Maximally Informative Statistics for Localization and Mapping

Matthew C. Deans
The Robotics Institute
Carnegie Mellon University
Pittsburgh, PA, 15217, USA
mdeans@cs.cmu.edu

Abstract

This paper presents an algorithm for localization and mapping for a mobile robot using monocular vision and odometry as its means of sensing. The approach uses the Variable State Dimension filtering (VSDF) framework to combine aspects of Extended Kalman filtering and nonlinear batch optimization. This paper describes two primary improvements to the VSDF. The first is to use an interpolation scheme based on Gaussian quadrature to linearize measurements rather than relying on analytic Jacobians. The second is to replace the inverse covariance matrix in the VSDF with its Cholesky factor to improve the computational complexity. Results of applying the filter to the problem of localization and mapping with omnidirectional vision are presented.

I. Introduction

It may be required for a robot to enter an unknown environment and to concurrently explore the area and produce a map while maintaining an accurate estimate of its position. If the robot were to have an *a priori* map, then localization with respect to the known map would be a relatively easy task. Alternatively, if the robot were to have a precise, externally referenced position estimate, then mapping would be a relatively easy task. However, problems in which the robot has no *a priori* map and no external position reference are particularly challenging. Such scenarios may arise for underwater robots, mining vehicles, planetary surfaces, or anywhere that maps are not available. This problem has been referred to as *concurrent localization and mapping* (CLM) and *simultaneous localization and mapping* (SLAM). We will use the latter in this paper.

In the work presented here, we model the robot environment as a 2-D planar world, so that the rover pose at time i is the 3-dof parameter vector $\mathbf{m_i}$ including position on the 2-D plane and orientation. Landmarks are assumed to be point features and the position of landmark j is the 2-dof parameter vector $\mathbf{x_j}$. The means of sensing considered in this work are an odometry measurement

$$\mathbf{d_i} = f(\mathbf{m_i}, \mathbf{m_{i-1}}) + \mathbf{w_i} \tag{1}$$

which measures the change in vehicle pose from $i-1$ to i, and a bearing measurement

$$b_k = g(\mathbf{m_{i(k)}}, \mathbf{x_{j(k)}}) + v_k \tag{2}$$

which measures the bearing from the rover position $\mathbf{m_{i(k)}}$ to a landmark at position $\mathbf{x_{j(k)}}$, where $i(k)$ and $j(k)$ indicate which pose and landmark correspond to bearing measurement k. Both $\mathbf{w_i}$ and v_k are modelled as i.i.d. Gaussian noise processes. For notational convenience, we denote the parameter vector including all unknowns as $\theta = \{\mathbf{m_1}, \mathbf{m_2}, \ldots, \mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_N}\}$ and the measurement vector $\mathbf{z} = \{d_1, d_2, \ldots, d_i, b_1, b_2, \ldots, b_k\}$. The generating function for all measurements as a function of all parameters is

$$\mathbf{z} = \mathbf{h}(\theta) + \mathbf{v}, \tag{3}$$

where $\mathbf{v} \sim \mathcal{N}(0, \mathbf{R})$ is a normally distributed random variable and following the assumption of i.i.d. noise above, $\mathbf{R}$ is diagonal. The generating function for a single odometry or bearing measurement will be denoted $\mathbf{z_k} = h_k(\theta) + v_k$ where, with a slight change of notation from above, $\mathbf{z_k}$ can be either type of measurement.

The bearing only sensor model we use here is motivated by the use of monocular vision, which is fairly cheap, small, robust, low weight and low power compared to active range-bearing sensors. Odometry is known to provide poor egomotion data but the goal is to see what can be done with these two simple sensing modalities alone. The incorporation of inertial measurement, external position references, and range sensors can only improve the end results. Furthermore, other models of landmarks and map parameterizations are possible but are not considered in this work. Finally, the method described here is extendable to the full 3-D problem, although admittedly the problem is more complex and may require more sophisticated parameterization.

II. Previous Work

There are two primary sources of literature related to the problem considered here. Bearings-only localization and mapping is similar to the SLAM problem in robotics and to the Structure from Motion (SFM) problem in computer vision.

Most of the SLAM literature in robotics explores the problem of sensor fusion for onboard egomotion sensing and range-bearing sensors such as radar, sonar, and lidar. Approaches such as iterated closest point (ICP) [1], Expectation-Maximization [2], and correlation [3] have been explored for the task. The predominant body of SLAM work uses Extended Kalman filtering (EKF) based approaches [4], [5], [6], [7] or related approaches such as Unscented Filter [8] or Covariance Intersection [9].

The photogrammetry and computer vision literature contain a significant amount of work related to the structure from motion (SfM) problem, in which monocular images alone are used to reconstruct the scene and recover the camera motion. Among the popular approaches are factorization [10], sequential multiframe geometric constraints [11], and nonlinear bundle adjustment [12].

The Variable State Dimension filter[13] is a combination of Extended Kalman filtering and nonlinear optimization. The filter was developed to be a recursive algorithm for Structure from Motion, and it has some of the characteristics of bundle adjustment and Kalman smoothing. The VSDF provides the foundation for the work in this paper.

III. The VSDF Algorithm

The variable state dimension filter (VSDF) combines aspects of the EKF with aspects of Gauss-Newton nonlinear optimization[14]. Since $\mathbf{z_k}$ is a Gaussian random variable with mean $\mathbf{h_k}(\theta)$ and variance $\mathbf{R_k}$, we can write the likelihood for $\mathbf{z}$ given θ

$$p(\mathbf{z}|\theta) \propto e^{-\sum_k (\mathbf{z_k}-\mathbf{h_k}(\theta))^T \mathbf{R_k}^{-1} (\mathbf{z_k}-\mathbf{h_k}(\theta))} \tag{4}$$

Gauss-Newton optimization searches for the parameter which minimizes the negative log of the likelihood

$$\begin{aligned} e &= -log(p(\mathbf{z}|\theta)) \\ &= \sum_k (\mathbf{z_k}-\mathbf{h_k}(\theta))^T \mathbf{R_k}^{-1} (\mathbf{z_k}-\mathbf{h_k}(\theta)) \end{aligned} \tag{5}$$

In order to minimize this cost function, the algorithm starts with an estimate of the state vector θ_0 and computes

$$\mathbf{a} = \sum_k -\mathbf{H_k}^T \mathbf{R_k}^{-1} (\mathbf{z_k}-\mathbf{h_k}(\theta)) \tag{6}$$

$$\mathbf{A} = \sum_k \mathbf{H_k}^T \mathbf{R_k}^{-1} \mathbf{H_k} \tag{7}$$

where $\mathbf{H_k} = \left.\frac{\partial \mathbf{h_k}}{\partial \theta}\right|_{\theta_0}$ is the measurement Jacobian, $\mathbf{a} = \nabla e$ is the gradient of (5) and $\mathbf{A} \approx \nabla^2 e$ is an approximation to the Hessian[14]. The algorithm computes an update to the state estimate by solving the linear system

$$\mathbf{A}\delta = \mathbf{a} \tag{8}$$

and updating the parameter vector

$$\theta \leftarrow \theta - \delta \tag{9}$$

Equations (6) through (9) are iterated to convergence, . Solutions found using Gauss-Newton are optimal in a least squares sense, which is also maximum likelihood for Gaussian noise. However, the vector θ contains the entire map and the entire vehicle trajectory, which makes Gauss-Newton slow for large datasets.

The VSDF provides a method for linearizing measurements, incorporating them into a Gaussian "prior". The filter equations may be derived by linearizing terms on the right hand side of (6) and (7). Suppose we wish to replace the term involving $\mathbf{z_k}$ in (5). We can compute a linear approximation to $\mathbf{h_k}()$

$$\mathbf{h_k}(\theta) \approx \mathbf{h_k}(\theta_0) + \mathbf{H_k}(\theta - \theta_0) \tag{10}$$

and in order to minimize the new cost function we simply replace the corresponding term in (6) and (7) with the same linearization.

In the VSDF, terms are replaced with a linearization of the form

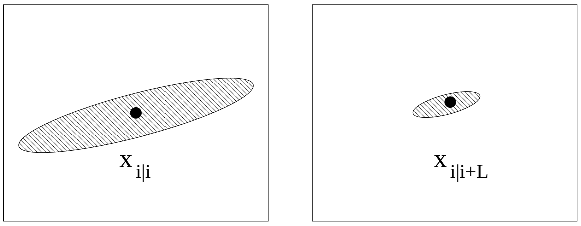
$$\begin{aligned} &(\theta-\theta_0)^T \mathbf{A_k}(\theta-\theta_0) \approx \\ &(\mathbf{z_k} - \mathbf{H_k}(\theta-\theta_0))^T R_k^{-1} (\mathbf{z_k} - \mathbf{H_k}(\theta-\theta_0)) \end{aligned} \tag{11}$$

where $\mathbf{A_k} = \mathbf{H_k}^T R_k^{-1} \mathbf{H_k}$ is the contribution of measurement k to the Hessian, and

$$\mathbf{a_k} = \mathbf{H_k}^T R_k^{-1} (\mathbf{z_k} - \mathbf{h_k}(\theta_0)) \tag{12}$$

is the constant contribution of measurement k to the gradient (6). In the original VSDF the linearization $\mathbf{H_i}$ is again taken to be the Jacobian of the measurement function evaluated at the state estimate.

This is similar to the EKF, except that the EKF linearizes each measurement immediately upon incorporation into the state estimate. The VSDF opens the possibility of linearizing the term at some later time[13]. The advantage in linearizing the measurement later is that the point of expansion for the linearization is extimated using more data, and therefore



(a) (b)

Fig. 1. Advantage of leaving nonlinear measurements in the filter (a) The estimate of the state x at time i given information up to time i leaves a large region of uncertainty (shaded). (b) After processing later observations, the VSDF has an estimate with lower variance, increasing the chances that our linearization will be a good one.

has smaller variance. We can expect the linearization to occur at a more accurately estimated point, as shown in Figure 1.

The Jacobian and Hessian can now be expressed as a combination of terms from the linearized measurements and the nonlinear measurements

$$\begin{aligned} \mathbf{a} &= \mathbf{a_0} + \mathbf{A_0}(\theta - \theta_0) + \sum \mathbf{H_k}^T \mathbf{R_k}^{-1}(\mathbf{z_k} - \mathbf{h_k}(\theta)) \\ \mathbf{A} &= \mathbf{A_0} + \sum \mathbf{H_k}^T \mathbf{R_k}^{-1} \mathbf{H_k} \end{aligned} \tag{13}$$

Once measurements are linearized, there are parts of the state space that will no longer be a part of new measurements coming in (like old robot poses). Those subspaces can be eliminated from the filter. If we partition the state vector into $\theta = [{\theta_1}^T {\theta_2}^T]^T$ and the gradient $\mathbf{a} = [\mathbf{a_1}^T \mathbf{a_2}^T]^T$ and

$$\mathbf{A} = \begin{pmatrix} \mathbf{A_{11}} & \mathbf{A_{12}} \\ \mathbf{A_{21}} & \mathbf{A_{22}} \end{pmatrix} \tag{14}$$

then we can eliminate the first subspace by updating the parameter vector, gradient, and Hessian as follows,

$$\begin{aligned} \theta &\leftarrow \theta_2, \quad \mathbf{a} \leftarrow \mathbf{a_2} \\ \mathbf{A} &\leftarrow \mathbf{A_{22}} - \mathbf{A_{21}} {\mathbf{A_{11}}}^{-1} \mathbf{A_{12}} \end{aligned} \tag{15}$$

See [13] for details. The filter continues to incorporate new measurements as they become available, and linearizes them at some later time.

IV. Maximally Informative Statistics

Good heuristics for how and when to linearize measurements can come from the notion of *maximally informative statistics*. A *statistic* $\tau = t(z)$ is some function of a data set which may be a reduction such as moment computation (mean, variance), finding the maximum or minimum of the data, or estimation of parameters for some parametric model. Typically the goal is to compute a statistic which allows inference to be made without reconsidering the entire data set.

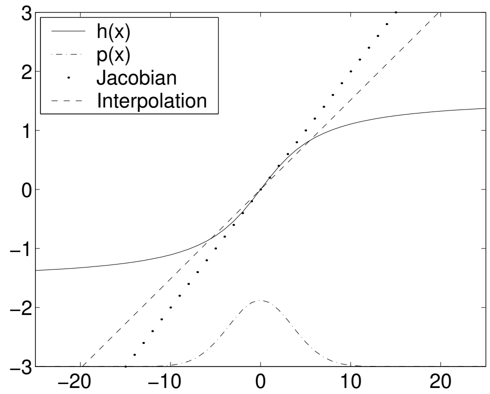


Fig. 2. Recursive filtering requires linearization of the nonlinear measurement function h(x) at a point. The Jacobian provides the instantaneous slope at a point estimate, but the maximally informative statistics principle indicates that an interpolation can fit the function better by minimizing expected square error under the probability distribution p(x)

A *sufficient statistic* is a statistic that can be used to make inferences about the data just as effectively as the original data itself. More formally, a statistic is sufficient if the distribution for an estimator under the statistic is the same as it is under the original data,

$$p(\theta|\tau) = p(\theta|\mathbf{z}) \tag{16}$$

An example of a sufficient statistic is the sample mean and sample variance for normally distributed data.

A *maximally informative statistic* is a generalization of sufficiency. There is not always a sufficient statistic, but we can always find some statistic $\tau \in T$ which satisfies

$$\tau = Argmin_{t \in T} \; D(p(\theta|t) || p(\theta|\mathbf{z})) \tag{17}$$

where $D(\cdot||\cdot)$ is the Kullback-Liebler divergence, or relative entropy, and T is some class of statistic. Here we will let T be the set of all mean vectors and covariance matrices. When a sufficient statistic exists within T, the sufficient statistic is the maximally informative one and the KL divergence becomes zero.

Under the assumption of normally distributed additive observation noise, we can write the true posterior for model parameters given data as

$$p(\theta|\mathbf{z}) \propto e^{-(\mathbf{z}-\mathbf{h}(\theta))^T R^{-1}(\mathbf{z}-\mathbf{h}(\theta))} \tag{18}$$

and the Gaussian approximation in state space can be written

$$p(\theta|\theta_0, \mathbf{C_0}) \propto e^{-(\theta-\theta_0)^T {\mathbf{C_0}}^{-1}(\theta-\theta_0)} \tag{19}$$

If we assume that $\mathbf{C_0}$ can be computed as $\mathbf{C_0}^{-1} = \mathbf{H}^T R^{-1}\mathbf{H}$ then our job becomes one of finding the $\mathbf{H}$ such that the KL divergence

$$D = E_p \left(\log \left(\frac{p(\theta|\mathbf{z})}{p(\theta|\theta_0, \mathbf{C_0})} \right) \right) \tag{20}$$

is minimized, which after manipulation becomes

$$D = \tag{21}$$
$$E_p \left[(\mathbf{h}(\theta) - \mathbf{H}(\theta - \theta_0))^T R^{-1} (\mathbf{h}(\theta) - \mathbf{H}(\theta - \theta_0)) \right]$$

where E_p denotes expectation under distribution p. Since R is positive definite, D is bounded below by zero and is minimized when the linearization $\mathbf{H}(\theta - \theta_0)$ is most accurate over the region of high probability within $p(\theta|\mathbf{z})$. Typically the means for computing the linearization $\mathbf{H}$ in Extended Kalman filtering is to compute the Jacobian[15], which is only accurate for infinitesimal departures $\theta - \theta_0$. Alternatives have been reported elsewhere, the DDF filter [16] replaces the Jacobian with a central divided difference, and the Unscented filter [8] uses a deterministic sampling scheme to compute the posterior covariance directly. Each of these approaches finds a linearization that is more accurate over the interval than the Jacobian computation, and performance increases over the EKF have been reported for both. In our work we compute a locally weighted linear interpolation of the function $\mathbf{h_k}(\cdot)$ using Gaussian quadrature.

Gaussian quadrature is a means of numerically computing an integral using a small number of carefully chosen points and associated weights[14]. Deterministic rules for computing the samples and weights exist. There are specific rules for computing the samples to use for evaluating expectations as in (21) depending on the form of the distribution over which the expectation is computed. Because the envelope function $p(\theta|\mathbf{z})$ above is approximated by a Gaussian, we compute the Gauss-Hermite[14] quadrature points y_i and associated weights w_i for the dimensions corresponding to the inputs to $h_k()$, namely the robot pose and landmark position related to that measurement.

Once the quadrature points are computed, we fit the linear system

$$\begin{pmatrix} \sqrt{w_1} h_k(y_1) \\ \sqrt{w_2} h_k(y_2) \\ \vdots \\ \sqrt{w_N} h_k(y_N) \end{pmatrix} = \begin{pmatrix} \sqrt{w_1}\mathbf{x_1}^T \\ \sqrt{w_2}\mathbf{x_2}^T \\ \vdots \\ \sqrt{w_N}\mathbf{x_N}^T \end{pmatrix} \mathbf{H_k} \tag{22}$$

using least squares. The resulting coefficient matrix $\mathbf{H_k}$ is used to update the Hessian matrix $\mathbf{A}$ in the VSDF.

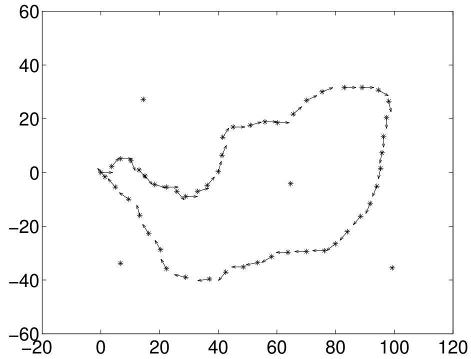


Fig. 3. Ground truth for an example problem with four landmarks and a trajectory consisting of 50 robot poses.

V. Using Cholesky factors

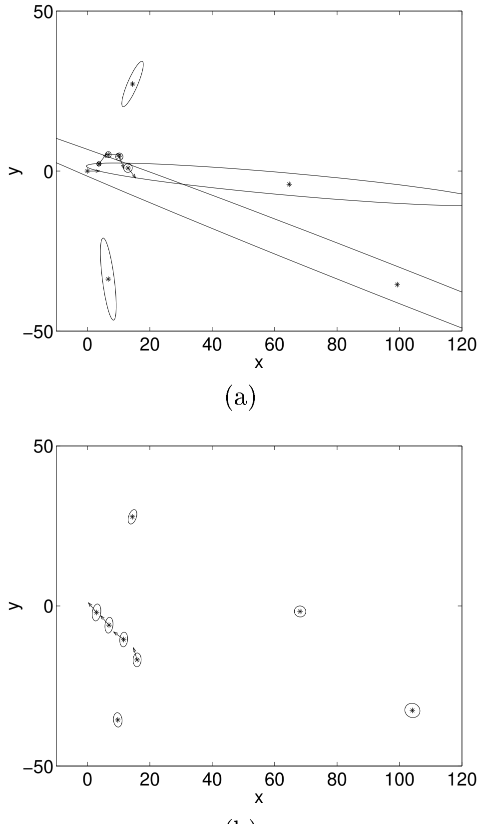
A reduction in computational complexity can be realized by working with the Cholesky factorization of the Hessian matrix rather than the Hessian itself. Cholesky factorization is a common means of solving a system of linear equations $Ax = b$ when the coefficient matrix A is symmetric and positive definite. The cholesky factorization $\mathbf{SS}^T = \mathbf{A}$ is first computed, then the two triangular systems

$$\begin{aligned} \mathbf{Sy} &= \mathbf{b} \\ \mathbf{S}^T\mathbf{x} &= \mathbf{y} \end{aligned} \tag{23}$$

are solved. The Cholesky decomposition of a dense $N \times N$ matrix can be computed in $O(N^3)$. The solution to the two triangular linear systems is $O(N^2)$.

If rather than storing and manipulating the full Hessian matrix $\mathbf{A}$ we can store and manipulate its Cholesky factor S, then the factorization step can be avoided and the solution to (8) can be computed with $O(N^2)$ backsubstitutions alone. The only remaining problem is to determine how to propagate the Cholesky factor from step to step in the filter.

There exist algorithms for performing the *update* and *downdate* of a Cholesky factor, where update is defined as the addition of a symmetric outer product $\mathbf{A}' = \mathbf{A} + v^T\sigma v$ and downdate is the subtraction of a symmetric outer product $\mathbf{A}' = \mathbf{A} - v^T\sigma v$. For a rank-1 update or downdate these algorithms require $O(N^2)$ computation, so a rank-k update can be done in $O(kN^2)$. If we already have the Cholesky factor for the prior Hessian $\mathbf{A_0}$, then the step which combines the prior and the likelihood is given by (13) which can be computed as a Cholesky update, and the marginalization of state dimensions to be removed from the filter is given in (15) which can be computed as a Choleksy downdate. Since we can perform



(a)

(b)

Fig. 4. Result from one trial on the example problem. (a) Initial state estimate using measurements from first five robot poses. (b) Final state estimate, including map and last five robot poses. This is the estimate after processing all information.

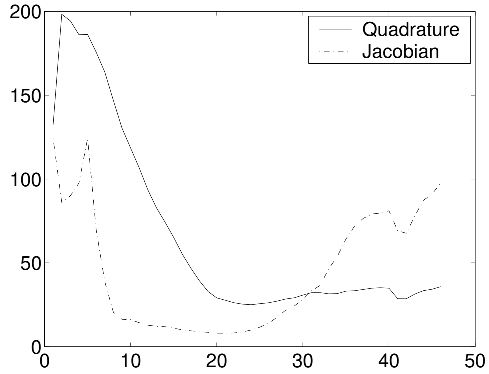


Fig. 5. Map reconstruction error over time, ensemble average over 100 trials. Solid line shows result of linearization using quadrature based interpolation, dash-dot line shows result of linearization using Jacobian.

Cholesky updates and downdates for adding and removing measurements and states in the filter, and use the Cholesky factors in the optimization step to solve the normal equations, we can do away with the full covariance matrix $\mathbf{A}$ and only maintain and use its Cholesky decomposition $\mathbf{S}$. This technique has been used to modify Kalman filters for parameter estimation problems[15].

The insertion and removal of measurements and states in the filter proceeds as before except that Cholesky updates and downdates replace the operations on $\mathbf{A_0}$.

VI. Experimental Results

Figure 3 shows an example problem with four landmarks and 50 robot poses in the trajectory. The small problem size is chosen to make the figures more legible. We ran 100 Monte Carlo trials of the filter algorithm by generating synthetic data $\mathbf{z}^{(\mathbf{r})}$ for $r = 1 \ldots 100$ using the generative model described in the introduction with Gaussian additive noise. For each trial the algorithm was used to produce a state estimate once using the Jacobian linearization $H = \frac{\partial h}{\partial \theta}$ and once using quadrature based interpolation. In this example the filter retains nonlinear measurements for 5 time steps before linearizing. Figure 4 shows an initial and final state estimate for one run of the filter. After filtering each data set, the squared error between the true map and the final estimated map was computed.

Figure 5 shows the map reconstruction error over time, compared with ground truth and averaged over the 100 runs. Some variation is expected from one run to the next, so we cannot expect the quadrature based method to perform strictly better than the Jacobian method on a per-trial basis, but over the course of many runs the quadrature method shows much better performance in terms of the accuracy of the final estimate. What is interesting to note in Figure 5 is that the Jacobian based method seems to converge to a solution with smaller reconstruction error early but then diverges. This may be because as the filter estimate changes the linearization as computed at old state estimates becomes less accurate and effects are not seen until the state estimate moves sufficiently far from where it was linearized.

Figure 6 shows the convergence of the x and y coordinate of the landmark that appears in the lower left of Figure 3, which is the most difficult to estimate given the problem geometry. The evolution of the estimated position is shown for ten runs along with the estimated variance, and convergence to the true solution is seen.

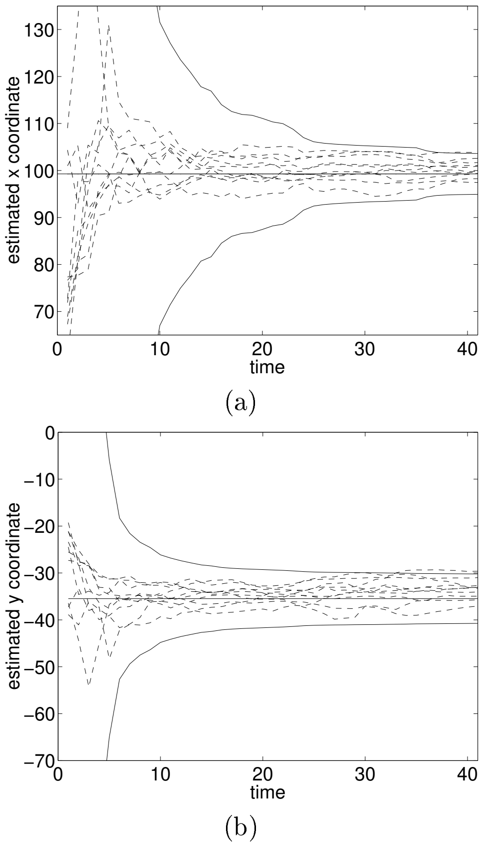


Fig. 6. Convergence of the lower right landmark, which is the one that is least certain in early estimates. (a) shows the convergence of the x coordinate for landmark #3. (b) shows the convergence of the y coordinate for landmark #3. Values for ten runs are plotted as dashed lines. Also plotted are the true value and the 95% confidence (3σ) region centered on the true value in solid lines.

VII. Summary and Conclusions

In this paper we investigate the implication of maximally informative statistics on linearization for recursive filtering. The maxinfo criterion is shown to be equivalent to the expected squared error between the true nonlinear model function and its linearization under the posterior. This metric is physically meaningful and very intuitive. It is used to determine a means of linearizing the measurements in the VSDF which is shown to outperform the analytic Jacobian for the problem considered here. The linearization itself is performed using Gaussian quadrature. We are investigating using the linearization error to decide when to linearize and when to leave measurements in the filter, although at some point computational resources may require linearization even if only a poor approximation is available.

We have also introduced a square root formulation of the VSDF which cuts the computational complexity from $O(N^2(L+N))$ to $O(N(L+N))$, where N is the number of landmarks in the map and L is the time lag for the VSDF. This is a significant for large maps since N could be in the hundreds or thousands. The algorithmic improvement only changes the way in which the sufficient statistics are represented and used in optimization, and does not affect the error performance.

References

[1] F. Lu and E. Milios. Globally consistent range scan alignment for environment mapping. *Autonomous Robots*, 4(4):333–349, 1997.

[2] W. Burgard, D. Fox, H. Jans, C. Matenar, and S. Thrun. Sonar-based mapping with mobile robots using EM. In *Proc. of the International Conference on Machine Learning*, 1999.

[3] J.-S. Gutmann and K. Konolige. Incremental mapping of large cyclic environments. In *Proc. IEEE Symp. CIRA*, pages 318–325, 1999.

[4] Randall Smith, Matthew Self, and Peter Cheeseman. *Estimating Uncertain Spatial Relationships in Robotics*, chapter 3, pages 167–193. Springer-Verlag, 1990.

[5] P. Moutarlier and R. Chatila. Stochastic multisensory data fusion for mobile robot location and environment modelling. In *5th Int. Symposium on Robotics Research*, 1989.

[6] H. Feder, J. Leonard, and C. Smith. Adaptive mobile robot navigation and mapping. *International Journal of Robotics Research*, 18(7):650–668, July 1999.

[7] M.W.M.G. Dissanayake and et al. A solution to the simultaneous localization and map building (slam) problem. Technical Report ACFR-TR-01-99, Australian Centre for Field Robotics, University of Sydney, Australia, 1999.

[8] J. K. Uhlmann, S. J. Julier, and M. Csorba. Nondivergent simultaneous map-building and localization using covariance intersection. In *SPIE Proceedings: Navigation and Control Technologies for Unmanned Systems II*, volume 3087, pages 2–11, 1997.

[9] S. Julier, J. Uhlmann, and H. Durrant-Whyte. A new approach for filtering nonlinear systems. In *Proceedings of the 1995 American Controls Conference*, pages 1628–1632, 1995.

[10] C. Tomasi and T. Kanade. Shape and motion from image streams: a factorization method. Technical Report CMU-CS-92-104, Carnegie Mellon University, 1992.

[11] A. Shashua. Trilinear tensor: The fundamental construct of multiple-view geometry and its applications. *Lecture Notes in Computer Science*, 1315:190–??, 1997.

[12] B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon. Bundle adjustment - a modern synthesis. In *To appear in Vision Algorithms: Theory & Practice*. Springer-Verlag, 2000.

[13] Philip F. McLauchlan. The variable state dimension filter applied to surface-based structure from motion. Technical Report VSSP-TR-4/99, University of Surrey, Guildford GU2 5XH, 1999.

[14] W. Press, S. Teukolsky, W. Vetterling, and B. Flannery. *Numerical Recipes in C*. Cambridge University Press, 1988.

[15] P. Maybeck. *Stochastic Models, Estimation, and Control*. Academic Press, Inc., 1979.

[16] M Nørgaard, N. K. Poulsen, and O. Ravn. Advances in derivative-free state estimation for nonlinear systems. Technical Report IMM-REP-1998-15, Technical University of Denmark, 2800 Lyngby, Denmark, 2000.