Collaborative Multi-Vehicle SLAM with Moving Object Tracking

Diluka Moratuwage, Ba-Ngu Vo and Danwei Wang

***Abstract*— Although simultaneous localization and mapping (SLAM) algorithms are widely appreciated in mobile robot navigation, they can be further improved to suit practical applications in dynamic environmental conditions. One such important improvement is the detection and tracking of moving objects present in the sensor field of view (FOV). In this paper we propose to extend our recently introduced Collaborative Multi-vehicle SLAM (CMSLAM) solution based on the random finite set (RFS) representation of the feature map and measurements, by tracking both static and dynamic features. We represent static features observed during the SLAM process, along with dynamic features present in the current sensor FOV, as an augmented RFS. The corresponding probability density is propagated using a Bayes recursion, from which the static feature map and the estimates of dynamic feature locations can be obtained. Measurement update in the CMSLAM process is carried out only using the static feature map to take advantage of obvious accuracy improvements.**

I. INTRODUCTION

Since the initial developments in [1], various improvements have been proposed to the simultaneous localization and mapping (SLAM) problem in order to achieve better performance in real-world applications. The extended Kalman filter based SLAM (EKF-SLAM) [2] solution introduced linearized non-linear motion and observation models with Gaussian noise assumption. The FastSLAM algorithm [3] introduced recursive Monte Carlo Sampling (particle filtering) into SLAM, which enabled non-linear non-Gaussian motion models to be used.

All these conventional SLAM solutions and their derivatives solve the SLAM problem by propagating a posterior probability density of a vector consisting of landmark map augmented with vehicle state. Due to this vector based representation, such algorithms require solving certain additional sub-problems such as data association, clutter filtering and map management outside the Bayesian recursion. In addition, landmark detection and landmark survival uncertainties are ignored limiting their practical applicability in real-world solutions. In order to address these issues, in [4] [5] Mullane et al. presented a novel SLAM framework by adopting random finite set (RFS) modeling of the landmark map and the measurements. In [4] [5] Mullane et al. presented a Rao-Blackwellised PHD filtering approach by factorizing the full SLAM posterior into the product of the landmark map posterior conditioned on the vehicle trajectory and the vehicle trajectory posterior. The vehicle trajectory was propagated using a particle filter, while the landmark map was propagated using a Gaussian mixture (GM) implementation of the PHD filter [6], catering for dynamic estimation of landmark locations in the presence of false measurements.

Although these developments made significant advancements, almost all such formulations ignores the dynamic features present in the sensor field of view (FOV). If moving features present in the sensor FOV are not being detected and tracked, they might be filtered out as clutter from the landmark map during the SLAM process. Hence, any high-level application that makes use of the landmark map for making decisions could easily presume that it is safe to navigate while dynamic objects are still present in the current sensor FOV, leading to serious safety issues. The most common solution for this issue is to consider the SLAM with detection and tracking of moving objects (DATMO) as two separate problems and solve them independent of each other at different levels. Although such a solution is adequate for almost all practical applications we believe that SLAM algorithms should be capable of solving both problems at the same time, providing a robust feature map (containing both static and dynamic feature information) for high-level applications.

The very first approach to combine both SLAM and DATMO was proposed by Wang et al. in [7]. In [8], Wang et al. extended their earlier work by presenting a theoretical framework. Their work was based on decomposing the SLAM with DATMO problem into two separate estimators, namely, the SLAM posterior and the DATMO posterior and propagating them simultaneously over time. This approach has a few drawbacks. Among them, decomposing the measurements into stationary objects and moving objects, inapplicability of complex multiple motion models per moving object are most notable. In order to classify measurements as stationary and dynamic, it is required to analyze the data over a sufficient period of time, using complex feature extraction algorithms. This may be undesirable in most practical applications if the results are to be used for making faster navigation decisions. In addition, most of the moving features such as pedestrians, animals or other vehicles may not have a single motion model, their motion models can be a mixture of motion models depending on the situation. Hence, the possibility of having multiple motion models per moving feature have to be taken into account.

Meanwhile in the target tracking community, Pasha et al. [9] proposed a new multi-target model that can accommodate switching dynamics of targets using a Markovian state transition. A closed-form solution for this so called jump markov

Diluka Moratuwage and Wang Dan Wei are with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, 639798. mora0001@ntu.edu.sg, EDWWANG@ntu.edu.sg

Ba-Ngu Vo is with the Department of Electrical and Computer Engineering, Curtin University, Australia. ba-ngu.vo@curtin.edu.au

multi-target model was also proposed based on the GM-PHD filter [6]. This solution is capable of solving a wide range of practical applications in the multi-target tracking literature that are deemed intractable using conventional solutions.

In this paper, we present a novel solution to the Collaborative Multi-vehicle SLAM with moving object tracking (CMSLAMMOT) problem, by extending our recently proposed CMSLAM solution based on the RFS representation [10] using Pasha's solution [9] in order to track moving objects, with complex multiple motion models, present in the sensor FOV. Efficacy of the proposed solution is demonstrated using simulation and real experimental results.

Section II formulates the CMSLAMMOT problem. Section III explains the static and dynamic feature map posteriors propagation method, while section IV explains the paths posterior estimation method. Section V and section VI presents the simulation results and experimental results respectively. Section VII concludes the paper.

II. FORMULATION OF PROBABILISTIC RANDOM FINITE SET CMSLAM WITH MOVING TARGET TRACKING

For the illustration purpose, let's consider the case, where two robots perform CMSLAM with MOT. Let $U_{1:k}^{(r)} = [U_1^{(r)}, U_2^{(r)}, ..., U_k^{(r)}]^T$ denote the time sequence of control commands applied to the vehicle $r, (r = 1, 2)$ up to time k, where $U_k^{(r)}$ denotes the control command applied at time k. Let the time sequence of pose 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 with respect to the global frame of reference, at time k. Let the time sequence of sets of measurements obtained using exteroceptive 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 at time k, from vehicle r, while $n_k^{(r)}$ denotes the number of measurements. Let the augmented feature map at time k be denoted by the set $M_k = \{m_{k,1}, m_{k,2}, ..., m_{k,l_k}\}$, where l_k denotes the number of features available in the map at time k and each m_k denotes the state of a feature.

In conventional SLAM solutions, the terms, feature and landmark are interchangeably used in order to specify a static distinguishable object whose position with respect to the vehicle can be extracted from the sensor measurements. But in CMSLAMMOT, we will adopt the term feature to specify any distinguishable object where unlike conventional SLAM it can either be a static feature or a moving (dynamic) feature. Moreover such features can change their dynamics, meaning a dynamic feature can become a static feature while a static feature can become a dynamic feature at any point in time. Hence, a more informative definition for the state of a feature is required.

A. RFS CMSLAMMOT filter

In the CMSLAMMOT problem we are required to evaluate the two posterior probability distributions,

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

and,

$$p_{k|k}(M_k | Z_{1:k}^{(1)}, Z_{1:k}^{(2)}, X_{0:k}^{(1)}, X_{0:k}^{(2)}) \quad (2)$$

where S_k denotes the set of static features present in the feature map at time k. The distribution (1) is the CMSLAM posterior and distribution (2) is the augmented feature map posterior conditioned on the history of vehicle poses. M_k consist of both static features observed during the CMSLAM process and dynamic features present in the current joint sensor field of view (FOV). We can rewrite M_k as, $M_k = S_k \uplus D_k$, where D_k is the set of dynamic features present in the current joint sensor FOV.

Let's start with Montemerlo's approach in FastSLAM [3] to factorize the CMSLAM posterior (1) into the history of vehicle poses posterior and the static feature map posterior conditioned on the history of vehicle poses as,

$$\begin{aligned} &p_{k|k}(S_k, X_{1:k}^{(1)}, X_{1:k}^{(2)} | Z_{1:k}^{(1)}, Z_{1:k}^{(2)}, U_{1:k}^{(1)}, U_{1:k}^{(2)}, X_0^{(1)}, X_0^{(2)}) \\ &= p_{k|k}(S_k | Z_{1:k}^{(1)}, Z_{1:k}^{(2)}, X_{1:k}^{(1)}, X_{1:k}^{(2)}) \\ &\times p_{k|k}(X_{1:k}^{(1)}, X_{1:k}^{(2)} | Z_{1:k}^{(1)}, Z_{1:k}^{(2)}, U_{1:k}^{(1)}, U_{1:k}^{(2)}, X_0^{(1)}, X_0^{(2)}) \end{aligned} \quad (3)$$

where static feature map posterior can be obtained by marginalizing the distribution (2) as follows,

$$\begin{aligned} &p_{k|k}(S_k | Z_{1:k}^{(1)}, Z_{1:k}^{(2)}, X_{0:k}^{(1)}, X_{0:k}^{(2)}) \\ &= \int p_{k|k}(S_k \uplus D_k | Z_{1:k}^{(1)}, Z_{1:k}^{(2)}, X_{0:k}^{(1)}, X_{0:k}^{(2)}) \delta D \end{aligned} \quad (4)$$

Although static feature map posterior (4) is computationally intractable as it is, we can obtain its corresponding PHD component by propagating the augmented feature map posterior (2) using a PHD filter, which is capable of tracking features with multiple motion models (including static features). Hence using the PHD of the static feature map posterior, the CMSLAM posterior distribution (3) can be evaluated. Moreover the locations of the dynamic features present in the joint sensor FOV can also be obtained using the PHD component that corresponds to dynamic feature map posterior. These steps are explained in greater details in the rest of the paper.

B. Augmented RFS Feature Map Representation

The feature map can be more naturally and effectively represented as a RFS, M_K,

$$M_k = \{m_{k,1}, m_{k,2}, \ldots, m_{k,l_k}\} \in F(\mathscr{M}) \quad (5)$$

where each $m_k \in \mathscr{M}$ and $\mathscr{M}$ is the state space of the features. $F(\mathscr{M})$ is the collection all finite subsets of $\mathscr{M}$. In order to cater for features with multiple motion models, we extend the state of a feature from the conventional SLAM solution to a broader definition as follows.

Let $m_{k,i}$ be the state of any feature in the feature map at time k, then $m_{k,i}$ is given by $(\zeta_{k,i}, \beta_{k,i})$ where $\zeta_{k,i}$ is the kinematic state and $\beta_{k,i}$ is the type of the feature. Features are classified based on their motion model. For example, static features are classified as type 0, i.e. $\beta_k = 0$, while moving features with motion model I, are of type 1, i.e. $\beta_k = 1$ and moving features with motion model II are of type 2, i.e. $\beta_k = 2$ and so on. In addition, features are allowed to switch between different motion models (types) over time allowing a comprehensive treatment for dynamic objects. This behaviour is modeled using a jump Markov transition model, catering for changes in motion models with varying probabilities.

C. Augmented RFS Feature Map Transition Model

The newly appearing features in the joint sensor FOV at time k are modeled as the RFS $\Gamma_k(X_k^{(1)}, X_k^{(2)})$. Let the RFS of landmark map at time $k-1$ be denoted by M_{k-1}, then the landmark map transition model at time k can be written as,

$$M_k = \Gamma_k(X_k^{(1)}, X_k^{(2)}) \cup \left[\bigcup_{\xi_{k-1} \in M_{k-1}} \Upsilon(\xi_{k-1}) \right] \quad (6)$$

where, the Bernoulli RFS $\Upsilon(\xi_{k-1})$ denotes the predicted state of the landmark $\xi_{k-1} \in M_{k-1}$, which takes the form $\{(\zeta_k, \beta_k)\}$ with probability of reappearance $P_S(\zeta_{k-1}, \beta_{k-1})$ or $\emptyset$ with probability $1 - P_S(\zeta_{k-1}, \beta_{k-1})$. The probability density of state transition from $(\zeta_{k-1}, \beta_{k-1})$ to (ζ_k, β_k), conditioned on the probability of survival is given by, $f_M(\zeta_k, \beta_k | \zeta_{k-1}, \beta_{k-1}, X_k^{(1)}, X_k^{(2)})$. Note that, $\Gamma_k(X_k^{(1)}, X_k^{(2)})$ is independent of the landmarks that are reappearing from the map M_{k-1}. The corresponding RFS map transition density can be written as,

$$\begin{aligned} &f_M(M_k | M_{k-1}, X_k^{(1)}, X_k^{(2)}) \\ &= \sum_{W \subseteq M_k} f_M(W | M_{k-1}) f_\Gamma(M_k - W | X_k^{(1)}, X_k^{(2)}) \end{aligned} \quad (7)$$

where, $f_M(W|M_{k-1})$ is the transition density of the set of features existing in the map, M_{k-1} from time $k-1$ to time k, while $f_\Gamma(M_k - W | X_k^{(1)}, X_k^{(2)})$ represents the density of the RFS $\Gamma_k(X_k^{(1)}, X_k^{(2)})$.

In order to track features with multiple motion models, we adopt the same method as presented in [9] and decompose the transition density as follows.

$$\begin{aligned} &f_M(\zeta_k, \beta_k | \zeta_{k-1}, \beta_{k-1}, X_k^{(1)}, X_k^{(2)}) \\ &= \hat{f}_M(\zeta_k | \zeta_{k-1}, \beta_{k-1}, X_k^{(1)}, X_k^{(2)}) t_{k|k-1}(\beta_k | \beta_{k-1}) \end{aligned} \quad (8)$$

where $t_{k|k-1}(\beta_k|\beta_{k-1})$ denotes the Markov transition probability of switching mode from β_{k-1} to β_k.

D. RFS Measurement Model

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

$$Z_k^{(r)} = C_k^{(r)} \cup \left[\bigcup_{\xi_k \in M_k} \Theta_k^{(r)}(\xi_k) \right] \quad (9)$$

where $\Theta_k^{(r)}(\xi_k)$ is a Bernoulli RFS representing the measurement corresponding to landmark $\xi_k \in M_k$ having state (ζ_k, β_k). Due to the limited field of view (FOV) of the sensor, $\Theta_k^{(r)}(\xi_k)$ can have a value of the form $\{z_k^{(r)}\}$ with probability of detection $P_D^{(r)}(\zeta_k, \beta_k | X_k^{(r)})$ or $\emptyset$ with a probability of $1 - P_D^{(r)}(\zeta_k, \beta_k | X_k^{(r)})$. Note that probability of detection, is a function of landmark state and vehicle position. Conditioned on detection, measurement likelihood function corresponds to the observation of landmark with state (ζ_k, β_k) by the rth vehicle is given by $g_k^{(r)}(z^{(r)} | \zeta_k, \beta_k, X_k^{(r)})$. The corresponding RFS measurement density can be written as,

$$\begin{aligned} &g_k^{(r)}(Z_k^{(r)} | M_k, X_k^{(r)}) \\ &= \sum_{W \subseteq Z_k^{(r)}} g_Z^{(r)}(W | M_k, X_k^{(r)}) g_C^{(r)}(Z_k^{(r)} - W) \end{aligned} \quad (10)$$

where, $g_Z^{(r)}(W|M_k, X_k^{(r)})$ denotes the density of the feature observations, while $g_C^{(r)}(Z_k^{(r)} - W)$ denotes the density of clutter.

E. Probability Density vs. Probability Hypothesis Density

Assume, the number of elements in a RFS M on $\mathscr{M}$ is Poisson distributed, with each one of them is identically and independently distributed. Then the probability density of RFS M can be written as [11] [12],

$$p(M) = \frac{\prod_{\zeta \in M} v(\zeta)}{exp(\int v(\zeta) d\zeta)} \quad (11)$$

where, v denotes the PHD of density $p(M)$. Total number of elements of M on a region $W \subseteq \mathscr{M}$ can be obtained as, $N = \int_W v(\zeta) d\zeta$ (for more information see [12]).

III. Static and Dynamic Feature Map Posteriors Estimation

The augmented feature map posterior conditioned on the history of vehicle poses is given by,

$$p_{k|k}(M_k | Z_{1:k}^{(1)}, Z_{1:k}^{(2)}, X_{0:k}^{(1)}, X_{0:k}^{(2)}) \quad (12)$$

This distribution can be evaluated by propagating the PHD of M_k using a PHD filter [11] [13]. We start with factorizing the posterior (12) into a prediction and update steps as follows,

$$\begin{aligned} &p_{k|k}(M_k | Z_{1:k}^{(1)}, Z_{1:k}^{(2)}, X_{0:k}^{(1)}, X_{0:k}^{(2)}) \\ &= g_k(Z_k^{(1)}, Z_k^{(2)} | M_k, X_k^{(1)}, X_k^{(2)}) \\ &\times \frac{p_{k|k-1}(M_k | Z_{1:k-1}^{(1)}, Z_{1:k-1}^{(2)}, X_{0:k}^{(1)}, X_{0:k}^{(2)})}{l_{k|k-1}(Z_k^{(1)}, Z_k^{(2)} | Z_{1:k-1}^{(1)}, Z_{1:k-1}^{(2)}, X_{0:k}^{(1)}, X_{0:k}^{(2)})} \end{aligned} \quad (13)$$

where the denominator in (13) is the normalization constant and,

$$\begin{aligned} &p_{k|k-1}(M_k | Z_{1:k-1}^{(1)}, Z_{1:k-1}^{(2)}, X_{0:k}^{(1)}, X_{0:k}^{(2)}) \\ &= \int f_M(M_k | M_{k-1}, X_k^{(1)}, X_k^{(2)}) \\ &\times p_{k-1}(M_{k-1} | Z_{1:k-1}^{(1)}, Z_{1:k-1}^{(2)}, X_{0:k-1}^{(1)}, X_{0:k-1}^{(2)}) \delta M_{k-1} \end{aligned} \quad (14)$$

where equations (13) and (14) are augmented feature map update posterior and prediction posterior respectively. Their PHDs are recursively propagated using a PHD filter as follows.

A. PHD prediction step

If PHD of the augmented feature map posterior with type β_{k-1} at time $k-1$ is given by,

$$D_{k-1|k-1}(\zeta_{k-1},\beta_{k-1}|Z^{(1)}_{1:k-1},Z^{(2)}_{1:k-1},X^{(1)}_{0:k-1},X^{(2)}_{0:k-1}) \quad (15)$$

then, PHD of the prediction distribution (14) corresponds to features with type β_k is given by,

$$\begin{aligned} &D_{k|k-1}(\zeta_k,\beta_k|Z^{(1)}_{1:k-1},Z^{(2)}_{1:k-1},X^{(1)}_{0:k},X^{(2)}_{0:k}) \\ &= b_k(\zeta_k,\beta_k|X^{(1)}_k,X^{(2)}_k) \\ &+ \sum_{\beta_{k-1}=0}^{a}\int P_S(\zeta_{k-1},\beta_{k-1}) f_M(\zeta_k,\beta_k|\zeta_{k-1},\beta_{k-1},X^{(1)}_k,X^{(2)}_k) \\ &\times D_{k-1|k-1}(\zeta_{k-1},\beta_{k-1}|Z^{(1)}_{1:k-1},Z^{(2)}_{1:k-1},X^{(1)}_{0:k-1},X^{(2)}_{0:k-1})d\zeta_{k-1} \end{aligned} \quad (16)$$

In here $b_k(\zeta_k,\beta_k|X^{(1)}_k,X^{(2)}_k)$ denotes the intensity of the newly appearing features of type β_k in the joint sensor FOV, which can be written using Campbell's theorem (for marked point processes) [12] as follows,

$$b_k(\zeta_k,\beta_k|X^{(1)}_k,X^{(2)}_k) = \pi_k(\beta_k)\tilde{b}_k(\zeta_k|\beta_k,X^{(1)}_k,X^{(2)}_k) \quad (17)$$

where $\pi_k(\beta_k)$ denotes the birth intensity and $\tilde{b}_k(\zeta_k|\beta_k,X^{(1)}_k,X^{(2)}_k)$ denotes the distribution of birth state, while a denotes the total number of feature types (having unique motion models).

B. PHD update step

Assuming the number of false features produced by the exteroceptive sensor mounted on the rth vehicle ($r=1,2$) is Poisson distributed with an average of $\lambda^{(r)}_c$, and their physical distribution given by $c^{(r)}_k(z^{(r)}_k)$ is uniform over the sensor FOV, then PHD of the feature map update posterior distribution (13) of features with type β_k can be obtained as,

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

where the abbreviation $D_{k|k-1}(\zeta_k,\beta_k)$ represents the PHD of the predicted feature map posterior and $P^{(r)}_D$ represents the detection probability of the sensor mounted on the rth vehicle given by,

$$P^{(r)}_D = P^{(r)}_D(\zeta_k,\beta_k|X^{(r)}_k) \quad (19)$$

and the summation in (17) is taken over all so called "binary partitions" $\mathscr{P}$ of $Z_k = Z^{(1)}_k \cup Z^{(2)}_k$ (see [14] for more information). The notation, "$\mathscr{P}\boxminus_2 Z_k$" stands for "$\mathscr{P}$ partitions Z_k into binary cells W", where $W\in\mathscr{P}$ has one of the following forms,

$$W=\left\{z^{(1)}_k\right\},\ W=\left\{z^{(2)}_k\right\},\ W=\left\{z^{(1)}_k,z^{(2)}_k\right\} \quad (20)$$

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

$$\rho_W(\zeta_k) = \begin{cases} \dfrac{P^{(1)}_D l^{(1)}_{z^{(1)}_k}(1-P^{(2)}_D)}{1+D_{k|k-1}[P^{(1)}_D l^{(1)}_{z^{(1)}_k}(1-P^{(2)}_D)]} & if\ W=\{z^{(1)}_k\} \\ \dfrac{(1-P^{(1)}_D)l^{(2)}_{z^{(2)}_k}P^{(2)}_D}{1+D_{k|k-1}[(1-P^{(1)}_D)l^{(2)}_{z^{(2)}_k}P^{(2)}_D]} & if\ W=\{z^{(2)}_k\} \\ \dfrac{P^{(1)}_D l^{(1)}_{z^{(1)}_k}P^{(2)}_D l^{(2)}_{z^{(2)}_k}}{D_{k|k-1}[P^{(1)}_D l^{(1)}_{z^{(1)}_k}P^{(2)}_D l^{(2)}_{z^{(2)}_k}]} & if\ W=\{z^{(1)}_k,z^{(2)}_k\} \end{cases} \quad (21)$$

and,

$$\omega_{\mathscr{P}} = \frac{\prod_{W\in\mathscr{P}} d_W}{\sum_{\mathscr{Q}\boxminus_2 Z_k}\prod_{W\in\mathscr{Q}} d_W} \quad (22)$$

and,

$$d_W = \begin{cases} 1+D_{k|k-1}[P^{(1)}_D l^{(1)}_{z^{(1)}_k}(1-P^{(2)}_D)] & if\ W=\{z^{(1)}_k\} \\ 1+D_{k|k-1}[(1-P^{(1)}_D)l^{(1)}_{z^{(1)}_k}P^{(2)}_D] & if\ W=\{z^{(2)}_k\} \\ D_{k|k-1}[P^{(1)}_D l^{(1)}_{z^{(1)}_k}P^{(2)}_D l^{(2)}_{z^{(2)}_k}] & if\ W=\{z^{(1)}_k,z^{(2)}_k\} \end{cases} \quad (23)$$

where for any function $h(\zeta_k,\beta_k)$, the abbreviation $D_{k|k-1}[h(\zeta_k,\beta_k)]$ is given by,

$$D_{k|k-1}[h(\zeta_k,\beta_k)] = \sum_{r_k=0}^{a}\int h(\zeta_k,\beta_k)D_{k|k-1}(\zeta_k,\beta_k)d\zeta_k \quad (24)$$

and

$$l^{(1)}_{z^{(1)}_k}(\zeta_k,\beta_k) = \frac{g^{(1)}_k(z^{(1)}_k|\zeta_k,\beta_k,X^{(1)}_k)}{\lambda^{(1)}_c c^{(1)}_k(z^{(1)}_k)} \quad (25)$$

$$l^{(2)}_{z^{(2)}_k}(\zeta_k,\beta_k) = \frac{g^{(2)}_k(z^{(2)}_k|\zeta_k,\beta_k,X^{(2)}_k)}{\lambda^{(2)}_c c^{(2)}_k(z^{(2)}_k)} \quad (26)$$

Hence, the PHD corresponds to any feature type β_k can be obtained using equation (18). The PHD of the static feature map can be obtained from,

$$D_{k|k}(\zeta_k,\beta_k=0|Z^{(1)}_{1:k},Z^{(2)}_{1:k},X^{(1)}_{0:k},X^{(2)}_{0:k}) \quad (27)$$

and PHD of all types of features can be obtained as,

$$\sum_{\beta_k=1}^{a} D_{k|k}(\zeta_k,\beta_k|Z^{(1)}_{1:k},Z^{(2)}_{1:k},X^{(1)}_{0:k},X^{(2)}_{0:k}) \quad (28)$$

Once the PHD is known, actual locations of features of any type can be easily found [6] [5]. In order to calculate the particle weights (discussed in the section IV), we need to evaluate the number of predicted and actual static features present in the map at time k. The number of predicted static features can be obtained as,

$$N_{k|k-1} = \int D_{k|k-1}(\zeta_k,\beta_k=0|Z^{(1)}_{1:k},Z^{(2)}_{1:k},X^{(1)}_{0:k},X^{(2)}_{0:k})\delta\zeta_k \quad (29)$$

and the number of actual static features after measurement update can be obtained as,

$$N_{k|k} = \int D_{k|k}(\zeta_k, \beta_k = 0 | Z^{(1)}_{1:k}, Z^{(2)}_{1:k}, X^{(1)}_{0:k}, X^{(2)}_{0:k}) \delta \zeta_k \quad (30)$$

The PHD prediction (16) and update (18) equations are evaluated using a GM-PHD filter [6] implementation.

IV. Paths Posterior Estimation

By far, particle filter based vehicle trajectory estimation method introduced in FastSLAM [3] can be considered as the most suitable approach for practical SLAM solutions due to the adoptability of complex non-linear motion models in highly dynamic, unstructured and uncertain environmental conditions. In the same spirit, we propose propagating the vehicle trajectory posterior using a particle filter as follows.

The vehicle trajectories posterior at time k is given by,

$$p_{k|k}(X^{(1)}_{1:k}, X^{(2)}_{1:k} | Z^{(1)}_{1:k}, Z^{(2)}_{1:k}, U^{(1)}_{1:k}, U^{(2)}_{1:k}, X^{(1)}_0, X^{(2)}_0) \quad (31)$$

which can be represented by a set of weighted particles denoted by Ω_k as follows,

$$\Omega_k = \left\{ w^{[i]}_k, X^{(1),[i]}_{1:k}, X^{(2),[i]}_{1:k} \right\}_{i=1}^{N_s} \quad (32)$$

where, $w^{[i]}_k$ is the weight of ith particle, while N_s denotes the number of particles. Similar to FastSLAM, the particle set is recursively propagated in time, using sequential importance re-sampling (SIR) approach as follows.

Assume that the joint vehicle trajectories posterior at time $k-1$ is represented by the set of weighted particles Ω_{k-1}. Then at time k, a new joint vehicle pose is sampled from each particle as follows,

$$\begin{aligned}(X^{(1),[i]}_k, & X^{(2),[i]}_k) \\ & \sim f_X(X^{(1)}_k, X^{(2)}_k | X^{(1),[i]}_{k-1}, X^{(2),[i]}_{k-1}, U^{(1)}_k, U^{(2)}_k)\end{aligned} \quad (33)$$

Although (33) is given as a joint transition, due to the conditional independence of each vehicle's motion, individual vehicle pose can be sampled using the corresponding motion model. The new joint vehicle pose, $(X^{(1),[i]}_k, X^{(2),[i]}_k)$, is then added to the set of particles Ω_{k-1}, creating a temporary set of particles distributed according to the proposal distribution given by,

$$q_{k|k}(X^{(1),[i]}_{1:k}, X^{(2),[i]}_{1:k} | Z^{(1)}_{1:k-1}, Z^{(2)}_{1:k-1}, U^{(1)}_{1:k}, U^{(2)}_{1:k}, X^{(1)}_0, X^{(2)}_0) \quad (34)$$

Now, each particle in the temporary set is assigned an importance weight given by,

$$\begin{aligned} w^{[i]}_k & \\ = & \frac{\text{target distribution}}{\text{proposal distribution}} \\ = & \frac{p_{k|k}(X^{(1),[i]}_{1:k}, X^{(2),[i]}_{1:k} | Z^{(1)}_{1:k}, Z^{(2)}_{1:k}, U^{(1)}_{1:k}, U^{(2)}_{1:k}, X^{(1)}_0, X^{(2)}_0)}{q_{k|k}(X^{(1),[i]}_{1:k}, X^{(2),[i]}_{1:k} | Z^{(1)}_{1:k-1}, Z^{(2)}_{1:k-1}, U^{(1)}_{1:k}, U^{(2)}_{1:k}, X^{(1)}_0, X^{(2)}_0)} \end{aligned} \quad (35)$$

Ignoring the normalization constant and factorizing the target and proposal distributions, it can be shown that,

$$w^{[i]}_k \propto l_{k|k-1}(Z^{(1)}_k, Z^{(2)}_k | Z^{(1)}_{1:k-1}, Z^{(2)}_{1:k-1}, X^{(1),[i]}_{0:k}, X^{(2),[i]}_{0:k}) \quad (36)$$

The value of $l_{k|k-1}(Z^{(1)}_k, Z^{(2)}_k | Z^{(1)}_{1:k-1}, Z^{(2)}_{1:k-1}, X^{(1),[i]}_{0:k}, X^{(2),[i]}_{0:k})$ can be obtained using the same factorization as in (13) on the static feature map posterior $p_{k|k}(S_k | Z^{(1)}_{1:k}, Z^{(2)}_{1:k}, X^{(1)}_{1:k}, X^{(2)}_{1:k})$ by using the approach proposed by Mullane et al. in [4] [5] as follows,

$$\begin{aligned} & l_{k|k-1}(Z^{(1)}_k, Z^{(2)}_k | Z^{(1)}_{1:k-1}, Z^{(2)}_{1:k-1}, X^{(1)}_{0:k}, X^{(2)}_{0:k}) \\ & = g_k(Z^{(1)}_k, Z^{(2)}_k | S_k, X^{(1)}_k, X^{(2)}_k) \\ & \times \frac{p_{k|k-1}(S_k | Z^{(1)}_{1:k-1}, Z^{(2)}_{1:k-1}, X^{(1)}_{0:k}, X^{(2)}_{0:k})}{p_{k|k}(S_k | Z^{(1)}_{1:k}, Z^{(2)}_{1:k}, X^{(1)}_{0:k}, X^{(2)}_{0:k})} \end{aligned} \quad (37)$$

which is true for all values of S_k. Therefore without loss of generality, by assigning $S_k = \emptyset$,

$$\begin{aligned} & l_{k|k-1}(Z^{(1)}_k, Z^{(2)}_k | Z^{(1)}_{1:k-1}, Z^{(2)}_{1:k-1}, X^{(1)}_{0:k}, X^{(2)}_{0:k}) \\ & = g_k(Z^{(1)}_k, Z^{(2)}_k | \phi, X^{(1)}_k, X^{(2)}_k) \\ & \times \frac{p_{k|k-1}(\phi | Z^{(1)}_{1:k-1}, Z^{(2)}_{1:k-1}, X^{(1)}_{0:k}, X^{(2)}_{0:k})}{p_{k|k}(\phi | Z^{(1)}_{1:k}, Z^{(2)}_{1:k}, X^{(1)}_{0:k}, X^{(2)}_{0:k})} \end{aligned} \quad (38)$$

Assuming the number of elements in S_k, is Poisson distributed, we can re-write (38) using the definition (11) as follows,

$$\begin{aligned} & l_{k|k-1}(Z^{(1)}_k, Z^{(2)}_k | Z^{(1)}_{1:k-1}, Z^{(2)}_{1:k-1}, X^{(1),[i]}_{0:k}, X^{(2),[i]}_{0:k}) \\ & = \prod_{z^{(1)}_k \in Z^{(1)}_k} \lambda^{(1)}_c c^{(1)}_k(z^{(1)}_k) \prod_{z^{(2)}_k \in Z^{(2)}_k} \lambda^{(2)}_c c^{(2)}_k(z^{(2)}_k) \\ & \times \exp(N^{[i]}_{k|k} - N^{[i]}_{k|k-1} - \lambda^{(1)}_c - \lambda^{(2)}_c) \end{aligned} \quad (39)$$

Now since the static map is empty, we consider all the received measurements as false alarms (clutter). In addition, it is assumed that these false features are independently and identically distributed over each vehicle's sensor FOV with the number of false features being Poisson distributed.

The importance weight of each particle in the temporary set is normalized and a new set of N_s particles are drawn with replacement, where each particles is sampled with a probability proportional to it's importance weight. The resultant particle set with their importance weights, denoted by Ω_k, represents the joint vehicle trajectory posterior (31).

V. Simulation results

Five dynamic features having non-linear dynamic motion models are used along with a set of 57 randomly placed static features (Fig.1). Two vehicles having identical vehicular parameters as summarized in Table I are driven on two predetermined trajectories while performing CMSLAMMOT.

Three different motion models are used. Model $\beta_k = 0$ corresponds to the static features having the motion model of dirac delta function. Model $\beta_k = 1$ and $\beta_k = 2$ denote two dynamic motion models having the following motion model,

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \\ \theta_{k+1} \end{bmatrix} = \begin{bmatrix} x_k + cos(\theta_k + \Omega_k) V_k \Delta T \\ y_k + sin(\theta_k + \Omega_k) V_k \Delta T \\ \theta_k + \Omega_k \end{bmatrix} \quad (40)$$

where, ΔT denotes the time step. Ω_k denotes the change of heading angle and V_k denotes the translational velocity, which are given by,

$$\Omega_k = \Omega + \vartheta_k \tag{41}$$

$$V_k = V + \nu_k \tag{42}$$

In here ϑ_k and ν_k represents zero mean Gaussian noise having standard deviations $\sigma_\vartheta = 0.5\ ^\circ$ and $\sigma_\nu = 0.3$ ms^{-1} respectively. In both models, $\beta_k = 1$ and $\beta_k = 2$, V is set to 5 ms^{-1} while Ω is set at 2° and -2° respectively.

Trajectories of the dynamic features are generated using (40). Switching between different motion models is modeled by the Markovian transition probability matrix,

$$[t_{k|k-1}(\beta_k|\beta_{k-1})] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0.6 & 0.4 \\ 0 & 0.4 & 0.6 \end{bmatrix} \tag{43}$$

and the distribution of the motion models at birth is given by,

$$[\pi_k(\beta_k)] = [.9\ \ .05\ \ .05] \tag{44}$$

Probability of survival of each feature is given by,

$$[P_S(\beta_k)] = [0.99\ 0.95\ 0.95] \tag{45}$$

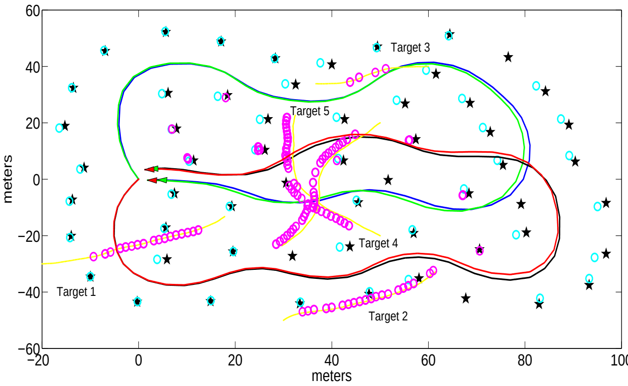
and the probability of detection of each feature type is given by,

$$[P_D(\beta_k)] = [0.95\ 0.93\ 0.93] \tag{46}$$

Fig. 1 shows estimated vehicle trajectories, estimated static feature locations and trajectories of estimated dynamic features from a sample run. Target 1 moves across the FOV of the second vehicle, while target 2 starts within the FOV and then move outward. Target 3 move towards the first vehicle from outside the FOV. Target 4 and Target 5 start within the FOV of first and second vehicles respectively and cross their FOVs navigating through the overlapped area of FOVs. A comparison of RMS vehicle position errors (of 15 Monte Carlo (MC) runs) between the original CMSLAM algorithm [10] and the proposed CMSLAMMOT solution is shown in Fig.2. It is apparent that the proposed CMSLAMMOT solution is capable of tracking dynamic features present in the sensor FOV without compromising the mapping and localization accuracy of the CMSLAM solution. Both vehicles produce almost identical RMS positional errors (with less than 2m loop closing error).

VI. Experimental Results

Two robots (each equipped with a laser scanner and a fiber optic gyroscope) are simultaneously driven on two pre-planned trajectories (with known waypoints) and two pedestrians (with reflectors to improve the feature detection/extraction) are asked to perform various maneuvers across and along the sensor FOV of each vehicle, and CMSLAMMOT is performed using the acquired data. Average speed of the vehicle is approximately 0.4 m/s and the range of observations are limited to 10m with the bearing range



(a) Vehicle 1

Fig. 1. A comparison of estimated vehicle trajectories (first vehicle in green and second in red) superimposed on the ground truth (first vehicle in blue and second in black). Actual static feature locations are shown in black stars and the corresponding estimates are shown in cyan circles. Actual trajectories of the dynamic features are shown in yellow and the corresponding estimates are shown in magenta circles.

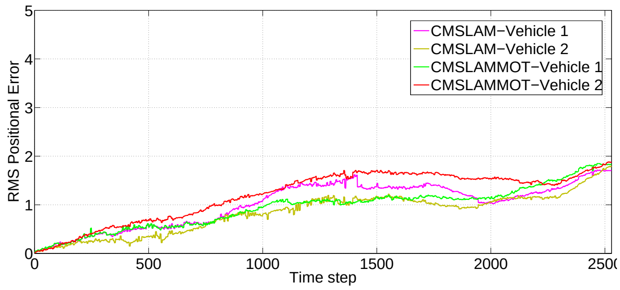


Fig. 2. A comparison of RMS vehicle position errors (of 15 MC runs) between the CMSLAM algorithm and the proposed CMSLAMMOT solution.

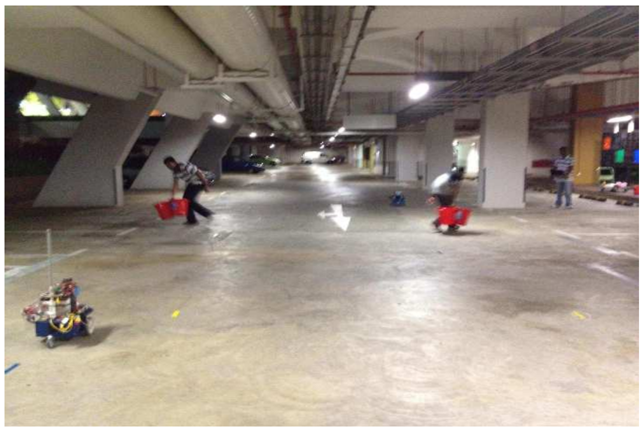


Fig. 3. A picture taken during the experiment. Two pedestrians are crossing the overlapped sensor FOVs at the same time.

TABLE I

PARAMETERS USED IN THE SIMULATION

Vehicle Parameters		Values
Velocity	V	2m/s
Sensor FOV	Range (r)	0 - 30m
	Bearing (b)	$-\pi/2$ - $+\pi/2$
Control Noise	Velociy (σ_v)	0.2m/s
	Steering Angle (σ_a)	2^0
Measurement Noise	Range (σ_r)	0.1m
	Bearing (σ_b)	1^0
Clutter rate	$\lambda_c^{(r)}$	3

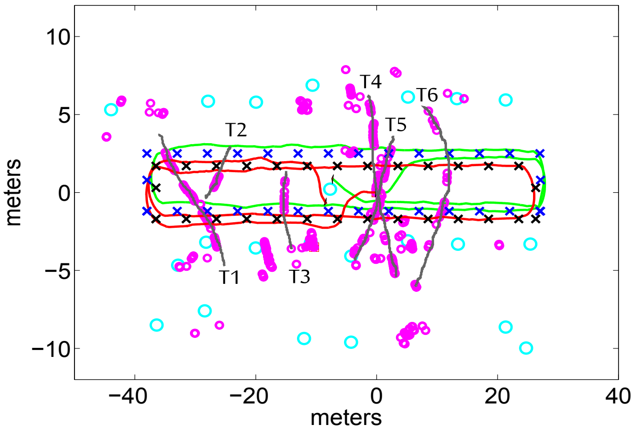


Fig. 4. Estimated vehicle trajectories (first vehicle in green and the second in red) superimposed on the known waypoints (first vehicle in blue crosses and the second in red crosses) along with estimated static feature locations (in cyan circles). Estimated dynamic feature trajectories are shown in magenta and the corresponding actual trajectories (approximately) are shown in grey. Each magenta circle corresponds to an estimated location of the trajectory of a dynamic feature.

from 90° to 90°. Number of particles used for the estimation of vehicle trajectories is 100.

Fig.4 shows the estimated vehicle trajectories and the estimated dynamic feature trajectories, superimposed on the actual vehicle trajectories and the actual dynamic feature trajectories (approximate). Actual dynamic feature trajectories are labelled from T1-T6. The trajectories T1, T2, T3 and T6 are generated by the pedestrians crossing the sensor FOV of a single robot, while trajectories T4 and T6 are generated by two pedestrians simultaneously crossing the overlapping sensor FOVs of both vehicles. It is clear that the tracking performance is improved when both vehicles observe the same feature(s) at the same time (T4 and T6). Apart from the obvious dynamic features, few more dynamic feature observations is present due to the observation of people randomly walking around during the experiment.

VII. CONCLUSION

In this paper we have presented a new solution to the Collaborative Multi-vehicle SLAM with moving object tracking (CMSLAMMOT) problem by extending our recently developed random finite set (RFS) based CMSLAM solution. We have modeled measurements and features as RFSs and formulated a Bayesian solution where both static and dynamic features are tracked using measurements with clutter and come up with a robust solution to the CMSLAMMOT problem. The proposed solution inherits inbuilt map management, data association and clutter filtering, while catering for feature detection and survival uncertainties. Performance of the proposed algorithm is demonstrated via simulation and experiment results. Future research will focus on applying these results on various dynamic environments with a larger number of moving objects.

REFERENCES

[1] R. Smith, M. Self and P. Cheeseman, *Estimating uncertain spatial relationships in robotics*. NewYork: Springer: Autonomous Robot Vehicles, 1990.

[2] 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.

[3] 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.

[4] 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.

[5] 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.

[6] 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.

[7] C.C. Wang and C. Thorpe, "Online Simultaneous Localization And Mapping with Detection And Tracking of Moving Objects: Theory and Results from a Ground Vehicle in Crowded Urban Areas," in *Proc. IEEE Int. Conf. Robot. Autom.*, September 2003, pp. 842–849.

[8] C.C. Wang, C. Thorpe and S. Thrun, "Online Simultaneous Localization And Mapping with Detection And Tracking of Moving Objects," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2002, pp. 2918–2924.

[9] S.A. PASHA, B.N. Vo, H.D. Tuan and W.K. Ma, "A Gaussian Mixture PHD filter for Jump Markov System Models," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 45, no. 3, pp. 919–936, July 2009.

[10] D.P. Moratuwage, B.N. Vo, D. Wang and H. Wang, "Extending Bayesian RFS SLAM to Multi-Vehicle SLAM," in *Proc. IEEE Int. Conf. Control, Autom., Robot. Comp. Vision. (accepted)*.

[11] R. Mahler, *Statistical Multisource-Multitarget Information Fusion*. Norwood, MA: Artech House, 2007.

[12] D. Stoyan, W.S. Kendall, and J. Mecke, *Stochastic Geometry and its Applications*. : John Wiley and Sons, Inc., 1995.

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

[14] ——, "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.