Robotica (2012) volume 30, pp. 245–256. © Cambridge University Press 2011
doi:10.1017/S0263574711000543

A two-tier map representation for compact-stereo-vision-based SLAM

D. C. Herath†*, S. Kodagoda‡ and G. Dissanayake‡

†*Thinking Head Project, MARCS Auditory Laboratories, University of Western Sydney, Australia.*
‡*ARC Centre of Excellence for Autonomous Systems, University of Technology, Sydney, Australia.*
E-mails: {*s.kodagoda, g.dissanayake*}*@cas.edu.au*

(Received in Final Form: May 11, 2011; accepted May 9, 2011. First published online: June 14, 2011)

SUMMARY
Vision sensors are increasingly being used in the implementation of Simultaneous Localization and Mapping (SLAM). Even though the mathematical framework of SLAM is well understood, considerable issues remain to be resolved when a particular sensing modality is considered. For instance, the observation model of a small baseline stereo camera is known to be highly nonlinear. As a consequence, state estimations obtained from standard recursive estimators, such as the Extended Kalman Filter, tend to be inconsistent. Further, vision-based approaches are plagued with high feature densities, and the consequent requisite of maintaining large feature databases for loop closure and data association. This paper proposes a two-tier solution for resolving these issues, inspired by the mechanics of human navigation. The proposed two-tier solution addresses the consistency issue by formulating the SLAM problem as a nonlinear batch optimization and presents a novel method for feature management through a two-tier map representation. Simulations and experiments are carried out in an office-like environment to validate the performance of the algorithm.

KEYWORDS: SLAM; Computer vision; Navigation; Robot localization; Pose estimation and registration.

1. Introduction

THE use of passive, inexpensive, and compact sensors for robotic navigation is an appealing area of research. Vision sensors qualify exceedingly well with the above criteria. In addition, vision sensors produce large amounts of data that could potentially be used in further augmenting the navigation capabilities of a mobile robot. They also have the versatility to be implemented in widely varied application domains. These desirable attributes have motivated investigations into the use of vision sensors in the context of Simultaneous Localization and Mapping (SLAM). Various camera configurations have been studied in the literature with varying degrees of success. On one end of the spectrum, single camera bearing-only[1–3] methods that have vigorously been studied with particular attention being paid to the development of initialization algorithms[1–4] that could countenance the absence of range information in the observations. On the other end of the spectrum, multiple camera systems have been studied. For instance, refs. [5, 6] use large baseline stereo cameras for SLAM, whereas ref. [7] uses a configuration with more than two cameras for refined observations.

However, there are only few instances reported on a concerted study on the use of small baseline (<0.1 m) stereo cameras as a viable sensor for SLAM. The motivation for pursuing a small baseline stereo sensor in the application of SLAM is mainly twofold. First, a camera with a small footprint is ideal for compact robotic systems that require higher degree of agility and speed. An example application of such requisites is the "urban search and rescue."[8] Second, additional information provided by a second camera could improve the tractability of SLAM solutions compared to bearing-only SLAM with a single camera. In this regard, low cost small baseline stereo vision provides an alternative to conventional sensors, providing balance amongst size, performance, and cost.

One of the main issues that arise in this particular sensor configuration is the highly nonlinear nature of the observation model. In general, the bias in the projective mapping has been discussed by several works, including refs. [9, 10]. On the other hand, Bailey *et al.*,[11] Huang and Dissanayake,[12] and Julier and Uhlman[13] have shown that the SLAM estimator outputs become inconsistent due to the nonlinear nature of the problem. Several approaches have been proposed in tackling these issues, such as the Sigma point filter or the Unscented Kalman Filter (UKF),[14–16] and techniques that deviate altogether from standard recursive techniques by formulating it as a nonlinear batch optimization problem.[17,18]

Another vision related issue is the temporal growth in the size of the feature map. It is mainly due to the high information content present in each image frame. This issue is further aggravated by limited field of view (FOV) of the camera lens and the operating environmental constraints, such as narrow corridors, where the visibility is limited, resulting in less repetitive observations to unique features. A natural technique adopted in managing such growth is to remove portions of the map containing the least informative features.[19] However, such techniques carry the risk of losing valuable information, especially during a potential loop closure.

Multiple map techniques have evolved as a means of managing the complexity of large-scale SLAM by way of

* Corresponding author. E-mail: damithc@gmail.com

https://doi.org/10.1017/S0263574711000543 Published online by Cambridge University Press

building manageable independent local metric maps, which are then represented either as a single global metric map or as a form of hierarchical relationships map.[20–23] For instance, in ref. [20], a technique called the Constrained Local Submap Filter was used to create independent, local submaps of features that are fused into a Global Map (GM) of the environment periodically to improve the computational complexity associated with managing the GM. In the Atlas framework,[21] a hybrid approach is taken where submaps are generated with their own coordinate frames and a topological graph representing the relationship between independent frames. In this framework, an offline processing is necessary to generate the global metric representation. Hierarchical SLAM[22] also takes a similar approach to Atlas framework but maintains an estimation of the relative locations between the maps in a relative stochastic map. There are other topological representations of metric maps, such as the unified Bayesian framework,[24] the multiresolution representations,[25] and T-SLAM,[26] where local metric maps are associated with topological places and are registered at the topological level.

In all of these multimap schemes, the key technique has been to combine local information in a global representation to manage the computational complexities in large-scale navigation. In our approach, the technique is fundamentally in the opposite direction, where local information is separated and marginalized from the global representation. The two-tier scheme presented in this paper relies on features with different characteristics for the individual maps and the emphasis is on disassociation of large number of map features from the global representation as opposed to map joining. The technique, which has its inspiration in human cognition, could find analogous representations postulated in recent human cognition research.[27] In the proposed scheme, the notion of Points of Interest (PoIs) supersede the notion of a GM.

This paper follows the authors' previous work[28,29] on the use of small baseline stereo cameras in SLAM, where several pertinent issues relating to the use of the particular sensor have been identified. This paper extends that work with further experimental results and the formalization of the two-tier map framework. The main contribution of this paper is the development of a novel two-tier SLAM framework, which addresses the issues caused by nonlinearity in the observation model and the growth of feature density. Improvements gained through the new framework are further illustrated through consistency analysis that shows the limitation of generally applied EKF SLAM solutions with respect to the sensor used.

The rest of the paper is structured as follows. Section 2 provides a detailed explanation of the issues pertinent to small baseline cameras. Section 3 synthesizes the two-tier solution in a nonlinear optimization framework and addresses the issues detailed in Section 2. Simulation and experimental results are presented in Section 4. Section 5 concludes the paper.

2. Issues with Small Baseline Stereo Cameras

2.1. Nonlinearity of the observation model

The projective mapping of stereo observations from the image plane to the Cartesian coordinate frame (camera centric coordinates) is related by the nonlinear equation given by

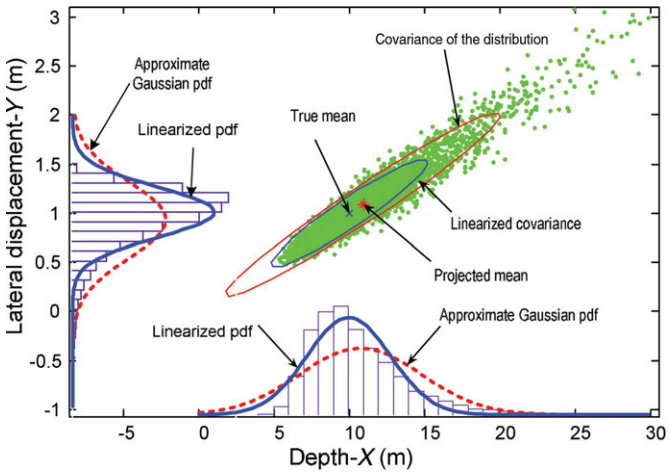


Fig. 1. (Colour online) Errors in projective mapping. Note that only two dimensions are used for clarity.

$$\mathbf{z}_c = F(u, v, d, B, f), \tag{1}$$

where $(u, v)$and (d) are image coordinates and disparity, respectively. B is the baseline and f is the focal length of the cameras. $\mathbf{z}_c = (x_c, y_c, z_c)$ is the observation vector in camera centric coordinates. (u, v, d) defines the raw observations in stereo vision systems and could be safely assumed as Gaussian distributions.[28]

When observing far features, a bias in the depth is introduced through the triangulation and the linearized error propagation.[10] The phenomenon can be explained using the following Monte Carlo simulation. Let's consider a situation where a stereo sensor ($B = 0.088$ m and $f = 150$ pixels) observes a single feature at a depth of 10 m and a lateral displacement of 1 m. Assume a large number of observations with respect to that feature in the image *(u,d)* drawn from Gaussian probability density functions *(pdfs)* with standard deviations of $\sigma_u = 1.34$ and $\sigma_d = 0.65$ pixels, respectively. These observations are then projected into the camera coordinates through Eq. (1). Figure 1 shows the projected observations in the Camera coordinates, which appear to be skewed with a tail toward top-right side of the graph. It also depicts the covariance of the projected observations estimated through linearized error propagation, which appears to be optimistic compared to the true error covariance. The true mean of the *pdf* is smaller than the mean obtained through linearization, indicating the presence of linearization errors.

The amount of linearization bias is in fact related to the baseline/depth ratio of the camera itself. Hence, some authors suggest an empirical minimum to be used when implementing such cameras in SLAM. For instance, Jung[5] suggests the ratio *(baseline/depth)* $>$ (1/30). However, limiting the effective range of the sensor, given a constant baseline adversely affects the performance of SLAM. Specially, features need to be observed for prolong periods of time, from a distance to improve the heading errors.

It is possible to observe the effects of changing baseline on the consistency of a filter through simulation. Using multiple simulations, consistency of the estimates could

https://doi.org/10.1017/S0263574711000543 Published online by Cambridge University Press

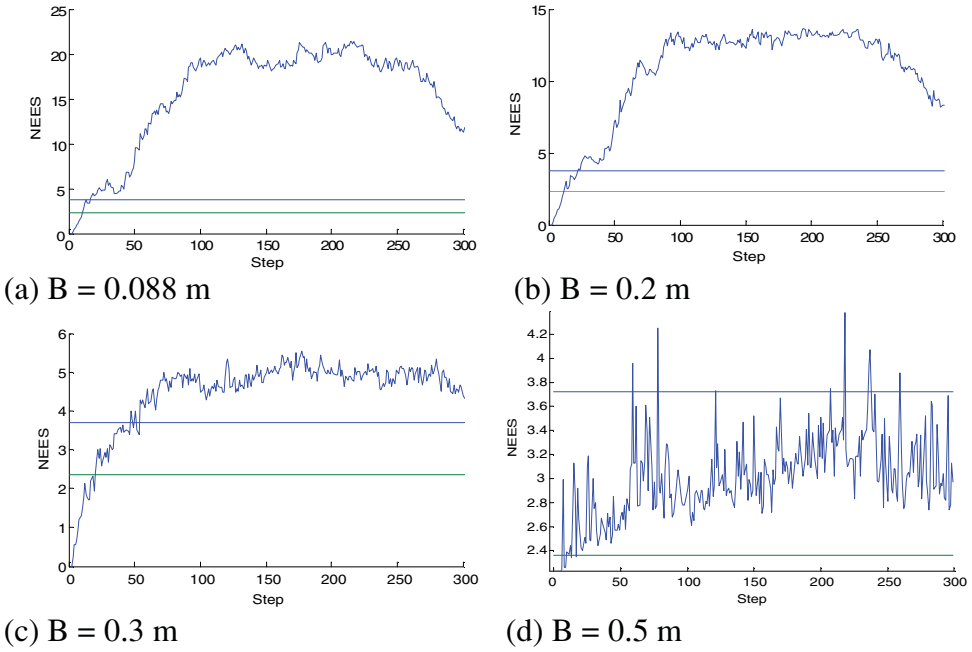


Fig. 2. (Colour online) Effects of baseline on the consistency of an UKF.

be analyzed through a statistical measure, such as the Normalized Estimation Error Squared (NEES),[30] which is defined by

$$\varepsilon(k) = (\mathbf{x}(k) - \hat{\mathbf{x}}(k \mid k))^{\mathrm{T}} \mathbf{P}(k \mid k)^{-1} (\mathbf{x}(k) - \hat{\mathbf{x}}(k \mid k)), \quad (2)$$

where $\mathbf{x}(k)$ and $\hat{\mathbf{x}}(k \mid k)$ are the state and the estimate of the state at time step k, respectively. $\mathbf{P}(k|k)$ is the state covariance matrix at time step k. Under the hypothesis H_0 that filter is consistent and the assumption that the process models are linear and the noise sources are distributed according to Gaussian models, $\varepsilon(k)$ is chi-square distributed with n_x degrees of freedom. Then, the test is whether

$$E[\varepsilon(k)] = n_x \quad (3)$$

could be accepted, where n_x is the dimension of the state. For a set of N independent simulations, the average NEES could be estimated as

$$\bar{\varepsilon}(k) = \frac{1}{N} \sum_{i=1}^{N} \varepsilon_i(k). \quad (4)$$

Then, under H_0, $N\bar{\varepsilon}(k)$ will have a chi-square distribution with Nn_x degrees of freedom. The hypothesis that the state errors are consistent with the filter estimated covariance is accepted if

$$\bar{\varepsilon}(k) \in [r_1, r_2], \quad (5)$$

where $[r_1, r_2]$ is the acceptance interval. For this analysis, a simulated robot is driven around a small loop observing features using a stereo sensor. Observation data are generated for four different baselines. These data are then processed through a standard UKF. Each simulation experiment consists of 50 independent trials. Thus, for an estimate of the robot pose consisting of three dimensions, a value of $r_2 = 3.72$ represents the upper 95% confidence bound and $r_2 = 2.36$ is the lower bound. When the estimated value is higher than the upper bound, the filter is said to be optimistic, when it is lower than the lower bound the filter is behaving conservatively. Figure 2 shows the average NEES results for the four different baselines. As seen from the figures, the estimates are inconsistent toward small baselines and improve as the baseline length increases.

2.2. Large feature density

Feature-based SLAM requires the identification of interest points in the environment. Historically, point detectors, such as the Harris corner,[31] have been used in vision-based SLAM implementations for this purpose. Recently, more robust algorithms, such as the SURF detector,[32] have considerably advanced the state of the art. However, a major drawback in such detectors is that they tend to generate large number of potential features in a single image frame. For instance, a typical image (Fig. 3) taken inside our lab normally generates features in excess of 90 using the SURF algorithm.

This issue is addressed through a two-tier map representation. The two maps have unique attributes assigned to them. GM is a compact global representation of the robots environment and Local Map (LM) is exclusively used for low-level navigation between intermediary points in the robots navigation horizon. The LM is marginalized at the end of each segment of the local navigation. Therefore, only the strategically selected GM features contribute to the monotonic increase in the map size. In the next section, the SLAM formulation will be discussed.

3. SLAM as a Nonlinear Batch Optimization

SLAM could be posed as a multivariate parameter estimation problem,[17,18] where a set of unknown state variables corresponding to the robot pose and map ($\mathbf{x}$) is estimated

https://doi.org/10.1017/S0263574711000543 Published online by Cambridge University Press



Fig. 3. (Colour online) SURF features in a typical indoor environment.

via the feature observations ($\mathbf{z}$) as well as odometry measurements ($\mathbf{u}$). This yields the following optimization problem:

$$\mathbf{x}^* \triangleq \arg\max_{\mathbf{x}} \quad p(\mathbf{z}, \mathbf{u}|\mathbf{x})p(\mathbf{x}), \tag{6}$$

where $p(\mathbf{z}, \mathbf{u}|\mathbf{x})$ is the observation likelihood and $p(\mathbf{x})$is the prior. The observation model that includes the ith observation to a feature and the odometric constraints at the ith robot pose could be factorized since these two types of measurements are conditionally independent as follows:

$$p(\mathbf{z}_i, \mathbf{u}_i|\mathbf{x}) = p(\mathbf{z}_i|\mathbf{x})\, p(\mathbf{u}_i|\mathbf{x}), \tag{7}$$

where $\mathbf{x}$ contains both map and the entire set of poses of the robot. Assuming observations to be Gaussian distributed

$$p(\mathbf{z}_i, \mathbf{u}_i|\mathbf{x}) = \eta\, \mathrm{e}^{-\frac{1}{2}(\varepsilon_i^{\mathrm{T}}\mathbf{R}_i^{-1}\varepsilon_i)}\, \mathrm{e}^{-\frac{1}{2}(\mu_i^{\mathrm{T}}\mathbf{U}_i^{-1}\mu_i)}, \tag{8}$$

with $\varepsilon_i = (\mathbf{z}_i - \mathbf{h}_i(\mathbf{x}))$, $i = 1, \ldots, z$ being the innovation of the stereo vision observation and $\mathbf{h}_i(\mathbf{x})$ being the predicted observation to the ith feature. $\mathbf{R}_i$ and $\mathbf{U}_i$ are the covariance matrices of stereo vision observation and odometry measurement, respectively. $\mu_i = \mathbf{u}_i - \mathbf{g}_i(\mathbf{x})$, where $\mathbf{g}_i(\mathbf{x})$ is the predicted odometry measurement. Now, assuming a uniform prior over the state variables, the cost function, $F(\mathbf{x})$ is derived by taking the negative of the log of the likelihood function (8)

$$\begin{aligned} F(\mathbf{x}) &= \frac{1}{2}\sum_{i=1}^{\mathrm{Z}} \varepsilon_i^{\mathrm{T}}\mathbf{R}_i^{-1}\varepsilon_i + \frac{1}{2}\sum_{i=1}^{n} \mu_i^{\mathrm{T}}\mathbf{U}_i^{-1}\mu_i, \\ &= \frac{1}{2}(\varepsilon^{\mathrm{T}}\mathbf{R}^{-1}\varepsilon + \mu^{\mathrm{T}}\mathbf{U}^{-1}\mu). \end{aligned} \tag{9}$$

Then, the maximum a posteriori estimate is

$$\mathbf{x}^* = \arg\min_x(\varepsilon^{\mathrm{T}}\mathbf{R}^{-1}\varepsilon + \mu^{\mathrm{T}}\mathbf{U}^{-1}\mu). \tag{10}$$

This represents the standard least-squares problem generally solved through Gauss–Newton iterations

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \Delta\mathbf{x}_k, \tag{11}$$

where $\Delta\mathbf{x}_k$ is the incremental change. In order to minimize the cost function, the algorithm starts at $k = 0$ with an initial estimate of the state vector $\hat{\mathbf{x}}_0$ and proceeds to calculate the approximations to the derivatives.

The first derivative

$$\nabla F(\mathbf{x}) = \mathbf{b} = -\mathbf{J}_h^{\mathrm{T}}\mathbf{R}^{-1}\varepsilon - \mathbf{J}_g^{\mathrm{T}}\mathbf{U}^{-1}\mu \tag{12}$$

and the Hessian matrix

$$\nabla^2 F(\mathbf{x}) \approx \mathbf{A} = \mathbf{J}_h^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{J}_h + \mathbf{J}_g^{\mathrm{T}}\mathbf{U}^{-1}\mathbf{J}_g, \tag{13}$$

where $\mathbf{J}_h$ and $\mathbf{J}_g$ are the respective Jacobian matrices of the functions $\mathbf{h}(\mathbf{x})$and $\mathbf{g}(\mathbf{x})$with respect to the state $\mathbf{x}$. Then, at the kth iteration,

$$\mathbf{A}_k\Delta\mathbf{x}_k = \mathbf{b}_k \tag{14}$$

and an improved estimate of the states are realized through Eq. (11). Iterative sequence continues until an acceptable minimum is reached. A good indication of a minimum is when the rate of change in residual is closer to zero.

4. Two-tier SLAM

4.1. Introduction

Humans are remarkably capable of finding their way in unknown environments without the engagement of controlled processes thus leaving residual resources available for other higher level cognitive processes.[33] This ability to efficiently maintain spatial representations in memory rests on a number of factors, including the efficient use of navigational aids, such as landmarks and other visual cues;[34] efficient, relatively effortless update of spatial representations involving the use of both short term and episodic memory; and long term and semantic memory stores.[35] Humans attach high weights to spatially familiar places in distinguishing anchor points or PoIs and procedural rules for way finding.[36] The proposed algorithm finds its inspiration in these remarkable capabilities of human perception in navigation.

Thus, an everyday navigation scenario could be used to best illustrate the intuition behind the proposed algorithm. Consider the problem depicted in Fig. 4. A person is required to navigate from point A to point B. The person would utilize the previously known three landmarks (PoIs) along the path for successful execution of the task. For instance, when the person reaches visual cue 1, he/she would recognize this and realize the need to turn right. Then, the second visual cue, a traffic roundabout, prompts him/her to take the left turn. Finally, the third visual cue prompts that the goal is nearby. Generally, the person would not pay extensive attention to any other visual cues along the way, apart from using them to help stay on the correct path (e.g., the curb markings on the road, pedestrian crossings, traffic signs, and traffic lights), short term memory.

https://doi.org/10.1017/S0263574711000543 Published online by Cambridge University Press

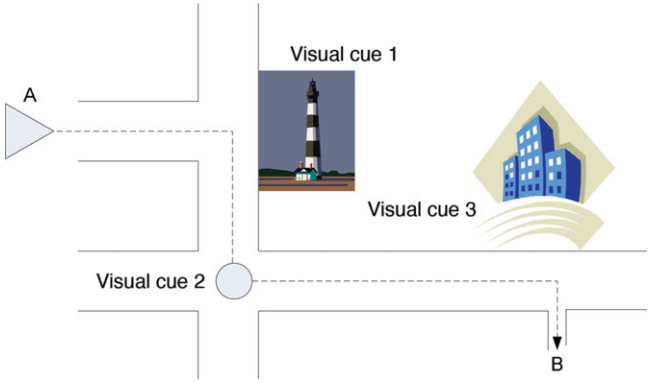


Fig. 4. (Colour online) An example of a human navigation task.

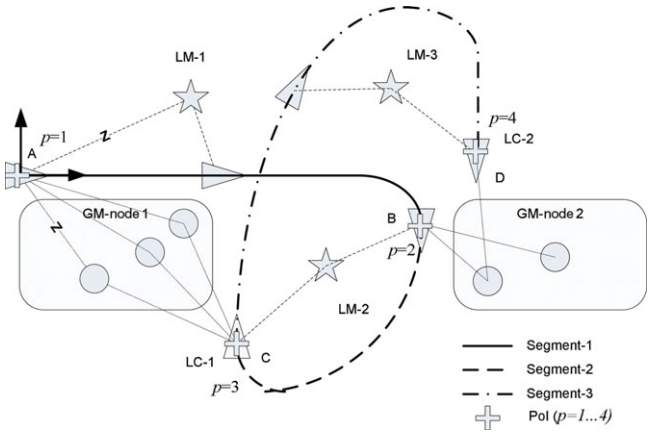


Fig. 5. A simplified illustration of the two-tier approach. The robot starts at the origin and ends its route at D with intermediate PoIs at B and C. The first Loop Closure (LC-1) occurs when the features observed at the first PoI are observed again from the third PoI. A second Loop Closure (LC-2) occurs when some of the features observed at second PoI are re-observed from fourth PoI.

Conceptually, the two-tier approach has similar notions of PoIs and lesser visual cues. For instance, PoIs are represented by strategically localized clusters of salient visual features collectively termed as a GM (GM-Node 1 and GM-Node2 in Fig. 5) where the term "node" referring to each individual cluster of the GM. This map enables the SLAM algorithm to be bounded globally and the corresponding map grows monotonically in dimensionality as the robot's exploration horizon expands. The second map, called the LM (LM-1, 2, and LM-3 in Fig. 5), enables the lower level navigation between two visual cues (from PoI: A-B, B-C and C-D in Fig. 5), and is conceptually analogous to the short-term visual cues.

When the navigation between PoIs (eg., A to B) is completed, the LM corresponding to the path segment (LM-1) is removed from the robot's state vector. Hence, the dimensionality of the LM state vector varies from segment to segment, but does not correlate with the growth of the robot's exploration horizon. This act of marginalization is the key factor for managing large feature densities.

4.2. Two-tier SLAM algorithm

Block diagram of the two-tier SLAM algorithm is shown in Fig. 6.

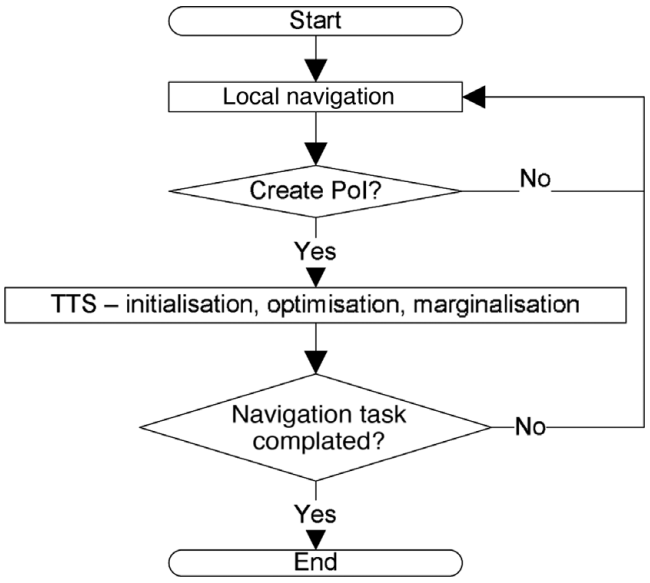


Fig. 6. Block diagram of the algorithm.

Start. The first PoI is created at the origin where the robot initiates its navigation (GM-Node 1 in Fig. 5). The other PoI's need to be selected such that the robot location error at these PoI's is sufficiently small in order to provide a suitable initialization point in the two-tier smoother (TTS) (item 2.) How far the robot can travel before this error is too large is a function of the sensor characteristics and the feature density. As the main focus of this paper is on two-tier algorithmic details, the PoIs were scheduled manually to initiate after 8 m of travel.

Analogous to the human behavior, where salient visual cues are used to localize within the global context, a set of map features (GM) is associated with each PoI for the localization of the robot within the global coordinate frame. Features corresponding to this high-level map dictate a high degree of viewpoint invariance to facilitate loop closure. In this work, SURF[32] that extracts and represent features using scale and affine invariant descriptors are used for GM representation. SURF-based GM is sparse and the features are only observed at certain PoIs in the robots trajectory.

1. *Local navigation.* The robot starts to navigate observing the environment (local map, LM-1 in Fig. 5). Viewpoint invariance of the features in the LM is not critical as long as the same feature can be observed from a range of successive frames. Thus, a low-computational cost feature extraction and tracking algorithms can be used to extract features suitable for the LM. In this work, the KLT algorithm[37,38] is used for this purpose.
2. *Two-tier smoother.* Once the robot reaches a PoI, it observes a set of global features (e.g., GM-node2), and subsequently, the algorithm sets out to execute the TTS (excluding first PoI) utilizing the LM and GM information currently available. For example, in Fig. 5, at LC-1, the TTS has GM-Node 1, 2, and LM-2 features in the augmented map state vector.
   The TTS is essentially a recursive batch optimization algorithm with dynamically varying state dimensions (*viz.*, Variable State Dimension Filter (VSDF)[39–41]—

https://doi.org/10.1017/S0263574711000543 Published online by Cambridge University Press

see Appendix for details). The TTS is thus formulated; assuming that the robot is at the pth PoI ($p > 1$), the current LM contains only the LM features that have been observed starting from the PoI at $p-1$. Thus, the new cost function to minimize contains a prior reflecting the linearized input data from previous TTS operations as well as the current measurement data

$$F(\mathbf{x}) = (\mathbf{x} - \hat{\mathbf{x}}_0)\mathbf{A}_0(\mathbf{x} - \hat{\mathbf{x}}_0)^{\mathrm{T}} + (\mathbf{d} - g(\mathbf{x}))^{\mathrm{T}}\mathbf{U}^{-1} \times (\mathbf{d} - g(\mathbf{x})) + (\mathbf{z} - h(\mathbf{x}))^{\mathrm{T}}\mathbf{R}^{-1}(\mathbf{z} - h(\mathbf{x})). \quad (15)$$

At this stage, the state vector contains only the previously observed features (prior to $p-1$) and robot poses up to PoI at $p-1$. Therefore, the state vector needs to be augmented with the new LM features, corresponding observations to these features, robot poses between PoIs $p-1$ and p and any newly initialized GM features. The change in state vector from the previous PoI is

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_r(p-1) \\ \mathbf{x}_{\mathrm{LM}}(p-1) \\ \mathbf{x}_{\mathrm{GM}}(p-1) \end{bmatrix} \rightarrow \quad \mathbf{x} = \begin{bmatrix} \mathbf{x}_r(p) \\ \mathbf{x}_{\mathrm{LM}}(p) \\ \mathbf{x}_{\mathrm{GM}}(p-1) \\ \mathbf{x}_{\mathrm{GM}}(p) \end{bmatrix}, \quad (16)$$

where $(\mathbf{x}_r(p))$ represents the limited pose history attributed to the pth LM and $(\mathbf{x}_{LM}(p))$ contains the LM features belonging to the pth LM. $\mathbf{x}_{\mathrm{GM}}(p-1)$ contains GM features that are already initialized during previous PoIs beginning from the origin and $(\mathbf{x}_{\mathrm{GM}}(p))$ is the newly observed GM features yet to be estimated.

This shows that the only component of this dynamic state vector that contributes to monotonic increase in the dimension is the observation of new GM features $(\mathbf{x}_{\mathrm{GM}}(p))$. Rest of the state changes with each PoI $(\mathbf{x}_{\mathrm{LM}}(p-1) \rightarrow \mathbf{x}_{\mathrm{LM}}(p))$ but does not contribute to continuous increase of the overall dimensionality of the state vector.

Having defined the state vector and the cost function, Levenberg–Marquardt optimization could be performed by defining the derivatives with respect to the current state vector

$$\begin{aligned} \mathbf{A} &= \mathbf{A}_0 + \mathbf{J}_h^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{J}_h + \mathbf{J}_g^{\mathrm{T}}\mathbf{U}^{-1}\mathbf{J}_g, \\ \mathbf{b} &= \mathbf{b}_0 + \mathbf{A}_0(\hat{\mathbf{x}}_0 - \hat{\mathbf{x}}^-) + \mathbf{J}_h^{\mathrm{T}}\mathbf{R}^{-1}\varepsilon + \mathbf{J}_g^{\mathrm{T}}\mathbf{U}^{-1}\mu. \end{aligned} \quad (17)$$

Once optimization has converged, LM features belonging to the current segment of the robot path, corresponding observations, robot poses, and any GM observations related to the removed robot poses are marginalized. This smoothing cycle is applied recursively at each new PoI. Thus, the consistency of the state estimate is preserved at regular intervals of the robots trajectory, while maintaining an efficient algorithm for large-scale navigation.

3. *Initialization of the TTS.* Smoother initialization could be performed based on odometry and feature observations. However, in the absence of quality odometric information (e.g., excessive slipping of wheels), an EKF-based solution could be used. (This is not to be confused with the consistency issue here as we only require a set of initial values that are closer to the truth. It will be shown in the results section that an EKF estimate of the mean gives a better sum-squared-error performance compared to similar error in plain odometry.) Thus, if the complete history of the robot poses estimated by the EKF within the two consecutive PoIs is

$$\hat{\mathbf{x}}_r^{\mathrm{EKF}} \triangleq \{\hat{\mathbf{x}}(t)\}; \quad t = 1, \ldots, N. \quad (18)$$

Then, the batch optimization starts with an initial estimate of the states

$$\hat{\mathbf{x}}_0 = \begin{bmatrix} \hat{\mathbf{x}}_r^{\mathrm{EKF}} \\ \hat{\mathbf{x}}_m^{\mathrm{EKF}} \end{bmatrix}, \quad (19)$$

where $\hat{\mathbf{x}}_m^{\mathrm{EKF}}$ is the map estimate from the EKF.

4. *Marginalization of the LM.* The relatively dense LM is periodically marginalized (at every PoI) from the state vector in order to reduce the overall computational cost of the SLAM algorithm. The marginalization is also performed through the TTS scheme.

Consider the state vector in Eq. (19), which is the starting point for the two-tier smoothing at a given (pth) PoI

$$\mathbf{x} = [\mathbf{x}_{r,p}, \mathbf{x}_m]^{\mathrm{T}}, \quad (20)$$

where $\mathbf{x}_m = [\mathbf{x}_{\mathrm{LM}}, \mathbf{x}_{\mathrm{GM}}]^{\mathrm{T}}$ is the combined two-tier at the current PoI. It is required to remove $(\mathbf{x}_{\mathrm{LM}})$ from $(\mathbf{x}_m)$. This could be achieved by straightforward removal of the corresponding map elements from the overall state vector. Thus,

$$\mathbf{x} = [\mathbf{x}_{r,p}, \mathbf{x}_{\mathrm{GM}}]^{\mathrm{T}}. \quad (21)$$

However, the smoothing algorithm now contains a prior inverse covariance term, which should also be marginalized accordingly. Assume that $\mathbf{A}_0$ is reordered to reflect the states to be removed $(\mathbf{A}_{\mathrm{LM},*})$ and retained $(\mathbf{A}_{R,*})$ such that

$$\mathbf{A}_0 = \begin{bmatrix} \mathbf{A}_{\mathrm{LM,LM}} & \mathbf{A}_{\mathrm{LM},R} \\ \mathbf{A}_{\mathrm{LM},R}^{\mathrm{T}} & \mathbf{A}_{R,R} \end{bmatrix}. \quad (22)$$

Then, using the matrix inversion lemma, the new prior after marginalization of LM could be calculated as

$$\mathbf{A}_0 \leftarrow \mathbf{A}_{R,R} - \mathbf{A}_{\mathrm{LM},R}^{\mathrm{T}}\mathbf{A}_{\mathrm{LM,LM}}^{-1}\mathbf{A}_{\mathrm{LM},R}. \quad (23)$$

The unique character of the LM features results in the marginalization without any information loss. This could be illustrated from an Information Filter perspective[42] noting that the Hessian $\mathbf{A}_0$ as the current information matrix. Then, each new observation "adds" information to the matrix

$$\mathbf{A} = \mathbf{A}_0 + \mathbf{J}_i^{\mathrm{T}}\mathbf{R}_i^{-1}\mathbf{J}_i. \quad (24)$$

In a general SLAM scenario, if a feature is marginalized, the potential information gain from a future observation to this feature $(\mathbf{J}_i^{\mathrm{T}}\mathbf{R}_i^{-1}\mathbf{J}_i)$ is lost, contributing to information loss. However, the LM features do not have an associated descriptor, which will allow data associations as these

https://doi.org/10.1017/S0263574711000543 Published online by Cambridge University Press

are not computed in tracking algorithms, such as KLT. Therefore, even if the LM features were retained in the state vector, they do not contribute to future information gains during a loop closure. Conversely, it could be argued that marginalization of these LM features therefore does not contribute to information loss.

Once the LM features are marginalized, the marginalization technique lends itself to the removal of robot poses prior to the current PoI as well since these robot poses are no longer needed in formulating the optimization problem.

5. *Removing observations associated with marginalized features.* As the TTS approach extends beyond the first loop closure, it becomes necessary to remove observations corresponding to the previous LM features, since they are marginalized as described in the previous section. Similarly, odometry information that is no longer required due to the marginalization of robot poses from the state vector should also be removed.
6. *Loop closure.* The GM features are used for closing the loop. This becomes quite simple given the *a priori* knowledge of the PoI locations.

5. Experimental Results

Several experiments have been conducted with both real and simulated data in order to validate algorithms presented in the paper, they also highlight the logical progression of thought that lead to the development of the overall framework, which is simple in approach yet powerful in addressing a host of related problems in stereo-vision-based SLAM.

5.1. Simulation results

Inconsistency of state estimation associated with the use of EKF is a well-known phenomena in large-scale maps;[11] however, what is less known is that the issue becomes critical in much smaller scale maps when a pair of stereo cameras are used with a small baseline.[29] In this section, several experiments are presented showing the limitations of EKF in the context of the sensor model used and a possible alternative that developed into the TTS.

1. *Estimator consistency and the observation model.* The impetus for this work came from the observation that even in a relatively small loop, SLAM estimates become inconsistent as soon as the observation model changed to the projective mapping described in Eq. (1) from a traditional range and bearing modal.[20]

   Figure 7 shows the simulated environment used in the first experiment. Here, a simulated mobile robot completes a single loop and the odometry is corrupted with Gaussian noise with zero mean and standard deviation $\sigma_{\text{velocity}} = 0.05$m/s, $\sigma_{\text{turnrate}} = 0.08$ rad/s, and $\sigma_{\text{cross}} = 0.0$.

   Figure 8 shows the results of this experiment with a traditional sensor (top) and a stereo sensor (bottom) using EKF. The stereo sensor is modeled to have a baseline of 0.088 m and noise parameters presented in Section 2. The right-hand column shows the corresponding average NEES over 50 Monte Carlo trials. The inconsistency is evident within a few iterations of the EKF in the case of the stereo sensor. The filter becomes optimistic.

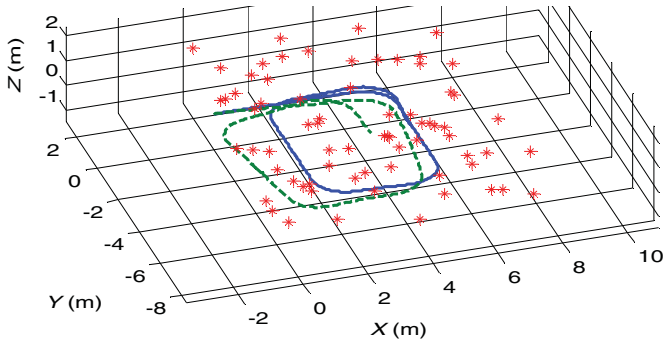


Fig. 7. (Colour online) Simulated environment. Solid line indicates the true path of the robot, dashed line shows the odometry, and asterisks indicate 3D features.

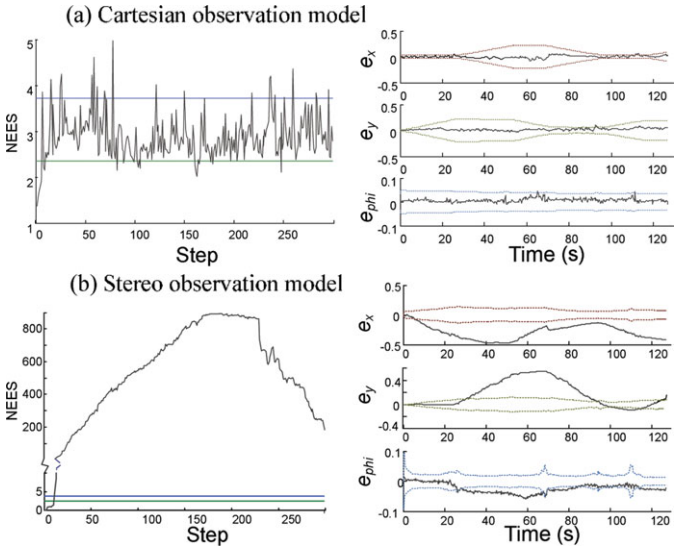


Fig. 8. (Colour online) Estimator results using EKF with (a) a traditional observation model and (b) a stereo model. Left column shows the average NEES for 50 trials with the 95% acceptance interval. Right column shows the robot pose errors with estimated $\pm 2\sigma$ bounds. Distance measures are in meters and angular measures are in radians.

Table I. A comparison of sum squared errors in the robot position for the different formulations.

	Odometry	EKF	UKF
Sum squared error	12.2	3.5	1.6

2. *A comparison of estimator error.* It was indicated earlier that even though the EKF results are inconsistent the mean estimates could be used as a better approximate over odometry in batch initialization. Table I shows a comparison of average sum squared error for odometry, EKF and an UKF estimates using the previously simulated environment with stereo observations. Filtered results perform better compared to raw odometry.
3. *TTS consistency.* Figure 9 shows the experimental results for a loop similar to that described in Fig. 5. The simulated environment consists of an evenly distributed set of features and the robot navigates in the path shown in Fig. 9(a). The simulated observations use the same model and noise parameters as in the first experiment.

https://doi.org/10.1017/S0263574711000543 Published online by Cambridge University Press

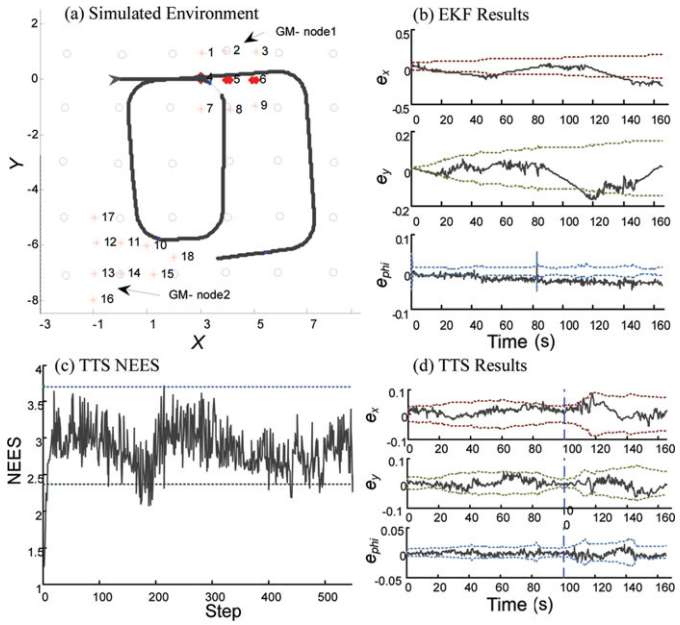


Fig. 9. (Colour online) Consistency experiment. (a) Simulated environment. (b) EKF Robot pose estimation errors with associated $\pm 2\sigma$ bounds. (c) Average NEES for 50 Monte Carlo trials. (d) TTS Robot pose estimation errors with associated $\pm 2\sigma$ bounds. The first loop closure occurs near the 100 s corresponding to the first node of the GM and the second loop closure occurs at 145 s corresponding to the second node of the GM.

Data association is assumed to be known. There are two nodes of the GM of which the first node initialized at the beginning of the robots trajectory (0,0) and the second node first initialized during the third leg of the smaller loop. The estimated error from an EKF SLAM is shown for comparison (Fig. 9b), which as discussed in refs. [28, 29] produces inconsistent state estimates. Figure 9(d) shows the error estimate from the new TTS approach.

As can be seen from the figure, the estimates are consistent and the two loop closures corresponding to the two nodes of the GM are visible at 100 s and 145 s, which show rapid reductions in uncertainty.

In Fig. 9(c), the average NEES for 50 trials of the TTS simulation is presented. It shows that the estimates are within the 2σ confidence bounds further confirming that the TTS is capable of producing consistent state estimates.

5.2. *Experimental results with real data*

For the experiments presented in this section, an ActiveMedia Robotics' PioneerDX was used as the mobile robotic platform (Fig. 10a (inset)). It consists of odometry sensors, a frontal array of sonar sensors, and a top mounted SICK Laser Measurement System (LMS). A Videre design small base line ($B = 0.088$ m) stereo camera was mounted on top of the LMS. Several experimental results obtained from real data acquired using this set up are presented below. In all the experiments, the robot was driven manually:

1. *Implications of visual features in the context of SLAM.* The experimental environment (Fig. 10a) consists of cubicles with narrow passageways (<2 m) limiting the effective FOV of the camera. As the robot moves, this results in features rapidly moving out of the camera FOV requiring new features to be initialized. The feature survival histogram shown in Fig. 10(b) of an EKF SLAM run in the experimental environment provides an empirical verification of this issue. The robot speed was set at approximately 0.2 ms^{-1} and camera frame rate was 4 Hz. In the trial shown, only 7% of the total features remained visible after 40 frames, which is about 10 s in duration and about 2 m of traverse. This results in a rapidly growing feature map. Implications of this are following: (a) the map becomes unmanageable within a short time span with respect to available computing power, and (b) data association becomes unreliable due to multiple hypothesizes satisfying a single feature selection criteria in a cluttered feature space.
2. *Ground truth.* Knowledge of ground truth is necessary to assess the filter performance. Even though, devices such as GPS are capable of providing accurate position estimates, they are ineffective in an indoor environment. Therefore, a separate feature-based SLAM solution using a laser range finder was used to generate the "ground truth." For this purpose, artificial beacons (laser reflectors) were strategically laid out in the experimental environment. Figure 11(a) shows a map generated through the laser-based EKF.
3. *Small loop.* The TTS scheme is applied to a small loop around the experimental area and resultant map is shown in Fig. 11(c). The robot pose estimate errors are shown in Fig. 11(d). For the small loop, estimates from the TTS are consistent.

   Figure 11(b) shows the Hessian of the resulting system. The significance of this is that it exposes the sparse nature

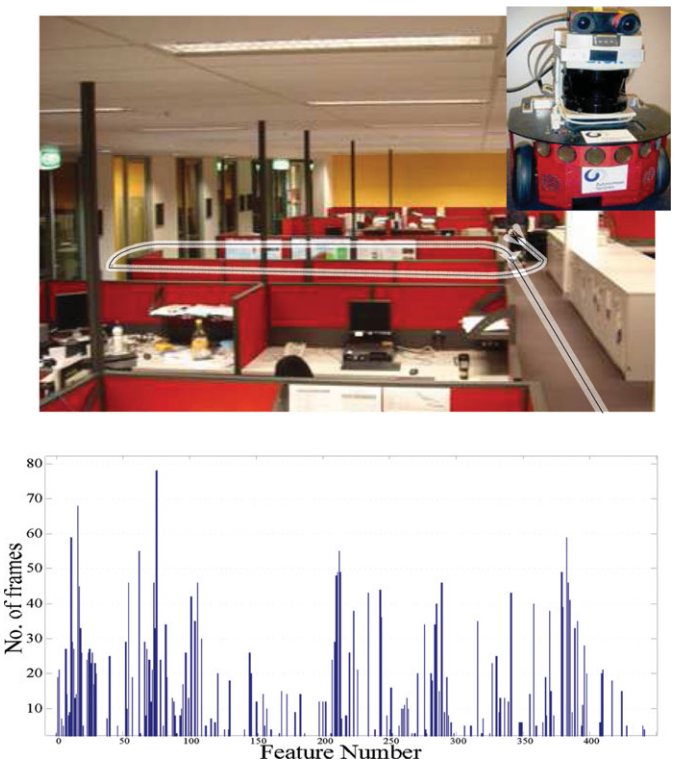


Fig. 10. (Colour online) (a) Experimental area and the pioneer robot (inset). (b) The survival rate of 450 KLT features for a typical loop around the experimental area.

https://doi.org/10.1017/S0263574711000543 Published online by Cambridge University Press

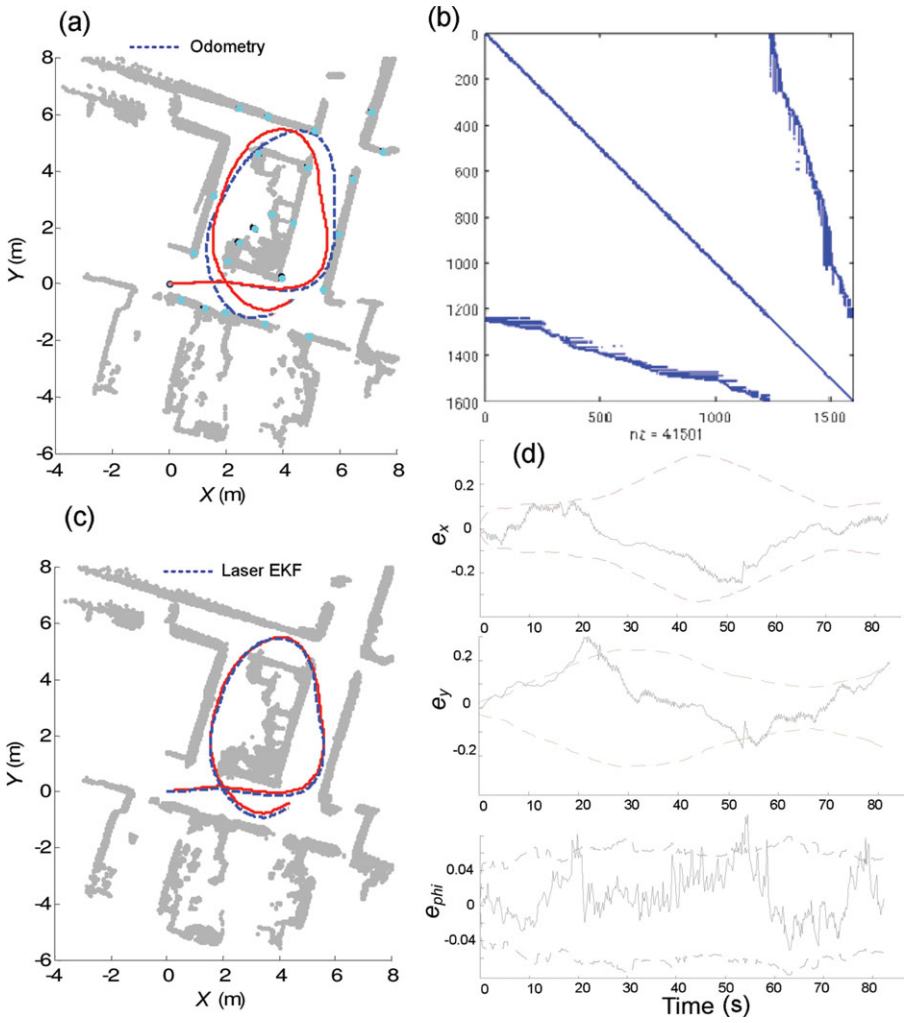


Fig. 11. (Colour online) Small loop experiment (a) map of the laser-based EKF, the robot pose estimates are used as "ground truth," (b) sparse Hessian associated with TTS, (c) map based on TTS estimate, and (d) robot pose error in TTS estimate.

of matrices involved in the solution, which could be harnessed to improve the computational cost. In recent years, studies have shown the potential for considerable performance gains through exploitation of these matrices and the nature of the problem.[18] The potential of such exploitations is part of the ongoing research on the use of TTS.

4. *A typical two-tier map.* Figure 12(a) shows the full map generated by TTS in a multiloop experiment around the experimental area. The algorithm created two GM nodes in this particular experiment. This resulted in two LMs. The first LM corresponds to the smaller inner loop and the second LM consists of the path that extends to a second outer loop. Error in robot pose estimates is shown in Fig. 12(b).

   First loop closure occurs when the robot revisits near the origin of the system. At the creation of the second PoI, a TTS optimization was carried out and the corresponding LM was marginalized. At the end of the robot trajectory, a TTS optimization is performed again. Considerable irregularities in the path estimate near the bottom-right corner of Fig. 12(a) are due to a person moving in front of the camera momentarily generating a large number of spurious observations. This phenomenon is also reflected in Fig. 12(b) with increased error just before 200 s. However, the filter becomes consistence again in few steps.

Reader's attention is drawn to the fact that the TTS approach enables highly compressed map representations compared to traditional methods. In this example, only 15 map features are used to represent the entire environment, which is shown in the final map in Fig. 12(c). If a general EKF approach was to be used, the map would constitute of approximately 450 features.

The experimental environment and the vision system, in general, spawn large numbers of spurious features, for reasons, such as parallax and stereo mismatches. These features are catastrophic in a SLAM implementation and require additional signal conditioning at the sensor level. SURF features for inclusion in the GM have been selected only if the features have been consecutively visible for a predetermined number of frames since the initial observation at the PoI. This isolates the highly visible features from the clutter and also reduces the number of GM features considerably.

https://doi.org/10.1017/S0263574711000543 Published online by Cambridge University Press

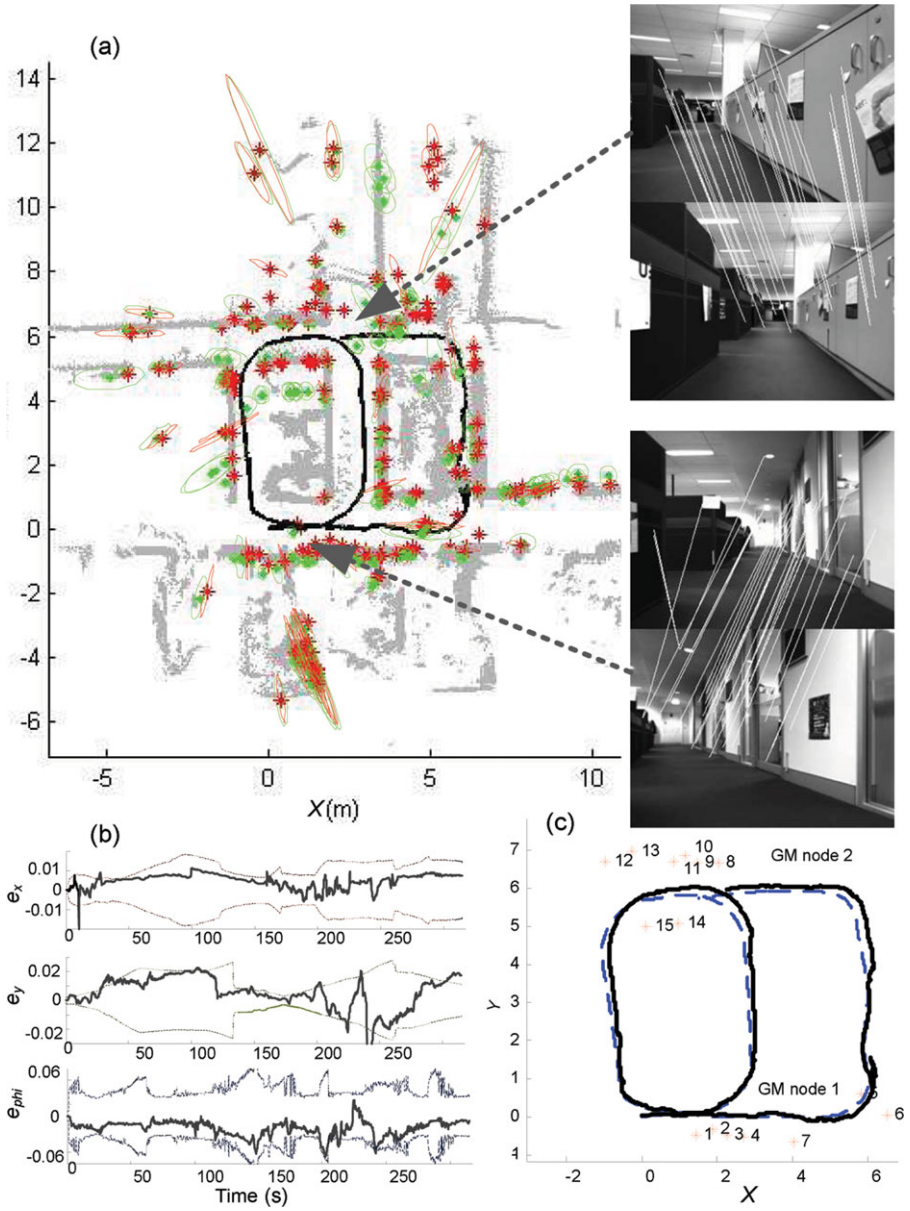


Fig. 12. (Colour online) Multiloop experimental results. (a) The full map. Green ellipses are the EKF estimates of LM features used for TTS initialization. Figure also shows two sets of pictures corresponding to the two areas near the PoIs. White lines indicate the SURF matches between each pair. (b) Robot pose error for the full length of the robot navigation. (c) The resultant GM and the robot path. Dashed line shows the "ground truth." GM consists of two nodes with features 1–7 representing first node and 8–15 the second node.

6. Conclusion

This paper presented a novel framework for the small baseline stereo-vision-based SLAM. The unique two-tier map representation provides a compact method for representing large environments through a scheme of PoIs and a GM. The LM allows for temporary use of large number of features for local navigation similar to how humans use short-term memory in navigation. This is realized through a fast-nondescriptor-based feature tracker. These features are marginalized to maintain the tractability of the developed algorithm within manageable constraints. With high-dimensional feature descriptors, the GM provides the loop closure information. It was shown through simulations and experiments that the two-tier approach provides consistent results in challenging indoor environments.

The main contributions that arise from this article are the development of a framework where multiple maps are used that correlate to different sensing modalities for SLAM and a coherent procedure for disassociating large number of features from the estimation scheme while maintaining consistency. The scheme is well suited for the stereo vision sensor concerned. However, the scheme has the potential to be generalized for different sensors. For instance, a combination of laser-range-finder-based LM and a vision-based GM is a hypothetical feasibility. It is also interesting to further explore the concept of PoIs, especially in relation to how humans cognize key elements in the environment in deriving spatial representations in unfamiliar terrain. This could eventually lead to the development of principled PoI deployment schemes that are goal-driven and efficient. Currently, the authors are investigating a human-in-the-loop

https://doi.org/10.1017/S0263574711000543 Published online by Cambridge University Press

approach[43] for deploying this framework in a realistic SLAM scenario with a wearable vision system.

Finally, as the optimization algorithms are currently run offline, we are investigating suitable techniques that could be utilized in improving the computational aspects in a large-scale real-world implementation. We are currently looking at work by Dellaert *et al.*[18,44] as a feasible starting point in this regard.

Acknowledgment

This work is supported by the ARC Centre of Excellence program, funded by the Australian Research Council (ARC) and the New South Wales State Government.

References

1. N. M. Kwok, G. Dissanayake and Q. P. Ha, "Bearing-Only SLAM Using a SPRT Based Gaussian Sum Filter," *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2005)*, (2005) pp. 1109–1114.
2. T. Lemaire, S. Lacroix and J. Sola, "A Practical 3D Bearing-Only SLAM Algorithm," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2005)*, (2005) pp. 2449–2454.
3. J. Sola, A. Monin, M. Devy and T. Lemaire, "Undelayed Initialization in Bearing Only SLAM," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2005)*, (2005) pp. 2499–2504.
4. J. M. M. Montiel, J. Civera and A. J. Davison, "Unified Inverse Depth Parametrization for Monocular SLAM," *Proceedings of the Robotics: Science and Systems Conference (RSS 2006)*, (2006) pp. 1–8.
5. I. K. Jung, Simultaneous Localization and Mapping in 3D Environments with Stereovision *Ph.D. Thesis* (Toulouse: LAAS, Institut National Polytechnique, 2004) pp. 1–118.
6. A. J. Davison, Mobile Robot Navigation Using Active Vision *D.Phil/Ph.D. Thesis* (Oxford, UK: University of Oxford, 1998).
7. S. Se, D. Lowe and J. Little, "Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks," *Int. J. Robot. Res.* **21**, 735–758 (Aug. 2002).
8. A. Davids, "Urban search and rescue robots: From tragedy to technology," *IEEE Intell. Syst.* **17**, 81–83 (2002).
9. L. Matthies and S. Shafer, "Error modeling in stereo navigation," *IEEE J. Robot. Autom.* **3**, 239–248 (Jun. 1987).
10. G. Sibley, L. Matthies and G. Sukhatme, "Bias Reduction and Filter Convergence for Long Range Stereo," *Proceedings of the 12th International Symposium of Robotics Research (ISRR 2005)*, San Francisco, CA, USA (2005) pp. 1–10.
11. T. Bailey, J. Nieto, J. Guivant, M. Stevens and E. Nebot, "Consistency of the EKF-SLAM Algorithm," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2006)*, Beijing, China (2006) pp. 3562–3568.
12. S. Huang and G. Dissanayake, "Convergence Analysis for Extended Kalman Filter Based SLAM," *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2006)*, Orlando, Florida (2006) pp. 412–417.
13. S. J. Julier and J. K. Uhlmann, "A Counter Example to the Theory of Simultaneous Localization and Map Building," *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2001)*, (2001) pp. 4238–4243.
14. S. J. Julier, J. K. Uhlrnann and H. F. Durrant-Whyte, "A New Approach for Filtering Nonlinear Systems," *Proceedings of the 14 th American Control Conference*, Seattle, WA, USA (1995) Vol. 3, pp. 1628–1632.
15. S. J. Julier and J. K. Uhlmann, "Unscented Filtering and Nonlinear Estimation," *Proceedings of the IEEE* (2004) Vol. 92, pp. 401–422.
16. S. Julier, J. Uhlmann and H. F. Durrant-Whyte, "A new method for the nonlinear transformation of means and covariances in filters and estimators," *IEEE Trans. Autom. Control* **45**, 477–482 (2000).
17. M. Deans, Bearings-Only Localization and Mapping *Ph.D. Thesis* (Pittsburgh, PA, USA: Robotics Institute, Carnegie Mellon University, 2005).
18. F. Dellaert and M. Kaess, "Square root SAM: Simultaneous localization and mapping via square root information smoothing," *Int. J. Robot. Res.* **25**, 1181–1203 (Dec. 2006).
19. G. Dissanayake, H. Durrant-Whyte and T. Bailey, "A Computationally Efficient Solution to the Simultaneous Localisation and Map Building (SLAM) Problem," *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2000)*, (2000) pp. 1009–1014.
20. S. B. Williams, G. Dissanayake and H. Durrant-Whyte, "An Efficient Approach to the Simultaneous Localisation and Mapping Problem," *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2002)*, (2002) pp. 406–411.
21. M. Bosse, P. Newman, J. Leonard, M. Soika, W. Feiten and S. Teller, "An Atlas Framework for Scalable Mapping," *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2003),* (2003) Vol. 2, pp. 1899–1906.
22. C. Estrada, J. Neira and J. D. Tardos, "Hierarchical SLAM: Real-time accurate mapping of large environments," *IEEE Trans. Robot.* **21**, 588–596 (2005).
23. T. Rudolph, P. Patrick and B. Wolfram, "Multi-Level Surface Maps for Outdoor Terrain Mapping and Loop Closing," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems* (2006) pp. 2276–2282.
24. J. L. Blanco, J. A. Fernandez-Madrigal and J. Gonzalez, "Toward a unified bayesian approach to hybrid metric–topological SLAM," *IEEE Trans. Robot.* **24**, 259–270 (2008).
25. A. Martinelli, A. Tapus, K. O. Arras and R. Siegwart, "Multi-Resolution SLAM for Real World Navigation," *Proceedings of the Robotics Research* (2005) pp. 442–452.
26. F. Ferreira, I. Amorim, R. Rocha and J. Dias, "T-SLAM: Registering Topological and Geometric Maps for Robot Localization in Large Environments," *Proceedings of the IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI 2008)*, (2008) pp. 392–398.
27. F. C. Fortenbaugh, J. C. Hicks, L. Hao and K. A. Turano, "High-speed navigators: Using more than what meets the eye," *J. Vis.* **6**, 565–579 (Apr. 20, 2006).
28. D. C. Herath, K. R. S. Kodagoda and G. Dissanayake, "Modeling Errors in Small Baseline Stereo for SLAM," *Proceedings of the 9th International Conference on Control, Automation, Robotics and Vision (ICARCV 2006)*, Singapore (2006) pp. 1–6.
29. D. C. Herath, K. R. S. Kodagoda and G. Dissanayake, "Stereo Vision Based SLAM: Issues and Solutions," **In**: *Vision Systems: Applications* (G. Obinata and A. Dutta, eds.) (Pro literatur verlag, Vienna, Austria, 2007) pp. 565–582.
30. Y. Bar-Shalom, X.-R. Li and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation* (Wiley InterScience, Somerset, New Jersey, 2001).
31. C. Harris and M. Stephens, "A Combined Corner and Edge Detector," *Proceedings of the Alvey Vision Conference* (1988) pp. 147–151.
32. H. Bay, T. Tuytelaars and L. V. Gool, "SURF: Speeded Up Robust Features," *Proceedings of the 9th European Conference on Computer Vision* (2006).
33. R. F. Wang and J. R. Brockmole, "Human navigation in nested environments," *J. Exp. Psychol. Learn. Mem. Cogn.* **29**, 398–404 (2003).

https://doi.org/10.1017/S0263574711000543 Published online by Cambridge University Press

34. R. F. Wang and E. S. Spelke, "Updating egocentric representations in human navigation," *Cognition* **77**, 215–250 (2000).
35. S. Gopal, R. L. Klatzky and T. R. Smith, "Navigator: A psychologically based model of environmental learning through navigation," *J. Environ. Psychol.* **9**, 309–331 (Dec. 1989).
36. R. G. Golledge, "Place recognition and wayfinding: Making sense of space," *Geoforum* **23**, 199–214 (1992).
37. T. Kanade and M. Okutomi, "A stereo matching algorithm with an adaptive window: Theory and experiment," *IEEE Trans. Pattern Anal. Mach. Intell.* **16**, 920–932 (Sep. 1994).
38. J. Shi and C. Tomasi, "Good Features to Track," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 1994)*, Seattle (1994) pp. 593–600.
39. P. F. McLauchlan, "The Variable State Dimension Filter applied to Surface-Based Structure from Motion," School of Electrical Engineering, Information Technology and Mathematics, University of Surrey, Guildford, CVSSP Technical Report VSSP-TR-4/99 (1999).
40. B. Triggs, P. McLauchlan, R. Hartley and A. Fitzgibbon, "Bundle Adjustment A Modern Synthesis" **In**: *Vision Algorithms: Theory and Practice* (W. Triggs, A. Zisserman and R. Szeliski, eds.) (Springer Verlag, 2000) pp. 298–375.
41. P. F. McLauchlan and D. W. Murray, "Active camera calibration for a head-eye platform using the variable state-dimension filter," *IEEE Trans. Pattern Anal. Mach. Intell.* **18**, 15–22 (1996).
42. S. Thrun, D. Koller, Z. Ghahramani, H. Durrant-Whyte and A. Y. Ng, "Simultaneous Mapping and Localization With Sparse Extended Information Filters: Theory and Initial Results," Computer Science Department, Carnegie Mellon University, Pittsburgh, PA, Technical Report CMU-CS-02-112 (2002).
43. D. C. Herath, *Stereo Vision Based Simultaneous Localisation and Mapping: A Human Centred Approach* (VDM Verlag Dr. Müller, Berlin, Germany, 2011).
44. M. Kaess, A. Ranganathan and F. Dellaert, "iSAM: Incremental smoothing and mapping," *Robot. IEEE Trans.* **24**, 1365–1378 (2008).

Appendix: The Variable State Dimension Filter

The VSDF[39–41] is similar to the EKF; however, the VSDF formulation also takes the advantages of the batch optimization techniques as well. The technique allows for solving the general nonlinear optimization problem involving a global state and a variable number of local states. The formulation is ideally suited to the two-tier framework in that the GM represents the global state of the problem while LM represents the variable local states, which allows for marginalization as the robot evolves in the environment. Once a state is removed from the state vector, related observations also need to be removed from the optimization problem, which can be dealt within the VSDF framework.

Given the cost function to minimize (9) and the associated derivatives (12) and (13), assume that the first observation ($i = 1$) is to be removed from the cost function. Then, it is required to assess its contribution to the derivatives and thus to the cost function. This amounts to approximating the contribution from the observation at the current state estimate, $\hat{\mathbf{x}}_0$ by linearizing $\mathbf{h}_1(\cdot)$

$$\mathbf{h}_1(\mathbf{x}) \approx \mathbf{h}_1(\hat{\mathbf{x}}_0) + \mathbf{J}_1(\mathbf{x} - \hat{\mathbf{x}}_0), \tag{A 1}$$

and incorporating it in a Gaussian prior $(\mathbf{x} - \hat{\mathbf{x}}_0)\mathbf{A}_0(\mathbf{x} - \hat{\mathbf{x}}_0)^{\mathrm{T}}$ such that the new cost function contains both linearized measurements and nonlinear measurements.[17]

$$F(\mathbf{x}) = (\mathbf{x} - \hat{\mathbf{x}}_0)\mathbf{A}_0(\mathbf{x} - \hat{\mathbf{x}}_0)^{\mathrm{T}} + \sum_{i=2}^{Z} \varepsilon_i^{\mathrm{T}} \mathbf{R}_i^{-1} \varepsilon_i + \sum_{i=1}^{n} \mu_i^{\mathrm{T}} \mathbf{U}_i^{-1} \mu_i, \tag{A 2}$$

where

$$\begin{aligned}(\mathbf{x} - \hat{\mathbf{x}}_0)\mathbf{A}_0(\mathbf{x} - \hat{\mathbf{x}}_0)^{\mathrm{T}} &\approx \varepsilon_1^{\mathrm{T}} \mathbf{R}_1^{-1} \varepsilon_1, \\ &\approx (\mathbf{z}_1 - \mathbf{J}_1(\mathbf{x} - \hat{\mathbf{x}}_0))^{\mathrm{T}} \mathbf{R}_1^{-1} (\mathbf{z}_1 - \mathbf{J}_1(\mathbf{x} - \hat{\mathbf{x}}_0)).\end{aligned} \tag{A 3}$$

The contribution from the first observation to the Hessian ($\mathbf{A}_0$) and the Jacobian ($\mathbf{b}_0$) are

$$\begin{aligned}\mathbf{A}_0 &= \mathbf{J}_1^{\mathrm{T}} \mathbf{R}_1^{-1} \mathbf{J}_1, \\ \mathbf{b}_0 &= \mathbf{J}_1^{\mathrm{T}} \mathbf{R}_1^{-1} (\mathbf{z}_1 - \mathbf{h}_1(\hat{\mathbf{x}}_0)).\end{aligned} \tag{A 4}$$

Then, the Hessian and the Jacobian of the new cost function are

$$\begin{aligned}\mathbf{A} &\approx \mathbf{A}_0 + \sum_{i=2}^{Z} \mathbf{J}_{hi}^{\mathrm{T}} \mathbf{R}_i^{-1} \mathbf{J}_{hi} + \sum_{i=1}^{n} \mathbf{J}_{gi}^{\mathrm{T}} \mathbf{U}^{-1} \mathbf{J}_{gi}, \\ \mathbf{b} &\approx \mathbf{b}_0 + \mathbf{A}_0(\mathbf{x} - \hat{\mathbf{x}}_0) + \sum_{i=2}^{Z} \mathbf{J}_{hi}^{\mathrm{T}} \mathbf{R}_i^{-1} \varepsilon_i + \sum_{i=1}^{n} \mathbf{J}_{gi}^{\mathrm{T}} \mathbf{U}^{-1} \mu_i.\end{aligned} \tag{A 5}$$

This technique could be applied to an arbitrary number of observation removals, and in an iteration of the Levenberg–Marquardt minimization, the general form of the cost function, its Hessian and the Jacobian, could thus be written as

$$\begin{aligned}\mathbf{F}(\mathbf{x}) &= (\mathbf{x} - \hat{\mathbf{x}}_0)\mathbf{A}_0(\mathbf{x} - \hat{\mathbf{x}}_0)^{\mathrm{T}} + \sum \varepsilon^{\mathrm{T}} \mathbf{R}^{-1} \varepsilon + \sum \mu^{\mathrm{T}} \mathbf{U}^{-1} \mu, \\ \mathbf{A} &= \mathbf{A}_0 + \sum \mathbf{J}_h^{\mathrm{T}} \mathbf{R}^{-1} \mathbf{J}_h + \sum \mathbf{J}_g^{\mathrm{T}} \mathbf{U}^{-1} \mathbf{J}_g, \\ \mathbf{b} &= \mathbf{b}_0 + \mathbf{A}_0(\mathbf{x} - \hat{\mathbf{x}}_0) + \sum \mathbf{J}_h^{\mathrm{T}} \mathbf{R}^{-1} \varepsilon + \sum \mathbf{J}_g^{\mathrm{T}} \mathbf{U}^{-1} \mu.\end{aligned} \tag{A 6}$$

$\mathbf{A}_0$ and $\mathbf{b}_0$ capture the contribution of the linearized observations.

https://doi.org/10.1017/S0263574711000543 Published online by Cambridge University Press