Kullback-Leibler Approach to Gaussian Mixture Reduction

ANDREW R. RUNNALLS
University of Kent
UK

A common problem in multi-target tracking is to approximate a Gaussian mixture by one containing fewer components; similar problems can arise in integrated navigation. A common approach is successively to merge pairs of components, replacing the pair with a single Gaussian component whose moments up to second order match those of the merged pair. Salmond [1] and Williams [2, 3] have each proposed algorithms along these lines, but using different criteria for selecting the pair to be merged at each stage. The paper shows how under certain circumstances each of these pair-selection criteria can give rise to anomalous behaviour, and proposes that a key consideration should be the Kullback-Leibler (KL) discrimination of the reduced mixture with respect to the original mixture. Although computing this directly would normally be impractical, the paper shows how an easily computed upper bound can be used as a pair-selection criterion which avoids the anomalies of the earlier approaches. The behaviour of the three algorithms is compared using a high-dimensional example drawn from terrain-referenced navigation.

Manuscript received January 20, 2006; revised September 1, 2006; released for publication December 4, 2006.

IEEE Log No. T-AES/43/3/908406.

Refereeing of this contribution was handled by Y. Oshman.

This work was partially supported by QinetiQ Ltd., Farnborough, UK, under funding from the Applied Research Programme of the UK Ministry of Defence.

Author's address: Computing Laboratory, University of Kent, Canterbury CT2 7NF, Kent, UK, E-mail: (arr@kent.ac.uk).

I. INTRODUCTION

Several data fusion algorithms, usually derived in some way from the Kalman filter, represent the state of the observed system as a mixture of Gaussian distributions. An important example is the multiple hypothesis approach to tracking multiple targets where there is ambiguity in assigning observations to tracks (see for example [4, sec. 6.7]) and this is the application motivating Salmond's and Williams's papers cited below. However, Gaussian mixture approaches are also useful in integrated navigation applications where, for example, there is some ambiguity in the position fixes used to augment an inertial navigation system. This is the application motivating the present note [5, 6].

A common drawback with these Gaussian mixture algorithms is that there is a tendency for the number of components of the mixture to grow without bound; indeed, if the algorithm were simply to follow the statistical model on which the method is based, the number of components would increase exponentially over time. To combat this, various pragmatic measures must be taken to keep the number of components in check. Typically this will be achieved either by discarding components with low probability, and/or by merging components which represent similar state hypotheses.

Salmond [1] proposed a mixture reduction algorithm in which the number of components is reduced by repeatedly choosing the two components that appear to be most similar to each other, and merging them. His criterion of similarity is based on concepts from the statistical analysis of variance, and seeks to minimise the increase in "within-component" variance resulting from merging the two chosen components.

Williams [2, 3] proposed a mixture reduction algorithm based on an integrated squared difference (ISD) similarity measure, which as he points out has the big advantage that the similarity between two arbitrary Gaussian mixtures can be expressed in closed form. The algorithm he proposes uses a hill-climbing optimisation to search for a reduced mixture with the greatest similarity to the original mixture; however, to find starting points for the optimisation process, he uses a pairwise merge algorithm similar to Salmond's, but using the ISD similarity measure.

In the present paper, we propose a third variation on the pairwise-merge approach, in which the measure of similarity between two components is based on the Kullback-Leibler (KL) discrimination measure [7].

The layout is as follows. Section II introduces a brief notation for Gaussian mixtures, defines the concept of a moment-preserving merge of two or more components of such a mixture, and outlines the

pairwise-merge type of mixture reduction algorithm being considered here. Section III introduces the KL discrimination measure. Section IV describes the criterion proposed in [1] for selecting which pair of components to merge at each stage, and identifies two properties of this criterion that may be considered anomalous. Section V similarly studies the ISD criterion proposed by Williams, and identifies a property of this criterion that may be considered undesirable in some applications, particularly where the system state vector has high dimensionality. Section VI proposes a dissimilarity measure for pair selection based on KL discrimination, and explores its properties; Section VII then discusses the advantages and disadvantages of a pairwise merge algorithm based on this dissimilarity measure. Section VIII compares the operation of the Salmond, Williams, and KL reduction algorithms in reducing a high-dimensional mixture arising in terrain-referenced navigation. Finally Section IX draws conclusions.

II. GENERAL BACKGROUND

A. Notation

We represent a component of a Gaussian mixture using notation of the form $(w,\boldsymbol{\mu},P)$; this represents a component with nonnegative weight w, mean vector $\boldsymbol{\mu}$, and covariance matrix P. (We assume throughout that components' covariance matrices are strictly positive definite, and not merely nonnegative definite.) We use notation such as $\{(w_1,\boldsymbol{\mu}_1,P_1),(w_2,\boldsymbol{\mu}_2,P_2),\ldots(w_n,\boldsymbol{\mu}_n,P_n)\}$ to denote a mixture of n such components; such a mixture must satisfy $w_1+\cdots+w_n=1$, and has probability density function (pdf):

$$f(\mathbf{x})=\sum_{i=1}^{n}\frac{w_i}{\sqrt{(2\pi)^d\det P_i}}\exp\left[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_i)^T P_i^{-1}(\mathbf{x}-\boldsymbol{\mu}_i)\right]$$

where d is the dimensionality of the state vector $\mathbf{x}$. A plain (unmixed) Gaussian distribution will be written using notation such as $\{(1,\mu,P)\}$.

B. Merging Two Components

Suppose we are given a mixture of two Gaussian components:

$$\{(w_1,\boldsymbol{\mu}_1,P_1),(w_2,\boldsymbol{\mu}_2,P_2)\} \tag{1}$$

(where $w_1+w_2=1$) and that we wish to approximate this mixture as a single Gaussian. A strong candidate is the Gaussian whose zeroth, first- and second-order moments match those of (1), i.e., the Gaussian with mean vector $\boldsymbol{\mu}$ and covariance matrix P as follows:

$$\begin{aligned}\boldsymbol{\mu}&=w_1\boldsymbol{\mu}_1+w_2\boldsymbol{\mu}_2\\ P&=w_1(P_1+(\boldsymbol{\mu}_1-\boldsymbol{\mu})(\boldsymbol{\mu}_1-\boldsymbol{\mu})^T)\\ &\quad+w_2(P_2+(\boldsymbol{\mu}_2-\boldsymbol{\mu})(\boldsymbol{\mu}_2-\boldsymbol{\mu})^T)\\ &=w_1P_1+w_2P_2+w_1w_2(\boldsymbol{\mu}_1-\boldsymbol{\mu}_2)(\boldsymbol{\mu}_1-\boldsymbol{\mu}_2)^T.\end{aligned}$$

(Theorem 2 shows that $\{(1,\boldsymbol{\mu},P)\}$ is the Gaussian whose KL discrimination from the mixture (1) is minimal.)

We refer to $(1,\boldsymbol{\mu},P)$ as the moment-preserving merge of $(w_1,\boldsymbol{\mu}_1,P_1)$ and $(w_2,\boldsymbol{\mu}_2,P_2)$. More generally, we can remove the restriction that $w_1+w_2=1$: given two weighted Gaussian components $(w_i,\boldsymbol{\mu}_i,P_i)$ and $(w_j,\boldsymbol{\mu}_j,P_j)$, with $w_1+w_2\leq 1$, their moment-preserving merge is the Gaussian component $(w_{ij},\boldsymbol{\mu}_{ij},P_{ij})$ as follows (cf. [3, eqs. 2–4]):

$$w_{ij}=w_i+w_j \tag{2}$$

$$\boldsymbol{\mu}_{ij}=w_{i|ij}\boldsymbol{\mu}_i+w_{j|ij}\boldsymbol{\mu}_j \tag{3}$$

$$\begin{aligned}P_{ij}&=w_{i|ij}P_i+w_{j|ij}P_j\\ &\quad+w_{i|ij}w_{j|ij}(\boldsymbol{\mu}_i-\boldsymbol{\mu}_j)(\boldsymbol{\mu}_i-\boldsymbol{\mu}_j)^T\end{aligned} \tag{4}$$

where we write $w_{i|ij}=w_i/(w_i+w_j)$ and $w_{j|ij}=w_j/(w_i+w_j)$.

C. Mixture Reduction Algorithm

Suppose that we are given a mixture with n components, and we wish to approximate it by a mixture of m components, where $m\leq n$. In this paper, we focus on algorithms which operate in the following general way.

While more than m components remain, choose the two components that in a sense to be defined are least dissimilar, and replace them by their moment-preserving merge.

The algorithm proposed in [1, sec. 4] is of this type, using the dissimilarity measure to be described in Section IV; the algorithm proposed in [2, 3] uses an algorithm of this type to determine starting points for an optimisation procedure.

III. KULLBACK-LEIBLER DISCRIMINATION

If $f_1(\mathbf{x})$ and $f_2(\mathbf{x})$ are pdfs over $\Re^d$, the KL discrimination[1] of f_2 from f_1 is defined as

$$d_{\mathrm{kl}}(f_1,f_2)=\int_{\Re^d}f_1(\mathbf{x})\log\frac{f_1(\mathbf{x})}{f_2(\mathbf{x})}d\mathbf{x}. \tag{5}$$

[1] Also referred to as cross-entropy, KL information, or KL divergence. However, Kullback and Leibler themselves [7] and several subsequent authors use the term "divergence" to refer to $d_{\mathrm{kl}}(f_1,f_2)+d_{\mathrm{kl}}(f_2,f_1)$. It is also sometimes called the KL distance, despite not satisfying the usual requirements for a distance measure.

Although clearly $d_{\mathrm{kl}}(f,f) = 0$, and $d_{\mathrm{kl}}(f,g) \geq 0$ (cf. [8, Theorem 2.6.3], [9, Theorem 4.3.1]), in general it is not true that $d_{\mathrm{kl}}(f,g) = d_{\mathrm{kl}}(g,f)$, nor that $d_{\mathrm{kl}}(f,g) + d_{\mathrm{kl}}(g,h) \geq d_{\mathrm{kl}}(f,h)$.

To give an informal motivation for KL discrimination, suppose that we have a stream of data $x_1, x_2, \ldots$ which we assume to be independent samples either from $f(x)$ or from $g(x)$, and we wish to decide which. From a Bayesian perspective, the approach we might take is to continue drawing samples until the likelihood ratio $\prod_i (f(x_i)/g(x_i))$ exceeds some predefined threshold, say 100:1 in favour on one candidate or the other. Equivalently, we will be aiming to achieve a sample large enough that the logarithm of the likelihood ratio falls outside the bounds $\pm \log 100$. Now suppose that (unknown to us) the data stream is actually coming from $f(x)$. Then the expected value of the log-likelihood-ratio for a single sample point will be $E(\log(f(x)/g(x))) = d_{\mathrm{kl}}(f,g)$. Consequently, the expected log-likelihood-ratio for the full sample will exceed $\log 100$ provided the sample size exceeds $(\log 100)/d_{\mathrm{kl}}(f,g)$. Roughly speaking, small values of $d_{\mathrm{kl}}(f,g)$ mean that we will need large samples to distinguish f from g, and conversely.

The remainder of this section introduces theorems about KL discrimination that we use in Section VI, and can be skipped on a first reading.

THEOREM 1 *Let $g_1(\mathbf{x})$ be the d-dimensional Gaussian pdf with mean vector $\boldsymbol{\mu}_1$ and positive-definite covariance matrix P_1, and let $g_2(\mathbf{x})$ be the d-dimensional Gaussian pdf with mean vector $\boldsymbol{\mu}_2$ and positive-definite covariance matrix P_2. Then:*

$$2d_{\mathrm{kl}}(g_1,g_2) = \mathrm{tr}(P_2^{-1}[P_1 - P_2 + (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T]) + \log \frac{\det(P_2)}{\det(P_1)}.$$

For a proof see for example [9, Theorem 7.2.8].

THEOREM 2 *Let $f(\mathbf{x})$ be a pdf over d dimensions with well-defined mean $\boldsymbol{\mu}_*$ and covariance matrix P_*, where P_* is strictly positive-definite. As before, let $(1,\boldsymbol{\mu},P)$ denote the Gaussian density with mean $\boldsymbol{\mu}$ and positive-definite covariance matrix P. Then the unique minimum value of $d_{\mathrm{kl}}(f,(1,\boldsymbol{\mu},P))$ is achieved when $\boldsymbol{\mu} = \boldsymbol{\mu}_*$ and $P = P_*$.*

For a proof see the Appendix.

THEOREM 3 *If $f(\mathbf{x})$, $h_1(\mathbf{x})$ and $h_2(\mathbf{x})$ are any pdfs over d dimensions and $0 \leq w \leq 1$ then, writing $\bar{w}$ for $1-w$:*

$$d_{\mathrm{kl}}(wh_1 + \bar{w}h_2, f) \leq w d_{\mathrm{kl}}(h_1,f) + \bar{w} d_{\mathrm{kl}}(h_2,f)$$
$$d_{\mathrm{kl}}(f, wh_1 + \bar{w}h_2) \leq w d_{\mathrm{kl}}(f,h_1) + \bar{w} d_{\mathrm{kl}}(f,h_2).$$

This is a standard result. For a proof see [9, Theorem 4.3.2] or [8, Theorem 2.7.2].

THEOREM 4 *If $f_1(\mathbf{x})$, $f_2(\mathbf{x})$ and $h(\mathbf{x})$ are any pdfs over d dimensions, $0 \leq w \leq 1$ and $\bar{w} = 1 - w$, then:*

$$d_{\mathrm{kl}}(wf_1 + \bar{w}h, wf_2 + \bar{w}h) \leq w d_{\mathrm{kl}}(f_1,f_2).$$

For a proof see the Appendix.

IV. SALMOND'S CRITERION

Let $\{(w_1,\boldsymbol{\mu}_1,P_1),\ldots(w_n,\boldsymbol{\mu}_n,P_n)\}$ be an n-component Gaussian mixture, and let $\boldsymbol{\mu}$ and P be, respectively, the overall mean and the overall variance of this mixture. Clearly

$$\boldsymbol{\mu} = \sum_{i=1}^{n} w_i \boldsymbol{\mu}_i$$

while P can be written as $P = W + B$ where W is the within-components contribution to the total variance, given by

$$W = \sum_{i=1}^{n} w_i P_i$$

while B is the "between-components" contribution given by

$$B = \sum_{i=1}^{n} w_i (\boldsymbol{\mu}_i - \boldsymbol{\mu})(\boldsymbol{\mu}_i - \boldsymbol{\mu})^T.$$

When two components are replaced by their moment-preserving merge, the effect is, roughly speaking, to increase W and decrease B by a corresponding amount, leaving the total variance P unchanged. Salmond's general idea [1, sec. 4] is to choose for merging two components i and j such that the increase in W is minimised. He shows that the change in W when components i and j are replaced by their moment-preserving merge is

$$\Delta W_{ij} = \frac{w_i w_j}{w_i + w_j} (\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)^T.$$

However, ΔW_{ij} is a matrix, whereas we require a scalar dissimilarity measure. Salmond proposes using the following measure:

$$D_{\mathrm{s}}^2(i,j) = \mathrm{tr}(P^{-1} \Delta W_{ij}). \tag{6}$$

Here the trace reduces its matrix argument to a scalar, and the premultiplication by P^{-1} ensures that the resulting dissimilarity measure is invariant under linear transformations of the state space.

However, the dissimilarity measure defined in (6) has two properties that may be considered undesirable as a basis for choosing which components to merge. First, the measure depends on the means of the components, but not on their individual covariance matrices, leading to the behaviour in the following.

EXAMPLE 1 A mixture comprises three two-dimensional components $\{(1/3,\boldsymbol{\mu},P_1),(1/3, \boldsymbol{\mu} + \delta\boldsymbol{\mu},P_1),(1/3,\boldsymbol{\mu},P_2)\}$, where $\delta\boldsymbol{\mu}$ is very small

(e.g. $\delta\boldsymbol{\mu} = (0.0001, 0.0001)^T$) but P_2 is very different from P_1:

$$P_1 = \begin{pmatrix} 1 & 0.9 \\ 0.9 & 1 \end{pmatrix}, \qquad P_2 = \begin{pmatrix} 1 & -0.9 \\ -0.9 & 1 \end{pmatrix}.$$

We wish to reduce the mixture to two components. Then, using (6), we choose to merge the first and third components, yielding a merged component $(2/3, \boldsymbol{\mu}, I_2)$, where I_2 is the two-dimensional identity matrix.

The reader may well consider that in this example it would be better to merge the first two components, yielding $(2/3, \boldsymbol{\mu} + (1/2)\delta\boldsymbol{\mu}, P_1 + (1/4)\delta\boldsymbol{\mu}\delta\boldsymbol{\mu}^T)$.

The second drawback arises from the presence of the overall covariance P within (6). This has the implication that adding a new component to a mixture may alter the order in which the existing components are merged, as shown in the following example.

EXAMPLE 2 A mixture over the two dimensions (x, y) consists of four components

$$A = (0.25, (0.661, 1)^T, I_2) \tag{7}$$

$$B = (0.25, (1.339, -1)^T, I_2) \tag{8}$$

$$C = (0.25, (-0.692, 1.1)^T, I_2) \tag{9}$$

$$D = (0.25, (-1.308, -1.1)^T, I_2). \tag{10}$$

(The means of the components are shown in Fig. 1.) We wish to reduce this mixture to three components. It is readily established that the overall mean of the mixture is $(0,0)^T$, and its covariance matrix is $2.105 I_2$. From the latter fact, it follows that criterion (6) will lead us simply to merge the two components whose means are closest together, namely A and C.

Now modify the original mixture by reducing the weights of components A to D to 0.2, and adding a fifth component $E = (0.2, (0, -10)^T, I_2)$. We wish to reduce this new mixture to three components. It turns out that criterion (6) now selects components A and B for the first merge, and components C and D for the second merge. This is because, although E is a weak candidate for either merge, its inclusion in the mixture has greatly increased its overall variance in the y-direction, meaning that (6) now weights differences in x more heavily than differences in y.

V. WILLIAMS'S CRITERION

Williams [2] and Williams and Maybeck [3] propose a method of Gaussian mixture reduction based on the ISD measure of the dissimilarity between two pdfs $f_1(\mathbf{x})$ and $f_2(\mathbf{x})$:

$$J_S = \int (f_1(\mathbf{x}) - f_2(\mathbf{x}))^2 d\mathbf{x}$$

(cf. [3, eq. 4]. This has the important property that the dissimilarity between two arbitrary Gaussian mixtures can be expressed in closed form (given in [3, eq. 10])—a property regrettably not shared by the measure proposed in the present paper.

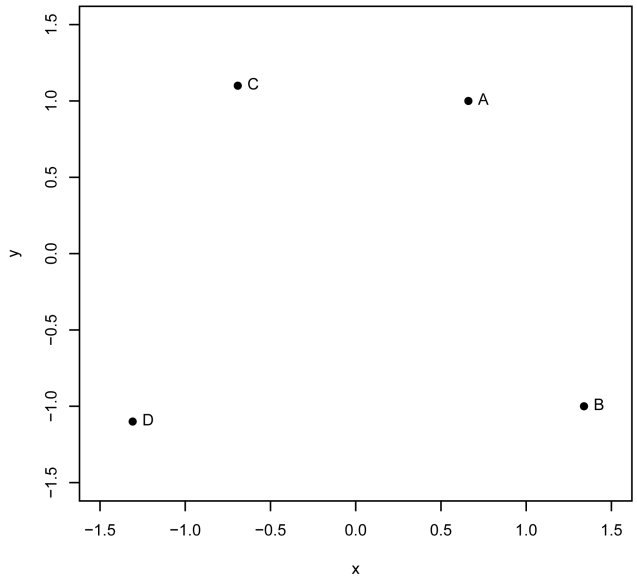


Fig. 1. Means of components in Example 2.

Their algorithm for reducing an n-component mixture to an m-component mixture ($m \leq n$) can be summarised as follows.

1) While more than m components remain consider all possible operations of the following two kinds:

a) deleting a component and renormalising the remaining mixture,

b) replacing a pair of components with their moment-preserving merge,

and in each case evaluate the ISD-dissimilarity of the resulting mixture from the original mixture. Apply the operation for which this dissimilarity is a minimum.

2) Use the resulting m-component mixture as the starting point for gradient-based optimisation technique, to seek an m-component mixture with lower dissimilarity to the original mixture.

The authors note that the optimisation at Step 2 will seek a local minimum rather than the global minimum, hence the need to choose the starting point carefully.

The ISD cost measure circumvents both of the drawbacks of Salmond's criterion. First, the measure depends explicitly on the covariance matrices as well as the means of the components. Second, the cost incurred by merging two components depends only on the parameters of those components, and not on other characteristics of the mixture of which they form a part. Consequently, the anomalies observed in Examples 1 and 2 do not arise.

However, the ISD criterion leads to puzzling behaviour of its own. To illustrate this, we focus on mixtures where the components are radially

symmetric, i.e., the covariance matrices are multiples of the identity matrix. Consider first the case where the starting mixture is $\{(w,\boldsymbol{\mu}-c\sigma\mathbf{u},\sigma^2 I_d),(w,\boldsymbol{\mu}+c\sigma\mathbf{u},\sigma^2 I_d)\}$, where $\boldsymbol{\mu}$ is arbitrary and $\mathbf{u}$ is a d-dimensional unit vector. The means of the two components of this mixture are distance $2c\sigma$ apart.

In this case it follows from [3, eq. 12] that the ISD cost of deleting one of the components (and raising the other component to unit weight) is given by

$$J_S = \frac{4w^2}{\sigma^d\sqrt{(4\pi)^d}} h_D(c) \tag{11}$$

where

$$h_D(c) = \tfrac{1}{2}(1-\exp(-c^2)) \tag{12}$$

while the cost of replacing the two components by their moment-preserving merge, namely $(2w,\boldsymbol{\mu},\sigma^2(I+c^2\mathbf{u}\mathbf{u}^T))$, is

$$J_S = \frac{4w^2}{\sigma^d\sqrt{(4\pi)^d}} h_M(c) \tag{13}$$

where

$$h_M(c) = \frac{1}{2}(1+\exp(-c^2)) + \frac{1}{\sqrt{1+c^2}} - \frac{2\sqrt{2}}{\sqrt{2+c^2}}\exp\left(-\frac{c^2}{2(2+c^2)}\right). \tag{14}$$

The functions $h_M(c)$ and $h_D(c)$ are both zero for $c=0$ and as c increases, both functions increase monotonically, tending towards $1/2$ as $c\to\infty$. It can be shown that $h_D(c) > h_M(c)$ except when c is zero, so the deletion option is not considered further.

In the example under consideration, σ acts simply as a scale factor, but it nevertheless appears in (13), raised moreover to the power d. This leads to some surprising behaviour in the way in which Williams's algorithm selects pairwise merges, as in the following twelve-dimensional example. (It is not unusual in inertial navigation applications for the state vector to have 15 or more dimensions.)

EXAMPLE 3 A mixture over the space $(x_1,\ldots x_{12})$ comprises four components

$$A = (0.25,(-20,-0.5,0,\ldots,0)^T,I_{12}) \tag{15}$$

$$B = (0.25,(-20,0.5,0,\ldots 0)^T,I_{12}) \tag{16}$$

$$C = (0.25,(20,-10,0,\ldots 0)^T,4I_{12}) \tag{17}$$

$$D = (0.25,(20,10,0,\ldots 0)^T,4I_{12}) \tag{18}$$

where in each mean vector the ellipsis … comprises eight zeroes. Note that components A and B have negligible probability within the region where $x_1 > 0$, and C and D have negligible probability within the region $x_1 < 0$.

Assume that we wish to reduce this four-component mixture to three components. Now, according to (13) the cost of replacing components A and B by their moment-preserving merge is

$$J_S = \frac{1}{4(4\pi)^6} h_M(0.5) \approx 6.39\times 10^{-12}$$

while the cost of replacing C and D by their moment-preserving merge is

$$J_S = \frac{1}{4\times 2^{12}(4\pi)^6} h_M(5) \approx 5.48\times 10^{-12}.$$

Consequently, the Williams algorithm will choose to merge C and D rather than merging A and B.

This is despite the fact that mixture of A and B is already unimodal, and is very similar in shape to their moment-preserving merge. In contrast, the mixture of C and D is decidedly bimodal, the means of these components being ten standard deviations apart.

In fact, direct numerical integration reveals that the KL discrimination of the mixture $\{A,B,CD\}$ (where CD is the result of merging C and D) from the original mixture $\{A,B,C,D\}$ is 0.468, so (following the discussion at the start of Section III) it would need only about 10 samples from the original mixture to distinguish it from $\{A,B,CD\}$ with a likelihood ratio of 100:1. In contrast, the discrimination of the mixture $\{AB,C,D\}$ from the original is only 7.52×10^{-5}, so requiring an average of over 60000 samples to achieve the same likelihood ratio.

The phenomenon illustrated by this example arises from the scale dependency of the ISD cost measure, as exhibited by the presence of the scale factor σ in the cost measure of (13). It is particularly pronounced in spaces of high dimensionality, and means that—at least in some applications—the ISD cost measure may not be suitable as a basis for Gaussian mixture reduction.

In Section VI we present an alternative criterion for mixture reduction which does not exhibit scale dependency, and which also avoids the drawbacks of Salmond's criterion.

VI. A DISSIMILARITY MEASURE BASED ON KL DISCRIMINATION

A. Motivation

At each iteration of the algorithm outlined in Section IIC, we wish to choose two components from the mixture for merging. Our ultimate objective is to find a weighted mixture of m Gaussian components in such a way as to keep the KL discrimination of the m-component mixture from the original n-component mixture as small as possible, subject to being able to accomplish this with an algorithm that is computationally reasonably fast. A reasonable criterion, therefore, is to choose two components in such a way as to minimise the KL discrimination of

the mixture after the merge from the mixture before the merge.

Unfortunately, there appears to be no closed-form expression for the KL discrimination of one (nontrivial) Gaussian mixture from another. (This fact deterred Williams [2, sec. 3.3.1.4] from pursuing a cost measure based on KL discrimination; were it not for this, he says it would be the "ideal cost function" for Gaussian mixture reduction.) However, Section III provided two theorems that enable us to put an upper bound on the discrimination of the mixture after the merge from the mixture before the merge. This leads us to the dissimilarity measure $B((w_i,\boldsymbol{\mu}_i,P_i),(w_j,\boldsymbol{\mu}_j,P_j))$ now to be defined.

B. Definition of $B((w_i,\boldsymbol{\mu}_i,P_i),(w_j,\boldsymbol{\mu}_j,P_j))$

Theorem 4 tells us that the discrimination of the mixture after merging components i and j from the mixture before the merge will not exceed $w_i + w_j$ times the discrimination of the single Gaussian $\{(1,\boldsymbol{\mu}_{ij},P_{ij})\}$ from the (normalised) mixture $\{(w_{i|ij},\boldsymbol{\mu}_i,P_i),(w_{j|ij},\boldsymbol{\mu}_j,P_j)\}$. (Refer to Section IIB for notation.)

Moreover Theorem 3 tells us that this discrimination, which we write as

$$d_{\mathrm{kl}}(\{(w_{i|ij},\boldsymbol{\mu}_i,P_i),(w_{j|ij},\boldsymbol{\mu}_j,P_j)\},\{(1,\boldsymbol{\mu}_{ij},P_{ij})\})$$

will not exceed

$$\frac{1}{w_i+w_j}(w_i d_{\mathrm{kl}}(\{(1,\boldsymbol{\mu}_i,P_i)\},\{(1,\boldsymbol{\mu}_{ij},P_{ij})\}) + w_j d_{\mathrm{kl}}(\{(1,\boldsymbol{\mu}_j,P_j)\},\{(1,\boldsymbol{\mu}_{ij},P_{ij})\})).$$

Putting these together, it follows that the discrimination of the mixture following the merge from the mixture before the merge will not exceed:

$$\begin{aligned}B((w_i,\boldsymbol{\mu}_i,P_i),(w_j,\boldsymbol{\mu}_j,P_j)) &= w_i d_{\mathrm{kl}}(\{(1,\boldsymbol{\mu}_i,P_i)\},\{(1,\boldsymbol{\mu}_{ij},P_{ij})\}) \\ &\quad + w_j d_{\mathrm{kl}}(\{(1,\boldsymbol{\mu}_j,P_j)\},\{(1,\boldsymbol{\mu}_{ij},P_{ij})\}). \qquad (19)\end{aligned}$$

We now show how this upper bound $B((w_i,\boldsymbol{\mu}_i,P_i),(w_j,\boldsymbol{\mu}_j,P_j))$ can be computed in practice. From Theorem 1, we have

$$\begin{aligned}2d_{\mathrm{kl}}(\{(1,\boldsymbol{\mu}_i,P_i)\},\{(1,\boldsymbol{\mu}_{ij},P_{ij})\}) &= \mathrm{tr}(P_{ij}^{-1}[P_i - P_{ij} + (\boldsymbol{\mu}_i-\boldsymbol{\mu}_{ij})(\boldsymbol{\mu}_i-\boldsymbol{\mu}_{ij})^T]) + \log\frac{\det(P_{ij})}{\det(P_i)} \\ &= \mathrm{tr}(P_{ij}^{-1}[P_i - P_{ij} + w_{j|ij}^2(\boldsymbol{\mu}_i-\boldsymbol{\mu}_j)(\boldsymbol{\mu}_i-\boldsymbol{\mu}_j)^T]) + \log\det(P_{ij}) - \log\det(P_i).\end{aligned}$$

A corresponding expression can be obtained for $2d_{\mathrm{kl}}(\{(1,\boldsymbol{\mu}_j,P_j)\},\{(1,\boldsymbol{\mu}_{ij},P_{ij})\})$ by replacing P_i by P_j and $w_{j|ij}$ by $w_{i|ij}$.

Consequently, substituting into (19) and using the fact that trace is a linear operator, we find

$$\begin{aligned}2B((w_i,\boldsymbol{\mu}_i,P_i),(w_j,\boldsymbol{\mu}_j,P_j)) &= \mathrm{tr}(P_{ij}^{-1}\breve{P}_{ij}) + (w_i+w_j)\log\det(P_{ij}) \\ &\quad - w_i\log\det(P_i) - w_j\log\det(P_j)\end{aligned}$$

where

$$\breve{P}_{ij} = w_iP_i + w_jP_j - (w_i+w_j)P_{ij} + \frac{w_iw_j}{w_i+w_j}(\boldsymbol{\mu}_i-\boldsymbol{\mu}_j)(\boldsymbol{\mu}_i-\boldsymbol{\mu}_j)^T. \qquad (20)$$

If we now substitute the expression for P_{ij} in (4) into (20), we find that $\breve{P}_{ij}$ equals zero. Therefore:

$$\begin{aligned}B((w_i,\boldsymbol{\mu}_i,P_i),(w_j,\boldsymbol{\mu}_j,P_j)) &= \tfrac{1}{2}[(w_i+w_j)\log\det(P_{ij}) \\ &\quad - w_i\log\det(P_i) - w_j\log\det(P_j)]. \qquad (21)\end{aligned}$$

C. Properties of $B(i,j)$

In the remainder of this paper we write $B(i,j)$ as a shorthand for $B((w_i,\boldsymbol{\mu}_i,P_i),(w_j,\boldsymbol{\mu}_j,P_j))$.

First of all, it is clear that the function is symmetric: $B(i,j) = B(j,i)$.

In one dimension, with $P_i = (\sigma_i^2)$, $P_j = (\sigma_j^2)$, (21) becomes

$$\frac{2B(i,j)}{w_i+w_j} = \log\left[w_{i|ij}\left(\frac{\sigma_i^2}{\sigma_j^2}\right)^{w_{j|ij}} + w_{j|ij}\left(\frac{\sigma_j^2}{\sigma_i^2}\right)^{w_{i|ij}} + w_{i|ij}w_{j|ij}\frac{(\mu_i-\mu_j)^2}{\sigma_i^{2w_{i|ij}}\sigma_j^{2w_{j|ij}}}\right] \qquad (22)$$

showing that in this case $B(i,j)$ depends only on the weights w_i, w_j and on the dimensionless quantities σ_i^2/σ_j^2 and $(\mu_i-\mu_j)/(\sigma_i^{w_{i|ij}}\sigma_j^{w_{j|ij}})$.

This conclusion can be extended to more than one dimension using a simultaneous diagonalisation procedure. Since P_i and P_j are both positive definite, by a small variation of the procedure described in [10, sec. 1c.3(ii)] we can find a square unitary[2] matrix U_{ij} and diagonal matrices D_i and D_j, with all their diagonal elements positive, such that:

$$P_i = U_{ij}^{-1}D_iU_{ij}^{-T}, \qquad P_j = U_{ij}^{-1}D_jU_{ij}^{-T}. \qquad (23)$$

Substituting from these equations into (4) we get

$$\begin{aligned}P_{ij} &= w_{i|ij}U_{ij}^{-1}D_iU_{ij}^{-T} + w_{j|ij}U_{ij}^{-1}D_jU_{ij}^{-T} + w_{i|ij}w_{j|ij}(\boldsymbol{\mu}_i-\boldsymbol{\mu}_j)(\boldsymbol{\mu}_i-\boldsymbol{\mu}_j)^T \\ &= U_{ij}^{-1}(w_{i|ij}D_i + w_{j|ij}D_j + w_{i|ij}w_{j|ij}\mathbf{u}_{ij}\mathbf{u}_{ij}^T)U_{ij}^{-T} \qquad (24)\end{aligned}$$

[2] i.e., a matrix with determinant unity. Note that the notation U stands for "unitary": U_{ij} is not in general upper triangular.

where we have written $\mathbf{u}_{ij}$ for $U_{ij}(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)$. Now, substituting from (23) and (24) into (21) and using the fact that $\det(AB) = \det A \times \det B$, we get

$$\begin{aligned}\frac{2B(i,j)}{w_i + w_j} &= \log\det(w_{i|ij}D_i + w_{j|ij}D_j + w_{i|ij}w_{j|ij}\mathbf{u}_{ij}\mathbf{u}_{ij}^T) \\ &\quad - w_{i|ij}\log\det(D_i) - w_{j|ij}\log\det(D_j) \\ &= \log\det(w_{i|ij}D_i + w_{j|ij}D_j + w_{i|ij}w_{j|ij}\mathbf{u}_{ij}\mathbf{u}_{ij}^T) \\ &\quad - \log\det(D_i^{w_{i|ij}}) - \log\det(D_j^{w_{j|ij}}) \\ &= \log\det(D_i^{-w_{i|ij}}D_j^{-w_{j|ij}} \\ &\quad \times [w_{i|ij}D_i + w_{j|ij}D_j + w_{i|ij}w_{j|ij}\mathbf{u}_{ij}\mathbf{u}_{ij}^T]) \\ &= \log\det(w_{i|ij}(D_iD_j^{-1})^{w_{j|ij}} + w_{j|ij}(D_jD_i^{-1})^{w_{i|ij}} \\ &\quad + w_{i|ij}w_{j|ij}D_i^{-w_{i|ij}}D_j^{-w_{j|ij}}\mathbf{u}_{ij}\mathbf{u}_{ij}^T) \end{aligned} \tag{25}$$

where the notation D^α denotes the diagonal matrix whose elements are the corresponding elements of D raised to the power α. Thus $B(i,j)$ depends only on the weights w_i, w_j and the dimensionless quantities $D_iD_j^{-1}$ and $D_i^{-w_{i|ij}/2}D_j^{-w_{j|ij}/2}\mathbf{u}_{ij}$.

By inspection of (25) it can be seen that $B(i,j) = 0$ if and only if at least one of the following three conditions holds: 1) $w_i = 0$, 2) $w_j = 0$, or 3) $\boldsymbol{\mu}_i = \boldsymbol{\mu}_j$ and $P_i = P_j$. A counterexample to the triangle inequality is given (in one dimension) by putting $w_1 = w_2 = w_3 = 1/3$, $\mu_1 = \mu_2 = \mu_3$ and $\sigma_3 = 2\sigma_2 = 4\sigma_1$; then from (22) we have $B(1,2) = B(2,3) \approx 0.07$ but $B(1,3) \approx 0.25$.

VII. DISCUSSION

We propose that, in each iteration of the algorithm outlined in Section IIC, we select for merging two components i and j, $i \neq j$, such that $B(i,j)$ is minimised. The dissimilarity measure $B(i,j)$ as given by (21) is reasonably easy to compute, with computational complexity at most $\mathcal{O}(d^3)$. Consequently, if our task is to reduce a mixture of n components to a mixture with cn components, where $c < 1$ is a constant, this will have total computational complexity of $\mathcal{O}(n^3d^3)$.

This criterion has qualitatively the right properties. Roughly speaking, it will tend to select for merging:

1) components with low weights. Note how the weights appear outside the logarithms in (21), and so can have a dominant effect,

2) components whose means are close together in relation to their variances, as measured by the length of the vector $D_i^{-w_{i|ij}/2}D_j^{-w_{j|ij}/2}\mathbf{u}_{ij}$ (cf. (25)),

3) components whose covariance matrices are similar, in the sense that the term $D_iD_j^{-1}$ in (25) is close to the identity matrix.

The $B(i,j)$ criterion avoids the drawbacks of Salmond's criterion, in that 1) it depends explicitly on the covariance matrices of components i and j, and will avoid merging components where these are very different, and 2) adding a new component to a mixture cannot alter the order in which existing components are merged. Nor does the $B(i,j)$ criterion exhibit the scale dependency of the ISD measure; for example, corresponding to (13), we get simply

$$B(i,j) = w\log(1 + c^2)$$

which does not depend on σ, or indeed on d. Consequently, in Example 3 we have $B(C,D)/B(A,B) \approx 14.6$, so A would certainly be merged with B in preference to merging C and D.

We make no claims for optimality for the resulting algorithm, but it is straightforward, and at each iteration we know that the KL discrimination of the postiteration mixture from the preiteration mixture cannot exceed $B(i,j)$.

The fact that $B(i,j)$ is merely an upper bound on the KL discrimination, rather than an exact value, is admittedly a drawback. Moreover, since KL discrimination does not satisfy the triangle inequality, there is no simple way of bounding the discrimination that arises over the course of two or more iterations of the algorithm. However, obtaining a direct estimate of the KL bound would appear to require a numerical method, e.g. numerical integration. Worse, this integration would need to be carried out multiple times: $\mathcal{O}(n^3)$ times if, as above, our task is to reduce n components to cn components. In many applications this will be computationally prohibitive. A possible compromise approach would be to use the $B(i,j)$ criterion to compile a shortlist of possible component merges, selection from within this shortlist being by direct numerical integration.

VIII. A PRACTICAL EXAMPLE

This section compares the operation of Salmond's criterion, the ISD measure, and the merging criterion introduced in Section VI as applied to reducing a Gaussian mixture over 15 dimensions from its original 16 components down to four components. This dataset arises from an application to terrain-referenced navigation, specifically from the simulation run previously reported in [5, Fig. 7].

The state vector comprises three elements of position error (north, up, and down), three components of velocity error, three platform misalignment angles, three accelerometer biases, and three gyro drift terms. However, for ease of visualisation, we illustrate here the algorithms' operation by examining the marginal distribution over the two horizontal components of position error.

Fig. 2(a) shows the starting mixture. Each Gaussian component is represented by an elliptical

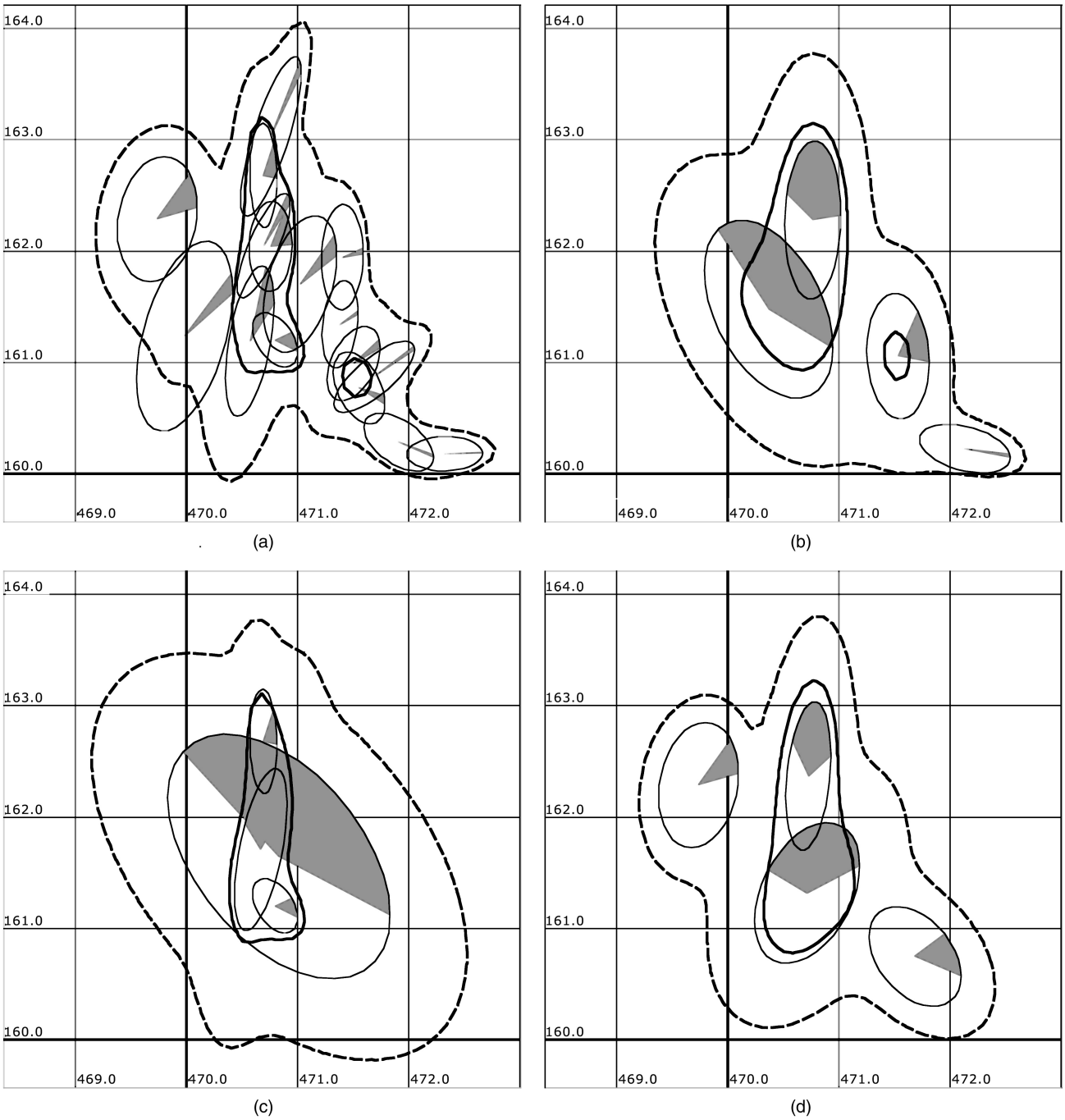


Fig. 2. Comparison of reduction algorithms applied to 16-component mixture over 15 dimensions. (a) Original mixture. (b) Salmond reduction. (c) Williams-Maybeck reduction. (d) KL bound reduction.

contour enclosing 50% of its probability volume. In each ellipse, a sector is shown shaded: the proportion of the area of the ellipse thus shaded represents the component's weight within the mixture. The thicker curves in the figure are two contours of the mixture as a whole: the dashed line encloses 95% of the mixture's volume, while the solid line encloses 50% of its volume (within two regions).

The mixture represents the navigation system's state estimate just a few seconds after terrain-referenced navigation started. Consequently there is still considerable uncertainty about the aircraft's position: the graticule in the figure comprises 1 km squares.

Fig. 2(b), (c), and (d) show the result of reducing this mixture to four components using the algorithms considered earlier in the paper. In Fig. 2(b), the reduction uses Salmond's criterion. Fig. 2(c) shows the result of applying the algorithm described in [3, sec. 4], except that (for comparability with the other algorithms) only pairwise merges of components are considered: i.e., the option of deleting components

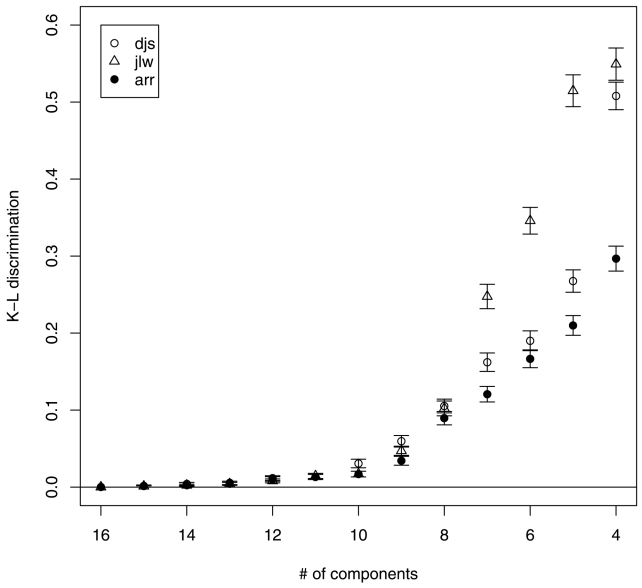


Fig. 3. KL discrimination of reduced mixture with respect to original mixture.

entirely is eschewed. Finally Fig. 2(d) shows the result of applying the $B(i,j)$ criterion. It is noted that the Williams-Maybeck reduction preserves the shape of the 50% mixture contour well (except for the loss of its secondary peak), but creates the greatest distortion of the 95% contour. In contrast, the reduction using $B(i,j)$ is the best at preserving the shape of the 95% contour, but causes greater distortion to the 50% contour. This doubtless reflects the fact that the Williams-Maybeck criterion is aiming to minimise absolute differences of the reduced pdf from the original, where the KL bound aims to avoid large ratio errors. The behaviour of Salmond's criterion is intermediate between the others, though interestingly it is the only one to preserve the secondary peak in the 50% contour (although shifting it somewhat to the north).

Fig. 3 considers the KL discrimination of the reduced mixture with respect to the original mixture, and shows how it evolves as the merging process progresses. Results for Salmond's criterion are shown as hollow circles, for the Williams-Maybeck measure as triangles, and for the $B(i,j)$ criterion as solid disks. The KL discrimination was calculated using Monte Carlo integration using 100,000 points drawn from the original mixture; to improve comparability, the same points were used for each integration. The figure includes $\pm 2\sigma$ tolerance bounds for each plotted point.

From the figure, it is evident that there is very little difference in the performance of the algorithms as they reduce the mixture down to 11 components. At this stage, the KL discrimination is about 0.013 for each algorithm, which means that it will require about 350 samples from the "true" mixture to distinguish it from the reduced one with an expected log-likelihood-ratio of $\log 100$. As the number of components is further reduced down to four, the discrimination increases more rapidly, reaching 0.51 for the Salmond criterion, 0.55 for the Williams-Maybeck criterion, and 0.30 for the $B(i,j)$ criterion, corresponding to sample sizes of 9, 8, and 15, respectively.

IX. CONCLUSION

This paper has examined two algorithms proposed in the literature for reducing a Gaussian mixture to a mixture with fewer components, namely those due to Salmond [1] and to Williams and Maybeck [2, 3]. An element of both of these algorithms is successively to merge pairs of components, at each stage replacing the merged pair by a single Gaussian component with the same moments up to the second order.

It has been shown that each of these algorithms can give rise to anomalous behaviour in the following certain circumstances.

1) Salmond's algorithm chooses for merging the pair of components whose means are closest together, even if their covariance matrices are very different.
2) In Salmond's algorithm, adding a new component to a mixture can alter the order in which existing components are merged, even if—indeed, especially if—the new component is far remote from the existing components and is therefore not itself a candidate for merging.
3) The Williams algorithm has a tendency to select for merging a pair of components with large variances, even if their means are much further apart (in relation to their standard deviations) than another pair of components with smaller variances. This effect is particularly pronounced with state vectors of high dimension.

The paper also proposed a new algorithm, again based on pairwise merging of components, but in which the choice of components for merging is based on an easily-computed upper bound of the KL discrimination of the postmerge mixture with respect to the premerge mixture, as defined in (21). It has been shown that this criterion avoids the anomalies described above.

An indicative example has been presented, using a dataset derived from terrain-referenced navigation, in which it is required to reduce a 16-component mixture over 15 dimensions down to four components. The final mixture arrived at by each of the three algorithms has been illustrated, along with data on the KL discrimination of the reduced mixture with respect to the original mixture, showing how this grows as the reduction process proceeds. In the example, the final

KL discrimination was over 30% lower using the new algorithm than with either of the others.

Further work is desirable to compare at greater length the performance of the algorithms considered here within particular application scenarios. A possible approach to this would be to repeat analyses along the lines of Section VIII for a large sample of mixture reduction problems within the particular application area.

APPENDIX. PROOFS OF THEOREMS

In each proof we use the standard inequality $\log x \leq x - 1$.

THEOREM 2 PROOF *Using* (5) *it is straightforward to show that*

$$\begin{aligned}
&2d_{\mathrm{kl}}(f,(1,\boldsymbol{\mu},P)) \\
&\quad = 2\int_{\Re^d} f(\mathbf{x})\log f(\mathbf{x})d\mathbf{x} + d\log 2\pi \\
&\qquad + \log\det P + \int_{\Re^d}(\mathbf{x}-\boldsymbol{\mu})^T P^{-1}(\mathbf{x}-\boldsymbol{\mu})f(\mathbf{x})d\mathbf{x} \\
&\quad = 2\int_{\Re^d} f(\mathbf{x})\log f(\mathbf{x})d\mathbf{x} + d\log 2\pi \\
&\qquad + \log\det P + \mathrm{tr}\left[P^{-1}\int_{\Re^d}(\mathbf{x}-\boldsymbol{\mu})(\mathbf{x}-\boldsymbol{\mu})^T f(\mathbf{x})d\mathbf{x}\right]
\end{aligned}$$

where we have used the standard identity $\mathbf{v}^T M\mathbf{v} = \mathrm{tr}(M\mathbf{v}\mathbf{v}^T)$ *(cf.* [10, *p.* 34]*). Now, writing* $\Delta\boldsymbol{\mu} = \boldsymbol{\mu} - \boldsymbol{\mu}_*$ *we have*

$$\int_{\Re^d}(\mathbf{x}-\boldsymbol{\mu})(\mathbf{x}-\boldsymbol{\mu})^T f(\mathbf{x})d\mathbf{x} = P_* + \Delta\boldsymbol{\mu}\Delta\boldsymbol{\mu}^T.$$

Consequently, if $\lambda_1,\ldots\lambda_d$ *are the eigenvalues of* $P^{-1}P_*$*, we have*

$$\begin{aligned}
&2[d_{\mathrm{kl}}(f,(1,\boldsymbol{\mu},P)) - d_{\mathrm{kl}}(f,(1,\boldsymbol{\mu}_*,P_*))] \\
&\quad = \log\det P - \log\det P_* \\
&\qquad + \mathrm{tr}[P^{-1}(P_* + \Delta\boldsymbol{\mu}\Delta\boldsymbol{\mu}^T)] - \mathrm{tr}[P_*^{-1}P_*] \\
&\quad = -\log\det(P^{-1}P_*) \\
&\qquad + \mathrm{tr}(P^{-1}P_*) + \mathrm{tr}(P^{-1}\Delta\boldsymbol{\mu}\Delta\boldsymbol{\mu}^T) - d \\
&\quad = -\log\prod_{i=1}^{d}\lambda_i + \sum_{i=1}^{d}\lambda_i - d + \mathrm{tr}(P^{-1}\Delta\boldsymbol{\mu}\Delta\boldsymbol{\mu}^T) \\
&\quad = \sum_{i=1}^{d}(-\log\lambda_i + \lambda_i - 1) + \mathrm{tr}(P^{-1}\Delta\boldsymbol{\mu}\Delta\boldsymbol{\mu}^T) \\
&\quad \geq 0
\end{aligned}$$

with equality only if $\Delta\boldsymbol{\mu} = 0$ *and* $\lambda_i = 1$ *for* $i = 1,\ldots d$*, i.e., if* $P^{-1}P_* = I$*. This proves the theorem.*

THEOREM 4 PROOF *The proof is similar to that of* [9, *Theorem* 4.3.3]*. We have*

$$\begin{aligned}
&wd_{\mathrm{kl}}(f_1,f_2) - d_{\mathrm{kl}}(wf_1 + \bar{w}h, wf_2 + \bar{w}h) \\
&\quad = w\int_{\Re^d} f_1(\mathbf{x})\log\frac{f_1(\mathbf{x})}{f_2(\mathbf{x})}d\mathbf{x} \\
&\qquad - \int_{\Re^d}(wf_1(\mathbf{x}) + \bar{w}h(\mathbf{x}))\log\frac{wf_1(\mathbf{x}) + \bar{w}h(\mathbf{x})}{wf_2(\mathbf{x}) + \bar{w}h(\mathbf{x})}d\mathbf{x} \\
&\quad = \int_{\Re^d} wf_1(\mathbf{x})\log\frac{f_1(\mathbf{x})[wf_2(\mathbf{x}) + \bar{w}h(\mathbf{x})]}{f_2(\mathbf{x})[wf_1(\mathbf{x}) + \bar{w}h(\mathbf{x})]}d\mathbf{x} \\
&\qquad + \int_{\Re^d}\bar{w}h(\mathbf{x})\log\frac{wf_2(\mathbf{x}) + \bar{w}h(\mathbf{x})}{wf_1(\mathbf{x}) + \bar{w}h(\mathbf{x})}d\mathbf{x} \\
&\quad \geq \int_{\Re^d} wf_1(\mathbf{x})\left[1 - \frac{f_2(\mathbf{x})[wf_1(\mathbf{x}) + \bar{w}h(\mathbf{x})]}{f_1(\mathbf{x})[wf_2(\mathbf{x}) + \bar{w}h(\mathbf{x})]}\right]d\mathbf{x} \\
&\qquad + \int_{\Re^d}\bar{w}h(\mathbf{x})\left[1 - \frac{wf_1(\mathbf{x}) + \bar{w}h(\mathbf{x})}{wf_2(\mathbf{x}) + \bar{w}h(\mathbf{x})}\right]d\mathbf{x} \\
&\quad = 1 - \int_{\Re^d}(wf_2(\mathbf{x}) + \bar{w}h(\mathbf{x}))\frac{wf_1(\mathbf{x}) + \bar{w}h(\mathbf{x})}{wf_2(\mathbf{x}) + \bar{w}h(\mathbf{x})}d\mathbf{x} \\
&\quad = 0.
\end{aligned}$$

ACKNOWLEDGMENT

The author would like to thank David Salmond for helpful comments on a draft of this paper.

REFERENCES

[1] Salmond, D. J.
Mixture reduction algorithms for target tracking in clutter.
Proceedings of SPIE, Signal and Data Processing of Small Targets, vol. 1305, 1990, 434–445.

[2] Williams, J. L.
Gaussian mixture reduction for tracking multiple maneuvering targets in clutter.
Master's thesis, M.S.E.E. Thesis, Air Force Institute of Technology, Wright-Patterson Air Force Base, OH, 2003.

[3] Williams, J. L., and Maybeck, P. S.
Cost-function-based Gaussian mixture reduction.
In *Sixth International Conference on Information Fusion*, ISIF, 2003.

[4] Blackman, S., and Popoli, R.
Design and Analysis of Modern Tracking Systems.
Norwood, MA: Artech House, 1999.

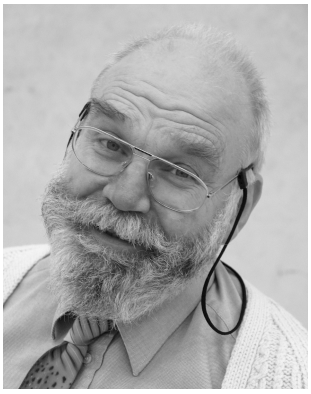
[5] Groves, P. D., Handley, R. J., and Runnalls, A. R.
Optimising the integration of terrain-referenced navigation with INS and GPS.
In *Proceedings of Institute of Navigation National GNSS2004*, Long Beach, CA, Sept. 2004, 1048–1059.

[6] Runnalls, A. R., Groves, P. D., and Handley, R. J.
Terrain-referenced navigation using the IGMAP data fusion algorithm.
In *61st Annual Meeting of the Institute of Navigation*, Institute of Navigation, Fairfax, VA, June 2005, 976–987.

[7] Kullback, S., and Leibler, R. A.
On information and sufficiency.
Annals of Mathematical Statistics, **22** (1951), 79–86.

[8] Cover, T. M., and Thomas, J. A.
Elements of Information Theory.
New York: Wiley, 1991.

[9] Blahut, R. E.
Principles and Practice of Information Theory.
Reading, MA: Addison-Wesley, 1987.
[10] Rao, C. R.
Linear Statistical Inference and its Applications (2nd ed.).
New York: Wiley, 1973.



Andrew R. Runnalls received his M.A. and Ph.D. degrees from Cambridge University, and was a Research Fellow of Gonville and Caius College. For ten years he was with the guidance systems division of GEC Avionics at Rochester, UK, where he played a key part in the development of the SPARTAN terrain-referenced navigation system, which was flight-trialled in F16, Tornado, A6 and other aircraft. He joined the Computing Laboratory of the University of Kent in 1988, where he has continued to work on statistical data fusion techniques, with particular interests in terrain-referenced navigation and maritime tracking. He is a Fellow of the Royal Institute of Navigation.