Progressive Bayes: A New Framework for Nonlinear State Estimation

Uwe D. Hanebeck*[a], Kai Briechle[b], and Andreas Rauh[c]

[a]Univ. Karlsruhe, Inst. of Computer Design and Fault Tolerance, 76128 Karlsruhe, Germany
[b]Technische Univ. München, Inst. of Automatic Control Eng., 80290 München, Germany
[c]Univ. Ulm, Dept. of Measurement, Control, and Microtechnology, 89069 Ulm, Germany

ABSTRACT

This paper is concerned with recursively estimating the internal state of a nonlinear dynamic system by processing noisy measurements and the known system input. In the case of continuous states, an exact analytic representation of the probability density characterizing the estimate is generally too complex for recursive estimation or even impossible to obtain. Hence, it is replaced by a convenient type of approximate density characterized by a finite set of parameters. Of course, parameters are desired that systematically minimize a given measure of deviation between the (often unknown) exact density and its approximation, which in general leads to a complicated optimization problem. Here, a new framework for state estimation based on progressive processing is proposed. Rather than trying to solve the original problem, it is exactly converted into a corresponding system of explicit ordinary first–order differential equations. Solving this system over a finite "time" interval yields the desired optimal density parameters.

1. INTRODUCTION

Many engineering applications require the states of a dynamic system for monitoring, supervision or control purposes. However, the system states are often not directly available for physical or economical reasons. Hence, measurement devices are employed that supply measurements related to the states. Here, we consider the state estimation problem, i.e., the problem of reconstructing the hidden state sequence of a nonlinear dynamic system based on the measurement sequence and the known system input sequence.

An estimate should be made available at every time step and should incorporate all the information gathered so far. Storing all data and reprocessing it at every time step is impractical. Hence, a recursive estimator is required that updates a given estimate based on the current information. We are not interested in point estimates but rather in the entire probability density function characterizing the current estimate. Furthermore, an analytic expression for this density is desired.

We assume nonlinear discrete–time systems with continuous states. In that case, an exact analytic representation of the probability density function characterizing the estimate is generally too complex or not practical for recursive application. Hence, approximations are inevitable.

Early approaches to analytic nonlinear estimation used Gaussian mixture approximations together with individual updating of the mixture components,[1] which yields suboptimal results. On the other hand, systematically minimizing a measure of distance between the true density and its approximation by calculating appropriate density parameters generally is a tough optimization task. Numerical algorithms such as the Expectation–Maximation (EM) algorithm[2] or gradient based schemes suffer from the local minima problem, i.e., their results strongly depend upon the initialization. In addition, convergence may be slow. In the context of density estimation, a deterministic annealing EM algorithm has been proposed to overcome these problems.[3] Beginning with an unimodal objective function at a high temperature, the objective function gradually approaches the original function as the temperature decreases. This method increases the probability of converging to a global optimum. Similar approaches based on moving from a tractable density to the desired density via a sequence of intermediate densities have been proposed in the context of particle filters.[4,5] An alternative approach to guarantee convergence of the EM algorithm is based on modifying the number of mixture components.[6,7]

In this paper, a new framework for the design of stochastic estimators will be introduced, which minimizes a given distance measure based on *both parametric and structural* adaptation of the approximation density. For that purpose, a parameterized true density is introduced, which starts from a simple density and *continuously* approaches the original true density to be approximated. Based on this type of progressive processing, the

* Email: Uwe.Hanebeck@ieee.org

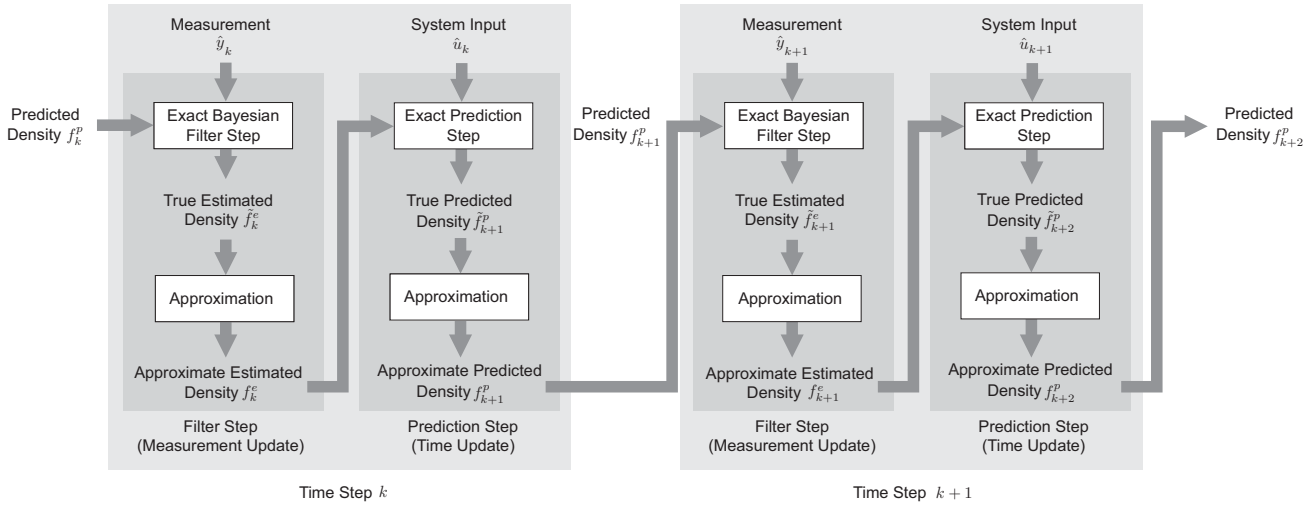

Figure 1. Time evolution of the density of the state estimate.

original optimization problem is converted into a corresponding system of explicit ordinary first–order differential equations. The desired optimal density parameters are then calculated by solving the differential equations over a finite "time" interval. Structural adaptation of the approximation density is performed during the progression when parametric adaptation is not sufficient to keep the desired measure of deviation within a pre–specified tolerance band.

The paper is organized as follows. The next chapter gives a detailed formulation of the estimation problem. The set of differential equations for parametric adaptation is derived in a very general setting in Section 3 and then successively specialized to specific assumptions. Structural adaptation is discussed in Section 4 for the special case of Gaussian mixture approximation densities.

2. PROBLEM FORMULATION

We consider scalar nonlinear discrete–time systems with a *system equation* describing the state evolution of the form

$$x_{k+1} = a_k(x_k, \hat{u}_k, w_k) \ ,$$

where x_k is the scalar system state at time t_k and $\hat{u}_k$ is the known system input. w_k is a random process which accounts for unknown disturbances acting upon the system. In the special case of additive noise, the system equation is given by

$$x_{k+1} = a_k(x_k, \hat{u}_k) + w_k \ ,$$

where, for simplicity, we assume white noise w_k with density $f_k^w(w_k)$. Scalar measurements $\hat{y}_k$ at time t_k are related to the state x_k via the *measurement equation*

$$\hat{y}_k = h_k(x_k, v_k) \ ,$$

where v_k is a random process accounting for the measurement disturbances. For the special case of additive measurement noise, we have

$$\hat{y}_k = h_k(x_k) + v_k \ ,$$

where we again assume white noise v_k with density $f_k^v(v_k)$.

A generic recursive estimator for the considered system model starts with a given estimate $f_0^e(x_0)$. In a Bayesian setting,[8] the *prediction step* or *time update* uses the system equation for propagating the given estimate forward in time according to[8]

$$f_{k+1}^p(x_{k+1}) = \int_{\mathbb{R}} f_{k+1}^T(x_{k+1})\, f_k^e(x_k)\, dx_k \ .$$

The *transition density* $f_{k+1}^T(x_{k+1})$ depends upon both the structure of the system equation and the noise density. In the additive noise case, it is given by

$$f_{k+1}^T(x_{k+1}) = f_k^w(x_{k+1} - \underline{a}_k(x_k, \hat{u}_k)) \ .$$

A new measurement is included by means of the *filter step* or *measurement update* according to Bayes' law[8]

$$f_k^e(x_k) = c_k \, f_k^L(x_k) \, f_k^p(x_k) \ ,$$

where $f_k^L(x_k)$ is the so–called *likelihood* and c_k is a normalization constant. In the special case of additive measurement noise, the likelihood is given by

$$f_k^L(x_k) = f_k^v(\hat{y}_k - h_k(x_k)) \ . \tag{1}$$

This type of generic recursive estimator can directly be applied to discrete state systems, where the integrals are replaced by appropriate summations. However, in the general case of continuous–valued states, it is only of conceptual value. This is caused by the fact, that there exists no analytic density that can be updated in both the prediction and the filter step without changing the type of representation. There are a few exceptions such as linear systems corrupted by Gaussian noise, where a given initial Gaussian density remains Gaussian and can be updated by means of the famous Kalman filter.

Hence, in the general case of nonlinear systems with continuous states, an approximation of the true densities is inevitable. True densities will from now on be denoted by a tilde, e.g. $\tilde{f}_k$. The corresponding approximation will be denoted by f_k. More specifically, here the true density denotes the density that results from performing one processing step based on the previous approximation, Figure 1. For approximation purposes, a convenient analytic density representation $f_k(x_k, \underline{\eta}_k)$ is used, which depends upon a parameter vector $\underline{\eta}_k$. Possible types of approximation densities include exponential densities and mixture densities like Gaussian mixtures.

Of course, an optimal parameter vector $\underline{\eta}_k$ is desired, that systematically minimizes a given distance measure $G(\underline{\eta}_k)$ between the true density $\tilde{f}_k$ and its approximation f_k. Hence, the estimation problem is reduced to an optimization problem, which consists of calculating the smallest set of parameters collected in a parameter vector $\underline{\eta}_k^{opt}$ for which the distance measure attains its minimum and is below a pre–specified threshold G^{max}, i.e., $G(\underline{\eta}_k^{opt}) < G^{max}$.

The main difficulty in calculating the optimal vector $\underline{\eta}_k$ is the existence of local minima. Hence, application of numerical minimization routines generally does not yield the desired optimal parameter vector. An additional challenge is given by the fact, that an analytic expression for the true density cannot always be *explicitly* calculated even for a single processing step. This is for example generally the case in the prediction step. Point–wise numerical evaluation of $\tilde{f}_{k+1}^p$ might be possible, but is very complex from a computational point of view.

3. PARAMETRIC ADAPTATION OF THE APPROXIMATION DENSITY

3.1. Key Idea and General Approach

In this subsection, the general approach to calculating an optimal parameter vector $\underline{\eta}_k$ for a given number of parameters is summarized. For the sake of clarity of the presentation, the time index k will be omitted in the remainder of this paper.

Step 1 (Progressive Processing): The key idea of the new approach is to perform progressive processing. Hence, instead of directly approximating the true density, the new approach starts with a tractable density and continuously approaches the true density via intermediate densities. This is achieved by parameterizing the true density. For that purpose, a progression parameter γ is introduced, which varies between zero and one. For $\gamma = 0$, the parameterized true density $\tilde{f}(x, \gamma)$ is initialized with some kind of density, that is simple to approximate. For $\gamma = 1$, the parameterized true density $\tilde{f}(x, \gamma)$ attains the original true density $\tilde{f}(x)$.

Step 2 (Approximation Density): The second step is to define an appropriate type of approximation density $f(x, \underline{\eta})$ depending upon a parameter vector $\underline{\eta}$ for approximating the true density $\tilde{f}(x, \gamma)$.

	Distance Measure	Approximation Density	Processing Step	Noise Structure & Density
Section 3.1.	Arbitrary	Arbitrary	Arbitrary	Arbitrary
Section 3.2.	Specific: *Squared Integral*	Arbitrary	Arbitrary	Arbitrary
Section 3.3.	Specific: *Squared Integral*	Specific: *Gaussian Mixture*	Arbitrary	Arbitrary
Section 3.4.	Specific: *Squared Integral*	Specific: *Gaussian Mixture*	Specific: *Measurement Update*	Arbitrary
Section 3.5.	Specific: *Squared Integral*	Specific: *Gaussian Mixture*	Specific: *Measurement Update*	Specific: *Additive Gaussian*

Figure 2. Overview of the hierarchical step by step specialization of the derivation of progressive processing with parametric modifications.

Step 3 (Progressive Variant of Deviation Measure): The third step is to define a measure of deviation $G(\underline{\eta}, \gamma)$ between the the parameterized true density $\tilde{f}(x, \gamma)$ and its approximation $f(x, \underline{\eta})$.

Step 4 (LODE for Density Parameters): The final step consists of deriving a system of explicit ordinary first–order differential equations of the form

$$\underline{b}(\underline{\eta}, \gamma) = \mathbf{P}(\underline{\eta})\, \dot{\underline{\eta}} \ ,$$

with $\underline{\eta}(\gamma = 0)$ given.

The general approach will now be successively specialized to a specific type of deviation measure, a specific type of approximation density, a specific type of processing step, and a specific type of noise structure and distribution, Figure 2.

3.2. Squared Integral Deviation

This section is concerned with deriving a system of linear ordinary first–order differential equations for the desired density parameter vector $\underline{\eta}$ given a specific type of distance measure, the squared integral deviation. The results are valid for an *arbitrary* approximation density $f(x, \underline{\eta})$ and an *arbitrary* processing step with an *arbitrary* noise structure and density.

The distance between the true density $\tilde{f}(x, \gamma)$ and its approximation $f(x, \underline{\eta})$ is defined as the squared integral deviation according to

$$G(\underline{\eta}, \gamma) = \frac{1}{2} \int_{\mathbb{R}} \left(\tilde{f}(x, \gamma) - f(x, \underline{\eta}) \right)^2 dx \ . \tag{2}$$

We assume a nominal parameter vector $\bar{\underline{\eta}}$ to be given and consider only small deviations $\Delta\underline{\eta}(\gamma)$ according to $\underline{\eta}(\gamma) = \bar{\underline{\eta}} + \Delta\underline{\eta}(\gamma)$. Around the nominal parameter vector $\bar{\underline{\eta}}$, the approximation density is replaced by a Taylor series expansion up to first order

$$f(x, \underline{\eta}) \approx f(x, \bar{\underline{\eta}}) + \underline{F}(x, \bar{\underline{\eta}})^T \, \Delta\underline{\eta}(\gamma) \qquad \text{with} \qquad \underline{F}(x, \bar{\underline{\eta}}) = \left. \frac{\partial\, f(x, \underline{\eta})}{\partial\, \underline{\eta}} \right|_{\underline{\eta} = \bar{\underline{\eta}}} \ .$$

The distance measure $G(\underline{\eta}, \gamma)$ can now be rewritten accordingly

$$G(\underline{\eta}, \gamma) \approx \frac{1}{2} \int_{\mathbb{R}} \left(\tilde{f}(x, \gamma) - f(x, \bar{\underline{\eta}}) - \underline{F}(x, \bar{\underline{\eta}})^T \, (\bar{\underline{\eta}} - \underline{\eta}) \right)^2 dx \ .$$

Taking the partial derivative of the distance measure $G(\underline{\eta}, \gamma)$ with respect to the parameter vector $\underline{\eta}$ and setting the result to zero, i.e., $\partial\, G / \partial\, \underline{\eta} \stackrel{!}{=} 0$, gives

$$\int_{\mathbb{R}} \left(\tilde{f}(x, \gamma) - f(x, \bar{\underline{\eta}}) - \underline{F}(x, \bar{\underline{\eta}})^T \, \bar{\underline{\eta}} \right) \underline{F}(x, \bar{\underline{\eta}})\, dx = \left(\int_{\mathbb{R}} \underline{F}(x, \bar{\underline{\eta}})\, \underline{F}(x, \bar{\underline{\eta}})^T \, dx \right) \underline{\eta}(\gamma) \ .$$

The partial derivative with respect to γ gives the desired system of explicit ordinary first–order differential equations

$$\int_{\mathbb{R}} \frac{\partial\, \tilde{f}(x,\gamma)}{\partial\, \gamma}\, \underline{F}(x,\bar{\underline{\eta}})\, dx = \left(\int_{\mathbb{R}} \underline{F}(x,\bar{\underline{\eta}})\, \underline{F}(x,\bar{\underline{\eta}})^T\, dx \right) \frac{\partial\, \underline{\eta}}{\partial\, \gamma} \quad ,$$

which can be rewritten as

$$\boxed{\underline{b}(\underline{\eta},\gamma) = \mathbf{P}(\underline{\eta})\, \dot{\underline{\eta}}}$$

where the coefficients are given by

$$\underline{b}(\underline{\eta},\gamma) = \int_{\mathbb{R}} \frac{\partial\, \tilde{f}(x,\gamma)}{\partial\, \gamma}\, \underline{F}(x,\bar{\underline{\eta}})\, dx \qquad \text{and} \qquad \mathbf{P}(\underline{\eta}) = \int_{\mathbb{R}} \underline{F}(x,\bar{\underline{\eta}})\, \underline{F}(x,\bar{\underline{\eta}})^T\, dx \quad .$$

3.3. Gaussian Mixture Approximation

For the special case of a Gaussian mixture approximation according to

$$f(x) = \sum_{i=1}^{L} w^{(i)} \mathcal{N}\left(x - \hat{x}^{(i)}, \sigma^{(i)}\right) \quad ,$$

where $\mathcal{N}(x-m,\sigma)$ is a Gaussian density with mean m and standard deviation σ and $w^{(i)}$ are weighting coefficients with $w^{(i)} > 0$ and $\sum_{i=1}^{L} w^{(i)} = 1$, the matrix $\mathbf{P}(\underline{\eta})$ is composed of L^2 three–by–three block matrices according to

$$\mathbf{P}(\underline{\eta}) = \begin{bmatrix} \mathbf{P}^{(1,1)} & \mathbf{P}^{(1,2)} & \dots & \mathbf{P}^{(1,L)} \\ \mathbf{P}^{(2,1)} & \mathbf{P}^{(2,2)} & \dots & \mathbf{P}^{(2,L)} \\ \vdots & \vdots & & \vdots \\ \mathbf{P}^{(L,1)} & \mathbf{P}^{(L,2)} & \dots & \mathbf{P}^{(L,L)} \end{bmatrix} \quad .$$

The individual block matrices $\mathbf{P}^{(i,j)}$ for $i = 1,\dots,L$ and $j = 1,\dots,L$ contain elements given by

$$P_{n,m}^{(i,j)} = \int_{\mathbb{R}} \frac{\partial\, f^{(i)}\left(x,\underline{\eta}^{(i)}\right)}{\partial\, \underline{\eta}_n^{(i)}} \frac{\partial\, f^{(j)}\left(x,\underline{\eta}^{(j)}\right)}{\partial\, \underline{\eta}_m^{(j)}}\, dx$$

for $n = 1,\dots,3$, and $m = 1,\dots,3$, where $\underline{\eta}^{(i)} = \begin{bmatrix} w^{(i)} & \hat{x}^{(i)} & \sigma^{(i)} \end{bmatrix}^T$ and $\underline{\eta} = \begin{bmatrix} \left(\underline{\eta}^{(1)}\right)^T & \left(\underline{\eta}^{(2)}\right)^T & \cdots & \left(\underline{\eta}^{(L)}\right)^T \end{bmatrix}^T$. The integration can be performed analytically, which finally gives

$$\mathbf{P}^{(i,j)} = \frac{1}{\sqrt{2\,\pi\,(s_i^2 + s_j^2)}} \exp\left(-\frac{1}{2} \frac{(m_i - m_j)^2}{s_i^2 + s_j^2} \right) \begin{bmatrix} P_{1,1}^{(i,j)} & P_{1,2}^{(i,j)} & P_{1,3}^{(i,j)} \\ P_{2,1}^{(i,j)} & P_{2,2}^{(i,j)} & P_{2,3}^{(i,j)} \\ P_{3,1}^{(i,j)} & P_{3,2}^{(i,j)} & P_{3,3}^{(i,j)} \end{bmatrix}$$

with

$$P_{1,1}^{(i,j)} = 1\,,\ P_{1,2}^{(i,j)} = w_j\, \frac{m_i - m_j}{s_i^2 + s_j^2}\,,\ P_{1,3}^{(i,j)} = w_j\, s_j\, \frac{(m_i - m_j)^2 - (s_i^2 + s_j^2)}{(s_i^2 + s_j^2)^2}\,,\ P_{2,1}^{(i,j)} = w_i\, \frac{m_j - m_i}{s_i^2 + s_j^2} \quad ,$$

$$P_{2,2}^{(i,j)} = w_i\, w_j\, \frac{s_i^2 + s_j^2 - (m_i - m_j)^2}{(s_i^2 + s_j^2)^2}\,,\ P_{2,3}^{(i,j)} = w_i\, w_j\, s_j\, \frac{(m_j - m_i)\left((m_i - m_j)^2 - 3\,(s_i^2 + s_j^2)\right)}{(s_i^2 + s_j^2)^3} \quad ,$$

$$P_{3,1}^{(i,j)} = w_i\, s_i\, \frac{(m_i - m_j)^2 - (s_i^2 + s_j^2)}{(s_i^2 + s_j^2)^2}\,,\ P_{3,2}^{(i,j)} = w_i\, w_j\, s_i\, \frac{(m_i - m_j)\left((m_i - m_j)^2 - 3\,(s_i^2 + s_j^2)\right)}{(s_i^2 + s_j^2)^3} \quad ,$$

$$P_{3,3}^{(i,j)} = w_i\, w_j\, s_i\, s_j\, \frac{(m_i - m_j)^4 + 3\,(s_i^2 + s_j^2)\,(s_i^2 + s_j^2 - 2\,(m_i - m_j)^2)}{(s_i^2 + s_j^2)^4} \quad .$$

Remark 3.1 *The elements of the submatrix* $\mathbf{P}^{(i,j)}$ *are weighted by*

$$\exp\left(-\frac{1}{2}\,\frac{(m_i - m_j)^2}{s_i^2 + s_j^2}\right) \quad ,$$

which rapidly decays with growing distance $m_i - m_j$ *between the means of the components* i *and* j*. Hence, when the submatrix corresponds to mixture components spaced far apart, it becomes negligible. This leads to a matrix* $\mathbf{P}(\underline{\eta})$ *that is effectively sparse. In the case of ordered scalar mixture components, the matrix* $\mathbf{P}(\underline{\eta})$ *becomes diagonal dominant, which significantly reduces the computational burden.*

3.4. Measurement Update

In this subsection, we will consider a specific processing step, the measurement update. In this case, a more specific expression for the coefficient vector $\underline{b}(\underline{\eta}, \gamma)$ is obtained. With $\tilde{f}(x,\gamma) = \tilde{f}^e(x,\gamma) = f^p(x)\,\tilde{f}^L(x,\gamma)$ we have

$$\underline{b}^{(i)}(\underline{\eta}, \gamma) = \int_{\mathbb{R}} f^p(x)\,\frac{\partial\,\tilde{f}^L(x,\gamma)}{\partial\,\gamma}\,\frac{\partial\,f^e(x,\underline{\eta})}{\partial\,\underline{\eta}^{(i)}}\,dx \quad .$$

With

$$f^p(x) = \sum_{j=0}^{L^p} f^{(p,j)}(x) = \sum_{j=0}^{L^p} w^{(p,j)}\,\mathcal{N}\left(x - \hat{x}^{(p,j)}, \sigma^{(p,j)}\right)$$

and

$$\frac{\partial\,f^e(x,\underline{\eta})}{\partial\,\underline{\eta}^{(i)}} = f^{(e,i)}\left(x, \underline{\eta}^{(i)}\right)\begin{bmatrix} \frac{1}{w^{(e,i)}} \\ \frac{x - \hat{x}^{(e,i)}}{\left(\sigma^{(e,i)}\right)^2} \\ \frac{\left(x - \hat{x}^{(e,i)}\right)^2 - \left(\sigma^{(e,i)}\right)^2}{\left(\sigma^{(e,i)}\right)^3} \end{bmatrix}$$

we obtain

$$\underline{b}^{(i)}(\underline{\eta}, \gamma) = \sum_{j=0}^{L^p} \int_{\mathbb{R}} f^{(p,j)}(x)\,\frac{\partial\,f^L(x,\gamma)}{\partial\,\gamma}\,f^{(e,i)}\left(x, \underline{\eta}^{(i)}\right)\begin{bmatrix} \frac{1}{w^{(e,i)}} \\ \frac{x - \hat{x}^{(e,i)}}{\left(\sigma^{(e,i)}\right)^2} \\ \frac{\left(x - \hat{x}^{(e,i)}\right)^2 - \left(\sigma^{(e,i)}\right)^2}{\left(\sigma^{(e,i)}\right)^3} \end{bmatrix} dx \quad ,$$

where L^e, L^p are the numbers of mixture components, $w^{(e,i)}$, $w^{(p,i)}$ are the weighting factors, $\hat{x}^{(e,i)}$, $\hat{x}^{(p,i)}$ are the mean values, and $\sigma^{(e,i)}$, $\sigma^{(p,i)}$ are the standard deviations of the estimated density $f^e(x)$ and the predicted density $f^p(x)$, respectively.

3.5. Measurement Update with Additive Gaussian Noise

In the case of additive Gaussian measurement noise with density

$$f^v(v) = \frac{1}{\sqrt{2\,\pi}\sigma^v}\exp\left\{-\frac{1}{2}\frac{v^2}{(\sigma^v)^2}\right\} \quad ,$$

a convenient type of progression schedule is obtained by starting with large measurement noise, which is continuously reduced until the desired standard deviation σ^v is obtained. For that purpose, we define a parameterized noise density $f^v(v,\gamma)$ with a standard deviation according to

$$\bar{\sigma}^v(\gamma) = \frac{1+\epsilon}{\gamma+\epsilon}\sigma^v = \begin{cases} \text{large} & \text{for } \gamma = 0 \\ \sigma^v & \text{for } \gamma = 1 \end{cases} \quad ,$$

where ϵ is a small constant and $\gamma \in [0,1]$. With (1), a specific expression for the derivative of the likelihood with respect to γ is now given by $\partial\,f^L(x,\gamma)/\partial\,\gamma = \partial\,f^v(\hat{y} - h(x),\gamma)/\partial\,\gamma$ with

$$\frac{\partial\,f^v(v,\gamma)}{\partial\,\gamma} = \frac{(1+\epsilon)^2\,(\sigma^v)^2 - (\gamma+\epsilon)^2\,v^2}{\sqrt{2\,\pi}\,(1+\epsilon)^3\,(\sigma^v)^3}\exp\left\{-\frac{1}{2}\frac{v^2}{(\bar{\sigma}^v)^2}\right\} \quad .$$

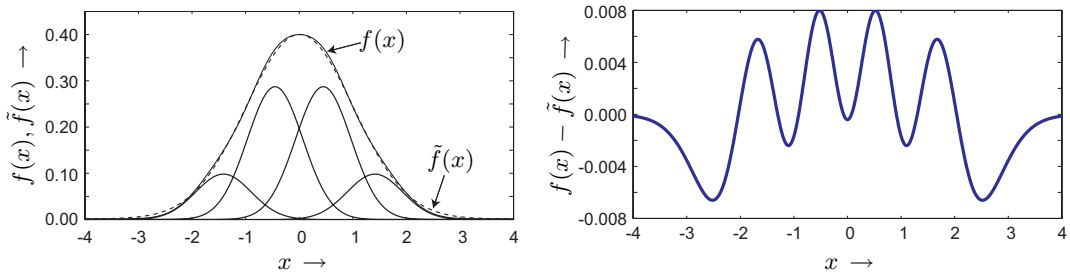

Figure 3. Approximating the standard Gaussian density by means of a Gaussian mixture with $L = 4$ mixture components. Left: True density $\tilde{f}(x)$, its Gaussian mixture approximation $f(x)$, and the individual mixture components $f_i(x)$, $i = 1, \ldots, 4$. Right: The deviation $f(x) - \tilde{f}(x)$ between true and approximate density.

4. STRUCTURAL ADAPTATION OF THE APPROXIMATION DENSITY

While performing the progression, the measure of deviation between the true density and its approximation is continuously checked. For that purpose, a more convenient normalized measure $G_N(\underline{\eta}, \gamma)$ is used instead of (2) that will be introduced in the next Subsection.

As long as $G_N(\underline{\eta}, \gamma)$ is within the prespecified tolerance band, i.e., $G_N^L < G_N(\underline{\eta}, \gamma) < G_N^U$, the progression is continued. Once $G_N(\underline{\eta}, \gamma)$ leaves the tolerance band, the progression is immediately stopped and the number of mixture components is modified.

For increasing the number of mixture components, the most critical component is identified and replaced by a Gaussian mixture with smaller individual variances. Two key ideas are employed for that purpose: The first idea is concerned with a simple criterion for finding the most critical mixture component, which will be introduced in Subsection 4.1. The second idea is to use offline generated splitting libraries for approximating a single Gaussian component by a Gaussian mixture, which will be discussed in Subsection 4.2.

Decreasing the number of components is performed by merging neighboring components with similar parameters. In addition, components with small weighting factors are removed.

Before restarting the progression again after modifying the number of mixture components, an additional correction step is required. This is due to the fact that the structural adaptation of the parameter vector is performed for a fixed value of the progression parameter γ and cannot be compensated by continuing the progression. The appropriate correction step will be derived in Subsection 4.3.

4.1. Identification of Critical Mixture Component

For evaluating the current approximation quality during the progression, a *normalized* distance measure according to

$$G_N(\underline{\eta}, \gamma) = \frac{\displaystyle\int_{\mathbb{R}} \left(\tilde{f}(x, \gamma) - f(x, \underline{\eta})\right)^2 \, dx}{\displaystyle\int_{\mathbb{R}} \left(\tilde{f}(x, \gamma)\right)^2 \, dx + \int_{\mathbb{R}} \left(f(x, \underline{\eta})\right)^2 \, dx}$$

is used. Compared to the original unnormalized measure in (2), this measure is more convenient for specifying deviation tolerances as it ranges between 0 and 1. A perfect match is indicated by $G_N(\underline{\eta}, \gamma) = 0$, the maximum deviation between the true density and its approximation is indicated by $G_N(\underline{\eta}, \gamma) = 1$.

Once the normalized distance measure is larger than a pre–specified threshold, i.e., $G_N(\underline{\eta}, \gamma) > G_{N,max}$ the most critical mixture component responsible for the deviation is identified by evaluating L individual distance measures according to

$$G^{(i)}(\underline{\eta}, \gamma) = \int_{\mathbb{R}} \left(\tilde{f}(x, \gamma) - f(x, \underline{\eta})\right)^2 f^{(i)}\left(x, \underline{\eta}^{(i)}\right) \, dx$$

for $i = 1, \ldots, L$.

4.2. Adaptation of the Number of Mixture Components

Increasing the number of mixture components could simply be performed by replacing the critical component by two components according to

$$w \exp\left\{-\frac{1}{2}\frac{(x-m)^2}{\sigma^2}\right\} \stackrel{!}{\approx} w_1 \exp\left\{-\frac{1}{2}\frac{(x-m_1)^2}{\sigma_1^2}\right\} + w_2 \exp\left\{-\frac{1}{2}\frac{(x-m_2)^2}{\sigma_2^2}\right\} \ .$$

This has been successfully applied to training hidden Markov models for speech recognition.[9–11] In order to modify the form of the total mixture density as little as possible, the parameters are selected as

$$m_1 = m - \epsilon \ , \ m_2 = m + \epsilon \ , \ \sigma_1 = \sigma_2 = \sigma \ , \ w_1 = w_2 = \frac{w}{2} \ , \tag{3}$$

where ϵ is a "small" constant.

However, this form of splitting up a density is difficult to apply in practical situations, since selection of the constant ϵ is always a compromise. When the two new densities are spaced far apart, i.e., ϵ is large, an appreciable approximation error is introduced. On the other hand, when the densities are close, i.e., ϵ is very small, the two densities are treated as one by the progression mechanism. In addition, the local approximation capability is not increased, since the standard deviations of the two densities are the same. Hence, it is advantageous to employ a splitting library that replaces a given Gaussian density by an appropriate Gaussian mixture with components of *smaller variance* and *distinct mean values*.

Without loss of generality, we consider building a library for splitting the standard Gaussian density (zero mean, unit variance). The results are then applied to splitting arbitrary Gaussian densities by means of suitable shifting and scaling. Here, we restrict attention to splitting libraries with equal standard deviations, i.e., the components of the Gaussian mixture have the same standard deviations $\sigma_i = \sigma$, $i = 1, \ldots, L$. In addition, the number L of mixture components is restricted to be a power of two, i.e., $L = 2, 4, 8, 16, \ldots$.

For generating a library, the density is first split into two Gaussian densities according to (3). The standard deviations are then jointly reduced in a progressive fashion, while the corresponding weighting coefficients and mean values are adapted accordingly. When the standard deviations cannot be reduced any further, the resulting two mixture densities are split up according to (3) yielding a total number of $L = 4$ mixture components. Again, the standard deviations are progressively reduced while adapting the corresponding weighting coefficients and mean values. This procedure is repeated until the desired number $L = 2, 4, 8, 16, \ldots$ of mixture components is achieved.

i	w_i	m_i	σ_i
1	0.35690452642552	−1.41312052325139	0.51751260421306
2	0.61042539185142	−0.44973059608233	0.51751260421306
3	0.61042539185143	0.44973059608228	0.51751260421306
4	0.35690452642554	1.41312052325134	0.51751260421306

Figure 4. Parameters of a Gaussian mixture with 4 mixture components for approximating the standard Gaussian density.

Parameters of Gaussian mixture approximations for a standard Gaussian density are given for $L = 4$ in Table 4. A graphical comparison of the original standard Gaussian density and its Gaussian mixture approximation is given in the left part of Figure 3 for $L = 4$ together with the individual mixture components. The corresponding approximation error is shown in the right part of Figure 3.

Similar procedures for library generation can be devised for the case of an arbitrary number of mixture components that are not a power of two. In addition, there might be applications where it is more appropriate to use inhomogeneous density splitting, i.e., splitting libraries where the individual mixture components have unequal standard deviations.

Merging is performed similar to the basic splitting procedure. Two Gaussian densities are merged into a single one according to

$$w_1 \exp\left\{-\frac{1}{2}\frac{(x-m_1)^2}{\sigma_1^2}\right\} + w_2 \exp\left\{-\frac{1}{2}\frac{(x-m_2)^2}{\sigma_2^2}\right\} \stackrel{!}{\approx} w \left\{-\frac{1}{2}\frac{(x-m)^2}{\sigma^2}\right\}$$

with

$$m = \frac{w_1 m_1 + w_2 m_2}{w_1 + w_2} \ , \ \sigma = \frac{w_1 \sigma_1 + w_2 \sigma_2}{w_1 + w_2} \ , \ w = w_1 + w_2 \ ,$$

when the corresponding parameters are close.

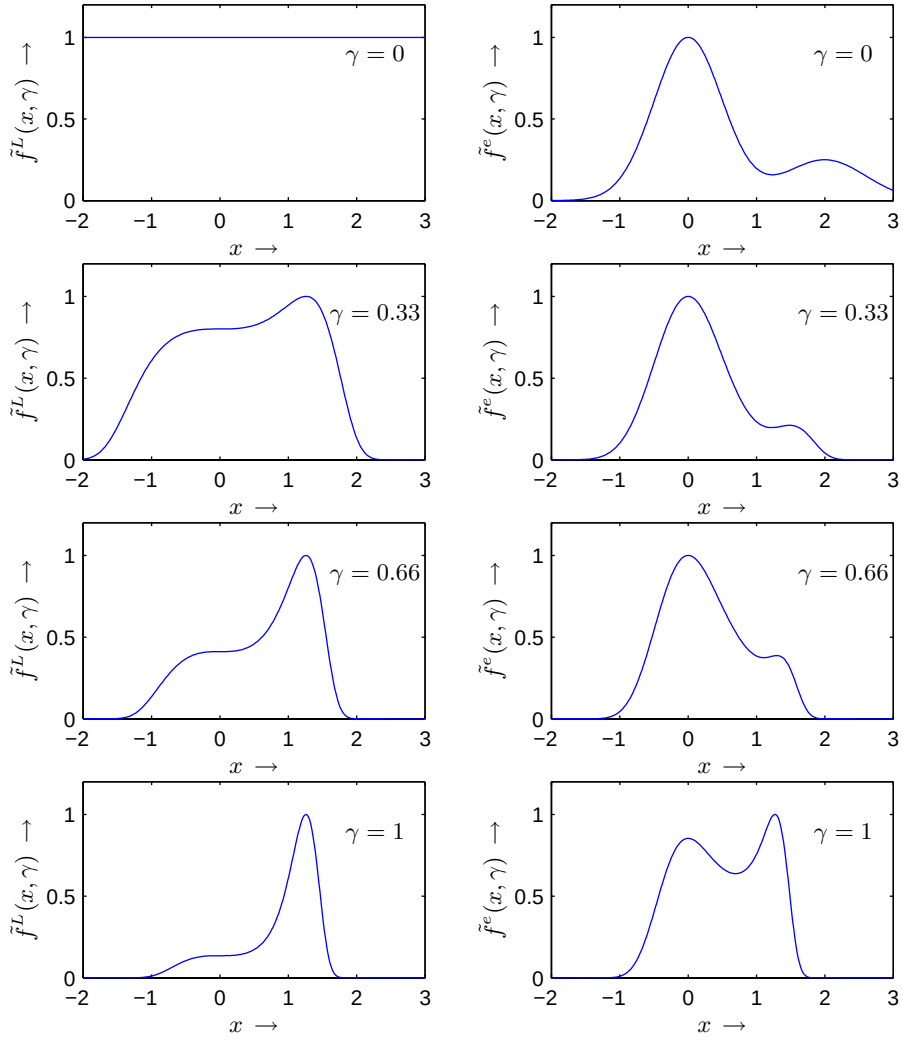

Figure 5. Left: Evolution of the true likelihood. Right: Evolution of the parameterized true posterior $\tilde{f}^e(x,\gamma)$. Please note that the true posterior for $\gamma = 0$, i.e., $\tilde{f}^e(x,\gamma = 0)$, is equal to the prior density $f^p(x)$.

4.3. Additional Correction Step

An additional correction step is required to remove the approximation error introduced by the structural adaptation, which is now derived by

$$\frac{\partial\, G(\underline{\eta},\gamma)}{\partial\, \underline{\eta}} = \underbrace{\int_{\mathbb{R}} \tilde{f}^e(x,\gamma)\, \underline{F}(x,\bar{\underline{\eta}})\, dx}_{T_1} - \underbrace{\int_{\mathbb{R}} f(x,\underline{\eta})\, \underline{F}(x,\bar{\underline{\eta}})\, dx}_{T_2} + \underbrace{\int_{\mathbb{R}} \underline{F}(x,\bar{\underline{\eta}})\, \underline{F}(x,\bar{\underline{\eta}})^T\, dx}_{\mathbf{P}(\bar{\underline{\eta}})} (\bar{\underline{\eta}} - \underline{\eta}) \stackrel{!}{=} 0 \ .$$

Hence, we obtain

$$\underline{\eta} = \bar{\underline{\eta}} + \mathbf{P}(\bar{\underline{\eta}})^{-1}(T_1 - T_2) \tag{4}$$

where in the case of a measurement update the expressions T_1 and T_2 can be simplified as follows

$$T_1 = \int_{\mathbb{R}} \tilde{f}^e(x,\gamma)\, \frac{\partial\, f^e(x,\underline{\eta})}{\partial\, \underline{\eta}^T}\bigg|_{\underline{\eta}=\bar{\underline{\eta}}} dx = \int_{\mathbb{R}} f^p(x)\, f^{(e,i)}\left(x, \bar{\underline{\eta}}^{(i)}\right)\, f^L(x,\gamma) \begin{bmatrix} \frac{1}{w^{(e,i)}} \\ \frac{x-\hat{x}^{(e,i)}}{(\sigma^{e,i})^2} \\ \frac{\left(x-\hat{x}^{(e,i)}\right)^2 - \left(\sigma^{(e,i)}\right)^2}{\left(\sigma^{(e,i)}\right)^3} \end{bmatrix} dx$$

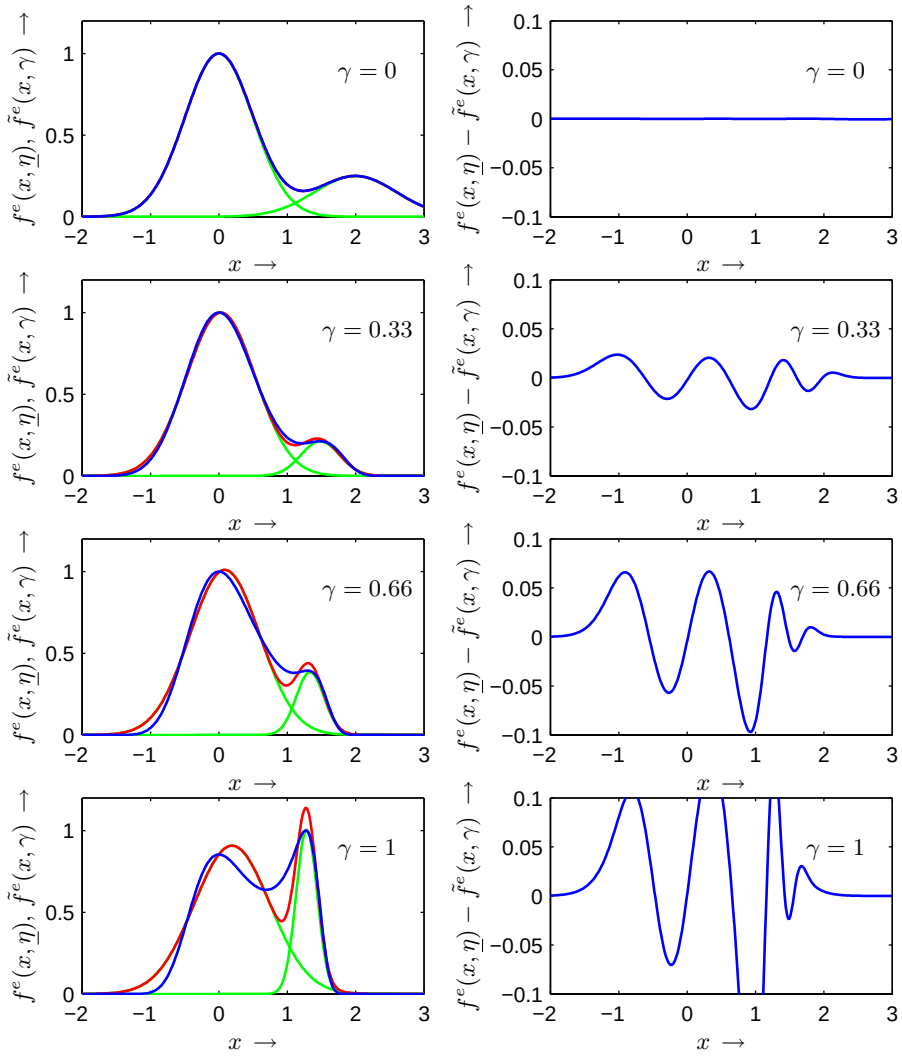

Figure 6. Left: Evolution of the true density, the approximate density, and the mixture components *without* structural adaptation. Right: Evolution of the approximation error.

and

$$T_2 = \int_{\mathbb{R}} f^e(x,\bar{\underline{\eta}})\, \frac{\partial\, f^e(x,\underline{\eta})}{\partial\, \underline{\eta}}\bigg|_{\underline{\eta}=\bar{\underline{\eta}}} dx = \int_{\mathbb{R}} f^e(x,\bar{\underline{\eta}})\, f^{(e,i)}\left(x, \bar{\underline{\eta}}^{(i)}\right) \begin{bmatrix} \frac{1}{w^{(e,i)}} \\ \frac{x-\hat{x}^{(e,i)}}{(\sigma^{e,i})^2} \\ \frac{\left(x-\hat{x}^{(e,i)}\right)^2-\left(\sigma^{(e,i)}\right)^2}{\left(\sigma^{(e,i)}\right)^3} \end{bmatrix} dx \ .$$

Thanks to the fact, that the parameter modifications introduced by increasing or decreasing the number of mixture components are very small, one or two correction steps according to (4) are generally sufficient before restarting the progression.

5. EXAMPLE

We consider a cubic measurement equation according to

$$\hat{y} = x^3 + v \ ,$$

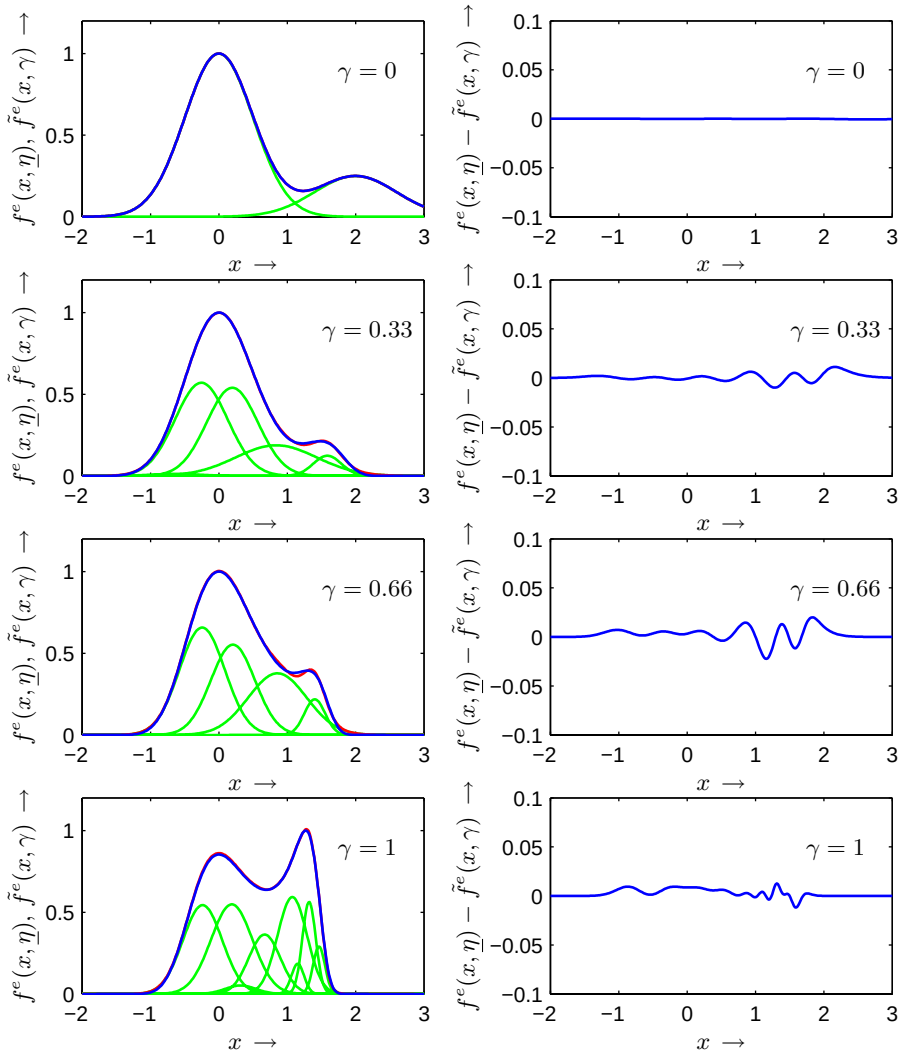

Figure 7. Left: Evolution of the true density, the approximate density, and the mixture components *with* structural adaptation. Right: Evolution of the approximation error.

with v zero mean Gaussian and unit variance. Given the prior density

$$f^p(x) = f^{(p,1)}(x) + f^{(p,2)}(x) = w^{(p,1)}\mathcal{N}\left(x - \hat{x}^{(p,1)}, \sigma^{(p,1)}\right) + w^{(p,2)}\mathcal{N}\left(x - \hat{x}^{(p,2)}, \sigma^{(p,2)}\right)$$

with $w^{(p,1)} = 1.0$, $\hat{x}^{(p,1)} = 0.0$, $\sigma^{(p,1)} = 0.5$, $w^{(p,2)} = 0.3$. $\hat{x}^{(p,2)} = 2.0$, $\sigma^{(p,2)} = 0.6$, we perform a single measurement update with $\hat{y} = 2$. The corresponding likelihood $f^L(x,\gamma)$ and the parameterized *true* posterior density $\tilde{f}^e(x,\gamma)$ are shown in Figure 5 for $\gamma = 0.0, 0.33, 0.66, 1.0$.

First, the progressive processing algorithm proposed in Chap. 3 has been applied *without* the structural adaptation proposed in Chap. 4. Both the calculation of the the coefficients of the set of differential equations and the evaluation of the normalized distance measure requires the solution of integrals, which have been evaluated numerically in this simulation. The resulting set of differential equations has been solved by applying a simple Euler method with 100 steps for γ. The evolution of the approximate posterior $f^e(x,\gamma)$ is shown in Figure 6. The number of mixture components remains unchanged, in this case $L = 2$ for $\gamma = 0.0, 0.33, 0.66, 1.0$. Although the result is optimal with only $L = 2$ mixture components available, it is obvious that the approximation can be enhanced by allowing for more mixture components.

The combined application of the results from Chap. 3 and Chap. 4, i.e., progressive processing including structural adaptation, yields the results shown in Figure 7. For that purpose, the maximum tolerable normalized deviation between the true density and its approximation has been set to 1%. Mixture components with weighting factors less than 10^{-4} are removed. Obviously, the proposed new approach successively increases the number of mixture components and ends up with an appropriate parameter vector with $L = 9$ components.

6. CONCLUSIONS

A new framework for designing stochastic estimators for nonlinear dynamic systems with *continuous* states has been introduced, which is based on approximating the generally intractable true densities by arbitrary *analytic* density representations. The first contribution is concerned with obtaining optimal parameters of an approximate density, that minimize a given measure of distance from the true density. Instead of applying numerical search and optimization techniques, which may suffer from local minima and bad convergence, the problem is *exactly* converted to a system of explicit ordinary first–order differential equations. The desired optimal density parameters are then calculated by solving the differential equations over a finite time interval. The second contribution is concerned with structural density adaptation. In that context, the special case of adapting the number of components of a Gaussian mixture density approximation has been considered. For that purpose, a novel approach for mixture density adaptation based on splitting libraries and on the efficient identification of critical mixture components has been proposed.

The new estimators fill the gap between simple estimators based on, e.g. linearization, and complex numeric approaches like particle filters[12] or grid–based approaches.[13] Since the estimator performance is adjusted by specifying the maximum tolerable deviation between the true density and its approximation, the designer can trade accuracy for computational power required. As a result, economic estimators can be designed that are adequate for the given application.

For the sake of simplifying the corresponding derivations, the paper is limited to the case of scalar states. However, a generalization to vector–valued states and measurements is straightforward and already available.

REFERENCES

1. D. L. Alspach and H. W. Sorenson, "Nonlinear Bayesian Estimation Using Gaussian Sum Approximation," *IEEE Transactions on Automatic Control* **AC–17**(4), pp. 439–448, 1972.
2. W. B. Poland and R. D. Shachter, "Mixtures of Gaussians and Minimum Relative Entropy Techniques for Modeling Continuous Uncertainties," in *Uncertainty in Artificial Intelligence: Proceedings of the Ninth Conference (UAI-1993)*, Morgan Kaufmann Publishers, (San Francisco, CA), 1993.
3. N. Ueda and R. Nakano, "Deterministic Annealing EM Algorithm," *Neural Networks* **11**(2), pp. 271–282, 1998.
4. C. Musso, N. Oudjane, and F. LeGland, "Improving Regularized Particle Filters," Tech. Rep. CUED/F-INFENG/TR 380, Cambridge University Engineering Department, 2000.
5. R. Neal, "Annealed Importance Sampling," Tech. Rep. No. 9805, Department of Statistics, University of Toronto, September 1998.
6. N. Ueda, R. Nakano, Z. Ghahramani, and G. E. Hinton, "SMEM Algorithm for Mixture Models," *Neural Computation* **12**(9), pp. 2109–2128, 2000.
7. N. Vlassis and A. Likas, "A Greedy EM Algorithm for Gaussian Mixture Learning," *Neural Processing Letters* **15**(1), pp. 77–87, 2002.
8. F. C. Schweppe, *Uncertain Dynamic Systems*, Prentice–Hall, 1973.
9. A. Sankar, "Experiments with a Gaussian Merging–Splitting Algorithm for HMM Training for Speech Recognition," in *Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop, Lansdowne, VA*, 1998.
10. R. Schlüter, W. Macherey, B. Möller, and H. Ney, "A Combined Maximum Mutual Information and Maximum Likelihood Approach for Mixture Density Splitting," in *Proceedings of the Sixth European Conference on Speech Communication and Technology, Budapest, Hungary*, **4**, pp. 1715–1718, 1999.
11. J. Simonin, S. Bodin, D. Jouvet, and K. Bartkova, "Parameter Tying for Flexible Speech Recognition," in *The Forth International Conference on Spoken Language Processing (ICSLP'96), Philadelphia, PA*, 1996.
12. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A Tutorial on Particle Filters for On–line Non–linear/Non–Gaussian Bayesian Tracking," *IEEE Transactions of Signal Processing* **50**(2), pp. 174–188, 2002.
13. N. Bergman, L. Ljung, and F. Gustafsson, "Terrain Navigation Using Bayesian Statistics," *IEEE Control Systems Magazine* **19**(3), pp. 33–40, 1999.