Exactly Sparse Extended Information Filters for Feature-Based SLAM

Matthew R. Walter, Ryan M. Eustice, and John J. Leonard

Abstract

Recent research concerning the Gaussian canonical form for Simultaneous Localization and Mapping (SLAM) has given rise to a handful of algorithms that attempt to solve the SLAM scalability problem for arbitrarily large environments. One such estimator that has received due attention is the Sparse Extended Information Filter (SEIF) by Thrun *et al.*, which is reported to be nearly constant time, irrespective of the size of the map. The key to the SEIF's scalability is to prune weak links in what is a dense information (inverse covariance) matrix to achieve a sparse approximation that allows for efficient, scalable SLAM. We demonstrate that the SEIF sparsification strategy yields error estimates that are overconfident when expressed in the global reference frame, while empirical results show that relative map consistency is maintained.

In this paper, we propose an alternative scalable estimator based in the information form that maintains sparsity while preserving consistency. The paper describes a method for controlling the population of the information matrix, whereby we track a modified version of the SLAM posterior, essentially by ignoring a small fraction of temporal measurements. In this manner, the Exactly Sparse Extended Information Filter (ESEIF) performs inference over a model that is conservative relative to the standard Gaussian distribution. We compare our algorithm to the SEIF and standard EKF both in simulation as well as on two nonlinear datasets. The results convincingly show that our method yields conservative estimates for the robot pose and map that are nearly identical to those of the EKF.

Index Terms

mobile robotics, SLAM, Kalman filters, information filters, robotic mapping, and robotic navigation.

I. Introduction

The capability to accurately navigate in *a priori* unknown environments is critical for autonomous robotics. Using a suite of inertial and velocity sensors, dead-reckoning provides position estimates subject to unbounded error growth with time. In some outdoor applications, one can utilize GPS fixes to periodically minimize this error. Unfortunately, GPS measurements are not available in many common environments (e.g. indoors and underwater), thus requiring an alternative means of keeping the error drift in check. Underwater vehicles, for example, often rely upon acoustic long-baseline (LBL) range measurements that are fused with motion sensor data [1]. Utilizing LBL navigation requires the deployment and calibration of a transponder network and limits the operating range of the vehicle. The need for such an infrastructure constrains the degree of autonomy that underwater robots are able to achieve.

Simultaneous Localization and Mapping (SLAM) offers a solution for unencumbered navigation that exploits the environment to maintain accurate pose estimates. By building a map on-line while using inertial and velocity measurements to predict vehicle motion, the robot utilizes observations of the environment to localize itself within the map. The stochastic nature of the vehicle motion and measurement models, together with noisy sensor data, further complicates the coupling between navigation and mapping that is inherent to SLAM. Many successful SLAM algorithms address these issues by formulating the problem in a probabilistic manner, tracking the joint posterior over the vehicle pose and map.

M. Walter is the corresponding author. He and J. Leonard are with the Computer Science and Artificial Intelligence Laboratory at the Massachusetts Institute of Technology, Cambridge, MA, USA. Email: {mwalter, jleonard}@mit.edu

R. Eustice was with the Joint Program in Oceanographic Engineering of the Massachusetts Institute of Technology, Cambridge, MA, and the Woods Hole Oceanographic Institution, Woods Hole, MA at the time of this work; he is presently with the Department of Naval Architecture and Marine Engineering at the University of Michigan, Ann Arbor, MI, USA. Email: eustice@umich.edu

In their seminal paper [2], Smith *et al.* show how this distribution can be modeled by a Gaussian that is completely described by a mean vector and covariance matrix, and tracked via an Extended Kalman Filter (EKF). In part as a result of its relative simplicity, this model has become the standard tool of choice for a majority of SLAM algorithms. With explicit knowledge of the correlation between the robot state and the map elements, the EKF exploits observations of the environment to improve the vehicle pose and map estimates. Maintaining these correlations, though, imposes an $\mathcal{O}(n^2)$ memory requirement, where n is proportional to the size of the map [3]. Furthermore, while the EKF efficiently predicts the vehicle motion, measurement updates for the standard EKF are quadratic in the number of states. As a consequence, it is well known that the standard EKF SLAM algorithm is limited to relatively small environments (i.e. on the order of a few hundred features) [4].

As robots are deployed in larger environments, extensive research has focused on the scalability restrictions of EKF SLAM. An intuitive way of dealing with this limitation is to divide the world into numerous sub-environments, each comprised of a more manageable number of l features. These submap approaches [5]–[8] shed some of the computational burden of the full EKF solution by performing estimation based only upon the robot's local neighborhood. The performance time for the Kalman updates is then $\mathcal{O}(l^2)$ rather than the standard $\mathcal{O}(n^2)$. One tradeoff of focusing on individual local maps is that some methods forgo an immediate estimate for the global map and, in general, are believed to sacrifice convergence speed [7]. Alternatively, the FastSLAM algorithm [9] takes advantage of the conditional independence of the map elements given knowledge of the robot pose to improve scalability. Employing a particle filter representation for the pose distribution, FastSLAM efficiently tracks a map for each particle with a collection of independent EKFs, one for every feature. The computational cost is proportional to the number of particles used and is low in situations where relatively few particles are sufficient to describe the robot pose. With larger uncertainties, though, the efficiency benefits are not as obvious since an increased number of particles is necessary [10] to characterize the pose distribution. Additionally, there is the problem of particle depletion [11] as particles describing the true trajectory are lost due to resampling.

Recently, strategies have emerged that offer the promise of scalability through the canonical parametrization for the SLAM distribution. Rather than a dense covariance matrix and mean vector, the canonical form completely describes the Gaussian by the information (inverse covariance) matrix and information vector. Analogous to the EKF, the evolution of the posterior is tracked over time via a two step process comprising the so-called Extended Information Filter (EIF) [12]. The dual of the covariance form, the EIF update step is efficient as it is quadratic in the number of measurements[1] and not the size of the map. On the other hand, the time projection step is, in general, quadratic in the number of landmarks. Also, recovering the mean from the information vector and matrix requires a costly $\mathcal{O}(n^3)$ matrix inversion. Together, these characteristics would seem to rule out the information parametrization as a viable remedy to the scalability problem of the standard EKF and are largely the reason for its limited application to SLAM.

Pivotal insights by Thrun *et al.* [13] and Frese *et al.* [14] reveal that the canonical form is, in fact, particularly beneficial in the context of feature-based SLAM as a majority of the off-diagonal elements in the normalized information matrix are inherently very small. By essentially approximating these entries as being zero, Thrun *et al.* take advantage of what is then a sparse information matrix, presenting the Sparse Extended Information Filter (SEIF), an adaptation of the EIF. In addition to incorporating new observations efficiently, the SEIF performs the time projection step at a significant savings in cost, offering a near constant-time solution to the SLAM problem. The caveat is that a subset of the mean is necessary to linearize the motion and measurement models as well as to enforce the sparsity of the information matrix. To that end, the authors estimate the mean of the robot pose and a limited number of features as the solution to a sparse set of linear equations that is approximated using relaxation.

Similar benefits extend from interpreting the canonical parametrization as a Gaussian Markov Random

[1]This assumes knowledge of the mean, which is necessary for observations that are nonlinear in the state.

Field (GMRF) [15] where small entries in the normalized information matrix correspond to weak links in the graphical model. By essentially breaking these weak links, Paskin [10] and Frese [16] approximate the graphical model with a sparse tree structure. Paskin's Thin Junction Tree Filter (TJTF) and Frese's Treemap filter are then each capable of efficient inference upon this representation, involving $\mathcal{O}(n)$ and $\mathcal{O}(\log n)$ time, respectively.

Recently, a number of batch algorithms have been proposed that solve for the maximum likelihood estimate (MLE) based upon the entire history of robot motion and measurement data [4], [17]–[20]. They solve for the MLE by optimizing a full, nonlinear log-likelihood function over a series of iterations, which provides robustness to linearization and data association errors. GraphSLAM [4], in similar fashion to the other batch algorithms, forms a graph in which the nodes correspond to the robot poses and map elements and the edges capture the motion and measurement constraints. At each iteration, the linearization yields a sparse information matrix to which they apply the variable elimination algorithm to marginalize over the map and reduce the graph to one over only the pose history. They subsequently solve for the posterior over the pose history and, in turn, the current ML map estimate.

Dellaert [20] adopts much the same strategy as it considers the posterior over the robot's entire pose history, taking advantage of what is then a naturally sparse information matrix [10], [21]. The approach formulates the corresponding mean as the solution to a set of linear equations for which the information matrix is the matrix of coefficients. The technique decomposes the information matrix into either its Cholesky or QR factorization, paying close attention to variable ordering. In turn, they jointly solve for the mean over the pose and map via back-substitution. As the author insightfully shows, this closely parallels aspects of the aforementioned graphical model methods. The results demonstrate promising advantages over the EKF with regards to performance though the algorithm currently does not address some important aspects of the SLAM problem such as data association.

Alternatively, Wang *et al.* [22] treat map building and localization as two separate estimation problems. They represent the distribution over the map as a canonical Gaussian that is maintained using measurements of the relative pose between pairs of landmarks. The advantage of sacrificing robot pose information is that the information matrix for the map is then naturally sparse. Meanwhile, the robot is continuously relocated within the map based upon observations of features. This estimate is fused with that of a standard EKF, which concurrently performs local SLAM, via covariance intersection [23] to estimate the robot pose. There are a number of similarities between this algorithm and the approach presented in this paper although the two approaches have been developed independently of one another.

The benefit of maintaining the *joint* distribution over the robot and map is that we can take advantage of dependence relationships between landmarks and the vehicle pose. Unfortunately, the consequence is that, while the information matrix is relatively sparse, it is nonetheless fully populated. In this paper, we analyze the process by which the SEIF actively breaks weak robot-landmark links to enforce a desired level of sparsity. We show that a consequence of the SEIF sparsification is an overconfident estimate for the global map and pose errors while the consistency of the local map relations is preserved.

As a remedy, we propose an efficient information-based formulation of the SLAM problem that actively controls the population of the information matrix without relying upon an approximation to the state distribution. By essentially relocalizing the robot within the map, we show that the filter maintains exact analytical sparseness while producing state estimates that are both globally *and* locally conservative relative to the full EKF solution. We evaluate the performance of our algorithm alongside the SEIF and full KF in a controlled linear Gaussian simulation to reveal the consistency traits inherent to the estimators. We then demonstrate our filter alongside the SEIF and full EKF on a pair of real-world datasets, including a benchmark nonlinear experiment. The results reveal that the SEIF is globally inconsistent while our algorithm yields estimates for the robot pose and map nearly identical to those of the EKF, yet both globally and locally conservative.

II. Background

Let $\boldsymbol{\xi}_t$ be a random vector governed by a multivariate Gaussian probability distribution, $\boldsymbol{\xi}_t \sim \mathcal{N}\big(\boldsymbol{\mu}_t, \Sigma_t\big)$, traditionally parametrized in full by the mean vector, $\boldsymbol{\mu}_t$, and covariance matrix, Σ_t. Expanding the quadratic term within the Gaussian exponential, we arrive at an equivalent representation for the multivariate distribution, $\mathcal{N}^{-1}\big(\boldsymbol{\eta}_t, \Lambda_t\big)$.

$$\begin{aligned} p\left(\boldsymbol{\xi}_t\right) &= \mathcal{N}\big(\boldsymbol{\mu}_t, \Sigma_t\big) \\ &\propto \exp\big\{-\tfrac{1}{2}(\boldsymbol{\xi}_t - \boldsymbol{\mu}_t)^\top \Sigma_t^{-1}(\boldsymbol{\xi}_t - \boldsymbol{\mu}_t)\big\} \\ &= \exp\big\{-\tfrac{1}{2}\left(\boldsymbol{\xi}_t^\top \Sigma_t^{-1}\boldsymbol{\xi}_t - 2\boldsymbol{\mu}_t^\top \Sigma_t^{-1}\boldsymbol{\xi}_t + \boldsymbol{\mu}_t^\top \Sigma_t^{-1}\boldsymbol{\mu}_t\right)\big\} \\ &\propto \exp\big\{-\tfrac{1}{2}\boldsymbol{\xi}_t^\top \Sigma_t^{-1}\boldsymbol{\xi}_t + \boldsymbol{\mu}_t^\top \Sigma_t^{-1}\boldsymbol{\xi}_t\big\} \\ &= \exp\big\{-\tfrac{1}{2}\boldsymbol{\xi}_t^\top \Lambda_t\boldsymbol{\xi}_t + \boldsymbol{\eta}_t^\top \boldsymbol{\xi}_t\big\} \propto \mathcal{N}^{-1}\big(\boldsymbol{\eta}_t, \Lambda_t\big) \end{aligned} \tag{1}$$

The canonical form of the Gaussian (1) is completely parametrized by the information matrix, Λ_t, and information vector, $\boldsymbol{\eta}_t$, which are related to the mean vector and covariance matrix by (2).

$$\Lambda_t = \Sigma_t^{-1} \quad \boldsymbol{\eta}_t = \Sigma_t^{-1}\boldsymbol{\mu}_t \tag{2}$$

A. Duality between Standard and Canonical Forms

The canonical parametrization for the multivariate Gaussian is the dual of the standard form in regards to the marginalization and conditioning operations [10], as demonstrated in Table I. Marginalizing over variables with the standard form is *easy* since we simply remove the corresponding elements from the mean vector and covariance matrix. However, the same operation for the canonical form involves calculating a Schur complement and is computationally *hard*. The opposite is true when calculating the conditional from the joint distribution; it is *hard* with the standard form yet *easy* with the canonical parametrization.

TABLE I

Summary of Marginalization and Conditioning Operations on a Gaussian Distribution Expressed in Covariance and Information Form

$p\left(\boldsymbol{\alpha}, \boldsymbol{\beta}\right) = \mathcal{N}\big(\left[\begin{smallmatrix}\boldsymbol{\mu}_\alpha \\ \boldsymbol{\mu}_\beta\end{smallmatrix}\right], \left[\begin{smallmatrix}\Sigma_{\alpha\alpha} & \Sigma_{\alpha\beta} \\ \Sigma_{\beta\alpha} & \Sigma_{\beta\beta}\end{smallmatrix}\right]\big) = \mathcal{N}^{-1}\big(\left[\begin{smallmatrix}\boldsymbol{\eta}_\alpha \\ \boldsymbol{\eta}_\beta\end{smallmatrix}\right], \left[\begin{smallmatrix}\Lambda_{\alpha\alpha} & \Lambda_{\alpha\beta} \\ \Lambda_{\beta\alpha} & \Lambda_{\beta\beta}\end{smallmatrix}\right]\big)$

	Marginalization $p\left(\boldsymbol{\alpha}\right) = \int p\left(\boldsymbol{\alpha}, \boldsymbol{\beta}\right) d\boldsymbol{\beta}$	Conditioning $p\left(\boldsymbol{\alpha} \mid \boldsymbol{\beta}\right) = p\left(\boldsymbol{\alpha}, \boldsymbol{\beta}\right) / p\left(\boldsymbol{\beta}\right)$
Covariance Form	$\boldsymbol{\mu} = \boldsymbol{\mu}_\alpha$ $\Sigma = \Sigma_{\alpha\alpha}$	$\boldsymbol{\mu}' = \boldsymbol{\mu}_\alpha + \Sigma_{\alpha\beta}\Sigma_{\beta\beta}^{-1}(\boldsymbol{\beta} - \boldsymbol{\mu}_\beta)$ $\Sigma' = \Sigma_{\alpha\alpha} - \Sigma_{\alpha\beta}\Sigma_{\beta\beta}^{-1}\Sigma_{\beta\alpha}$
Information Form	$\boldsymbol{\eta} = \boldsymbol{\eta}_\alpha - \Lambda_{\alpha\beta}\Lambda_{\beta\beta}^{-1}\boldsymbol{\eta}_\beta$ $\Lambda = \Lambda_{\alpha\alpha} - \Lambda_{\alpha\beta}\Lambda_{\beta\beta}^{-1}\Lambda_{\beta\alpha}$	$\boldsymbol{\eta}' = \boldsymbol{\eta}_\alpha - \Lambda_{\alpha\beta}\boldsymbol{\beta}$ $\Lambda' = \Lambda_{\alpha\alpha}$

B. Implied Conditional Independence

An advantageous property of the canonical parametrization is that the information matrix provides an explicit representation for the structure of the corresponding Gaussian Markov random field (GMRF) [10],

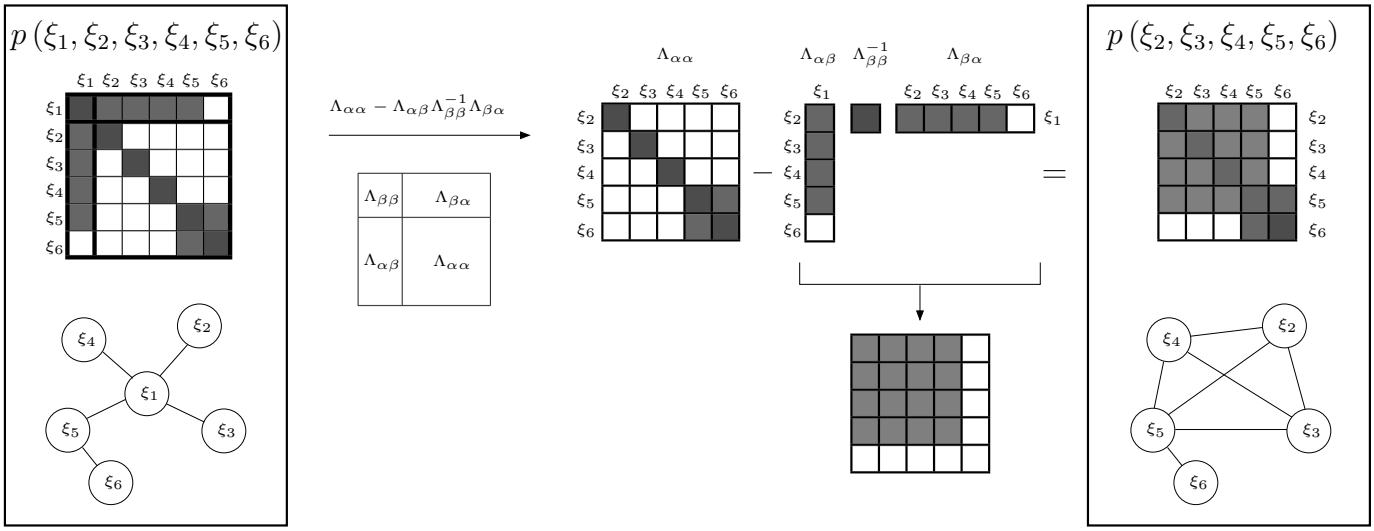


Fig. 1. An example of the effect of marginalization on the Gaussian information matrix. We start out with a joint posterior over $\xi_{1:6}$ represented by the information matrix and corresponding Markov network pictorialized on the left. The information matrix for the marginalized density, $p(\xi_{2:6}) = \int p(\xi_{1:6})\, d\xi_1$, corresponds to the Schur complement of $\Lambda_{\beta\beta} = \Lambda_{\xi_1\xi_1}$ in $\Lambda_{\xi_{1:6}\xi_{1:6}}$. This calculation essentially passes information constraints from the variable being removed, ξ_1, onto it's adjacent nodes, adding shared information between these variables. We see, then, that a consequence of marginalization is the population of the information matrix.

[15]. This property follows from the factorization of a general Gaussian density

$$\begin{aligned} p(\boldsymbol{\xi}) &\propto \exp\left\{-\tfrac{1}{2}\boldsymbol{\xi}^\top \Lambda \boldsymbol{\xi} + \boldsymbol{\eta}^\top \boldsymbol{\xi}\right\} \\ &= \prod_i \exp\left\{-\tfrac{1}{2}\left(\lambda_{ii}\xi_i^2 - \eta_i \xi_i\right)\right\} \cdot \prod_{\substack{i,j \\ i \neq j}} \exp\left\{-\tfrac{1}{2}\xi_i \lambda_{ij} \xi_j\right\} \\ &= \prod_i \Psi_i(\xi_i) \cdot \prod_{\substack{i,j \\ i \neq j}} \Psi_{ij}(\xi_i, \xi_j) \end{aligned}$$

where

$$\begin{aligned} \Psi_i(\xi_i) &= \exp\left\{-\tfrac{1}{2}\left(\lambda_{ii}\xi_i^2 - \eta_i \xi_i\right)\right\} \\ \Psi_{ij}(\xi_i, \xi_j) &= \exp\left\{-\tfrac{1}{2}\xi_i \lambda_{ij} \xi_j\right\} \end{aligned}$$

are the node and edge potentials, respectively, for the corresponding undirected graph. Random variable pairs with zero off-diagonal elements in the information matrix (i.e. $\lambda_{ij} = 0$) have an edge potential $\Psi_{ij}(\xi_i, \xi_j) = 1$, signifying the absence of a link between the nodes representing the variables. Conversely, non-zero shared information indicates that there is a link joining the corresponding nodes with the strength of the edge proportional to λ_{ij}. In turn, as the link topology for an undirected graph explicitly captures the conditional dependencies among variables, so does the structure of the information matrix. The presence of off-diagonal elements that are equal to zero then implies that the corresponding variables are conditionally independent, given the remaining states.

It is interesting to note that one comes to the same conclusion from a simple analysis of the conditioning operation for the information form. Per Table I, conditioning a pair of random variables, $\boldsymbol{\alpha} = [\boldsymbol{\xi}_i^\top \ \boldsymbol{\xi}_j^\top]^\top$, on the remaining states, $\boldsymbol{\beta}$, involves extracting the $\Lambda_{\alpha\alpha}$ sub-block from the information matrix. When there is no shared information between $\boldsymbol{\xi}_i$ and $\boldsymbol{\xi}_j$, $\Lambda_{\alpha\alpha}$ is block-diagonal as is its inverse (i.e. the covariance matrix). Conditioned upon $\boldsymbol{\beta}$, the two variables are uncorrelated and we can conclude that they are conditionally independent: $p\left(\boldsymbol{\xi}_i, \boldsymbol{\xi}_j \mid \boldsymbol{\beta}\right) = p\left(\boldsymbol{\xi}_i \mid \boldsymbol{\beta}\right) p\left(\boldsymbol{\xi}_j \mid \boldsymbol{\beta}\right)$.

The fact that the information matrix characterizes the conditional independence relationships emphasizes the significance of its structure. In particular, it is important to make a distinction between elements that

are truly zero and those that are just small in comparison to others. On that note, we return to the process of marginalization, which modifies zeros in the information matrix, thereby destroying some conditional indpendencies [10]. Consider a six state Gaussian random vector, $\boldsymbol{\xi} \sim \mathcal{N}^{-1}(\boldsymbol{\eta}, \Lambda)$, characterized by the information matrix and GMRF depicted in the left-hand side of Figure 1. The canonical form of the marginal density $p(\xi_{2:6}) = \int p(\xi_{1:6})\, d\xi_1 = \mathcal{N}^{-1}(\boldsymbol{\eta}', \Lambda')$ follows from Table I with $\boldsymbol{\alpha} = [\xi_2\ \xi_3\ \xi_4\ \xi_5\ \xi_6]^\top$ and $\beta = \xi_1$. The correction term in the Schur complement, $\Lambda_{\alpha\beta}\Lambda_{\beta\beta}^{-1}\Lambda_{\beta\alpha}$, is non-zero only at locations associated with variables directly linked with ξ_1. This set, denoted as $\mathbf{m}^+ = \{\xi_2, \xi_3, \xi_4, \xi_5\}$, comprises the Markov blanket for ξ_1. Subtracting the correction matrix modifies a number of entries in the $\Lambda_{\alpha\alpha}$ information sub-matrix, including some that were originally zero. Specifically, while no links exist between $\xi_{2:5}$ in the original distribution, the variables in $\mathbf{m}^+$ become fully connected as a result of marginalizing ξ_1. Marginalization results in the population of the information matrix, a characteristic that has important consequences when it comes to applying the information form to feature-based SLAM.

III. Feature-based SLAM Information Filters

We employ a feature-based representation of the environment, storing the map as a collection of geometric primitives, e.g. points and lines. The robot pose (position and orientation), $\mathbf{x}_t$, together with the set of n map features, $\mathbf{M} = \{\mathbf{m}_1, \mathbf{m}_2, \ldots, \mathbf{m}_n\}$, comprise the state vector, $\boldsymbol{\xi}_t = \begin{bmatrix}\mathbf{x}_t^\top & \mathbf{M}^\top\end{bmatrix}^\top$. In similar fashion to [2], we take a first-order linearization of the motion and measurement models and treat the uncertainty in the data as independent, white Gaussian noise. One can then show that the posterior obeys a Gaussian distribution:

$$p(\boldsymbol{\xi}_t \mid \mathbf{z}^t, \mathbf{u}^t) = \mathcal{N}(\boldsymbol{\mu}_t, \Sigma_t) = \mathcal{N}^{-1}(\boldsymbol{\eta}_t, \Lambda_t), \tag{3}$$

where $\mathbf{z}^t$ and $\mathbf{u}^t$ denote the history of observational data and motion control inputs, respectively. Throughout the paper, we will refer to (3) as the SLAM posterior.

Applying the notation introduced by Thrun *et al.* [13], the map is partitioned into two sets, $\mathbf{M} = \{\mathbf{m}^+, \mathbf{m}^-\}$, based upon the structure of the information matrix. The set of *active* features, $\mathbf{m}^+$, consists of the map elements with non-zero off-diagonal terms that link them with the robot, while $\mathbf{m}^-$ signifies the *passive* features that are conditionally independent of the vehicle pose. In the example displayed in the left-hand side of Figure 2, the active features are $\mathbf{m}^+ = \{\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3, \mathbf{m}_5\}$ and the single passive landmark is $\mathbf{m}^- = \mathbf{m}_4$.

An Extended Information Filter (EIF) tracks the SLAM distribution through time projection and measurement update steps in much the same way as the EKF. The remainder of this section is devoted to a detailed description of the canonical formulation to these processes.

A. Measurement Update Step

Observations of landmarks are key to reducing the uncertainty in the estimates for the robot pose and the map. The measurement model (4a) is a nonlinear function of the state corrupted by white Gaussian noise, $\mathbf{v}_t \sim \mathcal{N}(\mathbf{0}, \mathrm{R})$. Equation (4b) is the first-order linearization about the mean of the robot pose and observed features with the Jacobian, H, evaluated at this mean.

$$\mathbf{z}_t = \mathbf{h}(\boldsymbol{\xi}_t) + \mathbf{v}_t \tag{4a}$$

$$\approx \mathbf{h}(\bar{\boldsymbol{\mu}}_t) + \mathrm{H}(\boldsymbol{\xi}_t - \bar{\boldsymbol{\mu}}_t) + \mathbf{v}_t \tag{4b}$$

The process of updating the current distribution, $p(\boldsymbol{\xi}_t \mid \mathbf{z}^{t-1}, \mathbf{u}^t) = \mathcal{N}^{-1}(\bar{\boldsymbol{\eta}}_t, \bar{\Lambda}_t)$, to reflect a new observation follows from Bayes' rule,

$$p(\boldsymbol{\xi}_t \mid \mathbf{z}^t, \mathbf{u}^t) \propto p(\mathbf{z}_t \mid \boldsymbol{\xi}_t)\ p(\boldsymbol{\xi}_t \mid \mathbf{z}^{t-1}, \mathbf{u}^t), \tag{5}$$

where we exploit the conditional independence of the measurements given the state. The EIF estimates the canonical form of the new posterior via the update step:

$$p(\boldsymbol{\xi}_t \mid \mathbf{z}^t, \mathbf{u}^t) = \mathcal{N}^{-1}(\boldsymbol{\eta}_t, \Lambda_t)$$

$$\Lambda_t = \bar{\Lambda}_t + \mathrm{H}^\top \mathrm{R}^{-1} \mathrm{H} \tag{6a}$$

$$\boldsymbol{\eta}_t = \bar{\boldsymbol{\eta}}_t + \mathrm{H}^\top \mathrm{R}^{-1} \big(\mathbf{z}_t - \mathbf{h}\left(\bar{\boldsymbol{\mu}}_t\right) + \mathrm{H} \bar{\boldsymbol{\mu}}_t \big) \tag{6b}$$

For a detailed derivation, the reader is referred to [13].

At any timestep, the robot typically makes a limited number, m, of relative observations to individual landmarks. The measurement model is then a function only of the vehicle pose and this small subset of map elements, $\mathbf{m}_i$ and $\mathbf{m}_j$ and, in turn, a majority of terms in the Jacobian (7) are zero.

$$\mathrm{H} = \begin{bmatrix} \frac{\partial h_1}{\partial \mathbf{x}_t} & \cdots & \mathbf{0} & \cdots & \frac{\partial h_1}{\partial \mathbf{m}_i} & \cdots & \mathbf{0} \\ \vdots & & & \ddots & & & \vdots \\ \frac{\partial h_m}{\partial \mathbf{x}_t} & \cdots & \frac{\partial h_m}{\partial \mathbf{m}_j} & \cdots & \mathbf{0} & \cdots & \mathbf{0} \end{bmatrix} \tag{7}$$

The matrix outer-product in (6a), $\mathrm{H}^\top \mathrm{R}^{-1} \mathrm{H}$ is zero everywhere except at positions associated with the vehicle pose and observed features. More specifically, the matrix is populated at the $\mathbf{x}_t$, $\mathbf{m}_i$, and $\mathbf{m}_j$ positions along the diagonal as well as at the off-diagonal positions for the $(\mathbf{x}_t, \mathbf{m}_i)$ and $(\mathbf{x}_t, \mathbf{m}_j)$ pairs. The addition of this matrix to the original information matrix only modifies the terms exclusively related to the robot and the observed landmarks. The update then acts to either strengthen existing constraints between the vehicle and these features or to establish new ones (i.e., make them active).

Due to the sparseness of H, computing $\mathrm{H}^\top \mathrm{R}^{-1} \mathrm{H}$ involves $\mathcal{O}(m^2)$ multiplications. Assuming knowledge of the mean for the robot pose and observed features for the linearization, this matrix product is the most expensive component of (6). Since the number of observations, m, is limited by the robot's field of view, the EIF update time is bounded and does not grow with the size of the map. In general, though, we do not have an estimate for the current mean and computing it via (2) requires an $\mathcal{O}(n^3)$ matrix inversion. The exception is when the measurement model is linear, in which case the mean is not necessary and the update step is indeed constant-time.

B. Time Projection Step

The time projection step predicts the distribution over the new robot pose through what can be thought of as a two step process. First, we augment the canonical form with a new random variable that represents the new vehicle pose. We then marginalize over the old pose, leaving us with the up-to-date state distribution.

1) State Augmentation: A Markov model governs the motion of the robot and is, in general, a nonlinear function (8a) of the previous pose and the control input. The additive term, $\mathbf{w}_t \sim \mathcal{N}\big(\mathbf{0}, \mathrm{Q}\big)$, represents a Gaussian approximation to the uncertainty in the model. The first-order linearization about the mean robot pose, $\boldsymbol{\mu}_{x_t}$, follows in (8b) where F the Jacobian matrix.

$$\mathbf{x}_{t+1} = \mathbf{f}\big(\mathbf{x}_t, \mathbf{u}_{t+1}\big) + \mathbf{w}_t \tag{8a}$$

$$\approx \mathbf{f}\big(\boldsymbol{\mu}_{x_t}, \mathbf{u}_{t+1}\big) + \mathrm{F}\big(\mathbf{x}_t - \boldsymbol{\mu}_{x_t}\big) + \mathbf{w}_t \tag{8b}$$

First, we grow the state vector to include the new robot pose, $\hat{\boldsymbol{\xi}}_{t+1} = [\mathbf{x}_t^\top \ \mathbf{x}_{t+1}^\top \ \mathbf{M}^\top]^\top$. The distribution over $\hat{\boldsymbol{\xi}}_{t+1}$ follows from the current posterior, $p\left(\boldsymbol{\xi}_t \mid \mathbf{z}^t, \mathbf{u}^t\right) = \mathcal{N}^{-1}\big(\boldsymbol{\eta}_t, \Lambda_t\big)$, through the factorization

$$p\left(\hat{\boldsymbol{\xi}}_{t+1} \mid \mathbf{z}^t, \mathbf{u}^{t+1}\right) = p\left(\mathbf{x}_{t+1}, \boldsymbol{\xi}_t \mid \mathbf{z}^t, \mathbf{u}^{t+1}\right) = p\left(\mathbf{x}_{t+1} \mid \mathbf{x}_t, \mathbf{u}_{t+1}\right) \, p\left(\boldsymbol{\xi}_t \mid \mathbf{z}^t, \mathbf{u}^t\right)$$

where we have exploited the Markov property. Accordingly, the augmentation to the information matrix and vector is shown by Eustice *et al.* [21] to have the form given in (9). Notice that the new robot pose shares information with the previous pose but not the map. This is exemplified in the middle schematic within Figure 2 by the fact that the only effect on the structure of the graphical model is the addition of the $\mathbf{x}_{t+1}$ node linked to that of $\mathbf{x}_t$. Given $\mathbf{x}_t$, the $\mathbf{x}_{t+1}$ pose is conditionally independent of the map as a consequence of the Markov property.

$$p\left(\mathbf{x}_t, \mathbf{x}_{t+1}, \mathbf{M} \mid \mathbf{z}^t, \mathbf{u}^{t+1}\right) = \mathcal{N}^{-1}\big(\hat{\boldsymbol{\eta}}_{t+1}, \hat{\Lambda}_{t+1}\big)$$

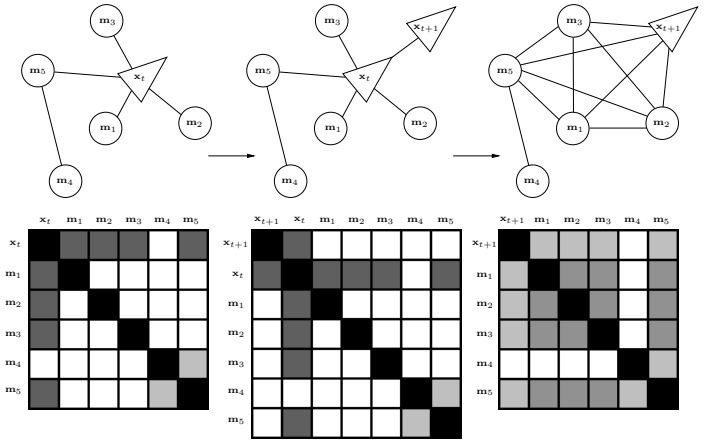


Fig. 2. A graphical explanation for the inherent density of the information matrix due to the motion update step. Darker shades in the matrix imply larger magnitude. On the left are the Markov network and sparse information matrix prior to time projection in which the robot shares information with the active features, $\mathbf{m}^+ = \{\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3, \mathbf{m}_5\}$. We augment the state with the new robot pose, which is linked only to the previous pose due to the Markovian motion model. Subsequently, we marginalize over $\mathbf{x}_t$, resulting in the representation shown on the right. The removal of $\mathbf{x}_t$ creates constraints between the robot and each map element in $\mathbf{m}^+$, which are now fully connected. Along with filling in the information matrix, we see from the shading that the time projection step weakens many constraints, explaining the approximate sparsity of the normalized information matrix.

$$\hat{\Lambda}_{t+1} = \left[\begin{array}{c|cc} \left(\Lambda_{x_t x_t} + \mathrm{F}^\top \mathrm{Q}^{-1} \mathrm{F}\right) & -\mathrm{F}^\top \mathrm{Q}^{-1} & \Lambda_{x_t M} \\ \hline -\mathrm{Q}^{-1}\mathrm{F} & \mathrm{Q}^{-1} & 0 \\ \Lambda_{M x_t} & 0 & \Lambda_{MM} \end{array}\right] = \left[\begin{array}{c|c} \hat{\Lambda}^{11}_{t+1} & \hat{\Lambda}^{12}_{t+1} \\ \hline \hat{\Lambda}^{21}_{t+1} & \hat{\Lambda}^{22}_{t+1} \end{array}\right] \tag{9a}$$

$$\hat{\boldsymbol{\eta}}_{t+1} = \left[\begin{array}{c} \boldsymbol{\eta}_{x_t} - \mathrm{F}^\top \mathrm{Q}^{-1}\left(\mathbf{f}\left(\boldsymbol{\mu}_{x_t}, \mathbf{u}_{t+1}\right) - \mathrm{F}\boldsymbol{\mu}_{x_t}\right) \\ \hline \mathrm{Q}^{-1}\left(\mathbf{f}\left(\boldsymbol{\mu}_{x_t}, \mathbf{u}_{t+1}\right) - \mathrm{F}\boldsymbol{\mu}_{x_t}\right) \\ \boldsymbol{\eta}_M \end{array}\right] = \left[\begin{array}{c} \hat{\boldsymbol{\eta}}^1_{t+1} \\ \hline \hat{\boldsymbol{\eta}}^2_{t+1} \end{array}\right] \tag{9b}$$

2) Marginalization: We complete the time projection step by marginalizing $\mathbf{x}_t$ from the posterior to achieve the desired distribution over $\boldsymbol{\xi}_{t+1} = [\mathbf{x}^\top_{t+1}\ \mathbf{M}^\top]^\top$.

$$p\left(\mathbf{x}_{t+1}, \mathbf{M} \mid \mathbf{z}^t, \mathbf{u}^{t+1}\right) = \int_{\mathbf{x}_t} p\left(\mathbf{x}_t, \mathbf{x}_{t+1}, \mathbf{M} \mid \mathbf{z}^t, \mathbf{u}^{t+1}\right) d\mathbf{x}_t$$

This brings us back to the expression for marginalization in the canonical form from Table I that we apply here:

$$p\left(\boldsymbol{\xi}_{t+1} \mid \mathbf{z}^t, \mathbf{u}^{t+1}\right) = \mathcal{N}^{-1}\left(\bar{\boldsymbol{\eta}}_{t+1}, \bar{\Lambda}_{t+1}\right)$$

$$\bar{\Lambda}_{t+1} = \hat{\Lambda}^{22}_{t+1} - \hat{\Lambda}^{21}_{t+1}\left(\hat{\Lambda}^{11}_{t+1}\right)^{-1}\hat{\Lambda}^{12}_{t+1} \tag{10a}$$

$$\bar{\boldsymbol{\eta}}_{t+1} = \hat{\boldsymbol{\eta}}^2_{t+1} - \hat{\Lambda}^{21}_{t+1}\left(\hat{\Lambda}^{11}_{t+1}\right)^{-1}\hat{\boldsymbol{\eta}}^1_{t+1} \tag{10b}$$

To better understand the consequences of this marginalization, we refer back to the discussion at the end of §II-B. Prior to marginalization, the old robot pose is linked to the active features, $\mathbf{m}^+$, while the new pose shares information only with $\mathbf{x}_t$. When we remove the old pose, though, a link is formed between the new pose and each feature in $\mathbf{m}^+$ and this set itself becomes fully connected. The information matrix that was originally sparse is now populated as a consequence of (10a). In the scenario depicted in the right-hand side of Figure 2, the only remaining zero entries correspond to the lone feature, $\mathbf{m}_4$, which will become connected to the robot upon the next observation. As Paskin [10] previously showed, the time projection step, then, naturally leads to a dense information matrix in online, feature-based SLAM.

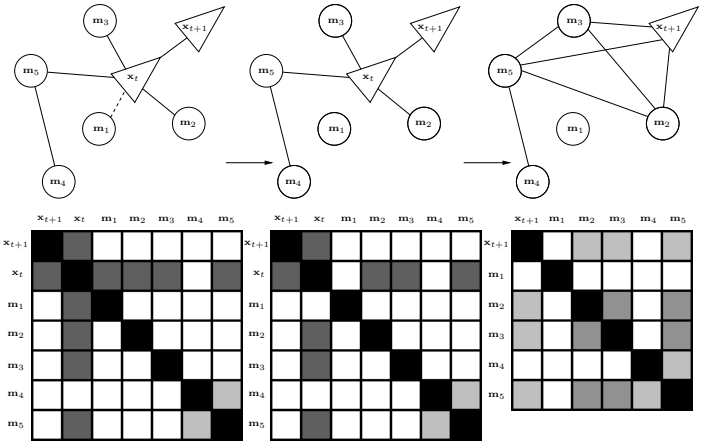


Fig. 3. A sequence of illustrations highlighting the benefits of breaking links between the robot and the map. If feature $\mathbf{m}_1$ is first made passive by removing the constraint to the old pose, $\mathbf{x}_t$, then marginalizing over $\mathbf{x}_t$ will not link it to the other active features. This implies that we can control the density of the information matrix by maintaining a bound on the number of active features.

The density of the matrix affects the computational cost of time prediction, specifically the marginalization component. The correction term of the Schur complement (10a) is calculated as the outer product with the off-diagonal submatrix for the old pose, $\hat{\Lambda}^{21}_{t+1}$. Computing the outer product is quadratic in the number of nonzero elements within this submatrix and, equivalently, the number of map elements linked to the old pose (i.e. the size of $\mathbf{m}^+$). As we have just mentioned, though, this number will only increase over time with the size of the map. Thus, while it may be possible to efficiently incorporate new measurements with an EIF, the price we pay is a quadratic complexity for the motion update step.

In its natural form, the EIF is no better than $\mathcal{O}(n^2)$ and doesn't offer an alternative to the EKF in addressing scalability issues. A closer look at the structure of the information matrix, though, reveals that an adapted form of the EIF may provide a solution. Returning to the example pictorialized in Figure 2, note that, aside from populating the information matrix, the time projection step weakens the off-diagonal links. This has been shown to result in a normalized information matrix that is nearly sparse [24]. The analysis of [10], [13], [14] reveals that, by approximating the matrix as being exactly sparse, it is possible to achieve significant gains when it comes to both storage and time requirements. Specifically, a bound on the number of links between the robot and the map allows for near constant-time implementation of the time projection step and controls the fill-in of the information matrix resulting from marginalization. The delicate issue is *how* to approximate the posterior, $p\left(\boldsymbol{\xi}_t \mid \mathbf{z}^t, \mathbf{u}^t\right)$, with a sparse canonical form.

IV. Sparsification via Enforced Conditional Independence

The fill-in of the information matrix induced by the motion update step, together with its computational complexity, are proportional to the number of links between the robot and the map. Unfortunately, while these links may weaken, they never disappear and lead to an ever-growing size of the active map [10]. In order to achieve a scalable SLAM information filter, we need to break links with the robot, thereby bounding the number of active landmarks. Adopting the notation used in [13], we let Γ_a signify the bound on the number of active map elements and use Γ_p to denote the number of inter-landmark links in the matrix.

As motivation, consider again the scenario depicted in Figure 2 in which four features are active. Marginalizing out $\mathbf{x}_t$ creates links among each element in $\mathbf{m}^+$, potentially violating the Γ_p bound. Instead, Figure 3 demonstrates that if $\mathbf{m}_1$ was first made passive, then the number of non-zero elements created as a result of marginalization can be controlled.

A. SEIF Sparsification

Recalling the conditional dependency implicit in the GMRF, the SEIF breaks links between the robot and a set of landmarks by replacing the SLAM posterior with a distribution that approximates the conditional independence between the pose and these features. Decompose the map into three disjoint sets, $\mathbf{M} = \{\mathbf{m}^0, \mathbf{m}^+, \mathbf{m}^-\}$, where $\mathbf{m}^-$ refers to the passive features that will *remain* passive, $\mathbf{m}^0$ are the active landmarks that will be *made* passive, and $\mathbf{m}^+$ denotes the active features that are to *remain* active. The sparsification routine proceeds from a decomposition of the SLAM posterior

$$\begin{aligned} p\left(\boldsymbol{\xi}_t \mid \mathbf{z}^t, \mathbf{u}^t\right) &= p\left(\mathbf{x}_t, \mathbf{m}^0, \mathbf{m}^+, \mathbf{m}^-\right) \\ &= p\left(\mathbf{x}_t \mid \mathbf{m}^0, \mathbf{m}^+, \mathbf{m}^-\right) p\left(\mathbf{m}^0, \mathbf{m}^+, \mathbf{m}^-\right) \qquad (11a) \\ &= p\left(\mathbf{x}_t \mid \mathbf{m}^0, \mathbf{m}^+, \mathbf{m}^- = \boldsymbol{\varphi}\right) p\left(\mathbf{m}^0, \mathbf{m}^+, \mathbf{m}^-\right) \qquad (11b) \end{aligned}$$

where we have omitted the dependence upon $\mathbf{z}^t$ and $\mathbf{u}^t$ for notational convenience. In (11b), we are free to condition on an arbitrary instantiation of the passive features, $\mathbf{m}^- = \boldsymbol{\varphi}$, due to the conditional independence between the robot and these landmarks.

The SEIF deactivates landmarks by replacing (11) with an approximation to the posterior that drops the dependence of the robot pose on $\mathbf{m}^0$:

$$\begin{aligned} \tilde{p}_{\text{SEIF}}\left(\boldsymbol{\xi}_t \mid \mathbf{z}^t, \mathbf{u}^t\right) &= \tilde{p}_{\text{SEIF}}\left(\mathbf{x}_t, \mathbf{m}^0, \mathbf{m}^+, \mathbf{m}^-\right) \\ &= p\left(\mathbf{x}_t \mid \mathbf{m}^+, \mathbf{m}^- = \boldsymbol{\varphi}\right) p\left(\mathbf{m}^0, \mathbf{m}^+, \mathbf{m}^-\right) \qquad (12) \end{aligned}$$

While the expression in (11b) is theoretically exact, it is no longer valid to condition upon a particular value for the passive map elements while ignoring the dependence upon $\mathbf{m}^0$ as we have done in (12). Given only a subset of the active map, the robot pose and passive features are *dependent*, suggesting that the particular choice for $\boldsymbol{\varphi}$ affects the approximation. In fact, the authors show in [25] that setting the passive features to any value other than their mean (i.e. $\boldsymbol{\varphi} \neq \boldsymbol{\mu}_{m^-}$) yields a mean of $\tilde{p}_{\text{SEIF}}\left(\boldsymbol{\xi}_t \mid \mathbf{z}^t, \mathbf{u}^t\right)$ that differs from that of the original posterior[2], $p\left(\boldsymbol{\xi}_t \mid \mathbf{z}^t, \mathbf{u}^t\right)$. Furthermore, we will demonstrate that by ignoring the dependence relationships in (12), the SEIF sparsification strategy leads to inconsistent covariance estimates.

B. Discussion on Overconfidence

An important consequence of the SEIF sparsification algorithm is that the resulting approximation to the SLAM posterior significantly underestimates the uncertainty in the state estimate. In this section, we show that this inconsistency is a natural consequence of imposing conditional independence between the robot pose and the $\mathbf{m}^0$ subset of the map. To illustrate this effect, consider a general three state Gaussian distribution,

$$p\big(a, b, c\big) = \mathcal{N}\left(\begin{bmatrix}\mu_a \\ \mu_b \\ \mu_c\end{bmatrix}, \begin{bmatrix}\sigma_a^2 & \rho_{ab}\sigma_a\sigma_b & \rho_{ac}\sigma_a\sigma_c \\ \rho_{ab}\sigma_a\sigma_b & \sigma_b^2 & \rho_{bc}\sigma_b\sigma_c \\ \rho_{ac}\sigma_a\sigma_c & \rho_{bc}\sigma_b\sigma_c & \sigma_c^2\end{bmatrix}\right) = \mathcal{N}^{-1}\left(\begin{bmatrix}\eta_a \\ \eta_b \\ \eta_c\end{bmatrix}, \begin{bmatrix}\lambda_{aa} & \lambda_{ab} & \lambda_{ac} \\ \lambda_{ab} & \lambda_{bb} & \lambda_{bc} \\ \lambda_{ac} & \lambda_{bc} & \lambda_{cc}\end{bmatrix}\right), \qquad (13)$$

that we would like to sparsify by forcing a and b to be conditionally independent given c:

$$p\left(a, b, c\right) = p\left(a, b \mid c\right) p\left(c\right) \xrightarrow{\text{approx.}} \tilde{p}\left(a, b, c\right) = p\left(a \mid c\right) p\left(b \mid c\right) p\left(c\right). \qquad (14)$$

Recalling the discussion in §II-B, the approximation is implemented in the canonical form by setting $\lambda_{ab} = 0$. In the standard form, this is equivalent to treating a and b as being uncorrelated in $p\left(a, b \mid c\right)$. The resulting approximation then follows as

$$\tilde{p}\big(a, b, c\big) = \mathcal{N}\left(\begin{bmatrix}\mu_a \\ \mu_b \\ \mu_c\end{bmatrix}, \begin{bmatrix}\sigma_a^2 & \rho_{ac}\rho_{bc}\sigma_a\sigma_b & \rho_{ac}\sigma_a\sigma_c \\ \rho_{ac}\rho_{bc}\sigma_a\sigma_b & \sigma_b^2 & \rho_{bc}\sigma_b\sigma_c \\ \rho_{ac}\sigma_a\sigma_c & \rho_{bc}\sigma_b\sigma_c & \sigma_c^2\end{bmatrix}\right). \qquad (15)$$

[2]The mean is preserved by the sparsification routine in [13] since they condition upon $\boldsymbol{\varphi} = \boldsymbol{\mu}_{m^-}$ and not $\boldsymbol{\varphi} = \mathbf{0}$ as is stated in the paper.

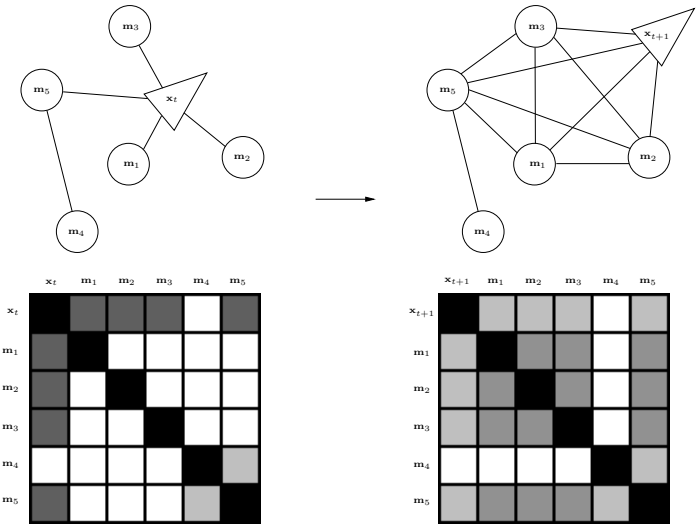


Fig. 4. At time t, the robot observes four features, $\{\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3, \mathbf{m}_5\}$, three of which are already active, while $\mathbf{m}_3$ is passive. The update strengthens the shared information between vehicle pose and $\mathbf{m}_1$, $\mathbf{m}_2$, and $\mathbf{m}_5$ and adds a link to $\mathbf{m}_3$ as we indicate on the left. The next time projection step forms a clique among the robot and these four features and populates the information matrix. The ESEIF sparsification strategy avoids this effect by controlling the number of active landmarks and, in turn, the size of this clique.

In order for the approximation to be consistent, it is necessary and sufficient that the resulting covariance matrix obey the inequality,

$$\tilde{\Sigma} - \Sigma = \begin{bmatrix} 0 & (\rho_{ac}\rho_{bc} - \rho_{ab})\,\sigma_a\sigma_b & 0 \\ (\rho_{ac}\rho_{bc} - \rho_{ab})\,\sigma_a\sigma_b & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \geq 0. \tag{16}$$

A necessary condition for (16) to hold is that the determinant of the upper-left 2×2 sub-block be non-negative [26]. Clearly, this is not the case for every $\rho_{ac}\rho_{bc} \neq \rho_{ab}$. Extending this insight to the SEIF sparsification strategy sheds some light on why enforcing the conditional independence between the robot pose and the $\mathbf{m}^0$ landmarks leads to overconfident state estimates.

V. Exactly Sparse Extended Information Filters

Recall that, as landmarks are added to the map, EIF SLAM algorithms create shared information with the robot pose. Over time, these off-diagonal elements in the information matrix may decay, but never to zero. Together with the time projection step, this leads to a populated matrix as discussed in [10] and noted earlier. The SEIF deals with this by substituting the SLAM posterior with an approximation in which the vehicle is conditionally independent of much of the map. The algorithm takes advantage of what is then a sparse information matrix to achieve near constant-time efficiency. The drawback, though, is that the SEIF yields overconfident state estimates as a result of approximating conditional independencies.

We propose the Exactly Sparse Extended Information Filter (ESEIF) as an alternative sparse information matrix SLAM algorithm; one which imposes exact sparsity in contrast to the approximate conditional independencies enforced in SEIF. As we show, the result is a computationally efficient SLAM algorithm that tracks a posterior distribution, $p_{\text{ESEIF}}\big(\boldsymbol{\xi}_t \mid \mathbf{z}^t, \mathbf{u}^t\big)$, that is consistent with respect to that of the full EKF/EIF.

A. ESEIF Sparsification

The general premise of ESEIF sparsification is straightforward: rather than deliberately breaking links between the robot and map, we maintain sparsity by controlling their initial formation. More specifically, the ESEIF manages the number of active landmarks by first marginalizing out the vehicle pose, essentially

"kidnapping" the robot. The algorithm subsequently relocalizes the vehicle within the map based upon new observations to a set of known landmarks. The set of features that were originally active have been made passive and the set of landmarks used for relocalization form the new active map.

```
Time Projection: (η⁺_{t-1}, Λ⁺_{t-1}) ⟶ (η⁻_t, Λ⁻_t) ;
Measurement Update (z_t = {z_active, z_passive}):
  if N_active + n(z_passive) ≤ Γ_a then
      Standard update: (η⁻_t, Λ⁻_t) --(6)--> (η⁺_t, Λ⁺_t) ;
      N_active = N_active + n(z_passive);
  else
      Partition z_t = {z_α, z_β} s.t. n(z_β) ≤ Γ_a ;
      (i) Update using z_α: (η⁻_t, Λ⁻_t) --(17)--> (η̄⁻_t, Λ̄⁻_t) ;
      (ii) Marginalize over the robot: (η̄⁻_t, Λ̄⁻_t) --(19)--> (η̆⁻_t, Λ̆⁻_t) ;
      (iii) Relocate the robot using z_β: (η̆⁻_t, Λ̆⁻_t) --(22)--> (η̆⁺_t, Λ̆⁺_t) ;
      N_active = n(z_β) ;
  end
```

Algorithm 1: A description of the ESEIF algorithm. Note that $\mathrm{N}_{\text{active}}$ denotes the number of features which are currently active.

The ESEIF sparsification algorithm takes the form of a variation on the measurement update step and is outlined in Algorithm 1. For a more detailed description, we consider a situation that would give rise to the GMRF depicted in Figure 4. At time t, suppose that the robot makes four observations of the environment, $\mathcal{Z}_t = \{\mathbf{z}_1, \mathbf{z}_2, \mathbf{z}_3, \mathbf{z}_5\}$, three of active features and one of a passive landmark:

$$\begin{aligned} &\text{Active:} \quad \mathbf{z}_1 = \mathbf{h}(\mathbf{x}_v, \mathbf{m}_1),\ \mathbf{z}_2 = \mathbf{h}(\mathbf{x}_v, \mathbf{m}_2),\ \mathbf{z}_5 = \mathbf{h}(\mathbf{x}_v, \mathbf{m}_5) \\ &\text{Passive:} \quad \mathbf{z}_3 = \mathbf{h}(\mathbf{x}_v, \mathbf{m}_3). \end{aligned}$$

Updating the current distribution, $p\left(\boldsymbol{\xi}_t \mid \mathbf{z}^{t-1}, \mathbf{u}^t\right)$, based upon all four measurements would strengthen the off-diagonal entries in the information matrix pairing the robot with the three observed active features, $\mathbf{m}_1$, $\mathbf{m}_2$, and $\mathbf{m}_5$. Additionally, the update would create a link to the passive landmark, $\mathbf{m}_3$, the end result being the information matrix and corresponding graphical model shown in the left-hand side of Figure 4. Suppose that activating $\mathbf{m}_3$ would violate the Γ_a bound. Aside from updating the filter and subsequently implementing the SEIF sparsification rule, one strategy would be to simply disregard the observation of the passive landmark entirely. This approach, though, is not acceptable since the size of the map that we can build is then dictated by the Γ_a bound. Alternatively, ESEIFs allow us to incorporate all measurement data while simultaneously maintaining the desired degree of sparsity.

In the ESEIF sparsification step, the measurement data is partitioned into two sets, $\mathbf{z}_\alpha$ and $\mathbf{z}_\beta$, where the first set of observations is used to update the filter and the second is reserved for performing relocalization. Several factors guide the specific allocation, including the number and quality of measurements necessary for relocalization. Of the four measurements available in our example, group that of the passive feature together with one of the active measurements for the update, $\mathbf{z}_\alpha = \{\mathbf{z}_1, \mathbf{z}_3\}$. The remaining two observations will be used for relocalization, $\mathbf{z}_\beta = \{\mathbf{z}_2, \mathbf{z}_5\}$. With that said, we now describe the two components of sparsification.

1) Posterior Update: We first perform a Bayesian update on the current distribution, $p\left(\boldsymbol{\xi}_t \mid \mathbf{z}^{t-1}, \mathbf{u}^t\right)$ to incorporate the information provided by the $\mathbf{z}_\alpha$ measurements:

$$p\left(\boldsymbol{\xi}_t \mid \mathbf{z}^{t-1}, \mathbf{u}^t\right) = \mathcal{N}^{-1}\left(\boldsymbol{\xi}_t; \boldsymbol{\eta}_t, \Lambda_t\right) \xrightarrow{\mathbf{z}_\alpha = \{\mathbf{z}_1, \mathbf{z}_3\}} p_1\left(\boldsymbol{\xi}_t \mid \left\{\mathbf{z}^{t-1}, \mathbf{z}_\alpha\right\}, \mathbf{u}^t\right) = \mathcal{N}^{-1}\left(\boldsymbol{\xi}_t; \bar{\boldsymbol{\eta}}_t, \bar{\Lambda}_t\right). \tag{17}$$

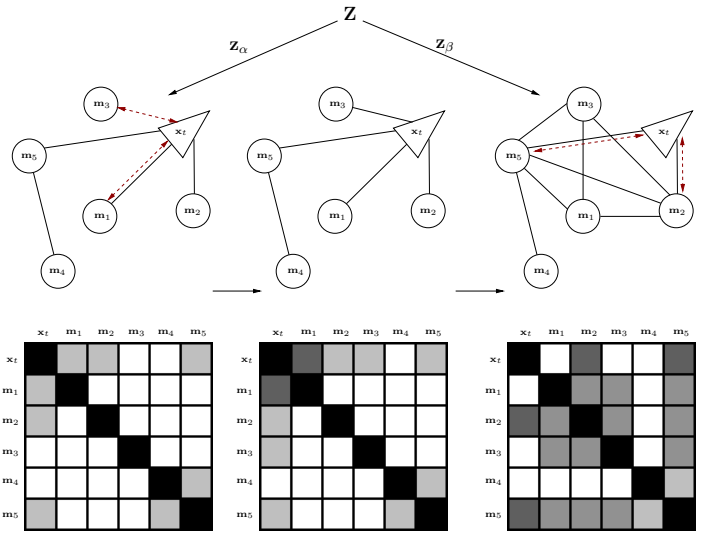


Fig. 5. A graphical description of the ESEIF sparsification strategy. At time t, the map is comprised of three active features, $\mathbf{m}^+ = \{\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_5\}$, and two passive features, $\mathbf{m}^- = \{\mathbf{m}_3, \mathbf{m}_4\}$, as indicated by the shaded off-diagonal elements in the information matrix. The robot makes three observations of active landmarks, $\{\mathbf{z}_1, \mathbf{z}_2, \mathbf{z}_5\}$, and one of a passive feature, $\mathbf{z}_3$. In the first step of the sparsification algorithm, shown in the left-most diagram, the ESEIF updates the distribution based upon a subset of the measurements, $\mathbf{z}_\alpha = \{\mathbf{z}_1, \mathbf{z}_3\}$. The result is a stronger constraint between $\mathbf{m}_1$ and the robot as well as the creation of a link with $\mathbf{m}_3$, which we depict in the middle figure. Subsequently, the ESEIF marginalizes out the vehicle pose, leading to connectivity among the active landmarks. The schematic on the right demonstrates the final step of sparsification in which the robot is relocated within the map based upon the remaining $\mathbf{z}_\beta = \{\mathbf{z}_2, \mathbf{z}_5\}$ measurements. The result is a joint posterior, $p_{\text{ESEIF}}\big(\boldsymbol{\xi}_t \mid \mathbf{z}^t, \mathbf{u}^t\big)$, represented by an exactly sparse information matrix where the size of the active map is controlled.

The $p_1\big(\boldsymbol{\xi}_t \mid \{\mathbf{z}^{t-1}, \mathbf{z}_\alpha\}, \mathbf{u}^t\big)$ posterior follows from the standard update equations (6) for the information filter. The Jacobian matrix, H, is nonzero only at indices affiliated with the robot pose and the $\mathbf{m}_1$ and $\mathbf{m}_3$ landmarks. As a result, the process strengthens the link between the robot and the active feature, $\mathbf{m}_1$, and creates shared information with $\mathbf{m}_3$, which was passive. The middle diagram of Figure 5 demonstrates this effect. With regards to the computational complexity, recall that the update step is constant-time with, in the nonlinear case, access to the mean estimate for the robot pose as well as $\mathbf{m}_1$ and $\mathbf{m}_3$.

2) Marginalization and Relocalization: Now that a new connection to the vehicle node has been added to the graph, there are too many active features. The ESEIF sparsification routine proceeds to marginalize out the robot pose to achieve the distribution over the map,

$$\begin{aligned} p_2\big(\mathbf{M}_t \mid \{\mathbf{z}^{t-1}, \mathbf{z}_\alpha\}, \mathbf{u}^t\big) &= \int_{\mathbf{x}_t} p_1\big(\boldsymbol{\xi}_t \mid \{\mathbf{z}^{t-1}, \mathbf{z}_\alpha\}, \mathbf{u}^t\big) d\mathbf{x}_t \\ &= \mathcal{N}^{-1}\big(\mathbf{M}_t; \check{\boldsymbol{\eta}}_t, \check{\Lambda}_t\big). \end{aligned} \tag{18}$$

In order to make the derivation a little clearer, decompose the canonical expression for $p_1\big(\boldsymbol{\xi}_t \mid \{\mathbf{z}^{t-1}, \mathbf{z}_\alpha\}, \mathbf{u}^t\big)$ into the robot pose and map components,

$$p_1\big(\boldsymbol{\xi}_t \mid \{\mathbf{z}^{t-1}, \mathbf{z}_\alpha\}, \mathbf{u}^t\big) = \mathcal{N}^{-1}\big(\boldsymbol{\xi}_t; \bar{\boldsymbol{\eta}}_t, \bar{\Lambda}_t\big)$$

$$\bar{\boldsymbol{\eta}}_t = \begin{bmatrix} \bar{\boldsymbol{\eta}}_{x_t} \\ \bar{\boldsymbol{\eta}}_M \end{bmatrix} \quad \bar{\Lambda}_t = \begin{bmatrix} \bar{\Lambda}_{x_t x_t} & \bar{\Lambda}_{x_t M} \\ \bar{\Lambda}_{M x_t} & \bar{\Lambda}_{MM} \end{bmatrix}.$$

The information matrix for the marginalized distribution then follows from Table I:

$$p_2\big(\mathbf{M}_t \mid \{\mathbf{z}^{t-1}, \mathbf{z}_\alpha\}, \mathbf{u}^t\big) = \mathcal{N}^{-1}\big(\mathbf{M}_t; \check{\boldsymbol{\eta}}_t, \check{\Lambda}_t\big)$$

$$\check{\Lambda}_t = \bar{\Lambda}_{MM} - \bar{\Lambda}_{M x_t} \left(\bar{\Lambda}_{x_t x_t}\right)^{-1} \bar{\Lambda}_{x_t M} \tag{19a}$$

$$\check{\boldsymbol{\eta}}_t = \bar{\boldsymbol{\eta}}_M - \bar{\Lambda}_{M x_t} \left(\bar{\Lambda}_{x_t x_t}\right)^{-1} \bar{\boldsymbol{\eta}}_{x_t}. \tag{19b}$$

This marginalization component of sparsification is computationally efficient. Inverting the robot pose sub-matrix, $\bar{\Lambda}_{x_t x_t} \in \mathbb{R}^{p\times p}$, is a constant-time operation since p is fixed. The ESEIF then multiplies the inverse by $\bar{\Lambda}_{M x_t} \in \mathbb{R}^{n\times p}$, the sub-block that captures the shared information between the robot and map. With a bound on the number of active landmarks, a limited number of k rows are populated and the matrix product is $\mathcal{O}(kp^2)$. In (19a), we then post-multiply by the transpose in $\mathcal{O}(k^2 p)$ time while, in (19b) we post-multiply by $\bar{\boldsymbol{\eta}}_{x_t} \in \mathbb{R}^{p\times 1}$, an $\mathcal{O}(kp)$ operation. With the valid assumption that $k \gg p$, the marginalization component of ESEIF sparsification is quadratic in the bounded number of active features and, thus, constant-time.

The $\bar{\Lambda}_{M x_t}(\bar{\Lambda}_{x_t x_t})^{-1}\bar{\Lambda}_{x_t M}$ outer product in the Schur complement (19a) is zero everywhere except for the entries that pair the active features. Recalling our earlier discussion on marginalization in §II-B, this establishes connectivity among the active features as we show in the right-hand side of Figure 5. Of course, unlike the figure shows, we do not have a representation for the robot pose, which brings us to the next step.

We conclude sparsification by relocalizing the vehicle within the map with the remaining $\mathbf{z}_\beta$ observations of a set of features denoted by the random vector $\mathbf{m}_\beta$. The expression for the new pose is a nonlinear function of $\mathbf{m}_\beta$ and the measurement data. We include an additive white Gaussian noise term, $\mathbf{w}_t \sim \mathcal{N}\big(\mathbf{0}, \mathrm{R}\big)$ to account for model uncertainty and sensor noise, giving rise to the expression in (20a). Equation (20b) is the first order linearization with respect to the mean vector for the observed features, $\breve{\boldsymbol{\mu}}_{m_\beta}$, from the map distribution (18). The Jacobian matrix, G, is sparse with nonzero entries only within the columns associated with the $\mathbf{m}_\beta$ landmarks. In turn, (20b) requires only the $\breve{\boldsymbol{\mu}}_{m_\beta}$ mean.

$$\mathbf{x}_t = \mathbf{g}\big(\mathbf{m}_\beta, \mathbf{z}_\beta\big) + \mathbf{w}_t \tag{20a}$$

$$\approx \mathbf{g}\big(\breve{\boldsymbol{\mu}}_{m_\beta}, \mathbf{z}_\beta\big) + \mathrm{G}\big(\mathbf{m} - \breve{\boldsymbol{\mu}}_t\big) + \mathbf{w}_t \tag{20b}$$

We augment the map state with this new pose, $\boldsymbol{\xi}_t = [\mathbf{x}_t^\top \ \mathbf{M}_t^\top]^\top$, and form the joint distribution,

$$p_{\text{ESEIF}}\big(\mathbf{x}_t, \mathbf{M}_t \mid \mathbf{z}^t, \mathbf{u}^t\big) = p\left(\mathbf{x}_t \mid \mathbf{m}_\beta, \mathbf{z}_\beta\right) p_2\big(\mathbf{M}_t \mid \left\{\mathbf{z}^{t-1}, \mathbf{z}_\alpha\right\}, \mathbf{u}^t\big), \tag{21}$$

where the factorization captures the conditional independence between the pose and the remaining map elements.

The problem of adding the robot pose is fundamentally the same as adding a new feature to the map or augmenting the state as part of the time prediction step (9). One can then easily show that (22) is the canonical parametrization for $p_{\text{ESEIF}}\left(\boldsymbol{\xi}_t \mid \mathbf{z}^t, \mathbf{u}^t\right)$.

$$p_{\text{ESEIF}}\big(\boldsymbol{\xi}_t \mid \mathbf{z}^t, \mathbf{u}^t\big) = \mathcal{N}^{-1}\big(\boldsymbol{\xi}_t; \breve{\boldsymbol{\eta}}_t, \breve{\Lambda}_t\big)$$

$$\breve{\Lambda}_t = \begin{bmatrix} \mathrm{R}^{-1} & -\mathrm{R}^{-1}\mathrm{G} \\ -\mathrm{G}^\top\mathrm{R}^{-1} & \breve{\Lambda}_t + \mathrm{G}^\top\mathrm{R}^{-1}\mathrm{G} \end{bmatrix} \tag{22a}$$

$$\breve{\boldsymbol{\eta}}_t = \begin{bmatrix} \mathrm{R}^{-1}\left(\mathbf{g}\big(\breve{\boldsymbol{\mu}}_{m_\beta}, \mathbf{z}_\beta\big) - \mathrm{G}\breve{\boldsymbol{\mu}}_t\right) \\ \breve{\boldsymbol{\eta}}_t - \mathrm{G}^\top\mathrm{R}^{-1}\left(\mathbf{g}\big(\breve{\boldsymbol{\mu}}_{m_\beta}, \mathbf{z}_\beta\big) - \mathrm{G}\breve{\boldsymbol{\mu}}_t\right) \end{bmatrix} \tag{22b}$$

As a consequence of the sparseness of G, a majority of terms within the $-\mathrm{R}^{-1}\mathrm{G} = -\big(\mathrm{G}^\top\mathrm{R}^{-1}\big)^\top$ blocks of the information matrix that link the robot to the map are zero. The landmarks used for relocalization are the only exception as we show in the right-hand diagram in Figure 5 with the robot linked to the $\mathbf{m}_\beta = \{\mathbf{m}_2, \mathbf{m}_5\}$ features but no others.

The ESEIF controls the information constraints between the vehicle and the map in a consistent manner since it does not break (i.e. set to zero) undesired links. Instead, the filter marginalizes over the pose, in effect, distributing the information encoded within these links to features in the active map, $\mathbf{m}^+$. The marginalization (19a) populates the information sub-matrix associated with $\mathbf{m}^+$, which then forms a clique in the graph. This fill-in would otherwise occur as part of the next time prediction step and, with the active

map growing ever larger, would fully populate the matrix. The ESEIF avoids extensive fill-in by bounding the number of active landmarks. When the active map reaches a predetermined size, the ESEIF "kidnaps" the robot, sacrificing temporal information as well as a controlled amount of fill-in. The algorithm then relocalizes the vehicle, creating a new set of active features. Since observations are typically confined to the robot's local environment, these features are spatially close. The active map is built up from neighboring landmarks until the next sparsification. As a result, the ESEIF forms marginalization cliques that resemble submaps that are structured according to robot's visibility and the density of features in the environment.

B. Mean Recovery

The sparse information filter provides for a *near* constant-time SLAM implementation. The caveat is, in part, a consequence of the fact that we no longer have access to the mean vector when the posterior is represented in the canonical form. Naïvely, we can compute the entire mean vector as $\boldsymbol{\mu}_t = \Lambda_t^{-1}\boldsymbol{\eta}_t$, though the cost of inverting the information matrix is cubic in the number of states, making it intractable even for small maps.

Instead, we pose the problem as one of solving the set of linear equations

$$\Lambda_t \boldsymbol{\mu}_t = \boldsymbol{\eta}_t \tag{23}$$

and take advantage of the sparseness of the information matrix. There are a number of techniques that iteratively solve such sparse, symmetric positive definite systems of equations including conjugate gradient descent [27] as well as relaxation-based algorithms such as Gauss-Seidel [28] and, more recently, the multilevel method proposed by [19]. The optimizations can often be performed over the course of multiple time steps since, aside from loop closures, the mean vector evolves slowly in SLAM. As a result, we can bound the number of iterations required at any one time step [17].

Oftentimes, we are only interested in a subset of the mean such as during the time projection step, which requires an estimate for the robot pose. We can then consider partial mean recovery [21] in which we partition (23) as

$$\begin{bmatrix} \Lambda_{ll} & \Lambda_{lb} \\ \Lambda_{bl} & \Lambda_{bb} \end{bmatrix} \begin{bmatrix} \boldsymbol{\mu}_l \\ \boldsymbol{\mu}_b \end{bmatrix} = \begin{bmatrix} \boldsymbol{\eta}_l \\ \boldsymbol{\eta}_b \end{bmatrix} \tag{24}$$

where $\boldsymbol{\mu}_l$ is the "local portion" that we want to solve for and $\boldsymbol{\mu}_b$ is the "benign portion" of the map. Given an estimate for $\boldsymbol{\mu}_b$, we can reduce (24) to an approximate solution for the local mean,

$$\hat{\boldsymbol{\mu}}_l = \Lambda_{ll}^{-1}\left(\boldsymbol{\eta}_l - \Lambda_{lb}\hat{\boldsymbol{\mu}}_b\right). \tag{25}$$

Due to the sparsity of Λ_{lb}, this formulation requires only a subset of $\hat{\boldsymbol{\mu}}_b$, corresponding to the Markov blanket for the local map. Assuming that we have an accurate estimate for the mean of this portion of the benign map, (25) provides an efficient approximation to the mean that we are interested in.

C. Data Association

The successful implementation of any SLAM algorithm requires the ability to correctly match observations of the environment with the associated landmarks in the map. The data association problem is often addressed by choosing the feature that best explains the measurement, subject to a threshold that identifies spurious observations. For a particular correspondence, the likelihood follows from the marginal distribution for the particular states associated with the hypothesis (typically the robot pose, $\mathbf{x}_t$, and a single landmark, $\mathbf{m}_i$), $p\left(\mathbf{x}_t, \mathbf{m}_i \mid \mathbf{z}^{t-1}, \mathbf{u}^t\right)$. Unfortunately, the information form is not amenable to computing this marginal from the full joint posterior since, referring back to Table I, the Schur complement requires the inversion of a large matrix.

Consequently, the traditional approach to data association is not an option for scalable information filters. Instead, Thrun *et al.* [13] approximate the measurement likelihood from a conditional distribution rather than the marginal. Specifically, the SEIF considers the Markov blanket, $\mathrm{MB}(\mathbf{x}_t, \mathbf{m}_i)$, for $\mathbf{x}_t$ and

$\mathbf{m}_i$ consisting of all states directly linked in the GMRF to either $\mathbf{x}_t$ or $\mathbf{m}_i$. The SEIF first computes the conditional distribution $p\left(\mathbf{x}_t, \mathbf{m}_i, \mathrm{MB}(\mathbf{x}_t, \mathbf{m}_i) \mid \mathbf{M}', \mathbf{z}^{t-1}, \mathbf{u}^t\right)$ where $\mathbf{M}'$ denotes all state elements not in $\{\mathbf{x}_t, \mathbf{m}_i, \mathrm{MB}(\mathbf{x}_t, \mathbf{m}_i)\}$. This distribution is then marginalized over the Markov blanket to achieve an approximation to the desired marginal, $p\left(\mathbf{x}_t, \mathbf{m}_i \mid \mathbf{M}', \mathbf{z}^{t-1}, \mathbf{u}^t\right)$, which is used to determine the likelihood of the hypothesis. The cost of conditioning on $\mathbf{M}'$ is negligible and does not depend on the size of the map. Once most of the map has been conditioned away, the matrix that is inverted as part of the subsequent marginalization is now small, on the order of the size of the Markov blanket. The resulting distribution has been successfully utilized for data association with SEIFs [29], though it has been demonstrated to yield overconfident estimates for the likelihood of measurements [30].

The marginal is easily determined from the standard parametrization, described by the mean and sub-blocks of the full covariance matrix corresponding to $\mathbf{x}_t$ and $\mathbf{m}_i$. Inverting the information matrix to access the covariance, though, is equivalent to performing the marginalization in the canonical form and is, thus, impractical. Alternatively, Eustice *et al.* [30] propose an efficient method for approximating the marginal that gives rise to a conservative measure for the hypothesis likelihood. The technique stems from posing the relationship, $\Lambda_t \Sigma_t = \mathrm{I}$, as a sparse system of linear equations, $\Lambda_t \Sigma_{\star i} = \mathbf{e}_i$, where $\Sigma_{\star i}$ and $\mathbf{e}_i$ denote the i^{th} columns of the covariance and identity matrices, respectively. They estimate the robot pose joint-covariance, $\Sigma_{\star x_t}$, online by solving the system of equations with one of the iterative algorithms mentioned for mean recovery. The algorithm combines this with a conservative estimate for the feature covariance to achieve the representation for the marginal covariance. The marginal, which is itself conservative, is then used for data association.

VI. Results

This section explores the effectiveness of the ESEIF algorithm in comparison to the SEIF and EKF when applied to different forms of the SLAM problem. We first present the results of a controlled linear Gaussian (LG) SLAM simulation that allows us to compare the consequences of the different sparsification strategies relative to the "gold standard" Kalman Filter (i.e. the optimal Bayes estimator). We then discuss the performance of the sparsified information algorithms on a pair of real-world, nonlinear SLAM problems including the benchmark Sydney Park outdoor dataset widely popular in the SLAM community.

A. Linear Gaussian Simulation

In an effort to better understand the theoretical consequences of enforcing sparsity in information filters, we first study the effects of applying the different approaches in a controlled simulation. In this example, the environment is comprised of a set of point features, located according to a uniform distribution that yields a desired density of 0.10 features per unit area. The robot moves translationally according to a linear, constant-velocity model and measures the relative position of a bounded number of neighboring features. Both the measurements as well as the vehicle motion are corrupted by additive white Gaussian noise.

We implement the ESEIF and SEIF using their corresponding sparsification routines to maintain a bound of $\Gamma_a = 10$ active features. In the case of ESEIF sparsification, we reserve as many of the measurements as possible for the relocalization component, to the extent that we do not violate the Γ_a bound (i.e. $|\mathbf{z}_\beta| \leq \Gamma_a$). Additionally, we apply the standard Kalman filter that, by the linear Gaussian (LG) nature of the simulation, is the optimal Bayesian estimator. Aside from the different sparsification routines, each estimator is otherwise identical.

Our main interest in the LG simulation is to evaluate the effect of the different sparsification strategies on the estimation accuracy. To that end, we perform a series of Monte Carlo simulations, using two formulations of the normalized estimation error squared (NEES) [31] as a measure of filter consistency. The first metric considers the *global* error between the unadulterated filter estimates for the vehicle and feature positions and their ground truth positions. We compute this score over several simulations and plot the averages in Figures 6(a) and 6(b) for the vehicle and a single landmark, respectively. The $97.5^{\%}$

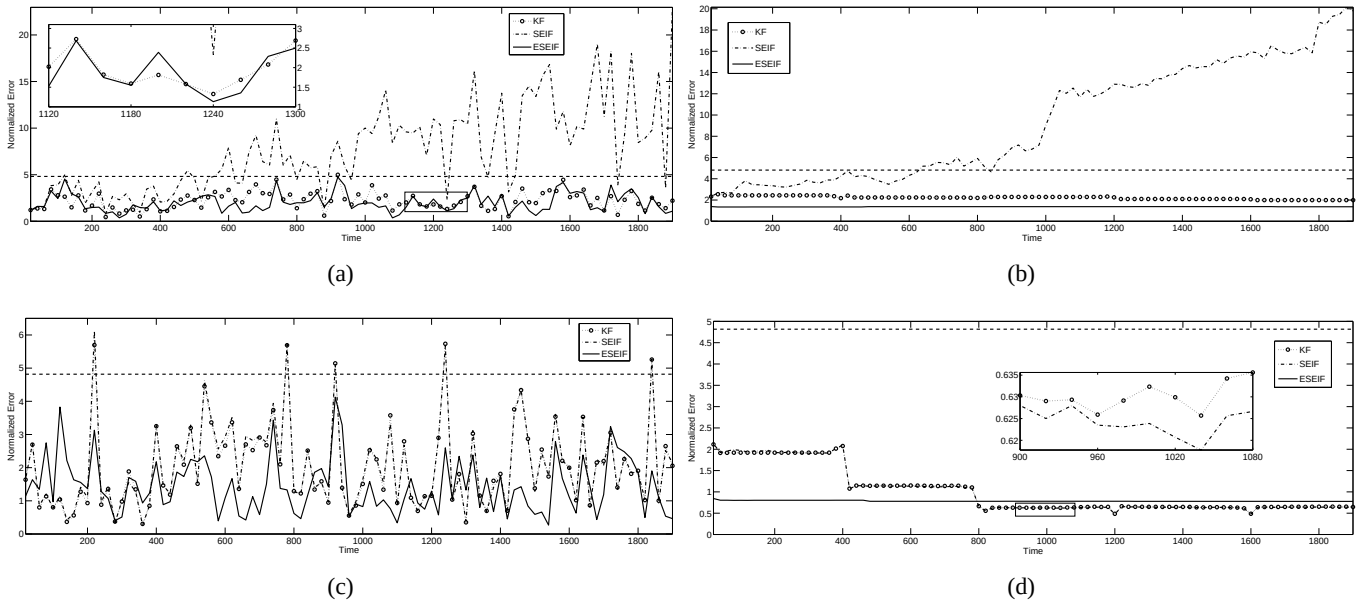
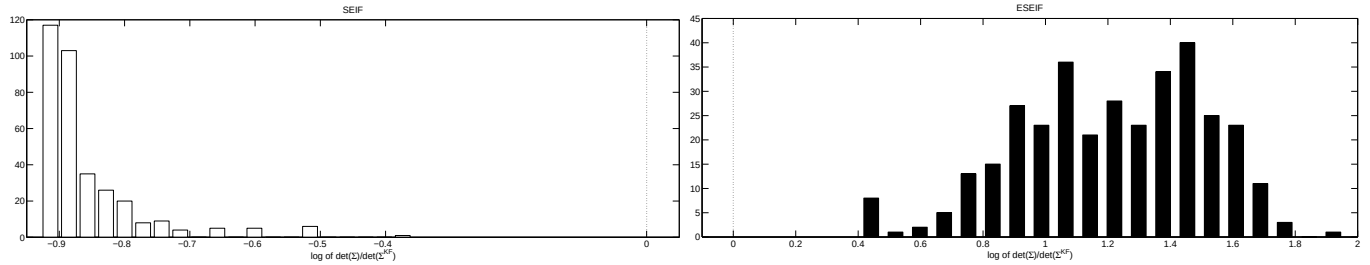


Fig. 6. Plots of the normalized estimation error squared (NEES) measured based upon a series of Monte Carlo simulations of linear Gaussian SLAM. The *global* errors associated with the estimates for (a) vehicle pose and (b) a single feature representative of the map are computed by comparing the direct filter estimates with ground truth and provide a measure of global consistency. The plots in (c) and (d) correspond to the *local* error in the vehicle and feature estimates, respectively, that follows from expressing the state relative to the first feature added to the map. The horizontal threshold denotes the $97.5^{\%}$ chi-square upper bound and serves as a test for the consistency of the different filters. For both the vehicle and the map, the global as well as local ESEIF errors satisfy the chi-square limit. The same is true of the local measure for the SEIF yet the global errors are significantly greater and far exceed the chi-square bound.

chi-square upper limit for the series of simulations is denoted by the horizontal threshold, which the KF normalized errors largely obey. The SEIF vehicle pose error is significantly larger than that of the KF and ESEIF, and exceeds the chi-square bound for most of the simulation. The same is true of the estimate for the landmark positions. This behavior indicates that SEIFs maintain an absolute state estimate that is inconsistent. In contrast, the ESEIF yields global errors for both the vehicle and map that are similar to the KF and pass the chi-square test. This suggests that the ESEIF SLAM distribution is globally consistent.

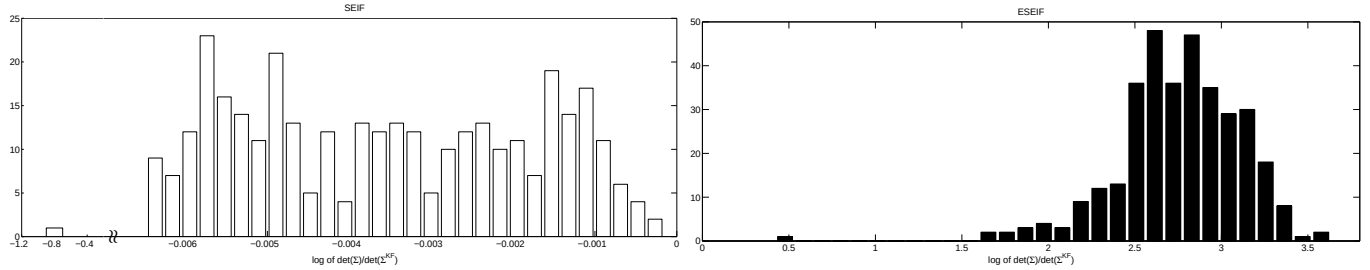
The second normalized error concerns the accuracy of the relative state elements. We first reference the vehicle and map positions relative to the first observed feature, $\mathbf{x}_m$, via the compounding operation, $\mathbf{x}_{mi} = \ominus\mathbf{x}_m \oplus \mathbf{x}_i$ [2]. We then measure the *local* error by comparing the relative map estimates to the root-shifted ground truth positions. The local error in the estimates of the vehicle and the same feature as in Figure 6(b) are shown in Figures 6(c) and 6(d), respectively, together with the $97.5^{\%}$ chi-square bound. Unlike the global estimates, the SEIF sparsification results in local errors that are nearly indistinguishable from those of the KF. Furthermore, the SEIF appears to satisfy the chi-square test as the errors rarely exceed the threshold. The local errors maintained by the ESEIF also fall well below the chi-square limit. It seems, then, that while SEIFs are globally inconsistent, the sparsification routine preserves the consistency of the relative map relationships. The ESEIF, on the other hand, maintains a posterior that is both globally and locally consistent.

The high global NEES scores for the SEIF are not so much a consequence of error in the vehicle and map estimates as they are of the overconfidence of the SEIF in these estimates. This becomes apparent when the SEIF's uncertainty estimates are compared against the true distribution maintained by the Kalman Filter. We recover the map covariance from the information matrix and, for each landmark, compute the log of the ratio of the covariance sub-block determinant to the determinant of the KF sub-block. Since the KF estimate represents the true distribution, log ratios less than zero signify overconfidence while values greater than zero imply conservative uncertainty estimates. Figure 7 presents a histogram plot of these ratios for the two information filters. The SEIF uncertainty bounds for the global map are significantly



(a) (b)

Fig. 7. Histograms for the LG simulation describing the global map uncertainty maintained by the (a) SEIF and (b) ESEIF as compared with that of the KF. For each feature, we compute the log of the ratio between the information filter covariance sub-block determinant and the determinant for the actual distribution as given by the KF. Values greater than zero imply conservative estimates for the uncertainty while log ratios less than zero indicate overconfidence. Note that all of the SEIF estimates are overconfident while those of the ESEIF are conservative.



(a) (b)

Fig. 8. The uncertainty attributed to the relative map estimates for the (a) SEIF and (b) ESEIF expressed relative to the optimal KF. The uncertainty ratios are determined as before, in this case based upon the local covariance estimates that follow from root-shifting the state to the first feature added to the map. While still overconfident, the SEIF local uncertainty estimates are significantly closer to the values maintained by the KF with the one exception being the representation of the original world origin in the relative map. The ESEIF, meanwhile, produces relative map estimates that are conservative.

smaller than those of the KF, indicating that the SEIF posterior is susceptible to overconfidence as a consequence of the sparsification strategy. This agrees with our discussion in §IV-B on the inherent implications of enforcing sparsity by approximating conditional independence. In comparison, the ESEIF maintains confidence estimates for each landmark that are conservative with respect to the Kalman Filter.

In similar fashion to the NEES score, when we transform the map relative to the first feature, we see in Figure 8(a) that the SEIF and KF estimates for the local uncertainty agree much more closely than do the global estimates. The one exception is the representation for the original world origin in the relative map, which the SEIF assigns a higher measure of confidence. Overall, though, the SEIF is far less overconfident in the accuracy of its local estimates, which helps to explain the reduced normalized error values we saw earlier. Meanwhile, the histogram in Figure 8(b) demonstrates that the ESEIF estimates for the local map accuracy remain conservative relative to the true distribution.

Figure 9 illustrates the computational benefits of the ESEIF over the KF. Plotted in Figure 9(a), the KF update time grows quadratically with the number of states. In contrast, the ESEIF and SEIF updates remain constant-time despite an increase in the state dimension. While this efficiency is inherent to information filter updates, sparseness is beneficial for the prediction step, which is quadratic in size of the map for non-sparse information filters. We see this benefit in Figure 9(b) as the prediction time is similar for all three filters, with a gradual increase with the number of features. Additionally, the memory requirements for sparse matrices are considerably less than those of the covariance matrix. Consider the density of the three matrices that are each 536×536 at the end of the simulation. The covariance matrix is fully-populated, yet $92^{\%}$ of the terms in the ESEIF information matrix are exactly zero as is $89^{\%}$ of the SEIF

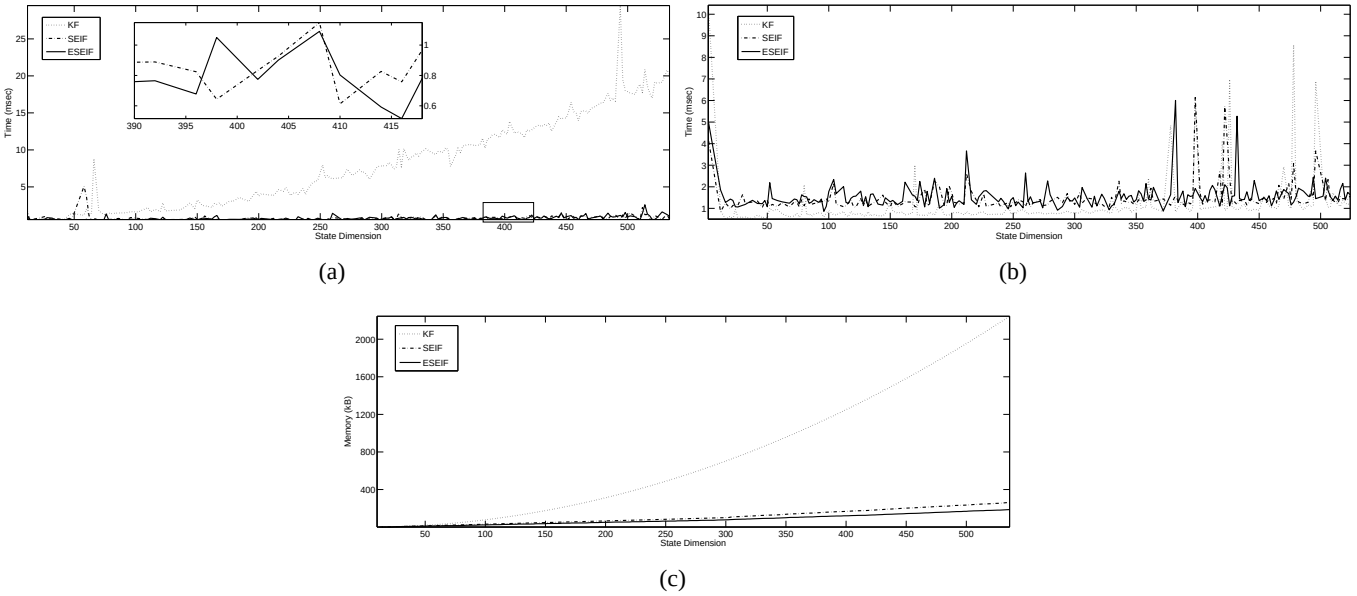


Fig. 9. A comparison of the performance of the ESEIF, SEIF, and KF for a LG simulation. The update times (a) for the ESEIF and SEIF are nearly identical and remain constant with the growth of the map. In contrast, the KF exhibits the well-known quadratic increase in complexity. The prediction times (b) gradually increase with the map size and are similar for the three filters by virtue of the sparsity of the information matrices. The plot in (c) reveals that the sparse information forms demand significantly less memory than the fully-populated covariance matrix.

matrix. Figure 9(c) plots the difference in the memory requirements as a function of the state dimension.

B. Experimental Validation

The linear Gaussian simulations allow us to explore the theoretical implications of sparsification and validate our claims that approximating the conditional independence of the robot and a set of map elements leads to an inconsistent distribution. The results empirically show that the ESEIF provides a sparse representation of the canonical Gaussian while simultaneously preserving consistency. Unfortunately, the simulations are not representative of most real-world applications, which generally involve motion and measurement models that are nonlinear and noise that is non-Gaussian. To study the performance of the ESEIF under these circumstances, we apply it to two nonlinear datasets, along with the SEIF and standard EKF.

Victoria Park Dataset

For the first real-world SLAM problem, we consider the benchmark Victoria Park dataset courtesy of E. Nebot of the University of Sydney [5]. The dataset is widely popular in the SLAM community as a testbed for different algorithms that address the scalability problem [5], [8], [13], [32]. In the experiment, a truck equipped with odometry sensors and a laser range-finder drives in a series of loops within Victoria Park, Sydney, shown in Figure 10 along with a rough plot of the GPS trajectory. We use a simple perceptual grouping implementation to detect tree trunks located throughout the park among the laser data, which is cluttered with spurious returns. We solve the data association problem offline to ensure that the correspondences are identical for each filter.

We apply the SEIF and ESEIF algorithms together with the EKF, which has been successfully applied to the dataset in the past [5]. We limit the size of the active map to a maximum of $\Gamma_a = 10$ features for the two information filters. As with the LG simulation, we place a priority on the relocation step when sparsifying the ESEIF, reserving as many tree observations as possible (i.e. no more than $\Gamma_a = 10$) for the sake of adding the vehicle back into the map. Any additional measurements are used to update the filter prior to marginalization. This helps to minimize the influence of spurious observations on the estimate for the relocated vehicle pose.

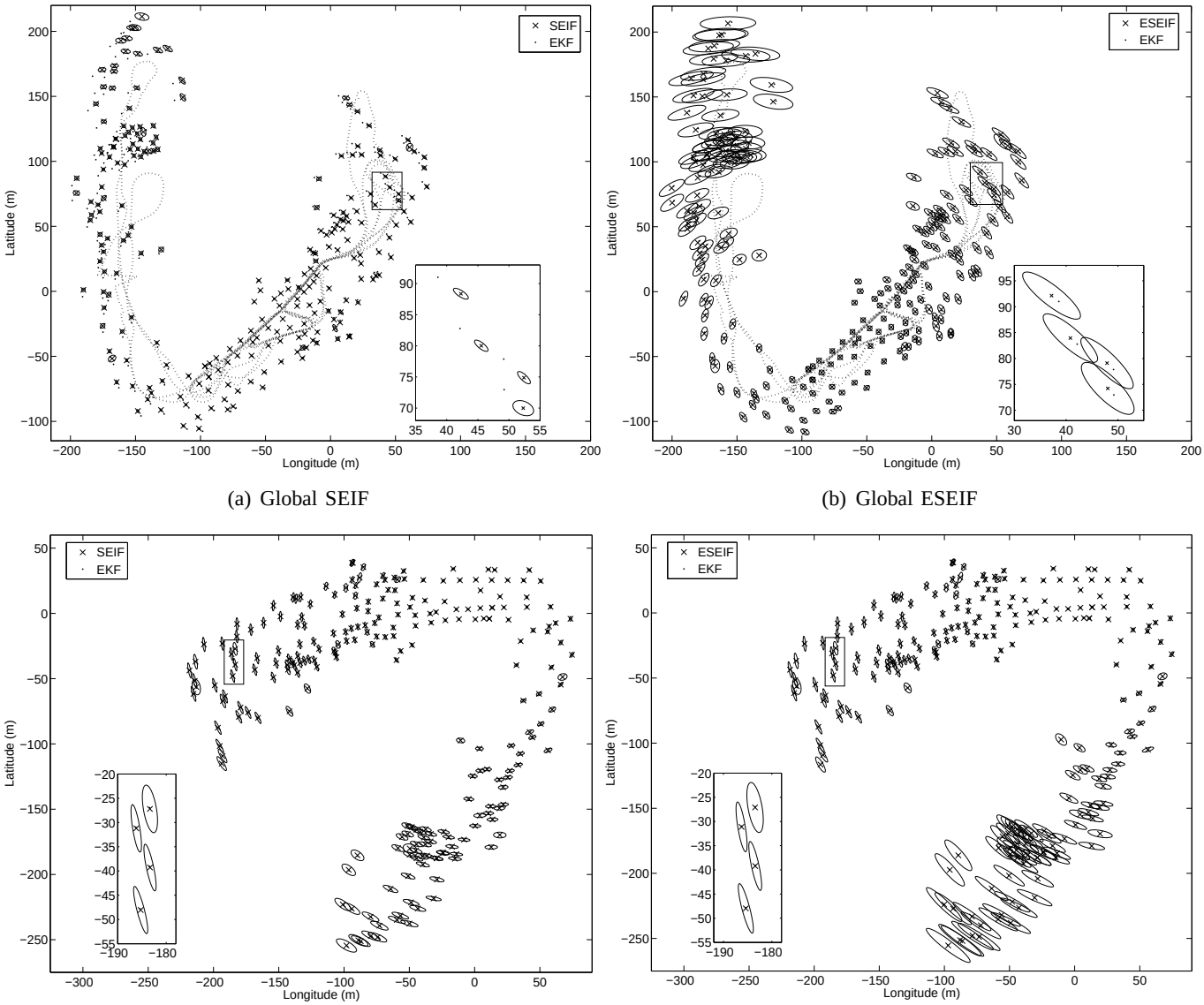
Fig. 10. An overhead image of Victoria Park in Sydney, Australia along with a rough plot of the GPS vehicle trajectory. The environment is approximately 250 meters East to West and 300 meters North to South.

The final SEIF and ESEIF maps are presented in Figures 11(a) and 11(b), respectively, along with the estimate for the robot trajectory. The ellipses denote the three-sigma uncertainty bounds estimated by the two filters. As a basis for comparison, we plot the map generated by the EKF, which is similar to results published elsewhere. One sees that the feature position estimates are similar for the three filters, yet the SEIF exhibits a larger deviation from the EKF map than does the ESEIF. The most obvious distinction between the two maps, though, is the difference in the estimated accuracy of the maps indicated by the uncertainty ellipses. While not ground truth, the EKF results represent the baseline that the information filters seek to emulate, yet many of the EKF feature estimates fall outside the three-sigma SEIF uncertainty bounds. This is particularly evident in the periphery as we reveal in the inset plot. The ESEIF confidence regions, on the other hand, capture all of the EKF landmark estimates.

The difference becomes more apparent when we directly compare the uncertainty measures for each feature. Figure 12(a) presents a histogram plot of the log ratio between the global feature covariance determinants for the SEIF and ESEIF with respect to the EKF determinants. The SEIF global uncertainty estimates are all smaller than those of the EKF while the ESEIF estimates are larger. This is consistent with the linear Gaussian simulation results and suggests that the SEIF sparsification strategy results in an overconfident SLAM posterior while the ESEIF produces a distribution that is conservative with respect to the EKF.

In similar fashion to the LG experiment, we observe contrasting behavior for the relative map that follows from root-shifting the state relative to the vehicle's final pose. The SEIF map shown in Figure 11(c) and the ESEIF map plotted in Figure 11(d) are both nearly identical to the relative EKF map. Furthermore, the three-sigma relative uncertainty bounds maintained by the two filters contain the EKF position estimates. Nonetheless, the SEIF is still more confident than the EKF as the histogram in Figure 12(b) indicates. Aside from the representation for the original world origin, though, the local SEIF uncertainties are nearly identical to those of the EKF. Together, this implies that the consistency of the relative state distribution is less sensitive to the approximations used in the SEIF sparsification. Meanwhile, the ESEIF estimates for the relative uncertainty remain conservative with respect to the EKF.

Figure 13(a) compares the total time required for the time prediction and measurement update steps for the ESEIF and EKF. We do not include the SEIF performance but note that it is similar to that of the ESEIF. The ESEIF implementation employed partial mean recovery (25), solving the full set of equations only upon sparsification. The EKF is more efficient when the map is small (less than 50 landmarks), a



(a) Global SEIF

(b) Global ESEIF

(c) Relative SEIF

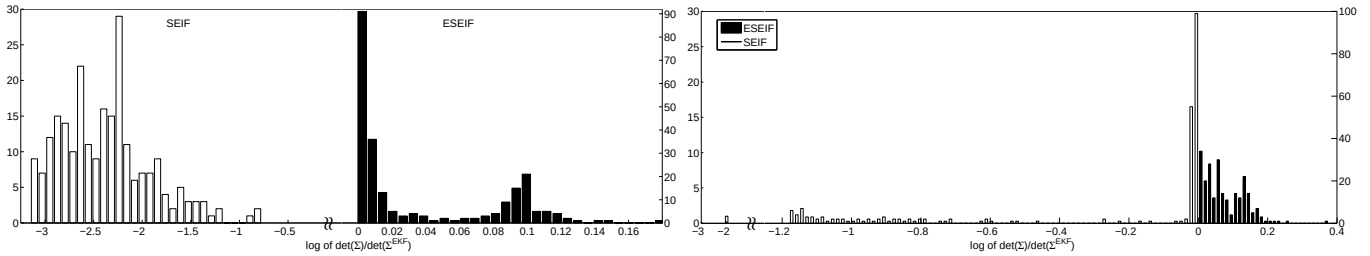
(d) Relative ESEIF

Fig. 11. Estimates for the vehicle trajectory and feature positions along with the three sigma confidence bounds for the Victoria Park dataset. The global maps generated by (a) the SEIF and (b) the ESEIF are similar to the EKF map. The SEIF uncertainty ellipses, though, are significantly smaller than those of the ESEIF and, in many cases, do not include the EKF feature estimates. In (c) and (d) we plot the relative SEIF and ESEIF maps, respectively, that follow from root-shifting the state into the reference frame of the robot at its final pose. The three relative maps are nearly identical and the SEIF uncertainty bounds are not nearly as small, capturing each of the EKF position estimates.

reflection of the ESEIF prediction time that is quadratic in the number of active features along with the mean estimation cost. Yet, as the map grows larger, the quadratic update of the EKF quickly dominates the filtering time of the ESEIF, which varies with the number of active features rather than the state dimension.

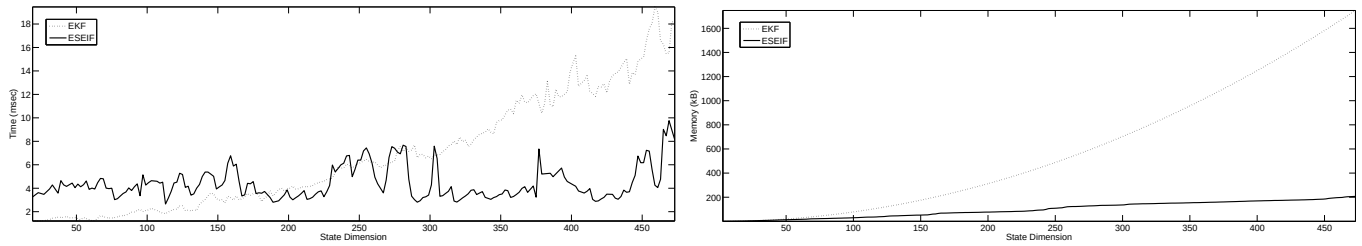
The plot in Figure 13(b) displays the EKF and ESEIF memory allocations. In order to store the correlations among the map and robot pose, the fully-populated EKF covariance matrix requires quadratic storage space. The ESEIF information matrix, though, is sparse with a bounded quantity of non-zero terms that pair the vehicle and map and a linear number of links between landmarks. As a result, we see that the ESEIF storage requirement is linear in the size of the map.

Hurdles Dataset



(a) (b)

Fig. 12. Histograms for the Victoria Park dataset comparing the ESEIF and SEIF uncertainty estimates to the results of the EKF. We again use the log of the ratio of the covariance sub-block determinants for each landmark. The plot in (a) describes the global map uncertainties while the histogram in (b) corresponds to the relative map. The SEIF marginal distributions are largely overconfident when compared with the EKF for the global map, but less so for the relative feature estimates. The representation of the world origin in the root-shifted map is the one outlier in the latter. The ESEIF is conservative with respect to the EKF both globally and locally.



(a) (b)

Fig. 13. Plots of the computational efficiency of the EKF and ESEIF for the Victoria Park dataset. In (a) we show the total prediction and update time as a function of state dimension. The complexity of the EKF increases with the size of the map while the ESEIF does not. Instead, the ESEIF cost is a function of the number of active features. Shown in (b), the EKF memory requirement is quadratic in the size of the map, yet only linear for the ESEIF.

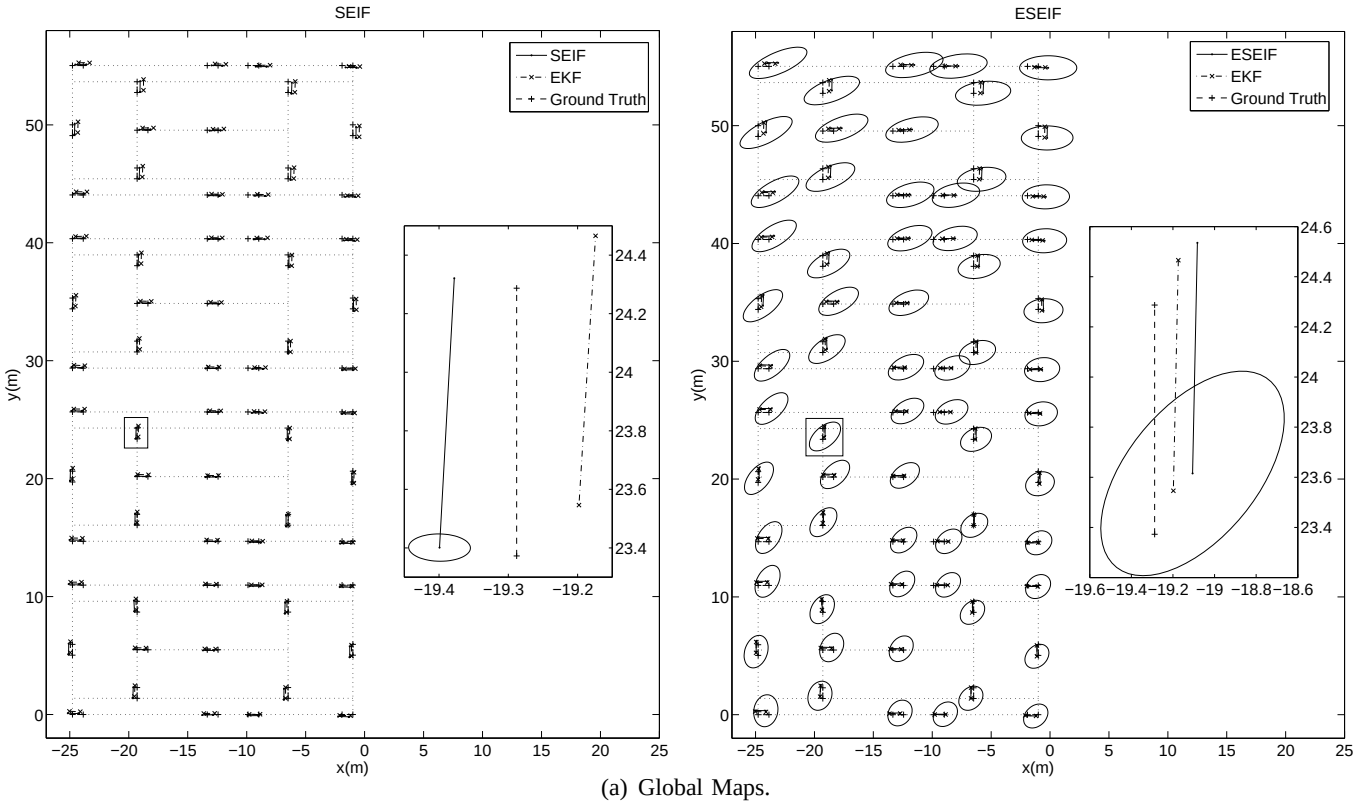
In the second experiment, a wheeled robot drives among 64 track hurdles positioned at known locations along the baselines of four adjacent tennis courts. The vehicle observes nearby hurdles with a SICK laser scanner and uses wheel encoders to measure pose velocity inputs for the kinematic motion model.

We again apply the ESEIF, SEIF, and EKF SLAM algorithms. The data association problem is solved independently such that the correspondences are identical for all three filters. The maximum number of active landmarks for the three information filters is set at $\Gamma_a = 10$ hurdles. As with the Victoria Park dataset, we prefer to relocalize the vehicle during sparsification with as many measurements as possible and use any surplus observations in the preceding update step.

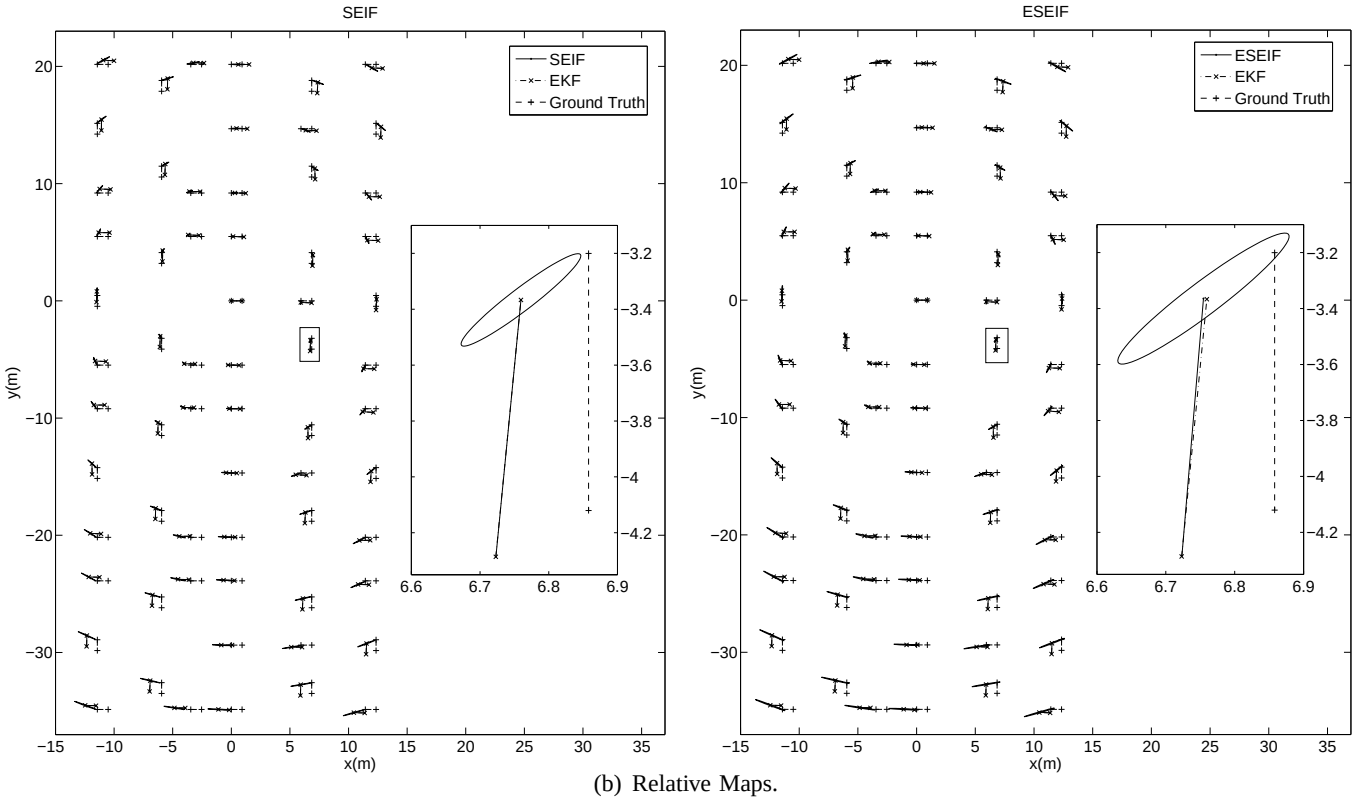
We present the final map estimates for the ESEIF and SEIF in Figure 14 along with the EKF map and the ground truth poses. The ellipses denote the three-sigma uncertainty bounds for the position of a leg of each hurdle. Qualitatively, the maps for the information filters closely agree with the EKF estimates as well as the true hurdle positions, however the same is not true for the global uncertainties. The SEIF is again unique in that sparsification results in global uncertainty estimates that are too small to capture a majority of the true hurdle positions, indicative of an overconfident SLAM posterior. Figure 14(b) shows that SEIF estimates are more accurate upon root-shifting to the first hurdle added to the map. The ESEIF global and relative maps are comparable to those of the SEIF and EKF as well as the ground-truth. Unlike the SEIF, though, both the global and local ESEIF uncertainty estimates are consistent with the EKF.

VII. Discussion

We have taken a closer look at the SEIF sparsification strategy and, in particular, the consequences on the uncertainty estimates. We presented an alternative algorithm for maintaining sparsity and have shown that it does not suffer from the same overconfidence. In this section, we elaborate on our claims regarding



(a) Global Maps.



(b) Relative Maps.

Fig. 14. The final maps for the hurdles dataset generated with the SEIF and ESEIF compared with the EKF estimates and the ground truth hurdle positions. The ellipses define the three-sigma uncertainty bounds on the location of the base leg of each hurdle. The only exception is the inset plot for the global ESEIF map where, for aesthetic reasons, we plot the one-sigma uncertainty region. In (a) we show the global estimates given directly from the three filters while (b) contains the relative maps transformed with respect to the first hurdle added to the map. As indicated in (a), the SEIF maintains global uncertainty estimates that are overconfident while the plot below reveals that the it retains the local map structure. In comparison, the ESEIF yields estimates that are consistent with the EKF both for the global and relative maps.

the consistency of the ESEIF. In addition, we draw comparisons between the ESEIF and the D-SLAM algorithm [22], which similarly achieves sparsity while preserving consistency.

A. Estimator Consistency

The results presented in the previous section empirically demonstrate that the SEIF global uncertainty estimates are noticeably overconfident while the ESEIF is globally and locally conservative. In the linear Gaussian case, this is sufficient to conclude that the ESEIF preserves the consistency of the SLAM posterior for the local and global representations. On the other hand, as the ESEIF is based upon the dual of the EKF, it is subject to the same convergence issues as the EKF for nonlinear applications [31]. While the results empirically demonstrate that the ESEIF is conservative with respect to the EKF, this does not guarantee that the ESEIF SLAM posterior is consistent with the true, non-Gaussian distribution. Nonetheless, the algorithm allows us to capitalize on the computational and storage benefits of a sparse information form without incurring additional inconsistency. The EKF has been successfully applied to a wide range of real-world datasets and the ESEIF provides a scalable means of achieving nearly identical estimates.

B. Comparison with D-SLAM

In [22], Wang *et al.* propose a similar algorithm that maintains a sparse canonical parametrization in a consistent manner. The approach decouples SLAM into separate localization and map building problems and addresses them concurrently with different estimators. The D-SLAM considers the map distribution, $p\left(\mathbf{M} \mid \mathbf{z}^t, \mathbf{u}^t\right)$, to be Gaussian and represents it in the canonical form. It then uses an EIF to maintain the information matrix and vector with updates based upon relative landmark measurements that have been extracted from the robot's observations of the environment. The EIF time projection step is trivial since the robot pose is not contained in this distribution and, in turn, the information matrix is naturally sparse. An estimate for the vehicle pose is determined from map observations by solving the kidnapped robot problem at each time step. Additionally, D-SLAM implements a standard EKF SLAM process for the robot's local neighborhood that provides a second estimate of pose. To account for unmodeled correlation between the two estimates, they are fused with covariance intersection [23] to achieve a conservative belief over pose. By decoupling the problem in this way, D-SLAM capitalizes on an exactly sparse information matrix without sacrificing consistency.

The key component to maintaining the sparseness of the information matrix follows from the observation that the time projection step for the robot pose causes fill-in. By periodically kidnapping and relocalizing the robot, the ESEIF controls the population of the information matrix. The D-SLAM algorithm takes this one step farther by essentially performing kidnapping and relocalization at each time step. As a result, they sacrifice nearly all information provided by the temporal constraints between successive poses. Additionally, in order to preserve exact sparsity for the map distribution, the algorithm does not incorporate any knowledge of the robot's pose when building or maintaining the map. We believe the D-SLAM estimator to be less optimal as it ignores markedly more information than the ESEIF, which only occasionally disregards temporal links.

VIII. Conclusion

To summarize, the computational demands of the Extended Kalman Filter limit its use in feature-based SLAM to small environments. This problem is currently a hot research topic in robotics and has lead to a number of different algorithms that scale with larger maps. In particular, the key observation that the canonical SLAM distribution is relatively sparse has given rise to scalable adaptations to the information filter. The algorithms take advantage of the fact that, when the density of the information matrix is bounded, estimation can be performed in near constant time, irrespective of the number of landmarks. The problem, though, is that while a majority of the elements in the information matrix are relatively

weak, the feature-based SLAM matrix is fully populated. In order to achieve the computational benefits of the sparse form, the algorithms explicitly break these weak links.

The Sparse Extended Information Filter sparsification strategy approximates the conditional independence between the robot and most of the map. We have examined the consequences of performing inference based upon this approximation to the SLAM posterior. The results demonstrate that the SEIF estimates for the robot pose and map suffer from global inconsistency, yet they empirically preserve relative relationships.

We have presented the Exactly Sparse Extended Information Filter as an efficient feature-based SLAM algorithm. The ESEIF maintains an exactly sparse information matrix without incurring additional global or local inconsistency. The paper has shown that occasionally marginalizing the robot pose from the distribution and subsequently relocalizing the vehicle within the map allows us to control the number of active features and, in turn, the population of the information matrix. The ESEIF then takes advantage of the benefits of a sparse canonical parametrization while maintaining conservative robot pose and map estimates.

We demonstrated the consistency of the ESEIF through a series of controlled linear Gaussian simulations. The algorithm was then applied to two different nonlinear datasets including a benchmark SLAM experiment. The results reveal that the ESEIF maintains estimates nearly identical to those of the EKF with savings in computation time and memory requirements. The ESEIF offers an improvement in scalability while it maintains estimates that are both globally *and* locally conservative.

IX. Acknowledgements

The authors would like to thank the reviewers for their insightful comments. This work was funded in part by the CenSSIS ERC of the NSF under grant EEC-9986821 and in part by the Office of Naval Research under grant N00014-02-C-0210.

References

[1] L. Whitcomb, D. Yoerger, H. Singh, and D. Mindell, "Towards precision robotic maneuvering, survey, and manipulation in unstructured undersea environments," in *Robotics Research – The Eighth International Symposium*, Y. Shirai and S. Hirose, Eds. London: Springer-Verlag, 1998, pp. 45–54.

[2] R. Smith, M. Self, and P. Cheeseman, "Estimating uncertain spatial relationships in robotics," in *Autonomous Robot Vehicles*, I. Cox and G. Wilfong, Eds. Springer-Verlag, 1990, pp. 167–193.

[3] P. Moutarlier and R. Chatila, "An experimental system for incremental environment modelling by an autonomous mobile robot," in *Proceedings of the 1st International Symposium on Experimental Robotics*, Montreal, Canada, June 1989, pp. 327–346.

[4] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. MIT Press, 2005.

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

[6] S. Williams, G. Dissanayake, and H. Durrant-Whyte, "An efficient approach to the simultaneous localisation and mapping problem," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Washington, DC, May 2002, pp. 406–411.

[7] J. Leonard and P. Newman, "Consistent, convergent, and constant-time SLAM," in *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*. Acapulco, Mexico: IJCAI, August 2003, pp. 1143–1150.

[8] M. Bosse, P. Newman, J. Leonard, and S. Teller, "Simultaneous localization and map building in large-scale cyclic environments using the atlas framework," *International Journal of Robotics Research*, vol. 23, no. 12, pp. 1113–1139, December 2004.

[9] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit, "FastSLAM: A factored solution to the simultaneous localization and mapping problem," in *Proceedings of the AAAI National Conference on Artificial Intelligence*. Edmonton, Canada: AAAI, 2002, pp. 593–598.

[10] M. Paskin, "Thin junction tree filters for simultaneous localization and mapping," University of California, Berkeley, Tech. Rep. UCB/CSD-02-1198, September 2002.

[11] R. van der Merwe, A. Doucet, N. de Freitas, and E. Wan, "The unscented particle filter," Cambridge University Engineering Department, Tech. Rep. CUED/F-INFENG/TR380, August 2000.

[12] P. S. Maybeck, *Stochastic Models, Estimation, and Control*. New York, NY: Academic Press, 1979, vol. 1.

[13] S. Thrun, Y. Liu, D. Koller, A. Ng, Z. Ghahramani, and H. Durrant-Whyte, "Simultaneous localization and mapping with sparse extended information filters," *International Journal of Robotics Research*, vol. 23, no. 7-8, pp. 693–716, July-August 2004.

[14] U. Frese and G. Hirzinger, "Simultaneous localization and mapping - a discussion," in *Proceedings of the IJCAI Workshop on Reasoning with Uncertainty in Robotics*, 2001, pp. 17–26.

[15] T. Speed and H. Kiiveri, "Gaussian Markov distributions over finite graphs," *Annals of Statistics*, vol. 14, no. 1, pp. 138–150, March 1986.

[16] U. Frese, "Treemap: An $\mathcal{O}(\log n)$ algorithm for simultaneous localization and mapping," in *Spatial Cognition IV*, C. Freksa, Ed. Springer–Verlag, 2005, pp. 455–476.

[17] T. Duckett, S. Marsland, and J. Shapiro, "Learning globally consistent maps by relaxation," in *Proceeding of the IEEE International Conference on Robotics and Automation (ICRA)*, San Francisco, 2000, pp. 3841–3846.
[18] J. Folkesson and H. Christensen, "Graphical SLAM - a self-correcting map," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, New Orleans, LA, April 2004, pp. 383–390.
[19] U. Frese, P. Larsson, and T. Duckett, "A multilevel relaxation algorithm for simultaneous localization and mapping," *IEEE Transactions on Robotics*, vol. 21, no. 2, pp. 196–207, April 2005.
[20] F. Dellaert, "Square root sam," in *Proceedings of Robotics: Science and Systems (RSS)*, Cambridge, MA, June 2005, pp. 177–184.
[21] R. Eustice, H. Singh, and J. Leonard, "Exactly sparse delayed-state filters," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Barcelona, Spain, April 2005, pp. 2417–2424.
[22] Z. Wang, S. Huang, and G. Dissanayake, "D-slam: Decoupled localization and mapping for autonomous robots," in *Proceedings of the 12th International Symposium of Robotics Research*, San Francisco, CA, October 2005.
[23] S. Julier and J. Uhlmann, "A non-divergent estimation algorithm in the presence of unknown correlations," in *Proceedings of the American Control Conference*, Albuquerque, NM, June 1997.
[24] U. Frese, "A proof for the approximate sparsity of SLAM information matrices," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Barcelona, Spain, April 2005, pp. 331–337.
[25] R. Eustice, M. Walter, and J. Leonard, "Sparse extended information filters: Insights into sparsification," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Edmonton, Alberta, Canada, August 2005, pp. 641–648.
[26] G. Strang, *Linear Algebra and Its Applications*, 2nd ed. New York: Academic Press, 1980.
[27] J. Shewchuck, "An introduction to the conjugate gradient method without the agonizing pain," Carnegie Mellon University, Tech. Rep. CMU-CS-94-125, August 1994.
[28] R. Barrett, M. Berry, T. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, and H. V. der Vorst, *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods*, 2nd ed. Philadelphia, PA: SIAM, 1994.
[29] Y. Liu and S. Thrun, "Results for outdoor-SLAM using sparse extended information filters," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Taipei, Taiwan, May 2003, pp. 1227–1233.
[30] R. Eustice, H. Singh, J. Leonard, M. Walter, and R. Ballard, "Visually navigating the RMS Titanic with SLAM information filters," in *Proceedings of Robotics: Science and Systems (RSS)*, Cambridge, MA, June 2005, pp. 57–64.
[31] Y. Bar-Shalom, X. Rong Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*. New York: John Wiley & Sons, Inc., 2001.
[32] 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 *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*. Acapulco, Mexico: IJCAI, 2003, pp. 1151–1156.