Department of Electrical
and
Computer Systems Engineering

Technical Report
MECSE-4-2007

A Review of Robotic SLAM

D. J. Spero and R. A. Jarvis

A Review of Robotic SLAM

Dorian J. Spero and Ray A. Jarvis
Intelligent Robotics Research Centre
Monash University,
Clayton, Victoria 3800, Australia
Email: dorian@ieee.org

Abstract

Since its inception in the late 1980s, the process of simultaneous localisation and map building (SLAM) has become a key subject of discourse amongst the robotics community. Many consider it to be instrumental to autonomous mobile robot navigation in an *a priori* unknown environment, especially when planning efficient and purposeful trajectories. But due to data association uncertainty, navigation error (e.g., odometric drift) and sensor noise, SLAM has proven to be a complex problem. This paper first describes the SLAM problem and then reviews the current state of the art in solving it with regard to real-world operation.

KEYWORDS: Mobile Robots; SLAM; EKF; FastSLAM; Scan Matching; The Kidnapped Way.

1 Introduction

Simultaneous localisation and map building (SLAM) is the dual process of building a map of the environment, comprising landmarks and possibly other features (obstacles, topography, etc.), and using this map to ascertain the robot's absolute pose. The robot starts at an unknown location in an *a priori* unknown environment. It then uses its onboard sensors to observe the local landmarks, and from this information, computes its own pose while simultaneously estimating the locations of these landmarks. As the robot moves through the environment, its changing observational viewpoint enables the incremental building of a complete map of landmarks, which are continuously exploited to track the robot's current pose relative to its initial pose.

In mathematical terms, the objective of the SLAM process is to estimate the system state $\mathbf{x}_k$ at discrete time instant k, given by

$$\mathbf{x}_k = \begin{bmatrix} \mathbf{x}_{r_k} \\ \mathbf{x}_1 \\ \vdots \\ \mathbf{x}_n \end{bmatrix} \tag{1}$$

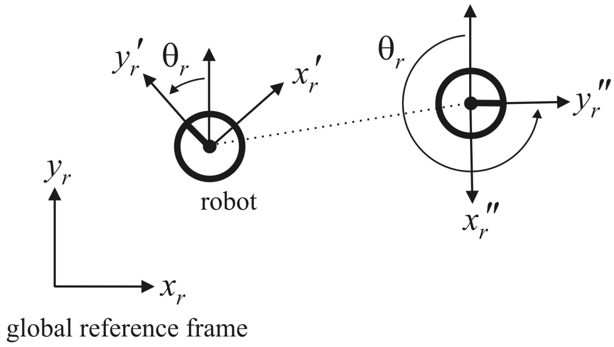

Figure 1: Robot Coordinate System

where $\mathbf{x}_{r_k}$ is the robot's state and the set $M = \{\mathbf{x}_i \,|\, 1 \leq i \leq n\}$ represents the map of observed landmarks. The landmark states $\mathbf{x}_i$, not given as a function of time here, are generally assumed to be stationary; so moving environmental features are treated as unwanted noise. Moving features, however, can still be useful to the SLAM process if their dynamism is predictable (e.g., the sun with respect to Earth), intermittent (e.g., vehicles that are temporarily parked) or negligible within a sparse environment.

For a 2D Cartesian based map, the robot's state can be defined by its pose (position and orientation) in space

$$\mathbf{x}_{r_k} = \begin{bmatrix} x_{r_k} \\ y_{r_k} \\ \theta_{r_k} \end{bmatrix} \tag{2}$$

relative to a global reference frame, as shown in Figure 1. The landmarks in the map M are commonly represented as points in space and therefore their states may be defined by

$$\mathbf{x}_i = \begin{bmatrix} x_i \\ y_i \end{bmatrix}. \tag{3}$$

However, equations (2) and (3) vary according to the robot's intended application and the particular SLAM strategy used.

Now that a definition of SLAM has been given, what makes the practical operation of SLAM a problem? The answer to this question, as discussed in [1], lies in the difficulty of coping with three distinct forms of uncertainty:

1. Data association uncertainty
2. Navigation error
3. Sensor noise

Data association uncertainty occurs because of the robot's inability to properly identify a landmark perceived from different poses as the same. It is therefore possible to wrongly associate landmarks, thereby corrupting the map. This problem is generally referred to as the *data association problem* or *correspondence problem* [2, 3]. Navigation error is caused by the inevitable divergence between the robot's assumed motion, via the vehicle model, and its actual motion. This divergence can lead to an accumulative error in the robot's estimated pose as well as exacerbate the data association problem. The final form of uncertainty is the result of imperfect sensing devices, the measurements of which are inherently noisy and sometimes completely erroneous.

Together, these uncertainties culminate into a complex SLAM problem, one which some behaviourists believe is not worth solving [4]. Their argument is that humans and animals can navigate perfectly well without precise quantitative knowledge of their location; so why should their mechanical counterparts need to perform SLAM? A common rebuttal to this argument is that there is a range of useful applications, such as cross-country and interplanetary exploration, undersea navigation and mining, where the robot needs to track its precise position over a long-term period without the aid of an *a priori* map or artificial infrastructure [5]. By using SLAM, the robot is able to navigate efficiently and purposefully within its *a priori* unknown environment, while strategically carrying out its mission. In fact, some researchers go beyond this conservative view by stilting the solution to the SLAM problem as the cornerstone or "Holy Grail" of robot autonomy [6]. For these reasons, the SLAM problem has received a considerable amount of research attention, and judging by its dominant discourse at international conferences, the number of active researchers in this area is growing rapidly.

A number of methods have been proposed to solve the SLAM problem, each with relative strengths and practical limitations. This paper provides a review of these methods and delves into the inner workings of some of the more notable cases. Since all the relevant works cannot be cited here for reasons of brevity, a representative sample will be used to convey the current state of the art. Additionally, the various assumptions and contrivances adopted, which are often hidden under a shroud of mathematical rigour in the literature, will be examined in relation to their impact on real-world operation.

The various methods will be compared on the basis of several key properties. These properties may include, for example, the map representation (e.g. Cartesian landmark locations, occupancy grid, or polygons); the representation of uncertainty in the map (e.g. Gaussian / mixture of Gaussians, maximum likelihood, or a particle set); restrictions on sensor noise; optimality of convergence; computational complexity; accuracy (with respect to ground truth); generality / applicability; robustness to unmodeled events (e.g. large non-systematic motion errors, dynamic environmental features, or incorrect data association); and whether the map is incrementally built, and if so, the consistency of the resultant map.

2 Review of SLAM Methods

This section begins with what is currently the most popular approach: the *estimation-theoretic approach*. It will be used later as a benchmark for comparison between the other approaches.

2.1 The Estimation-Theoretic Approach

The *estimation-theoretic* or *extended Kalman filter (EKF) based approach* was first introduced by Smith, Self and Cheeseman in their seminal paper [7], which described the use of an EKF [8] to build a stochastic map of spatial relationships. This work was extended shortly after by Moutarlier and Chatila [9], who considered the correlated noise between landmarks in the map to preserve the filter's consistency. Leonard and Durrant-Whyte [10][1] then implemented it using an indoor mobile robot equipped with sonar sensors. Since then, a considerable amount of progress has been made in the development of the EKF based approach, including such contributions as its application to different domains [11, 12]; the use of various sensors [13, 14]; proofs of its convergence properties [6] or lack thereof [15]; and methods that somewhat address its high computational complexity of $\mathcal{O}(n^2)$ [16, 17, 18, 19, 20].

The mathematical framework of the EKF is based on a state space representation of the robot and its environment. In presenting the mathematical framework here, the system state vector $\mathbf{x}_k$ given in Section 1 will be used and several models will be introduced. The first model, called the system plant model, describes how the system states change as a function of time k and is conventionally written as a non-linear state transition equation of the form

$$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_k) + \mathbf{v}_k \tag{4}$$

where $\mathbf{u}_k$ represents the control input asserted in the time interval $(t_{k-1}, t_k]$, $\mathbf{v}_k$ denotes temporally uncorrelated Gaussian noise with zero mean $(E[\mathbf{v}_k] = 0, \forall k)$ and covariance $\mathbf{Q}_k$, and $\mathbf{f}(\cdot,\cdot)$ is a non-linear function that maps $\mathbf{x}_{k-1}$ to $\mathbf{x}_k$ given $\mathbf{u}_k$. Similarly, a robot or vehicle model is used to capture the robot's progression from its previous state, $\mathbf{x}_{r_{k-1}}$, to the next, $\mathbf{x}_{r_k}$, as determined by its kinematics, and can be written as

$$\mathbf{x}_{r_k} = \mathbf{f}_r(\mathbf{x}_{r_{k-1}}, \mathbf{u}_{r_k}) + \mathbf{v}_{r_k}. \tag{5}$$

Assuming that the landmarks in the map M are stationary, the landmark model is trivially

$$\mathbf{x}_{i,\,k} = \mathbf{x}_{i,\,k-1} \tag{6}$$

and therefore the dynamics of the system is confined to the robot model. During the robot's motion, it uses an onboard sensor, or a multisensor arrangement [21], to observe the local landmarks and measure their relative positions. This is represented by an observation model where the observation at time k, denoted $\mathbf{z}_k$, is expressed in the form

$$\mathbf{z}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{w}_k \tag{7}$$

where $\mathbf{w}_k$ is a random vector of temporally uncorrelated measurement noise with zero mean $(E[\mathbf{w}_k] = 0, \forall k)$ and covariance $\mathbf{R}_k$, and $\mathbf{h}(\cdot)$ is a non-linear function that models the relationship between the observation of system states and the states themselves.

Based on the system and observation models given in (4) and (7), respectively, the EKF fuses all the available information about the system's state to

[1] The origin of the phrase "simultaneous localisation and map building".

compute a state estimate with minimum mean-squared error (MMSE). This is accomplished through a recursive, three-stage cycle consisting of *prediction*, *observation*, and *update* steps [22].

Since the EKF equations [23, 8] composing these steps have been widely published with many notational nuances, the notation used to present them here will be briefly described first. The notation $\hat{\mathbf{x}}_k^-$ will represent the *a priori* state estimate at time k or, in other words, the state prediction derived from information up to time $k-1$ (i.e., $\hat{\mathbf{x}}_k^- = \hat{\mathbf{x}}_{k\,|\,k-1}$). Conversely, $\hat{\mathbf{x}}_k^+$ will represent the *a posteriori* state estimate at time k and hence is conditioned on information up to this time (i.e., $\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_{k\,|\,k}$). Note that the '$^+$' and '$^-$' superscripts will also be used for other state variables to convey the same meaning.

2.1.1 Prediction

The first step of the filter involves generating predictions of the system's state $\hat{\mathbf{x}}_k^-$, its covariance $\mathbf{P}_k^-$, and the observation $\hat{\mathbf{z}}_k^-$ at time k. These predictions are calculated as follows:

$$\hat{\mathbf{x}}_k^- = \mathbf{f}(\hat{\mathbf{x}}_{k-1}^+, \mathbf{u}_k) \tag{8}$$

$$\hat{\mathbf{z}}_k^- = \mathbf{h}(\hat{\mathbf{x}}_k^-) \tag{9}$$

$$\mathbf{P}_k^- = \nabla\mathbf{f}_{\mathbf{x}_{k-1}} \mathbf{P}_{k-1}^+ \nabla\mathbf{f}_{\mathbf{x}_{k-1}}^T + \mathbf{Q}_k \tag{10}$$

$$\text{where} \quad \nabla\mathbf{f}_{\mathbf{x}_{k-1}} \triangleq \left.\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right|_{(\hat{\mathbf{x}}_{k-1}^+, \mathbf{u}_k)} \tag{11}$$

The Jacobian $\nabla\mathbf{f}_{x_{k-1}}$, defined in equation (11), is derived from linearising the non-linear function $\mathbf{f}$ through a first-order Taylor series expansion about the point $\hat{\mathbf{x}}_{k-1}^+$. Also, note that equation (10) does not take into account the uncertainty in the control inputs $\mathbf{u}_k$; however, this can be remedied by adding the term $\nabla\mathbf{f}_{u_k}\mathbf{U}_k\nabla\mathbf{f}_{u_k}^T$ (where $\mathbf{U}_k$ is the control covariance) to the right side of this equation.

2.1.2 Observation

After the robot makes a partial observation $\mathbf{z}_k$ of the true landmark states in $\mathbf{x}_k$, the innovation ν_k is calculated using

$$\nu_k = \mathbf{z}_k - \hat{\mathbf{z}}_k^- \tag{12}$$

under the assumption of perfect data association. The corresponding innovation covariance $\mathbf{S}_k$ is calculated as follows:

$$\mathbf{S}_k = \nabla\mathbf{h}_{\mathbf{x}_k} \mathbf{P}_k^- \nabla\mathbf{h}_{\mathbf{x}_k}^T + \mathbf{R}_k \tag{13}$$

$$\text{where} \quad \nabla\mathbf{h}_{\mathbf{x}_k} \triangleq \left.\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right|_{\hat{\mathbf{x}}_k^-} \tag{14}$$

Similar to the Jacobian $\nabla\mathbf{f}_{x_{k-1}}$ described earlier, the Jacobian $\nabla\mathbf{h}_{\mathbf{x}_k}$ is a linearisation of the observation function $\mathbf{h}$.

2.1.3 Update

The final step involves updating the state estimate $\hat{\mathbf{x}}_k^+$ and its covariance $\mathbf{P}_k^+$ according to the following equations:

$$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + \mathbf{W}_k \nu_k \tag{15}$$

$$\mathbf{P}_k^+ = \mathbf{P}_k^- - \mathbf{W}_k \mathbf{S}_k \mathbf{W}_k^T \tag{16}$$

where the Kalman gain $\mathbf{W}_k$ is given by

$$\mathbf{W}_k = \mathbf{P}_k^- \nabla \mathbf{h}_{\mathbf{x}_k}^T \mathbf{S}_k^{-1} \tag{17}$$

Overall, this filter provides a theoretically sound solution to SLAM and a means of systematically studying its convergence properties, the evolution of the map, and the propagation of positional uncertainties. However, from a practical standpoint, there are several issues that adversely affect its applicability. To begin with, the approximation errors caused by linearising the system and measurement functions can lead to filter instability and an inconsistent map [15, 24], especially if the time step interval Δt_k (where $\Delta t_k = t_k - t_{k-1}$) is not sufficiently small. Julier and Uhlmann partially solved this problem by introducing the unscented Kalman filter (UKF) [25], which tends to be more suited to highly non-linear functions than the EKF. However, both of these extensions to the standard Kalman filter are still limited by their inherent assumptions, such as Gaussianity and independence of model errors, which realistically may not hold true.

Another limitation of the EKF is that landmarks need to be uniquely identified by the data association process. For instance, it is not enough to just be able to recognise that a certain percept is a tree; the tree has to be matched to its corresponding landmark state in the map. Since data association is commonly performed using the *gated nearest-neighbour (NN) algorithm* [2], this type of identification becomes increasingly less reliable as environmental clutter or uncertainty in the robot's estimated state $\hat{\mathbf{x}}_k^-$ grows. This can cause false data associations, which then often lead to catastrophic failure [22]. The likelihood of this happening can be reduced by applying a more robust data association technique such as the *joint compatibility test* [26] or the *graph theoretic approach* [27]. Also, ambiguous observation data can be better handled using *multiple hypothesis tracking (MHT)* [28, 29], which maintains each possible interpretation of the data over time using multiple, probabilistically weighted, maps. These enhancements, however, add to the computational complexity of the EKF.

Lastly, the biggest problem with the EKF is arguably its reliance on stringent models to satisfy its predictive behaviour. This reliance means that the operational performance of the EKF is extremely specific to the extent to which the robot and its environment are predisposed to the modeling process. Consequently, the robot designs and environments that cannot be easily modeled or manipulated are often avoided, and those that can are tightly bounded with little tolerance for the unknown. There are other probabilistic approaches to SLAM that are not as rigid.

2.2 Other Probabilistic Approaches

To begin to describe these other approaches, it is worthwhile to look at the SLAM problem from a probabilistic point of view. The SLAM problem in this context is considered to be a density estimation problem where the solution involves finding the joint posterior probability of the robot's pose $\mathbf{x}_{r_k}$ and map M at time k. This posterior can be written as

$$p(\mathbf{x}_{r_k}, M \,|\, \mathbf{z}_{0:k}, \mathbf{u}_{0:k}) \tag{18}$$

where $\mathbf{z}_{0:k}$ and $\mathbf{u}_{0:k}$ represent the observation and control history, respectively. For notational convenience, this posterior will be denoted $b_k(\mathbf{x}_{r_k}, M)$ from this point on, and correspondingly referred to as the robot's *belief state* at time k.

The probabilistic SLAM approaches, including the EKF, predominantly estimate the belief $b_k(\mathbf{x}_{r_k}, M)$ using some form of *Bayes filter* [30] (which is a temporal extension of the archetypical *Bayes rule* [31]). In doing so, they often treat the SLAM problem as a Markov process through which it is assumed that the current belief state, $b_k(\mathbf{x}_{r_k}, M)$, depends only on the immediately preceding state, $b_{k-1}(\mathbf{x}_{r_{k-1}}, M)$, independent of how the preceding state was reached. The belief probability can therefore be calculated recursively, as shown by the generic Bayes filter:

$$\begin{aligned} b_k(\mathbf{x}_{r_k}, M) = \eta \; p(\mathbf{z}_k \,|\, \mathbf{x}_{r_k}, M) \cdot \\ \int p(\mathbf{x}_{r_k} \,|\, \mathbf{x}_{r_{k-1}}, \mathbf{u}_k) \; b_{k-1}(\mathbf{x}_{r_{k-1}}, M) \; d\mathbf{x}_{r_{k-1}} \end{aligned} \tag{19}$$

where η is a normalisation constant, $p(\mathbf{z}_k \,|\, \mathbf{x}_{r_k}, M)$ is a probabilistic measurement model, and $p(\mathbf{x}_{r_k} \,|\, \mathbf{x}_{r_{k-1}}, \mathbf{u}_k)$ is a probabilistic motion model.

However, there are several problems with implementing equation (19) in its generic form. To begin with, the potentially high dimensionality of the map can make the estimation of the belief $b_k(\mathbf{x}_{r_k}, M)$ computationally intractable. This is difficult to avoid in practice, as the number of landmarks in the map can easily be in the order of hundreds or even thousands. In addition, the belief function is hard to factorise due to the uncertainty of the robot and landmark positions being intricately intertwined [7, 9]. The need to maintain these intricate correlations only complicates the task of addressing the high computational complexity. Another problem is that the full posterior over a continuous space, which possesses infinitely many dimensions, cannot be represented by a digital computer [32].

Thus, working instantiations of Bayes filter are the product of additional assumptions and approximations. It is primarily these assumptions, along with their implications, that differentiate the currently existing probabilistic approaches. The type of assumptions adopted shape the strengths and limitations of each approach, as will become apparent in the following reviews.

The *expectation maximisation* (EM) based approach, proposed in [33], solves the SLAM problem by estimating the mode of the posterior $p(M \,|\, \mathbf{z}_{0:k}, \mathbf{u}_{0:k})$ (also denoted $b_k(M)$ for notational convenience) to find the most likely map M^*, along with the most likely path taken by the robot. Formally, this can be expressed as solving the maximum likelihood (ML) estimation problem

$$M^* = \underset{M}{\operatorname{argmax}}\, b_k(M). \tag{20}$$

The posterior $b_k(M)$, based on the derivation given in [33], can be written as

$$\begin{aligned} b_k(M) &= \int b_k(\mathbf{x}_{r_k}, M)\, d\mathbf{x}_{r_k} \\ &\propto \int \cdots \int \prod_{j=0}^{k} p(\mathbf{z}_j \,|\, \mathbf{x}_{r_j}, M) \;\cdot \\ &\quad \prod_{j=1}^{k} p(\mathbf{x}_{r_j} \,|\, \mathbf{x}_{r_{j-1}}, \mathbf{u}_j)\, d\mathbf{x}_{r_1} \ldots d\mathbf{x}_{r_k} \end{aligned} \tag{21}$$

where the robot's initial pose is, arbitrarily, set to $\mathbf{x}_{r_0} = [0\ 0\ 0]^T$. This equation is void of any constants, normalisation or otherwise, as the objective is to only maximise the posterior $b_k(M)$, not to calculate its true value.

The main problem with solving equation (20) is the high computational complexity. The maximisation of the likelihood function, defined in equation (21), involves searching in the space of all maps, and in each map, integrating over all possible poses at every instant in time. Since this is generally not feasible, an optimisation technique that performs local hill-climbing in likelihood space is used. This technique, based on the classical EM algorithm [34, 35], involves iterating two steps: an *expectation step*, or *E-step*, and a *maximisation step*, or *M-step*.

2.2.1 E-step

In this phase, probabilistic estimates for the poses $\mathbf{x}_{r_0}, \ldots, \mathbf{x}_{r_k}$ are calculated based on the currently best map M and data $D_{0:k}$ (where $D_{0:k} = \{\mathbf{z}_{0:k}, \mathbf{u}_{0:k}\}$). This can be considered a low-dimensional localisation problem, which is solvable using standard *Markov localisation* [36]. However, there is a slight difference that needs to be considered. Each posterior $p(\mathbf{x}_{r_j} \,|\, D_{0:k})$ is estimated using the data from the entire time interval $\{0, \ldots, k\}$, which requires two localisation passes: one forwards in time, giving $p(\mathbf{x}_{r_j} \,|\, D_{0:j})$, and the other backwards in time, giving $p(\mathbf{x}_{r_j} \,|\, D_{j+1:k})$.

2.2.2 M-step

The maximisation step involves calculating the most likely map M^* based on the pose estimates obtained in the E-step. In essence, this is a map optimisation problem whereby the robot's poses $\mathbf{x}_{r_{0:k}}$ are treated as latent variables. As given in [30], the map M^* is calculated by maximising the expectation over the joint log-likelihood of the robot's path $\mathbf{x}_{r_{0:k}}$ and data $D_{j+1:k}$:

$$M^{[\gamma+1]} = \underset{M}{\operatorname{argmax}}\, E[\log p(\mathbf{x}_{r_{0:k}}, D_{0:k} \,|\, M^{[\gamma]}) \,|\, D_{0:k}] \tag{22}$$

Here the superscript '$^{[\gamma]}$' denotes the iteration of the optimisation algorithm. The algorithm generates a sequence of maps $M^{[0]}, M^{[1]}, M^{[2]}, \ldots$ with monotonically increasing likelihood until a local maximum is reached. Since finding this local maximum can still be a difficult problem in a high-dimensional space, it has become common practice to represent the map as a discrete *occupancy grid* [37, 38] and solve equation (22) for each grid cell independently.

The EM approach has several key advantages over the EKF. Firstly, it provides a solution to the data association problem that does not require the unique identification of landmarks; in fact observed landmarks can be somewhat indistinguishable. Data association is performed through gradual reinforcement or degradation of matching probabilities as all the observation data over time is considered. This allows past data association decisions to be revised and possibly corrected. The EM approach can also estimate the mode of complex posteriors, and does not assume Gaussian noise like the EKF.

However, there are a few weaknesses that need to be considered. Unlike the EKF, the EM approach processes the entire data set multiple times, and as a consequence, does not provide an incremental solution to SLAM where a map is incrementally built as new observation data is received. Another weakness is that the EM algorithm is traditionally suited to offline processing. Online versions have been proposed (e.g., [39]), however, they partially sacrifice the robustness in the data association process to accommodate the restricted computational time windows. Lastly, the EM approach can become trapped in a local maxima and, hence, arrive at a suboptimal solution.

Montemerlo *et al.* [40] have recently proposed a new probabilistic approach to SLAM that is based on *particle filtering* [41, 42, 43]. This approach, called FastSLAM, estimates the posterior $p(\mathbf{x}_{r_{0:k}}, M \mid \mathbf{z}_{0:k}, \mathbf{u}_{0:k})$ (also denoted $b_k(\mathbf{x}_{r_{0:k}}, M)$), which is a slight variation of the commonly sought posterior given in equation (19). That is, instead of estimating the posterior over momentary robot poses, it estimates the posterior over robot paths. Before describing how this is done, the topic of particle filtering will be briefly summarised first (see [42] for a comprehensive review).

The idea behind particle filtering is to approximate the posterior density in a Markov chain through a process known as *importance sampling* [44]. In essence, the posterior is represented by a set of m random sample states or particles $S_k = \{s_k^j \mid 1 \leq j \leq m\}$ drawn from it. Each particle is given a weighting ω_k^j called an *importance factor*, which signifies the particle's quality relative to the other particles. The weighted particle set S_k is then processed in lieu of the full posterior. Note that the full posterior can still be roughly reconstructed, e.g., using a *histogram* or *kernel based density estimation technique* [45], because of the duality between particles and the posterior from which they are drawn.

A vanilla particle filter process can be described abstractly as follows. First, the initial set of particles S_0 are randomly drawn from the state space. For time $k \geq 1$, the set S_{k-1} is filtered (i.e. transformed into S_k) by computing two stages: a *prediction stage* and an *update stage*. In the prediction stage, a particle $\bar{s}_k^j$ is generated for each particle $s_{k-1}^j \in S_{k-1}$ according to an actuation model. The resulting particles, commonly referred to as the *proposal distribution*, are placed in a temporary set $\bar{S}_k$. In the update stage, the weight of all the particles in this temporary set are re-evaluated based on the latest observation information to produce the *target distribution*. Finally, m particles are drawn (with replacement) from $\bar{S}_k$ to give S_k, which involves drawing the higher weighted particles from $\bar{S}_k$ and resampling the others. The particles thereby converge toward better solutions, analogous to the evolution of chromosomes in *genetic* *algorithms* [46, 47]. The specifics of this process, however, vary according to the application and the particular particle filter used.

In the context of mobile robot localisation, the particle filter approach known as *Monte Carlo localisation (MCL)* [48] has been shown in studies such as [49] to be more robust than the EKF. MCL can also represent complex posteriors; solve the kidnapped robot problem; and operate as an *anytime algorithm* [50] under limited computational resources. However, for years particle filters were confined to these low-dimensional problems, due to the number of particles needed to populate a d-dimensional space increasing exponentially with d. Particle filters were therefore too inefficient to be used for high-dimensional problems like SLAM. However, this changed when Murphy [51] identified a structural property of SLAM that could be exploited to develop an efficient particle filter. This structural property is based on the condition that correlations in the uncertainty of different map landmarks arise only from uncertainty in the robot's pose. Therefore, if hypothetically the robot knows its trajectory perfectly, the landmark states can be estimated independently of each other. This conditional independence has led to the use of the so-called *Rao-Blackwellised particle filter* [42] (named after its relation to the *Rao-Blackwell theorem* [52]), which analytically marginalises out some of the variables attributed to a problem's structure for an efficient solution.

The FastSLAM approach is an instantiation of the Rao-Blackwellised particle filter. It uses the structural property identified by Murphy to estimate the posterior $b_k(\mathbf{x}_{r_{0\,:\,k}}, M)$ in the factorised form [40, 53]:

$$b_k(\mathbf{x}_{r_{0\,:\,k}}, M) = b_k(\mathbf{x}_{r_{0\,:\,k}}) \prod_{i=1}^{n} p(\mathbf{x}_i \mid \mathbf{x}_{r_{0\,:\,k}}, \mathbf{z}_{0\,:\,k}) \tag{23}$$

This factorisation is exact and universal to the SLAM problem. Essentially, it decomposes the posterior over robot paths and maps into $n + 1$ recursive estimators: one estimator over robot paths $b_k(\mathbf{x}_{r_{0\,:\,k}})$ and n separate estimators over landmark states $p(\mathbf{x}_i \mid \mathbf{x}_{r_{0\,:\,k}}, \mathbf{z}_{0\,:\,k})$ conditioned on each hypothetical path.

FastSLAM estimates the robot path posterior $b_k(\mathbf{x}_{r_{0\,:\,k}})$ using a particle filter. Each of the particles in this filter maintains its own map that consists of n independent EKFs, one for each of the landmarks. Thus, the j-th particle at time k can be written in the form

$$S_k^j = \{\mathbf{x}_{r_k}^j,\ \underbrace{\mu_{1,\,k}^j, \Sigma_{1,\,k}^j}_{\text{landmark } \mathbf{x}_1},\ \ldots,\ \underbrace{\mu_{n,\,k}^j, \Sigma_{n,\,k}^j}_{\text{landmark } \mathbf{x}_n}\ \} \tag{24}$$

where the mean $\mu_{i,\,k}^j$ and covariance $\Sigma_{i,\,k}^j$ are the Gaussian parameters of each landmark posterior. Therefore FastSLAM integrates particle filtering with Kalman filtering; however, in this context, each EKF is only estimating a single landmark position and, hence, is low-dimensional.

Currently, there are several variants of the FastSLAM algorithm, including *FastSLAM version 1.0* [40], *2.0* [54], and FastSLAM with unknown data association [53]. Assuming that the data association is uniquely known and the initial set S_0 has been initialised, the filtering algorithm proceeds as follows. First, the path posterior is extended by sampling a new pose $\mathbf{x}_{r_k}$ for each particle in the prior sample set S_{k-1}. FastSLAM 1.0 samples new poses based on the most recent control input $\mathbf{u}_k$:

$$\mathbf{x}_{r_k}^j \sim p(\mathbf{x}_{r_k} \,|\, \mathbf{x}_{r_{k-1}}^j, \mathbf{u}_k) \tag{25}$$

Although the measurements $\mathbf{z}_k$ are ignored, they are later incorporated through the resampling process. Nevertheless, this way of sampling new poses has been identified as being inefficient [54], especially when the robot's motion errors are large relative to the measurement noise. When this is the case, sampled poses tend to fall into areas of low measurement likelihood and, consequently, are poorly weighted. It is then likely that a large proportion of the sampled poses will be terminated, or wasted, through the resampling process. FastSLAM 2.0 addresses this problem by incorporating the measurements into the proposal distribution

$$\mathbf{x}_{r_k}^j \sim p(\mathbf{x}_{r_k} \,|\, \mathbf{x}_{r_{0\,:\,k-1}}^j, \mathbf{z}_{0\,:\,k-1}, \mathbf{u}_{0\,:\,k-1}) \tag{26}$$

which constitutes the primary difference between the two versions.

The second step of the filtering process involves updating the observed landmark estimates. This is performed by linearising the measurement function $\mathbf{h}$ and applying the standard EKF measurement update equations [8] (refer to [53] for a detailed description). These first two steps are then repeated m times to produce a set of m particles. The final step involves correcting the proposal distribution through resampling. Each particle is first assigned an importance weight, given by

$$\omega_k^j = \frac{\text{target distribution}}{\text{proposal distribution}} \tag{27}$$

Then m particles are drawn (with replacement) with a probability proportional to their weights. In the case of FastSLAM 1.0, this resampling process accounts for the latest measurements $\mathbf{z}_k$, which were earlier ignored. The purpose of resampling in FastSLAM 2.0, however, is more mundane. It is used merely to correct mismatches in the normalisation between particles [54].

In terms of performance, the FastSLAM approach has several key strengths. First and foremost, data association decisions can be robustly made on a per-particle basis, analogous to multiple hypothesis tracking (MHT) (discussed in Section 2.1). Therefore, instead of just maintaining the data association with the maximum likelihood, the posterior tracks multiple data associations that are resolved over time. Another strength is its computational complexity of $\mathcal{O}(m \log n)$ when the maps are represented by binary trees [40], which is theoretically lower than the quadratic complexity of the vanilla EKF. Also, FastSLAM can cope with a non-linear vehicle model without the need for linearisation.

The primary weakness of FastSLAM is that the resampling process continually reduces the diversity in the particle set by repeatedly discarding some particles and duplicating others [55]. If the resampling steps of every particle is traced back in time, there will be a point at which all the particles share a common history of the robot's trajectory and hence the same ancestor. Therefore the hypotheses of the robot's trajectory and landmark positions prior to this point of commonality cannot be revised. The resulting lack of particle diversity, called the *impoverishment problem* [56], restricts the size of the *loop* that can be corrected (the concept of closing an exploratory loop of accumulating positional errors is described in [57, 58, 59, 60]) and can lead to a suboptimal solution. A closely related problem occurs when there is an insufficient number of particles in the vicinity of the correct state [61]. This deprivation problem, which is inherent to all proactive approaches, is especially troublesome in large environments or when the robot is proverbially kidnapped. A lack of particles or the time lag involved in distributing particles can ultimately cause the filter to diverge. While increasing the number of particles can offset these problems, it adds to the computational complexity and therefore there is a limitation to the amount of particles that can be processed in real-time. Finally, a large computational effort can be wasted in updating particles with a negligible weight, as the variance of the weights tends to increase stochastically over time.

2.3 Scan Matching

Another category of SLAM approaches includes those that are based on aligning neighbouring sensor scans, e.g., from a laser or sonar scanner, to estimate the relative translations and rotations of the robot between scans. These *scan matching* approaches align the overlapping segments of the scan set by minimising some distance metric between inter-scan primitives or raw data. This is somewhat similar to *model-based matching* [62], however, scan matching does not use an accurate, dependable model as the base for comparison. Instead, it finds the congruence between noisy data sets that are negatively affected by occlusion and hence the robot's limited field of view.

The majority of scan matching approaches are derived from the *Iterative Closest Point (ICP) algorithm* [63, 64] and its many variants [65]. These approaches are based on iteratively refining an initial robot pose estimate obtained through odometry, which limits the search space. However, it is assumed that the displacement between the initial estimate and the robot's true pose is small enough to arrive at the globally optimal match.

The various approaches mainly differ in the primitives they select and match; the type of distance metric used (e.g., sum of squared distances between corresponding pairs); the weighting of correspondences; and the rejection of outliers. For example, Cox matches scan points to the line segments of a hand-crafted map [66]. Lu and Milios matches points to points in an *a priori* unknown, arbitrary environment (not necessarily polygonal) [67]. Their method does not rely on the uniqueness of landmarks and derives robustness from using the bulk of the scan points in the matching process. Gutmann and Konolige use a combination of the above two methods to take advantage of the computational efficiency of Cox's method and the universal capabilities of Lu and Milios's method [59]. They also take into consideration the topological relationships between neighbouring robot poses, associated via odometry and scan overlaps, to maintain a consistent map in large cyclic environments. Jensen and Siegwart establishes correspondences between points based on a probabilistic distance metric that incorporates both sensor noise and robot pose uncertainty [68]. This provides a way of robustly detecting outliers, and as a result, their algorithm exhibits a faster convergence than the standard ICP algorithm. Nüchter *et al.* also achieve a faster convergence by first subsampling the point data and then using *kd-trees* to efficiently find the closest points [69].

There are other, less common types of scan matching approaches. Some of these approaches are based on finding statistical correlations between scans, such as Weiss and Puttkamer's histogram matching approach [70] and Biber's *normal distributions transform (NDT)* [71]. These approaches do not require explicit correspondences between individual scan elements, however, they rely on the chosen statistical criteria effectively modeling the environment.

There are also approaches that have the ability to globally localise the robot without the aid of initial pose information. Crowley *et al.* accomplish this by using a training set of range scan profiles from various known poses to construct a lookup table, which can then be indexed to identify possible origins of a scan [72]. Gutmann *et al.* exploit the structured nature of the *RoboCup soccer field* [73] to match line segments extracted from a scan to those of an *a priori* map [74]. Weber *et al.* similarly use an *a priori* map of a structured environment, but instead of matching line segments, they match edges and concave/convex corners [75]. The matching process involves heuristically searching for corresponding patterns of inter-feature relationships, which are invariant to the robot's observational viewpoint. Tomono matches what are called *directed points*, comprising points and their tangent directions, which are also viewpoint invariant [76]. This researcher's approach solves the SLAM problem; however, it is computationally complex and the map does not converge over time. To address the complexity problem, global localisation is only applied when the robot fails to find a match using a localised search in the vicinity of the odometry estimate. This can lead to a suboptimal solution, as large odometry errors in a partially symmetrical environment can produce multiple hypotheses which all need to be considered.

The Kidnapped Way, proposed by Spero [77], matches sensor observations to continually solve the kidnapped robot problem over time. In this case, odometry and an associated vehicle model are purposely disregarded, and there is no assumption of continuity in the robot's motion. The robot's locomotive mechanism is thus irrelevant, novelly giving the robot anonymity and enabling SLAM to be implemented as a standalone, portable device with a similar flexibility to a *Global Positioning System (GPS)* receiver [78]. The robot can be robustly built without concern for its odometric accuracy or modeling complexities. However, this approach can fail in a highly symmetrical or featureless environment, and it can be computationally expensive to process a very large map.

Finally, there are hybrid approaches that combine some other SLAM approach with scan matching. For instance, Hähnel *et al.* combine FastSLAM with scan matching to minimise odometry error, thereby reducing the number of particles needed to build large-scale maps [79]. Pradalier and Sekhavat [80], on the other hand, use scan matching to improve the data association robustness of an EKF variant called the *geometric projection filter (GPF)* [81].

2.4 Qualitative Approaches

The last category of SLAM approaches in many ways mimics the qualitative, relativistic knowledge used in an animal or human's mental representation (or *cognitive map* [82]) of navigable environments [83, 84, 85, 86]; and hence has a biological premise. Qualitative SLAM approaches obviate the need for rigorous models of the robot's locomotion mechanism and sensors. They also do not strive for a metrically accurate map. In combination, these attributes give them a heightened robustness and computational efficiency.

These approaches, including [87, 88, 89] amongst many others, observe the topological spatial relationships between landmarks or obstacles to navigate and map the environment. Since this paper is largely concerned with quantitative approaches, they are only mentioned here for completeness. However, they do highlight the inverse relationship between rigorous modeling and robustness/generality.

3 Conclusion

This paper described the SLAM problem and then presented a review of the seminal approaches used to solve it. Most of these approaches, like the EKF, rely on stringent models and assumptions with regard to the robot's locomotion mechanism, sensor noise and the environment, and hence tend to only operate in a context specific situation. Additionally, the dubious task of modeling these aspects tends to entice the roboticist to contrive the situation so that the artificial model boundaries remain intact. This of course goes against the underlying ethos of SLAM: exploration of the unknown. Yet, for highly constrained situations, these approaches can produce a reasonably accurate solution. The Kidnapped Way, which goes against this modeling motif, is more in tune with the unstructured and chaotic nature of the real-world; however, its development is still in its infancy.

References

[1] J. J. Leonard, P. M. Newman, R. J. Rikoski, J. Neira, and J. D. Tardós, "Towards robust data association and feature modeling for concurrent mapping and localization," in 10^{th} *Int. Symp. Robotics Research*, Lorne, Victoria, Australia, 9-12 Nov. 2001, pp. 7–20.

[2] Y. Bar-Shalom and T. E. Fortmann, *Tracking and Data Association.* Boston, MA: Academic Press, 1988.

[3] I. J. Cox, "A review of statistical data association techniques for motion correspondence," *Int. J. Computer Vision*, vol. 10, no. 1, pp. 53–66, 1993.

[4] R. A. Brooks, "Elephants don't play chess," *Robotics and Autonomous Systems*, vol. 6, no. 1-2, pp. 3–15, 1990.

[5] D. J. Spero, "A review of outdoor robotics research," Dept. Electrical and Computer Systems Engineering, Monash University, Melbourne, Australia, Tech. Rep. MECSE-17-2004, 24 Nov. 2004. [Online]. Available: http://www.ds.eng.monash.edu.au/techrep/reports/

[6] M. W. M. G. Dissanayake, P. Newman, S. Clark, H. F. Durrant-Whyte, and M. Csorba, "A solution to the simultaneous localization and map building (SLAM) problem," *IEEE Trans. Robotics and Automation*, vol. 17, no. 3, pp. 229–241, 2001.

[7] R. Smith, M. Self, and P. Cheeseman, "A stochastic map for uncertain spatial relationships," in 4^{th} *Int. Symp. Robotics Research.* MIT Press, 1987.

[8] P. S. Maybeck, *Stochastic Models, Estimation, and Control*. New York: Academic Press, 1979.

[9] P. Moutarlier and R. Chatila, "Stochastic multisensory data fusion for mobile robot location and environment modelling," in 5^{th} *Int. Symp. Robotics Research*, Tokyo, Japan, 28-31 Aug. 1989, pp. 85–94.

[10] J. J. Leonard and H. F. Durrant-Whyte, "Simultaneous map building and localization for an autonomous mobile robot," in *IEEE/RSJ Int. Workshop on Intelligent Robots and Systems*, vol. 3, Osaka, Japan, 3-5 Nov. 1991, pp. 1442–1447.

[11] S. B. Williams, P. Newman, G. Dissanayake, and H. Durrant-Whyte, "Autonomous underwater simultaneous localisation and map building," in *Proc. IEEE Int. Conf. Robotics and Automation*, San Francisco, CA, 24-28 Apr. 2000, pp. 1793–1798.

[12] J. Guivant, E. Nebot, and S. Baiker, "Localization and map building using laser range sensors in outdoor applications," *J. Robotic Systems*, vol. 17, no. 10, pp. 565–583, 2000.

[13] J. A. Castellanos, J. Neira, and J. D. Tardós, "Multisensor fusion for simultaneous localization and map building," *IEEE Trans. Robotics and Automation*, vol. 17, no. 6, pp. 908–914, 2001.

[14] A. J. Davison and D. W. Murray, "Simultaneous localization and map-building using active vision," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 865–880, 2002.

[15] S. J. Julier and J. K. Uhlmann, "A counter example to the theory of simultaneous localization and map building," in *Proc. IEEE Int. Conf. Robotics and Automation*, Seoul, Korea, 21-26 May 2001, pp. 4238–4243.

[16] M. W. M. G. Dissanayake, H. F. Durrant-Whyte, and T. Bailey, "A computationally efficient solution to the simultaneous localisation and map building (SLAM) problem," in *Proc. IEEE Int. Conf. Robotics and Automation*, vol. 2, San Francisco, CA, 24-28 Apr. 2000, pp. 1009–1014.

[17] J. E. Guivant and E. M. Nebot, "Optimization of the simultaneous localization and map-building algorithm for real-time implementation," *IEEE Trans. Robotics and Automation*, vol. 17, no. 3, pp. 242–257, 2001.

[18] K. S. Chong and L. Kleeman, "Feature-based mapping in real, large scale environments using an ultrasonic array," *Int. J. Robotics Research*, vol. 18, no. 1, pp. 3–19, 1999.

[19] M. Csorba and H. F. Durrant-Whyte, "A new approach to map building using relative position estimates," in *Proc. SPIE: Navigation and Control Technologies for Unmanned Systems II*, Orlando, FL, 23 Apr. 1997, pp. 115–125.

[20] J. K. Uhlmann, S. J. Julier, and M. Csorba, "Nondivergent simultaneous map-building and localization using covariance intersection," in *Proc. SPIE: Navigation and Control Technologies for Unmanned Systems II*, Orlando, FL, 23 Apr. 1997, pp. 2–11.

[21] R. C. Luo and M. G. Kay, "Multisensor integration and fusion in intelligent systems," *IEEE Trans. Systems, Man and Cybernetics*, vol. 19, no. 5, pp. 901–931, 1989.

[22] M. W. M. G. Dissanayake, P. Newman, H. F. Durrant-Whyte, S. Clark, and M. Csorba, "An experimental and theoretical investigation into simultaneous localisation and map building," in *Proc. Sixth Int. Symp. Experimental Robotics*, Sydney, Australia, 26-28 Mar. 1999, pp. 265–274.

[23] A. Gelb, *Applied Optimal Estimation.* Cambridge, MA: MIT Press, 1974.

[24] J. A. Castellanos, J. Neira, and J. D. Tardós, "Limits to the consistency of EKF-based SLAM," in 5^{th} *IFAC Symp. Intelligent Autonomous Vehicles*, Lisbon, Portugal, 5-7 Jul. 2004.

[25] S. J. Julier and J. K. Uhlmann, "New extension of the Kalman filter to nonlinear systems," in *Proc. SPIE: Signal Processing, Sensor Fusion, and Target Recognition VI*, Orlando, FL, 21-24 Apr. 1997, pp. 182–193.

[26] J. Neira and J. D. Tardós, "Data association in stochastic mapping using the joint compatibility test," *IEEE Trans. Robotics and Automation*, vol. 17, no. 6, pp. 890–897, 2001.

[27] T. Bailey, E. M. Nebot, J. K. Rosenblatt, and H. F. Durrant-Whyte, "Data association for mobile robot navigation: A graph theoretic approach," in *Proc. IEEE Int. Conf. Robotics and Automation*, vol. 3, San Francisco, CA, 24-28 Apr. 2000, pp. 2512–2517.

[28] D. B. Reid, "An algorithm for tracking multiple targets," *IEEE Trans. Automatic Control*, vol. AC-24, no. 6, pp. 843–854, 1979.

[29] I. J. Cox and J. J. Leonard, "Probabilistic data association for dynamic world modeling: A multiple hypothesis approach," in *Fifth Int. Conf. Advanced Robotics*, Pisa, Italy, 19-22 Jun. 1991, pp. 1287–1294.

[30] S. Thrun, "Probabilistic algorithms in robotics," *AI Magazine*, vol. 21, no. 4, pp. 93–109, 2000.

[31] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, 3rd ed. Singapore: McGraw-Hill, 1991.

[32] S. Thrun, "Robotic mapping: A survey," Dept. Computer Science, Carnegie Mellon University, Pittsburgh, PA, Tech. Rep. CMU-CS-02-111, Feb. 2002.

[33] S. Thrun, W. Burgard, and D. Fox, "A probabilistic approach to concurrent mapping and localization for mobile robots," *Machine Learning*, vol. 31, no. 1-3, pp. 29–53, 1998.

[34] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Royal Statistical Society, Series B*, vol. 39, no. 1, pp. 1–38, 1977.

[35] G. J. McLachlan and T. Krishnan, *The EM Algorithm and Extensions.* New York: Wiley, 1997.

[36] W. Burgard, D. Fox, D. Hennig, and T. Schmidt, “Estimating the absolute position of a mobile robot using position probability grids,” in *Proc. Natl. Conf. Artificial Intelligence*, Portland, OR, 4-8 Aug. 1996, pp. 896–901.

[37] A. Elfes, “Occupancy grids: A probabilistic framework for robot perception and navigation,” Ph.D. dissertation, Dept. Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA, 1989.

[38] H. P. Moravec and A. Elfes, “High resolution maps from wide angle sonar,” in *Proc. IEEE Conf. Robotics and Automation*, St. Louis, MO, 25-28 Mar. 1985, pp. 116–121.

[39] C. Martin and S. Thrun, “Real-time acquisition of compact volumetric 3D maps with mobile robots,” in *Proc. IEEE Int. Conf. Robotics and Automation*, Washington, DC, 11-15 May 2002, pp. 311–316.

[40] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit, “FastSLAM: A factored solution to the simultaneous localization and mapping problem,” in *Proc. Natl. Conf. Artificial Intelligence*, Alta., Canada, 28 Jul.-1 Aug. 2002, pp. 593–598.

[41] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, “Novel approach to nonlinear/non-Gaussian Bayesian state estimation,” *IEE Proc. F (Radar and Signal Processing)*, vol. 140, no. 2, pp. 107–113, 1993.

[42] A. Doucet, N. de Freitas, and N. Gordon, Eds., *Sequential Monte Carlo Methods in Practice*. New York: Springer-Verlag, 2001.

[43] S. Thrun, “Particle filters in robotics,” in *Proc.* 18^{th} *Conf. Uncertainty in Artificial Intelligence*, Alberta, Canada, 1-4 Aug. 2002, pp. 511–518.

[44] D. B. Rubin, “Using the SIR algorithm to simulate posterior distributions,” in *Bayesian Statistics 3*, J. M. Bernardo, M. H. DeGroot, D. V. Lindley, and A. F. M. Smith, Eds. Oxford, UK: Oxford University Press, 1988.

[45] B. W. Silverman, *Density Estimation for Statistics and Data Analysis*. London: Chapman and Hall, 1986.

[46] J. H. Holland, *Adaptation in Natural and Artificial Systems*. Ann Arbor, MI: University of Michigan Press, 1975.

[47] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*. Reading, MA: Addison-Wesley, 1989.

[48] F. Dellaert, D. Fox, W. Burgard, and S. Thrun, “Monte Carlo localization for mobile robots,” in *Proc. IEEE Int. Conf. Robotics and Automation*, Detroit, MI, 10-15 May 1999, pp. 1322–1328.

[49] J. S. Gutmann and D. Fox, “An experimental comparison of localization methods continued,” in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, Lausanne, Switzerland, 30 Sep.-5 Oct. 2002, pp. 454–459.

[50] S. Zilberstein, “Resource-bounded sensing and planning in autonomous systems,” *Autonomous Robots*, vol. 3, no. 1, pp. 31–48, 1996.

[51] K. P. Murphy, "Bayesian map learning in dynamic environments," in *Neural Information Processing Systems*, Denver, CO, 29 Nov.-4 Dec. 1999, pp. 1015–1021.

[52] G. Casella and C. P. Robert, "Rao-Blackwellisation of sampling schemes," *Biometrika*, vol. 83, no. 1, pp. 81–94, 1996.

[53] S. Thrun, M. Montemerlo, D. Koller, B. Wegbreit, J. Nieto, and E. Nebot, "FastSLAM: An efficient solution to the simultaneous localization and mapping problem with unknown data association," *J. Machine Learning Research*, 2004, to appear.

[54] 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 *Eighteenth Int. Joint Conf. Artificial Intelligence*, Acapulco, Mexico, 9-15 Aug. 2003, pp. 907–912.

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

[56] O. King and D. A. Forsyth, "How does CONDENSATION behave with a finite number of samples?" in 6^{th} *Euro. Conf. Computer Vision*, Dublin, Ireland, 26 Jun.-1 Jul. 2000, pp. 695–709.

[57] R. Chatila and J. Laumond, "Position referencing and consistent world modeling for mobile robots," in *IEEE Int. Conf. Robotics and Automation*, St. Louis, MO, 25-28 Mar. 1985, pp. 138–145.

[58] F. Lu and E. Milios, "Globally consistent range scan alignment for environment mapping," *Autonomous Robots*, vol. 4, no. 4, pp. 333–349, 1997.

[59] J. S. Gutmann and K. Konolige, "Incremental mapping of large cyclic environments," in *Proc. IEEE Int. Symp. Computational Intelligence in Robotics and Automation*, Monterey, CA, 8-9 Nov. 1999, pp. 318–325.

[60] S. Thrun, W. Burgard, and D. Fox, "A real-time algorithm for mobile robot mapping with applications to multi-robot and 3D mapping," in *Proc. IEEE Int. Conf. Robotics and Automation*, San Francisco, CA, 24-28 Apr. 2000, pp. 321–328.

[61] R. van der Merwe, A. Doucet, N. de Freitas, and E. Wan, "The unscented particle filter," Dept. Engineering, Cambridge University, Cambridge, UK, Tech. Rep. CUED/F-INFENG/TR 380, 2000.

[62] W. E. L. Grimson, *Object Recognition by Computer: The Role of Geometric Constraints*. Cambridge, MA: MIT Press, 1991.

[63] Y. Chen and G. Medioni, "Object modeling by registration of multiple range images," in *Proc. IEEE Int. Conf. Robotics and Automation*, Sacramento, CA, 9-11 Apr. 1991, pp. 2724–2729.

[64] P. J. Besl and H. D. McKay, "A method for registration of 3-D shapes," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 239–256, 1992.

[65] S. Rusinkiewicz and M. Levoy, "Efficient variants of the ICP algorithm," in *Proc. Third Int. Conf. 3-D Digital Imaging and Modeling*, Quebec City, Que., Canada, 28 May-1 Jun. 2001, pp. 145–152.

[66] I. J. Cox, "Blanche - an experiment in guidance and navigation of an autonomous robot vehicle," *IEEE Trans. Robotics and Automation*, vol. 7, no. 2, pp. 193–204, 1991.

[67] F. Lu and E. Milios, "Robot pose estimation in unknown environments by matching 2D range scans," *J. Intelligent and Robotic Systems*, vol. 18, no. 3, pp. 249–275, 1997.

[68] B. Jensen and R. Siegwart, "Scan alignment with probabilistic distance metric," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, Sendai, Japan, 28 Sep.-2 Oct. 2004, pp. 2191–2196.

[69] A. Nüchter, K. Lingemann, J. Hertzberg, H. Surmann, K. Pervölz, M. Hennig, K. R. Tiruchinapalli, R. Worst, and T. Christaller, "Mapping of rescue environments with Kurt3D," in *IEEE Int. Workshop on Safety, Security and Rescue Robotics*, Kobe, Japan, 6-9 Jun. 2005, in press.

[70] G. Weiss and E. V. Puttkamer, "A map based on laserscans without geometric interpretation," in *Proc. Int. Conf. Intelligent Autonomous Systems*, Karlsruhe, Germany, 27-30 Mar. 1995, pp. 403–407.

[71] P. Biber, "The normal distributions transform: A new approach to laser scan matching," in *IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, Las Vegas, NV, 27-31 Oct. 2003, pp. 2743–2748.

[72] J. L. Crowley, F. Wallner, and B. Schiele, "Position estimation using principal components of range data," in *Proc. IEEE Int. Conf. Robotics and Automation*, Leuven, Belgium, 16-20 May 1998, pp. 3121–3128.

[73] H. Kitano, M. Asada, Y. Kuniyoshi, I. Noda, and E. Osawa, "RoboCup: the robot world cup initiative," in *Proc. First Int. Conf. Autonomous Agents*, Marina del Rey, CA, 5-8 Feb. 1997, pp. 340–347.

[74] J. S. Gutmann, T. Weigel, and B. Nebel, "Fast, accurate, and robust self-localization in polygonal environments," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, Kyongju, South Korea, 17-21 Oct. 1999, pp. 1412–1419.

[75] J. Weber, K. W. Jörg, and E. V. Puttkamer, "APR - global scan matching using anchor point relationships," in 6^{th} *Int. Conf. Intelligent Autonomous Systems*, Venice, Italy, 25-27 Jul. 2000, pp. 471–478.

[76] M. Tomono, "A scan matching method using Euclidean invariant signature for global localization and map building," in *IEEE Int. Conf. Robotics and Automation*, New Orleans, LA, 26 Apr.-1 May 2004, pp. 866–871.

[77] D. J. Spero, "Simultaneous localisation and map building: The kidnapped way," Ph.D. dissertation, Intelligent Robotics Research Centre, Monash University, Melbourne, Australia, 2005.

[78] J. Borenstein, H. R. Everett, L. Feng, and D. Wehe, "Mobile robot positioning: Sensors and techniques," *J. Robotic Systems*, vol. 14, no. 4, pp. 231–249, 1997.

[79] D. Hähnel, W. Burgard, D. Fox, and S. Thrun, "An efficient fastSLAM algorithm for generating maps of large-scale cyclic environments from raw laser range measurements," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, Las Vegas, NV, 27-31 Oct. 2003, pp. 206–211.

[80] C. Pradalier and S. Sekhavat, "Concurrent matching, localization and map building using invariant features," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, Lausanne, Switzerland, 30 Sep.-5 Oct. 2002, pp. 514–520.

[81] P. M. Newman, "On the structure and solution of the simultaneous localisation and map building problem," Ph.D. dissertation, Australian Centre for Field Robotics, University of Sydney, Australia, 1999.

[82] E. C. Tolman, "Cognitive maps in rats and men," *The Psychological Review*, vol. 55, pp. 189–208, 1948.

[83] J. Piaget and B. Inhelder, *The Child's Conception of Space*. New York: W. W. Norton, 1967.

[84] A. W. Siegel and S. H. White, "The development of spatial representations of large-scale environments," in *Advances in Child Development and Behavior*, H. W. Reese, Ed. New York: Academic Press, 1975.

[85] R. G. Golledge, Ed., *Wayfinding Behavior: Cognitive Mapping and Other Spatial Processes*. Baltimore, MD: Johns Hopkins University Press, 1999.

[86] D. M. Mark, C. Freksa, S. C. Hirtle, R. Lloyd, and B. Tversky, "Cognitive models of geographical space," *Int. J. Geographical Information Science*, vol. 13, no. 8, pp. 747–774, 1999.

[87] R. A. Brooks, "A robust layered control system for a mobile robot," *IEEE J. Robotics and Automation*, vol. RA-2, no. 1, pp. 14–23, 1986.

[88] T. S. Levitt and D. T. Lawton, "Qualitative navigation for mobile robots," *Artificial Intelligence*, vol. 44, no. 3, pp. 305–360, 1990.

[89] B. Kuipers and Y. T. Byun, "A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations," *Robotics and Autonomous Systems*, vol. 8, no. 1-2, pp. 47–63, 1991.