A Random Set Formulation for Bayesian SLAM

John Mullane, Ba-Ngu Vo[†], Martin D. Adams, Wijerupage Sardha Wijesoma

School of Electrical & Electronic Engineering, Nanyang Technological University, Singapore
[†]*School of Electrical & Electronic Engineering, University of Melbourne, Victoria, 3010, Australia*
{john0002,eadams,eswwijesoma}@ntu.edu.sg
[†]*bv@ee.unimelb.edu.au*

***Abstract*— This paper presents an alternative formulation for the Bayesian feature-based simultaneous localisation and mapping (SLAM) problem, using a random finite set approach. For a feature based map, SLAM requires the joint estimation of the vehicle location and the map. The map itself involves the joint estimation of both the number of features and their states (typically in a 2D Euclidean space), as an *a priori* unknown map is completely unknown in both landmark location *and* number. In most feature based SLAM algorithms, so-called 'feature management' algorithms as well as data association hypotheses along with extended Kalman filters are used to generate the joint posterior estimate. This paper, however, presents a recursive filtering algorithm which jointly propagates both the estimate of the number of landmarks, their corresponding states, and the vehicle pose state, without the need for explicit feature management and data association algorithms.**

Using a finite set-valued joint vehicle-map state and set-valued measurements, the first order statistic of the set, called the *intensity*, is propagated via the probability hypothesis density (PHD) filter, from which estimates of the map and vehicle can be jointly extracted. Assuming a mildly non-linear Gaussian system, an extended-Kalman Gaussian Mixture implementation of the recursion is then tested for both feature-based robotic mapping (known location) and SLAM. Results from the experiments show promising performance for the proposed SLAM framework, especially in environments of high spurious measurements.

I. Introduction

The feature-based (FB) SLAM scenario is a vehicle moving through an environment represented by an unknown number of features. The classical problem definition is one of "*a state estimation problem involving a variable number of dimensions*" [1]. The vehicle is assumed to acquire absolute measurements of its surrounding environment using on board range-bearing measuring sensors (for the planar case). The SLAM problem requires a robot to navigate in an unknown environment and use its suite of onboard sensors to both construct a map and localize itself within that map without the use of any *a priori* information. This requires joint estimates of the three dimensional robot pose, the number of features in the map as well as their two dimensional Euclidean co-ordinates. For a real world application, this should be performed incrementally as the robot maneuvers about the environment. As the robot motion introduces error, coupled with a feature sensing error, both localistion and mapping must be performed simultaneously [2]. For any given sensor, an FB measurement is subject to detection and data association uncertainty, spurious measurements and measurement noise, as well as biasses.

To encapsulate the inherent measurement and vehicle uncertainty in a SLAM system, Bayesian filtering solutions have become extremely popular in recent years [3]. The majority of proposed algorithms, stemming from the seminal work [2], adopt an augmented state containing both the vehicle pose estimate and the estimate of the map. Kalman based solutions are then applied to jointly estimate the vehicle pose and map, whilst maintaining the state correlations. Under Gaussian noise assumptions, the Kalman filter produces the *optimal minimum-variance Bayesian estimate* of the joint-vehicle map state. The example discussed in [2] had a map containing features of unity detection probability, assumed the measurement-feature association was known, and that the sensor reported no spurious measurements. With these strict assumptions, the Kalman based SLAM estimate is indeed Bayes optimal. This work was incorporated into multiple Kalman-based solutions to the FB-SLAM problem [3].

Sequential Monte Carlo (SMC) solutions to Bayesian SLAM also gained popularity [1] through the use of Rao-Blackwellised particle filters. FastSLAM [4] displayed impressive results by sampling over the vehicle trajectory and applying independent Kalman filters to estimate the location of the hypothesised map features. By conditioning the map estimates on the history of vehicle poses, a conditional measurement independence is invoked which allows the correlations introduced in [2] to be discarded. A Gaussian mixture solution to the Bayesian SLAM problem was also described in [5] which approximated both the transition and measurement densities as Gaussian mixtures and propagated the joint state through a Bayes recursion.

Current existing solutions to the FB-SLAM problem can be regarded as being *vector-based* algorithms. That is, both the joint vehicle-map state and the measurements are modeled as vectors. Since the order of elements in a vector is fixed, such a representation implies that the order of measurements received at any given time, are from the same order of features in the map state. That is, there is assumed to be no data association uncertainty. While vector state representation has never before been highlighted as the source of this oversight, the SLAM community has long acknowledged this drawback, and numerous solutions have been proposed to solve the measurement-feature association problem [6], [7], [8]. The presence of detection uncertainty and spurious measurements have also been long acknowledged, and subsequently feature initialisation and termination algorithms are frequently incorporated into the vector-based SLAM algorithm [3]. Again, the paper emphasises for the first time that these are required due to the inability of a vector representation to incorporate uncertainty in the *number* of dimensions, and highlights that such methods (which are independent of the filter recursion) compromise filter performance. As shown in this paper, this can result in filter divergence and large mapping error, especially in scenarios of high clutter and large data association ambiguity. Through the re-formulation of the classical FB-SLAM problem, and by explicitly incorporating the problem of a variable number of dimensions into the filter recursion, increased robustness in noisy scenarios is possible.

The latest emerging multi-target tracking algorithms [9] [10] represent the states and measurement as finite sets. Random finite sets (RFSs) are then used to model uncertainty in both the number of states/measurements as well as their individual values. A finite set-valued measurement, for example, allows for the inclusion of spurious measurements directly into the measurement equation which is then the union of the set-state dependant measurements (as is the case in the classical Bayesian SLAM formulations) and the set of spurious measurements. A finite set-valued map state can be made up of the set union of the existing features and the new features which may appear in the map due to the motion of the robot. In [9], the mathematics are established for a finite set-valued Bayesian filtering formulation of the multi-target tracking problem, where the number of targets and their corresponding states can then be jointly estimated.

This paper subsequently casts the FB-SLAM problem into a random set theoretic filtering problem that incorporates the joint estimation of the vehicle pose, feature number and corresponding feature locations. The paper is organized as follows: Section II firstly outlines the general vector-based Bayesian SLAM formulation. It then highlights the short comings of the classical method of modeling the feature measurements in section II. The RFS-SLAM formulation is then introduced in section III. Section IV introduces an augmented joint vehicle-map RFS to incorporate vehicle location uncertainty. A first-order approximation (the *probability hypothesis density*, or PHD) of the augmented state recursion is then presented, and the PHD-SLAM filter is introduced. Using Gaussian noise assumptions, an extended-Kalman Gaussian Mixture implementation is developed in section V. This implementation accounts for the non-linearity in the measurement equation and jointly estimates the feature number in the map, their corresponding states and the vehicle pose. Furthermore, this can be achieved without the need for explicit data association decisions and/or feature management algorithms. Simulated mapping and pose estimation results are shown in section VI where the proposed GM-PHD SLAM filter is tested on simulated data with high spurious measurements. Results show the efficacy of the proposed framework for solving the Bayesian SLAM problem.

II. General Formulation of the Bayesian FB-SLAM Problem

The aim of FB-SLAM is to jointly estimate the $\mathfrak{q}$ map feature locations, $m_1, \ldots, m_{\mathfrak{q}}$ as well as the vehicle trajectory $x^k = [x_0, \ldots, x_k]$, given the history of vehicle control inputs, $u^{k-1} = [u_0, \ldots, u_{k-1}]$, the history of sensor measurements $Z^k = [Z_0, \ldots, Z_k]$ and the initial vehicle state x_0. In this work, the measurement, $Z_k = \{z_k^1, \ldots, z_k^{\mathfrak{z}(k)}\}$, consists of $\mathfrak{z}(k)$ range-bearing measurements registered by an onboard exteroceptive ranging sensor. The map $M = \{m_1, \ldots, m_{\mathfrak{q}}\}$ is *a priori* unknown and consists of features at unknown Euclidean co-ordinates, $m_1, \ldots, m_{\mathfrak{q}}$. Furthermore, the total number of features in the map, $\mathfrak{q}$, is also *a priori* unknown, and estimating M involves jointly estimating $\mathfrak{q}$ as well as their corresponding states. Therefore at each time step k, the joint posterior probability density (pdf) of the map and vehicle trajectory can be written (assuming that such a density exists) as,

$$p_k(x^k, M|Z^k, u^{k-1}, x_0). \quad (1)$$

From an optimal Bayesian perspective, the posterior probability density should capture all relevant statistical information about the vehicle state and the map. The posterior density can, in theory, be propagated in time via the Bayes recursion:

$$p(x^k, M|Z^k, u^{k-1}, x_0) = \frac{g(Z_k|M, x_k)p(x^k, M|Z^{k-1}, u^{k-1}, x_0)}{\int\int g(Z_k|M, x_k)p(x^k, M|Z^{k-1}, u^{k-1}, x_0)dx_k dM} \quad (2)$$

where,

$$p(x_k, M|Z^{k-1}, u^{k-1}, x_0) = \int f(x_k|x_{k-1}, u_{k-1})p(x^{k-1}, M|Z^{k-1}, u^{k-2}, x_0)dx_{k-1}.$$

The motion of the vehicle is modeled as a first order Markov process with transition density $f(x_k|x_{k-1}, u_{k-1})$. This formulation can be easily extended to dynamic maps and multiple sensors. For clarity of exposition, the static-map-single-sensor case is adhered to.

In the general Bayesian formulation of SLAM [3], eqn. (1) is the joint probability density of a random vector containing the map states, $m_1, \ldots, m_{\mathfrak{q}}$, concatenated with the vehicle state, x^k. In a practical scenario at each time instance, with a limited sensor field of view (FOV), the map $M_k = [m_1, \cdots, m_q]$ comprises the states of the q features assumed to exist in the map (which have passed through the FOV), where $q \leq \mathfrak{q}$. At each time, estimates of the joint state can be obtained using MAP or MMSE estimation criterion [2]. As the measurement is also a vector of individual range-bearing readings, a direct implementation of this Bayesian recursion in its classical vector form, implicitly assumes that the order of the measurements equals the order of the features in the map vector. This is not a new observation and has resulted in the development of numerous data association algorithms to solve the measurement-feature correspondence problem [7], [6], [8]. These are usually 'inserted' in to the Bayesian recursion before performing the update of eqn.(2). Furthermore, to estimate the number of observed features, q, so called 'feature-management' algorithms are used which maybe incorporate the data association decisions [6] directly, or adopt discrete Bayesian filtering methods from occupancy grid mapping algorithms [4]. These SLAM approaches generally either assume a Gaussian system which leads to Kalman based solutions or employ Sequential Monte Carlo methods which avoid the restrictive uni-modal Gaussian approximations and potential linearisation errors.

From a Bayesian perspective however, it is argued in this work, that the vector-based density of (1) lacks Bayes optimality for the SLAM problem, as some aspects of the system uncertainty are overlooked. From an intuitive perspective, this is evident from the need to include measurement-feature hypothesis decisions and feature validity checks to estimate the number of features, q. Independent (from the filter recursion) routines for dealing with spurious measurements and data association uncertainty, whilst showing promising results in practise, compromise the Bayes optimality of the posterior joint density. From examination of the measurement model in the following section, further theoretical issues in the vector-based formulation become evident.

The Random Vector Measurement Model

In SLAM, the map state at time k, M_k is assumed to consist of q features whose states, $m_1, \ldots, m_q$, are to be estimated. A prediction of the vehicle pose, x_k, is also available from the transition density $f(\cdot)$ given the control input, u_{k-1}. The measurement is then modeled as a vector,

$$Z_k = \boldsymbol{h}(x_k, m_1, \ldots, m_q) + \boldsymbol{w}_k. \tag{3}$$

where $\boldsymbol{h}(\cdot)$ is generally a non-linear function mapping the feature and vehicle locations into the relative range and bearing measurement, and $\boldsymbol{w}$ is a Gaussian distributed random vector, which encapsulates the additive measurement noise. The measurement likelihood, $g(Z_k|M_k, x_k)$ is treated as the likelihood of receiving a vector that contains q measurements, one from each of the q features $m_1, \ldots, m_q$. Examination of the measurement model and a vector-based state, reveals an implicit assumption that the number of features present in M_k (but not necessarily their location) is known *a priori*. Moreover, it is assumed that each feature generates an observation, and the order of the measurements is the same as the order of the features in M_k (i.e. z_k^1 is from the feature at m_1 etc.). Also the model of eqn.(3) has no inclusion of spurious measurement and only accounts for measurements from each of the features at locations, $\{m_1, \ldots, m_q\}$, corrupted by Gaussian noise.

Therefore, the standard measurement model used in numerous Bayesian SLAM solutions, overlooks detection uncertainty, spurious measurements, and measurements from newly observed features. While the existence of such sensor uncertainty has long been acknowledged by the SLAM community [3], [4], this paper for the first time explicitly highlights the inabilities of the classical measurement equation to model such uncertainties. By using finite set-valued measurements, these uncertainties can be explicitly accounted for in the measurement model and consequently encapsulate the detection and data association sensing uncertainty directly into the resulting filter recursion. This is not the case of filters which use the vector-based measurement of eqn.(3). The following section therefore outlines a set-valued Bayesian SLAM formulation, which adopts finite set valued measurements.

III. RFS Formulation of the Bayesian SLAM Problem

To incorporate the fact that new features enter the map, M_k, with time, let the map state M_k be an RFS which evolves in time according to,

$$M_k = M_{k|k-1} \cup B_k \tag{4}$$

comprising the set union of the RFS multi-feature transition prediction of the previous map RFS, $M_{k|k-1}$ and the RFS of the new features at time k, B_k. These sets are assumed mutually independent. Note that vector-based SLAM algorithms do not include the possibility of new features coming within the sensors field of view in the state transition equation. As static features are assumed, the map set propagates in time via,

$$M_{k|k-1} = M_{k-1}.$$

To contend with the realistic situation of missed detections and clutter, the measurement is also modeled as an RFS. Given the predicted vehicle state, x_k, and the map M_k, the measurement consists of a set union,

$$Z_k = \bigcup_{m \in M_k} \Theta(m, x_k) \cup C_k(x_k) \tag{5}$$

where $\Theta_k(m, x_k)$ is the RFS of the measurement generated by a feature at m and $C_k(x_k)$ is the RFS of the spurious measurements at time k. Therefore Z_k consists of a random number of range-bearing measurement in $\mathbb{R}^2$ where the number of detected measurement may differ from the number of features due to potential missed detections. Eqn. (5) thus encapsulates measurement noise, detection uncertainty and spurious measurements, compared to the vector-based model of eqn (3) which only considers measurement noise. It is also assumed that $\Theta_k(m, x_k)$, and $C_k(x_k)$ are independent RFSs.

For each feature, $m \in M$, and $z \in Z_k$,

$$\Theta(m, x_k) = \{z\} \tag{6}$$

with probability density $p_D(m, x_k) g(z|m, x_k)$ and $\Theta_k(x_k, m) = \emptyset$ with probability $1 - p_D(m, x_k)$, where $p_D(m_q, x_k)$ is the probability of the sensor detecting the q^{th} feature. $p_D(m, x_k)$ is the vehicle state dependant detection probability which is a function of the finite sensor FOV. Using Finite Set Statistics [11], the probability density that the sensor produces the measurement set Z_k given the vehicle state x_k and map M_k at time k is then given by [9]:

$$g(Z_k|M_k, x_k) = \sum_{W \subseteq Z_k} \theta_k(W|M_k, x_k) c_k(Z_k - W) \tag{7}$$

with $\theta_k(\cdot|M_k, x_k)$ denoting the density of the RFS of observations, $\Theta(m, x_k)$, generated from the features in the observed map M_k given the state of the vehicle, and $c_k(\cdot)$ denoting the density of the clutter RFS, C_k. Note that the difference operation used in (7) is the set difference. The density of a random finite set requires more general mathematical constructs than that used for vectors [12]. $\theta_k(\cdot|M_k, x_k)$ describes the likelihood of receiving a measurement from the elements of the set-valued map which incorporates detection uncertainty and measurement noises. $c_k(\cdot)$ models the spurious measurement rate of the sensor and is typically *a priori* assigned [6] [7]. Expanding the multi-target RFS Bayes recursion of [11] to include the vehicle state, the optimal Bayesian SLAM filter then jointly propagates the set of features and the vehicle location according to,

$$p(x^k, M|Z^k, u^{k-1}, x_0) = \frac{g(Z_k|M, x_k)p(x^k, M|Z^{k-1}, u^{k-1}, x_0)}{\int\int g(Z_k|M, x_k)p(x^k, M|Z^{k-1}, u^{k-1}, x_0)dx_k\mu(dM)} \tag{8}$$

where,

$$p(x_k, M|Z^{k-1}, u^{k-1}, x_0) = \int f(x_k|x_{k-1}, u_{k-1})p(x^{k-1}, M|Z^{k-1}, u^{k-2}, x_0)dx_{k-1}$$

and μ is a reference measure on the space of features. Note that, contrary to previous SLAM formulations, M and Z are now modeled by RFS's. As in a direct implementation of the vector-based Bayesian SLAM recursion of eqn.(2), computational complexities and multiple integrals generally lead to intractable solutions. Fortunately, approximations have been developed in the tracking literature [9], which can be incorporated into this RFS SLAM formulation.

IV. Probability Hypothesis Density (PHD) SLAM Filter

Instead of propagating the posterior density, the PHD filter propagates only its first order statistical moment [9], known as its *intensity*. This corresponds to the 'expectation' of an RFS [12]. For an RFS M_k, with probability distribution P, the intensity is a non-negative function v, such that for each region S in the space of features,

$$\int |M_k \cap S| \, P(dM_k) = \int_S v(m)dm. \tag{9}$$

Since, $|M_k \cap S| = \sum_{x\in X} \mathbf{1}_S(x)$, is the number of features, the integral of the intensity v over any region S gives the expected number of elements of M_k that are in S. Simply setting S to be the entire mapped region an estimate of the number of features in the observed map set, M_k, can be jointly estimated along with their locations. The (coordinates of the) peaks of the intensity are points (in the space of features) with the highest local concentration of expected number of features and hence can be used to generate estimates for the elements of M_k. The integral of the PHD gives the expected number of features and the peaks of the PHD function can be used as estimates of their locations.

Since the intensity is the first statistic of a random finite set, the PHD filter is analogous to the constant gain Kalman filter which propagates the first order statistic (the mean) of the vector-based state. However, the *intensity*, $v_k(m)$, contains information on both the number of features in the map set, and their corresponding states (along with the uncertainty in their state estimation). Under the assumption of independent and Poisson distributed RFS's, a recursion for the intensity was derived in [9]. This tracking-orientated approach, considered only the intensity of the target (feature) RFS whereas SLAM requires the joint estimation of both the set of features and the vehicle state.

Expanding on [9], let ζ_k denote the a feature state, m, concatenated with one of N hypothesised vehicle trajectories, $x_k^{(n)}$[1]. Conditioning each feature state, m, on the history of vehicle poses introduces a conditional independence between feature measurements allowing the joint states, ζ_k to be independently propagated through the PHD SLAM framework [4]. Each augmented feature evolves in time according to the transition $f(\zeta_k|\zeta_{k-1}, u_{k-1})$ and, if detected by the sensor, generates a measurement z with likelihood $p_D(\zeta_k)g(z|\zeta_k)$. Let the vehicle state be sampled by N particles, to produce $N\times|M_k|$ augmented states, ζ_k.

If $\mathbb{L}$ denotes the space of features and $\mathbb{K}$ denotes the space of vehicle states, Campbell's theorem [13] implies that the intensity of the point process on $\mathbb{L}\times\mathbb{K}$ formed by the cartesian product of a point process on $\mathbb{L}$, with intensity $\tilde{v}$, and a point process on the mark space (a vehicle pose particle) $\mathbb{K}$,

$$v(x_k, m) = p(x_k|m)\tilde{v}(m), \tag{10}$$

where $p(\cdot|m)$ is the mark distribution given a point m of the original point process on $\mathbb{L}$. Moreover, if the point process on $\mathbb{L}$ (the set of features) is Poisson, then the product point process on $\mathbb{L}\times\mathbb{K}$ is also Poisson [13]. As the RFS of the joint vehicle and map state is therefore Poisson, the derivation established in [9], can be incorporated in this work to include the joint vehicle-feature augmented state. Given a set of augmented features, ζ_k, joint estimates of the number of features, their locations as well as the vehicle state can then be obtained. The PHD-SLAM recursion is therefore,

$$\begin{aligned} v_{k|k-1}(\zeta_k) &= \int f(\zeta_k|\zeta_{k-1}, u_{k-1})v_{k-1}(\zeta_{k-1})d\zeta_{k-1} + b_k \\ &= \int f(\zeta_k|x_{k-1}, m_k, u_{k-1})v_{k-1}(x_{k-1}, m_k)dx_{k-1} + b_k \end{aligned}$$

$$v_k(\zeta_k) = \Big[1 - p_D(\zeta_k) + \sum_{z\in Z_k} \frac{\Lambda(z|\zeta_k)}{c_k(z) + \int \Lambda(z|\xi)v_{k|k-1}(\xi)d\xi}\Big]v_{k|k-1}(\zeta_k) \tag{11}$$

[1]Note that for notation clarity, here $x_k = x^k$.

where at time k,

b_k	$=$	intensity of the new feature RFS B_k,
$\Lambda(z\|\zeta_k)$	$=$	$p_D(\zeta_k)g(z\|\zeta_k)$,
$g(z\|\zeta_k)$	$=$	likelihood of z, given the joint state ζ_k,
$p_D(\zeta_k)$	$=$	probability of detection of the feature in ζ_k ,given the pose in ζ_k,
c_k	$=$	intensity of the clutter RFS C_k.

In [10], Gaussian noise assumptions were used to obtain closed form solutions for the target tracking PHD filter. Similarly for the PHD-SLAM filter, Gaussian mixture (GM) techniques can be applied to solve the PHD-SLAM joint intensity recursion of eqn.(11). It is also possible to use a particle-based approach [14], however, for mildly non-linear problems the Gaussian mixture approach is much more efficient. The following section thus presents a GM implementation of the PHD-SLAM filter.

V. Gaussian Mixture (GM) PHD-SLAM

Let the joint intensity, $v_{k-1}(\zeta_{k-1})$, at time $k-1$ be a Gaussian mixture of the form,

$$v_{k-1}(\zeta_{k-1}) = \sum_{i=1}^{N\times J_{k-1}} w_{k-1}^{(i)} \mathcal{N}\big(\zeta;\mu_{k-1}^{(i)},P_{k-1}^{(i)}\big) \tag{12}$$

composed of $N{\times}J_{k-1}$ Gaussians, with $w_{k-1}^{(i)}$, $\mu_{k-1}^{(i)}$ and $P_{k-1}^{(i)}$ being their corresponding weights, means and covariances respectively. Note that the weight, $w_{k-1}^{(i)}$ is a weight on *both* a particular feature state, m, and a particular vehicle pose $x_{k-1}^{(n)}$, i.e. on the joint state $\zeta_{k-1}^{(i)}$.

Since the map is assumed static, the joint state transition density is, $f(x_k^{(n)}|x_{k-1}^{(n)},u_{k-1})\delta(m_k-m_{k-1})$ where $x_{k-1}^{(n)}$ is one of N vehicle pose particles at time $k-1$. Let the new feature intensity at time k also be a Gaussian mixture,

$$b_k = \sum_{i=1}^{N\times J_{b,k}} w_{b,k}^{(i)} \mathcal{N}\big(\zeta;\mu_{b,k}^{(i)},P_{b,k}^{(i)}\big) \tag{13}$$

where $w_{b,k}^{(i)}$, $\mu_{b,k}^{(i)}$ and $P_{b,k}^{(i)}$ determine the shape of the new feature GM proposal density according to a chosen strategy. This is analogous to the proposal distribution in the particle filter [4] and provides an initial estimate of the new features entering the map (see section V-A). Again, each new feature density component, $\mathcal{N}(m;\cdot)$ is concatenated with each predicted vehicle pose particle, $x_k^{(n)}$ to form the $N{\times}J_{b,k}$ components of the GM new feature intensity. Therefore, the predicted intensity, $v_{k|k-1}(\zeta_k)$ is also a Gaussian mixture,

$$v_{k|k-1}(\zeta_k) = \sum_{i=1}^{J_{k|k-1}} w_{k|k-1}^{(i)} \mathcal{N}\big(\zeta;\mu_{k|k-1}^{(i)},P_{k|k-1}^{(i)}\big) \tag{14}$$

where, $J_{k|k-1} = N(J_{b,k}+J_{k-1})$ and,

$$\left.\begin{array}{l} w_{k|k-1}^{(i)} = w_{k-1}^{(i)} \\ \mu_{k|k-1}^{(i)} = \zeta_{k|k-1}^{(i)} \\ P_{k|k-1}^{(i)} = P_{k-1}^{(i)} \end{array}\right\}\text{for i} \in \{1,\ldots,N{\times}J_{k-1}\}$$

$$\left.\begin{array}{l} w_{k|k-1}^{(i)} = w_{b,k}^{(i)} \\ \mu_{k|k-1}^{(i)} = \mu_{b,k}^{(i)} \\ P_{k|k-1}^{(i)} = P_{b,k}^{(i)} \end{array}\right\}\text{for i} \in \{N{\times}J_{k-1}+1,\ldots,N{\times}J_{b,k}\}.$$

Assuming a Gaussian measurement likelihood, $g(z|\zeta_k)$, it can be seen from eqn.(11), that the joint posterior intensity, $v_k(\zeta_k)$, is consequently a Gaussian mixture,

$$v_k(\zeta_k) = v_{k|k-1}(\zeta_k)\Bigg[1 - P_D(\zeta_k) + \sum_{z\in Z_k}\sum_{i=1}^{J_{k|k-1}} v_{G,k}^{(i)}(z|\zeta_k)\Bigg] \tag{15}$$

where,

$$v_{G,k}^{(i)}(z|\zeta_k) = w_k^{(i)}\mathcal{N}(\zeta;\mu_{k|k}^{(i)},P_{k|k}^{(i)}) \tag{16}$$

$$w_k^{(i)} = \frac{P_D(\zeta_k)w_{k|k-1}^{(i)}q^{(i)}(z,\zeta_k)}{c_k(z) + \sum_{j=1}^{J_{k|k-1}} P_D(\zeta_k)w_{k|k-1}^{(j)}q^{(j)}(z,\zeta_k)}$$

with, $q^{(i)}(z,\zeta_k) = \mathcal{N}\big(z;H_k\mu_{k|k-1}^{(i)},S_k^{(i)}\big)$. The components are obtained from the standard EKF update equations,

$$\mu_{k|k}^{(i)} = \mu_{k|k-1}^{(i)} + K_k^{(i)}(z - H_k\mu_{k|k-1}^{(i)}) \tag{17}$$

$$P_{k|k}^{(i)} = [I - K_k^{(i)}\nabla H_k]P_{k|k-1}^{(i)} \tag{18}$$

$$K_k^{(i)} = P_{k|k-1}^{(i)}\nabla H_k^T[S_k^{(i)}]^{-1} \tag{19}$$

$$S_k^{(i)} = R_k + \nabla H_k P_{k|k-1}^{(i)}\nabla H_k^T \tag{20}$$

with ∇H_k being the Jacobian of the measurement equation with respect to the features estimated location. As stated previously, the clutter RFS, C_k, is assumed Poisson distributed [6], [7] in number and uniformly spaced over the sensor surveillance region. Therefore the clutter intensity is,

$$c_k(z) = \lambda_c V\mathcal{U}(z) \tag{21}$$

where λ_c is the average number of clutter returns, V is the volume of the sensor's surveillance region and $\mathcal{U}(\cdot)$ denotes a uniform distribution over range and bearing. Gaussian pruning and merging methods are used as in [10].

A. The New feature Proposal Strategy

The new feature proposal density, eqn.(13), is similar to the proposal function used in particle filters, and is used to give some *a priori* information to the filter about where features are likely to appear in the map. In SLAM, with no *a priori* information, b_k, may be uniformly distributed in a non-informative manner about the space of features (analogous to the prior map used in occupancy grid algorithms). However, in this work the feature birth proposal at time k is chosen to be the set of measurements at time $k-1$, Z_{k-1}. The sum $\sum_{i=1}^{N \times J_{b,k}} w_{b,k}^{(i)}$ then gives an estimate of the expected number of new features to appear at time k.

VI. Algorithm Performance

This section analyses the performance of the proposed GM-PHD SLAM filter in a simulated environment, and compares it to a FastSLAM implementation using maximum likelihood data association decisions and Log-Odds feature management [4]. The vehicle is assumed to be traveling at $3ms^{-1}$ while subject to velocity and steering input noises of $1ms^{-1}$ and 5^o respectively. Only 10 particle samples are used for both filters and both filters receive the same noisy input samples and sensor measurements. Two simulated comparisons are performed in an 'easy' and 'difficult' scenario. For the 'easy' scenario, the clutter parameter, $\lambda_c = 0$, feature detection probability is 0.95, and the measurement noises are $0.25m$ in range and 0.5^o in bearing. For the 'difficult' scenario, $\lambda_c = 10$, feature detection probability is again 0.95 and the measurements noises are set at $12.5m$ in range and 25^o in bearing. The effect of the artificially large measurement noises are to give the appearance of closely spaced features, hampering data association decisions and feature map building.

Figure 1 shows the estimated vehicle trajectory and corresponding feature map from both filters. Both results compare well with ground truth (green). This result verifies the accuracy of the proposed PHD-SLAM filter, in its ability to jointly estimate the vehicle trajectory, the number of features, and their corresponding location, without the need for external data association and feature map management methods, as are required by FastSLAM (and other vector-based solutions).

The missed feature declaration highlights an issue of the proposed method with respect to $p_D(\zeta_k)$. In the presented implementation, this is simply a binary function which has an assumed value of 0.95 if the feature is predicted to be within the sensor field of view, and 0 if it is not. Vehicle and feature estimation uncertainty may result in a feature erroneously being hypothesised of being within the field of view, or vice-versa. If the proposed filter then receives a measurement contrary to the prediction, the resulting feature weight may be detrimentally reduced, and a missed feature declaration may occur. The uncertainty in the estimated sensor field of view is not considered in this implementation.

The raw measurements as well as the final posterior joint estimate of both filters for the 'hard' scenario are presented respectively in figures 2 and 3. As is clearly evident, the proposed filter displays dramatically reduced feature-based mapping error in the face of large data association uncertainty and large quantities of spurious measurements, reporting only a single false feature and a single missed feature over the entire run. This is expected, as the clutter rate is integrated directly into the filter recursion in an optimal manner and feature management is performed jointly with feature and vehicle location estimation. Figure 4 shows the estimated number of features in the map over time, for both the discussed filters, as well as the number of false measurements at each time instant. The proposed filter accurately tracks the true number of features over time, whereas the FastSLAM filter deviates drastically in the face of the challenging spurious measurements and data association ambiguities. Estimating the true number of state dimensions influences the accuracy of the overall FB-SLAM filter. The estimated vehicle trajectory also displays less error than that of the FastSLAM approach. Similarilty to FastSLAM, increased trajectory estimation accuracy may be possible by increasing the number of pose samples.

Figure 5 compares the estimated vehicle heading over the course of the test, highlighting the increased accuracy of the proposed filter. As is evident from the update of eqn.(15), the proposed algorithm scales linearly with $\mathcal{O}(NJ_{k|k-1}\mathfrak{z}(k))$, which equals that of a naive FastSLAM implementation. Future work will address reducing this to a Log order complexity of the number predicted map states $J_{k|k-1}$. The presented results illustrate the effectiveness of the new finite-set based SLAM framework, and the proposed GM-PHD implementation, when compared to vector-based solutions which fail to jointly conside the entire system uncertainty.

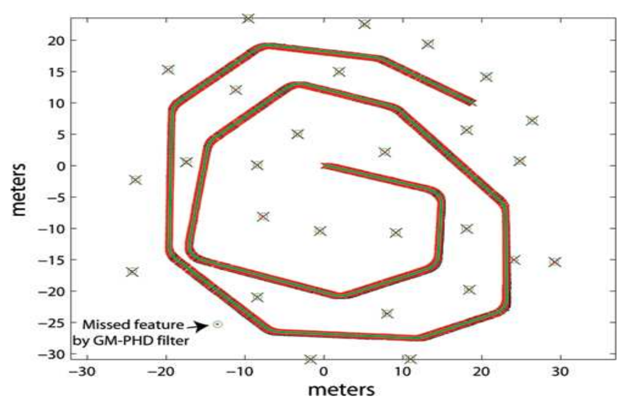

Fig. 1. *Comparative results for the proposed GM-PHD SLAM filter (black) and that of FastSLAM (red), compared to ground truth (green).*

VII. Conclusion

This paper outlined an alternative formulation to the Bayesian SLAM problem, using random set theory. The set theoretic approach allows for detection uncertainty, spurious measurements as well as data association uncertainty to be incorporated directly into the filter recursion. This is in contrast to vector-based SLAM which requires additional algorithms and pre/post processing to solve the data

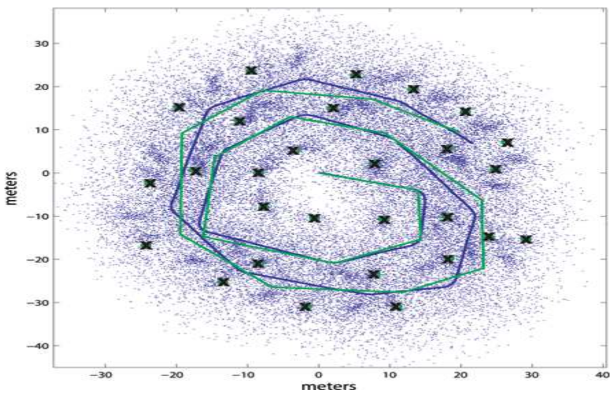

Fig. 2. *The predicted vehicle trajectory (blue) along with the raw sensor measurements for the 'hard' scenario, at a clutter density of* $0.03m^{-2}$. *Also superimposed are the ground truth trajectory and feature map (black crosses).*

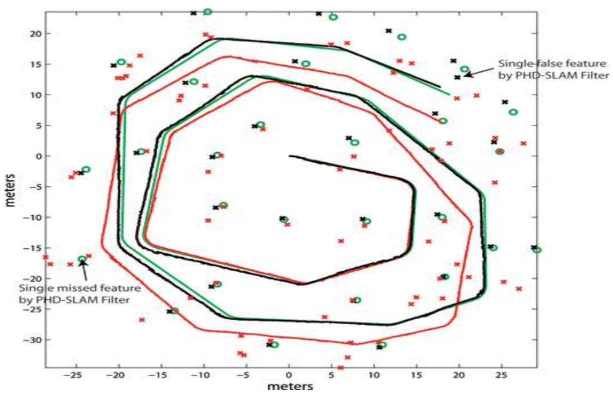

Fig. 3. *The estimated trajectories of the GM-PHD SLAM filter (black) and that of FastSLAM (red). Estimated feature locations (crosses) are also shown with the true features (green circles)*

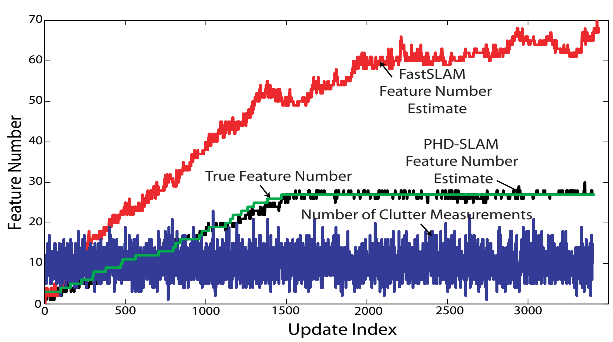

Fig. 4. *The error in vehicle heading estimate for the proposed (black) and FastSLAM (blue) filters.*

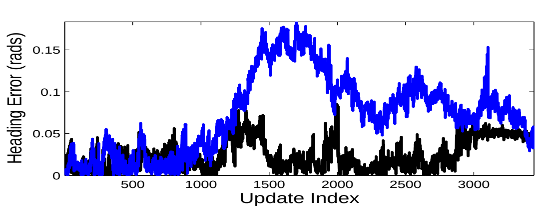

Fig. 5. *The error in vehicle heading estimate for the proposed (black) and FastSLAM (red) filters.*

association problem prior to filter update, and to extract estimates of the number of features present in the map. These are necessary as such sources of uncertainty are not considered in the classical vector-based measurement model and subsequent filter recursion. Previous Bayesian SLAM solutions also lack a concept of Bayesian optimality as the variable dimensionality problem is not jointly considered.

Propagating the first order statistic of the random set (the probability hypothesis density) is a common method of reducing the computational requirements of implementing the set-valued Bayesian recursion. By augmenting the feature state with a history of vehicle poses, conditional independencies between the features and the vehicle state are introduced. The joint vehicle feature RFS was shown to maintain the necessary Poisson assumptions for application of the tracking based PHD recursion for the PHD-SLAM problem. A Gaussian mixture implementation of the PHD-SLAM filter was outlined assuming a Gaussian system with non-linear measurement and process models. The proposed finite-set filter was compared to a FastSLAM implementation with explicit (per particle) data association decisions and feature management methods. Results show the proposed filter performing similarly to FastSLAM in an 'easy' scenario, and considerably outperforming it in a 'hard' scenario. This work validates the alternative SLAM formulation proposed here and motivates further research.

REFERENCES

[1] S. Thrun. Particle filters in robotics. In *17th Annual Conference on Uncertainty in AI*, 2002.

[2] R. Smith, M. Self, and P. Cheeseman. A stochastic map for uncertain spatial relationships. In *The fourth international symposium of Robotics Research*, pages 467–474, 1987.

[3] H.F. Durrant-Whyte and T. Bailey. Simultaneous localization and mapping: Part I. *IEE Robotics and Automation Magazine*, 13(2):99–110, June 2006.

[4] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit. Fastslam 2.0: an improved particle filtering algorithm for simultaneous localization and mapping that provably converges. In *18th Int. Joint Conf. on Artificial Intelligence*, pages 1151–1156. Morgan-Kaufmann Publishers, 2003.

[5] S. Majumder. *Sensor Fusion and Feature Based Navigation for Subsea Robots*. PhD thesis, The University of Sydney, August 2001.

[6] W.S. Wijesoma, L.D.L Perera, and M.D. Adams. Toward multidimensional assignment data association in robot localization and mapping. *IEEE Transactions on Robotics*, 22(2):350–365, April 2006.

[7] D. Makarsov and H.F. Durrant-Whyte. Mobile vehicle navigation in unknown environments: a multiple hypothesis approach. *IEE Proceedings of Contr. Theory Applict.*, vol. 142, July 1995.

[8] J. Niera and J.D. Tardos. Data association in stochastic mapping using the jointcompatibility test. *IEEE Transactions on Robotics and Automation*, 17(6):890–897, December 2001.

[9] R. Mahler. Multi-target bayes filtering via first-order multi-target moments. *IEEE Transactions on AES*, 4(39):1152–1178, October 2003.

[10] B.N. Vo and W.K. Ma. The gaussian mixture probability hypothesis density filter. *IEEE Transactions on Signal Processing*, 54(11):4091–4104, November 2006.

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

[12] D. Stoyan, W.S Kendall, and J. Mecke. *Stochastic Geometry and Its Applications*. John Wiley and Sons, New York, 2nd edition, 1995.

[13] J.F.C Kingman. *Poisson Processes (Oxford Studies in Probability)*. Oxford University Press, USA, January 1993.

[14] B. Vo, S. Singh, and A. Doucet. Sequential monte carlo methods for multi-target filtering with random finite sets. *IEEE Transactions on Aerospace and Electronic Systems*, 41(4):1224–1245, October 2005.