Extending Bayesian RFS SLAM framework to Multi-Vehicle SLAM

Diluka Moratuwage*, Ba-Ngu Vo†, Sardha Wijesoma*, Danwei Wang*
*School of Electrical and Electronic Engineering
Nanyang Technological University
Singapore
†School of Electrical Electronic and Computer Engineering
The University of Western Australia
Australia

***Abstract*—In this paper we present a novel solution to the Multi-Vehicle SLAM (MVSLAM) problem by extending the random finite set (RFS) based SLAM filter framework using two recently developed multi-sensor information fusion approaches in finite set statistics (FISST). Our solution is based on the modeling of the measurements and the landmark map as RFSs and factorizing the MVSLAM posterior into a product of the joint vehicle trajectories posterior and the landmark map posterior conditioned the vehicle trajectories. The joint vehicle trajectories posterior is propagated using a particle filter while the landmark map posterior conditioned on the vehicle trajectories is propagated using a Gaussian Mixture (GM) implementation of the probability hypothesis density (PHD) filter.**

I. Introduction

A number of Multi-Vehicle SLAM (MVSLAM) solutions are found in the robotics literature [1] [2] [3] [4] [5] [6] [7] developed by extending the conventional vector based mono-SLAM algorithms such as extended Kalman filter based SLAM (EKF-SLAM) [8], FastSLAM [9] and sparse extended information filter based SLAM (SEIF-SLAM) [10]. All these mono-SLAM algorithms solve the SLAM problem by propagating a posterior probability density of a vector consisting of the landmark map augmented with the vehicle state in time. Hence all such algorithms require solving certain additional sub-problems such as data association, clutter filtering and map management in order to produce a consistent solution. In addition, landmark detection uncertainty or data association uncertainty are not taken into account. As a result, all existing MVSLAM algorithms inherit these problems.

In order to address these issues in the conventional mono-SLAM solutions, random finite set (RFS) modeling was adopted into SLAM. The very first RFS based SLAM solution was presented by Mullane et al. in [11], where they modeled the measurements and augmented vehicle-landmark map state as RFSs. The augmented vehicle-landmark map state was propagated in time using a probability hypothesis density (PHD) filter [12] [13] from which the state of the vehicle and the landmark locations were jointly estimated. Further improving their original solution, in [14] [15] Mullane et al. proposed a Rao-Blackwellised PHD filtering solution by factorizing the full SLAM posterior into a product of the vehicle trajectory posterior and the landmark map posterior conditioned on the vehicle trajectory. This solution addressed the map management, landmark detection uncertainty and false measurements (clutter) in a single filtering step by representing the landmark map and measurements as RFSs and modeling the landmark map transition model more natural and appropriate manner. Moreover This approach catered propagation of vehicle trajectory posterior using a particle filter and the landmark map posterior conditioned on the vehicle trajectory using a Gaussian mixture (GM) implementation [16] of a PHD filter.

In this paper we present a new solution to the Multi-Vehicle SLAM (MVSLAM) problem by extending the RFS based SLAM filter framework. The proposed solution is based on the factorization of the full MVSLAM posterior into a product of the joint vehicle trajectories posterior and the landmark map posterior conditioned on the joint vehicle trajectories. The landmark map and the measurements are modeled as RFSs and the joint vehicle trajectories are propagated in time via a particle filter while the landmark conditioned on the vehicle trajectories is propagated using a GM implementation of a PHD filter.

II. Random Finite Set Multi-Vehicle SLAM Problem

We illustrate the RFS Multi-Vehicle SLAM algorithm for the case of two vehicles although it can be extended into a larger number of robots using the methods we present here.

Let the landmark map be denoted by the set $M_k = \{m_{k,1}, m_{k,2}, ..., m_{k,l_k}\}$ at time k, where l_k denotes the number of landmarks present in the map. Let the time sequence of poses history of each vehicle be denoted by $X_{1:k}^{(r)} = [X_1^{(r)}, X_2^{(r)}, ..., X_k^{(r)}]^T$, where $X_k^{(r)}$ denotes the pose of vehicle r, at time k. Let the time sequence of sets of range measurements obtained using range and bearing sensors mounted on each vehicle be denoted by $Z_{1:k}^{(r)} = [Z_1^{(r)}, Z_2^{(r)}, ..., Z_k^{(r)}]$, where $Z_k^{(r)} = \{z_{k,1}^{(r)}, z_{k,2}^{(r)}, ..., z_{k,n_k^{(r)}}^{(r)}\}$ denotes the measurement set received from vehicle r at time k, while $n_k^{(r)}$ denotes the number of measurements. Let $U_{1:k}^{(r)} = [U_1^{(r)}, U_2^{(r)}, ..., U_k^{(r)}]^T$ denote the time sequence of control commands applied to each vehicle $r, (r = 1, 2)$ up to time k, where $U_k^{(r)}$ denotes the

control command applied at time k. Using these information, we evaluate the MVSLAM posterior probability distribution given by,

$$p_{k|k}(M_k, X_{1:k}^{(1)}, X_{1:k}^{(2)} | Z_{1:k}^{(1)}, Z_{1:k}^{(2)}, U_{1:k}^{(1)}, U_{1:k}^{(2)}, X_0^{(1)}, X_0^{(2)}) \quad (1)$$

where $X_0^{(1)}$ and $X_0^{(2)}$ respectively denote the initial poses of first and second vehicles.

III. Formulation of Proposed Random Finite Set Multi-Vehicle SLAM Solution

The Multi-Vehicle SLAM posterior (1) is evaluated by factorizing as a product of the joint vehicle poses posterior and the landmark map posterior conditioned on the vehicle poses as follows,

$$\begin{aligned} &p_{k|k}(M_k, X_{1:k}^{(1)}, X_{1:k}^{(2)} | Z_{1:k}^{(1)}, Z_{1:k}^{(2)}, U_{1:k}^{(1)}, U_{1:k}^{(2)}, X_0^{(1)}, X_0^{(2)}) \\ &= p_{k|k}(M_k | Z_{1:k}^{(1)}, Z_{1:k}^{(2)}, X_{0:k}^{(1)}, X_{0:k}^{(2)}) \\ &\times p_{k|k}(X_{1:k}^{(1)}, X_{1:k}^{(2)} | Z_{1:k}^{(1)}, Z_{1:k}^{(2)}, U_{1:k}^{(1)}, U_{1:k}^{(2)}, X_0^{(1)}, X_0^{(2)}) \end{aligned} \quad (2)$$

This factorization makes it possible to propagate the joint vehicle trajectories posterior using a particle filter allowing application of complex non-linear motion models. Moreover, by representing the measurements and the landmark map as RFSs and modeling the landmark map transition model using finite set statistical (FISST) methods, it is possible to evaluate the landmark map posterior in the presence of false measurements (clutter) with data association and landmark detection uncertainty.

A. RFS Landmark Map Transition Model

Let the RFS representing the landmark map at time $k-1$ be denoted by M_{k-1}, then the landmark map transition model at time k is given by,

$$M_k = \Gamma_k(X_k^{(1)}, X_k^{(2)}) \cup \left[\bigcup_{\zeta_{k-1} \in M_{k-1}} \Upsilon(\zeta_{k-1}) \right] \quad (3)$$

where the RFS $\Gamma_k(X_k^{(1)}, X_k^{(2)})$ denotes the newly appearing landmarks in the joint sensor FOV, while the Bernoulli RFS $\Upsilon(\zeta_{k-1})$ denotes the predicted state of the landmark $\zeta_{k-1} \in M_k$.

B. RFS Landmark Measurement Model

Let M_k denote the predicted landmark map, while $C_k^{(r)}$ denote the RFS representing the clutter received from exteroceptive sensor mounted on rth vehicle at time k, then corresponding measurements can be represented by the RFS,

$$Z_k^{(r)} = C_k^{(r)} \cup \left[\bigcup_{\zeta_k \in M_k} \Theta_k^{(r)}(\zeta_k) \right] \quad (4)$$

where $\Theta_k^{(r)}(\zeta_k)$ is a Bernoulli RFS representing the measurement corresponding to landmark $\zeta_k \in M_k$.

IV. The PHD filter for Landmark Map Posterior Estimation

The landmark map posterior conditioned on the vehicle trajectories is evaluated using a PHD filter [12]. Let the PHD of the landmark map posterior at time k is given by,

$$\begin{aligned} &D_{k|k}(\zeta_k | Z_{1:k}^{(1)}, Z_{1:k}^{(2)}, X_{0:k}^{(1)}, X_{0:k}^{(2)}) \\ &= \int p_{k|k}(M_k \cup \zeta_k | Z_{1:k}^{(1)}, Z_{1:k}^{(2)}, X_{0:k}^{(1)}, X_{0:k}^{(2)}) \delta M_k \end{aligned} \quad (5)$$

then the number of landmarks in the map M_k in a region S can be obtained by,

$$N_{k|k} = \int_S D_{k|k}(\zeta_k | Z_{1:k}^{(1)}, Z_{1:k}^{(2)}, X_{0:k}^{(1)}, X_{0:k}^{(2)}) d\zeta_k \quad (6)$$

A. Landmark Map Prediction

The landmark map prediction posterior density is given by,

$$\begin{aligned} &p_{k|k-1}(M_k | Z_{1:k-1}^{(1)}, Z_{1:k-1}^{(2)}, X_{0:k}^{(1)}, X_{0:k}^{(2)}) \\ &= \int f_M(M_k | M_{k-1}, X_k^{(1)}, X_k^{(2)}) \\ &\times p_{k-1}(M_{k-1} | Z_{1:k-1}^{(1)}, Z_{1:k-1}^{(2)}, X_{0:k-1}^{(1)}, X_{0:k-1}^{(2)}) \delta M_{k-1} \end{aligned} \quad (7)$$

and the corresponding predicted PHD can be obtained using the PHD of the landmark map at time $k-1$ as,

$$\begin{aligned} &D_{k|k-1}(\zeta_k | Z_{1:k-1}^{(1)}, Z_{1:k-1}^{(2)}, X_{0:k}^{(1)}, X_{0:k}^{(2)}) \\ &= b_k(\zeta_k | X_k^{(1)}, X_k^{(2)}) \\ &+ D_{k-1|k-1}(\zeta_k | Z_{1:k-1}^{(1)}, Z_{1:k-1}^{(2)}, X_{0:k-1}^{(1)}, X_{0:k-1}^{(2)}) d\zeta_{k-1} \end{aligned} \quad (8)$$

where $b_k(\zeta_k | X_k^{(1)}, X_k^{(2)})$ denotes the intensity of the newly appearing landmarks in the joint sensor FOV.

B. Landmark Map Update

The landmark map update posterior density is given by,

$$\begin{aligned} &p_{k|k}(M_k | Z_{1:k}^{(1)}, Z_{1:k}^{(2)}, X_{0:k}^{(1)}, X_{0:k}^{(2)}) \\ &= g_k(Z_k^{(1)}, Z_k^{(2)} | M_k, X_k^{(1)}, X_k^{(2)}) \\ &\times \frac{p_{k|k-1}(M_k | Z_{1:k-1}^{(1)}, Z_{1:k-1}^{(2)}, X_{0:k}^{(1)}, X_{0:k}^{(2)})}{l_{k|k-1}(Z_k^{(1)}, Z_k^{(2)} | Z_{1:k-1}^{(1)}, Z_{1:k-1}^{(2)}, X_{0:k}^{(1)}, X_{0:k}^{(2)})} \end{aligned} \quad (9)$$

Assuming the number of false measurements produced by each vehicle is Poisson distributed at an average of $\lambda^{(r)}$, and their physical distribution given by $c^{(r)}(z^{(r)})$, the corresponding updated PHD can be obtained using two methods as follows.

1) Iterative update method: The resultant PHD can be obtained by iteratively updating the predicted PHD (8) [13] using each vehicles' observations as follows.

$$\begin{aligned} &D_{k|k}^{(1)}(\zeta_k | Z_{1:k}^{(1)}, Z_{1:k-1}^{(2)}, X_{0:k}^{(1)}, X_{0:k}^{(2)}) \\ &= (1 - P_D^{(1)}) D_{k|k-1}(\zeta_k) \\ &+ \sum_{z^{(1)} \in Z_k^{(1)}} \frac{P_D^{(1)} g_k^{(1)}(z^{(1)}) D_{k|k-1}(\zeta_k)}{\lambda^{(1)} c^{(1)}(z^{(1)}) + \int P_D^{(1)} g_k^{(1)}(z^{(1)}) D_{k|k-1}(\zeta_k) d\zeta_k} \end{aligned} \quad (10)$$

and

$$D^{(2)}_{k|k}(\zeta_k|Z^{(1)}_{1:k}, Z^{(2)}_{1:k}X^{(1)}_{0:k}, X^{(2)}_{0:k}) = (1-P^{(2)}_D)D^{(1)}_{k|k}(\zeta_k) + \sum_{z^{(2)}\in Z^{(2)}_k} \frac{P^{(2)}_D g^{(2)}_k(z^{(2)})D^{(1)}_{k|k}(\zeta_k)}{\lambda^{(2)}c^{(2)}(z^{(2)}) + \int P^{(2)}_D g^{(2)}_k(z^{(2)})D^{(1)}_{k|k}(\zeta_k)d\zeta_k} \tag{11}$$

where, the abbreviations $D_{k|k-1}(\zeta_k)$ and $D^{(1)}_{k|k}(\zeta_k)$ are given by,

$$D_{k|k-1}(\zeta_k) = D_{k|k-1}(\zeta_k|Z^{(1)}_{1:k-1}, Z^{(2)}_{1:k-1}, X^{(1)}_{0:k}, X^{(2)}_{0:k}) \tag{12}$$

$$D^{(1)}_{k|k}(\zeta_k) = D^{(1)}_{k|k}(\zeta_k|Z^{(1)}_{1:k}, Z^{(2)}_{1:k-1}X^{(1)}_{0:k}, X^{(2)}_{0:k}) \tag{13}$$

and, $P^{(1)}_D$, $P^{(2)}_D$, $g^{(1)}_k(z^{(1)})$ and $g^{(2)}_k(z^{(1)})$ are given by,

$$P^{(1)}_D = P^{(1)}_D(\zeta_k|X^{(1)}_k), \quad P^{(2)}_D = P^{(2)}_D(\zeta_k|X^{(2)}_k) \tag{14}$$

$$g^{(1)}_k(z^{(1)}) = g^{(1)}_k(z^{(1)}|\zeta_k, X^{(1)}_k) \tag{15}$$

$$g^{(2)}_k(z^{(2)}) = g^{(2)}_k(z^{(2)}|\zeta_k, X^{(2)}_k) \tag{16}$$

where $P^{(r)}_D$ denotes the probability of detection of a landmark by rth vehicle, which is often considered as a constant, while $g^{(r)}_k(z^{(r)})$ is the measurement likelihood function.

Hence the PHD of the measurement updated landmark map posterior (5) is approximated as,

$$D_{k|k}(\zeta_k|Z^{(1)}_{1:k}, Z^{(2)}_{1:k}, X^{(1)}_{0:k}, X^{(2)}_{0:k}) \approx D^{(2)}_{k|k}(\zeta_k|Z^{(1)}_{1:k}, Z^{(2)}_{1:k}, X^{(1)}_{0:k}, X^{(2)}_{0:k}) \tag{17}$$

This is a less computationally intensive approach for evaluating the PHD of the landmark map update posterior. Although computationally efficient, this method produces varient PHDs under reordering of the sensor updates. This means that the sensor update reordering results in different measurement-updated PHDs, yielding different estimations of the landmark map posterior. Although this approach may not be completely satisfactory from a theoretical point of view, in practice it can be seen that this doesn't produce significantly noticeable performance difference in the simulations.

2) General multi-sensor update method: The PHD update can be obtained using the general multi-sensor update method [17] as follows,

$$D_{k|k}(\zeta_k|Z^{(1)}_{1:k}, Z^{(2)}_{1:k}, X^{(1)}_{0:k}, X^{(2)}_{0:k}) = (1-P^{(1)}_D)(1-P^{(2)}_D)D_{k|k-1}(\zeta_k) + \left[\sum_{\mathcal{P}\boxminus_2 Z_k} \omega_{\mathcal{P}} \sum_{W\in\mathcal{P}} \rho_W(\zeta_k)\right] D_{k|k-1}(\zeta_k) \tag{18}$$

where the summation is taken over all so called "binary partitions" $\mathcal{P}$ of $Z_k = Z^{(1)}_k \cup Z^{(2)}_k$ (see [17] for more information). The notation, "$\mathcal{P}\boxminus_2 Z_k$" stands for "$\mathcal{P}$ partitions Z_k into binary cells W", where $W \in \mathcal{P}$ has one of the following forms,

$$W = \left\{z^{(1)}_k\right\}, \; W = \left\{z^{(2)}_k\right\}, \; W = \left\{z^{(1)}_k, z^{(2)}_k\right\} \tag{19}$$

The values of $\rho_W(\zeta_k)$ and $\omega_{\mathcal{P}}$ are given by,

$$\rho_W(\zeta_k) = \begin{cases} \dfrac{P^{(1)}_D l^{(1)}_{z^{(1)}}(\zeta_k)(1-P^{(2)}_D)}{1+D_{k|k-1}[P^{(1)}_D l^{(1)}_{z^{(1)}}(1-P^{(2)}_D)]} & if\ W=\{z^{(1)}_k\} \\ \dfrac{(1-P^{(1)}_D) l^{(2)}_{z^{(2)}}(\zeta_k)P^{(2)}_D}{1+D_{k|k-1}[(1-P^{(1)}_D) l^{(2)}_{z^{(2)}}P^{(2)}_D]} & if\ W=\{z^{(2)}_k\} \\ \dfrac{P^{(1)}_D l^{(1)}_{z^{(1)}}(\zeta_k)P^{(2)}_D l^{(2)}_{z^{(2)}}(\zeta_k)}{1+D_{k|k-1}[P^{(1)}_D l^{(1)}_{z^{(1)}}P^{(2)}_D l^{(2)}_{z^{(2)}}]} & if\ W=\{z^{(1)}_k, z^{(2)}_k\} \end{cases} \tag{20}$$

and,

$$\omega_{\mathcal{P}} = \frac{\prod_{W\in\mathcal{P}} d_W}{\sum_{\mathcal{Q}\boxminus_2 Z_k} \prod_{W\in\mathcal{Q}} d_W} \tag{21}$$

where,

$$l^{(1)}_{z^{(1)}}(\zeta_k) = \frac{g^{(1)}_k(z^{(1)}_k|\zeta_k, X^{(1)}_k)}{\lambda^{(1)}_k c^{(1)}_k(z^{(1)}_k)} \tag{22}$$

$$l^{(2)}_{z^{(2)}_k}(\zeta_k) = \frac{g^{(2)}_k(z^{(2)}_k|\zeta_k, X^{(2)}_k)}{\lambda^{(2)}_k c^{(2)}_k(z^{(2)}_k)} \tag{23}$$

and,

$$d_W = \begin{cases} 1+D_{k|k-1}[P^{(1)}_D l^{(1)}_{z^{(1)}}(1-P^{(2)}_D)] & if\ W=\{z^{(1)}_k\} \\ 1+D_{k|k-1}[(1-P^{(1)}_D) l^{(1)}_{z^{(1)}}P^{(2)}_D] & if\ W=\{z^{(2)}_k\} \\ 1+D_{k|k-1}[P^{(1)}_D l^{(1)}_{z^{(1)}}P^{(2)}_D l^{(2)}_{z^{(2)}}] & if\ W=\{z^{(1)}_k, z^{(2)}_k\} \end{cases} \tag{24}$$

For any function $h(\zeta_k)$, $D_{k|k-1}[h(\zeta_k)]$ is given by,

$$D_{k|k-1}[h(\zeta_k)] = \int h(\zeta_k)D_{k|k-1}(\zeta_k)d\zeta_k \tag{25}$$

As usual $P^{(r)}_D = P^{(r)}_D(\zeta_k|X^{(r)}_k)$ denotes the probability of detection of a landmark ζ_k, by rth vehicle and is considered as a constant. Although computationally more demanding, this approach produces an invariant PHD in a statistically principled manner.

C. Implementation of the PHD filter

The PHD prediction and update equations are implemented using a GM implementation of the PHD filter [16]. The PHD of the landmark map posterior at time $k-1$ and the newly appearing landmarks in the joint sensor FOV are represented by a mixture of Gaussians, then the resultant measurement updated PHD at time k can be obtained as a mixture of Gaussian components from which the landmark locations and the number of landmarks can be obtained [16] in order to solve the MVSLAM problem.

V. Particle Filter for Joint Vehicle Trajectories and Landmark Map Posterior Estimation

The joint vehicle trajectories posterior is evaluated using a particle filter, which can be represented by a set of weighted particles denoted by Ω_k as follows,

$$\Omega_k = \left\{w^{[i]}_k, X^{(1),[i]}_{1:k}, X^{(2),[i]}_{1:k}\right\}_{i=1}^{N_s} \tag{26}$$

where, N_s denotes the number of particles and $w_k^{[i]}$ is the weight of ith particle which can be obtained as,

$$w_k^{[i]} \propto l_{k|k-1}(Z_k^{(1)}, Z_k^{(2)}|Z_{1:k-1}^{(1)}, Z_{1:k-1}^{(2)}, X_{0:k}^{(1),[i]}, X_{0:k}^{(2),[i]}) \tag{27}$$

where $l_{k|k-1}(Z_k^{(1)}, Z_k^{(2)}|Z_{1:k-1}^{(1)}, Z_{1:k-1}^{(2)}, X_{0:k}^{(1)}, X_{0:k}^{(2)})$ is the normalization constant in (9). Assuming the number of elements in M_k is Poisson distributed, extending the approach proposed by Mullane et al. in [14] [15], it can be shown that,

$$\begin{aligned} & l_{k|k-1}(Z_k^{(1)}, Z_k^{(2)}|Z_{1:k-1}^{(1)}, Z_{1:k-1}^{(2)}, X_{0:k}^{(1)}, X_{0:k}^{(2)}) \\ &= \prod_{z^{(1)} \in Z_k^{(1)}} \lambda^{(1)} c^{(1)}(z^{(1)}) \prod_{z^{(2)} \in Z_k^{(2)}} \lambda^{(2)} c^{(2)}(z^{(2)}) \\ &\times \exp(N_{k|k} - N_{k|k-1} - \lambda^{(1)} - \lambda^{(2)}) \end{aligned} \tag{28}$$

where $N_{k|k-1}$ and N_k respectively denote the number of landmarks present in the predicted landmark map and the number of landmarks present after updating the landmark map.

Now since the PHD of the landmark map posterior is conditioned on the joint vehicle trajectories, the joint landmark map and vehicle trajectories posterior at time k can be written as,

$$\left\{ w_k^{[i]}, X_k^{(1),[i]}, X_k^{(2),[i]}, D_{k|k}(\zeta_k|Z_{1:k}^{(1)}, Z_{1:k}^{(2)}, X_{0:k}^{(1),[i]}, X_{0:k}^{(2),[i]}) \right\}_{i=1}^{N_s} \tag{29}$$

VI. Simulation Results

Simulation results are used to evaluate the performance of the proposed MVSLAM algorithms. From here onwards the proposed MVSLAM solutions are referred to as RFS-MVSLAM-I and RFS-MVSLAM-II respectively denoting the solutions based on "iterative update method" and the "general multi-sensor update method". Two vehicles with identical control and sensor parameters are driven on two different vehicle trajectories on a simulaiton environment (Fig. 1) consisting of 67 static features. The control parameters were chosen as, the standard deviation of velocity as $0.3\ ms^{-1}$, the standard deviation of heading change as 2^0. The sensor parameters were chosen as, the standard deviation of range as $0.3\ m$ and standard deviation of bearing as 0.5^0. The maximum range of a sensor is set to $30\ m$. The number of false measurements generated by each vehicle is assumed to be Poisson distributed with an average of 5 per scan and has a uniform spatial distribution in the sensor FOV (-180^0 to 180^0). The probability of detection of a feature is is set as 0.95 and the probability of survival of a feature in the landmark map is set as 0.99.

A comparison of estimated vehicle trajectories against the actual vehicle trajectories obained from two sample runs of RFS-MVSLAM-I and RFS-MVSLAM-II algorithms are shown in Fig.2. The RMS vehicle position errors from 25 MC runs are compared in Fig.3. In the long run it can be seen that the RFS-MVSLAM-II algorithm produces better results in the overlapping region compared to the RFS-MVSLAM-I algorithm due to the explicit treatment for mutual feature observations. It can also be seen that the loop closing error remains at an acceptable value of less than $3\ m$ for both RFS-MVSLAM-I AND RFS-MVSLAM-II algorithms.

VII. Computational Complexity

The computational complexity of the proposed MVSLAM solution based on the "Iterative update method" method is $O(\tilde{l}_k n_k N_s)$, where $\tilde{l}_k$ is the number of features present in the joint sensor FOV and n_k is the total number of measurements received from both vehicles. In comparison, for the two vehicle case, if the number of measurements received from each sensor is approximately equal to $\tilde{n}_k$ the computational complexity of the plain vanilla "General multi-sensor update method" based MVSLAM solution is no less than $O(\tilde{l}_k \tilde{n}! N_s)$.

In our solution, we avoid this heavy computational cost by avoiding the creation of partitions that contains elements of the form $\{z_k^{(1)}, z_k^{(2)}\}$, where the corresponding Cartesian distance between them (which is calculated after converting into Cartesian coordinates using the current vehicle estimates) is considerably larger. The reason behind this is that, the two measurements $z_k^{(1)}$ and $z_k^{(2)}$ are not most possibly generated due to the observation of the same feature (mutual feature observations). Hence we can reduce the number of generated binary partitions yielding a reduced computational cost. If the number of common feature observations by two vehicles are given by ϵ_k, then the number of binary partitions generated is given by,

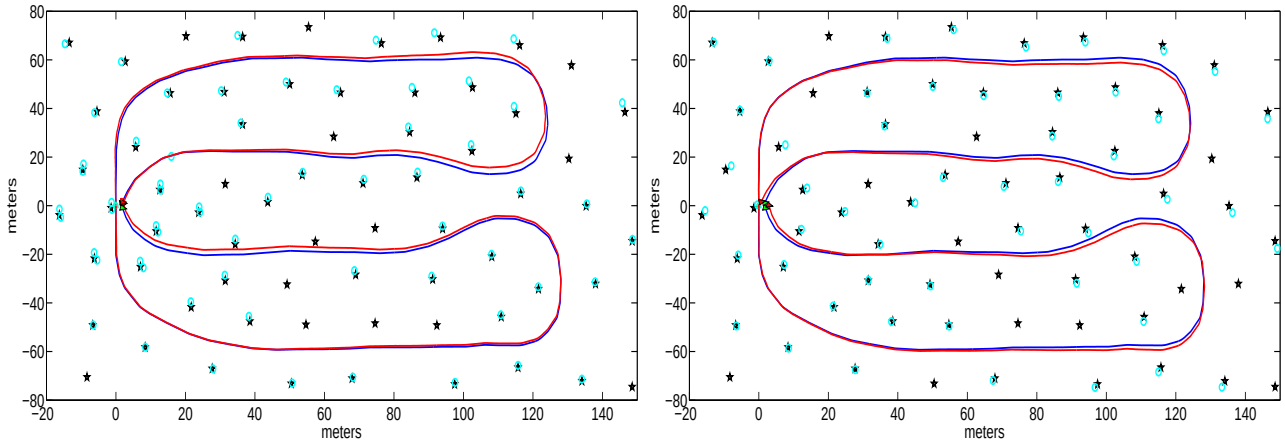
$$\Psi_k = 1 + \binom{\epsilon_k}{1} + \binom{\epsilon_k}{2} + \binom{\epsilon_k}{3} + ... + \binom{\epsilon_k}{\epsilon_k} \tag{30}$$

as a result, the computational complexity can be reduced to $O(\tilde{l}_k \Psi_k n_k N_s)$.

Since the number of mutual feature observations at any given time is comparatively smaller, this results in a improved computational performance. Further improvements are obtained by adopting sensor consistency gating. The same method can be applied to reduce the computational complexity of a larger number of robots. Nevertheless the computational complexity grows with the number of vehicles due to the larger number partitions generated.

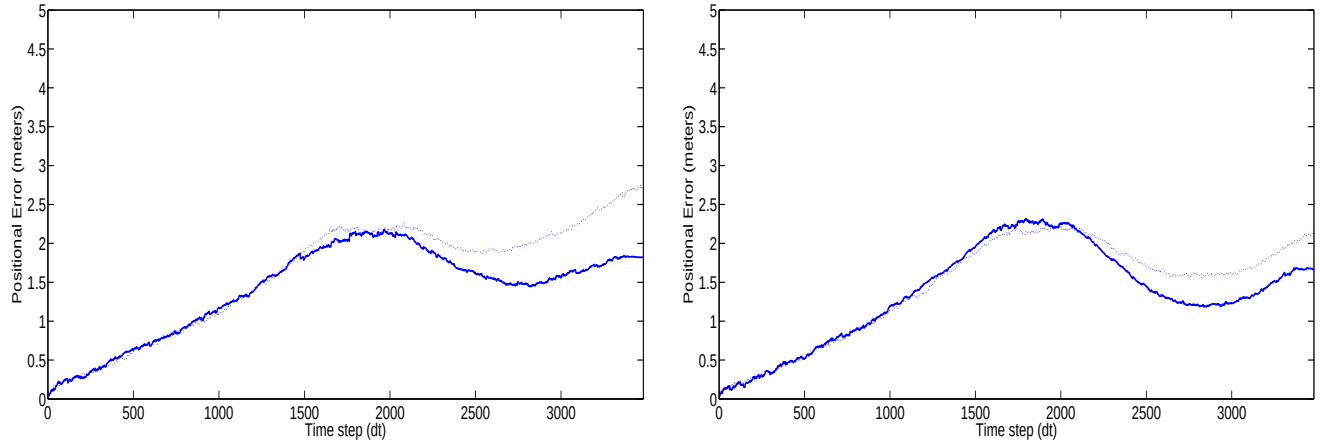
VIII. Conclusion

In this paper we have presented a noval Multi-Vehicle SLAM (MVSLAM) solution by extending the RFS based SLAM filter framework using two recently developed multi-sensor fusion techniques in finite set statistics (FISST). The RFS representation of the landmark map and the measurements enables modeling of the landmark map transition model and observation model in a more natural manner, resulting a Bayesian MVSLAM algorithm with inbuilt map management, data association and clutter filtering. The performance characteristics obtained via the simulation results suggests that the proposed solutions produces robust results under false measurements with data association and landmark detection uncertainty.



(a) RFS-MVSLAM-I with $\lambda^{(r)} = 5$ (b) RFS-MVSLAM-II with $\lambda^{(r)} = 5$

Fig. 2. A comparison of estimated vehicle trajectories (in red), superimposed on ground truth (in blue) with estimated features (cyan circles) and actual features (black stars), of a sample run of each algorithm with the clutter rate $\lambda^{(r)} = 5$.



(a) RFS-MVSLAM-I (b) RFS-MVSLAM-II

Fig. 3. RMS Vehicle position errors of 25 MC runs. Dotted graph corresponds to the first vehicle, while the non-dotted graph corresponds to the second vehicle.

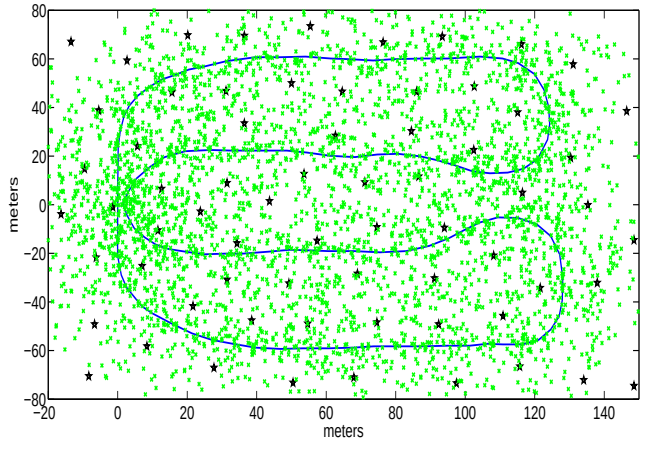


Fig. 1. Ground truth of the vehicle trajectories (in blue) with randomly placed landmarks (black stars) are shown with false features (in green) received from a sample run. The number of landmarks present in the simulation enviornment is 67.

ACKNOWLEDGMENT

This work was supported by National Research Foundation (NRF), Singapore and Center for Environmental Sensing and Modeling (CENSAM) under the auspices of the Singapore-MIT Alliance for Research and Technology (SMART). The work of B.-N. Vo is supported by Australian Research Council under the Future Fellowship FT0991854.

REFERENCES

[1] J.W. Fenwick, P.M. Newman and J.J. Leonard, "Cooperative Concurrent Mapping and Localization," in *Proc. IEEE Int. Conf. Robot. Autom.*, August 2002, pp. 1810–1817.

[2] S.B. Williams, G. Dissanayake and H. Durrant-Whyte, "Towards multi-vehicle simultaneous localization and mapping," in *Proc. IEEE Int. Conf. Robot. Autom.*, August 2002, pp. 2743–2748.

[3] S. Thrun and Y. Liu, "Multi-robot SLAM with sparse extended information filters," in *Proc. Int. Symp. Robot. Res.*, Sienna, Italy, October 2003, pp. 254–266.

[4] A. Howard, "Multi-robot simultaneous localization and mapping using Particle Filters," *The Int. J. Robot. Res.*, vol. 25, no. 12, pp. 1243–1256, December 2006.

[5] D. Fox, J. Ko, K. Konolige, B. Limketkai, D. Schulz and B. Stewart, "Distributed Multirobot Exploration and Mapping," *Proc. IEEE*, vol. 94, no. 7, pp. 1325–1339, July 2006.

[6] X. Zhou and S. Roumeliotis, "Multi-robot SLAM with unknown initial correspondence: the robot rendezvous case," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Sys.*, Oct 2006, pp. 1785–1792.

[7] L. Carlone, M.K. Ng, J. Du, B. Bona and M. Indri, "Rao-Blackwellized Particle Filters Multi Robot SLAM with Unknown Initial Correspondences and Limited Communication," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2010, pp. 243–249.

[8] G. Dissanayake, P.M. Newman, S. Clark, H. Durrant-Whyte and M. Csorba, "A Solution to the Simultaneous Localization and Map Building (SLAM) Problem," *IEEE Trans. Robot. Autom.*, vol. 17, no. 3, pp. 229–241, June 2001.

[9] M. Montemerlo, "FastSLAM: a factored solution to the simultaneous localization and mapping problem with unknown data association," Ph.D. dissertation, School of Computer Science, Carnegie Melon University, 2003.

[10] S. Thrun, Y. Liu, D. Koller, A.Y. Ng, Z. Ghahramani and H. Durrant-Whyte, "Simultaneous Localization and Mapping with Sparse Extended Information Filters," *The Int. J. Robot. Res.*, vol. 23, no. 7-8, pp. 693–716, August 2004.

[11] J. Mullane, B.N. Vo, M.D. Adams and W.S. Wijesoma, "A Random Set Formulation for Bayesian SLAM," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, September 2008, pp. 1043–1049.

[12] R. Mahler, "Multitarget Bayes Filtering via First-Order Multitarget Moments," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 39, no. 4, pp. 1152–1178, January 2004.

[13] ——, *Statistical Multisource-Multitarget Information Fusion*. Norwood, MA: Artech House, 2007.

[14] J. Mullane, B.N. Vo and M.D. Adams, "Rao-Blackwellised PHD SLAM," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2010, pp. 5410–5416.

[15] J. Mullane, B.N. Vo, M.D. Adams and B.T. Vo, "A Random-Finite-Set Approach to Bayesian SLAM," *IEEE Trans. Robot. Autom.*, vol. 27, no. 2, pp. 268–282, February 2011.

[16] B.N. Vo and W.K. Ma, "The Gaussian Mixture Probability Hypothesis Density Filter," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4091–4104, November 2006.

[17] R. Mahler, "The multisensor PHD filter: I. General solution via multitarget calculus," in *Proc. SPIE Sig. Process., Sensor Fusion., and Target Recog. XVIII*. SPIE, April 2009, pp. 1043–1049.