A decentralised particle filtering algorithm for multi-target tracking across multiple flight vehicles

Lee-Ling Ong, Ben Upcroft, Tim Bailey, Matthew Ridley, Salah Sukkarieh and Hugh Durrant-Whyte
ARC Centre of Excellence in Autonomous Systems (CAS)
The University of Sydney, NSW, Australia
Email: s.ong@cas.edu.au

***Abstract*— This paper presents a decentralised particle filtering algorithm that enables multiple vehicles to jointly track 3D features under limited communication bandwidth. This algorithm, applied within a decentralised data fusion (DDF) framework, deals with correlated estimation errors due to common past information when fusing two discrete particle sets. Our solution is to transform the particles into Gaussian mixture models (GMMs) for communication and fusion. Not only can decentralised fusion be approximated by GMMs, but this representation also provides summaries of the particle set. Less bandwidth per communication step is required to communicate a GMM than the particle set itself hence conversion to GMMs for communication is an advantage. Real airborne data is used to demonstrate the accuracy of our decentralised particle filtering algorithm for airborne tracking and mapping.**

Fig. 1. Two Flight Vechicles: (Brumby Mk3 aircrafts)

I. Introduction

This paper presents an application of non-Gaussian, nonlinear decentralised data fusion (DDF) with particles for platforms such as unmanned aerial vehicles (UAVs). Real airborne data is used to demonstrate the accuracy of our decentralised particle filtering algorithm for airborne tracking and mapping.

Flight vehicles can survey large areas in a short time especially compared to ground vehicles. However, these vehicles experience aggressive dynamics including excessive roll rates and extreme flight speeds which affect sensor pointing direction. This causes features to be observed for only a few frames. However, multiple vehicles offer the advantage of large numbers of observations at various angles resulting in increased accuracy for tracking and mapping. Sharing information and fusing estimates between platforms using a decentralised architecture ensures modularity, robustness and scalability [1]. Scenarios where this application are of use include; environmental monitoring, bush fire-fighting and search and rescue [2].

The flight vehicles shown in Figure 1 are equipped with monocular visual sensors. The observation model is therefore, bearing/elevation-only and cannot be modelled as Gaussian. Multiple observations from different poses would be required for recursive triangulation to obtain a Gaussian observation likelihood. As observations are few, non-Gaussian representations are more suitable, in which particle filters are ideal. Furthermore, Hendby *et al.* [3] show that particle filters outperform methods based on linearisation such as the extended Kalman filter (EKF) and the unscented Kalman filter (UKF) particularly if the initialisation error is large.

Extensive research on particle filters has resulted in fast nonlinear, non-Gaussian operations for Bayesian filtering [4]–[6]. Other representations for general recursive filtering considered in the literature include grid-based techniques [7], Gaussian Mixture Models (GMMs) [8] and Parzen representations [9]. Grid based representations are not compact and do not scale well with dimension compared to particles. GMMs and Parzen representations require an approximate observation likelihood transformation from the sensor space to Cartesian space and result in a multiplicative increase of parameters at every local update, unlike particle filters. Here, we show a practical implementation of DDF using particle filters.

In a DDF framework, each platform runs its own local filter and communicates information to other nodes in the neighbourhood. Incoming information is fused with the local state to produce a global state of the world. There are three basic constraints to a DDF system [1] which are:

1) There exists no single central fusion centre and no node is central to the successful operation of the network.
2) Communications are kept on a strictly node-to-node basis.
3) There is no global knowledge of the network topology.

The advantages imposed by these three constraints include modularity, scalability, survivability and increased robustness. As nodes in a DDF system operate independently and communicate locally, an operating failure in a node would not affect integrity of the operation of surviving nodes. As knowledge of the global network topology is not required by a node, the system can be scaled by simply connecting new sensing nodes to the system.

The problem encountered in the application of particle filtering in DDF is dealing with correlated estimation errors due to common past information between two discrete particle sets, in order to achieve accurate fusion results. Our solution is to transform the particle set to a Gaussian Mixture Model (GMM) for fusion. GMMs maintain summaries of the representation and provides a fusion method that considers the common information issue. As GMMs requires less "floats" to store its components in comparison with the particle sets, communicating the distribution as a GMM provides the added advantage of reduced bandwidth requirements. Transformations to GMMs also smooth and regularise the sample set, allowing the particles to be distributed more evenly.

The paper is organised as follows: After presenting some related work (Section II), a generalised DDF node is described in Section III showing how common past information is maintained in a node. Section IV explains the problem with accounting for common past information in decentralised particle filtering. The algorithm of performing consistent DDF on particle filters is then introduced. In Section V, experimental results are presented to demonstrate the accuracy of the algorithm for airborne tracking and map building. Section VI concludes and presents future directions.

II. Related Work

A scalable Gaussian based DDF architecture has been successfully implemented on UAVs by Nettleton [10], using a Kalman Filter and its Information form. Local and communicated information was fused asynchronously via additive information matrices. However, this methodology does not lend itself to extensions for non-Gaussian distributions.

Rosencrantz *et al.* [11] and Ihler *et al.* [12] demonstrated DDF using non-Gaussian representations but did not consider accounting for common information. Ihler *et al.* applied a message-passing estimation technique known as non-parametric belief propagation based on a generalisation of particle filtering in sensor networks. These messages are estimates of the location and uncertainty of the sensor nodes themselves, represented as either samples or analytical functions. Our application differs from Ihler et al.'s as it is features in the environment rather than sensor locations that is estimated and tracked. Rosencrantz *et al.* decentralised a standard particle filter by communicating and fusing the most informative subsets of samples. The algorithm was applied on indoor mobile robots playing the game of laser tag.

III. Decentralised Node Structure

The operations in a decentralised node is illustrated in Figure 2. In a DDF system, each platform makes an observation over which a likelihood is generated. Data association is then performed with existing local filter tracks where either fusion or track initialisation takes place. Each local filter undergoes a standard cycle of a local observation update (multiplication

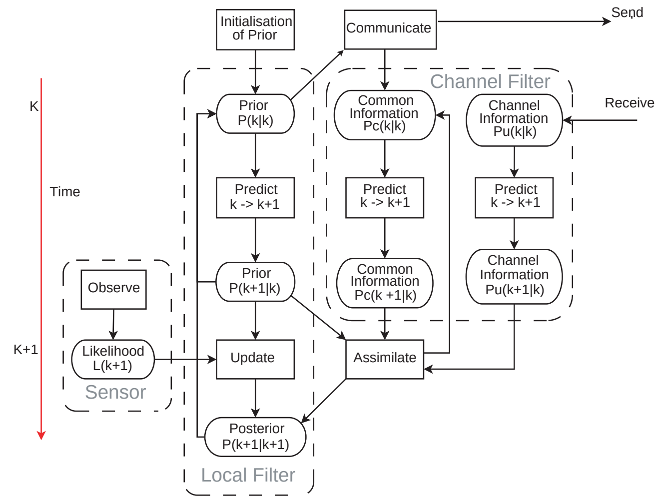

Fig. 2. Flow chart of the operations performed in DDF. Local filters update observation likelihoods from local sensors and fuse information received from channel filters. Channel filters maintain a record of common information between two nodes.

of prior and likelihood) and prediction (convolution of prior with process model).

At set times, the local particle sets are then transformed into a more compact representation and communicated to neighbouring nodes in the network via the channel filters [1]. The channel filter also receives information from neighbouring nodes. When this occurs, data association is performed. If associated to a track, the received information is fused after common past information is removed. The common information at the channel filter is also updated with the received information.

A. Common information

In order to perform DDF consistently, new information has to be recovered from the received estimate by removing common past information [13]. Figure 3 shows how common information $P\left(\mathbf{x}_k|\mathbf{Z}_i \bigcap \mathbf{Z}_j\right)$ arises.

At time (a), Node i makes an observation Z_i, of a feature, updates the local filter resulting in a posterior $P\left(\mathbf{x}_k|\mathbf{Z}_i\right)$, and sends its estimate to Node j, which instantiates the filter for this target. This communicated estimate $P\left(\mathbf{x}_k|\mathbf{Z}_i\right)$ also becomes the common information $P\left(\mathbf{x}_k|\mathbf{Z}_i \bigcap \mathbf{Z}_j\right)$, between these two nodes. At (b), new observations are updated at Nodes i and j. Hence at (c), Node j's feature estimate is $P\left(\mathbf{x}_k|\mathbf{Z}_i \bigcup \mathbf{Z}_j\right)$, a combination of information from Node i at time (a) and information from local updates at Node j at time (b).

To avoid errors arising from correlations, the common past information has to be removed from the estimate prior to fusion through a division so that at (d), the estimate at both Nodes i is based only on "new" information from Node j. Hence, at (d) the common information is now what Node j communicated at step (c). The fusion between two nodes amounts to a division (shown in (c) of Figure 3 operation to remove the common information and a multiplication [14]which

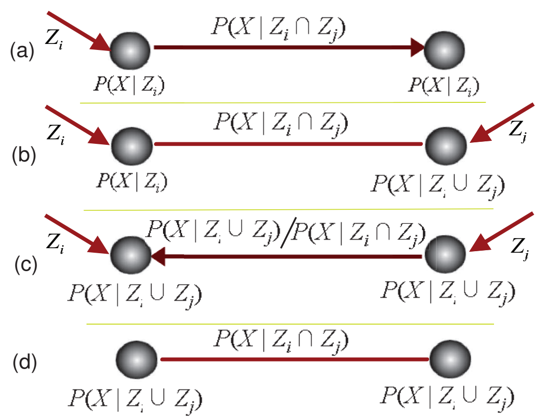

Fig. 3. How common information arises from the network is shown. Communication of estimates (indicated by arrows) are from Node i to Node j at (a) and vice versa at (c). At (c), the estimate communicated contains past information sent from Node i . This is information about observations updated at Node i in time (a). This common information has to be removed by division so only "new" information is fused at Node i to avoid double counting.

is:

$$P\left(\mathbf{x}_k|\mathbf{Z}_i \bigcup \mathbf{Z}_j\right) \propto \frac{P\left(\mathbf{x}_k|\mathbf{Z}_i\right) P\left(\mathbf{x}_k|\mathbf{Z}_j\right)}{P\left(\mathbf{x}_k|\mathbf{Z}_i \bigcap \mathbf{Z}_j\right)} \tag{1}$$

Obtaining a mathematically consistent and tractable formulation of this division operation is the problem for decentralised particle filtering which is explained in Section IV-A.

IV. Decentralised Particle Filter Algorithms

Particle filters are a Monte Carlo estimation method based on importance sampling, adapted to sequential filtering for dynamic systems [15]. The probability distribution of the state, is represented by particles at a given moment in time k, as a set of weighted samples $\{\mathbf{x}_k^{(i)}, w_k^{(i)}\}_{i=1}^N$, such that the density is approximated by an empirical estimate,

$$P\left(\mathbf{x}_k|\mathbf{Z}^k\right) \approx \sum_{i=1}^{N} w_k^{(i)} \delta\left(\mathbf{x}_k^{(i)}\right) \tag{2}$$

where $\delta\left(\cdot\right)$ is the Dirac delta function.

The basic particle filter has two key weaknesses which hinder efficient application to many estimation problems. The first is sample impoverishment, where, during resampling, certain particles are selected multiple times and others not at all, thereby reducing the total number of independent samples. The second weakness is an inability to adequately explore the state-space if the support of the prior distribution has little overlap with the likelihood function. The solution applied in this paper is to fit mixture models to the samples [6], which, is a form of regularisation. Mixture models additionally provide a means to perform DDF as well.

A. Issues with fusing particle filters in DDF

The division operation (Equation 1) can be performed analytically with Gaussian representations for tree-connected networks. If the correlation between the estimates to be fused is unknown, a covariance intersect filter can be applied [13].

However, for particle filters, the problem that occurs is fusing two particle sets directly . There is only support for an infinitesimally small interval at the particle. Elsewhere the value of the particle is zero. Hence, there is no overlap between samples even in the same space. Unless two samples lie exactly at the same spot, multiplication between the two sample sets would result in zero as shown in Figure 4. For example, the multiplication of the first sample of set 1 with the first sample in set 2 is $\delta(x-0.5)\delta(x-1)=0$.

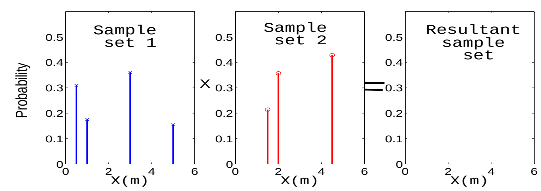

Fig. 4. Samples from one particle set do not have the same support on the space as samples from another set. The particle at 0.5 m from Set 1 will be multiplied by a value of zero at 0.5 m from Set 2 because there is no support at that point at Set 2.

Rosencrantz *et al.* [11] performed fusion on particle filters by adding the most informative subset of samples from two nodes together. This method is mathematically inconsistent with Equation 1 and common past information is not accounted for as a division operation cannot be performed on discrete samples for the same reasons as multiplication.

B. Fusion solution

The solution applied to accommodate fusion is a transformation of each particle set to a continuous distribution such as Gaussian mixture model (GMM) for the fusion process. In [16], we showed that GMMs are preferred compared to Parzen distributions because the fusion method is less computationally expensive although the method for fusing Parzen representations [17] is more accurate. It was shown in [16] that GMMs require less bandwidth per communication step compared to particles, hence, we summarise the particles to GMMs prior to communication. The conversion to a Gaussian mixture model or Parzen representation is the most computationally complex process. Should a faster way of conversion be achieved, Parzen representations may be a more desirable option.

1) Gaussian Mixture Models: A Gaussian mixture model for a random variable $\mathbf{x}$ is:

$$P(x) = \sum_{i=1}^{n} \gamma_i G_i(x; \mu_i, \Sigma_i) \tag{3}$$

where x is in the domain of $\mathbf{x}$, G_i, is the ith Gaussian component, and γ_i are the weights where $\sum_{i=1}^{n} \gamma_i = 1$. The multivariate Gaussian distribution of the state x with mean μ and covariance $\mathbf{\Sigma}$ is defined as:

$$P(x) = \frac{1}{(2\pi)^{n/2}|\mathbf{\Sigma}|^{1/2}} \exp^{-\frac{1}{2}[\mathbf{x}-\mu]^T \mathbf{\Sigma}^{-1}[\mathbf{x}-\mu]} \tag{4}$$

2) Conversion to a continuous distribution: The method of converting to a continuous distribution (shown in Figure 5) is based on Musso [6] where each sample is converted to a kernel $K_h(\mathbf{x})$:

$$K_h(\mathbf{x}) = h^D K(\mathbf{x}) \tag{5}$$

where D is the number of dimensions, $K(.)$ is the rescaled kernel density and $h > 0$ is the window or scaling parameter. The kernel selected is Gaussian with

$$h = (\frac{4}{D+2})^e N^{-e} \tag{6}$$

where $e = \frac{1}{D+4}$, and N is the number of samples.

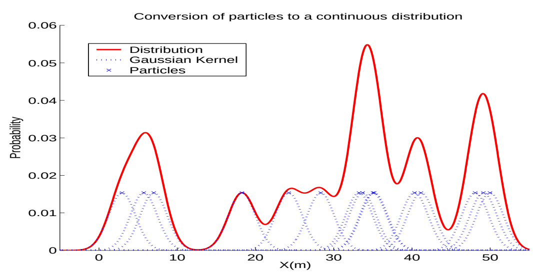

Fig. 5. Conversion of particles to a continuous distribution by placing Gaussian kernels over each particle

Communicating the continuous distribution in this form would be slightly worse than communicating the sample set itself as there is a kernel for each particle. Hence, approximating this distribution by a more compact one such as GMMs is more desirable. West's joining algorithm [18] is applied to merge pairs of components from the sum of Gaussian kernels ($\sum_{i=1}^{n} \gamma_i G_i(x;\mu_i)$) converted from the particles, successively until the desired level of component reduction has been achieved. The distance measure utilized to gauge the similarity of component i and component j of the GMM is a Mahalanobis-type distance measure:

$$d_{ij}^2 = \frac{\gamma_i \gamma_j}{\gamma_i + \gamma_j}(\mu_i - \mu_j)^T \Sigma^{-1}(\mu_i - \mu_j) \tag{7}$$

where μ is the state vector of the component, Σ is the mixture covariance matrix and γ is the component weight.

3) The fusion method: This method requires that first the local particle set be also converted to a GMM. Fusion is then performed using a pairwise component covariance intersect (CI) update [19] with the communicated GMM. Illustrated in Equations 8,9 and 10 are the CI operations where Σ_{ij}, μ_{ij} and γ_{ij} are the new covariance, mean, and weight of the component after fusion between the ith component of the local estimate and the jth component of the communicated estimate. A CI weighting parameter ω is selected to minimise the determinant of the result. The new particle set is then obtained by sampling from the fused GMM.

$$\Sigma_{ij}^{-1} = \omega \Sigma_i^{-1} + (1-\omega)\Sigma_j^{-1} \tag{8}$$

$$\mu_{ij} = \Sigma_{ij}(\omega \Sigma_i^{-1}\mu_i + (1-\omega)\Sigma_j^{-1}\mu_j) \tag{9}$$

$$\gamma_{ij} = \gamma_i \times \gamma_j \times W \tag{10}$$

where $W = \frac{1}{(2\pi)^{n/2}|\mathbf{K}|^{1/2}} \exp\{-\frac{1}{2}(\mu_i - \mu_j)^T \mathbf{K}^{-1}(\mu_i - \mu_j)\}$ and $\mathbf{K} = \frac{\Sigma_i}{\omega} + \frac{\Sigma_j}{1-\omega}$

V. Experimental Results

In this application, two flight vehicles undergo trajectories approximately 100 m above ground with average flight speeds of 144 km/hr. The flight vehicles start tracking at opposite ends of the trajectories. The feature extraction rate from a vertically-mounted monochrome camera is 25Hz. White features of 2×2m placed on the ground, shown in Figure 6 are tracked using our decentralised particle filtering algorithm. The range-cutoff used when initialising a new filter is 350 m shown in Figure 7. Every second, each alternate platform communicates summaries of the sample set to each other. This application is part of the second phase of the Autonomous Navigation and Sensing Experimental Research (ANSER II) project which aims at demonstrating DDF techniques for general non-Gaussian, non-point feature information. Such information includes air-borne and ground-based observations of natural features and targets from both imaging and range sensors, together with information from databases and human operators.

Fig. 6. Snapshots of the environment where the white features are tracked

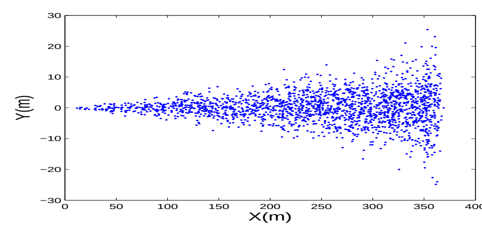

Fig. 7. Initialisation of a particle filter with bearing and elevation of 0 degrees and 1 degree standard deviation.

The system process model used for prediction was the Integrated Ornstein-Uhlenbeck process [7] which allows for bounding of the Brownian velocity over time. This prevents excessively large motion that can occur when the features is not observed for an extended period. The observations ($\mathbf{z}_k$) are a sequence of bearing (φ) and elevation (ϑ) measurements:

$$\mathbf{z}_k = [\varphi\ \vartheta]^T = \begin{bmatrix} \tan^{-1}(y_k/x_k) \\ \tan^{-1}(z_k/\sqrt{x_k^2 + y_k^2}) \end{bmatrix} + \mathbf{v}_k \tag{11}$$

where $\mathbf{v}_k$ is the measurement noise.

A. Data Association

In our operating environment shown in Figure 6, features are sparse, unlike in urban environments. Hence, robust

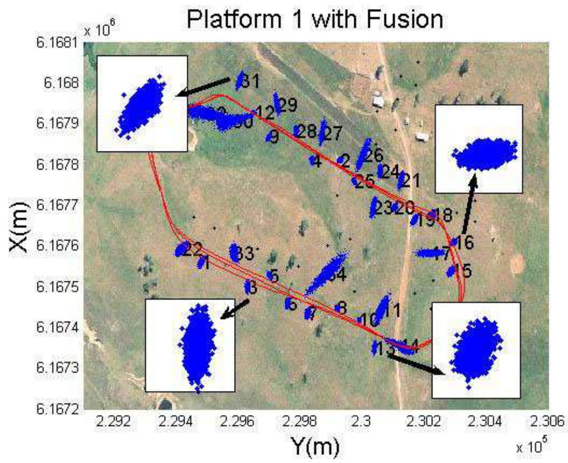

Fig. 8. GMM Fusion results from Node 1. 34 features were tracked.

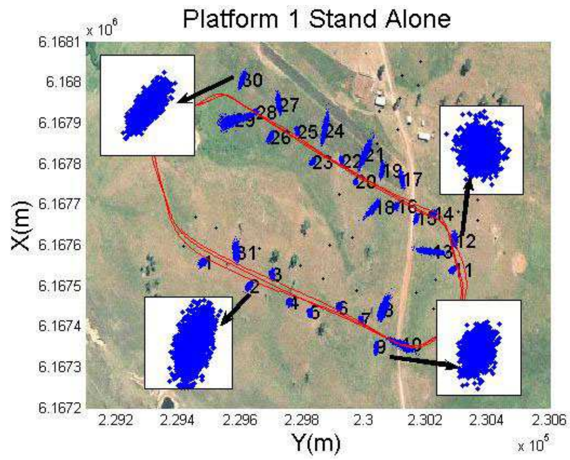

Fig. 10. Platform 1 StandAlone Result. 31 features were tracked.

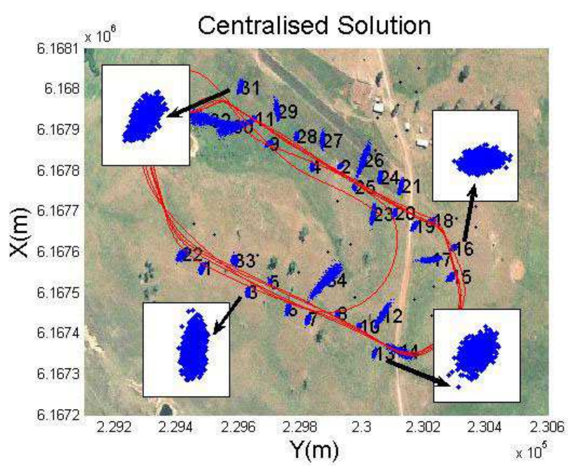

Fig. 9. Centralised Solution. 34 features were tracked.

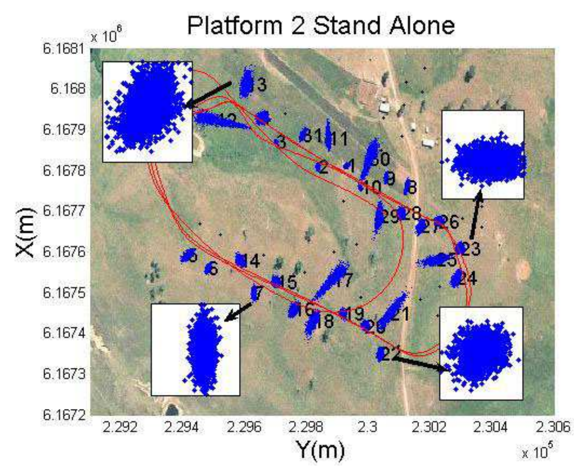

Fig. 11. Platform 2 StandAlone Result. 31 features were tracked.

data association such as Joint Probabilistic Data Association (JPDA) [20] is not applied. Instead a simple validation method using the Malahanobis distance [21] is used to correctly match local observations and received estimates to local tracks.

B. Analysis of Results

The final post-processed results are shown from Figures 8 to 11. Observations from both platforms were communicated at 25Hz to obtain the centralised solution. To measure the performance of our algorithm, the fusion solution is compared to the centralised one as we assume that the centralised solution is the most optimal.

To show that the decentralised solution is closer to the centralised result in comparison with platforms operating alone, a relative entropy or Kullback-Leibler(KL) divergence [22] is used. The KL divergence between two probability mass functions $p(x)$ and $q(x)$ is defined as [22]:

$$D(p||q) = \sum_{x \in X} p(x) \log \frac{p(x)}{q(x)} \tag{12}$$

The relative entropy is always non-negative and is zero if and only if $p = q$.

Figure 12 illustrates the average KL-Divergence results for three features which are the features numbered 2,3,4 on the centralised solution map. These features were selected as they were observed for the longest period of time by both vehicles. As both platforms start on opposite sides of the trajectory, a feature track initialised at one platform will be initalised at the other platform using the GMM fusion solution. However, without fusion, the track for this feature will only be initialised at the second platform when its sensor detects it. At this stage, the uncertainty of the feature for the first platform would have increased, while for the GMM fusion solution, communicated updates would reduce the uncertainty of the feature. The decentralised nodes have lower KL-divergence values and hence exhibit performances closer to the centralised solution than platforms operating alone.

To show that common past information is accounted for, the distribution of the GMM fusion solution cannot be more compact than the centralised solution. An Entropy measure [23] is used to determine the information content of a distribution.

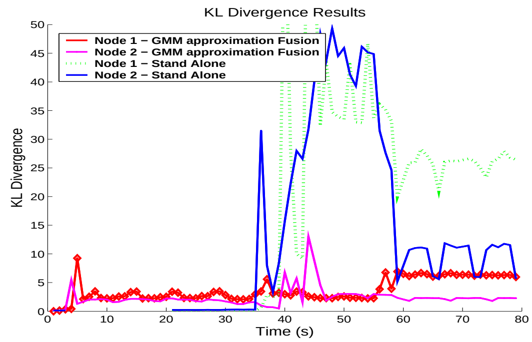

Fig. 12. Node 1 - KL Divergence results for the Stand Alone Nodes and Node 1 and 2 communicating and fusing GMMs

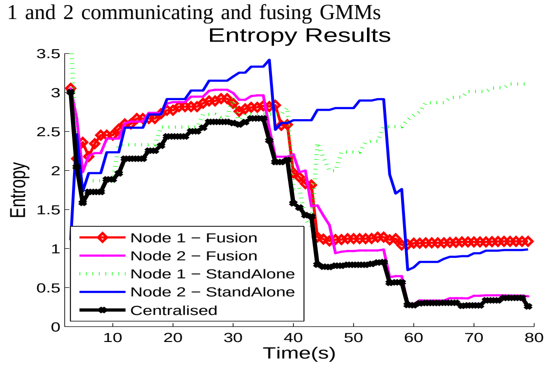

Fig. 13. Node 1 - Entropy results for the Stand Alone Nodes and Node 1 and 2 communicating and fusing GMMs

The more compact a distribution, the lower the Entropy value. Figure 13 shows the Entropy results averaged over same three features, (Features numbered 2,3,4 on the centralised solution map), and confirm that the entropies of the decentralised solutions are higher than the centralised one but less than their coresponding stand alone solutions as time increases. Hence, the decentralised solutions have less compact distributions compared to the centralised one.

VI. Conclusion

This paper presented an algorithm for decentralised particle filtering to jointly track 3D features under limited communication bandwidth. Here, the particle distribution was transformed to GMMs for communication and fusion. Our experimental results have demonstrated the accuracy of our decentralised particle filtering algorithm. Sharing information between platforms increases the number of features tracked and the compactness of each particle filter.

Areas for future work is the development of consistent fusion methods for GMMs and faster methods of transformations form particles to GMMs. Future work will also include a demonstration of decentralised particle filtering using vision sensors on airborne vehicles, ground vehicles and stationary ground nodes.

Acknowledgment

This work is supported by the ARC Centre of Excellence programme, funded by the Australian Research Council (ARC) and the New South Wales State Government and is supported by BAE Systems, ATC, UK.

References

[1] S. Grime, "Communication in decentralised sensing architectures," Ph.D. dissertation, The University of Oxford, 1992.

[2] F. Bourgault, G. Mathews, A. Brooks, and H. Durrant-Whyte, "An experiment in decentralized coordinated search," in *9th International Symposium on Experimental Robotics 2004*, Singapore, 2004.

[3] G. Hendby, R. Karlsson, F. Gustafsson, and N. Gordon, "Recursive triangulation using bearings only sensors," in *IEE Seminar on Target Tracking: Algorithms and Applications*, Birmingham, UK, March, 2006.

[4] M. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, 2002.

[5] A. Doucet, N. de Freitas, and N. Gordon, "An introduction to sequential Monte Carlo methods," in *Sequential Monte Carlo Methods in Practice*, A. Doucet, N. de Freitas, and N. Gordon, Eds. Springer-Verlag, 2001, pp. 3–14.

[6] C. Musso, N. Oudjane, and F. Le Gland, "Improving regularised particle filters," in *Sequential Monte Carlo Methods in Practice*, A. Doucet, N. de Freitas, and N. Gordon, Eds. Springer-Verlag, 2001, pp. 247–272.

[7] L. Stone, C. Barlow, and T. Corwin, *Bayesian Multiple Target Tracking*. Artech House, 1999.

[8] D. Alspach and H. Sorenson, "Nonlinear bayesian estimation using gaussian sum approximations," *IEEE Transactions on Automatic Control*, vol. 17(4), pp. 439–448, August 1972.

[9] E. Parzen, "On estimation of a probability density function and mode," *The Annals of Mathematical Statistics*, vol. 33(3), pp. 1065–1076, September, 1962.

[10] E. Nettleton, H. Durrant-Whyte, and S. Sukkarieh, "A robust architecture for decentralised data fusion," in *The 11th International Conference on Advanced Robotics (ICAR'03)*, Coimbra, Portugal, 2003.

[11] M. Rosencrantz, G. Gordon, and S. Thrun, "Decentralised data fusion with distributed particle filters," in *Proceedings of the Conference on Uncertainty in AI (UAI)*, Acapulco, Mexico, 2003.

[12] A. Ihler, E. Sudderth, W. Freeman, and A. Willsky, "Efficient multiscale sampling from products of gaussian mixtures," in *Proceedings of the Twentieth Conference on Uncertainty in Artificial Intelligence*, 2004.

[13] S. Julier and J. Uhlmann, "A nondivergent estimation algorithm in the presence of unknown correlations," *In Proceedings of The American Control Conference*, vol. 4, pp. 2369–2373, July, 1997.

[14] Y. E. Bar-Shalom, *Multitarget-multisensor tracking: advanced appplications*. Artech House, 1990.

[15] N. J. Gordon, D. Salmond, and A. Smith, "Novel approach to nonlinear/non-gaussian bayesian state estimation," *IEE Proceedings-F,*, vol. 140(2), pp. 107–113, 1993.

[16] L. Ong, B. Upcroft, M. Ridley, T. Bailey, S. Sukkarieh, and H. Durrant-Whyte, "Decentralised data fusion with particles," in *Australasian Conference on Robotics and Automation 2005 (ACRA '05)*, Sydney, Australia, 2005.

[17] M. Ridley, B. Upcroft, L. Ong, S. Kumar, and S. Sukkarieh, "Decentralised data fusion with parzen density estimates," in *International Conference on Intelligent Sensors, Sensor Networks and Information Processing 2004 (ISSNIP '04)*, Melbourne, Australia, 2004.

[18] M. West, "Approximating posterior distributions by mixtures," *Journal of Royal Statistical Society, Series B*, vol. 55(2), pp. 409–442, 1993.

[19] B. Upcroft, S. Kumar, L. Ong, M. Ridley, T. Bailey, S. Sukkarieh, and H. Durrant-Whyte, "Rich probabilitstic representation for bearing only decentralised data fusion," in *Proceedings of the IEEE Conference on Information Fusion*, Philedelphia, USA, 2005.

[20] T. Fortmann, Y. Bar-Shalom, and M. Scheffe, "Sonar tracking of multiple targets using joint probabilistic data association," *IEEE Journal of Oceanic Engineering*, vol. OE-8, p. 173184, July 1983.

[21] R. Duda and P. Hart, *Pattern Classification and Scene Analysis*. John Wiley and Sons, 1973.

[22] T. Cover and J. Thomas, *Elements of Information Theory*, ser. Wiley Series in Telecommunications. New York: Wiley, 1991.

[23] C. Shannon, "A mathematical theory of communication," *Bell Systems Techical Journal*, vol. 2,7, pp. 379–423,623–656.