast to existing methods, the proposed approach facilitates a tradeoff of computational demand for accuracy. This is achieved by the choice of the order of the component-wise Taylor-series expansion as well as by the number of splitting operations for variance reduction.

Additionally, tight lower and upper bounds of the true entropy value were derived, which can be evaluated in closed form. By the use of such a pair of bounds, some kind of confidence interval for the approximate entropy value of the proposed component-wise Taylor-series expansion method can be calculated. Thanks to the proposed computationally efficient refinement method for the upper bound, utilizing the bound directly for approximating the true entropy value is also possible.

An interesting point for future research is concerned with effectively determining the number of splitting operations to be used. As each splitting operation introduces a small approximation error, it may be necessary to eventually stop splitting to prevent this error becoming dominant.

Based on the splitting operation, a refinement of the proposed lower bound can be introduced. It can be shown that any splitting of components of a Gaussian mixture that preserves mean and covariance gives a lower bound on the entropy value.

Appendix

A. Zeroth-order Taylor-series Expansion

For $R = 0$, the zeroth-order Taylor-series expansion is given by

$$\begin{aligned} H(\underline{\boldsymbol{x}}) &\approx -\sum_{i=1}^{L} \int_{\mathbb{R}^N} \omega_i \cdot \mathcal{N}(\underline{x}; \underline{\mu}_i, \mathbf{C}_i) \cdot \log g(\underline{\mu}_i) \,\mathrm{d}\underline{x} \\ &= -\sum_{i=1}^{L} \omega_i \cdot \log g(\underline{\mu}_i) \\ &=: H_0(\underline{\boldsymbol{x}}) \ . \end{aligned}$$

This approximation of the entropy is identical to the first-order Taylor-series expansion due to the fact that the first central moment of a Gaussian density is zero.

B. Second-order Taylor-series Expansion

Employing the second-order derivative of a Gaussian density with respect to its mean vector (6) and with the result of Appendix A, the second-order Taylor-series expansion is given by

$$\begin{aligned} H(\underline{\boldsymbol{x}}) \approx H_0(\underline{\boldsymbol{x}}) - \sum_{i=1}^{L} \frac{\omega_i}{2} \int_{\mathbb{R}^N} \mathcal{N}(\underline{x}; \underline{\mu}_i, \mathbf{C}_i) \cdot \\ \mathbf{F}(\underline{\mu}_i) \odot (\underline{x} - \underline{\mu}_i)(\underline{x} - \underline{\mu}_i)^{\mathrm{T}} \,\mathrm{d}\underline{x} \\ = H_0(\underline{\boldsymbol{x}}) - \sum_{i=1}^{L} \frac{\omega_i}{2} \mathbf{F}(\underline{\mu}_i) \odot \mathbf{C}_i \end{aligned}$$

with

$$\begin{aligned} \mathbf{F}(\underline{x}) = \frac{1}{f(\underline{x})} \sum_{j=1}^{L} \omega_j \cdot \mathbf{C}_j^{-1} \Bigg(\frac{1}{f(\underline{x})} (\underline{x} - \underline{\mu}_j)(\nabla f(\underline{x}))^{\mathrm{T}} + \\ (\underline{x} - \underline{\mu}_j) \Big(\mathbf{C}_j^{-1} (\underline{x} - \underline{\mu}_j) \Big)^{\mathrm{T}} - \mathbf{I} \Bigg) \cdot \mathcal{N}(\underline{x}; \underline{\mu}_j, \mathbf{C}_j) \ . \end{aligned}$$

References

[1] C. E. Shannon, "A Mathematical Theory of Communication," *Bell System Technical Journal*, vol. 27, no. Part I, pp. 379–423, 1948.

[2] R. C. H. Cheng and N. A. K. Amin, "Estimating Parameters in Continuous Univariate Distributions with a Shifted Origin," *Journal of the Royal Statistical Society: Series B*, vol. 45, no. 3, pp. 394–403, 1983.

[3] T. Cover and J. Thomas, *Elements of Information Theory*. New York: John Wiley & Sons, 1991.

[4] J. Manyika and H. Durrant-Whyte, "Information as a basis for management and control in decentralised fusion architectures," in *IEEE Conference on Decision and Control (CDC)*, 1992.

[5] P. Viola and W. M. Wells III, "Alignment by Maximization of Mutual Information," *International Journal of Computer Vision*, vol. 24, no. 2, pp. 137–154, 1997.

[6] V. Maz'ya and G. Schmidt, "On approximate approximations using gaussian kernels," *IMA J. Numer. Anal.*, vol. 16, pp. 13–29, 1996.

[7] J. R. Hershey and P. A. Olsen, "Approximating the Kullback-Leibler Divergence between Gaussian Mixture Models," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 4, Apr. 2007, pp. IV–317–IV–320.

[8] D. E. Catlin, *Estimation, Control, and the Discrete Kalman Filter*, 1st ed., ser. Applied Mathematical Sciences. New York: Springer-Verlag, 1989, vol. 71.

[9] J. Goldberger, S. Gordon, and H. Greenspan, "An Efficient Image Similarity Measure based on Approximations of KL-Divergence Between Two Gaussian Mixtures," in *Proceedings of the Ninth IEEE International Conference on Computer Vision*, vol. 1, Oct. 2003, pp. 487–493.

[10] S. J. Julier and J. K. Uhlmann, "A New Extension of the Kalman Filter to Nonlinear Systems," in *International Symposium on Aerospace/Defence Sensing, Simulation and Control*, 1997.

[11] J. W. Fisher III and J. C. Principe, "A methodology for information theoretic feature extraction," in *Proceedings of the IEEE International Joint Conference on Neural Networks*, A. Stuberud, Ed., vol. 3, 1998, pp. 1712–1716.

[12] J. W. Fisher and T. Darrell, "Speaker Association With Signal-Level Audiovisual Fusion," *IEEE Transactions on Multimedia*, vol. 6, no. 3, pp. 406–413, Jun. 2004.

[13] K. B. Petersen and M. S. Pedersen, "The Matrix Cookbook," Feb. 2008. [Online]. Available: http://matrixcookbook.com/

[14] M. J. Beal, "Variational Algorithms for Approximate Bayesian Inference," Ph.D. dissertation, Gatsby Computational Neuroscience Unit, University College London, 2003.

[15] A. R. Runnalls, "Kullback-Leibler Approach to Gaussian Mixture Reduction," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 43, no. 3, pp. 989–999, Jul. 2007.

[16] U. D. Hanebeck, K. Briechle, and A. Rauh, "Progressive Bayes: A New Framework for Nonlinear State Estimation," in *Proceedings of SPIE, AeroSense Symposium*, vol. 5099, Orlando, Florida, May 2003, pp. 256–267.

[17] S. Kullback and R. A. Leibler, "On Information and Sufficiency," *Annals of Mathematical Statistics*, vol. 22, no. 2, pp. 79–86, 1951.

[18] D. J. Salmond, "Mixture reduction algorithms for target tracking," in *IEE Colloquium on State Estimation in Aerospace and Tracking Applications*, London, UK, Dec. 1989, pp. 7/1–7/4.

[19] E. Parzen, "On Estimation of a Probability Density Function and Mode," *The Annals of Mathematical Statistics*, vol. 33, no. 3, pp. 1065–1076, Sep. 1962.