Generalized Information Representation and Compression Using Covariance Union

Ottmar Bochardt[b], Ryan Calhoun[a], Simon Julier[c], Jeffrey Uhlmann[a]

[a] Dept. of Computer Science, University of Missouri-Columbia;
[b] IDAK Industries, Canada; [c] ITT AES / Naval Research Laboratory

Abstract - ***In this paper we consider the use of Covariance Union (CU) with multi-hypothesis techniques (MHT) and Gaussian Mixture Models (GMMs) to generalize the conventional mean and covariance representation of information. More specifically, we address the representation of multimodal information using multiple mean and covariance estimates. A significant challenge is to define a rigorous fusion algorithm that can bound the complexity of the filtering process. This requires a mechanism for subsuming subsets of modes into single modes so that the complexity of the representation satisfies a specified upper bound. We discuss how this can be accomplished using CU. The practical challenge is to develop efficient implementations of the CU algorithm. Because of the novelty of the CU algorithm, there are no existing real-time implementations for use in real applications. In this paper we address this deficiency by considering a general-purpose implementation of the CU algorithm based on general nonlinear optimization techniques. Computational results are reported.***

Keywords: Covariance Intersection, Covariance Union, Data Fusion, Kalman Filter, Multimodal distributions.

1 Introduction

Level-1 information management has matured significantly over the last decade with the development of rigorous algorithms that are robust to the effects of unmodeled correlations and corrupt and/or spurious information in the context of general distributed data fusion networks. Despite the dramatic theoretical and practical results in the Level-1 arena, there have been very few inroads made into higher level information management applications. This is due in large measure to the discrepancy between the relatively simple types of information encountered in low level tracking and control applications and the much more varied and richer forms of information that must be processed in high level applications.

In this paper we explore a methodology for generalizing the unimodal information representation scheme used in Level-1 contexts to permit the representation of information that has a more complicated multimodal structure. This is accomplished by the use of a set of unimodal state estimates to capture the multiplicity of possible states of the target of interest. The challenge is to be able to bound the computational complexity issues that arise from this approach. In this paper we describe how a mechanism called Covariance Union (CU) [5, 2] can be applied to reduce the complexity of a multimodal representation to satisfy a fixed complexity budget while rigorously guaranteeing information integrity.

The structure of the paper is as follows: Section 1 discusses the issue of information representation. Section 2 discusses the need for an information compression mechanism to bound the computational complexity of the fusion process. CU is shown to be a solution to this problem. Section 3 discusses computational issues that must be addressed in order for CU to be applied in practice. Practical algorithms for implementing CU are described. Section 5 provides experimental results demonstrating the application of CU. Section 6 discusses the results presented in the paper.

2 Information Representation

Determining how to represent information and uncertainty is a key first step that impacts all aspects of the data fusion problem. The representation must provide both an estimate of the state of the target or system of interest *and* its associated degree of error or uncertainty, and the uncertainty must be defined in a form that permits it to be empirically determined. There must be a rigorous algorithm for fusing information in the representation, and the computational complexity of the representation and its associated fusion algorithms must be bounded for practical application.

By far the most widely used information representation is the mean and covariance form, where the mean vector defines the best estimate of the state of the target and the error covariance provides an upper bound on the expected squared error associated with the mean. For example, the measured position of an object in two dimensions can be represented as a vector $\mathbf{a}$ consisting of the object's estimated mean position, e.g., $\mathbf{a} = [\mathbf{x}, \mathbf{y}]^T$, and an error covariance matrix $\mathbf{A}$ that expresses the uncertainty associated with the estimated mean. If the error in the estimated mean vector is denoted as $\tilde{\mathbf{a}}$, then the error covariance matrix is an estimate of the expected squared error, $\mathrm{E}[\tilde{\mathbf{a}}\tilde{\mathbf{a}}^T]$.

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
2006		**00-00-2006 to 00-00-2006**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Generalized Information Representation and Compression Using Covariance Union	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Dept. of Computer Science,University of Missouri-Columbia,Columbia,MO,65211**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
The 9th International Conference on Information Fusion (Fusion 2006) 10-13 July 2006, Florence, Italy

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**6**	

The estimate is said to be consistent (or conservative) if and only if $\mathbf{A} \geq \mathrm{E}[\tilde{\mathbf{a}}\tilde{\mathbf{a}}^T]$ or, equivalently, $\mathbf{A}$ - $\mathrm{E}[\tilde{\mathbf{a}}\tilde{\mathbf{a}}^T]$ is positive definite or semidefinite (i.e., has no negative eigenvalues). The full estimate of a target's state is given by the mean and covariance pair $(\mathbf{a}, \mathbf{A})$.

Given two mean and covariance estimates $(\mathbf{a}, \mathbf{A})$ and $(\mathbf{b}, \mathbf{B})$, the data fusion problem consists of determining a fused estimate $(\mathbf{c}, \mathbf{C})$ that is guaranteed to be consistent and summarizes the information in the two estimates with error (in terms of the size of $\mathbf{C}$) that is less than or equal to that of either estimate. If the two estimates are consistent and have a precisely known degree of correlation, the Kalman filter can be applied; otherwise, Covariance Intersection (CI) must be used. Both algorithms yield guaranteed consistent results when used appropriately. The limitations of the mean and covariance representation of information can be found in a variety of practical contexts. For example, suppose a vehicle is being tracked along a road in an urban environment. Assuming that it travels at a speed that is average for the road, its future position can be predicted forward a short length of time reasonably accurately; however, if it encounters a T-junction at which it must turn left or right, there are two distinct possible future positions. The future state can be represented with a single mean and covariance estimate, but doing so requires establishing a mean position at the junction with a covariance large enough to account for its position after a left *or* right turn. This produces a clearly unsatisfactory result in which the mean vector does not correspond to either of the possible states of the vehicle and consequently has a very large error covariance. Intuitively it seems clear that a better option would be to maintain information about the *two* possible future states rather than subsuming them into a single mean and covariance estimate.

Historically there have been two distinct approaches for representing "multimodal" information (e.g., as in the above example). One involves Multiple Hypothesis Tracking (MHT), which maintains multiple mean covariance estimates corresponding to distinct possible states [1]. The other approach is to attempt to maintain a parameterization of the Probability Density Function (PDF) that defines the uncertainty distribution associated with state of the target. In practice, PDF approximation methods typically only represent the significant modes of the distribution in terms of their means and covariances, thus making its representation all but identical to MHT. A key distinction is that the PDF-based approach treats the set of estimates as defining a union of Gaussian probability distributions. More specifically, the distribution is expressed as a Gaussian Mixture Model (GMM) of the form:

$$p(\mathbf{x}) = \sum_{i=1}^{N} p_i \mathcal{N}\left\{\mathbf{x}; \mu_i, \mathbf{P}_i\right\} \tag{1}$$

where the ith mode has mean μ_i, covariance $\mathbf{P}_i$ and weight p_i. The weights are all non-negative and sum to one.

The reason for adopting this form is that GMMs can conveniently approximate a wide class of PDFs and are identical in implementation to MHT. Unfortunately, representation is only one aspect of the overall information management problem. There also must be tractable algorithms for fusing information in a given representation.

The fusion of a set S of mean and covariance estimates, each defining a possible state of the target only one of which is guaranteed to be consistent, with another set T can be accomplished under the MHT interpretation simply by forming the Cartesian product $S \times T$ and applying the appropriate fusion algorithm (Kalman or CI) to the pairs. Unfortunately, this yields a combined estimate that has $O(|S|*|T|)$, which implies that the complexity of the fused estimate exceeds that of the original estimates. This increasing complexity will tend to exhaust available resources and therefore must be mitigated.

3 Representation Compression

One of the most important features of the mean and covariance representation of information is its constant complexity. Specifically, the amount of information required to describe the state of the target does not increase as new information is incorporated. However, when the representation of state is generalized to maintain more than one mean and covariance estimate, corresponding to different modes, the update/fusion operation multiplies the number of modes. In order to manage the complexity of the representation some form of representation compression must be applied.

In most MHT applications, the proliferation of hypotheses is managed by pruning the least likely ones according to some measure. A practical problem with pruning is that the likelihood measure typically includes many assumptions (e.g., PDF-related) that lead to more loss of correct hypotheses than is expected, and any loss of the hypothesis that corresponds to the true state of the target undermines the rigor of the entire information management framework. Therefore, pruning cannot be the primary mechanism for the limiting the representational complexity of our multimodal estimates.

If it is not possible to prune estimates (discard modes), then the only alternative is to somehow coalesce similar modes to stay within a fixed representational complexity budget. The key question is how to perform this coalescing so that the integrity of the information is maintained. If it is assumed that one of mode of an estimate corresponds to the true state of the target, and the others are spurious, then a mechanism called Covariance Union (CU) can be applied. For example, given n modes represented by estimates $(\mathbf{a}_1, \mathbf{A}_1) \ \ldots \ (\mathbf{a}_n, \mathbf{A}_n)$, CU produces an estimate $(\mathbf{u}, \mathbf{U})$ that is guaranteed to be consistent as long one of the mode estimates $(\mathbf{a}_i, \mathbf{A}_i)$ is consistent. This is achieved by guaranteeing that the estimate $(\mathbf{u}, \mathbf{U})$ is consistent with respect to *each* of the estimates:

$$\mathbf{U} \geq \mathbf{A}_1 + (\mathbf{u} - \mathbf{a}_1)(\mathbf{u} - \mathbf{a}_1)^T \tag{2}$$

$$\mathbf{U} \geq \mathbf{A}_2 + (\mathbf{u} - \mathbf{a}_2)(\mathbf{u} - \mathbf{a}_2)^T \tag{3}$$

$$\vdots \tag{4}$$

$$\mathbf{U} \geq \mathbf{A}_n + (\mathbf{u} - \mathbf{a}_n)(\mathbf{u} - \mathbf{a}_n)^T \tag{5}$$

where some measure of the size of $\mathbf{U}$, e.g., determinant, is minimized. The consistency of the CU estimate is assured for each of the n inequalities because the difference between the mean $\mathbf{u}$ and $\mathbf{a}_i$ is accounted for in the covariance $\mathbf{U}$ by the addition of the square of that difference to the covariance $\mathbf{A}_i$.

Given a complexity budget of N modes, the the fusion of two N-mode estimates will produce a new estimate with N^2 modes which must be reduced to N modes. This can be achieved by applying a clustering algorithm (e.g., standard k-means clustering based on a covariance-weighted distance measure such as Mahalanobis). Each of the N clusters can be combined into a single mean and covariance estimate using CU, and the rigor of the framework is guaranteed because one of the N estimates will be consistent as long as one of the original N^2 estimates was consistent.

This application of CU for mode reduction is appropriate for MHT-type applications. However, CU must be generalized to accommodate weights/probabilities associated with modes when the representation is interpreted to be a Gaussian mixture approximation of a multimodal probability distribution. This requires a generalization of the definition of consistency for multimodal estimates. We require that each probability p_i be greater than or equal to the actual probability that estimate/mode i corresponds to the true state of the target. The problem is that any small but nonzero probability implies that the associated estimate may represent the true state of the target, so consistency requires it to have the same influence on the CU result as an estimate with a much higher probability. The only difference is that the final result can be interpreted as having an associated probability that is equal to $\min(1, \sum_i p_i)$, where the *min* function is required because the weights are assumed to be conservative and thus may sum to a value greater than unity. Thus, the MHT case is equivalent to having no probability estimates, which requires unity to be assumed for every mode.

4 Computational Methods

Unlike Covariance Intersection, for which efficient semidefinite matrix optimization methods can be applied, Covariance Union involves inequalities with terms that depend on the means of the estimates. This dependency on the means requires a more sophisticated variant of the methods that are applied for straight semidefinite matrix equations. For our experiments, however, we have applied simpler generic optimization methods, which are discussed in this section.

The optimization problem's feasible region is the intersection of a set of inequalities, each of which can be written as a linear matrix inequality in $\mathbf{u}$ and $\mathbf{U}$:

$$\begin{bmatrix} (\mathbf{U} - \mathbf{A}_k) & (\mathbf{u} - \mathbf{a}_k) \\ (\mathbf{u} - \mathbf{a}_k)^T & 1 \end{bmatrix} \geq \mathbf{0} \tag{6}$$

The intersection of all of the constraints can then be represented as a larger block-diagonal inequality in which the diagonal elements are the LMI's shown above. This defines a region which is convex but nonsmooth. The fact that the constraints are nonsmooth rules out most commonly available high-performance optimization packages since they typically expect the objective and constraint functions to be twice continuously differentiable.

The trace measure is linear and so can be posed as a standard SDP problem [6]. There is no such formulation for other measures such as determinant or the Frobenius norm, so a general-purpose nonlinear optimizer such as *SolvOpt* [3] must be used to handle arbitrary norms. *SolvOpt* is an implementation of Shor's r-algorithm [4]. The initial feasible solution is generated by simply setting $\mathbf{u}$ to zero and summing the right-hand sides of the simplified constraints:

$$\mathbf{u}_0 = \mathbf{0} \tag{7}$$

$$\mathbf{U}_0 = \sum_{k=1}^{n} \left(\mathbf{A}_k + \mathbf{a}_k \mathbf{a}_k^T \right) \tag{8}$$

We have developed several approximate solutions that can also be applied which are much faster while still preserving consistency. These methods are suitable for real-time use and could also be used to generate better starting points for iterative improvement. Most of them rely on separation of the $\mathbf{u}$ and $\mathbf{U}$ optimizations to achieve computational savings. If the $\mathbf{u}$ vector is fixed at a specific value then the problem is considerably simplified: find a minimal $\mathbf{U}$ such that $\mathbf{U} \geq \mathbf{F}_k$ where the $\mathbf{F}_k$ are constant. This simpler problem can yield closed-form solutions when there are only two estimates to be combined. For example, if determinant is the measure used then the resulting $\mathbf{U}$ can be computed directly via simultaneous diagonalization:

$$\mathbf{U} = \left(\mathbf{V}^T\right)^{-1} \max(\mathbf{V}^T \mathbf{A} \mathbf{V}, \mathbf{V}^T \mathbf{B} \mathbf{V}) \, \mathbf{V}^{-1} \tag{9}$$

where *max* is the component-wise maximum of two diagonal matrices. $\mathbf{V}$ contains the generalized eigenvectors of $\mathbf{A}$ and $\mathbf{B}$. Using Matlab it would be computed as $[\mathbf{V}, \mathbf{D}] = eig\,(\mathbf{A}, \mathbf{B})$.

One such approximation is to assume that real-life applications produce estimates in which the optimal mean $\mathbf{u}$ can be modeled as a convex combination of the input means. This constrains $\mathbf{u}$ to a bounded region in $\mathbf{R}^n$. Indeed if there are only two estimates $(\mathbf{a}, \mathbf{A})$ and $(\mathbf{b}, \mathbf{B})$ to be unioned then $\mathbf{u}$ is constrained to the line segment between $\mathbf{a}$ and $\mathbf{b}$:

Let $\mathbf{c} = \mathbf{b} - \mathbf{a}, \mathbf{u} = \mathbf{a} + \omega \mathbf{c}$. The convex combination problem can then be stated as:

Find a minimal $\mathbf{U}$ such that:

$$\mathbf{U} \geq \mathbf{A} + \omega^2 \mathbf{c}\mathbf{c}^T \tag{10}$$

$$\mathbf{U} \geq \mathbf{B} + (1 - \omega)^2 \, \mathbf{c}\mathbf{c}^T \tag{11}$$

This can easily be solved via any number of simple one-dimensional search techniques, using the previously noted formulae to compute $\mathbf{U}$ for a fixed value of $\mathbf{u}$.

In our experiments it has been observed that convex-combination CU produces reasonably good approximations to the optimal values when applied to two estimates in low dimensions. However, its performance has not yet been fully characterized. It was evaluated using a determinant on pairs of estimates whose mean components and covariance eigenvalues were randomly chosen on the interval $(0, 1)$ and the dimensionality n varied from 2 to 20. For the two-dimensional data the determinant of $\mathbf{U}$ produced by the convex-combination CU averaged only 4% larger than the optimal value. However, for $n = 20$ it was 20% larger. So its performance degraded as n was increased (as could be expected from the definition of determinant and the method used to construct the test set) but the increase appeared to be only proportional to $\sqrt{n}$.

Another fast real-time approximation can be derived by noting that the optimal two-element CU update tends to produce a $\mathbf{u}$ vector for which the two constraints are similar in size and shape. In other words, it has a tendency to select a $\mathbf{u}$ vector for which:

$$\mathbf{A} + (\mathbf{u} - \mathbf{a})(\mathbf{u} - \mathbf{a})^T \approx \mathbf{B} + (\mathbf{u} - \mathbf{b})(\mathbf{u} - \mathbf{b})^T \quad (12)$$

This observation suggests a strategy in which $\mathbf{u}$ is fixed at the point where the difference is minimized. If the Frobenius norm of the difference is minimized then it leads to a closed-form solution for $\mathbf{u}$:

$$\mathbf{u} = \left(\mathbf{a} + \mathbf{b} + \left(\left(\mathbf{c}^T\mathbf{c}\right)\mathbf{I} + \mathbf{c}\mathbf{c}^T\right)^{-1}(\mathbf{A} - \mathbf{B})\,\mathbf{c}\right)/2 \quad (13)$$

$$\mathbf{c} = (\mathbf{a} - \mathbf{b})/2 \quad (14)$$

This solution has only been tested with random data. It produces good estimates when the differences between the estimates' means is large compared to the differences between the estimates' covariance matrices.

Large problems with many estimates can be broken down into a set of smaller problems by recursively solving two estimates at a time. For example, if there are three estimates $(\mathbf{a}_1, \mathbf{A}_1)$, $(\mathbf{a}_2, \mathbf{A}_2)$, and $(\mathbf{a}_3, \mathbf{A}_3)$ they can be separated into two smaller problems:

1. Compute $(\mathbf{u}_1, \mathbf{U}_1)$ as the union of $(\mathbf{a}_1, \mathbf{A}_1)$ and $(\mathbf{a}_2, \mathbf{A}_2)$.
2. Compute $(\mathbf{u}_2, \mathbf{U}_2)$ as the union of $(\mathbf{u}_1, \mathbf{U}_1)$ and $(\mathbf{a}_3, \mathbf{A}_3)$.
3. $(\mathbf{u}_2, \mathbf{U}_2)$ is the solution.

The main advantage of this approach is that two-element unions can be solved quickly via convex combination CU using closed-form formulas described earlier. But the method has one serious drawback that is illustrated in Figure 1: it does not guarantee consistency. It does guarantee that the covariance matrices $\mathbf{U}_k$ will never shrink and will most likely grow on every iteration, but there is no guarantee that when $\mathbf{U}_{k+1}$ is re-centered at a new mean $\mathbf{u}_{k+1}$ that it will still be consistent with the earlier estimates. Previous experiments did not observe this effect due to the extra slack provided by the convex-combination formulation. The solution can be expected to be consistent as long as the errors/biases in the combined estimates are statistically independent. The CU equations can be easily generalized to account for potentially correlated biases in the means[1].

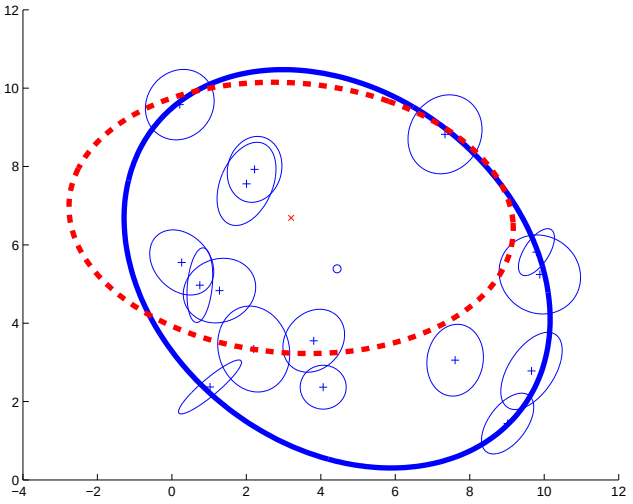

Figure 1: An example illustrating inconsistency with recursively applying the two-element unions. The means and 1σ ellipses of the input set $(\mathbf{a}, \mathbf{A})$ are shown as the set of thin solid ellipses with their means at $+$. The batch CU estimate is the thick solid ellipse with the mean at $\circ$. The pairwise CU is the thick, dashed ellipse with its mean at $\times$. A necessary condition for $(\mathbf{u}, \mathbf{U})$ to be consistent is that all of the input means should lie within the 1σ covariance ellipse. However, many of the means for the input set lie outside for the pairwise fused result.

4.1 Implementation

The *SolvOpt* package is able to find a minimizing vector x according to a cost function $f(x)$, which may be optionally constrained by some function $g(x)$. We choose x to be the n elements of $\mathbf{u}$ plus the $\frac{n(n+1)}{2}$ elements of the upper triangle of $\mathbf{U}$.

We minimize the determinant of the covariance, $\mathbf{U}$, subject to the constraint that

$$\mathbf{X}_k = \mathbf{U} - \mathbf{A}_k - (\mathbf{u} - \mathbf{a}_k)(\mathbf{u} - \mathbf{a}_k)^T \quad (15)$$

has non-negative eigenvalues, for all $k \in [1, \ldots, m]$, where m is the number of estimates given.

To find $|\mathbf{U}|$, we perform an LU decomposition of matrix $\mathbf{U}$, to generate an upper triangular matrix $\mathbf{W}$ and a lower triangular matrix $\mathbf{L}$, such that $\mathbf{LW} = \mathbf{U}$. $\mathbf{L}$ and $\mathbf{W}$ are given by

$$\mathbf{L}_{ii} = 1 \quad (16)$$

$$\mathbf{L}_{ij} = \frac{1}{\mathbf{W}_{ii}}\left(\mathbf{U}_{ij} - \sum_{k=1}^{j}\mathbf{L}_{ik}\mathbf{W}_{kj}\right) \;;\; i > j \quad (17)$$

[1]The case of common bias terms just requires an additional parameter α_i per estimate: $\mathbf{U} \geq \mathbf{A}_i/\alpha_i + (\mathbf{u} - \mathbf{a}_i)(\mathbf{u} - \mathbf{a}_i)^T/(1 - \alpha_i)$

$$\mathbf{W}_{ij} = \mathbf{U}_{ij} - \sum_{k=1}^{i} \mathbf{L}_{ik}\mathbf{W}_{kj} \quad (18)$$

Then $|\mathbf{U}| = \Pi_{i=1}^{n}\mathbf{W}_{ii}$. The complexity cost of this operation is $O\left(n^3\right)$.

The single value *SolvOpt* uses to constrain the minimization must be nonpositive. Since we want to constrain the eigenvalues of (15) to be nonnegative for all $k \in [1, \ldots, m]$, we simply find the most negative of all nk eigenvalues, λ_{min}, and return $-\lambda_{min}$ as the constraint.

To compute the n eigenvalues of each $\mathbf{X}_k$, we follow a two-step procedure:

1. Find the Hessenberg form of $\mathbf{H}_k = Hess\left(\mathbf{X}_k\right)$
2. Apply the QR transform to $\mathbf{H}_k$ until the eigenvalues are isolated on the diagonal

The Hessenberg form of a symmetric matrix is tridiagonal, which simplifies the actual eigenvalue calculations. This technique works because the original matrix $\mathbf{X}_k$ and its Hessenberg form $\mathbf{H}_k$ have the same eigenvalues.

The QR algorithm iterates on $\mathbf{H}_k$ until it approaches the Shur normal form, which contains the eigenvalues on the diagonal.

Each QR decomposition of $\mathbf{H}_k$ results in $\mathbf{Q}$, which is orthogonal, and $\mathbf{R}$, which is upper triangular, such that $\mathbf{QR} = \mathbf{H}_k$. The algorithm proceeds as follows

$$\mathbf{QR} = \mathbf{H}_{k,s} \quad (19)$$

$$\mathbf{H}_{k,s+1} = \mathbf{RQ} \quad (20)$$

for $s = 0, 1, 2, \ldots$, until $\mathbf{H}_k$ is in the Shur normal form.

As has been discussed, *SolvOpt* evaluates the cost and constraint function callbacks to minimize $|\mathbf{U}|$ over the $n + \frac{n(n+1)}{2}$ elements of $\mathbf{u}$ and the triangle of $\mathbf{U}$. To merge m estimates, the cost function performs $O\left(n^3\right)$ operations, the constraints function $O\left(mn^3\right)$. The number of iterations which *SolvOpt* must perform varies widely, from 1500 to 15000, depending on the batch dimensions and also the input data values. In the next section we present results showing the overall computational cost of this approach.

5 Experimental Results

In this section we present experimental results for different implementations of the CU algorithm, using *SolvOpt*, written in both Matlab and C. We have timed the application of CU on sets of random data to explore actual execution times for various dimensions n, and modes N. The times listed in the following tables were obtained on a single 1.5 GHz Pentium computer.

Avg. execution times for Matlab (in secs)

Dimensions	2 Modes	4 Modes	8 Modes	16 Modes
2	0.91	1.21	1.94	2.22
4	22.76	10.75	12.78	21.63
6	40.95	80.58	55.68	74.41
8	230.50	204.36	231.83	276.55

Average execution times for C (in seconds)

Dimensions	2 Modes	4 Modes	8 Modes	16 Modes
2	0.00	0.00	0.01	0.03
4	0.43	0.62	1.89	2.73
6	2.42	6.25	14.18	30.61
8	11.50	37.05	63.16	146.87

These results show that the generality of the *SolvOpt* algorithm incurs a significant computational cost that makes it impractical for most real-time applications when the dimensionality and number of nodes is high.

6 Discussion

In this paper we have examined the problem of representing multimodal information using MHT and GMMs. We have discussed the fusion of information represented in the form of multiple mean and covariance estimates corresponding to distinct possible states, or modes of a distribution, for a tracked target. We have discussed how the fusion operation results in a multiplicative increase in the complexity of the representation that will grow exponentially over time unless bounded by a mechanism that can compress the representation to a fixed number of modes. We have described ow Covariance Union can be used to coalesce modes while preserving the rigor of the information management framework. Experiments demonstrate the effectiveness of our approach.

The main result of this paper is our *SolvOpt*-based algorithm, with implementations in Matlab and C, for computing CU solutions. Experimental results corroborate the correctness of the algorithm, but they also show that it is is not practical for real-time applications. It is expected, however, that our experimental codes will provide the "gold standard" against which faster approximations of CU can be derived.

References

[1] Y. Bar-Shalom and T. E. Fortmann. *Tracking and Data Association*. Academic Press, New York NY, USA, 1988.

[2] S. Julier and J. K. Uhlmann. Fusion of time delayed measurements with uncertain time delays. In *2005 American Control Conference*, Portland, OR, USA, 8 – 10 June 2005.

[3] A. Kuntsevish and F. Kappel. Solvopt - the solver for local nonlinear optimization problems. *Technical Report*, http://www.uni-graz.at/imawww/kuntsevich/solvopt/, 1997.

[4] N.Z. Shor. *Minimization Methods for Non-Differentiable Functions*, volume 3. Springer-Verlag, Berlin, Germany, 1985.

[5] Jeffrey Uhlmann. Covariance consistency methods for fault-tolerant distributed data fusion. *Information Fusion*, 4(3):201–215, Sept 2003.

[6] L. Vandenberghe and S. Boyd. Semidefinite programming. *SIAM Review*, 38(1):49–95, 1996.