A Frequentist Approach to Mapping under Uncertainty ⋆

S. Chakravorty [a] R. Saha [b]

[a] *Assistant Professor, Aerospace Engineering, Texas A&M University, College station, TX*

[b] *Graduate Research Assistant, Aerospace Engineering, Texas A&M University, College station, TX*

Abstract

An asynchronous stochastic approximation based (Frequentist) approach is proposed for mapping using noisy mobile sensors under two different scenarios: 1) perfectly known sensor locations and 2) uncertain sensor locations. The frequentist methodology has linear complexity in the map components, is immune to the data association problem and is provably consistent. The frequentist methodology, in conjunction with a Bayesian estimator, is applied to the Simultaneous Localization and Mapping (SLAM) problem of Robotics. Several large maps are estimated using the hybrid Bayesian/ Frequentist scheme and results show that the technique is robust to the computational and performance issues inherent in the purely Bayesian approaches to the problem.

1 Introduction

The problem of mapping of large distributed environments using mobile noisy robotic sensors is addressed in this paper. Two case are considered: (a) the sensor location is known perfectly and (b) the sensor location is uncertain. We propose a frequentist approach for the solution of the problem and establish performance guarantees for the technique. The methodology, in conjunction with a Bayesian estimator, is applied to the so-called Simultaneous Localization and Mapping (SLAM) problem of robotics and several large distributed (dense) environments are mapped using the hybrid methodology to show its efficacy.

In the past several years, the topic of mobile sensor networks has been actively researched. In particular, distributed sensing and motion planning algorithms for such multiple sensor systems in uncertain environments have been devised [1–10]. Much of this work has concentrated on the discrete problems such as target tracking [3, 9, 10] while relatively fewer researchers have concentrated on the sensing of a distributed (dense) environment [2, 6, 7]. The sensing methodology used in much of this literature is the Kalman filter which is the Bayes filter for linear Gaussian systems. However, it can be shown that the Bayesian problem becomes computationally prohibitively expensive to solve in dense and large dimensional environments when the location of the sensors is uncertain. This is the case because the estimates of the components of the environment get correlated requiring that we maintain a high dimensional distribution on the different realizations of the environment, which is computationally infeasible for large dense maps. Further, getting rid of these correlations leads to inconsistent estimators (please see section III A of this paper for a simple example illustrating this issue). Further, under sensor location uncertainty, there arises the issue of data association: the problem of determining the unique one to one correspondence between observations and environmental components. The Kalman filter is especially brittle when faced with this problem and suitable data association techniques have to be used in conjunction with the filter [11, 12]. To the best of our knowledge, the problems mentioned above have not been addressed in the mobile sensor networks literature. In this paper, we propose an alternative asynchronous, stochastic approximation based (frequentist) approach to the problem of sensing large and dense environments, under sensor noise as well as uncertainty in the location of the sensor. Though we only consider the case of a single sensor, the methodology can be generalized to the multiple sensor case in a straightforward fashion and is left for future research. It is shown that the technique is robust to the computational problems inherent with the Bayesian approach to the sensing problem while provably retaining its performance guarantees, vis a vis consistency and immunity to the data association problem, in large and dense maps.

⋆ This paper was not presented at any IFAC meeting. Corresponding author: S. Chakravorty

Email addresses: schakrav@aero.tamu.edu (S. Chakravorty), roshmik@neo.tamu.edu (R. Saha).

In the Robotics community, researchers have focused on the mapping problem without sensor location uncertainty [13] as well as with sensor location uncertainty [11, 12]. The latter problem is popularly known in the community as the Simultaneous Localization and Mapping (SLAM) problem, in which the robot localizes itself within an unknown map while simultaneously estimating the unknown map. There are two main categories of approaches to SLAM: recursive and trajectory based, depending on whether the method tracks the entire trajectory of the robot, or only the current state/ pose. In the recursive Kalman filter/ Information filter based approach [11, 12, 14, 15], the map is appended to the filter as a parameter and the joint pose-map pdf is estimated using the Kalman recursion in either the covariance or the information matrix form. In either form, the computational complexity of the algorithms is intractable if the consistency of the filter has to be maintained, since neglecting the cross correlation terms in the extended Kalman filter (EKF) or the off-diagonal terms in the sparse extended Information filter (SEIF) leads to a loss of consistency. The Rao Blackwellized Particle Filter (RBPF) based SLAM algorithms, a trajectory based SLAM algorithm, keep track of the entire trajectory of the robot which decorrelates the observations of the various map components [16, 17]. Another trajectory based method, called Consistent Pose Estimation (CPE) [11] relies on maintaining a graph on the poses at which various scans of the map were made and then, optimizing the inter-node distances such that the likelihood of the observed data is maximized [18, 19]. The CPE method is an offline technique while the RBPF based SLAM techniques are inconsistent over long time horizons as the filters become inconsistent due to the "particle depletion" problem. Both these techniques tend to replace the "curse of dimensionality" inherent in the Bayesian formulation with a "curse of history".
It can be seen from the previous discussion that the central issue with the Bayesian formulation of the SLAM problem is the fact that computational complexity and consistency act at cross purposes, and alleviating one tends to worsen the other. The frequentist mapping methodology of this paper is applied to the SLAM problem using a "first localize -then map" philosophy wherein the robot is first localized with respect to a set of features in the map to obtain a belief distribution on the location of the sensor, and then this belief distribution is used by the frequentist technique to map the rest of the dense environment. The hybrid SLAM method proposed here alleviates the central "computation vs consistency" issue. The frequentist part of the algorithm has complexity linear in the map and is provably consistent. The complexity of the Bayesian part of the formulation can be kept under control owing to the sparseness of the set of landmarks/ features. Further, the frequentist part of the formulation is immune to the data association problem, while the Bayesian part is robust owing to the sparseness of the features/ landmarks.

The use of frequentist estimators based on recursive maximum likelihood (RML) or recursive least squares (RLS) is standard practice in the Hidden Markov Model (HMM) literature [20], and an application of this methodology in the mapping/ SLAM context is made in the reference [21]. These methods usually need to evaluate the filter derivative which is an $O(N^2)$ operation, where N is the number of particles used to represent the pdf of the state. This is usually impractical and only through recent advances in the particle filtering community, the above operation can now be done with $O(NlogN)$ complexity [22]. In contrast, the method presented here does not require the filter derivative, and if the Bayesian part of the hybrid formulation is implemented using a particle filter, the complexity of the frequentist algorithms is $O(N)$, where N is the number of particles used to represent the robot state pdf. This is due to the fact that general HMM techniques do not recognize, or exploit, the special structure inherent in the Mapping problem, in particular, that the problem may be solved in two steps by: a) estimating the sensor location independently of the rest of the environment, either through a global sensor such as Global Positioning system (GPS) or by localizing the sensor with respect to features in the environment, and b) mapping the rest of the environment based on this estimate.

The rest of the paper is organized as follows. In section II, we present the environment/ sensor model and the problem formulation. In section III, we propose the frequentist mapping technique. In section IV, we establsih the consistency of the frequentist approach. In section V, we apply the frequentist technique, in conjunction with a Bayes filter, to the SLAM problem, and present several experiments wherein large environments are mapped using the hybrid methodology. Preliminary versions of this paper have been published in [23–25]. In references [23, 24], the pure mapping problem was addressed, i.e., there was no uncertainty in the sensor location while in [25], a preliminary version of the problems in this paper were addressed. In particular, this paper considers the problem of dense maps and addresses the issue of data association in such maps, in contrast to [25], while also providing several validating examples.

2 Preliminaries

In this section, we shall outline the model that we use for the environment and the sensor. Next, we shall formulate the problems that we wish to address in this paper.

2.1 Model

Consider a single autonomous agent and let its state be denoted by the variable s (also sometimes called the robotic pose), and let the state of the environment be denoted by the variable $Q = \{q_1, \cdots, q_M\}$, where q_k are components of the environment (for instance, these would be the individual grids in a occupancy grid decomposition of the environment) and can take one of D discrete values. The environment is assumed to be stationary and its components mutually independent, i.e.,

$$p^*(Q) = \prod_{i=1}^{M} p^*(q_i), \tag{1}$$

where $p^*(Q)$ represents the probability of the realization Q of the environment, and $p^*(q_i)$ represents the probability of realization q_i for the i^{th} component of the environment. It can be anticipated that a large part of most environments can be modeled in this fashion. Any deterministic environment trivially satisfies the above assumption. The probability of observing the i^{th} environmental component in the state $\hat{q}_i$, where $\hat{q}_i$ can also take one of D values, and given that it is observed from the pose s, is given by:

$$p(\hat{q}_i/s) = \sum_{z} 1(\hat{q}_i/s, z) p(z/s), \tag{2}$$

where z is one of the finitely many observtions that are possible from state s, $p(z/s)$ denotes the likelihood of having the observation z given that the observation is made from state s, and $1(\hat{q}_i/s, z)$ is the indicator function for $\hat{q}_i$ taking a certain value given that the observation is z and is made from state s.

The observation likelihood $p(z/s)$ can be obtained in terms of the map probabilities as follows (using the Law of Total Probability):

$$p(z/s) = \sum_{q_1 \cdots q_M} p(z/s, q_1, \cdots q_M) p^*(q_1) \cdots p^*(q_M). \tag{3}$$

Plugging Eq. 2 into Eq. 3 implies:

$$\begin{aligned} & p(\hat{q}_i/s) \\ &= \sum_{z} 1(\hat{q}_i/s, z) \sum_{q_1,..,q_M} p(z/s, q_1, .., q_M) p^*(q_1)..p^*(q_M) \\ &= \sum_{q_1,..,q_M} \sum_{z} 1(\hat{q}_i/s, z) p(z/s, q_1..q_M) p^*(q_1) \cdots p^*(q_M) \end{aligned} \tag{4}$$

The likelihood $p(z/s, q_1, \cdots q_M)$ denotes the probability of an observation z given the particular realization of the environment $Q = \{q_1, \cdots, q_M\}$, and can be extracted from the noise characteristics of the sensor model. Please see Section IV A to see how this can be done. Note $\sum_z 1(\hat{q}_i/s, z) p(z/s, q_1 \cdots q_M) = p(\hat{q}_i/s, q_1 \cdots q_M)$, and plug into the equation above to obtain:

$$\begin{aligned} & p(\hat{q}_i/s) \\ &= \sum_{q_1..q_M} p(\hat{q}_i/s, q_1..q_M) p^*(q_1)..p^*(q_M), \\ &= \sum_{q_i} p^*(q_i) \sum_{q_k, k=1..M, k\neq i} p(\hat{q}_i/s, q_1, ..q_M) p^*(q_1)..p^*(q_M). \end{aligned} \tag{5}$$

We have

$$\begin{aligned} & \sum_{q_k, k=1..M, k\neq i} p(\hat{q}_i/s, q_1, ..q_M) p^*(q_1)..p^*(q_M) \\ & = p(\hat{q}_i/s, q_i). \end{aligned} \tag{6}$$

The above equation and Eq. 5 imply:

$$p(\hat{q}_i/s) = \sum_{q_i} p(\hat{q}_i/s, q_i) p^*(q_i), \tag{7}$$

which can be written in compact matrix form as:

$$\hat{P}_i(s) = A_i^*(s) P_i^*. \tag{8}$$

The vector $\hat{P}_i(s)$ stacks the observation probabilities $p(\hat{q}_i/s)$ of the i^{th} map component, and the matrix $A_i^*(s)$ is the true observation model of the i^{th} component when observed from pose s. It is a D x D matrix whose $(m, n)^{th}$ element is $p(\hat{q}_i = m/q_i = n, s)$. By true observation model $A_i^*(s)$, we mean the observation model formed by using the true map probabilities $P^* = \{P_1^*, \cdots P_M^*\}$. This is the fundamental observation equation for the mapping problem. Note that, in general, the observation matrix for the i^{th} map component, $A_i^*(s)$, depends on the location of the sensor s, the true map probability for the i^{th} component, P_i^*, as well as the map probabilties for the other map components. In general, the probabilities of a map component are affected only by its neighbouring components, the actual number that affect the component depend on the sensor. Given an accurate sensor such as a laser range finder, the number is extremely small while for noisy sensors such as Sonar, the number is much higher. Please see Section IV A to see how such an observation model can be obtained given the statistical noise characteristics of the sensor, the sensor location and the map probabilities.

Remark 1 *The above formulation of the observation model is very general and can be applied to most sensing scenarios. In certain special cases, the observation models $A_i^*(s)$ for the i^{th} map component may be independent of the rest of the map components, for instance, in sparse maps. In such cases the analysis is much easier and the conditions required for consistency are weaker. This spe-*

cial case was considered in the reference [23].

2.2 Problem Formulation

In this paper, we will address the following problems:

Problem 1 ***Mapping with Known Sensor Locations:*** *Given the time varying pose/ location of the sensor, $\{s_t\}$, and the noisy sensor readings from these poses, $\{z_t\}$, how do we construct a recursive estimator for the map probabilities P_i^*?*

Problem 2 ***Mapping with Uncertain Sensor Locations:*** *Given the time varying belief (probability density function: pdf) on the location/ pose of a sensor, $\{b_t(s)\}$, and the noisy readings $\{z_t\}$ from the belief states, how do we construct a recursive estimator for the map probabilities P_i^*?*

The primary difference between the two problems above is the uncertainty in the pose of the sensor. In the first case, it is perfectly known while in the second case it is uncertain and instead, specified by a pdf. The pdf on the location/ pose of the sensor may be specified either by localizing the sensor with respect to some landmarks/ beacons in the map (see application to SLAM in Section V), or through a global sensor such as GPS. Addressing the problems above allows us to avoid the "Complexity vs Consistency" trade-off inherent in the Bayesian solution of the problems, in particular, Problem 2.

3 The Frequentist Mapping Approach

In this section, we shall outline our approach to the two problems formulated in the previous section. Our approach relies on averaging, or the Law of Large Numbers, and hence, is termed frequentist. In the first subsection, we propose our solution to the problem of mapping with known sensor locations and generalize it to the problem of mapping with uncertain sensor locations in the next subsection.

3.1 Mapping with Known Sensor Locations

In this section, first, we shall derive the equations for the case when the sensor makes observations from a single pose and then generalize it to the case when the pose is time varying.

Please recall the observation model for a particular map component, Eq. 8. **This equation is the fundamental equation for the frequentist approach and provides an avenue for estimating the true map probabilities** P_i^*. Suppose we make repeated observations of the i^{th} component from pose s. We could count the number of times that we observe the i^{th} component in its various states, and form a consistent estimate of the observation probability vector $\hat{P}_i(s)$ by averaging, i.e.,

$$\hat{P}_i(s) = E_z[\mathbf{1}(\hat{q}_i/s, z)] \equiv E_z[c_i(s, z)] = \lim_N \frac{1}{N} \sum_{t=1}^{N} \mathbf{1}(\hat{q}_{i,t}/s, z_t). \quad (9)$$

In the above, given an observation z, the observation vector $c_i(s, z) = \mathbf{1}(\hat{q}_i/s, z)$ (it will be clear when we go to the uncertain pose case as to why choose this notation), where $\mathbf{1}(\hat{q}_i/s, z) = [1(\hat{q}_i = 1/s, z) \cdots 1(\hat{q}_i = D/s, z)]'$ is a D-dimensional vector indicator function. The above equation is correct due to the Law of Large Numbers. Then, using the knowledge of $A_i^*(s)$, we can obtain the true environmental probabilities P_i^* as

$$P_i^* = A_i^*(s)^{-1} \hat{P}_i(s). \quad (10)$$

Next, we may relax the assumption that the observations are made from the pose s and have that the observations are made from the time varying poses $\{s_t\}$, with true observation models $A_i^*(s_t)$. Recall that by true observation model $A_i^*(.)$, we imply the observation model constructed using the true map probabilities P^*. If we keep track of the relative frequencies of observations of the i^{th} component in its various different states, then the estimate of the true probabilities P_i^* can be recovered asymptotically using a time averaged observation model as follows:

$$P_i^* = \bar{A}_i^{-1} \hat{P}_i, \quad (11)$$

$$\hat{P}_i = \lim_N \frac{1}{N} \sum_{t=1}^{N} c_i(s_t, z_t), \quad (12)$$

$$\bar{A}_i = \lim_N \frac{1}{N} \sum_{t=1}^{N} A_i^*(s_t). \quad (13)$$

In order to derive the above expression, note that if we interpret the frequency of seeing the i^{th} map component in its $\hat{q}_i$ level, $\pi(\hat{q}_i)$, during the course of the mapping experiment as a probability, and if we also interpret the frequency of the robot being in a state s, $\pi(s)$, as a probability, then it follows using the simple rules of conditional probability that:

$$\pi(\hat{q}_i) = \sum_{q_i, s} p^*(\hat{q}_i/q_i, s) p^*(q_i) \pi(s) = \sum_{q_i} [\sum_{s} p^*(\hat{q}_i/q_i, s) \pi(s)] p^*(q_i). \quad (14)$$

The state s_t evolves according to a Markov Chain and given that the chain is ergodic (i.e., converges to a stationary distribution for all initial distributions), the left

hand side $\pi(\hat{q}_i)$ in the above equation is given by Eq. 12, and the matrix $[\sum_s p^*(\hat{q}_i/q_i, s)\pi(s)]$ is given by Eq. 13, and hence, the estimation equations for the time varying pose case follow.

The estimation equation for recovering the map probabilities, Eq. 11, is an asymptotic equation, i..e, it is true only as $N \rightarrow \infty$. However, we need a recursive estimator for the map probabilities for all N and still provide the guarantee that the map probabilities converge as $N \rightarrow \infty$. This may be done as follows. Note that essentially we want to solve the equation $\hat{P}_i = \bar{A}_i P_i^*$ for P_i^*. Given $\hat{P}_i$, $\bar{A}_i$, and that $-\bar{A}_i$ is Hurwitz, a recursive way to solve the equation is the following:

$$P_{i,t+1} = P_{i,t} + \gamma(\hat{P}_i - \bar{A}_i P_{i,t}). \quad (15)$$

To see why this is the case, note that for small γ, the above is the forward Euler approximation of the ordinary differential equation (ODE):

$$\dot{P}_i = \hat{P}_i - \bar{A}_i P_i. \quad (16)$$

Transforming the co-ordinates such that P_i^* is the origin, it is easy to see that P_i^* is the unique globally exponentially stable equilibrium of the ODE if $-\bar{A}_i$ is Hurwitz and full-rank. However, we do not have the asymptotic values $\hat{P}_i$ and $\bar{A}_i$ at any finite time during the algorithm's progress. Thus, at time t, given the pose of the sensor s_t and the reading z_t, we approximate $\hat{P}_i$ and $\bar{A}_i$ by their one step noisy estimates, i.e.,

$$\hat{P}_i \approx c_i(s_t, z_t),\ \bar{A}_i \approx A_i(s_t). \quad (17)$$

Utilizing the above approximations in Eq. 16, the true environmental probabilities P_i^* can be estimated using the following recursion for $P_{i,t}$, the estimate of P_i^* at time t, given $A_i^*(s_t)$ is positive definite (which is true under mild conditions, see Section IV).

Estimator E1:

$$P_{i,t} = \Pi_{\mathcal{P}}\{P_{i,t-1} + \gamma_t(c_i(s_t, z_t) - A_i^*(s_t)P_{i,t-1})\}, \quad (18)$$

where $\mathcal{P}$ represents the space of probability vectors in $\Re^D$, and $\Pi_{\mathcal{P}}(.)$ denotes a projection onto this compact set. We need the projection operator since the map probability estimates $P_{i,t}$ need to be probability vectors and the unconstrained recursion above need not result in a probability vector. The sequence $\{\gamma_t\}$ is usually of the form $at^{-\alpha}, \alpha < 1$, where a and α are learning rate parameters and standard for any stochastic approximation algorithm. The "noisy" algorithm above is a Stochastic Approximation algorithm and its convergence to the true map probabilities can be shown using techniques from Stochastic Approximation [26, 27]. We mention here that stochastic approximation algorithms as above are used in Q-learning, neural networks and system identification [28].

However, there remains the problem of using the "true" observation models $A_i^*(s_t)$ in order to form the estimates. This is unreasonable since it depends on the true map probabilities that we are trying to estimate. However, the estimates $P_t = \{P_{i,t}\}$ of the map probabilities can be used in Eqs. 6 -8 to form the observation models $A_i(s_t, P_t)$ as an approximation of the true observation models $A_i^*(s_t)$. Hence, we run the estimator E1 above using the approximate observation models $A_i(s_t, P_t)$ instead of the true observation models $A_i^*(s_t)$. These models can be inferred from the model of the particular type of sensor used for sensing the environment [13] (please see Section V A).

Remark 2 *A few details regarding the projection operator $\Pi_{\mathcal{P}}$ is provided in this remark. Note that $p(q_i = 1) + \cdots p(q_i = D) = 1$ and all of the terms are positive since they are probabilities. Then, we may eliminate one of the probabilities by replacing $p(q_i = D)$ by $1 - p(q_i = 1) + \cdots p(q_i = D-1)$ and having the constraint that every term is positive and their sum is less than one. This is easier understood in the 2-d case, i.e., when any map component can take one of only two values 0/1. In this case, $P(q_i = 0) = p_{i2} = 1 - P(q_i = 1)$. Denoting the probability $p(q_i = 1)$ by P_i, the constraint becomes that $0 \leq P_i \leq 1$. Thus, in that case, the map probabilities can be denoted by the vector $P = [P_1, \cdots, P_M]$, comprised of the occupancy probabilities of each of the map components. Then, the above algorithm reduces to the following:*

$$P_{i,t+1} = \Pi_{\mathcal{P}}[P_{i,t} + \gamma_t\{c_i(s_t, z_t) - p(\hat{q}_i = O/q_i = O, s_t)P_{i,t} - p(\hat{q}_i = O/q_i = E, s_t)(1 - P_{i,t})\}], \quad (19)$$

where $c_i(s_t, z_t) = 1(\hat{q}_{i,t} = 0/s_t, z_t)$, i.e., whether the grid is occupied or not, and $\Pi_{\mathcal{P}}[.]$ represents projection onto the interval $[0, 1]$. Many other possible projection mechanisms are possible but this is the method we use throughout this paper.

3.2 Mapping with Uncertain Sensor Locations

In this section, we relax the assumption that the pose of the robot is known perfectly. Instead, we assume that we are given a belief, i.e., a probability distribution, on the possible poses of the robot. Given the belief on the pose of the robot, $b_t(s)$ at time t, and a reading z_t of the environment, the frequentist mapping method is now used to map the (dense) environment Q. However, it is immediately apparent that there is an inherent "data association" problem with the mapping problem in this scenario. The observation, $\hat{q}_i$, of an environmental component q_i is no longer certain, since it varies with the

pose of the robot. Consider the simple situation illustrated in Fig. 1. The map component q_1, given reading z_2, is empty or occupied depending on whether the robot is at pose s_1 or s_2 respectively. However, since we have uncertainty in the location of the sensor, we cannot be sure as to whether the reading of q_1 is empty or occupied. Thus, the uncertainty in the observation of the map components has to be incorporated into the mapping technique. This may be done as follows. Given the uncertainty in the pose of the robot $b(s)$ and the reading of the environment z, the observation of the i^{th} component of the environment $\hat{q}_i$ is given by the probability vector (derived using the rules of conditional probability, and Bayes rule)

$$c_i^*(b(s), z) \equiv [p(\hat{q}_i = 1/b, z), \cdots p(\hat{q}_i = D/b, z)]' = \sum_s \mathbf{1}(\hat{q}_i/s, z) \frac{p^*(z/s)b(s)}{p^*(z/b)}, \quad (20)$$

$$p^*(z/s) = \sum_{q_1, \cdots, q_N} p(z/s, q_1, \cdots q_N) p^*(q_1) \cdots p^*(q_N), \quad (21)$$

where $p^*(z/b) = \sum_s p^*(z/s)b(s)$ is the factor used to normalize $c_i(.)$ and $p^*(z/s)$ is the true likelihood of the observation z given that it is made from pose s. In order to derive the above expression, note that using the thorem of total probability, $p(\hat{q}_i/b, z) = \sum_s p(\hat{q}_i/s, b, z)p(s/z, b)$. We can expand the term $p(s/z, b)$ using Bayes rule which gives us $p(s/z, b) = \frac{p^*(z/s)b(s)}{p^*(z/b)}$, after using the fact that $p^*(z/s, b) = p^*(z/s)$, and $p(\hat{q}_i/s, b, z) = \mathbf{1}(\hat{q}_i/s, z)$.
As in the perfect pose information case, averaging over all observations z (which can be formed by a time average due to the Law of Large Numbers), allows us to estimate the probability of observing state $\hat{q}_i$ given the belief state $b(s)$, i.e.,

$$\hat{P}_i(b) = E_z[c_i^*(b, z)] = \lim_N \frac{1}{N} \sum_{t=1}^{N} c_i^*(b, z_t). \quad (22)$$

Note that the above probabilistic description of the observation solves the "data association" problem: we are no longer certain if the observed value of the i^{th} map component is in its k^{th} level, instead we associate a probability with this observation. The probability of observing the map component q_i at level $\hat{q}_i$, given the belief on the pose $b(s)$ is also given by

$$p(\hat{q}_i/b) = \sum_s b(s) \sum_{q_1..q_N} p(\hat{q}_i/q_1..q_N, s) p^*(q_1) \cdots p^*(q_N), \quad (23)$$

which can be written in compact matrix form as follows:

$$\hat{P}_i(b) = A_i^*(b) P_i^*, \quad (24)$$

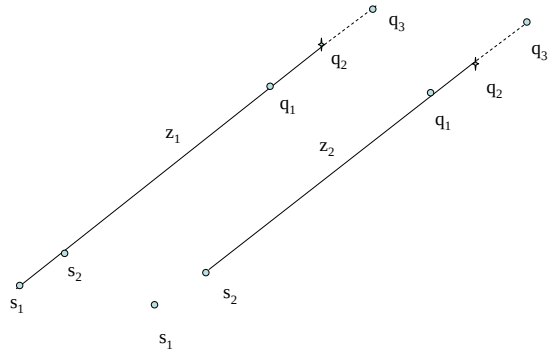


Fig. 1. The problem of data association

where

$$A_i^*(b) = \sum_s A_i^*(s) b(s). \quad (25)$$

Note here that this equation is exactly analogous to the frequentist mapping Eq. 8, wherein the exact pose knowledge s has been replaced by the belief on the pose of the robot $b(s)$. The observation model $A_i^*(s)$ is replaced by the expected observation model with the expectation being taken with respect to the belief on the pose of the robot. Thus, similar to the case with perfect pose information, if we were to remain in the belief state $b(s)$ and make repeated observations of the i^{th} component of the environment, we would be able to recover the left hand side of the above Eq. 24, $\hat{P}_i(b)$, by averaging the (probabilistic) observations of the i^{th} component, $c_i(b, z_t)$ (cf. Eq. 22). Hence, the true environmental probabilities may be recovered asymptotically by inverting Eq. 24. Generalizing the situation to the case when we have a time-varying belief on the pose of the robot, $b_t(s)$, the true environmental probabilities can be estimated recursively using the following analog of frequentist estimator E1 .

Estimator E2

$$P_{i,t} = \Pi_{\mathcal{P}}\{P_{i,t-1} + \gamma_t(c_i^*(b_t, z_t) - A_i^*(b_t)P_{i,t-1})\}, \quad (26)$$

As in the pure mapping case, since the variables $c_i^*(b_t, z_t)$ and $A_i^*(b_t)$ are dependent on the true map probabilities P_i^*, the estimator is actually run by using the current estimate of the true observation models/ observation likelihood. In other words, the above algorithm is run using $c_i(b_t, z_t, P_t)$ and $A_i(b_t, P_t)$, where the current estimate of the map probabilities P_t is used, instead of the true map probabilities P^*, in Eq. (20) to form $c_i(b_t, z_t, P_t)$, and in Eqs. (23)-(24) to form $A_i(b_t, P_t)$.

4 Consistency

In this section, the consistency of the frequentist mapping algorithm under uncertain robot poses is established. In our context, consistency implies that the estimated map probabilities converge to the true map probabilities with probability one, or almost surely. We shall prove the result using the powerful ODE method from the stochastic approximation literature [26, 27]. To begin, we present a short introduction of the method to clarify the basic idea behind the method.
Consider the mapping algorithm from above, Eq. 26, without the projection operation and using the estimates of $c_i^*(.)$ and $A_i^*(.)$, formed using the current map estimates P_t, as mentioned at the end of the previous section:

$$P_{i,t+1} = P_{i,t} + \gamma[c_i(X_t, P_t) - A_i(X_t, P_t)P_{i,t}], \quad (27)$$

where $X_t = (b_t, z_t)$, the 2-tuple consisting of the belief state and the observation at any instant. It is easily seen that the state of the algorithm X_t evolves according to a Markov chain, whose transition probabilities, in general, depend on the map probability estimates P_t. If the learning rate parameter γ is small, then the value of the map probabilities does not change quickly, and can be assumed to be essentially equilibrated over N steps, and then

$$P_{i,t+N} \approx P_{i,t} + \gamma N(\frac{1}{N})\sum_{k=1}^{N}[c_i(X_{t+k}, P_t) - A_i(X_{t+k}, P_t)P_{i,t}], \quad (28)$$

Then, using the law of large numbers, it follows that if N is large enough:

$$\frac{1}{N}\sum_{k=1}^{N}[c_i(X_{t+k}, P_t) - A_i(X_{t+k}, P_t)P_{i,t}] \approx \bar{h}_i^*(P_t) - \bar{A}_i(P_t)P_{i,t}, \quad (29)$$

where $\bar{h}_i^*(.)$ and $\bar{A}_i(.)$ are the averaged values of $c_i(.)$ and $A_i(.)$ respectively. Then, it follows that

$$P_{i,t+N} - P_{i,t} \approx N\gamma[\bar{h}_i^*(P_t) - \bar{h}_i(P_t)], \quad (30)$$

where $\bar{h}_i(P_t) = \bar{A}_i(P_t)P_{i,t}$, which happens to be the forward Euler approximation (with step size $N\gamma$) of the differential equation:

$$\dot{P}_i = \bar{h}_i^*(P) - \bar{h}_i(P). \quad (31)$$

The idea behind the method is that the asymptotic performance of the estimation/ mapping algorithm can be analyzed by analyzing the behaviour of the "mean/ average" ODE above. This method is very popular in analyzing the behaviour of algorithms in many different fields including reinforcement learning [28], neural networks [29], system identification and stochastic adaptive control [26, 27]. In the following, we analyze the frequentist mapping algorithm using the ODE method.

Consider the mapping algorithm as presented previously:

$$P_{i,t+1} = \Pi_{\mathcal{P}}\{P_{i,t} + \gamma_t[c_i(X_t, P_t) - A_i(X_t, P_t)P_{i,t}]\}, \quad (32)$$

where $X_t = (b_t, z_t)$ is the 2-tuple consisting of the belief state and the observation at any instant. The mean true observation probabilities of the i^{th} map component and the mean "current" predicted value are defined as:

$$h_i^*(b, p) \equiv E_z^*[c_i(b, z, P)] = \sum_z p^*(z/b)c_i(b, z, P), \quad (33)$$

$$h_i(b, P) \equiv E_z[c_i(b, z, P)] = A_i(b, z, P)P_i = \sum_z p(z/b)c_i(b, z, P), \quad (34)$$

where $p^*(z/b)$ is the probability of an observation z given the true map probabilities P^*, and $p(z/b)$ are the probabilities given the estimate of the map probabilities P, and given that the belief state is b. Recall that $c_i(b, z, P)$ is the vector containing the observation probabilities of the i^{th} map component in its various different states, given that the belief on the robot pose is b, the reading from the sensor is z and the estimate of the map probabilities is P. Note that $c_i(b, z, P)$ is the approximation of the true observation probability vector $c_i^*(b, z, P)$ where the vector of true map probabilities, P^*, is replaced by the approximate map probabilities P (see the end of Section II C). If the map probabilities were truly P, then if we averaged $C_i(b, z, P)$ over all observations z, we would obtain the quantity $h_i(b, P)$. However, since the observation z is generated by the true map probabilities P^*, not P, we would obtain the quantity $h_i^*(b, P)$ which is in general, different from $h_i(b, P)$. In fact, only at $P = P^*$ are the two quantities equal and the algorithm uses this fact to guide the map estimates towards the true map probabilities. It can be seen that

$$p^*(z/b) = \sum_{s,q_1..q_M} p(z/s, q_1..q_M)p^*(q_1)..p^*(q_M)b(s), \quad (35)$$

$$p(z/b) = \sum_{s,q_1..q_M} p(z/s, q_1..q_M)p(q_1)..p(q_M)b(s). \quad (36)$$

In the following, for the sake of simplicity (the extension is reasonably straightforward), we deal exclusively with the 2-dimensional case, i.e., when each map component q_i can take one of two values 0/1. The map probabilities can now be denoted by the vector $P = [P_1, P_2....P_M]$, where P_i denotes the probability that the i^{th} map compo-

nent takes value 1 **(i.e., is occupied).**

First, we make the following assumption.

A 1 *Corresponding to every map probability P, let the belief process b_t have a stationary distribution $\pi_\infty(b,P)$. Moreover, let the belief process be geometrically ergodic. Let*

$$\bar{h}_i^*(P) = \int_{b\in B_i} h_i^*(b,P)\frac{d\pi_\infty(b,P)}{\pi_\infty(B_i,P)}, \tag{37}$$

$$\bar{h}_i(P) = \int_{b\in B_i} h_i(b,P)\frac{d\pi_\infty(b,P)}{\pi_\infty(B_i,P)}, \tag{38}$$

where B_i are all the belief states that map component i is observed from. In particular, the above assumption implies that there exist $K<\infty$, $\rho<1$ such that

$$E[C_i(b_t,z_t,P)-\bar{h}_i^*(P)] \le K\rho^t,$$
$$E[A_i(b_t,P)P_i-\bar{h}_i(P)] \le K\rho^t, \tag{39}$$

i.e., the quantities $C_i(.)$ and $A_i(.)$ converge to their average values exponentially fast.

Define $\bar{H}^*(P) = [\bar{h}_1^*(P),\cdots\bar{h}_M^*(P)]^t$, and $\bar{H}(P) = [\bar{h}_1(P),\cdots\bar{h}_M(P)]^t$. Then, under assumption A 1, it can be shown that the asymptotic behaviour of the mapping algorithm is characterized by the solution of the following ODE ([27], pg. 187, Ch. 6, Theorem 6.1):

$$\dot{P} = \bar{H}^*(P) - \bar{H}(P). \tag{40}$$

In particular, the following result holds.

Proposition 1 *Let the point $P=P^*$ be an asymptotically stable equilibrium of the ODE (40) with domain of attraction D^*. Let $C\subseteq D^*$ be some compact subset of D^*. Let the learning rate parameters $\{\gamma_t\}$ be such that $\sum_t\gamma_t=\infty$, and $\gamma_t\to 0$ as $t\to\infty$. If the trajectories of the mapping algorithm (32) enter the subset C infinitely often, then the estimates $P_t\to P^*$ almost surely.*

Hence, it is left for us to show that the set of true map probabilities $P^*=[P_1^*,....,P_M^*]$ is an asymptotically stable equilibrium of ODE (40). In order to show this, we will show that the linearization of the mean ODE (40) about P^* is asymptotically stable and hence, so is the nonlinear ODE ([30], Chapter 3, Theorem 3.7). We limit our treatment to the case when $D=2$, i.e., the map components can take one of two values. The gradient of the vector field $\bar{H}^*(P)-\bar{H}(P)$ is defined by the matrix:

$$\nabla(\bar{H}^*(P)-\bar{H}(P)) = [\partial_i(\bar{h}_j^*(P)-\bar{h}_j(P))]. \tag{41}$$

We make the following assumption.

A 2 *We assume that*

$$\partial_i(\bar{h}_i^*(P)-\bar{h}_i(P)) < -\epsilon,\ \forall\, i, \tag{42}$$

$$|\sum_{j\neq i}[\partial_j(\bar{h}_i^*(P)-\bar{h}_i(P))]| \le |\partial_i(\bar{h}_i^*(P)-\bar{h}_i(P))|, \tag{43}$$

where all the partial derivatives above are evaluated at $P=P^*$, and $\epsilon>0$ is a positive constant.

We will provide the justification of the above assumptions after the following proposition.

Proposition 2 *Let assumption A 2 be satisfied. Then the true map probability vector P^* is an asymptotically stable equilibrium of ODE 40 with a non-empty region of attraction D^*. Thus, if the mapping algorithm estimates P_t visit a compact subset $C\subseteq D^*$ infinitely often, then $P_t\to P^*$ almost surely, due to Proposition 1.*

Proof: Under assumption A 2, the linearization of the mean ODE (40) about P^* is row-dominant, and hence, all its eigenvalues lie in the open left half plane and their real parts are bounded atleast ϵ away from the imaginary axis [31]. Therefore, it follows that P^* is an asymptotically stable equilibrium point of the mean ODE and hence, all the other results follow. *Q.E.D*

Finally, we furnish the justification for assumption A 2. In order to do this, note that

$$\partial_j(\bar{h}_i^*(P)-\bar{h}_i(P)) = \int_{b\in B_i}\partial_j(h_i^*(b,P)-h_i(b,P))\frac{d\pi_\infty(b,P)}{\pi_\infty(B_i,P)}. \tag{44}$$

Consider the term $\partial_j(h_i^*(b,P)-h_i(b,P))$ for some belief state b. In the following to simplify notation, all the partial derivatives are assumed to be evaluated at $P=P^*$. It may be shown that:

$$\partial_j(h_i^*(b,P)-h_i(b,P)) = -\sum_z \partial_j p(z/b)c_i(b,z,P), \tag{45}$$

where the partial derivative above is evaluated at $P=P^*$, and that:

$$\partial_j p(z/b) = p^*(z/b,q_j=1)-p^*(z/b,q_j=0), \tag{46}$$

i.e., the difference in the probabilities of observing z given belief state b, and whether q_j is in state 1 or state 0. Then, it follows that

$$\partial_j(h_i^*(b,P)-h_i(b,P)) =$$

$$-\sum_z (p^*(z/b, q_j = 1) - p^*(z/b, q_j = 0))p^*(\hat{q}_i = 1/z, b),$$

where recall that the variable $p^*(\hat{q}_i = 1/z, b) = c_i^*(b, z)$ is the "true" probability that the observation of the i^{th} map component is 1 given the belief state b and the observation z (see Section 2.2). Hence, it follows that

$$\partial_i(h_i^*(b, P) - h_i(b, P)) =$$
$$-\sum_z [p^*(z/b, q_i = 1) - p^*(z/b, q_i = 0)]p^*(\hat{q}_i = 1/z, b),$$
$$= -[p^*(\hat{q}_i = 1/q_i = 1, b) - p^*(\hat{q}_i = 1/q_i = 0, b)]$$

Hence, $\partial_i(h_i^*(b, P) - h_i(b, P)) < -\epsilon$ if $p^*(\hat{q}_i = 1/q_i = 1, b) - p^*(\hat{q}_i = 1/q_i = 0, b) > \epsilon$. The above equation thus implies that the probability of observing the map component occupied when it is actually occupied should be more than the probability of seeing it occupied when it is actually unoccupied (i.e., a spurious observation due to some other map component). This in turn is a "good sensor" assumption, i.e, we see the right observation more times than the wrong one. This corresponds to the heuristic that we neglect or disregard the observations of the map components that are too far from our current belief state. Thus, the set B_i in Eq. 44 above should consist of only those belief states from which the observation of map component i can be reliable. Ensuring that the set B_i is chosen in the above fashion implies that $\partial_i(h_i^*(b, P) - h_i(b, P)) < -\epsilon$ for all $b \in B_i$, and hence, it follows that Eq. (42) is automatically satisfied. Recall the definitions of $h_i(b, P)$ and $h_i^*(b, P)$ (cf. Eqs. (33)-(34)). The difference between the two signifies the average observation prediction error of the i^{th} map component given that the map probability estimates are P. Recall that it is zero for $P = P^*$. Thus, $\partial_j(h_i^*(b, P) - h_i(b, P))$ represents the sensitivity of this error to the j^{th} component of the map probabilities. Now, if we require that

$$|\sum_{j \neq i} [\partial_j(h_i^*(b, P) - h_i(b, P))]| \leq |\partial_i(h_i^*(b, P) - h_i(b, P))|,$$

for all $b \in B_i$, then Eq. (43 in assumption 2 is automatically satisfied. The equation above implies that the sensitivity of the observation error of the i^{th} map component to its own map probabilities should dominate the cumulative sensitivity of the error to all other map components. This is a structural assumption that is required regardless of whether the robot pose is uncertain or perfectly known. In fact, it may be reasonably expected that it is satisfied if the map components are updated only from "good" belief states, i.e., belief states from which the sensors can be assumed to be reliable. Experimental evidence seems to suggest the same as well. The development above can then be summarized in the following proposition.

Proposition 3 *Given any map component q_i let there exist a set of belief states G_i s.t. for all beliefs $b \in G_i$, the following hold:*

$$p^*(\hat{q}_i = 1/q_i = 1, b) - p^*(\hat{q}_i = 1/q_i = 0, b) > \epsilon,$$
$$|\partial_i(h_i^*(b, P) - h_i(b, P))| \geq |\sum_{j \neq i} \partial_j(h_i^*(b, P) - h_i(b, P))|,$$

where all the partial derivatives above are evaluated at $P = P^*$. If the sets B_i are chosen such that $B_i \subseteq G_i$, then assumption 2 is automatically satisfied and hence, Proposition 2 holds.

This completes our proof of the consistency of the mapping algorithm given an uncertain time-varying belief state. In practice, the sets B_i have to be chosen heuristically, and some trial and error might be required before we arrive at the right choice for a particular type of sensor. In the examples in Section IV, the sets B_i were chosen such that we discard observations of more than 60 m for the accurate laser range finder and observations of more than 15 m for the noisy sonar sensor.

4.1 Numerical Example

In this subsection, we provide a very simple example that clearly illustrates the convergence of the mapping algorithms as established above. We also show using the same example the computation versus consistency problem inherent in the Bayesian approach.

Consider the situation shown in Fig. 1. We assume that the robot can be at one of the two locations s_1 and s_2 with probabilities b_1 and b_2. We have a perfect range sensor. There are two grids: q_1 and q_2, q_1 is empty and q_2 is occupied. Thus, there are only two possible observations: z_1, that made from s_1 and z_2, that made from s_2, as shown in the figure. Since, we do not know exactly where the robot is located, we might sometimes think that we have a reading fom q_3, which is actually occluded. Note that we either have no information about q_3 or think its occupied, its never seen empty.
First, we show the consistency of the frequentist approach. Let the current estimates of the map occupancy probabilities for q_1, q_2 and q_3 be p_1, p_2 and p_3 respectively (probability that the map components are occupied). Consider the probability of observing q_i occupied given the above estimates, the reading z and the belief state b, $c_i(b, z, P)$ as defined previously (cf. Eq. 20):

$$c_i(b, z, P) = \sum_{s=s_1, s_2} 1(\hat{q}_i = O/z, s)\frac{p(z/s)b(s)}{p(z/b)}, i = 1, 2, 3. \tag{47}$$

We see that $p(z_1/s_1) = p(z_1/q_1 = E, q_2 = O, s_1)p(q_1 = E)p(q_2 = O) = (1 - p_1)p_2$. Similarly, $p(z_1/s_2) = (1 -$

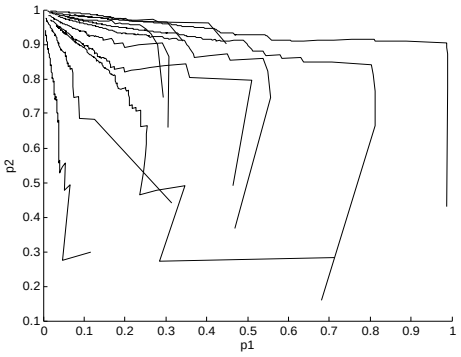


Fig. 2. Convergence of Map Probabilities

$p_1)(1-p_2)p_3$, $p(z_2/s_1) = p_1$, $p(z_2/s_2) = (1-p_1)p_2$, and hence, $p(z_1/b) = (1-p_1)p_2b_1 + (1-p_1)(1-p_2)p_3b_2$, and $p(z_2/b) = p_1b_1 + (1-p_1)p_2b_2$. Substituting the above into the expression for $c_i(b, z, P)$ leads us to the following: $c_1(b, z_1, P) = 0$, $c_1(b, z_2, P) = \frac{b_1p_1}{p(z_2/b)}$, $c_2(b, z_1, P) = \frac{(1-p_1)p_2b_1}{p(z_1/b)}$, $c_2(b, z_2, P) = 1$. Since, we have a perfect sensor, the observation matrices are identity, and thus, the update equation simply becomes:

$$p_{i,t} = (1 - \gamma_t)p_{i,t-1} + \gamma_t c_i(b, z_t, P_{t-1}), \quad (48)$$

where $\gamma_t = \frac{1}{at}$. The sanity check that needs to be performed on the above algorithm is that the true map probabilities $p_1 = 0, p_2 = 1$ are fixed points of the above algorithm. We show this for p_2, the case of p_1 is trivial. Given $p_1 = 0, p_2 = 1$, and z_1 is observed, p_2 will be updated by the algorithm to $c_2(b, z_1, P) = \frac{(1-p_1)p_2b_1}{p(z_1/b)} = \frac{1.1.b_1}{b_1+1.0.p_3.b_2} = 1$. The update of p_2 for z_2 is always 1. Hence, indeed the true map probabilitites are fixed points of the algorithm.

Of course, this does not mean that the algorithm converges to the true map probabilities given arbitrary initial conditions on p_1, p_2 and p_3. In Fig. 2, we show simulations of the progress of the mapping algorithm for various initial conditions of p_1 and p_2, and for $b_1 = 0.4$ and $b_2 = 0.6$ (the behaviour is similar for other choices of $b1$ and $b2$). The plots show the trajectories of the algorithm in the $p_1 - p_2$ plane for various different initial conditions. It is seen from the figure that the algorithm does indeed converge to the true map probabilities for all initial conditions.

Now, we show that the Bayesian approach fails unless the correlation between the components q_1, q_2 and the robot pose s is maintained. Consider first the case when we do not consider correlations of any map component to any other map component as well as the robotic pose. In such an approach, the update of the map component would be

$$p(q_i = O/\mathcal{F}^t) = \eta p(z_t/b, q_i = O)p(q_i = O/\mathcal{F}^{t-1}). \quad (49)$$

The sanity check has to be that the true map probabilities is a fixed point of the above update equation. Consider the update of $p(q_2 = O)$ given $p_1 = 0, p_2 = 1$. If observation z_1 turns up, the update of $p(q_2 = O)$ is $p(q_2 = O) = \frac{p(z_1/b,q_2=O)p(q_2=O)}{p(z_1/b,q_2=E)p(q_2=E)+p(z_1/b,q_2=O)p(q_2=O)} = \frac{(1-p_1)p_2b_1}{(1-p_1)p_2b_1+p_3b_2} = \frac{b_1}{b_1+p_3b_2} \neq 1$, since p_3 cannot be equal to 0 for its either occluded or mistakenly thought to be occupied. Hence, the true map probabilities are not a fixed point of the above update equation, and hence, the update is not consistent.

Next, we show the case where we maintain the correlations between the map component and the robot pose but not with other map components. In this case, the joint distribution is

$$p(s, q_1, q_2) = b(s)p(q_1/s)p(q_2/s). \quad (50)$$

Thus

$$p(q_i) = \sum_s b(s)p(q_i/s). \quad (51)$$

The update equation for $p(q_i/s)$ is then given by:

$$p(q_i/s, \mathcal{F}^t) = \eta p(z/s, q_i)p(q_i/s, \mathcal{F}^{t-1}). \quad (52)$$

Let $p(q_2/s_1) = p_{21}$, $p(q_2/s_2) = p_{22}$. Again, we assume the true map probabilities, $p_1 = 0, p_2 = 1$, and show that they are not a fixed point of the above iteration. Given z_1, $p(q_2 = O/s_1) = \frac{p(z_1/s_1,q_2=O)p(q_2=O/s_1)}{p(z_1/s_1,q_2=E)p(q_2=E/s_1)+p(z_1/s_1,q_2=O)p(q_2=O/s_1)} = \frac{(1-p_1)p_{21}}{(1-p_1)p_{21}+0} = 1$, and similarly given z_1, $p(q_2 = O/s_2) = \eta p(z_1/s_2, q_2 = O)p(q_2 = O/s_2) = 0$. Then, $p(q_2) = p_{21}b_1 + p_{22}b_2 = b_1 \neq 1$. Hence, even in this case, the true map probabilities are not the fixed point of the Bayesian update and hence, not consistent.

Now, suppose that we maintain the correlations between the pose s and the components q_1, q_2. The recursive update law is as follows:

$$p(s, q_1, q_2/\mathcal{F}^t) = \eta p(z_t/s, q_1, q_2)p(s, q_1, q_2/\mathcal{F}^{t-1}). \quad (53)$$

Now, we may verify that the true map components $p_1 = 0, p_2 = 1$ are indeed a fixed point of the above recursion. In order to show this, note that given the observation z_1 and the true map components, $p(z_1/s, q_1, q_2) = 0$ for all possible combinations of (s, q_1, q_2) except $s = s_1, q_1 = E, q_2 = O$. Hence, it follows that given the true map probabilities and the observation z_1, the map probabilities are updated to $p(s_1, q_1 = E, q_2 = O) = 1$ and hence

the true map probabilities $p_1 = 0, p_2 = 1$ do not change and thus, are fixed points of the above iteration. Similarly, it can be shown that given z_2, the true map probabilities do not change. Thus, we see that the only way that a consistent estimate of the map can be formed in the Bayesian approach, is by considering the joint distribution $p(s, q_1, q_2)$, i.e., the map components have to be correlated to each other as well as the robot pose. Note that the above does not prove that the method is consistent just that the true map probabilities are indeed fixed points of the recursion, which is more than can be said of the Bayesian recursions if the correlations are not considered.

5 Application: Simultaneous Localization and Mapping (SLAM)

In this section, we apply the frequentist mapping methodology to the simultaneous localization and mapping (SLAM) problem. The SLAM problem consists of a robot localizing itself within an unknown map while simultaneously trying to estimate the map. Typically a purely Bayesian approach is taken to the problem which leads to computation vs. consistency problem as mentioned in the introduction [11,12]. We adopt a "first localize -then map" philosophy to the problem which leads to a hybrid solution technique that avoids many of the pitfalls of the Bayseian approach.

We assume that the system localizes itself with respect to a sparse set of features/ landmarks $\Theta = \{\theta_1, \cdots, \theta_K\}$. Then, the belief (or probability distribution) over the pose-features pair is formed recursively using a Bayes filter (such as a Kalman filter in the Linear Gaussian case):

$$b_t(s, \Theta) = p(z_t^\theta / \Theta, s) \sum_{s'} p(s/s', u_{t-1}) b_{t-1}(s', \Theta), \quad (54)$$

where z_t^θ represents the noise corrupted observation of the landmarks Θ at time t, and u_{t-1} denotes the control acting on the system at time $t-1$. The identification and recognition of these features and landmarks in an autonomous fashion is a challenging problem in itself, but can be solved using suitable feature-based SLAM algorithms [14, 15, 32]. Given the joint distribution of the pose-landmark pair, the belief on the pose of the vehicle is formed by marginalizing the dependence on the landmarks, and is output to the frequentist part of the mapping algorithm. The frequentist part of the method, i.e., Estimator E2, is now used to map the rest of the (dense) environment using the belief output from the Bayesian part of the methodology. Thus, the hybrid methodology can be represented as the following algorithm:

Suppose that there are $M + N$ total components in a map. At the basic algorithmic level, in the Kalman filter based approach, the computational complexity is $O(N + M)^2$ in order to retain consistency. In the Hybrid formulation, suppose that N is the number of features that is used to localize the robot and M is the rest of the map. Then, at the basic algorithm level, the computational complexity of the hybrid approach is $O(N^2) + O(M)$ if a Kalman filter is used to localize the robot with respect to the features. Thus, if $N << M$, then the hybrid formulation possesses orders of magnitude better computational benefits compared to Bayesian methods such as the Kalman filter, while being provably consistent. Moreover, due to the sparseness of the landmark/ features, the data association problem for the Bayesian sub-problem (i.e., which observation is from which landmark) is significantly simpler. In conjunction with the immunity of the frequentist estimator to the data association problem, this leads to significantly improved robustness of the hybrid formulation to the data association problem.

Hybrid SLAM
Input $b_0(s)$, initial map occupancy probabilities $P_i(0)$, and reading of environment z_1, $t = 0$.
Do till convergence of map probabilities

Bayesian: Extract the readings of the landmarks, z_t^θ, from the raw sensor readings z_t, and form the belief on the state of the robot using Eq. (54) and marginalizing over the landmarks.
Frequentist: Take the rest of the data z_t^Q and update the occupancy probabilities of those components of the (dense) map Q that are observed given z_t^Q (cf. Eq. 20), using the frequentist estimator E2 (cf Eq. 26).

end

5.1 Experiments

In this section the hybrid methodology is applied to large maps with multiple cycles in them.. We have chosen to test our methods on such maps because of the well-known difficulties such maps pose to SLAM algorithms. The mobile robot is a differential drive vehicle. The equations of motion of such a robot is given by the following [33]:

$$\dot{x} = \frac{R}{2}(u_l + u_r)cos\theta, \quad (55)$$

$$\dot{y} = \frac{R}{2}(u_l + u_r)sin\theta, \quad (56)$$

$$\dot{\theta} = \frac{R}{L}(u_r - u_l), \quad (57)$$

where (x, y, θ) specify the pose of the robot,(u_l, u_r) are the left and right wheel angular velocities, and (R, L) are the radius of the wheel, and width of the robot respectively. The dimensions of the robot were as follows:

a wheel radius of 25 cm, and a width of 50 cm. The measurements equations for the features are:

$$r^{(k)} = \sqrt{(x - x^{(k)})^2 + (y - y^{(k)})^2},\ \phi^{(k)} = tan^{-1}\frac{(y - y^{(k)})}{(x - x^{(k)})} \quad (58)$$

where $(x^{(k)}, y^{(k)})$ denotes the location of the k^{th} feature and $(r^{(k)}, \phi^{(k)})$ represents the range and bearing to the k^{th} feature, relative to the sensor location. Experiments are performed for two different kinds of sensors: a) a noisy sensor with both range and bearing errors (sonar) with $\sigma_r = 0.2$ m and $\sigma_\theta = 0.6$ deg, and b) an accurate range sensor such as a SICK (brand) laser range sensor with $\sigma_r = 0.01$ m and $\sigma_\theta = 0.05$ deg. The maximum range of the sensors was assumed to be 40 m. We assumed that there were feature points in the map that could be identified from the raw sensory data using suitable signal processing techniques. Mostly, these feature points were assumed to be either corners, or mid-points of corridors/ straight line segments. However, in our simulations, we do not identify these feature points from the raw sensory data as our intent is to show the efficacy of our methodology, and not the autonomous identification of such feature points. Any suitable feature-based SLAM algorithm can be used to achieve this goal. The observation of the feature-points are used in an Extended Kalman Filter (EKF) to form the belief on the robot pose, i.e., solve the Bayesian sub-problem of the hybrid formulation. Next, the belief state is used to map the rest of the dense map using the frequentist estimator E2. The process noise in the wheel encoders is $\sigma_u = 0.5$ rad/s. The average robot wheel speed is 5 rad/sec and the integration time step for the EKF is 0.5 sec.

In the following, we show how the observation models $A(s)$ that are to required in the frequentist estimators E1/ E2 are extracted from the sensor noise models. We show the case of a range sensor corrupted by Gaussian noise. The method can be extended to sensors with both range and bearing errors in a relatively straightforward fashion.

Suppose the robot is at some point A and the observation is made along the ray AP (see Fig. 3). The senor gives a range reading along the ray AP, and is assumed to be corrupted with zero mean Gaussian noise. Let q_k be some grid on AP. We want to evaluate the probabilities $p(\hat{q}_k = O/\hat{q}_k = O)$ and $p(\hat{q}_k = O/\hat{q}_k = E)$, given that the observation is made along ray AP. Note that AP denotes the pose of the robot since it encodes the location of the sensor as well as the orientation of the sensor. Let the grids between A and q_k be denoted by $q_{l1}, q_{l2}, \cdots q_{ln}$ and the grids beyond q_k as $q_{h1}, \cdots q_{hm}$. We can calculate the required probabilities using the following formulas which are a direct consequence of the law of total probability, and the assumption of the mutual independence of the different map components :

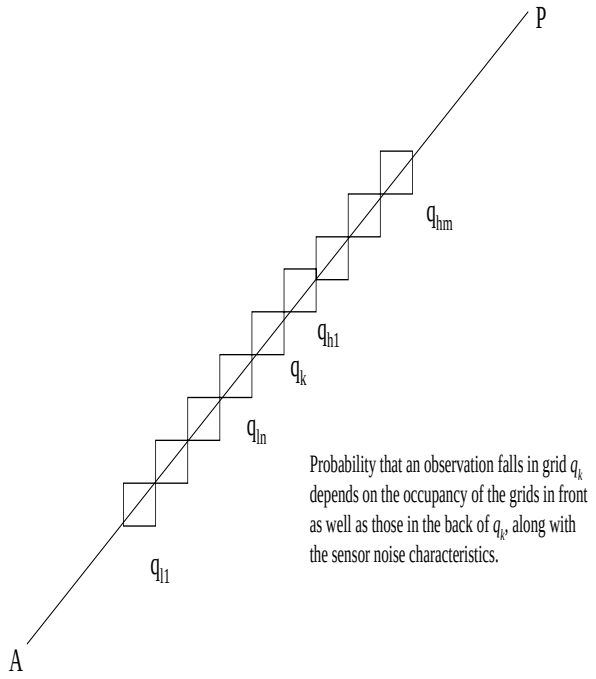
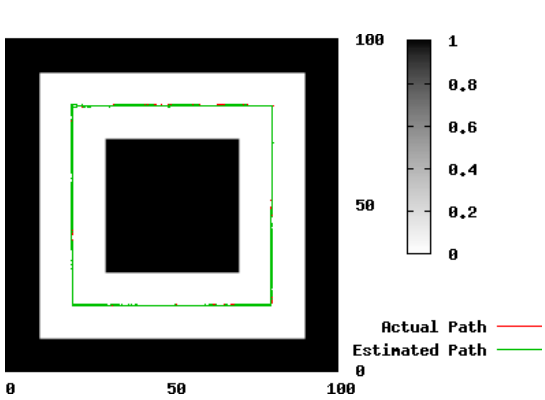


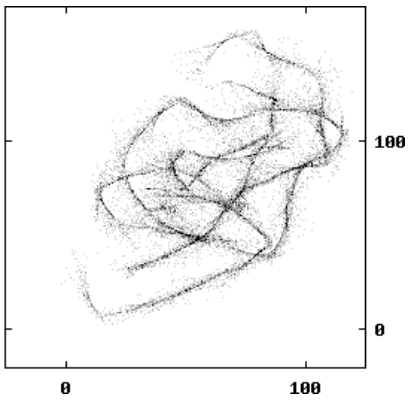
Fig. 3. Observation Model for a Range Sensor

$$\begin{aligned}
P(\hat{q}_k=O/q_k=O) = & P(\hat{q}_k=O/q_{l1}=O)P(q_{l1}=O) \\
& + P(\hat{q}_k=O/q_{l1}=E,q_{l2}=O)P(q_{l2}=O)P(q_{l1}=E) + \ldots \\
& + P(\hat{q}_k=O/q_{l1}=E,\ldots,q_{l(n-1)}=E,q_{ln}=O)P(q_{ln}=O) \\
& \quad \prod_{i=1}^{n-1} P(q_{li}=E) \\
& + P(\hat{q}_k=O/q_{l1}=E,\ldots,q_{ln}=E,C_k=O)\prod_{i=1}^{n} P(q_{li}=E)
\end{aligned} \quad (59)$$

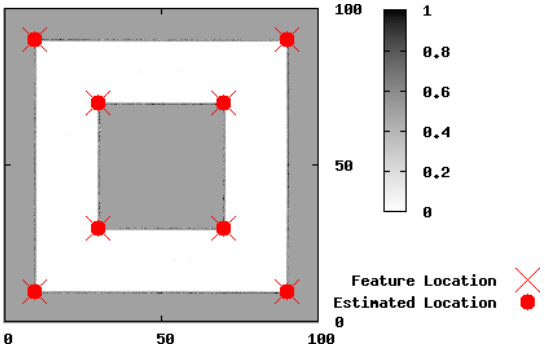
$$\begin{aligned}
P(\hat{q}_k=O/q_k=E) = & P(\hat{q}_k=O/q_{l1}=O)P(q_{l1}=O) \\
& + P(\hat{q}_k=O/q_{l1}=E,q_{l2}=O)P(q_{l2}=O)P(q_{l1}=E) + \ldots \\
& + P(\hat{q}_k=O/q_{l1}=E,\ldots,q_{l(n-1)}=E,q_{ln}=O)P(q_{ln}=O) \\
& \quad \prod_{i=1}^{n-1} P(q_{li}=E) \\
& + P(\hat{q}_k=O/q_{l1}=E,\ldots,q_{ln}=E,q_{h1}=O)P(q_{h1}=O) \\
& \quad \prod_{i=1}^{n} P(q_{li}=E) \\
& + P(\hat{q}_k=O/q_{l1}=E,\ldots,q_{ln}=E,\hat{q}_k=E,q_{h1}=E,q_{h2}=O) \\
& \quad P(q_{h2}=O)P(q_{h1}=E)\prod_{i=1}^{n} P(q_{li}=E) + \ldots \\
& + P(\hat{q}_k=O/q_{l1}=E,\ldots,q_{ln}=E,\hat{q}_k=E,q_{h1}=E,\ldots,q_{hm}=O)
\end{aligned}$$



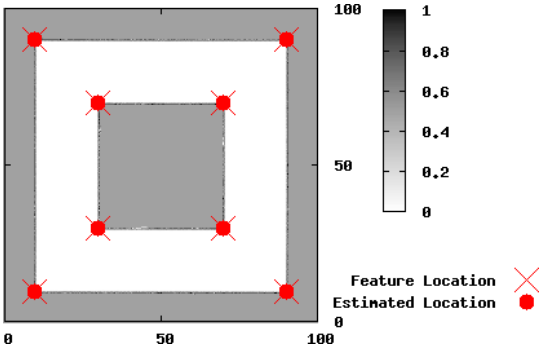
(a) Original map



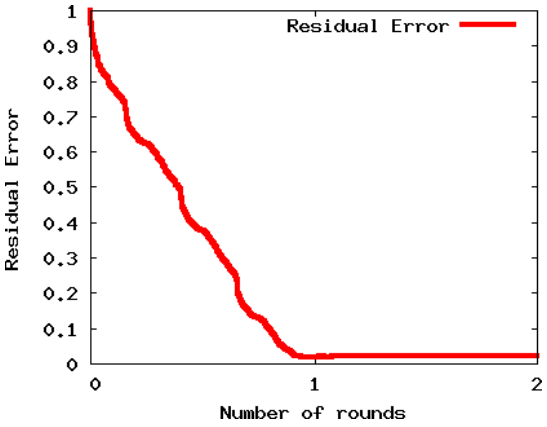
(b) Raw odometry map



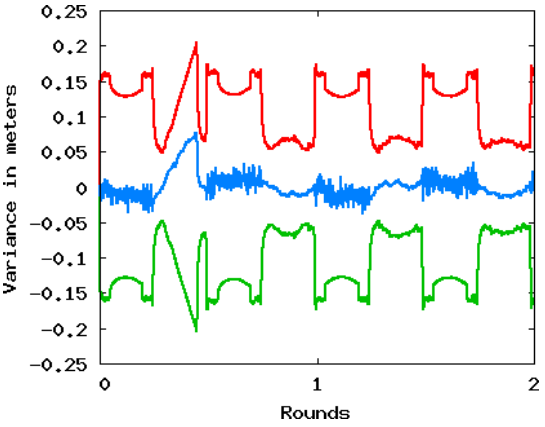
(c) Final map after one round



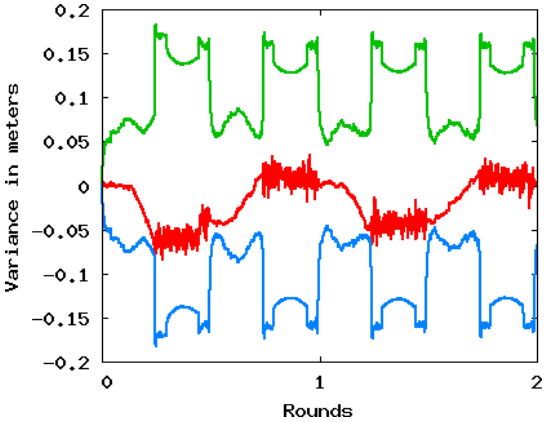
(d) Final map after two rounds



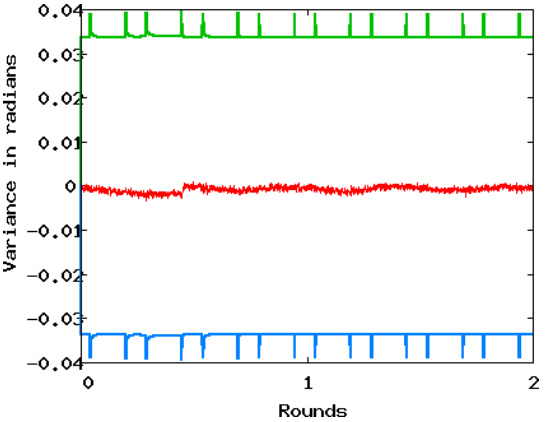
(e) Residual error plot



(f) x error and 3σ envelop

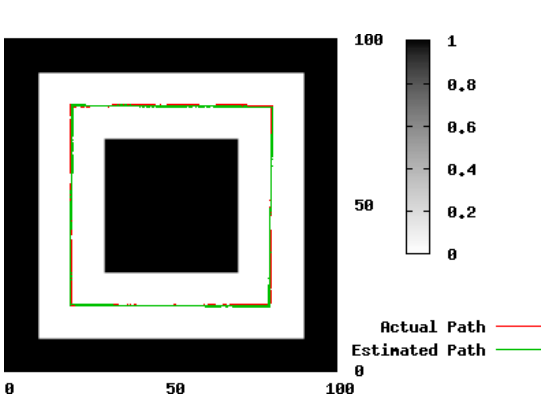


(g) y error and 3σ envelop

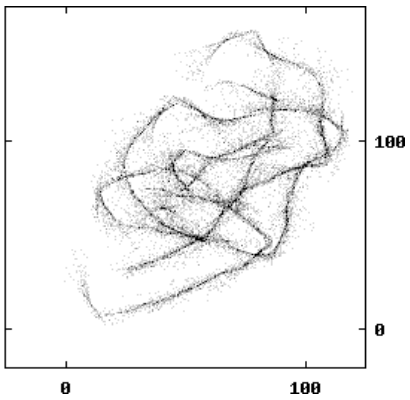


(h) ϕ error and 3σ envelop

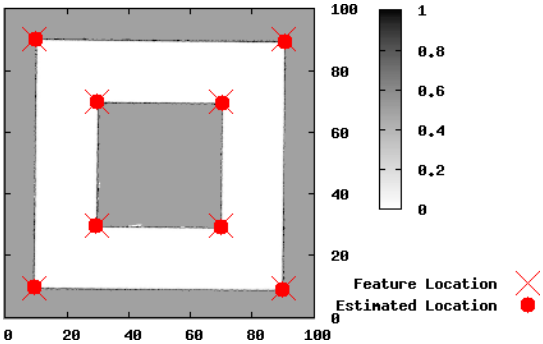
Fig. 4. Experimental results for Map 1 with accurate sensor



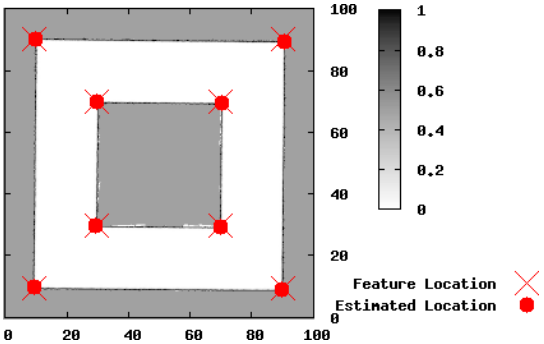
(a) Original map



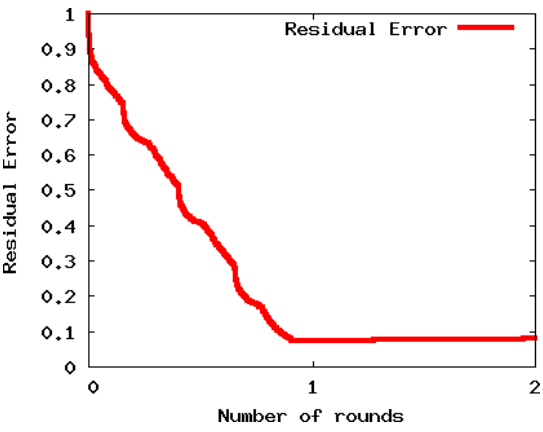
(b) Raw odometry map



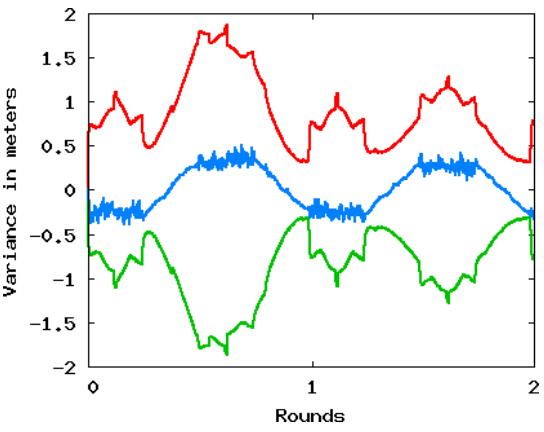
(c) Final map after one round



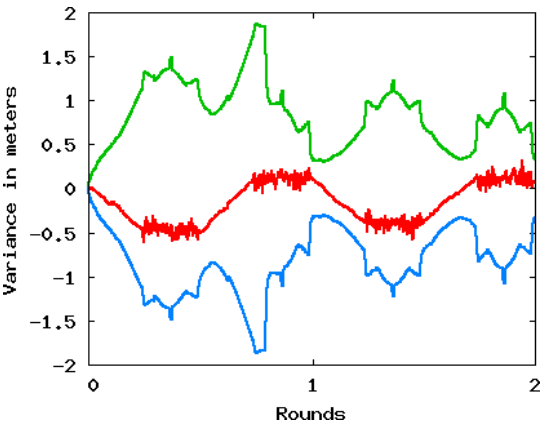
(d) Final map after two rounds



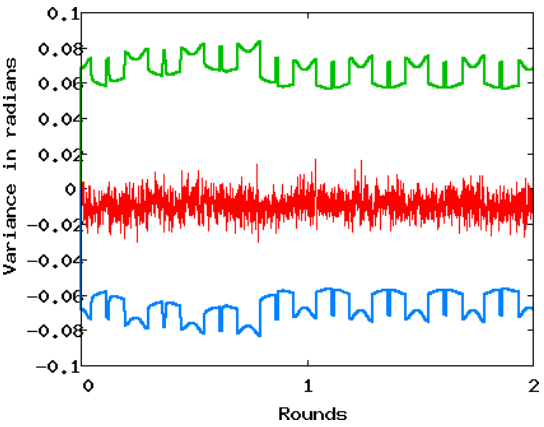
(e) Residual error plot



(f) x error and 3σ envelop

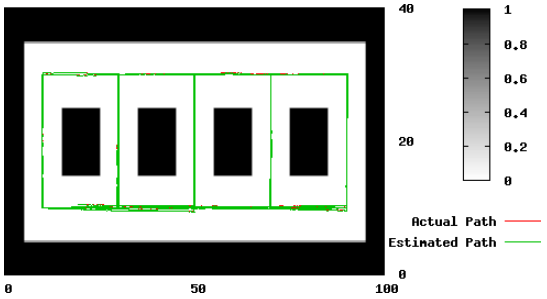


(g) y error and 3σ envelop

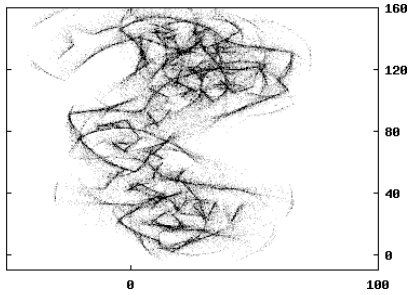


(h) ϕ error and 3σ envelop

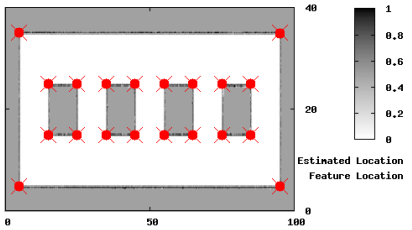
Fig. 5. Experimental results for Map 1 with noisy sensor



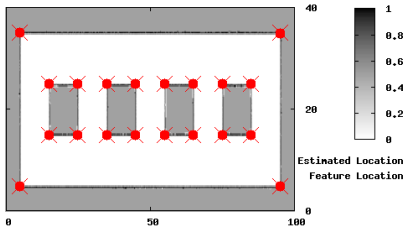
(a) Original map



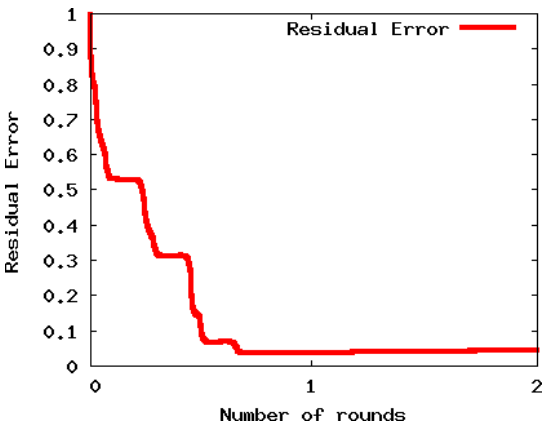
(b) Raw odometry map



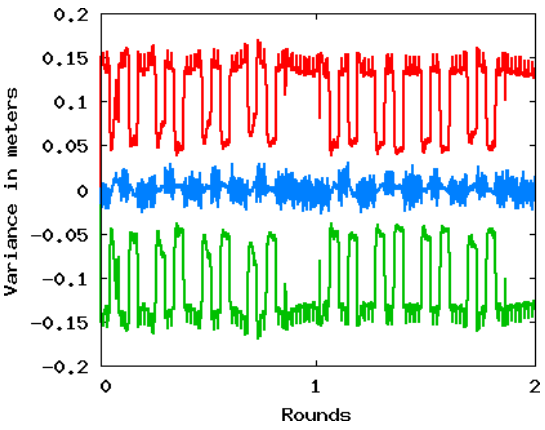
(c) Final map after one round



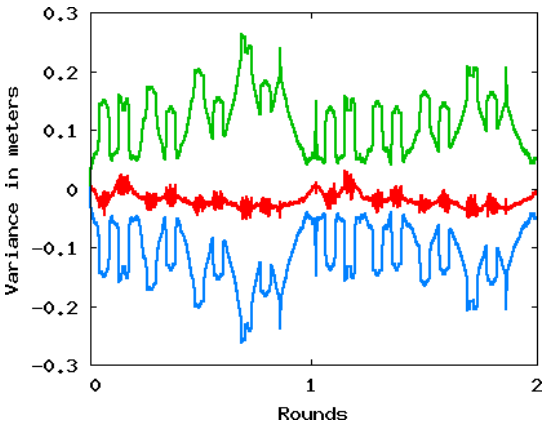
(d) Final map after two rounds



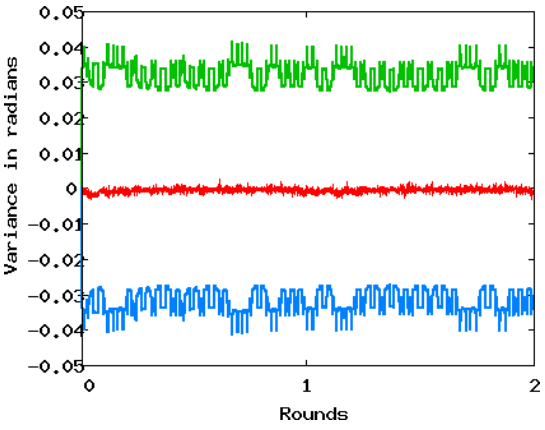
(e) Residual error plot



(f) x error and 3σ envelop

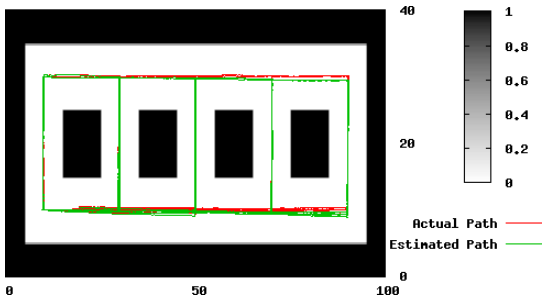


(g) y error and 3σ envelop

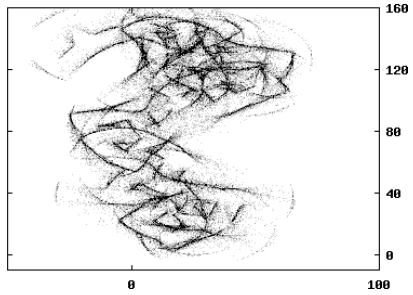


(h) ϕ error and 3σ envelop

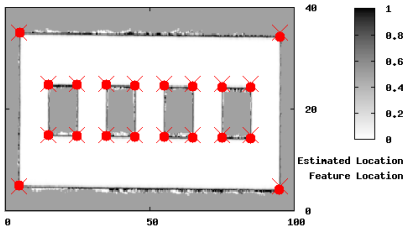
Fig. 6. Experimental results for Map 2 with accurate sensor



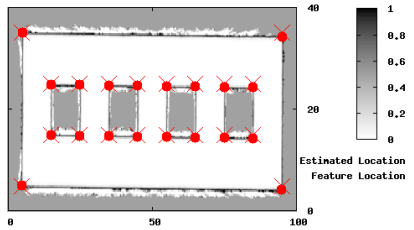
(a) Original map



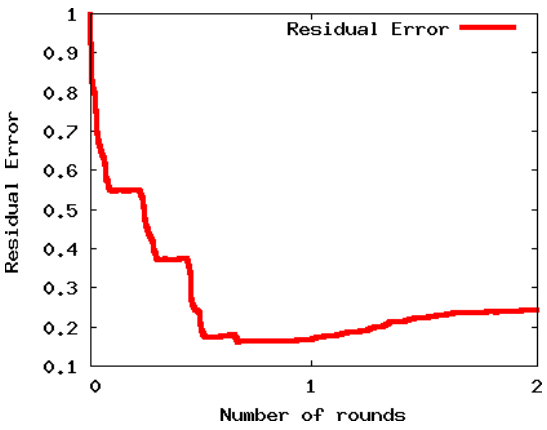
(b) Raw odometry map



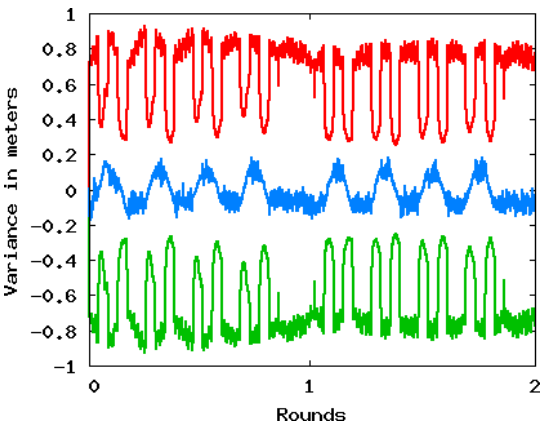
(c) Final map after one round



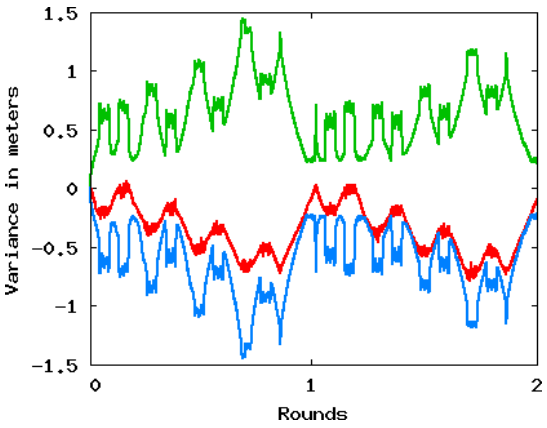
(d) Final map after two rounds



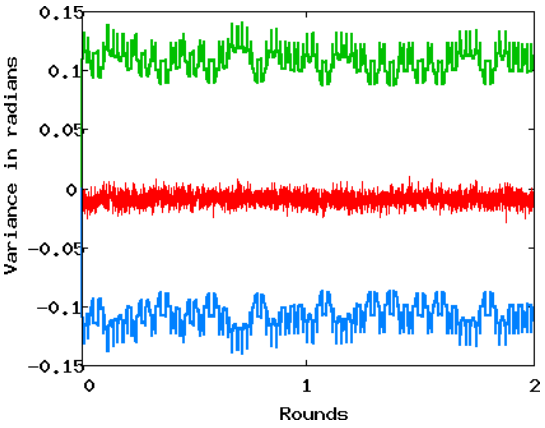
(e) Residual error plot



(f) x error and 3σ envelop



(g) y error and 3σ envelop



(h) ϕ error and 3σ envelop

Fig. 7. Experimental results for Map 2 with noisy sensor

$$P(q_{hm}=O)\prod_{i=1}^{m-1}P(q_{hj}=E)\prod_{i=1}^{n}P(q_{li}=E) \tag{60}$$

Note that the quantities $P(\hat{q}_k = O/q_{l1} = O)$, $P(\hat{q}_k = O/q_{l1} = E, q_{l2} = O)$ etc. denote the probability that the observation falls in a particular grid given that the obstacle was in some other grid. These probabilities can be easily calculated using the noise model of the sensor. The observation matrix A_k for a grid q_k is then given by:

$$A_k = \begin{pmatrix} P(\hat{q}_k = O/q_k = O) & P(\hat{q}_k = O/q_k = E) \\ P(\hat{q}_k = E/q_k = O) & P(\hat{q}_k = E/q_k = E) \end{pmatrix}$$

$P(\hat{q}_k = E/q_k = O) = 1 - P(\hat{q}_k = O/q_k = O)$ and $P(\hat{q}_k = E/q_k = E) = 1 - P(\hat{q}_k = E/q_k = E)$. Note here that the observation matrix is dependent on the real probabilities of the map components $P(q_K = O)$ and $P(q_k = E)$.

Figs 4-7 show the results of our simulation experiments. Two different maps are considered in these experiments. The first map (Map 1) consists of a large cyclic corridor of side 100m while Map 2 is a long hallway (100m x 40m) with 4 cyclic corridors. A total of 2 laps of each map is made. The total length of the runs was approximately 1 km for Maps 1 and 2. Each of these maps are sensed using both the noisy sensor, as well as the accurate Laser range sensor. Figs 4-5 represent the results for Map 1 while Figs 6-7 represent the results for Map 2. In each of these figures, Subfigure (a) shows the original map along with the actual as well as the estimated robot trajectory, Subfigure (b) shows the raw odometry data, Subfigures (c) and (d) shows the estimated map, along with the features and their estimates, after the completion of one and two rounds respectively. In Subfigure (e), we show the total error in the map as a function of the number of rounds the robot makes and is defined as the sum of the absolute values of the component wise error between the estimated map and the true map divided by the total number of grids. This may be interpreted as the fraction of the map that is unconverged. Subfigures (f)-(h) show the error in the estimates of the x, y, and θ co-ordinates of the robot along with their associated 3σ uncertainty bounds. The features used in these maps were the corners of the corridors and were assumed to be reliably identified.

These figures give us an idea as to how well the algorithm is performing and also give us valuable practical insight into the algorithm. The reason we chose these examples is because of the well-known challenge maps with multiple cycles pose to SLAM algorithms which is evidenced from the raw odometry plots (Subfigure (b) in the plots). These raw data plots show that pure odometry is not enough for accomplishing the mapping and localization tasks. The results show that the algorithm is able to map a large area with multiple cycles without much of a problem even though the sensors are noisy. Qualitatively (Subfigures (c), (d) in the plots), in that the edges of the maps are much sharper, as well as quantitatively (Subfigure (e) in the plots), in that the error is approximately in the range 0.10-0.25 for the noisy sensor while it is less than 0.05 for the accurate sensor, the results improve with the more accurate Laser range sensor when compared to the noisy sensor. Thus, this reinforces the intuitive idea that much larger maps can be mapped with an accurate sensor such as the Laser range sensor when compared to a noisier sensor such as sonar. From Subfigure (e), it should be clear that the estimate of the map practically converges within one round. This is the case since the number of observations of any component that is in the robot's field of view is high enough during the first round. It can also be seen from the total map error plots (Subfigure (e)) that the mapping algorithm seems to converge exponentially fast. A rigorous proof of this observation is beyond the scope of this paper and is left for future work. The algorithm had no problems in closing large loops as the ones shown here and we did not have to make any heuristic corrections when such a loop was closed. In fact, the size of the map, or the number of cycles in it, is really never a problem for this method as long as the EKF remains consistent. In subfigures (f)-(h), the true errors in the estimates of the pose of the robot remains within the 3σ uncertainty bounds and show that the EKF used for the Bayesian sub-problem does indeed remain consistent. This is not a rigorous test of the consistency of an EKF but is routinely used in the filtering community as a test for the consistency of a filter. The hybrid algorithm has also been tested on several other maps with satisfactory results which we cannot show here due to the paucity of space. We note here that in Fig. 7, Subfig. (g), the error in the estimates in the y direction is very close to the 3σ boundary and thus, the EKF starts to become inconsistent. This results in the entire hybrid algorithm becoming inconsistent and is evidenced in the relatively high error (≈ 0.25) in the figure when compared to the other maps (< 0.1). Thus, maintaining the consistency of the Bayesian filter is key to the overall performance of the hybrid scheme.

Thus, we may conclude the following from the experiments above: 1) the theoretical results presented in the first part of this paper are borne out by the experiments, 2) larger maps can be estimated using more accurate sensors, 3) the estimator is seen to converge exponentially fast to the true map probabilties, and 4) the method results in consistent maps as long as the Bayesian estimator used in the hybrid scheme (in this case, the EKF) remains consistent. Presumably, the consistency of the Bayesian estimator can be maintained in larger maps if advanced nonlinear filtering techniques such as the un-

scented Kalman filter (UKF), mixture of Gaussian filters and nonlinear filters based on solution of the Fokker-Planck equation.

6 Conclusion

We have proposed a frequentist approach to mapping with mobile sensors under uncertainty and applied the methodology to the SLAM problem of Robotics. We have proved the consistency of the algorithm and shown that it is robust to the data association problem while having complexity linear in the map components. The hybrid SLAM methodology was tested on several large maps and the experimental results confirm the applicability of the mapping methodology. In future research, we shall test the algorithms on larger and more complicated maps than the ones presented in the paper, as well as experimentally evaluate it using real data. We shall extend the methodology to maps with dynamic elements and the distributed multiple robot mapping problem. We shall also consider the problem of estimating dynamic spatially correlated maps.

Acknowledgements

This work was supported in part by Army Research Office (ARO) grant number W911NF-07-1-0429. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the ARO or the U.S. Government.

References

[1] T. H. Chung *et al.*, "On a decentralized active sensing strategy using mobile sensor platforms in a network," in *Proc. IEEE Int. Conf on Dec. Cont.*, 2004.

[2] S. Aranda *et al.*, "On optimal sensor placement and motion coordination for target tracking," in *Proc. IEEE Int. conf. Dec. Cont.*, 2005.

[3] R. Olfati-Saber, "Distributed tracking for mobile sensor networks with information-driven mobility," in *Proc. Amer. Cont. Conf. (ACC)*, 2007.

[4] G. M. Hoffman *et al.*, "Mutual information methods with particle filters for mobile sensor network control," in *Proc. IEEE CDC*, 2006.

[5] R. Bajcsy, "Active perception," *Proceedings of the IEEE*, vol. 76, pp. 996–1005, 1988.

[6] P. Ogren, E.Fiorelli, and N. E. Leonard, "Cooperative control of mobile sensor networks: Adaptive gradient climbing in a distributed environment," *IEEE Transactions on Automatic Control*, vol. 49, pp. 1292–1302, 2004.

[7] S. Martinez, "Distributed representation of spatial fields through an adaptive interpolation scheme," in *Proceedings of the ACC*, 2007.

[8] S. Martinez, J. Cortes, and F. Bullo, "Motion co-ordination using distributed information," *IEEE Control Systems Magazine*, vol. 27, pp. 75–88, 2007.

[9] R. Olfati-Saber, "Distributed kalman filter with embedded consensus filter," in *Proc. IEEE CDC*, 2005.

[10] R. Olfati-Saber and J. S. Shamma, "Consensus filters for sensor networks and distributed sensor fusion," in *Proc. IEEE CDC*, 2005.

[11] T.Bailey and H. Durrant-Whyte, "Simultaneous localization and mapping (slam): Part ii," *IEEE Robotics and Automation Magazine*, pp. 108–117, Sep. 2006.

[12] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping (slam): Part i," *IEEE Robotics and Automation Magazine*, pp. 99–108, June 2006.

[13] A. Elfes, "Using occupancy grids for mobile robot perception and navigation," *IEEE Computer*, vol. 22, pp. 46–57, 1989.

[14] S. Thrun, Y. Liu, D. Koeller, A. Y. Ng, S. Ghahramani, and H. Durrant-Whyte, "Simultaneous localization and mapping with sparse extended information filters," *International Journal of Robotics Research*, vol. 23, pp. 693–716, 2004.

[15] R. Eustice, H. Singh, and J. Leonard, "Exactly sparse delayed state filters," in *Proc. IEEE Int. Conf. Rob. Aut.*, 2005.

[16] M. Montemerlo, S. Thrun, and W. Whittaker, "Conditional particle filters for simultanesous mobile robot localization and people tracking," in *IEEE ICRA*, 2002.

[17] K. Murphy, "Bayesian map learning in dynamic environments," in *Advances in Neural Information Processing Systems (NIPS)*, 1999.

[18] F. Lu and E. Milios, "Globally consistent range scan alignment for environment mapping," *Autonomous Robots*, vol. 18, pp. 249–275, 1997.

[19] S. Thrun and M. Montemerlo, "The GraphSLAM algorithm with applications to large-scale mapping of urban structures," *International Journal on Robotics Research*, vol. 25, no. 5/6, pp. 403–430, 2005.

[20] Y. Ephraim and N. Merhav, "Hidden markov processes," *IEEE Transactions Inf. Theory*, vol. 48, pp. 1518–1569, 2002.

[21] R. Martinez-Cantin, N. deFreitas, and J. A. Castellanos, "Analysis of particle methods for simultaneous robot localization and mapping and a new algorithm: Marginal-slam," in *Proceedings of the IEEE Int. Conf. Rob. automat.*, 2007.

[22] M. Klaas, N. DeFreitas, and A. Doucet, "Towards practical n^2 monte carlo: the marginal filter," in *Uncertainty in Artificial Intelligence*, 2005.

[23] S. Chakravorty and J. L. Junkins, "Motion planning in uncertain environments with vision like sensors," *Automatica*, vol. 43, pp. 2104–2111, 2007.

[24] J. Davis and S. Chakravorty, "Motion planning under uncertainty: Application to an unmanned helicopter," *Journal of Guid., Contrl. and Dyn.*, vol. 30, pp. 1268–1277, 2007.

[25] S. Chakravorty, "A hybrid approach to the simultaneous planning localization and mapping problem," in *Proceedings of the American Control Conference*, 2008.

[26] A. Benveniste, M. Metivier, and P. Priouret, *Adaptive Algorithms and Stochastic Approximations*. Berlin Hiedelberg: Spronger-Verlag, 1990.

[27] H. J. Kushner and G. G. Yin, *Stochastic Approximation and Recursive Algorithms and Applications*. New york, NY: Springer, 2003.

[28] D. P. Bertsekas and J. N. Tsitsiklis, *Neuro-Dynamic Programming*. Boston, MA: Athena Scientific, 1996.

[29] S. Haykin, *Neural Networks*. Upper Saddle River, NJ: Prentice-Hall, 1994.

[30] H. K. Khalil, *Nonlinear Systems*. Upper Saddle River, NJ: Prentice Hall, 1996.

[31] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge, UK: Cambridge University Press, 1993.

[32] J. Leonard and H. Durrant-Whyte, *Directed Sonar Sensing for Mobile Robot Navigation*. Boston, MA: Kluwer academic, 1992.

[33] S. M. LaValle, *Planning Algorithms*. Cambridge, UK: Cambridge University Press, 2005.