

Relating Random Vector and Random Finite Set Estimation in Navigation, Mapping, and Tracking

Keith Y. K. Leung, *Member, IEEE*, Felipe Inostroza, *Student Member, IEEE*,
and Martin Adams, *Senior Member, IEEE*

Abstract—Navigation, mapping, and tracking are state estimation problems relevant to a wide range of applications. These problems have traditionally been formulated using random vectors in stochastic filtering, smoothing, or optimization-based approaches. Alternatively, the problems can be formulated using random finite sets, which offer a more robust solution in poor detection conditions (i.e., low probabilities of detection, and high clutter intensity). This paper mathematically shows that the two estimation frameworks are related, and equivalences can be determined under a set of ideal detection conditions. The findings provide important insights into some of the limitations of each approach. These are validated using simulations with varying detection statistics, along with a real experimental dataset.

Index Terms—Robotic navigation, tracking, SLAM, random finite sets.

I. INTRODUCTION

NAVIGATION, mapping, and tracking, are all state estimation problems relevant to a number of fields such as mobile robotics, computer vision, and aerospace. Navigation is the process of estimating one's pose and is also referred to as the localization problem. Mapping is the process of estimating the locations of objects of interest that are usually static. When the two estimation problems are combined, it is referred to as the simultaneous localization and mapping (SLAM) problem in robotics, or structure from motion (SFM) in computer vision. When the objects in the map are dynamic, it is usually referred to as the target tracking problem. In solving these state estimation problems, a probabilistic approach is usually employed. SLAM and target tracking have traditionally been solved using stochastic Bayesian filtering [1]. In recent years, non-linear optimization (or batch estimation) methods that have traditionally been used in solving SFM and photogrammetry problems, have become more prominent in solving SLAM problems. Irrespective

Manuscript received September 29, 2016; revised March 9, 2017; accepted April 18, 2017. Date of publication May 4, 2017; date of current version June 28, 2017. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. D. Robert Iskander. This work was supported in part by the AMTC Universidad de Chile, and in part by FONDECYT projects 3150066 and 1150930. (*Corresponding author: Keith Y. K. Leung.*)

K. Y. K. Leung is with Applanix Corporation (Trimble), Richmond Hill, ON L4B 3B3, Canada (e-mail: kleung@applanix.com).

F. Inostroza and M. Adams are with the Department of Electrical Engineering, Advanced Mining Technology Center (AMTC), Universidad de Chile, Santiago 837-0451, Chile (finoestro@ug.uchile.cl; martin@ing.uchile.cl).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TSP.2017.2701330

as to whether filtering or smoothing is adopted, the problems have traditionally been formulated using (RVs) to represent the state. In recent years, the random finite set (RFS) formulation was introduced to the target tracking community, which treats the target state as a finite-set-valued random variable, and hence the cardinality (size) of the set is also a random variable. The RFS estimation framework, also referred to as multi-target tracking, calculates the multi-target distribution of targets and provides not only spatial information, but also the number of expected targets.

The objective of this article is to show that the RFS estimation framework has a close relationship with the random vector (RV) framework, in both Bayes filtering and batch estimation. Furthermore, this article will show that when assumptions are made on the detection and clutter statistics, and an assumption on the set cardinality is imposed onto the RFS framework, equivalences with the RV framework, for both stochastic filtering and batch estimation, can be inferred. From this, the conditions under which one approach may be favoured over the other will be shown, and the foundations of both RV and RFS representations are compared in order to draw such conclusions. This article is an extension of [2], in which the RFS filtering formulation of SLAM was proposed as a generalization of the RV formulation. In this article, the relationship between the two formulations is drawn for a wider range of problems. Furthermore, this article explains the relationship in both stochastic filtering and batch estimation. For experimental validation, this article examines a wider selection of scenarios under simulations in comparison to [2], and the results gathered from over 2000 simulation runs are presented. Real experimental datasets are also used.

The remainder of the article is organized as follows. Section II provides background information on the RFS estimation framework. Section III includes the mathematical formulations for navigation, mapping, and tracking problems in both the RFS and RV frameworks. Section IV shows the mathematical connection between the two frameworks, which requires a strict set of ideal detection conditions. In Section V these conditions will be relaxed to show connections between the RFS framework and particular RV SLAM concepts, namely the multi-hypothesis (MH)-factored solution to SLAM (Fast-SLAM) and maximum likelihood (ML) SLAM. Section VI discusses important ramifications of the ideal detection conditions which are, possibly unwittingly, applied in RV frameworks. The findings will be validated using simulations and real

experimental data in Section VII. Finally, Section VIII concludes the article.

II. BACKGROUND INFORMATION

From point process theory, mathematical tools called finite set statistics (FISST) were developed by Mahler [3], which provide the foundation for state estimation and data fusion using RFSs. The initial application of this approach was in target tracking. Aside from the ability to jointly estimate cardinality and the spatial distribution of targets, there are several benefits in using an RFS estimation approach compared to an RV approach. Typically in RV approaches, data association (or determining correspondences between measurements and state components for estimation updates) is performed using an algorithm that is separate from the estimator (e.g., choosing the correspondences which maximize the measurement likelihood). Within a Bayesian framework, when correspondences are likely with multiple objects, an MH method [4], [5] can be used to track the results of different data associations. Alternatively, the joint probabilistic data association (JPDA) filter [6], [7] can provide an average estimate composed of (measurement-likelihood) weighted updates from multiple hypotheses. In contrast, data association is mathematically a part of the estimation update process in the RFS approach. In the RFS Bayes filter, target estimates are updated by simultaneously combining them with every measurement to account for every possible correspondence hypothesis. Furthermore, the addition of state components based on new measurements is also handled natively within the RFS formulation. The use of sets allows this operation to be performed in a more mathematically principled manner, as opposed to arbitrarily changing the dimension of an RV. Another benefit of using RFSs is the inclusion of detection statistics (i.e., the probability of detection, and the amount of clutter or outliers expected from a sensor or feature detector) within the estimation framework.

Similar to RV filtering, RFS-based filtering methods also stem from the recursive Bayesian filtering paradigm.¹ FISSTs include meaningful definitions of set integrals and derivatives, and allow the application of Bayes estimation techniques for use with RFSs. However, as with the RV form, the general filter is also computationally intractable, except for certain classes of models which admit closed form solutions. The probability hypothesis density (PHD)² is the first statistical moment of an RFS's probability density function (PDF). Propagating this, instead of the full RFS PDF, approximates the RFS Bayes filter as the PHD filter [8]. Representing the PHD using a Gaussian mixture (GM) results in the GM-PHD filter [9]. This was applied to the SLAM problem in [10]. A closely-related and similar single-cluster PHD filter [11] was also applied to SLAM with moving targets in [12]. A generalization of the PHD filter, known as the cardinalized probability hypothesis density (CPHD) filter [13], relaxes the Poisson assumption by treating RFSs as independently and identically distributed (IID) clusters. This provides better

filtering performance at a higher computational cost. Alternatively, the RFS Bayes filter can be assumed to follow a multi-Bernoulli (MB) distribution, where elements within the RFS are assumed independent, and Bernoulli-distributed. This leads to the MB filter, which produces a biased cardinality estimate. The cardinality balanced MB filter was introduced as a remedy [14]. To facilitate the inclusion of track labels into the filtering process, the generalized labelled multi-Bernoulli (GLMB) filter was introduced [15]. In the same work, and in [16], the δ -GLMB filter was introduced as a sub-class of GLMB filters that offers better computational properties for target tracking applications. The labelled multi-Bernoulli (LMB) filter detailed in [17] can also be derived from the GLMB, and was applied to a navigation and mapping problem in [18].

Aside from Bayesian filtering, non-linear least squares (batch) estimation is another approach to solving navigation and mapping problems. This is more commonly seen in SFM and SLAM [19]–[23], but is also used in target tracking [24]–[26] in the form of forward-backward smoothers. For the non-linear least squares approach, various algebraic decompositions allow the RV problem to be solved efficiently [27]. However, finding a practically feasible solution for (non-backward-forward smoothing) batch estimation in RFS form is still an open problem to the best knowledge of the authors. Nevertheless, a mathematical connection can still be made between the batch estimation problem of the two forms, as will be shown in this article.

III. MATHEMATICAL FRAMEWORKS

In navigation, the goal is to estimate a vehicle or sensor pose, given measurements to objects with known positions. Conversely, in mapping and target tracking, the state of the vehicle or sensor is known, while the positions of objects (i.e., the static map or dynamic targets) must be estimated over time. SLAM and SFM are more general in that they combine both the navigation and mapping or tracking problems such that both vehicle or sensor pose, and the state of objects must be jointly estimated. In general, the underlying stochastic system for the estimation problem can be represented using the, in general, non-linear discrete-time equations:

$$\mathbf{x}_k = \mathbf{g}_x(\mathbf{x}_{k-1}, \mathbf{u}_k, \boldsymbol{\delta}_k) \quad (1)$$

$$\mathbf{m}_k^i = \mathbf{g}_m^i(\mathbf{m}_{k-1}^i, \boldsymbol{\gamma}_k^i) \quad (2)$$

$$\mathbf{z}_k^j = \mathbf{h}(\mathbf{x}_k, \mathbf{m}_k^i, \boldsymbol{\epsilon}_k) \quad (3)$$

where

- \mathbf{x}_k represents the vehicle or sensor pose at time-step k ,
- \mathbf{g}_x is the vehicle motion model,
- \mathbf{u}_k is the process control input at time-step k ,
- $\boldsymbol{\delta}_k$ is the process noise at time-step k ,
- \mathbf{m}^i represents the position of a map element or target i ,
- \mathbf{g}_m^i is the motion model for map element or target i ,
- $\boldsymbol{\gamma}_k^i$ is the process noise in \mathbf{g}_m^i ,
- \mathbf{z}_k^j is the j -th measurement vector at time-step k ,
- \mathbf{h} is the sensor-specific measurement model,
- $\boldsymbol{\epsilon}_k$ is the (typically spatial) measurement noise.

For the case of navigation or mapping, where objects of interest are static, the process model (2) becomes $\mathbf{m}_k^i = \mathbf{m}_{k-1}^i$.

¹In the tracking community, the RV Bayes filter is often referred to as the single-target Bayes filter, where as the RFS form is referred to as the multi-target Bayes filter.

²The PHD is also known as the *intensity* in the point process literature.

Furthermore, for the mapping and target tracking problems where the sensor state is known, (1) would not be used.

The number of measurements/detections, n_k , produced by the measurement/detection process is a random variable. The set \mathcal{Z}_k of measurement/detection vectors at time-step k is given by:

$$\mathcal{Z}_k \equiv \{ \mathbf{z}_k^1, \mathbf{z}_k^2, \dots, \mathbf{z}_k^{n_k} \}. \quad (4)$$

Note that the model representing the collected measurements/detections, based on a sensor scan, should fundamentally be an RFS rather than an RV, since their number can, in general, change from scan to scan, and the order in which they occur has no significant meaning [10].

A. Random Vector (RV) Formulation

In the RV formulation, the map or targets observed up to time-step k are concatenated into a single vector:

$$\mathbf{m}_k = (\mathbf{m}_k^1, \mathbf{m}_k^2, \dots, \mathbf{m}_k^m) \quad (5)$$

where m is the number of targets/map features deemed to exist, according to track/map management routines. Note that the j th measurement/detection \mathbf{z}_k^j will *not*, in general, correspond to the j th map element \mathbf{m}_k^j in (5), since the order in which measurements/detections occur may not correspond to the implicit ordering of the elements in (5). Therefore, in general, in (3), $i \neq j$.

For brevity, \mathbf{m}_k will be referred to as the RV map from hereon. The following formulation still holds for target tracking problems where \mathbf{m}_k is a non-static target [28].

Let θ represent the correspondences, where $\theta^j = i$ indicates that measurement j is associated with map element i , and $\theta^j = 0$ if measurement j is not associated. In the RV formulation, the goal is to calculate the posterior that represents the estimated state of the vehicle $\mathbf{x}_{0:k}$, the state of all map elements that have been observed \mathbf{m}_k and the unknown association history $\theta_{1:k}$:

$$p(\mathbf{x}_{0:k}, \mathbf{m}_k, \theta_{1:k} | \mathcal{Z}_{1:k}, \mathbf{u}_{0:k}) \quad (6)$$

where the notation $k_1 : k_2$ denotes the inclusive time interval from k_1 to k_2 for the referred quantity (e.g., $\mathbf{x}_{0:k} = \{\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_k\}$). For the navigation problem, \mathbf{m}_k is known and would be excluded from (6). Similarly, for mapping or target tracking, $\mathbf{x}_{0:k}$ is known and would be excluded.

Equation (6) actually corresponds to the posterior density modelling a Multi-Hypothesis Correlation (MHC) approach to estimation. A subtle issue regarding (6) exists, as stated in [3] (page 340):

“It is not possible to estimate θ [the association hypothesis] as a state parameter without implicitly introducing knowledge of the observation process into the definition of a multitarget state.”

This is because θ implicitly depends on a specific order of the measurement/detection elements $\mathbf{z}_k^1, \mathbf{z}_k^2, \dots, \mathbf{z}_k^{n_k}$ in (4). However, collected measurements/detections have no a priori order, which is why they are modelled as a set. Therefore, the association variable $\theta_{1:k}$ itself cannot be part of the state, as it would result in a clear paradox - where a state variable depends on the observation (its ordering) even though one attempts to estimate it directly in Bayes rule. This issue will be re-addressed in

Section IV-A, when the RFS based measurement likelihood is discussed.

In most RV based tracking and SLAM formulations, data association, i.e. the calculation/estimation of $\theta_{1:k}$, is carried out separately, and independently of the Bayesian update used to estimate the joint RV state $(\mathbf{x}_{0:k}, \mathbf{m}_k)$. Under this premise, the above issue of considering θ as a state variable, is usually ignored. The use of a measurement gating or data association algorithm, such as [29], [30], is usually applied to determine the correspondences (associations) between map elements and measurements before estimation updates. Once the association variables have been determined, only then is Bayes theorem, or ML estimation, used to update the joint distribution on the RV $(\mathbf{x}_{0:k}, \mathbf{m}_k)$. Therefore, the distribution which is actually updated is³:

$$p(\mathbf{x}_{0:k}, \mathbf{m}_k | \mathcal{Z}_{1:k}, \mathbf{u}_{0:k}, \theta_{1:k}). \quad (7)$$

Using a probabilistic framework and a filtering approach, this posterior is calculated recursively by Bayesian estimation, based on the assumed data association hypothesis $\theta_{1:k}$:

$$\begin{aligned} & p(\mathbf{x}_{0:k}, \mathbf{m}_k | \mathcal{Z}_{1:k-1}, \mathbf{u}_{0:k}, \theta_{1:k-1}) \\ &= p(\mathbf{x}_k | \mathbf{x}_{k-1}, \mathbf{u}_k) p(\mathbf{x}_{0:k-1}, \mathbf{m}_k | \mathcal{Z}_{1:k-1}, \mathbf{u}_{0:k-1}, \theta_{1:k-1}) \end{aligned} \quad (8)$$

$$\begin{aligned} & p(\mathbf{x}_{0:k}, \mathbf{m}_k | \mathcal{Z}_{1:k}, \mathbf{u}_{0:k}, \theta_{1:k}) \\ &= \frac{p(\mathcal{Z}_k | \mathbf{x}_{0:k}, \mathbf{m}_k, \theta_k) p(\mathbf{x}_{0:k}, \mathbf{m}_k | \mathcal{Z}_{1:k-1}, \mathbf{u}_{0:k}, \theta_{1:k-1})}{p(\mathcal{Z}_k | \mathcal{Z}_{1:k-1}, \mathbf{x}_{0:k}, \mathbf{u}_{0:k})}. \end{aligned} \quad (9)$$

Alternatively, and applying the same arguments regarding the data association variables $\theta_{1:k}$ as above, the SLAM problem is also often solved by a non-linear least squares (batch) estimation approach [20], [21]:

$$\begin{aligned} & \arg \max_{\mathbf{x}_{0:k}, \mathbf{m}_k} p(\mathbf{x}_{0:k}, \mathbf{m}_k | \mathcal{Z}_{1:k}, \mathbf{u}_{0:k}, \theta_{1:k}) \\ &= \arg \max_{\mathbf{x}_{0:k}, \mathbf{m}_k} p(\mathbf{x}_0, \mathbf{m}_k) \prod_{l=1}^k p(\mathcal{Z}_l | \mathbf{x}_l, \mathbf{m}_k, \theta_l) p(\mathbf{x}_l | \mathbf{x}_{l-1}, \mathbf{u}_l), \end{aligned} \quad (10)$$

where, in the case of SLAM, the terms within the product are the likelihoods of the exteroceptive measurements and process (vehicle motion model) inputs.

B. Random Finite Set (RFS) Formulation

Alternatively, the estimation problem can be modelled by placing the (RVs) for the map into an RFS, and defining the observed map elements up to time-step k as:

$$\mathcal{M}_k \equiv \{ \mathbf{m}_k^1, \mathbf{m}_k^2, \dots, \mathbf{m}_k^{m_k} \}, \quad (11)$$

where, in contrast to the RV map model, the cardinality, $|\mathcal{M}_k| = m_k$, is also a random variable. Using the RFS map model, the

³Note that (7) also contains a subtle issue since, in a strict mathematical sense, the association variable $\theta_{1:k}$ requires knowledge of the order of the elements within \mathbf{m}_k , which corresponds to knowledge related to the state [3]. Nevertheless, this joint density is given here to indicate the way in which RV tracking and SLAM approaches are often formulated.

required posterior can be rewritten as⁴:

$$p(\mathbf{x}_{0:k}, \mathcal{M}_k | \mathcal{Z}_{1:k}, \mathbf{u}_{0:k}) \quad (12)$$

This can also be solved by Bayesian estimation:

$$\begin{aligned} & p(\mathbf{x}_{0:k}, \mathcal{M}_k | \mathcal{Z}_{1:k-1}, \mathbf{u}_{0:k}) \\ &= p(\mathbf{x}_k | \mathbf{x}_{k-1}, \mathbf{u}_k) p(\mathbf{x}_{0:k-1}, \mathcal{M}_k | \mathcal{Z}_{1:k-1}, \mathbf{u}_{0:k-1}), \end{aligned} \quad (13)$$

$$\begin{aligned} & p(\mathbf{x}_{0:k}, \mathcal{M}_k | \mathcal{Z}_{1:k}, \mathbf{u}_{0:k}) \\ &= \frac{p(\mathcal{Z}_k | \mathbf{x}_{0:k}, \mathcal{M}_k) p(\mathbf{x}_{0:k}, \mathcal{M}_k | \mathcal{Z}_{1:k-1}, \mathbf{u}_{0:k-1})}{p(\mathcal{Z}_k | \mathcal{Z}_{1:k-1}, \mathbf{x}_{0:k}, \mathbf{u}_{0:k})}. \end{aligned} \quad (14)$$

Although an RFS batch estimation formulation is still an open problem, it is possible to express it as:

$$\begin{aligned} & \arg \max_{\mathbf{x}_{0:k}, \mathcal{M}_k} p(\mathbf{x}_{0:k}, \mathcal{M}_k | \mathcal{Z}_{1:k}, \mathbf{u}_{0:k}) \\ &= \arg \max_{\mathbf{x}_{0:k}, \mathcal{M}_k} p(\mathbf{x}_0, \mathcal{M}_k) \prod_{l=1}^k p(\mathcal{Z}_l | \mathbf{x}_l, \mathcal{M}_k) p(\mathbf{x}_l | \mathbf{x}_{l-1}, \mathbf{u}_l). \end{aligned} \quad (15)$$

The above optimization is difficult because it needs to be performed over continuous variables for the vehicle and map state, as well as the discrete map cardinality variable.

An advantage of RFS estimation is the possibility of incorporating detection statistics within the set measurement likelihood. The *probability of detection*, $P_D(\mathbf{x}_k, \mathbf{m}_k^i)$, is the probability of obtaining a measurement given a robot pose, \mathbf{x}_k , and a map element state, \mathbf{m}_k^i . Furthermore, $p_{\kappa}(\mathcal{Z}_k)$ is the likelihood of a set of measurements being clutter (or false alarms). The relationship between the RFS and RV formulations will be examined next.

IV. RELATING THE RFS AND THE RV FRAMEWORKS

In this section, the specific conditions under which the RFS formulation would be equivalent to the general RV framework will be mathematically shown.

Starting in the RFS framework, in the Bayes filter equations of (13) and (14), as well as (15) for batch estimation, several RFS PDFs are present. From the Bayes filter prediction (13), there is the conditional PDF that is expressed over the vehicle trajectory and the map:

$$p(\mathbf{x}_{0:k}, \mathcal{M}_k | \mathcal{Z}_{1:k-1}, \mathbf{u}_{0:k}). \quad (16)$$

Common to both the filtering and batch estimation approaches is the set measurement likelihood,

$$p(\mathcal{Z}_k | \mathbf{x}_{0:k}, \mathcal{M}_k), \quad (17)$$

which appears in (14) and (15). In the denominator of (14), there is the normalizing factor:

$$p(\mathcal{Z}_k | \mathcal{Z}_{1:k-1}, \mathbf{x}_{0:k}, \mathbf{u}_{0:k}). \quad (18)$$

This factor is the same in both the RFS and RV formulations (denominators of both (9) and (14)). For the distributions in

⁴Note that (12) is a FISST density [3] which was shown in [31] to be equivalent to a standard probability density with respect to an unnormalized Poisson point process.

(16) and (17) however, the conditions under which they would become equivalent in the RFS and RV frameworks will now be shown.

Theorem 1: Given the joint distribution for the vehicle and RFS map, if the cardinality of the RFS map is deterministic, and the map elements can only be placed in an RV in one particular order, then the joint distribution is equivalent to that for the vehicle and RV map.

Proof: Suppose the joint distribution for the vehicle and RV map is

$$\begin{aligned} & p(\mathbf{x}_{0:k}, \mathbf{m}_k | \mathcal{Z}_{1:k-1}, \mathbf{u}_{0:k}, \boldsymbol{\theta}_{1:k-1}) \\ &= p(\mathbf{x}_{0:k}, (\mathbf{m}_k^1, \mathbf{m}_k^2, \dots, \mathbf{m}_k^m) | \mathcal{Z}_{1:k-1}, \mathbf{u}_{0:k}, \boldsymbol{\theta}_{1:k-1}) \end{aligned} \quad (19)$$

Using FISST [3], the distribution over the vehicle and the map can be expressed as:⁵

$$\begin{aligned} & p(\mathbf{x}_{0:k}, \mathcal{M}_k | \mathcal{Z}_{1:k-1}, \mathbf{u}_{0:k}) \\ &= m_k! p_{|\mathcal{M}_k|}(m_k) p(\mathbf{x}_{0:k}, (\mathbf{m}_k^1, \dots, \mathbf{m}_k^{m_k}) | \mathcal{Z}_{1:k-1}, \mathbf{u}_{0:k}) \end{aligned} \quad (20)$$

where m_k is a random variable. If the cardinality m_k of the RFS map is deterministic and equal to m , then $p_{|\mathcal{M}_k|}(m) = 1$, and (20) becomes

$$m! p(\mathbf{x}_{0:k}, (\mathbf{m}_k^1, \dots, \mathbf{m}_k^m) | \mathcal{Z}_{1:k-1}, \mathbf{u}_{0:k}). \quad (21)$$

The factorial term represents all the different permutations of the members in \mathcal{M}_k , and all permutations have the same likelihood (i.e., they are symmetric). If only one particular ordering is allowed, then the likelihoods of all other orderings become zero, and (21) becomes

$$p(\mathbf{x}_{0:k}, (\mathbf{m}_k^1, \dots, \mathbf{m}_k^m) | \mathcal{Z}_{1:k-1}, \mathbf{u}_{0:k}). \quad (22)$$

Note that if (19) contains a different map element ordering, the same one can be selected from within (21). Finally, if correspondence information is given, then (22) equals

$$p(\mathbf{x}_{0:k}, (\mathbf{m}_k^1, \mathbf{m}_k^2, \dots, \mathbf{m}_k^m) | \mathcal{Z}_{1:k-1}, \mathbf{u}_{0:k}, \boldsymbol{\theta}_{1:k-1}), \quad (23)$$

which is equivalent to the joint likelihood of the vehicle pose and the RV map in (19). ■

The significance of Theorem 1 is that it shows how the RV framework requires a deterministic map size and selects one ordering of map elements out of all permutations in the estimation process. Furthermore, correspondence selection by a data association routine is required.

A. A Note on RV and RFS Measurement Likelihoods

In Section III-A, a subtle, but important, issue was noted in connection with (6). It was explained that considering $\boldsymbol{\theta}$ as a state variable, implicitly requires a certain ordering of the measurements/detections $\mathbf{z}_k^1, \mathbf{z}_k^2, \dots, \mathbf{z}_k^{n_k}$ in \mathcal{Z}_k , which in fact does not exist. In [3](page 341), Mahler posed the question:

“In choosing such an ordering, are we introducing extraneous information – and thus also a statistical bias?”

⁵The density for the RFS map is a Janossy density.

Hence it does not seem to be possible to estimate the correct association hypothesis θ without introducing information into the state model corresponding to an a priori order of the measurements/detections, which have yet to be recorded.

It was shown that this problem can, and should, be solved in order to derive a statistically unbiased multitarget measurement model [3]. The RFS measurement likelihood of (17), if based on an *individual association hypothesis* θ , would be expressed as $p(\mathcal{Z}_k^\theta | \mathbf{x}_{0:k}, \mathcal{M}_k^\theta, \theta)$. This form of the likelihood would yield the problem of an association dependent state in both (14) and (15). The problem is solved by replacing the likelihood functions $p(\mathcal{Z}_k^\theta | \mathbf{x}_{0:k}, \mathcal{M}_k^\theta, \theta)$ with their *average value* w.r.t. to θ to form the unbiased RFS measurement likelihood $p(\mathcal{Z}_k | \mathbf{x}_{0:k}, \mathcal{M}_k)$:

$$p(\mathcal{Z}_k | \mathbf{x}_{0:k}, \mathcal{M}_k) \equiv \frac{1}{N_{\mathcal{Z}_k}} \sum_{\theta} p(\mathcal{Z}_k^\theta | \mathbf{x}_{0:k}, \mathcal{M}_k^\theta, \theta) \quad (24)$$

where $N_{\mathcal{Z}_k}$ is a normalization constant such that (24) integrates to unity. Under this definition, it can be seen that the association variable θ is effectively marginalized out, and thus has no effect on the RFS measurement likelihood $p(\mathcal{Z}_k | \mathbf{x}_{0:k}, \mathcal{M}_k)$.

Within the RFS multi-object framework, the measurement likelihood can be more generally expressed in terms of a joint state's probability of detection $P_D(\mathbf{x}_k, \mathbf{m})$, its probability of misdetection $1 - P_D(\mathbf{x}_k, \mathbf{m})$ and the likelihood of measurements \mathcal{Z}_k being clutter $p(\mathcal{Z}_k)$. When the measurements are assumed to follow a multi-Bernoulli distribution, the RFS measurement likelihood function is given as (25) shown at the bottom of this page, [3]. Note that, as shown in (24), the RFS measurement likelihood results from averaging over all data association hypotheses θ , taking them all into account equally. This measurement likelihood therefore consists of additive terms based upon each measurement to state association hypothesis θ . Typically, measurements are assumed to be conditionally independent, when conditioned on the state. This assumption is made here also and would make detections conditionally independent. For instance, the detections of two objects, where one occludes the other, are independent when conditioned on both objects' state.

Theorem 2: Given the RFS measurement likelihood,⁶ if 1) the correspondence between measurements and map elements is given, and 2) if it is possible to assume that only associated

map elements are detected, with perfect certainty, and 3) non-associated measurements are always considered to be clutter, then the RFS measurement likelihood would be equivalent to the usually adopted RV measurement likelihood.

Proof: Given the correspondence θ , let:

$$\mathcal{Z}_k^\theta \equiv \{z_k^j \in \mathcal{Z}_k | \theta^j \neq 0\} \quad (26)$$

$$\overline{\mathcal{Z}_k^\theta} \equiv \mathcal{Z}_k \setminus \mathcal{Z}_k^\theta \quad (27)$$

$$\mathcal{M}_k^\theta \equiv \{m_k^i \in \mathcal{M}_k | \exists j, \theta^j = i\} \quad (28)$$

$$\overline{\mathcal{M}_k^\theta} \equiv \mathcal{M}_k \setminus \mathcal{M}_k^\theta \quad (29)$$

define the set of associated and non-associated measurements, as well as the associated and non-associated map elements, respectively.

Based on the definitions of the measurement and map sets ((26) and (28)), given the correspondence θ , the RV measurement likelihood term in both (9) and (10), can be expressed as

$$p(\mathcal{Z}_k | \mathbf{x}_{0:k}, \mathbf{m}_k, \theta) = p(\mathcal{Z}_k^\theta | \mathbf{x}_{0:k}, \mathcal{M}_k^\theta, \theta). \quad (30)$$

The RFS measurement likelihood of (25) contains a summation over θ , which considers all possible permutations of data association, where measurements can be associated with a state (e.g. map) element, or considered clutter. Map elements may be associated with a measurement or misdetection. Unpaired measurements give the clutter factor, $p_\kappa(\overline{\mathcal{Z}_k^\theta})$, while unpaired map elements give the misdetection factors, $(1 - P_D(\cdot))$. Paired couples (where $\theta^j = i \neq 0$) provide the probabilities of detection, $P_D(\cdot)$, and the measurement likelihood factor for the given data association hypothesis, $p(\mathcal{Z}_k^\theta | \mathcal{M}_k^\theta, \mathbf{x}_{0:k}, \theta)$.

If a set of correspondences θ^* can be assumed,

$$p(\mathcal{Z}_k^\theta | \mathcal{M}_k^\theta, \mathbf{x}_{0:k}, \theta) = 0 \quad \text{if } \theta \neq \theta^*. \quad (31)$$

Furthermore, it is necessary to assume that the associated map elements are the only ones detected, with perfect certainty:

$$P_D(\mathbf{x}_k, \mathbf{m}) = \begin{cases} 1 & \text{if } \mathbf{m} \in \mathcal{M}_k^{\theta^*} \\ 0 & \text{otherwise} \end{cases}. \quad (32)$$

The assumptions of (31), and (32) have important ramifications on the performance of RV SLAM filters, which will be discussed in Section VI.

$$p(\mathcal{Z}_k | \mathcal{M}_k, \mathbf{x}_{0:k}) = \sum_{\theta} \left(\underbrace{p(\mathcal{Z}_k^\theta | \mathcal{M}_k^\theta, \mathbf{x}_{0:k}, \theta)}_{\text{Spatial measurement likelihood for state-measurement pairs under association } \theta} \underbrace{\left[\prod_{\mathbf{m} \in \mathcal{M}_k^\theta} P_D(\mathbf{x}_k, \mathbf{m}) \right]}_{\text{For all associated state elements: Product of their detection probabilities}} \underbrace{\left[\prod_{\mathbf{m} \in \overline{\mathcal{M}_k^\theta}} (1 - P_D(\mathbf{x}_k, \mathbf{m})) \right]}_{\text{For all unassociated state elements: Product of their misdetection probabilities}} \underbrace{p_\kappa(\overline{\mathcal{Z}_k^\theta})}_{\text{Likelihood of non-associated measurements being clutter}} \right) \quad (25)$$

⁶This is usually referred to as the *multitarget likelihood function*.

In addition, it is also necessary to assume that non-associated measurements are always considered clutter:

$$p_{\kappa} \left(\overline{\mathcal{Z}_k^{\theta^*}} \right) = 1. \quad (33)$$

Substituting (31), (32), and (33) into (25) gives:

$$\begin{aligned} & p(\mathcal{Z}_k | \mathcal{M}_k, \mathbf{x}_{0:k}), \quad \text{given } \theta^* \\ & = p \left(\mathcal{Z}_k^{\theta^*} | \mathcal{M}_k^{\theta^*}, \mathbf{x}_{0:k}, \theta^* \right), \end{aligned} \quad (34)$$

which is equivalent to the RV likelihood (30). ■

The significance of Theorem 2 is that it shows the assumptions on the detection statistics that are implicit in the RV framework, where outlier rejection algorithms are assumed to discard measurements deemed not to originate from the map. From the remaining set of measurements, data association is assumed to determine correspondences with map elements. It is entirely possible that the assumed data association is incorrect, which could potentially lead to estimate divergence.

In summary, the joint distribution for the vehicle and map, and the measurement likelihoods, are conditional PDFs that appear in the RFS and RV forms of both the Bayes filter, and the batch estimation frameworks. Through Theorems 1 and 2, these conditional PDFs, and hence the estimation problems of navigation, mapping, and tracking, can be inferred as equivalent under the following conditions, which from here-on will be referred to as the *ideal detection conditions*:

- 1) The size of the map estimate is deterministic.
- 2) Data association is assumed.
- 3) The probability of detection of a map element with an associated measurement is unity, but zero if unassociated.
- 4) The likelihood of non-associated measurements being clutter is unity.

V. RELAXING THE IDEAL DETECTION CONDITIONS

Under the Bayesian estimation framework, the complete relaxation of the ideal detection conditions introduced in Section IV-B gives rise to the general RFS measurement likelihood of (25), which if substituted into (14) is the basis of RFS based filters including the PHD [8], CPHD [13], MB [14] and GLMB [16] filters.

This section relaxes the ideal detection conditions by firstly not assuming a given data association and secondly by not assuming unity probability of detection of associated map features. Theoretical comparisons of RFS based SLAM solutions with these “relaxed” ideal detection conditions, are then made with currently published RV SLAM solutions, which assume similar conditions.

A. Relaxing Condition 2: No Assumed Data Association

This section will explore the relationship between the RFS SLAM and RV MH-FastSLAM formulations. MH-FastSLAM factorizes the joint SLAM posterior into a distribution on the robot’s trajectory and the map, conditioned on that trajectory, and allows for multiple data association hypotheses. This concept will now be applied to the RFS SLAM posterior of (12), adopting the RFS measurement likelihood of (25), in order to

determine if RV MH-FastSLAM can be derived from (13) and (14) when ideal detection condition 2) is relaxed.

The full RFS SLAM posterior can be factored as follows:

$$\begin{aligned} & p(\mathbf{x}_{0:k}, \mathcal{M}_k | \mathcal{Z}_{1:k}, \mathbf{u}_{0:k}) \\ & = p(\mathcal{M}_k | \mathbf{x}_{0:k}, \mathcal{Z}_{1:k}, \mathbf{u}_{0:k}) p(\mathbf{x}_{0:k} | \mathcal{Z}_{1:k}, \mathbf{u}_{0:k}). \end{aligned} \quad (35)$$

In MH-FastSLAM, the distribution on the robot’s trajectory $p(\mathbf{x}_{0:k} | \mathcal{Z}_{1:k}, \mathbf{u}_{0:k})$ is represented as N particles:

$$p(\mathbf{x}_{0:k} | \mathcal{Z}_{1:k}, \mathbf{u}_{0:k}) \approx \sum_{q=1}^N w_q \delta(\mathbf{x}_{0:k} - \mathbf{x}_{0:k}^q), \quad (36)$$

where each robot trajectory particle $\mathbf{x}_{0:k}^q$ has weight w_q and $\delta(\cdot)$ is the Dirac delta function.

Prediction is then carried out such that:

$$\mathbf{x}_{0:k}^q \longrightarrow \mathbf{x}_{0:k+1}^q | \mathcal{Z}_{1:k}, \mathbf{u}_{k+1} \quad (37)$$

$$\mathcal{M}_k \longrightarrow \mathcal{M}_{k+1} | \mathcal{Z}_{1:k}, \mathbf{u}_{k+1} \quad (38)$$

That is,

$$\begin{aligned} & p(\mathbf{x}_{0:k+1}, \mathcal{M}_{k+1} | \mathcal{Z}_{1:k}, \mathbf{u}_{0:k+1}) \\ & \approx \sum_{q=1}^N w_q \delta(\mathbf{x}_{0:k+1} - \mathbf{x}_{0:k+1}^q) p(\mathcal{M}_{k+1} | \mathbf{x}_{0:k+1}^q, \mathcal{Z}_{1:k}, \mathbf{u}_{0:k+1}). \end{aligned} \quad (39)$$

A new measurement RFS \mathcal{Z}_{k+1} yields the Bayesian update

$$\begin{aligned} & p(\mathbf{x}_{0:k+1}, \mathcal{M}_{k+1} | \mathcal{Z}_{1:k+1}, \mathbf{u}_{0:k+1}) \\ & = \eta p(\mathcal{Z}_{k+1} | \mathcal{M}_{k+1}, \mathbf{x}_{0:k+1}) p(\mathbf{x}_{0:k+1}, \mathcal{M}_{k+1} | \mathcal{Z}_{1:k}, \mathbf{u}_{0:k+1}), \end{aligned} \quad (40)$$

where η is a normalization constant and the Markov assumption has been applied such that

$$p(\mathcal{Z}_{k+1} | \mathcal{M}_{k+1}, \mathbf{x}_{0:k+1}, \mathcal{Z}_{1:k}, \mathbf{u}_{1:k}) = p(\mathcal{Z}_{k+1} | \mathcal{M}_{k+1}, \mathbf{x}_{0:k+1}). \quad (41)$$

Substituting (39) into (40) gives

$$\begin{aligned} & p(\mathbf{x}_{0:k+1}, \mathcal{M}_{k+1} | \mathcal{Z}_{1:k+1}, \mathbf{u}_{0:k+1}) \\ & \approx \eta \sum_{q=1}^N (w_q \delta(\mathbf{x}_{0:k+1} - \mathbf{x}_{0:k+1}^q) p(\mathcal{Z}_{k+1} | \mathcal{M}_{k+1}, \mathbf{x}_{0:k+1}^q) \\ & \quad \times p(\mathcal{M}_{k+1} | \mathbf{x}_{0:k+1}^q, \mathcal{Z}_{1:k})). \end{aligned} \quad (42)$$

Applying Bayes theorem to the map \mathcal{M}_{k+1} gives

$$\frac{p(\mathcal{M}_{k+1} | \mathbf{x}_{0:k+1}^q, \mathcal{Z}_{1:k})}{p(\mathcal{Z}_{k+1} | \mathcal{Z}_{1:k}, \mathbf{x}_{0:k+1}^q)} = \frac{p(\mathcal{M}_{k+1} | \mathbf{x}_{0:k+1}^q, \mathcal{Z}_{1:k+1})}{p(\mathcal{Z}_{k+1} | \mathcal{M}_{k+1}, \mathbf{x}_{0:k+1}^q)}, \quad (43)$$

which allows the last two terms in (42) to be replaced as follows

$$\begin{aligned} & p(\mathbf{x}_{0:k+1}, \mathcal{M}_{k+1} | \mathcal{Z}_{1:k+1}, \mathbf{u}_{0:k+1}) \\ & \approx \eta \sum_{q=1}^N (w_q \delta(\mathbf{x}_{0:k+1} - \mathbf{x}_{0:k+1}^q) p(\mathcal{Z}_{k+1} | \mathcal{Z}_{1:k}, \mathbf{x}_{0:k+1}^q) \\ & \quad \times p(\mathcal{M}_{k+1} | \mathbf{x}_{0:k+1}^q, \mathcal{Z}_{1:k+1})), \end{aligned} \quad (44)$$

where $p(\mathcal{Z}_{k+1}|\mathcal{Z}_{1:k}, \mathbf{x}_{0:k+1}^q)$ contributes to the new trajectory particle weight and can be expanded as

$$\begin{aligned} & p(\mathcal{Z}_{k+1}|\mathcal{Z}_{1:k}, \mathbf{x}_{0:k+1}^q) \\ &= \int_{\mathcal{M}_{k+1}} p(\mathcal{Z}_{k+1}|\mathcal{M}_{k+1}, \mathcal{Z}_{1:k}, \mathbf{x}_{0:k+1}^q) \\ & \quad \times p(\mathcal{M}_{k+1}|\mathcal{Z}_{1:k}, \mathbf{x}_{0:k+1}^q) d\mathcal{M}_{k+1}. \end{aligned} \quad (45)$$

To determine the RFS equivalent of MH-FastSLAM, it is necessary to substitute the full measurement likelihood equivalent of (25), $p(\mathcal{Z}_{k+1}|\mathcal{M}_{k+1}, \mathbf{x}_{0:k+1})$ into (45). As in the case of standard RV SLAM approaches, an exact equivalence can only be shown by applying the conditions in (32) and (33). In this case, from (25)

$$\begin{aligned} & p(\mathcal{Z}_{k+1}|\mathcal{M}_{k+1}, \mathbf{x}_{0:k+1}) = \sum_{\theta} p(\mathcal{Z}_{k+1}^\theta|\mathcal{M}_{k+1}^\theta, \mathbf{x}_{0:k+1}, \theta) \\ &= \sum_{\theta} \prod_{\substack{\mathbf{m} \in \mathcal{M}_{k+1}^\theta \\ \mathbf{z} \in \mathcal{Z}_{k+1}^\theta}} p(\mathbf{z}^j|\mathbf{m}^i, \mathbf{x}_{k+1}), \quad \theta^j = i. \end{aligned} \quad (46)$$

Substitution of (46) into (45) yields

$$\begin{aligned} & p(\mathcal{Z}_{k+1}|\mathcal{Z}_{1:k}, \mathbf{x}_{0:k+1}^q) \\ &= \sum_{\theta} \int_{\mathcal{M}_{k+1}} \prod_{\substack{\mathbf{m} \in \mathcal{M}_{k+1}^\theta \\ \mathbf{z} \in \mathcal{Z}_{k+1}^\theta}} p(\mathbf{z}^j|\mathbf{m}^i, \mathbf{x}_{k+1}) \\ & \quad \times p(\mathcal{M}_{k+1}|\mathcal{Z}_{1:k}, \mathbf{x}_{0:k+1}^q) d\mathcal{M}_{k+1} \end{aligned} \quad (47)$$

which, according to Theorem 1, can be expressed as

$$\begin{aligned} & p(\mathcal{Z}_{k+1}|\mathcal{Z}_{1:k}, \mathbf{x}_{0:k+1}^q) = \\ & \underbrace{\sum_{\theta} \int \dots \int \prod_{\substack{\mathbf{m} \in \mathcal{M}_{k+1}^\theta \\ \mathbf{z} \in \mathcal{Z}_{k+1}^\theta}} p(\mathbf{z}^j|\mathbf{m}^i, \mathbf{x}_{k+1}) p(\mathbf{m}^i|\mathbf{x}_{0:k+1}^q, \mathcal{Z}_{1:k}) d\mathbf{m}^i}_{\text{FastSLAM single target weight } w_{\text{RV}}(\mathcal{Z}_{1:k+1}, \mathcal{X}_{k+1}, \theta)}, \end{aligned} \quad (48)$$

where $w_{\text{RV}}(\mathcal{Z}_{1:k+1}, \mathcal{X}_{k+1}, \theta)$ corresponds to the particle weight of RV FastSLAM. Finally, substitution of (48) into (44) gives

$$\begin{aligned} & p(\mathbf{x}_{0:k+1}, \mathcal{M}_{k+1}|\mathcal{Z}_{1:k+1}, \mathbf{u}_{0:k+1}) \\ & \approx \eta \sum_{q=1}^N \sum_{\theta} w_q w_{\text{RV}}(\mathcal{Z}_{1:k+1}, \mathcal{X}_{k+1}, \theta) \delta(\mathbf{x}_{0:k+1} - \mathbf{x}_{0:k+1}^q) \\ & \quad \times p(\mathcal{M}_{k+1}|\mathbf{x}_{0:k+1}^q, \mathcal{Z}_{1:k+1}). \end{aligned} \quad (49)$$

Exactly as in RV MH-FastSLAM, (49) shows that for each particle q , a new, identical copy of that particle, in terms of its trajectory, is introduced, each with different association hypotheses θ , due to the double summation. The weight of each particle is now the product $w_q w_{\text{RV}}(\mathcal{Z}_{1:k+1}, \mathcal{X}_{k+1}, \theta)$, corresponding to the particle weights in RV MH-FastSLAM.

For reasons of tractability, in MH-FastSLAM, the updated map $p(\mathcal{M}_{k+1}|\mathbf{x}_{0:k+1}^q, \mathcal{Z}_{1:k+1})$, in (49), is calculated using the extended Kalman filter (EKF).

$$\arg \max_{\mathbf{x}_{0:k}, \mathcal{M}_k} p(\mathbf{x}_{0:k}, \mathcal{M}_k|\mathcal{Z}_{1:k}, \mathbf{u}_{0:k}) \quad (50)$$

$$\begin{aligned} &= \arg \max_{\mathbf{x}_{0:k}, \mathcal{M}_k} p(\mathbf{x}_0, \mathcal{M}_k) \prod_k \left[p(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{u}_k) p(\mathcal{Z}_k^{\theta^*}|\mathcal{M}_k^{\theta^*}, \mathbf{x}_{0:k}, \theta^*) \prod_{\mathbf{m} \in \mathcal{M}_k^{\theta^*}} P_D(\mathbf{x}_k, \mathbf{m}) \prod_{\mathbf{m} \in \overline{\mathcal{M}_k^{\theta^*}}} (1 - P_D(\mathbf{x}_k, \mathbf{m})) \right] \\ &= \arg \max_{\mathbf{x}_{0:k}, \mathcal{M}_k} p(\mathbf{x}_0, \mathcal{M}_k) \prod_k \left[p(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{u}_k) \prod_{\substack{\mathbf{m} \in \mathcal{M}_k^{\theta^*} \\ \mathbf{z} \in \mathcal{Z}_k^{\theta^*}}} p(\mathbf{z}^j|\mathbf{m}^i, \mathbf{x}_k) \prod_{\mathbf{m} \in \mathcal{M}_k^{\theta^*}} P_D(\mathbf{x}_k, \mathbf{m}) \prod_{\mathbf{m} \in \overline{\mathcal{M}_k^{\theta^*}}} (1 - P_D(\mathbf{x}_k, \mathbf{m})) \right]. \end{aligned}$$

$$\arg \min_{\mathbf{x}_{0:k}, \mathcal{M}_k} (-\log p(\mathbf{x}_{0:k}, \mathcal{M}_k|\mathcal{Z}_{1:k}, \mathbf{u}_{0:k})) = \arg \min_{\mathbf{x}_{0:k}, \mathcal{M}_k} \left[-c - \log(p(\mathbf{x}_0, \mathcal{M}_k)) \right] \quad (51)$$

$$\left. - \sum_k \left[\log(p(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{u}_k)) + \sum_{\substack{\mathbf{m} \in \mathcal{M}_k^{\theta^*} \\ \mathbf{z} \in \mathcal{Z}_k^{\theta^*}}} \log(p(\mathbf{z}^j|\mathbf{m}^i, \mathbf{x}_k)) + \sum_{\mathbf{m} \in \mathcal{M}_k^{\theta^*}} \log(P_D(\mathbf{x}_k, \mathbf{m})) + \sum_{\mathbf{m} \in \overline{\mathcal{M}_k^{\theta^*}}} \log((1 - P_D(\mathbf{x}_k, \mathbf{m}))) \right] \right\}.$$

$$p(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{u}_k) = \det((2\pi)^{|\mathbf{x}_k|} \mathbf{Q}_k)^{-\frac{1}{2}} \exp \left\{ -\frac{1}{2} (\mathbf{x}_k - \mathbf{g}_x(\mathbf{x}_{k-1}, \mathbf{u}_k))^T \mathbf{Q}_k^{-1} (\mathbf{x}_k - \mathbf{g}_x(\mathbf{x}_{k-1}, \mathbf{u}_k)) \right\}, \quad \mathbf{Q}_k = \mathbb{E} [\delta_k \delta_k^T]. \quad (52)$$

$$p(\mathbf{z}_k^j|\mathbf{m}^i, \mathbf{x}_k) = \det((2\pi)^{|\mathbf{z}_k^j|} \mathbf{R}_k)^{-\frac{1}{2}} \exp \left\{ -\frac{1}{2} (\mathbf{z}_k^j - \mathbf{g}_m(\mathbf{x}_k, \mathbf{m}^i))^T \mathbf{R}_k^{-1} (\mathbf{z}_k^j - \mathbf{g}_m(\mathbf{x}_k, \mathbf{m}^i)) \right\}, \quad \mathbf{R}_k = \mathbb{E} [\epsilon_k \epsilon_k^T]. \quad (53)$$

B. Relaxing Condition 3: Non-Unity Detection Probabilities

Recently, much of the SLAM literature has focused on ML batch estimation solutions of the form of (10), which in its RFS form is given in (15). Such solutions are typically divided into two parts, one being referred to as the SLAM “front end”, which attempts to solve the measurement to feature data association problem independently of what is referred to as the SLAM “back end” which then executes the ML estimation step in (10) [23]. This section now relaxes ideal detection condition 3), maintaining a state dependent probability of detection $P_D(\mathbf{x}_k, \mathbf{m})$. Under these assumptions, the substitution of (25) into (15) yields (50). The negative log-likelihood can then be applied yielding (51). Under the Gaussian process and spatial measurement model assumptions of (52) and (53), where δ_k and ϵ_k were first introduced in (1) and (3) respectively, this yields an equation similar to the RV GraphSLAM solution [20]:

$$\begin{aligned}
 -\log p(\mathbf{x}_{0:k}, \mathcal{M}_k | \mathcal{Z}_{1:k}, \mathbf{u}_{0:k}) &= \text{const.} + \frac{1}{2} \mathbf{x}_0^T \mathbf{\Omega}_0 \mathbf{x}_0 \quad (54) \\
 &+ \frac{1}{2} \sum_k (\mathbf{x}_k - \mathbf{g}_x(\mathbf{x}_{k-1}, \mathbf{u}_k))^T \mathbf{Q}_k^{-1} (\mathbf{x}_k - \mathbf{g}_x(\mathbf{x}_{k-1}, \mathbf{u}_k)) \\
 &+ \frac{1}{2} \sum_k \sum_{\substack{\mathbf{m} \in \mathcal{M}_k^{\theta^*} \\ \mathbf{z} \in \mathcal{Z}_k^{\theta^*}}} (\mathbf{z}_k^j - \mathbf{g}_m(\mathbf{x}_k, \mathbf{m}^i))^T \mathbf{R}_k^{-1} (\mathbf{z}_k^j - \mathbf{g}_m(\mathbf{x}_k, \mathbf{m}^i)) \\
 &- \sum_k \left[\sum_{\mathbf{m} \in \mathcal{M}_k^{\theta^*}} \log P_D(\mathbf{x}_k, \mathbf{m}) + \sum_{\mathbf{m} \in \mathcal{M}_k^{\theta^*}} \log(1 - P_D(\mathbf{x}_k, \mathbf{m})) \right],
 \end{aligned}$$

where the prior $p(\mathbf{x}_0, \mathcal{M}_k)$ in (50) has been factored into two independent priors $p(\mathbf{x}_0) = \eta \exp\{-\frac{1}{2} \mathbf{x}_0^T \mathbf{\Omega}_0 \mathbf{x}_0\}$, where $\mathbf{\Omega}_0$ is the initial robot pose covariance matrix, and $p(\mathcal{M}_k)$. Since no prior knowledge is usually available about the map \mathcal{M}_k , $p(\mathcal{M}_k)$ is assumed uniform and subsumed into the constant.

The difference between (54) and the RV GraphSLAM solution is the final line in (54), which would be zero under ideal detection condition 3). It is interesting to note that various RV solutions to GraphSLAM incorporate extra terms, in an attempt to improve the robustness of the SLAM back-end in the presence of false loop closures [23], [32]. Although a direct relation between these methods and (54) cannot be precisely defined at this stage, SLAM solutions which minimize the log likelihood in (54), taking into account $P_D(\mathbf{x}_k, \mathbf{m})$, could provide an interesting avenue for further research.

VI. THE “IDEAL DETECTION CONDITIONS” CONUNDRUM

Note that the only way to derive current RV Bayesian and ML SLAM estimators from the general RFS measurement likelihood of (25) is by invoking assumptions (31) and/or (32). Close examination of these assumptions exposes a conundrum in the foundations of such estimators. As was already shown in Section IV-B, the RFS measurement likelihood is formed by averaging over all association hypotheses, taking them all into account equally. To derive the basis of the RV SLAM formulations in Section IV-B from, and to relate the basis of current RV

ML SLAM approaches in Section V-C to, the RFS measurement likelihood of (25), the selection of a single association $\theta = \theta^*$ is necessary. This gives rise to (31), which may introduce a statistical bias into RV SLAM estimators [3].

Even in the derivation of RV MH-FastSLAM (Section V-B) in which ideal detection condition 2) was relaxed, the assumption of (32) was still necessary. This assumption yields another conundrum. In (32) a detection probability (of 0 or 1) is imposed based on a given association θ . The detection probability $P_D(\mathbf{x}_k, \mathbf{m})$ in (25) depends only on the joint state $(\mathbf{x}_k, \mathbf{m})$. However, when applying (32) it is possible that a given map feature - robot pose state pair $(\mathbf{x}_k, \mathbf{m})$ can have different detection probabilities under different association hypotheses. Again, extraneous information may be included within the state, introducing a possible bias.

The results in Section VII will verify that under the conditions of low clutter and well separated map features, with respect to the robot’s trajectory, the restrictive assumptions of (31) and/or (32) used in RV SLAM formulations can be approximately satisfied, yielding good results. When these “ideal detection conditions” are violated however, as eventually occurs in most realistic situations, the statistical biases incurred under these assumptions become apparent, resulting in the divergence of RV based SLAM algorithms.

VII. EXPERIMENTS

To validate the theoretical findings of the relationship between the RFS and RV formulations through the ideal detection conditions, SLAM simulations, as well as a real experimental dataset, will be used. The SLAM problem is selected because of its generality as a combination of the navigation and mapping/tracking problems. Since a computationally feasible approach to an RFS batch estimator does not yet exist, the validation will be conducted using the Bayes filter. Without approximations, the full Bayesian estimators are intractable in both the RFS and RV frameworks. Therefore, principled realizations of both formulations will be used for comparisons. The PHD-SLAM algorithm as implemented in [33], and first introduced in [10], will be compared to the RV based FastSLAM [34]⁷ and MH-FastSLAM [36] algorithms through the use of simulations and real experimental data. These algorithms are used because of their similarity with the PHD-SLAM algorithm of [10], [33], as they are both based on the Rao-Blackwellized particle filter (PF).

To make a fair comparison, each of the tested algorithms starts with the same number (200) of particles. Both FastSLAM and MH-FastSLAM were implemented with a binary Bayes filter [37], [38] (separate from the actual SLAM filter) that accounted for detection statistics for the purpose of tracking the probability of existence of map elements. As part of the map management routine, map element estimates with log-odds of existence below -5.0 were pruned. For data association in FastSLAM, the set of correspondences that gave the highest joint measurement

⁷FastSLAM 1.0 is used because it is algorithmically more similar to Rao-Blackwellized (RB)-PHD SLAM. FastSLAM 2.0 [35] is expected to run faster with fewer particles, but show similar results as it still relies on a data association routine that is separate from the filter.

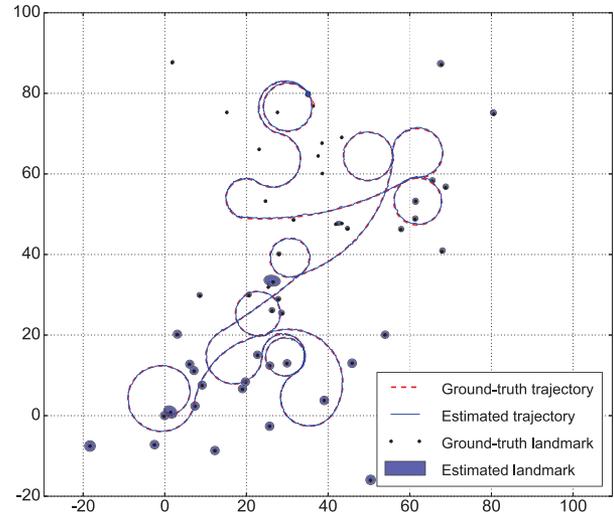
likelihood were used for each update.⁸ For computational tractability in the MH-FastSLAM algorithm, the hypotheses with the five highest joint measurement likelihoods were used [40] (i.e., each particle spawns into five particles representing different correspondences (the second summation over θ , RHS of (49)). After resampling, the particle count returned to 200. It is of interest to note that the PHD-SLAM algorithm has cubic complexity with respect to the number of targets or landmarks [41]. This is because PHD-SLAM performs a calculation involving every measurement to feature state combination, thus circumventing the data association problem. On the other hand, FastSLAM has a linear complexity on the number of targets and measurements, since it is reliant on an assumed single target to measurement association hypothesis, thus ignoring other potential association possibilities. However, for a fair comparison, the added complexity of the chosen data association algorithm used with FastSLAM should also be considered.

To measure the performance of each algorithm, the error in the vehicle trajectory estimates, measured from the particles with the highest importance weighting factors, were compared. Furthermore, the map errors related to the highest-weight particle were measured. Note that the amount of sensor noise affects the level of error, and the theoretical limit on the resulting estimate uncertainty (i.e., the Cramer-Rao lower bound). The sensor spatial noise uncertainty parameters for simulations and experiments were held constant, while the detection statistics were varied. Since the estimated and ground truth maps may differ in terms of both cardinality and spatial errors, a multi-object metric, which penalizes estimators for both types of error, is necessary. In the ensuing analyses the multi-object Optimal Sub-Pattern Assignment (OSPA) [42] and Cardinalized Optimal Linear Assignment (COLA) [43] metrics are adopted. The OSPA metric converts cardinality errors into state (e.g. spatial) errors by multiplying the cardinality error by the maximal allowed state (e.g. displacement) error c , referred to as the cut-off parameter. The combined converted cardinality and state errors therefore have units of the state error (e.g. meters). Conversely, the COLA metric, which was inspired by the OSPA metric, converts state (e.g. spatial) errors into fractional cardinality errors, by dividing the state errors by c . The combined cardinality and converted state errors therefore have the units of cardinality error (i.e. a counting error with no units). Both are true metrics, and in the case of robotic mapping, they exhibit subtle differences in their evaluations of mapping errors [43]. In a manner similar to the Mahalanobis distance, the COLA metric is dimensionless, and can be useful in assessing mapping performance when few features are deemed to correspond to any ground truth values.

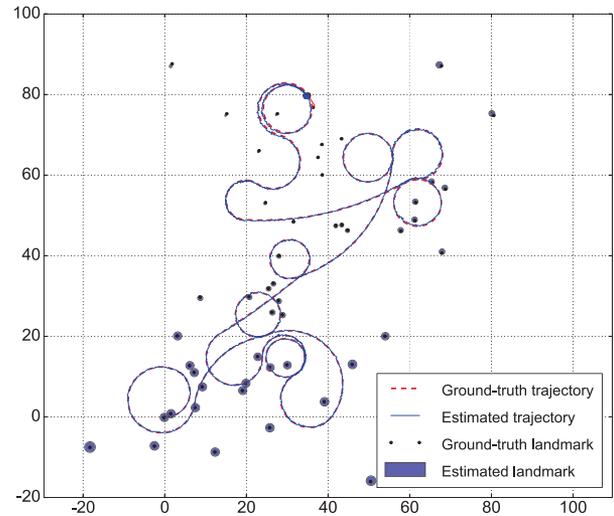
A. Simulations

Simulations are used for the initial validation of the theoretical results since detection statistics can be accurately controlled

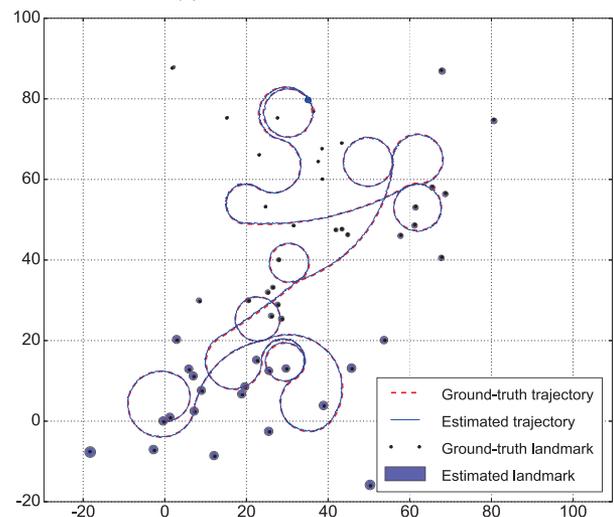
⁸Since there is no uncertainty associated with each particle's pose, the determined correspondences by maximum joint likelihood using algorithms such as the Hungarian method [39] should be similar to those produced by other data association methods that account for the joint likelihood, such as the joint compatibility branch and bound (JCBB) algorithm [30].



(a) RB-PHD-SLAM estimates



(b) MH-FastSLAM estimates



(c) FastSLAM estimates

Fig. 1. The map and trajectory estimates under close-to-ideal conditions ($P_D = 0.99$, $\kappa = 0.000001 \text{ m}^{-2}$).

and ground truth is available. In a simulated 2-D space, the vehicle traversed the environment shown in Fig. 1(a), where the black dots are the locations of map elements, and the red dashed line is the true vehicle trajectory, starting at $(0, 0)$, and ending near $(35, 80)$. A range-bearing sensor is simulated such that map elements with distances of 5 m to 25 m in any direction, may be detected according to the probability of detection. Independent of the real detections, false measurements were added according to the clutter intensity. The number of clutter measurements was Poisson distributed, while the clutter intensity was uniformly distributed over the measurement space. Spatial noise was injected into all map element estimates at each time-step to promote estimate convergence [44] as well as into the range measurements themselves. Over 2000 trials were conducted to obtain results over a wide range of detection probabilities and clutter intensities.

Under a high probability of detection and low clutter intensity ($P_D = 0.99$, $\kappa = 0.000001 \text{ m}^{-2}$, which equates to 0.0019 expected false measurements per time step), it was postulated that PHD-SLAM would perform very similarly to FastSLAM and MH-FastSLAM. Their trajectory estimates are shown in Fig. 1. Visually, the estimates appear the same. Fig. 3(a)–(c) show the Euclidean position and orientation errors of the highest-weighted particle in each of the three algorithms in comparison to ground truth. The order of magnitude of the errors are similar in all three cases. For the map estimate, Fig. 4(a) shows the OSPA errors for the three SLAM algorithms, based on a cut-off parameter $c = 10.0$ m and power $p = 1$ [42], along with the cardinality estimate in Fig. 4(c). The estimated number of map elements produced by PHD-SLAM does not follow the actual number of map elements as closely as FastSLAM and MH-FastSLAM. This small discrepancy is caused by map elements that are close to the sensing boundary, where spatial uncertainty causes the probability of detection to be incorrectly estimated, thus leading to an error in the cardinality estimate, contributing to a larger OSPA error. In general, mismatches in the detection model parameters are known to result in biased cardinality estimates in RFS filtering [45]. Overall however, the three methods performed similarly, which verifies the hypothesis for close-to-ideal detection conditions.

Under non-ideal detection conditions of low detection probabilities and high clutter intensity ($P_D = 0.50$, $\kappa = 0.005000 \text{ m}^{-2}$, which equates to 9.45 expected false measurements per time step), estimate divergence is expected for FastSLAM and MH-FastSLAM due to their inherent assumptions on detection statistics and difficulties in data association. In comparison, PHD-SLAM should be able to make use of the detection statistics and perform more robustly. Fig. 2 shows the map and trajectory estimates. Similar to the close-to-ideal case, PHD-SLAM was able to produce a consistent map, with a trajectory estimate that followed closely to the ground truth trajectory. On the contrary, the map and trajectory estimates produced by FastSLAM diverged as evident in Fig. 2(c). Even for MH-FastSLAM, which allows for multiple data association hypotheses, the clutter was too great and, due to the conundrum of assuming (32), the estimate in Fig. 2(b) also diverged. This is further reflected in the trajectory errors in Fig. 3(d)–(f).

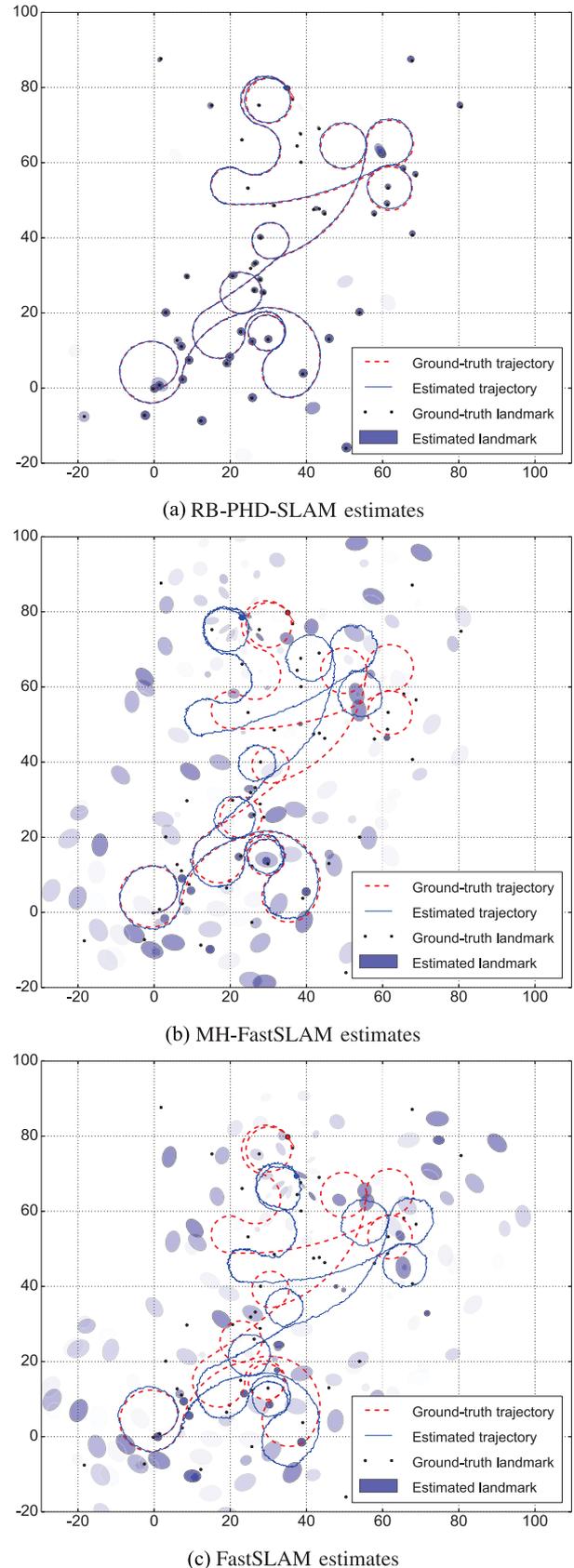


Fig. 2. The map and trajectory estimates under non-ideal conditions ($P_D = 0.50$, $\kappa = 0.005000 \text{ m}^{-2}$). An ellipse with high opacity corresponds to a Gaussian weight / probability of existence close to 1.

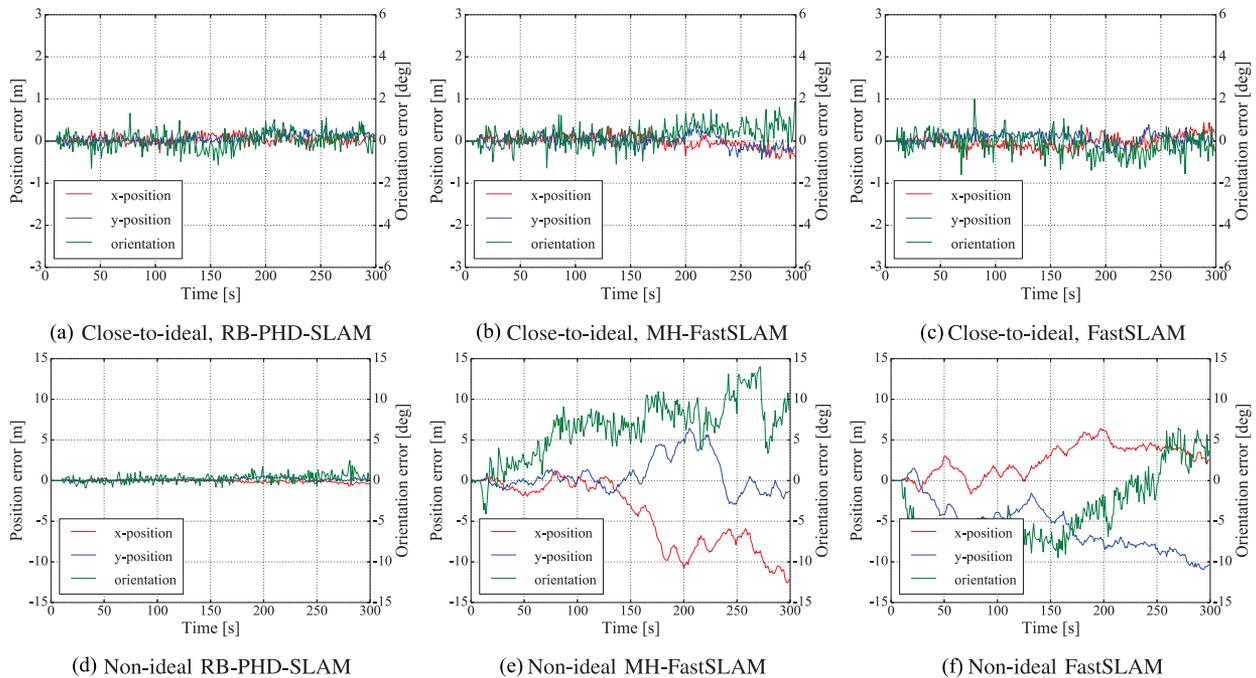
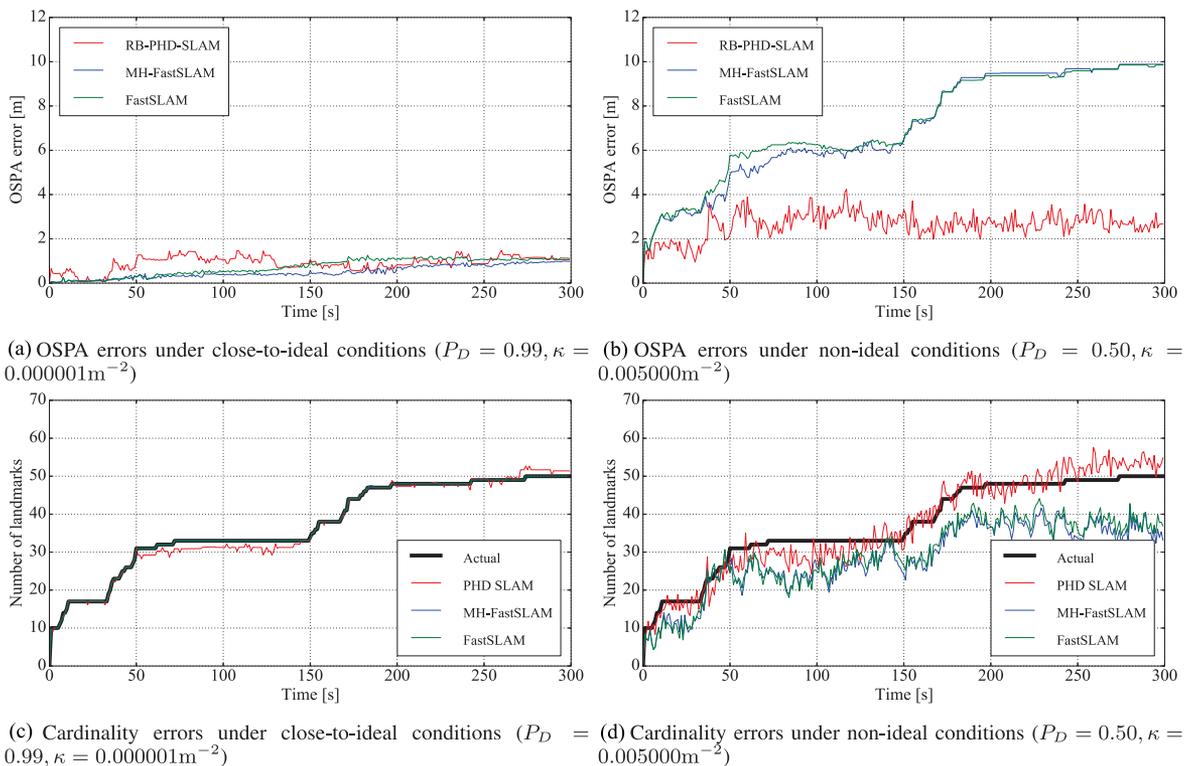
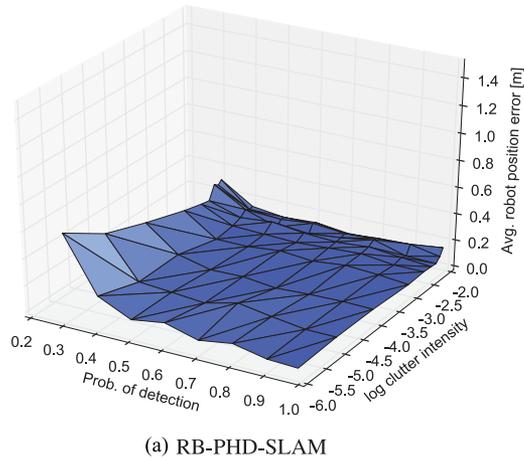


Fig. 3. Robot trajectory Euclidean and orientation errors.

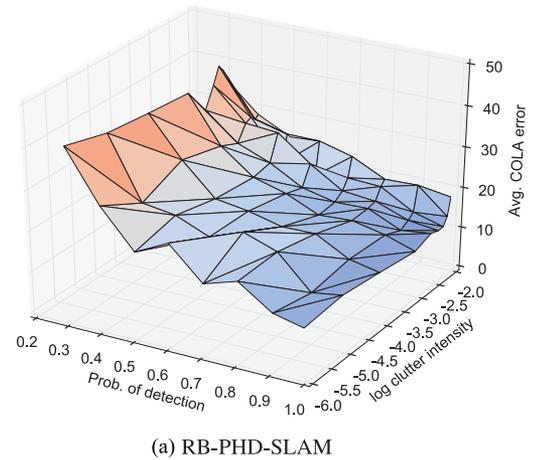
Fig. 4. OSPA errors (Euclidean distance with cutoff parameter $c = 10.0$ m and $p = 1$) and cardinality errors for the map.

The poor map estimates produced by FastSLAM, and MH-FastSLAM are reflected in the OSPA errors and the estimated number of map elements shown in Fig. 4(b) and (d), respectively. Both FastSLAM, and MH-FastSLAM underestimated the number of elements in the map. One reason for this is that the binary filter used for tracking probabilities of existence only

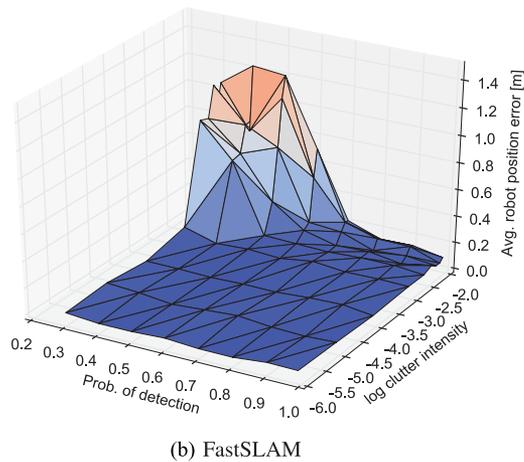
accounts for the average number of false measurements by approximating the probability of false alarm from this average. Since clutter is simulated as Poisson distributed, a high average number of false measurements also implies a high variance in that number, which is unaccounted for in the binary filter.



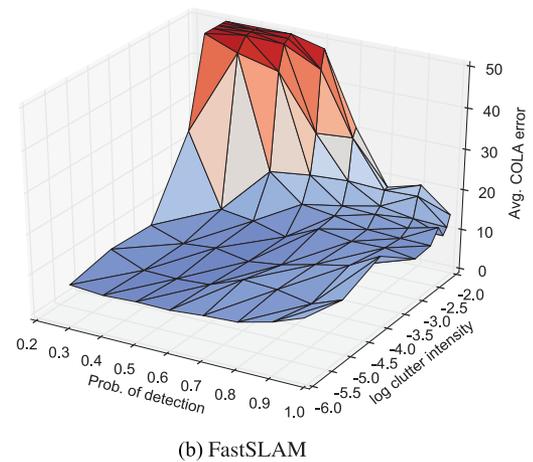
(a) RB-PHD-SLAM



(a) RB-PHD-SLAM



(b) FastSLAM



(b) FastSLAM

Fig. 5. Averaged robot position estimate error at the end of simulation trials for varying values of probability of detection and clutter intensity.

To provide a more comprehensive oversight on how detection statistics affect both RFS and RV methods, the averaged COLA errors at the end of approximately 2000 simulation trials were obtained for various probabilities of detection (from 0.2 to 0.99) and clutter intensities (from 10^{-6} m^{-2} to 10^{-2} m^{-2} , corresponding to an average of 0.0019 and 18.9 false alarms per time-step, respectively). For a stricter penalization of spatial errors, the parameters of the COLA metric were $c = 2.0 \text{ m}$ and $p = 1$ [43]. The COLA metric was used in this case, because with these parameters, many of the estimated features under the low probability of detection and/or high clutter scenarios would not be assigned to ground truth values, thus resulting in multiple OSPA metric values very close to its limiting value c [43]. Each detection statistical setting was repeated 10 times. Fig. 5 shows the averaged Euclidean robot positional errors. Similar to the previous results from individual trials, the estimated errors increased for low probabilities of detection and high clutter intensities for RV FastSLAM. For PHD SLAM, the vehicle positional error only increased slightly as the probability of detection decreased. In terms of the map estimate, Fig. 6 shows that the mapping errors from the FastSLAM estimates are again high for non-ideal detection conditions. The map estimation error for RB-PHD SLAM also increased as the probability of detection

Fig. 6. Averaged map element COLA errors ($c = 2.0 \text{ m}$, $p = 1$) at the end of simulation trials for varying values of probability of detection and clutter intensity.

decreased. For probability of detection values of approximately 0.3 and below, the PHD filter appears more fragile and tends to underestimate the cardinality (by over-correcting the probability of existence of a map element when it is not detected). However, good spatial estimates of map elements are maintained. This is why the map error increases with low probability of detection, and vehicle positional error increases slightly, but remains relatively low.

B. Victoria Park Dataset

To further validate the hypotheses on the performance of the various SLAM algorithms under different detection conditions, the Victoria Park dataset [46] was used to provide a more realistic scenario for making comparisons.

Suitable parameters for the spatial noise associated with the measurements had to be determined through trial and error. Similarly, the dataset contains a small amount of clutter, but the exact detection statistics are unknown, and their parameters had to be determined by trial and error.⁹ The results in Fig. 7 show

⁹Alternatively, methods exist for estimating detection statistics such as [45].

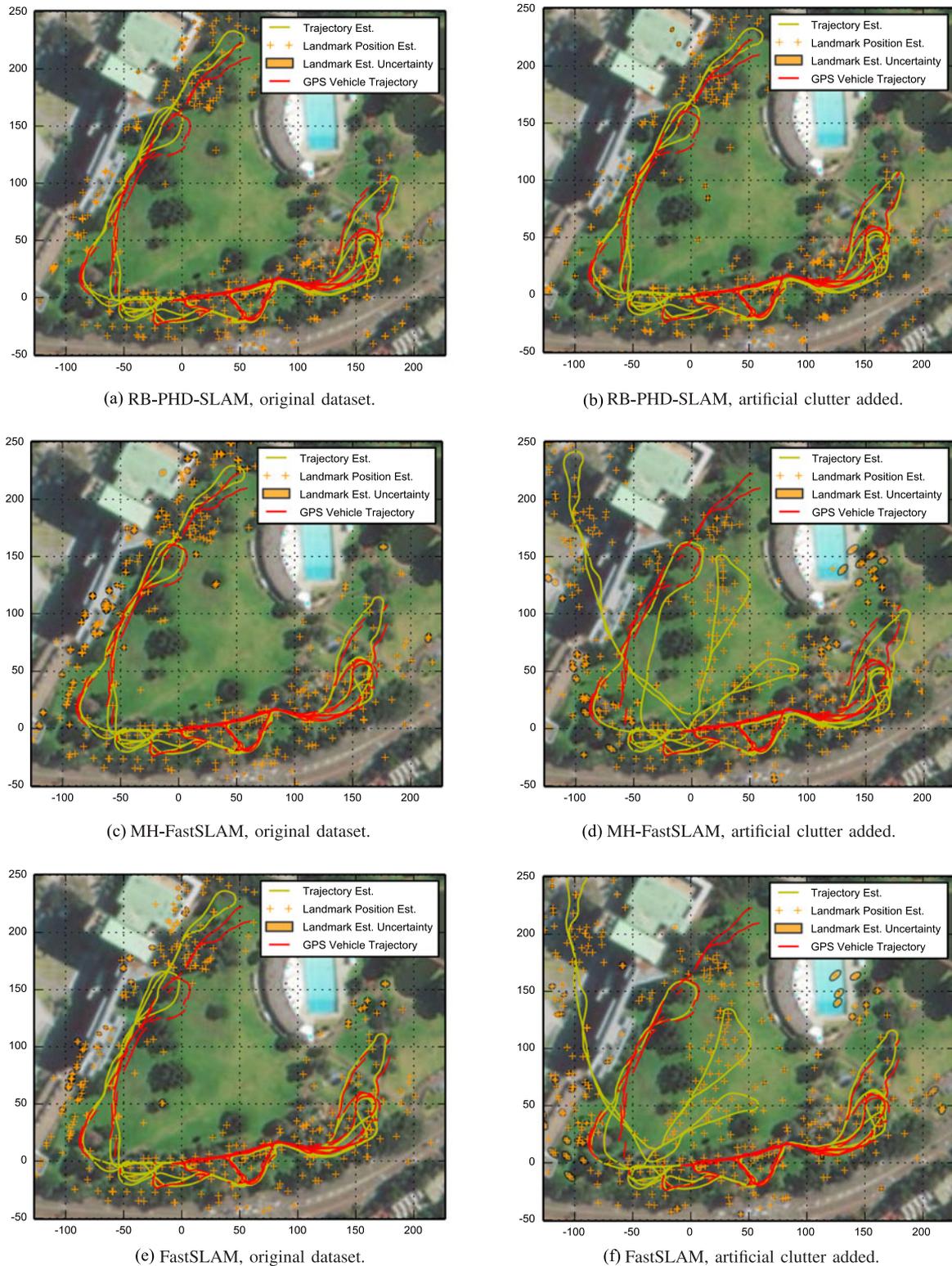


Fig. 7. Results from processing the Victoria Park dataset.

the estimates produced by the three SLAM algorithms. The figures on the left are the results from processing the original dataset, which are very similar to each other, and are consistent with the results in [34], [35]. To make the detection conditions less ideal, artificial clutter was added to the dataset. The number

of false measurements at each measurement time followed a Poisson distribution, and were uniformly distributed within the sensor field of view. Hence, on top of the 52974 measurements in the dataset, approximately 21500 artificial clutter measurements were added.

With the additional clutter, Fig. 7(b) shows that RB-PHD-SLAM performed similarly to the original dataset scenario but that FastSLAM (Fig. 7(f)) and MH-FastSLAM (Fig. 7(d)) diverged. It is worth noting that the estimate divergences occurred in parts of the trajectory with a low number of real detections, such as the region near coordinates $(-50, 50)$. The low ratio of real versus clutter measurements caused the accompanying data association routines to incorrectly associate clutter measurements to estimates of real map elements, resulting in biased estimates. Overall, the results are consistent with those produced by simulations and validate the theoretical findings.

VIII. CONCLUSION

The formulation of navigation, mapping, and tracking problems using the RFS framework can be related to the RV formulation, only through the application of ideal detection conditions, which assume no clutter, a single and correct permutation of data association and a known map cardinality. In contrast, the RFS approach considers all possible permutations of data association while accounting for detection statistics, and unknown map cardinality. This is true not only for Bayesian filtering, but also ML batch estimation. A feasible solution to RFS-based batch estimation is however still an open problem and therefore the findings were validated using Bayesian filtering approaches.

The RFS based PHD-SLAM algorithm was shown to perform similarly to the RV based FastSLAM and MH-FastSLAM algorithms under close-to-ideal detection conditions. When conditions became non-ideal with low probabilities of detection and high clutter, PHD-SLAM continued to produce estimates with lower errors, whereas the estimates from the RV approaches diverged. This is due to their implicit assumption of ideal detection conditions, and the possible bias caused by the utilization of a single data association hypothesis.

It should also be noted that the PHD filter is the simplest approximation of an RFS Bayes filter. More advanced RFS filters, such as the CPHD and the MB filters, are expected to yield improved results. In particular, the LMB filter has been implemented for solving the SLAM problem in [17]. Through the understanding obtained in this article, the simpler RV formulation can be successful when detection conditions are close to ideal, where data association is unlikely to fail. However, in realistic cases of feature misdetections and high clutter, the more complex RFS framework provides a more robust approach.

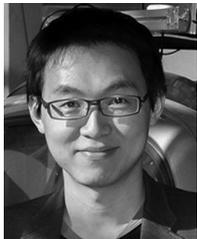
ACKNOWLEDGMENT

The robot used in the experiments is from Clearpath Robotics.

REFERENCES

- [1] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: Part I," *IEEE Robot. Autom. Mag.*, vol. 13, no. 2, pp. 99–110, Jun. 2006.
- [2] K. Y. K. Leung, F. Inostroza, and M. Adams, "Generalizing random-vector SLAM with random finite sets," in *Proc. IEEE Int. Conf. Robot. Autom.*, Seattle, WA, USA, May 2015, pp. 4583–4588.
- [3] R. P. S. Mahler, *Statistical Multisource-Multitarget Information Fusion*, vol. 685. Boston, MA, USA: Artech House, 2007.
- [4] D. B. Reid, "An algorithm for tracking multiple targets," *IEEE Trans. Autom. Control*, vol. AC-24, no. 6, pp. 843–854, Dec. 1979.
- [5] R. L. Streit and T. E. Luginbuhl, "Maximum likelihood method for probabilistic multihypothesis tracking," *Proc. SPIE*, vol. 2235, pp. 394–405, 1994.
- [6] J. Dezert and Y. Bar-Shalom, "Joint probabilistic data association for autonomous navigation," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 29, no. 4, pp. 1275–1286, Oct. 1993.
- [7] T. E. Fortmann, Y. Bar-Shalom, and M. Scheffe, "Sonar tracking of multiple targets using joint probabilistic data association," *IEEE J. Ocean. Eng.*, vol. OE-8, no. 3, pp. 173–184, Jul. 1983.
- [8] R. P. S. Mahler, "Multitarget bayes filtering via first-order multitarget moments," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 39, no. 4, pp. 1152–1178, Oct. 2003.
- [9] 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, Nov. 2006.
- [10] J. Mullane, B.-N. Vo, M. D. Adams, and B.-T. Vo, "A random-finite-set approach to Bayesian SLAM," *IEEE Trans. Robot.*, vol. 27, no. 2, pp. 268–282, Apr. 2011.
- [11] A. Swain and D. Clark, "The single-group PHD filter: An analytic solution," in *Proc. Int. Conf. Inf. Fusion*, 2011, pp. 1–8.
- [12] C. S. Lee, D. Clark, and J. Salvi, "SLAM with dynamic targets via single-cluster PHD filtering," *IEEE J. Sel. Top. Signal Process.*, vol. 7, no. 3, pp. 543–552, Jun. 2013.
- [13] B.-T. Vo, B.-N. Vo, and A. Cantoni, "Analytic implementations of the cardinalized probability hypothesis density filter," *IEEE Trans. Signal Process.*, vol. 55, no. 7, pp. 3553–3567, Jul. 2007.
- [14] B.-T. Vo, B.-N. Vo, and A. Cantoni, "The cardinality balanced multi-target multi-bernoulli filter and its implementations," *IEEE Trans. Signal Process.*, vol. 57, no. 2, pp. 409–423, Feb. 2009.
- [15] B.-T. Vo and B.-N. Vo, "Labeled random finite sets and multi-object conjugate priors," *IEEE Trans. Signal Process.*, vol. 61, no. 13, pp. 3460–3475, Jul. 2013.
- [16] B.-N. Vo, B.-T. Vo, and D. Phung, "Labelled random finite sets and the Bayes multi-target tracking filter," *IEEE Trans. Signal Process.*, vol. 62, no. 24, pp. 6554–6567, Dec. 2014.
- [17] S. Reuter, B.-T. Vo, B.-N. Vo, and K. Dietmayer, "The labeled multi-Bernoulli filter," *IEEE Trans. Signal Process.*, vol. 62, no. 12, pp. 3246–3260, Jun. 2014.
- [18] H. Deusch, S. Reuter, and K. Dietmayer, "The labeled multi-Bernoulli SLAM filter," *IEEE Signal Process. Lett.*, vol. 22, no. 10, pp. 1561–1565, Oct. 2015.
- [19] J. Folkesson and H. Christensen, "Graphical—A self-correcting map," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2004, pp. 383–390.
- [20] S. Thrun and M. Montemerlo, "The GraphSLAM algorithm with applications to large-scale mapping of urban structures," *Int. J. Robot. Res.*, vol. 25, nos. 5/6, pp. 403–429, 2006.
- [21] M. Kaess, A. Ranganathan, and F. Dellaert, "iSAM: Incremental smoothing and mapping," *IEEE Trans. Robot.*, vol. 24, no. 6, pp. 1365–1378, Dec. 2008.
- [22] H. Strasdat, J. M. M. Montiel, and A. J. Davison, "Visual SLAM: Why filter?" *Image Vis. Comput.*, vol. 30, no. 2, pp. 65–77, 2012.
- [23] P. Agarwal, G. D. Tipaldi, L. Spinello, C. Stachniss, and W. Burgard, "Robust map optimization using dynamic covariance scaling," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2013, pp. 62–69.
- [24] D. Clark, "First-moment multi-object forward-backward smoothing," in *Proc. Int. Conf. Inf. Fusion*, 2010, pp. 1–6.
- [25] B.-T. Vo, D. Clark, B.-N. Vo, and B. Ristic, "Bernoulli forward-backward smoothing for joint target detection and tracking," *IEEE Trans. Signal Process.*, vol. 59, no. 9, pp. 4473–4477, Sep. 2011.
- [26] R. Mahler, B.-N. Vo, and B.-T. Vo, "The forward-backward probability hypothesis density smoother," in *Proc. Int. Conf. Inf. Fusion*, 2010, pp. 1–8.
- [27] G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard, "g2o: A general framework for graph optimization," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2011, pp. 3607–3613.
- [28] S. Särkkä, *Bayesian Filtering and Smoothing*. Cambridge, U.K.: Cambridge Univ. Press, 2013.
- [29] I. J. Cox, "A review of statistical data association techniques for motion correspondence," *Int. J. Comput. Vis.*, vol. 10, no. 1, pp. 53–66, 1993.
- [30] J. Neira and J. D. Tardós, "Data association in stochastic mapping using the joint compatibility test," *IEEE Trans. Robot. Autom.*, vol. 17, no. 6, pp. 890–897, Dec. 2001.
- [31] B. N. Vo, S. Singh, and A. Doucet, "Sequential monte carlo implementation of the PHD filter for multi-target tracking," in *Proc. Int. Conf. Inf. Fusion*, 2003, pp. 1561–1565.

- [32] N. Sünderhauf and P. Protzel, "Switchable constraints for robust pose GraphSLAM," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2012, pp. 1879–1884.
- [33] K. Y. K. Leung, F. Inostroza, and M. Adams, "Evaluating set measurement likelihoods in random-finite-set SLAM," in *Proc. Int. Conf. Inf. Fusion*, Salamanca, Spain, Jul. 2014, pp. 1–8.
- [34] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit, "FastSLAM: A factored solution to the simultaneous localization and mapping problem," in *Proc. 18th Nat. Conf. Artif. Intell.*, 2002, pp. 593–598.
- [35] 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 *Proc. 18th Int. Joint Conf. Artif. Intell.*, 2003, pp. 1151–1156.
- [36] J. Nieto, J. Guivant, E. Nebot, and S. Thrun, "Real time data association for fastSLAM," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2003, vol. 1, pp. 412–418.
- [37] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. vol. 1. Cambridge, MA, USA: MIT Press, 2005.
- [38] K. Y. K. Leung, F. Inostroza, and M. Adams, "An improved weighting strategy for rao-blackwellized probability hypothesis density simultaneous localization and mapping," in *Proc. IEEE Int. Conf. Control, Autom. Inf. Sci.*, 2013, pp. 103–110.
- [39] H. W. Kuhn, "The Hungarian method for the assignment problem," *Naval Res. Logist. Q.*, vol. 2, nos. 1/2, pp. 83–97, 1955.
- [40] K. G. Murty, "An algorithm for ranking all the assignments in order of increasing cost," *Oper. Res.*, vol. 16, no. 3, pp. 682–687, 1968.
- [41] K. Y. K. Leung, F. Inostroza, and M. Adams, "Multifeature-based importance weighting for the PHD SLAM filter," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 52, no. 6, pp. 2697–2714, Dec. 2016.
- [42] D. Schuhmacher, B.-T. Vo, and B.-N. Vo, "A consistent metric for performance evaluation of multi-object filters," *IEEE Trans. Signal Process.*, vol. 56, no. 8, pp. 3447–3457, Aug. 2008.
- [43] P. Barrios, M. Adams, K. Y. K. Leung, F. Inostroza, G. Naqvi, and M. E. Orchard, "Metrics for evaluating feature-based mapping performance," *IEEE Trans. Robot.*, vol. 33, no. 1, pp. 198–213, Feb. 2017.
- [44] D. Simon, *Optimal State Estimation*. Hoboken, NJ, USA: Wiley, 2006.
- [45] B.-T. Vo, B.-N. Vo, R. Hoseinnezhad, and R. P. S. Mahler, "Robust multi-Bernoulli filtering," *IEEE J. Sel. Top. Signal Process.*, vol. 7, no. 3, pp. 399–409, Jun. 2013.
- [46] J. Guivant, E. Nebot, and S. Baiker, "Autonomous navigation and map building using laser range sensors in outdoor applications," *J. Robot. Syst.*, vol. 17, no. 10, pp. 565–583, 2000.



Keith Y. K. Leung (M'06) received the B.A.Sc. and M.A.Sc. degrees from the University of Waterloo, Waterloo, ON, Canada, both in mechanical engineering, in 2005 and 2007, respectively, and the Ph.D. degree from the University of Toronto, Toronto, ON, in 2012. He was a member of the Autonomous Space Robotics Lab (ASRL), University of Toronto, and the Flight Systems Control (FSC) Lab, Institute for Aerospace Studies (UTIAS). From 2012 to 2016, he was a Postdoctoral Researcher at the Advanced Mining Technology Center (AMTC) and Department of

Electrical Engineering, Universidad de Chile. His area of expertise is in state estimation, robot navigation, as well as simultaneous localization and mapping (SLAM). He is currently a vision processing, perception, and robotics expert with Applix (Trimble).



Felipe Inostroza (S'11) received the degree in engineering science and the Master's degree in electrical engineering from Universidad de Chile, Santiago, Chile, in 2013 and 2015, respectively. He is currently working toward the doctoral degree in the Department of Electrical Engineering, Universidad de Chile. His research interests include robotics in general, and simultaneous localization and mapping (SLAM) in particular. He is currently studying the effects of including detection statistics into the SLAM problem.



Martin Adams (SM'08) received the degree in engineering science from the University of Oxford, Oxford, U.K, in 1988, and the D.Phil. degree from the Robotics Research Group, University of Oxford, in 1992. He is a Professor of Electrical Engineering in the Department of Electrical Engineering, Universidad de Chile, Santiago, Chile. He is also a Principle Investigator in the Advanced Mining Technology Centre (AMTC). He continued his research in autonomous robot navigation as a Project Leader and Part Time Lecturer with the Institute of Robotics,

Swiss Federal Institute of Technology (ETH), Zurich, Switzerland. He was employed as a Guest Professor and taught control theory in St. Gallen, Switzerland, from 1994 to 1995. From 1996 to 2000, he served as a Senior Research Scientist in robotics and control, in the field of semiconductor assembly automation, at the European Semiconductor Equipment Centre (ESEC), Switzerland. From 2000 to 2010, he was an Associate Professor in the School of Electrical and Electronic Engineering, Nanyang Technological University (NTU), Singapore. His research work focuses on autonomous robot navigation, space situational awareness, sensing, sensor data interpretation and control, and he has published many technical papers in these fields. He has been the Principle Investigator and Leader of many robotics and industrially sponsored projects, coordinating researchers from local industries and local and overseas universities and has served as an Associate Editor on various journal and conference editorial boards.