Covariance Recovery from a Square Root Information Matrix for Data Association

Michael Kaess

CSAIL, Massachusetts Institute of Technology, 32 Vassar St., Cambridge, MA 02139

Frank Dellaert

College of Computing, Georgia Institute of Technology, 801 Atlantic Dr., Atlanta, GA 30332

Abstract

Data association is one of the core problems of simultaneous localization and mapping (SLAM), and it requires knowledge about the uncertainties of the estimation problem in the form of marginal covariances. However, it is often difficult to access these quantities without calculating the full and dense covariance matrix, which is prohibitively expensive. We present a dynamic programming algorithm for efficient recovery of the marginal covariances needed for data association. As input we use a square root information matrix as maintained by our incremental smoothing and mapping (iSAM) algorithm. The contributions beyond our previous work are an improved algorithm for recovering the marginal covariances and a more thorough treatment of data association now including the joint compatibility branch and bound (JCBB) algorithm. We further show how to make information theoretic decisions about measurements before actually taking the measurement, therefore allowing a reduction in estimation complexity by omitting uninformative measurements. We evaluate our work on simulated and real-world data.

Key words: data association, smoothing, simultaneous localization and mapping

1. Introduction

Data association is an essential component of simultaneous localization and mapping (SLAM) [1]. The data association problem in SLAM, which is also known as the correspondence problem, consists of matching the current measurements with their corresponding previous observations. Correspondences can be obtained directly between measurements taken at different times, or by matching the current measurements to landmarks already contained in the map based on earlier measurements. A solution to the correspondence problem provides frame-to-frame matching, but also allows for closing large loops in the trajectory. Such loops are more difficult to handle as the estimation uncertainty is much larger than between successive frames, and the measurements might even be taken from a different direction.

Performing data association can be difficult especially in ambiguous situations, but is greatly simplified when the state estimation uncertainties are known. Parts of the overall SLAM state estimate uncertainty are needed to make a probabilistic decision based on the maximum likelihood (ML) criterion or when using the joint compatibility branch and bound (JCBB) algorithm by Neira and Tardos [2], a popular algorithm for SLAM [3, 1]. The parts that are required are so-called marginal covariances that represent the uncertainties between a relevant subset of the variables, for example a pose and a landmark pair.

However, it is generally difficult to recover the exact marginal covariances in real-time. But as mobile robot applications require decisions to be made in real-time, we need an efficient solution for recovering the marginal covariances. While it is trivial to recover the covariances from an Extended Kalman Filter (EKF), its uncertainties are inconsistent when non-linear measurement functions are present, which is typically the case. Other solutions to SLAM, for example based on iterative equation solvers such as [4, 5, 6, 7], cannot directly access the marginal covariances. An alternative is to use conservative estimates of the marginal covariances as in [8], however, they will provide less constraints for ambiguous data association decisions and therefore fail earlier.

Our solution provides efficient access to the marginal covariances based on the square root information matrix. Such a factored information matrix is maintained by our incremental smoothing and mapping (iSAM) algorithm [9], which efficiently updates the factored representation when new measurements arrive. Our solution consists of a dynamic programming algorithm that recovers only parts of the full covariance matrix based on the square root information matrix, thereby avoiding to calculate the full dense covariance matrix, which contains a number of entries that is quadratic in the number of variables.

Email addresses: kaess@mit.edu (Michael Kaess), dellaert@cc.gatech.edu (Frank Dellaert)

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
02 JUL 2009		**00-00-2009 to 00-00-2009**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Covariance Recovery from a Square Root Information Matrix for Data Association	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Massachusetts Institute of Technology,Computer Science and Artificial Intelligence Laboratory (CSAIL),Cambridge,MA,02139**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Data association is one of the core problems of simultaneous localization and mapping (SLAM), and it requires knowledge about the uncertainties of the estimation problem in the form of marginal covariances. However, it is often difficult to access these quantities without calculating the full and dense covariance matrix, which is prohibitively expensive. We present a dynamic programming algorithm for efficient recovery of the marginal covariances needed for data association. As input we use a square root information matrix as maintained by our incremental smoothing and mapping (iSAM) algorithm. The contributions beyond our previous work are an improved algorithm for recovering the marginal covariances and a more thorough treatment of data association now including the joint compatibility branch and bound (JCBB) algorithm. We further show how to make information theoretic decisions about measurements before actually taking the measurement, therefore allowing a reduction in estimation complexity by omitting uninformative measurements. We evaluate our work on simulated and real-world data.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**15**	

The contributions over our previous work [10, 9] are an improved marginal covariance recovery algorithm and a detailed discussion of the algorithm. We also add the JCBB algorithm to our discussion of data association techniques, and use a uniform mathematical presentation to contrast the presented methods. Beyond typical data association work, we further show how to use these marginal covariances to determine the value of a specific measurement, allowing to drop redundant or uninformative measurements in order to increase estimation efficiency. We present detailed evaluations on simulated and real-world data. And finally we provide insights into using JCBB versus the RANSAC algorithm by Fischler and Bolles [11].

2. Covariance Recovery

We show how to efficiently obtain selected parts of the covariance matrix, the so-called marginal covariances, based on a square root information matrix. But first we briefly introduce the square root information matrix and an efficient algorithm for calculating it.

2.1. Square Root Information Matrix

The square root information matrix appears in the context of smoothing and mapping (SAM) [12], a smoothing formulation of the SLAM problem. The smoothing formulation includes the complete robot trajectory, that is all poses $\mathbf{x}_i$ ($i \in \{0 \dots M\}$) in addition to the landmarks $\mathbf{l}_j$ ($j \in \{1 \dots N\}$). This is in contrast to typical filtering methods that only keep the most recent pose by marginalizing out previous poses. Smoothing provides the advantage of a sparse information matrix, therefore allowing to efficiently solve [12] the equation system.

The SLAM problem typically contains non-linear functions through robot orientation and bearing measurements and therefore requires iterative linearization and solution steps. Please see [12, 9] for a detailed treatment of the process and measurement models, and their linearization and combination into one large least-squares system. One step of the resulting linearized SLAM problem can be written as

$$\arg\min_{\mathbf{x}} \|A\mathbf{x} - \mathbf{b}\|^2 \tag{1}$$

where A is the measurement Jacobian of the SLAM problem at the current linearization point, $\mathbf{x}$ the unknown state vector combining poses and landmarks, and $\mathbf{b}$ the so-called right-hand side that is irrelevant in this work. Solutions to the state vector $\mathbf{x}$ in (1) can be found based on the *square root information matrix* R, an upper triangular matrix that is found by Cholesky factorization of the information matrix $\mathcal{I} := A^T A = R^T R$ or directly by QR factorization of the measurement Jacobian $A = Q \begin{bmatrix} R \\ 0 \end{bmatrix}$. The upper triangular shape of the square root information matrix allows efficient solution of the SLAM problem by back-substitution.

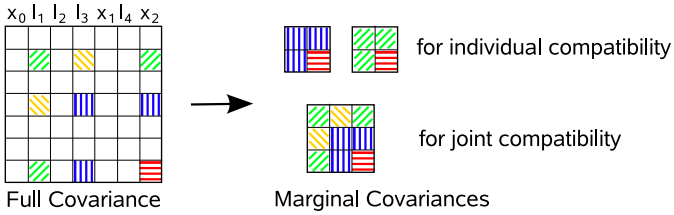


Figure 1: Only a small number of entries of the dense covariance matrix are of interest for data association. In this example, both the individual and the combined marginals between the landmarks $\mathbf{l}_1$ and $\mathbf{l}_3$ and the latest pose $\mathbf{x}_2$ are retrieved. As we show here, these entries can be obtained without calculating the full dense covariance matrix.

In practice it is too expensive to refactor the information matrix each time a new measurement arrives. Instead, our incremental smoothing and mapping (iSAM) algorithm [9] updates the square root information matrix directly with the new measurements. Periodic variable reordering keeps the square root information matrix sparse, allowing efficient solution by back-substitution as well as efficient access to marginal covariances, which is described next.

2.2. Recovering Marginal Covariances

Knowledge of the relative uncertainties between subsets $\{j_1, \dots, j_K\}$ of the SLAM variables are needed for data association. In particular, the marginal covariances

$$\begin{bmatrix} \Sigma_{j_1 j_1} & \Sigma^T_{j_2 j_1} & \cdots & \Sigma^T_{j_K j_1} \\ \Sigma_{j_2 j_1} & \Sigma_{j_2 j_2} & \cdots & \Sigma^T_{j_K j_2} \\ \vdots & \vdots & \ddots & \vdots \\ \Sigma_{j_K j_1} & \Sigma_{j_K j_2} & \cdots & \Sigma_{j_K j_K} \end{bmatrix} \tag{2}$$

are the basis for advanced data association techniques that we discuss in detail in Section 3, as well as for information theoretic decisions about the value of measurements as discussed in Section 4. This marginal covariance matrix contains various blocks from the full covariance matrix, as is shown in Fig. 1. Calculating the full covariance matrix to recover these entries is not an option because the covariance matrix is always densely populated with n^2 entries, where n is the number of variables. However, we show in the next section that it is not necessary to calculate all entries in order to retrieve the exact values of the relevant blocks.

Recovering the exact values for all required entries without calculating the complete covariance matrix is not straightforward, but can be done efficiently by again exploiting the sparsity structure of the square root information matrix R. In general, the covariance matrix is obtained as the inverse of the information matrix

$$\Sigma := (A^T A)^{-1} = (R^T R)^{-1} \tag{3}$$

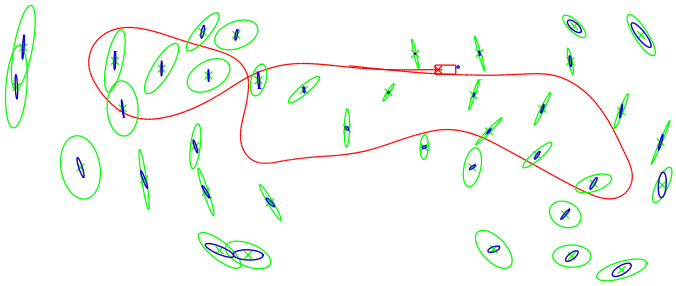


Figure 2: Marginal covariances *projected into the current robot frame* (robot indicated by red rectangle) for a short trajectory (red curve) and some landmarks (green crosses). The exact covariances (blue, smaller ellipses) obtained by our fast algorithm coincide with the exact covariances based on full inversion (orange, mostly hidden by blue). Note the much larger conservative covariance estimates (green, large ellipses) as recovered in our earlier work [9].

based on the factor matrix R by noting that

$$R^T R\Sigma = I \tag{4}$$

and performing a forward, followed by a back-substitution

$$R^T Y = I, \quad R\Sigma = Y. \tag{5}$$

Because the information matrix is not band-diagonal in general, this would seem to require calculating all n^2 entries of the fully dense covariance matrix, which is infeasible for any non-trivial problem. This is where we exploit the sparsity of the square root information matrix R. Both, Golub and Plemmons [13] and Triggs *et al.* [14] present an efficient method for recovering only the entries σ_{ij} of the covariance matrix Σ that coincide with non-zero entries in the factor matrix $R = (r_{ij})$

$$\sigma_{ll} = \frac{1}{r_{ll}} \left(\frac{1}{r_{ll}} - \sum_{j=l+1, r_{lj} \neq 0}^{n} r_{lj} \sigma_{jl} \right) \tag{6}$$

$$\sigma_{il} = \frac{1}{r_{ii}} \left(- \sum_{j=i+1, r_{ij} \neq 0}^{l} r_{ij} \sigma_{jl} - \sum_{j=l+1, r_{ij} \neq 0}^{n} r_{ij} \sigma_{lj} \right) \tag{7}$$

for $l = n, \ldots, 1$ and $i = l-1, \ldots, 1$, where the other half of the matrix is given by symmetry. Note that the summations only apply to non-zero entries of single columns or rows of the sparse matrix R. This means that in order to obtain the top-left-most entry of the covariance matrix (note that the corresponding entry in R is always non-zero), we at most have to calculate all other entries that correspond to non-zeros in R. For any other non-zero entry, even less work is required, as only entries further to the right are needed. For recovering all entries corresponding to non-zero entries in R, this algorithm yields $O(n)$ time complexity for band-diagonal matrices and matrices with only a small number of entries far from the diagonal, but can be more expensive for general sparse R. In particular, if the otherwise sparse matrix contains a dense block of side length s, the complexity is $O(nk + s^3)$ and the constant factor s^3 can become dominant for practical purposes.

Data association might additionally require access to entries of the covariance matrix for which the corresponding entries of the square root information matrix are zero, but they can easily be calculated based on a dynamic programming approach. Note that the algorithm in (6) and (7) does not restrict which entries we can recover. Rather, it tells us that the minimum we have to calculate is always a subset of the entries corresponding to non-zeros in R. Our dynamic programming approach shown in Alg. 1 performs the minimum amount of calculations needed to recover any set of entries that we want to calculate, such as a marginal covariance between a small set of variables. An example of how the recovery proceeds is shown in Fig. 3. Fig. 2 shows the marginal covariances obtained by this algorithm for the first part of the Victoria Park sequence. Note that they coincide with the exact covariances obtained by full matrix inversion.

3. Data Association Techniques

In this section we present some of the most common data association techniques and show that they require access to marginal covariances as provided by our work. In particular we discuss the nearest neighbor method, the maximum likelihood formulation and the joint compatibility branch and bound algorithm. We also discuss the landmark-free case, where data association is based for example on dense laser scan matching.

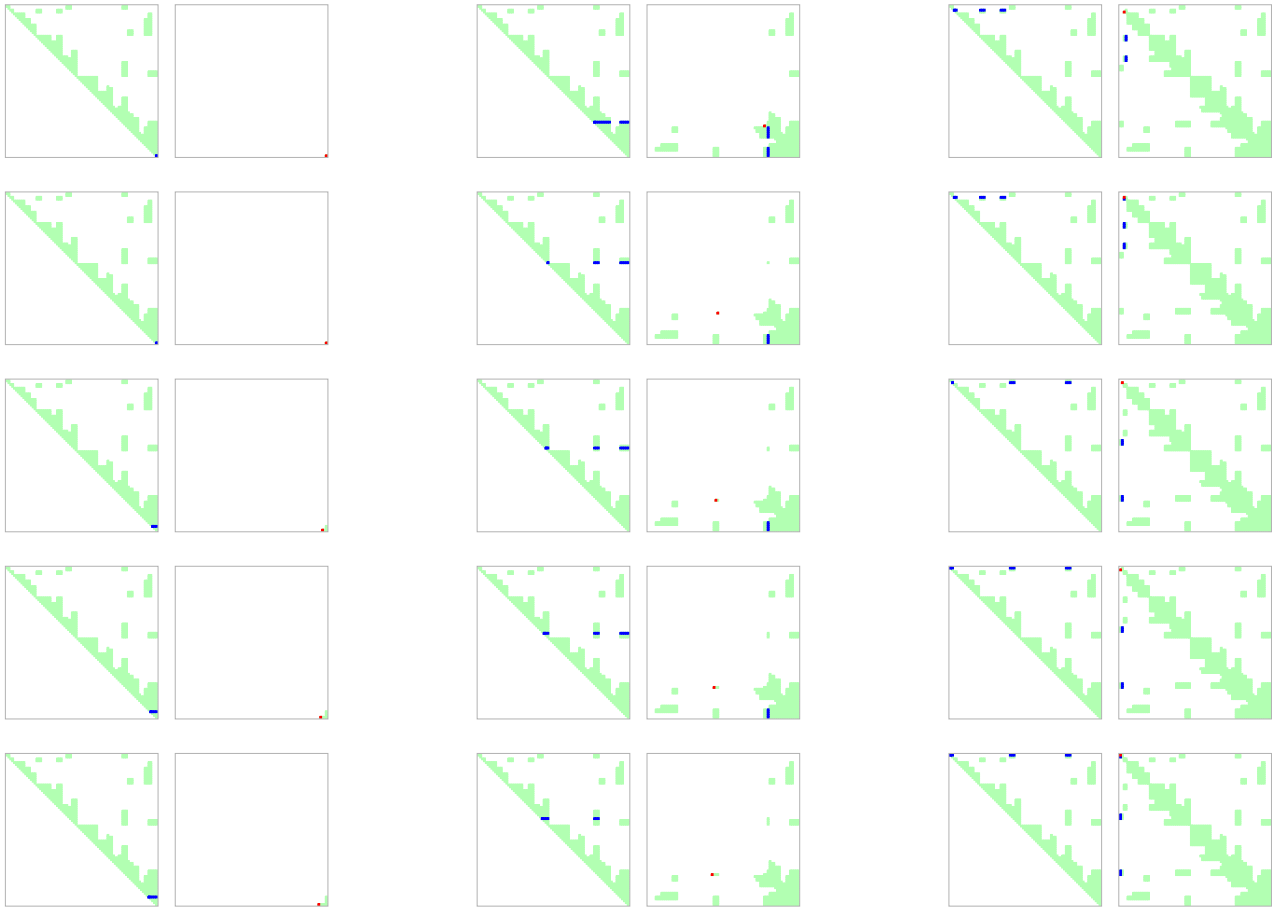
3.1. Nearest Neighbor (NN)

For completeness we include the often used nearest neighbor (NN) approach to data association, even though it is not sufficient for most practical applications. NN assigns each measurement to the closest landmark predicted by the current state estimate, as shown in Fig. 4. The predicted measurement $\mathbf{z}_{ij}$ taken at pose $\mathbf{x}_i$ of landmark $\mathbf{l}_j$ is given by the measurement model

$$\mathbf{z}_{ij} := h_{ij}(\mathbf{x}) + \mathbf{v}_{ij} \tag{8}$$

where $\mathbf{v}_{ij}$ is additive zero-mean Gaussian noise with covariance Γ. Note that $\mathbf{x}$ is the current state vector that includes at least the robot pose $\mathbf{x}_i$ and the landmark $\mathbf{l}_j$.

We formulate the correspondence problem for a specific measurement $k \in \{1 \ldots K\}$ in the following way: We have an actual measurement $\tilde{\mathbf{z}}_k$ that we know was taken at time i_k and we want to determine which landmark j_k gave rise to this measurement. We define the nearest neighbor cost $D^{2,\mathrm{NN}}_{kj}$ of the hypothesis $j_k = j$ simply as the squared



(a) First 5 steps. (b) Steps 245 to 249. (c) Last 5 steps.

Figure 3: The process of recovering marginal covariances: The three columns show successive steps of recovering the entries of the covariance matrix that correspond to non-zero entries in the square root information matrix. The overall process takes 716 steps. For each step the square root information matrix is shown in the left square and the partially computed covariance matrix in the right square. The matrix entry to be calculated is shown in red, while entries required for its calculation are blue and the remaining non-zero entries are green.

distance between the actual measurement $\tilde{\mathbf{z}}_k$ and its prediction $\hat{\mathbf{z}}_{i_k j} = h_{i_k j}(\hat{\mathbf{x}})$ based on the mean of the current state estimate $\hat{\mathbf{x}}$ as follows

$$D_{kj}^{2,\mathrm{NN}} := \left\| h_{i_k j}(\hat{\mathbf{x}}) - \tilde{\mathbf{z}}_k \right\|^2 . \tag{9}$$

We accept the hypothesis that landmark j gave rise to measurement $\tilde{\mathbf{z}}_k$ only if this squared distance falls below a threshold

$$D_{kj}^{2,\mathrm{NN}} < D_{\max}^2 \tag{10}$$

where the threshold $D_{\max}^2$ limits the maximum squared distance allowed between a measurement and its prediction in order to still be considered as a potential match. The threshold is typically chosen based on the nature of the data, i.e. the minimum distance between landmarks as well as the expected maximum uncertainty of the measurements.

When considering multiple measurements at the same time, the NN approach can be formulated as a minimum cost assignment problem that also takes mutual exclusion into account. Mutual exclusion means that once a landmark is assigned to a measurement it is no longer available for other assignments. For K measurements and N landmarks, we form a $K \times N$ matrix that contains the cost for each possible assignment. In order to deal with unassigned measurements, which can arise from newly observed landmarks or from noise, we augment the matrix by K columns that all contain the threshold D_{max}^2

$$\mathcal{D} = \begin{bmatrix} D_{11}^2 & D_{12}^2 & \cdots & D_{1N}^2 & D_{\max}^2 & \cdots \\ D_{21}^2 & \ddots & & \vdots & \vdots & \\ \vdots & & & & & \\ D_{K1}^2 & \cdots & & D_{KN}^2 & D_{\max}^2 & \end{bmatrix} . \tag{11}$$

We use the Jonker-Volgenant-Castanon (JVC) algorithm

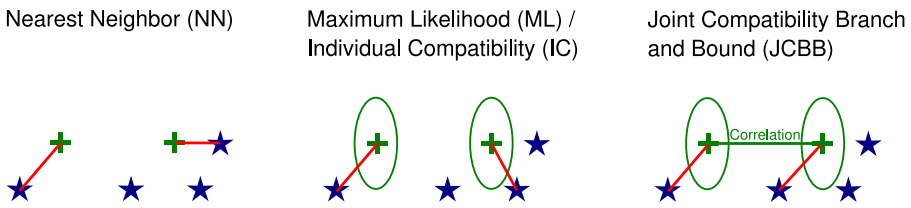


Figure 4: Comparison of some common data association techniques discussed in this work. Nearest neighbor assigns a measurement (blue star) to the closest landmark (green cross). Individual compatibility takes the estimation uncertainty between sensor and landmark into account, here indicated as green ellipses. The assignment is different as it now corresponds to nearest neighbor under a Mahalanobis distance. Joint compatibility branch and bound additionally takes into account the correlation between the landmarks, again yielding a different assignment for this example.

[15] to optimally solve this assignment problem.

The greatest advantage of NN over other methods is that it does not require any knowledge of the uncertainties of the estimation problem. However, that is also its greatest weakness, as NN will eventually fail once the uncertainties become too large, as is the case when closing large loops.

3.2. Maximum Likelihood (ML)

The maximum likelihood (ML) solution to data association [16] is based on probabilistic considerations and takes into account the relative estimation uncertainties between the current robot location and the landmark in the map, as indicated in Fig. 4. This technique is also known as individual compatibility (IC) matching [2]. It is similar to the NN method, but with the Euclidean distance replaced by the Mahalanobis distance based on the projected estimation uncertainty.

The ML data association decision is based on a probabilistic formulation of the question whether a measurement was caused by a specific landmark or not. Particularly, for an individual measurement $\tilde{\mathbf{z}}_k$ we are interested in the probability $P(\tilde{\mathbf{z}}_k, j_k = j|Z^-)$ that this measurement was caused by landmark j, given all previous measurements Z^-. This expression does not yet contain any connection to the SLAM state estimate. However, we can simply introduce the state $\mathbf{x}$, a vector that combines all landmarks $\mathbf{l}_j$ and poses $\mathbf{x}_i$, and then integrate it out again

$$\begin{aligned} P(\tilde{\mathbf{z}}_k, j_k = j|Z^-) &= \int_{\mathbf{x}} P(\tilde{\mathbf{z}}_k, j_k = j, \mathbf{x}|Z^-) \\ &= \int_{\mathbf{x}} P(\tilde{\mathbf{z}}_k, j_k = j|\mathbf{x}, Z^-) P(\mathbf{x}|Z^-) \\ &= \int_{\mathbf{x}} P(\tilde{\mathbf{z}}_k, j_k = j|\mathbf{x}) P(\mathbf{x}|Z^-) \end{aligned} \tag{12}$$

where we applied the chain rule to obtain the measurement likelihood $P(\tilde{\mathbf{z}}_k, j_k = j|\mathbf{x}, Z^-) = P(\tilde{\mathbf{z}}_k, j_k = j|\mathbf{x})$, which is independent of all previous measurements Z^- given the state $\mathbf{x}$. We already know the prior $P(\mathbf{x}|Z^-)$, as that is the current state estimate, or in the case of iSAM simply a normal distribution with mean $\hat{\mathbf{x}}$ and covariance Σ. We also know the measurement likelihood $P(\tilde{\mathbf{z}}_k, j_k = j|\mathbf{x})$ as it is defined by the predictive distribution from (8). The complete probability distribution is therefore an integral over normal distributions that can be simplified to a single normal +distribution

$$\begin{aligned} &P(\tilde{\mathbf{z}}_k, j_k = j|Z^-) \\ &= \int_{\mathbf{x}} \frac{1}{\sqrt{|2\pi\Gamma|}} e^{-\frac{1}{2}\left\|h_{i_k j}(\mathbf{x}) - \tilde{\mathbf{z}}_k\right\|^2_{\Gamma}} \frac{1}{\sqrt{|2\pi\Sigma|}} e^{-\frac{1}{2}\|\mathbf{x}-\hat{\mathbf{x}}\|^2_{\Sigma}} \\ &\approx \frac{1}{\sqrt{|2\pi C_{i_k j}|}} e^{-\frac{1}{2}\left\|h_{i_k j}(\hat{\mathbf{x}}) - \tilde{\mathbf{z}}_k\right\|^2_{C_{i_k j}}} \end{aligned} \tag{13}$$

where $\|\mathbf{x}\|^2_{\Sigma} := \mathbf{x}^T \Sigma^{-1} \mathbf{x}$, and the covariance $C_{i_k j}$ is defined as

$$C_{i_k j} := \left.\frac{\partial h_{i_k j}}{\partial \mathbf{x}}\right|_{\hat{\mathbf{x}}} \Sigma \left.\frac{\partial h_{i_k j}}{\partial \mathbf{x}}\right|^T_{\hat{\mathbf{x}}} + \Gamma. \tag{14}$$

Note that for the approximation in (13) to be precise, a good linearization point must be chosen, which iSAM [9] provides based on periodic relinearization steps. We drop the normalization factor of (13) as it does not depend on the actual measurement but is constant given the state. Further, we take the negative logarithm of the remaining expression from (13) to obtain the maximum likelihood cost function

$$D^{2,\mathrm{ML}}_{kj} := \|h_{i_k j}(\hat{\mathbf{x}}) - \tilde{\mathbf{z}}_k\|^2_{C_{i_k j}} \tag{15}$$

where again we evaluate the hypothesis that a specific measurement $\tilde{\mathbf{z}}_k$ taken in image i_k was caused by the j^{th} landmark. Note that this distance function is exactly the same as for the NN problem in (9) except that it takes into account the uncertainties of the state estimate given by the covariance Σ. As this squared distance function follows a chi-square distribution, we base the acceptance decision

$$D^{2,\mathrm{ML}}_{kj} < \chi^2_{d,\alpha} \tag{16}$$

on the chi-square test, where α is the desired confidence level and d is the dimension of the measurement. Going back to probabilities for a moment, the threshold being exceeded means that the difference of the sample (the actual measurement $\tilde{\mathbf{z}}_k$) from the mean of the actual distribution (measurement $\hat{\mathbf{z}}_{i_k j}$ predicted by the measurement model

Algorithm 1 Dynamic programming algorithm for marginal covariance recovery inspired by Golub's partial sparse matrix inversion algorithm [13]. The function `recover` obtains arbitrary entries of the covariance matrix with the minimum number of calculations, while reusing previously calculated entries. We have chosen to use a hash table for fast random access to the already calculated entries. The example function `marginal_cov` returns the marginal covariance matrix for the variables specified by its argument `indices`. For practical implementations, care has to be taken to avoid stack overflows caused by the recursive calls. Entries should be processed starting from the right-most column, while taking the variable ordering into account.

```
n = rows(R)
for i=1:n do
  # precalculate, needed multiple times
  diag[i] = 1 / R[i,i]
done

# efficiently recover an arbitrary entry,
# hashed for fast random access
function hash recover(i, l) =
  # sum over sparse entries of one row
  # see equations (6) and (7)
  function sum_j(i) =
    sum = 0
    for each entry rij of sparse row i of R do
      if j<>i then
        if j>l then
          lj = recover(l, j)
        else
          lj = recover(j, l)
        endif
        sum += rij * lj
      endif
    done
    return sum
  if i = l then # diagonal entries, equation (6)
    return (diag[l] * (diag[l] - sum_j(l)))
  else # off-diagonal entries, equation (7)
    return (- sum_j(i) * diag[i])
  endif

# example: recover marginal covariance of variables
# given by indices
function marginal_cov(R, indices) =
  n_indices = length(indices)
  for r=1:n_indices do
    for c=1:n_indices do
      P[r,c] = recover(indices[r],indices[c])
    done
  done
return P
```

based on the state estimate given by $\hat{\mathbf{x}}$ and Σ) is statistically significant, and we can assume that the measurement was not caused by that specific landmark as assumed by our hypothesis. For example, for a confidence level of 95% and a three dimensional measurement, the appropriate threshold is $\chi^2_{3,0.95} = 7.8147$. The relevant chi-square values are either obtained from a lookup table, or are calculated based on the incomplete gamma function and a root-finding method [17].

When considering multiple measurements simultaneously the resulting matching problem can again be reduced to a minimum cost assignment problem and solved using JVC in much the same way as was shown for NN in the previous section. More details on this minimum cost assignment problem and on how to deal with spurious measurements in a more principled probabilistic framework are provided by Dellaert [18].

In summary we can state that ML data association requires access to the following subset of the full covariance matrix

$$\Sigma_{\mathbf{x}_{ij}} = \begin{bmatrix} \Sigma_{jj} & \Sigma_{ij}^T \\ \Sigma_{ij} & \Sigma_{ii} \end{bmatrix} \tag{17}$$

for each landmark j that is considered as a candidate, see Fig. 1 for an example. As we have shown in Section 2, these entries can be obtained efficiently from the square root information matrix.

3.3. Joint Compatibility Branch and Bound (JCBB)

Instead of making independent decisions for each measurement, we now consider all measurements at the same time, which allows us to also take correlations between landmarks into account, as shown in Fig. 4. This is called joint compatibility [2], and works in ambiguous configurations in which IC often fails. Such ambiguous configurations are often encountered in real-world data association problems, in particular under high motion uncertainty and when closing large loops.

This time we are interested in the probability of all measurements simultaneously given an estimate for the landmark locations as well as the robot pose. A joint hypothesis

$$\mathbf{j} = (\mathbf{j}_1, \ldots, \mathbf{j}_K)^T \tag{18}$$

for K measurements

$$\mathbf{k} = (\mathbf{k}_1, \ldots, \mathbf{k}_K)^T \tag{19}$$

assigns each measurement $\mathbf{k}_\iota$ to a landmark $\mathbf{j}_\iota$, see Fig. 5 for an example. We combine the individual measurement functions $\mathbf{z}_{i\mathbf{j}_1}, \ldots, \mathbf{z}_{i\mathbf{j}_K}$ into the joint measurement vector $\mathbf{z}_{i\mathbf{j}}$

$$\mathbf{z}_{i\mathbf{j}} := \begin{pmatrix} \mathbf{z}_{i\mathbf{j}_1} \\ \vdots \\ \mathbf{z}_{i\mathbf{j}_K} \end{pmatrix} = \begin{pmatrix} h_{i\mathbf{j}_1}(\mathbf{x}) + \mathbf{v} \\ \vdots \\ h_{i\mathbf{j}_K}(\mathbf{x}) + \mathbf{v} \end{pmatrix} = h_{i\mathbf{j}}(\mathbf{x}) + \mathbf{v}_{i\mathbf{j}}. \tag{20}$$

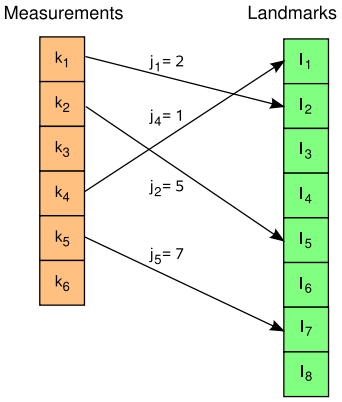


Figure 5: Example of a joint hypothesis $\mathbf{j}$ that assigns a set of measurements to landmarks. Measurements that are not assigned can be used to initialize new landmarks. Not all landmarks are necessarily visible or detected. Note that mutual exclusion prevents two measurements in the same frame from being assigned to the same landmark.

Analogous to the ML case we get the same expression for the probability

$$P(\tilde{\mathbf{z}}_\mathbf{k}, \mathbf{j}_\mathbf{k} = \mathbf{j}|Z^-) \approx \frac{1}{\sqrt{|2\pi C_{i\mathbf{j}}|}} e^{-\frac{1}{2}\|h_{i\mathbf{j}}(\hat{\mathbf{x}}) - \tilde{\mathbf{z}}_\mathbf{k}\|^2_{C_{i\mathbf{j}}}} \tag{21}$$

from (13), but the mean $\tilde{\mathbf{z}}_\mathbf{k}$ is now a joint measurement, and the covariance $C_{i\mathbf{j}}$ is defined as

$$C_{i\mathbf{j}} := \left.\frac{\partial h_{i\mathbf{j}}}{\partial \mathbf{x}}\right|_{\hat{\mathbf{x}}} \Sigma \left.\frac{\partial h_{i\mathbf{j}}}{\partial \mathbf{x}}\right|^T_{\hat{\mathbf{x}}} + \Gamma_K \tag{22}$$

based on the Jacobian of the joint measurement function $h_{i\mathbf{j}}$ at the current state estimate $\hat{\mathbf{x}}$ with covariance Σ.

For a specific joint measurement $\tilde{\mathbf{z}}_\mathbf{k} := (\tilde{\mathbf{z}}_{\mathbf{k}_1}, \ldots, \tilde{\mathbf{z}}_{\mathbf{k}_K})^T$ the joint compatibility cost or squared distance function is now given by

$$D^{2,\mathrm{JC}}_{\mathbf{kj}} = \|h_{i\mathbf{j}}(\hat{\mathbf{x}}) - \tilde{\mathbf{z}}_\mathbf{k}\|^2_{C_{i\mathbf{j}}}\,. \tag{23}$$

The *joint compatibility test* for the joint hypothesis $\mathbf{j}$ is given by

$$D^{2,\mathrm{JC}}_{\mathbf{kj}} < \chi^2_{d,\alpha} \tag{24}$$

where α is again the desired confidence level, but d is now the sum of the dimensions of all measurements that are part of the hypothesis.

Because the measurements are no longer independent, the search space is too large to exhaustively enumerate all possible assignments. In contrast, for NN and ML data association we simply enumerate each possible landmark-to-feature assignment. For K features and N landmarks there are

$$\Pi^{\mathrm{individual}} = KN \tag{25}$$

different costs to evaluate. Only then do we deal with unassigned measurements and mutual exclusion by means of the minimum cost assignment problem. For joint compatibility, however, the measurements are not independent and we therefore have to evaluate the cost of each joint hypothesis that assigns each feature either to a landmark or leaves it unassigned. The overall number of these joint hypotheses is far larger than the number of individual hypotheses. In particular, we can assign one of N landmarks to the first feature or leave it unassigned, and at the same time one of the $N-1$ remaining ones (considering mutual exclusion) to the second or leave it unassigned and so on, yielding

$$\Pi^{\mathrm{joint}} = \frac{(N+1)!}{(N+1-K)!} \tag{26}$$

possible joint hypotheses to evaluate, which is $O(N^K)$.

The combinatorial complexity of the state space is addressed by the joint compatibility branch and bound (JCBB) algorithm by Neira and Tardos [2]. *Branch and bound* represents the space of all possible hypotheses by a tree, as shown in Fig. 6, where each leaf node represents a specific hypothesis. The goal of the algorithm is to find the hypothesis with the highest number of jointly compatible matchings at the lowest squared distance $D^{2,\mathrm{JC}}_{\mathbf{kj}}$ according to (23). The algorithm starts with an empty hypothesis, the root of the tree, and processes nodes from a queue starting from the most promising one as determined by a lower bound of the cost. For this purpose the algorithm adds to the queue new hypotheses for successors of nodes that it encounters. Efficiency is achieved by discarding subtrees whose lower bound is higher than the current best hypothesis. Furthermore it is essential to evaluate the joint compatibility cost incrementally to avoid inversion of an increasingly large matrix. For more details see the original paper by Neira and Tardos [2].

For the case of JCBB data association we need more entries than in the ML case:

$$\Sigma_{\mathbf{x}_{i\mathbf{j}}} = \begin{bmatrix} \Sigma_{\mathbf{j}_1\mathbf{j}_1} & \cdots & \Sigma^T_{\mathbf{j}_K\mathbf{j}_1} & \Sigma^T_{i\mathbf{j}_1} \\ \vdots & \ddots & \vdots & \vdots \\ \Sigma_{\mathbf{j}_K\mathbf{j}_1} & \cdots & \Sigma_{\mathbf{j}_K\mathbf{j}_K} & \Sigma^T_{i\mathbf{j}_K} \\ \Sigma_{i\mathbf{j}_1} & \cdots & \Sigma_{i\mathbf{j}_K} & \Sigma_{ii} \end{bmatrix} \tag{27}$$

Fig. 1 shows which entries of the full covariance matrix these correspond to, based on a simple example. In particular we need additional off-diagonal entries $\Sigma_{\mathbf{j}_\iota\mathbf{j}_{\iota'}}$ not contained in the set of individual covariances $\Sigma_\mathbf{y}$ for the same landmarks. These additional off-diagonal entries specify the correlations between landmarks and are essential for ambiguous situations which often arise in large loop closings. As we have shown in Section 2, these entries can be obtained efficiently from the square root information matrix.

3.4. Search Region and Pose Uncertainty

How far do we need to search for potential loop closures? We face this question for example for landmark-based data association, as we want to quickly identify

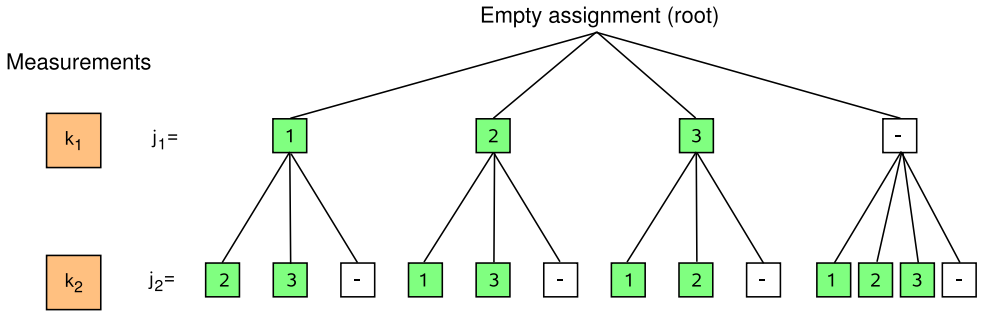


Figure 6: Tree of possible joint hypotheses for a small example with 3 landmarks (green) and 2 measurements (orange). The tree has one level per measurement, and the branching degree depends on the number of landmarks. Note that a measurement can remain unassigned, indicated by a dash ("-"), and that mutual exclusion is enforced.

landmarks that are potential candidates for a match in a given correspondence decision. The same problem also appears in place recognition, such as our work in [19], where restricting the search region allows keeping the computational requirements of place recognition low even for large-scale applications. Furthermore, the problem appears in pose-only estimation, such as laser-based scan matching, where we need to identify earlier parts of the trajectory (and therefore parts of the map) that are candidates for a loop closing.

The question about restricting the search region can again be answered by recovering parts of the covariance matrix. In the simplest case we only recover the uncertainty of the current pose, which for iSAM is the bottom right-most block of the full covariance matrix and therefore trivial to recover from the square root information matrix. However, imagine a vehicle driving in a straight line for a long distance and then performing a small loop. The absolute pose uncertainty is very large as we are far from the starting point. This would result in a large search region for loop closing. However, the actual uncertainty within the small loop is much smaller and is obtained by taking into account the marginal covariances between the current pose and some previous pose along the loop. The off-diagonal blocks contained in this marginal covariance describe the correlation between poses, and therefore the search region will now be much smaller.

Similarly to the ML data association case, the question to ask is how likely it is that two poses $\mathbf{x}_i$ and $\mathbf{x}_{i'}$ refer to nearby locations in space given all measurements Z that we have seen so far. Note that location does not include the robot's orientation. With the same argument as earlier we obtain

$$\begin{aligned} & P(\mathbf{x}_i = \mathbf{x}_{i'} | Z) \qquad (28) \\ = & \int_x P(\mathbf{x}_i = \mathbf{x}_{i'} | \mathbf{x}, Z) P(\mathbf{x} | Z) \\ = & \int_x \frac{1}{\sqrt{|2\pi\Lambda|}} e^{-\frac{1}{2}\|f(\mathbf{x}_{i'}) - \mathbf{x}_i\|^2_\Lambda} \frac{1}{\sqrt{|2\pi\Sigma|}} e^{-\frac{1}{2}\|\mathbf{x} - \hat{\mathbf{x}}\|^2_\Sigma} \\ \approx & \frac{1}{\sqrt{|2\pi C_{ii'}|}} e^{-\frac{1}{2}\|f(\hat{\mathbf{x}}_{i'}) - \hat{\mathbf{x}}_i\|^2_{C_{ii'}}} \end{aligned}$$

where we define "nearby" through a Gaussian distribution that is similar to the process model, but without odometry, and ignoring orientation

$$\mathbf{x}_{i'} = f(\mathbf{x}_i) + \mathbf{w} \qquad (29)$$

where $\mathbf{w}$ is normally distributed zero-mean noise with covariance matrix Λ, and the covariance $C_{ii'}$ is defined as

$$C_{ii'} := \left.\frac{\partial f}{\partial \mathbf{x}}\right|_{\hat{\mathbf{x}}} \Sigma \left.\frac{\partial f}{\partial \mathbf{x}}\right|^T_{\hat{\mathbf{x}}} + \Lambda. \qquad (30)$$

4. Selecting Informative Measurements

Marginal covariances are also useful to answer the following question, that is essential in reducing computational complexity of SLAM algorithms: Which measurements provide the most information about the state estimate? The answer certainly depends on the application requirements in terms of processing speed, and there is a tradeoff between omitting information and the quality of the estimation result. However, often there are also redundant or uninformative measurements that do not add any valuable information and can therefore safely be discarded. An answer to this question allows us to only use informative measurements in the estimation process, reducing computational complexity. It can also be used to guide the search for measurements, as exploited by Davison [20] for active search, but that is not the goal of our work. In this section we discuss the theory of how to determine the information that a measurement contributes following the work by Davison [20], but diverging in important points.

Let us start by identifying from a set of measurements the single measurement that, if applied, results in the state estimate with lowest uncertainty, or in other words, results in the highest gain in information. The current state estimate is given by the probability distribution $P(\mathbf{x}|Z^-)$ over the state $\mathbf{x}$ given all previous measurements Z^-. The posterior $P(\mathbf{x}|\mathbf{z}_j, Z^-)$ provides the distribution over the state $\mathbf{x}$ after applying a measurement $\mathbf{z}_j$ on landmark j. The measurement $\mathbf{z}_j$ follows the distribution $\mathbf{z}_j = h_{ij}(\mathbf{x}) + \nu$

from (8). Note our new notation $\mathbf{z}_j$: We do not use a specific measurement, but rather the expected measurement $\hat{\mathbf{z}}_j = h_{ij}(\hat{\mathbf{x}}) + \nu$ for a *specific landmark* j given the state estimate $\hat{\mathbf{x}}$, as we want to identify the measurement that is expected to yield the largest reduction in uncertainty *before* the measurement is actually made. The expected reduction in uncertainty or gain in information about the state $\mathbf{x}$ when making measurement $\mathbf{z}_j$ is given by the mutual information [21]

$$\begin{aligned} I(\mathbf{x};\mathbf{z}_j) &:= H(\mathbf{x}) - H(\mathbf{x}|\mathbf{z}_j) \\ &= \int_{\mathbf{x},\mathbf{z}_j} P(\mathbf{x},\mathbf{z}_j) \log \frac{P(\mathbf{x},\mathbf{z}_j)}{P(\mathbf{x})P(\mathbf{z}_j)}, \end{aligned} \tag{31}$$

where $H(\mathbf{x})$ is the entropy of the current state and $H(\mathbf{x}|\mathbf{z}_j)$ is the conditional entropy of the state after applying the measurement.

As iSAM represents Gaussian distributions, we obtain an expression for mutual information for this special case. We start with a random variable $\mathbf{a}$ with Gaussian distribution

$$P(\mathbf{a}) = \frac{1}{\sqrt{|2\pi\Sigma_\mathbf{a}|}} e^{-\frac{1}{2}\|\mathbf{a}-\hat{\mathbf{a}}\|^2_{\Sigma_\mathbf{a}}} \tag{32}$$

that is partitioned into two random variables $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ so that the joint mean $\hat{\mathbf{a}}$ and covariance $\Sigma_\mathbf{a}$ are given by

$$\hat{\mathbf{a}} = \begin{pmatrix} \hat{\boldsymbol{\alpha}} \\ \hat{\boldsymbol{\beta}} \end{pmatrix}, \quad \Sigma_\mathbf{a} = \begin{bmatrix} \Sigma_{\boldsymbol{\alpha}\boldsymbol{\alpha}} & \Sigma_{\boldsymbol{\alpha}\boldsymbol{\beta}} \\ \Sigma_{\boldsymbol{\beta}\boldsymbol{\alpha}} & \Sigma_{\boldsymbol{\beta}\boldsymbol{\beta}} \end{bmatrix} \tag{33}$$

It is shown by Davison [20] that the mutual information between the two Gaussian distributions $P(\boldsymbol{\alpha})$ and $P(\boldsymbol{\beta})$ is given by

$$\begin{aligned} I(\boldsymbol{\alpha};\boldsymbol{\beta}) &= E\left[\log_2 \frac{P(\boldsymbol{\alpha}|\boldsymbol{\beta})}{P(\boldsymbol{\alpha})}\right] \\ &= \frac{1}{2}\log_2 \frac{|\Sigma_{\boldsymbol{\alpha}\boldsymbol{\alpha}}|}{\left|\Sigma_{\boldsymbol{\alpha}\boldsymbol{\alpha}} - \Sigma_{\boldsymbol{\alpha}\boldsymbol{\beta}}\Sigma^{-1}_{\boldsymbol{\beta}\boldsymbol{\beta}}\Sigma_{\boldsymbol{\beta}\boldsymbol{\alpha}}\right|}. \end{aligned} \tag{34}$$

Note that the result is given in *bits* as we use the logarithm base 2.

To obtain the mutual information between the state space $\mathbf{x}$ and any of the N measurement functions $\mathbf{z}_j$ we combine these random variables into a new one

$$\mathbf{w} = (\mathbf{x}, \mathbf{z}_1, \ldots, \mathbf{z}_N)^T \tag{35}$$

that follows a Gaussian distribution with mean given by $\hat{\mathbf{w}} = (\hat{\mathbf{x}}, \hat{\mathbf{z}}_1, \ldots, \hat{\mathbf{z}}_N)^T$ and covariance

$$\Sigma_\mathbf{w} = \begin{bmatrix} \Sigma & \Sigma\frac{\partial h_1}{\partial \mathbf{x}}^T & \cdots & \Sigma\frac{\partial h_N}{\partial \mathbf{x}}^T \\ \frac{\partial h_1}{\partial \mathbf{x}}\Sigma & \frac{\partial h_1}{\partial \mathbf{x}}\Sigma\frac{\partial h_1}{\partial \mathbf{x}}^T + \Gamma_1 & \cdots & \frac{\partial h_1}{\partial \mathbf{x}}\Sigma\frac{\partial h_N}{\partial \mathbf{x}}^T \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial h_N}{\partial \mathbf{x}}\Sigma & \frac{\partial h_N}{\partial \mathbf{x}}\Sigma\frac{\partial h_1}{\partial \mathbf{x}}^T & \cdots & \frac{\partial h_N}{\partial \mathbf{x}}\Sigma\frac{\partial h_N}{\partial \mathbf{x}}^T + \Gamma_N \end{bmatrix} \tag{36}$$

where some of the terms were defined in (14) based on (13), and the remaining ones follow by analogy.

It is now easy to select the *best* measurement by calculating all $I(\mathbf{x};\mathbf{z}_j)$, but how do we treat the remaining measurements? We identify three different possibilities:

1. We calculate the mutual information between the state vector and all possible combinations of measurements. This is the correct solution as the measurements are not independent. However, unless the number of measurements is very small this solution is infeasible because the number of possible combinations 2^N is exponential in the number of landmarks N that can be measured.
2. We select the best measurement, then use the measurement to update the state space and start the feature selection again with the remaining measurements as done in [20]. Instead of taking the expected reduction in uncertainty into account, this solution uses the actual measurement to update the state space. However, the decision is sequential and therefore not guaranteed to be optimal. This solution is practical if updating the state space and recovering the necessary covariances are cheap operations.
3. We select the best measurement without updating the state space, and then ask the same question as before: Which of the remaining measurements is expected to yield the lowest uncertainty? This avoids the combinatorial complexity as well as updating of the state space with a measurement. While this solution is also not optimal, it is much cheaper than the other solutions, which is why we make use of this approach as described next.

For the third approach, one idea for finding the best measurement from the remaining ones is to look at the mutual information between measurements in order to decide which ones are redundant, as suggested in [20]. However, from these quantities we cannot directly calculate the correct information gains, as they do not consider mutual information between combinations of measurements.

To find the correct information gain for the third approach, we calculate the mutual information $I(\mathbf{x};\mathbf{z},\mathbf{z}_1)$ of the state space $\mathbf{x}$ and measurements $\mathbf{z}_1$ and $\mathbf{z}$ for each remaining measurement $\mathbf{z}$. After selecting a measurement $\mathbf{z}_2$ we continue with the mutual information $I(\mathbf{x};\mathbf{z},\mathbf{z}_1,\mathbf{z}_2)$ for the remaining measurements and so on. Our approach requires the same quantities as in [20], although the calculations get slightly more expensive as increasingly large blocks from the covariance matrix $\Sigma_\mathbf{w}$ are needed. However, the increase in cost is not significant because the size of the state space $\mathbf{x}$ is of the same order as the size of all measurements together. Note that our approach correctly takes care of redundant features.

We are not only interested in the order of the measurements in terms of their value for the estimation process, but also want to omit measurements that are not informative enough. For that purpose we ignore features that fall below a certain threshold, for example 2 bits.

Note that we need the same marginal covariances already required for joint compatibility in (27).

Table 1: Execution times for the Victoria Park sequence under different data association techniques. The number of measurements declines with increasing threshold on the minimum information a measurement has to provide, until data association eventually fails as shown in Fig. 8.

	Execution time	Number of measurements
NN	$207s$	3640
ML	$423s$	3640
JCBB	$590s$	3640
JCBB, 2 bit threshold	$588s$	3628
JCBB, 3 bit threshold	$493s$	2622
JCBB, 4 bit threshold	fails	-

5. Experiments and Results

We present timing results for recovering the exact marginal covariances. We analyze data association as well as the effect of measurement selection based on expected information gain for the well-known Sydney Victoria Park dataset. We also present results from loop closing for visual SLAM. All results are obtained on a Core 2 Duo 2.2GHz laptop computer.

5.1. Laser-Range-Based SLAM

To show the merits of JCBB, we use a simulated environment with 100 poses, 27 landmarks and 508 measurements in Fig. 7. The trajectory length is about $50m$. We added significant noise to both odometry and landmark measurements with all standard deviations $0.1m$ and $0.1rad$. Maximum likelihood data association fails to successfully close the loop after a wrong data association decision is made. JCBB on the other hand successfully establishes the correct correspondences, because it takes correlations between landmarks into account.

For evaluation of both marginal covariance recovery and information gain thresholding we use the standard Sydney Victoria Park dataset, see Figure 8(a). The dataset consists of laser-range data and vehicle odometry, recorded in a park with sparse tree coverage. It contains 7247 frames along a trajectory of 4 kilometer length, recorded over a time frame of 26 minutes. As repeated measurements taken by a stopped vehicle do not add any new information, we have dropped these, leaving 6969 frames. We have extracted 3640 measurements from the laser data by a simple tree detector.

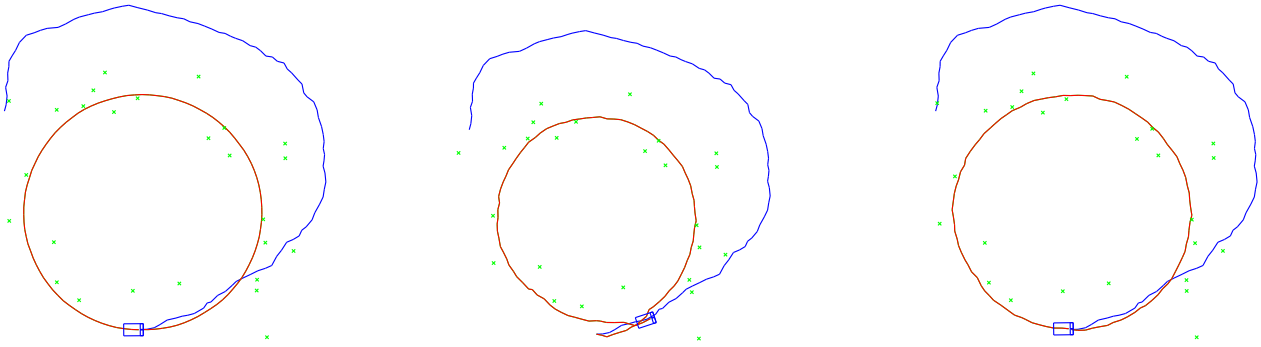
Table 1 compares execution times for different data association techniques applied to the Victoria Park sequence. The results for NN data association do not include recovery of marginal covariances and therefore represent the computation time of the underlying iSAM estimation algorithm [9]. While additionally recovering the exact marginal covariances and performing JCBB data association in every single step nearly triples the execution time, the performance still exceeds the requirements for real-time performance by a factor of 2.5.

Also shown in Table 1 are results for omitting uninformative measurements when performing JCBB. A threshold of 2 bit only removes a few measurements yielding only a slightly lower execution time. The result also shows that the selection of measurements does not add a significant overhead as the marginal covariances are already recovered for JCBB data association. For a 3 bit threshold the number of measurements is significantly lower, resulting in a significant speedup due to lower complexity of both the estimation and the marginal covariance recovery. Finally, for a 4 bit threshold too many measurements are removed and the data association fails to close a loop correctly, leading to an inconsistent map as shown in Fig. 8(d). Note that we periodically relinearize the system in iSAM to keep linearization errors small. The failure therefore arises because JCBB identifies a wrong match based on high motion uncertainty, which is aided by the relative sparsity of landmarks in this dataset.

Marginal covariance recovery is quite efficient with our dynamic algorithm. In Fig. 9(a) we compare the number of entries of the covariance matrix that have to be recovered with the number of actually required entries for JCBB data association. The figure shows both linear (top) and log scale (bottom). The number of recovered entries is much lower than the actual number of non-zero entries in the square root information matrix because our dynamic algorithm only calculates the entries that are actually needed. If, lets say no variables from the left half of the covariance matrix are needed, then the entries corresponding to non-zero entries in the left half of R also do not have to be calculated. Furthermore, from the remaining part of the matrix not all entries corresponding to non-zeros in the square root information matrix are necessarily required, as some variables might not depend on others even if those are further to the right.

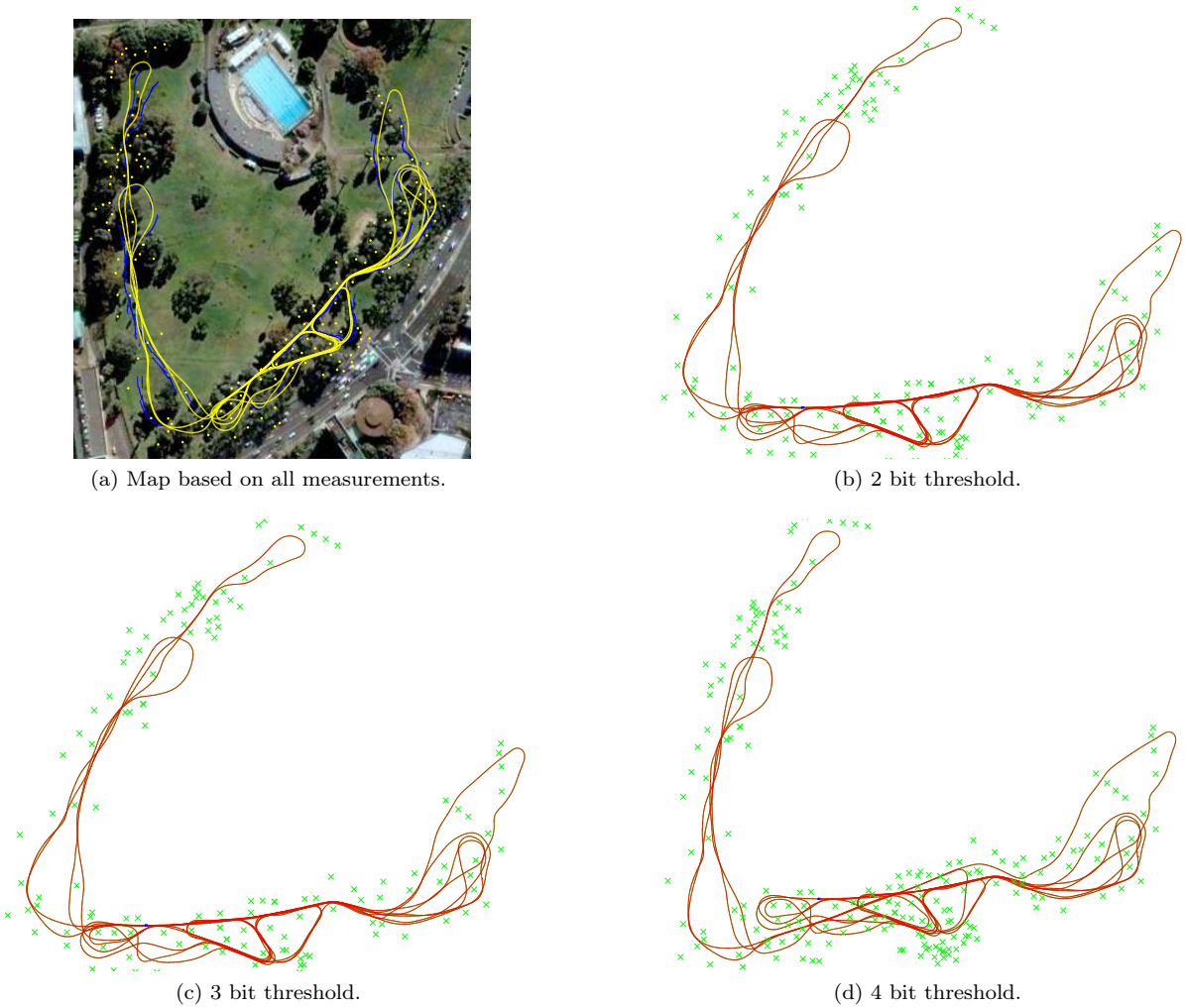
The number of entries calculated by the dynamic approach stays almost linear in this example. This really depends on the order of the variables in the square root information matrix, as it is more expensive to obtain covariances for entries that are further to the left side of the matrix. One can imagine modifying the variable ordering to take this into account, while only marginally increasing the fill-in of the square root information matrix. The spikes in Fig. 9 often coincide with an increased number of non-zero entries in the square root information matrix, which are caused by incremental updates during loop closing events. The significant increase on the right is a combination of a denser square root information matrix (see spikes in blue curve) with required variables being further to the left of the matrix and additionally more entries being requested (see green curve) as more landmarks are visible.

The timing results in Fig. 9(b) show that our algorithm outperforms other covariance recovery methods. In partic-



(a) Ground truth. (b) ML data association fails. (c) JCBB succeeds despite high noise.

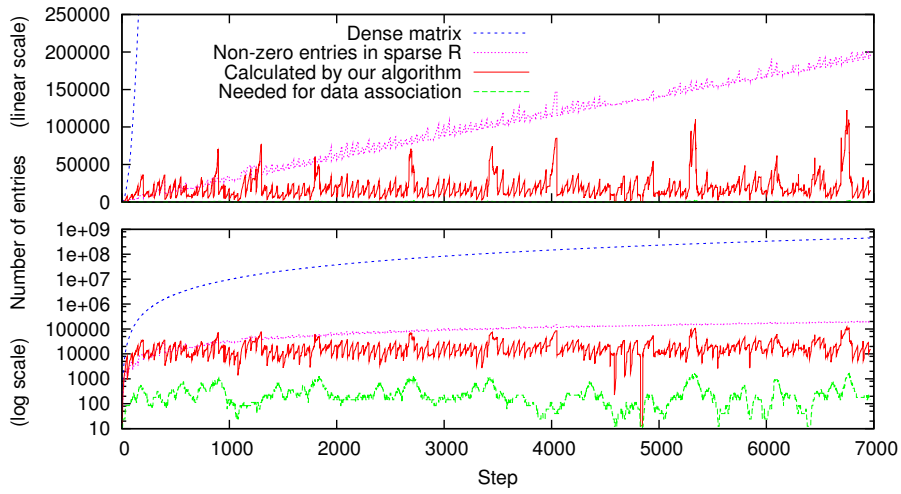
Figure 7: A simulated loop with high noise that requires the JCBB algorithm for successful data association. The red line is the estimated robot trajectory, the blue line the odometry and the green crosses are the landmarks.



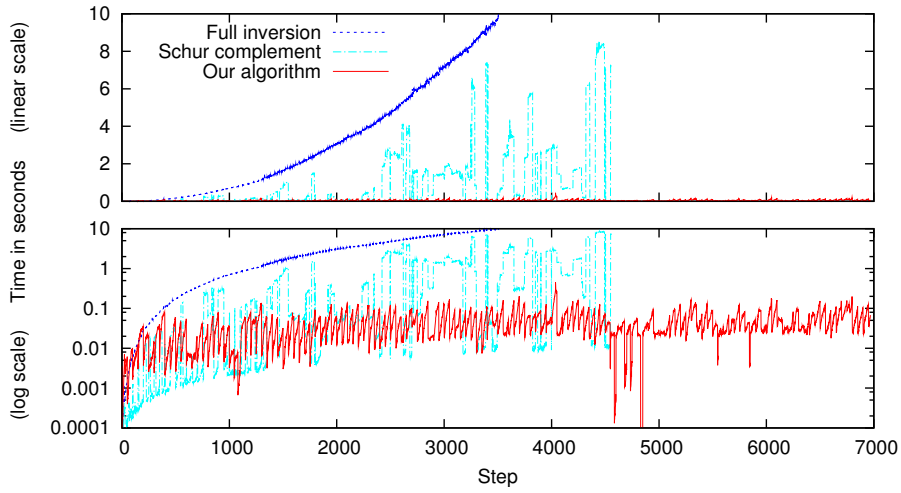
(a) Map based on all measurements. (b) 2 bit threshold.

(c) 3 bit threshold. (d) 4 bit threshold.

Figure 8: Maps for the Victoria Park sequence. (a) Based on all measurements, the trajectory and landmarks are shown in yellow (light), manually overlaid on an aerial image for reference. Differential GPS was not used in obtaining the experimental results, but is shown in blue (dark) for comparison - note that in many places GPS was not available, presumably due to obstruction by trees. (b)(c)(d) Maps resulting from omitting measurements with expected information below a threshold according to Table 1. For 2 and 3 bit thresholds the maps are visually identical. For a 4 bit threshold data association fails in the rightmost part of the trajectory, yielding an inconsistent map.



(a)



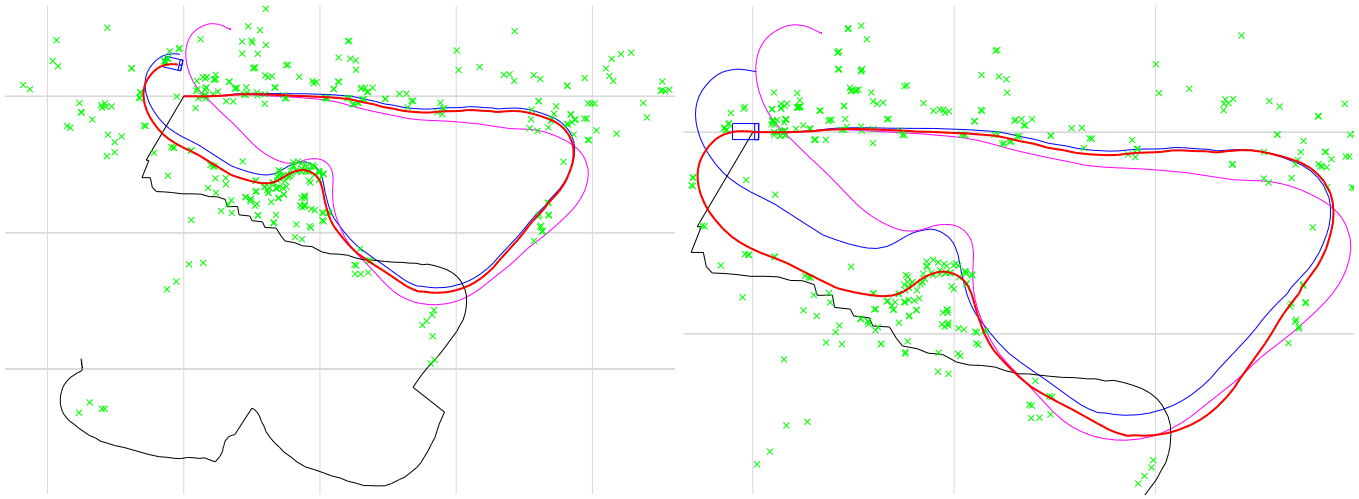
(b)

Figure 9: a) Number of entries of the covariance matrix that have to be recovered compared with the number actually required for data association, both in linear and log scale. For reference we also show the number of entries in R and the number of entries if R was dense. Note that our dynamic programming algorithm calculates a significantly lower number of entries than there are non-zero entries in R. b) Timing for our algorithm in linear and log scale. For reference we also show timing for full covariance matrix recovery using efficient sparse LDL matrix factorization as well as for only recovering a dense sub-matrix using the Schur complement (cut off after step 4500).

ular, we compare with full covariance recovery based on a very efficient sparse LDL matrix factorization [22]. Often the required entries are not distributed across the complete matrix, and only a dense sub-block of the covariance matrix is needed. This block can be recovered by a Schur complement [1], which we also implemented based on the LDL factorization. The Schur complement could be sped up by ordering visible landmarks further to the right side of the matrix, however, this in turn will generate more fill-in in the factorization, making both SLAM and the Schur complement more expensive. Our algorithm recovers the results within the real-time constraints of the data, even towards the end of the sequence, while the other algorithms became prohibitively expensive and have been stopped before the end of the sequence.

5.2. Visual SLAM

We evaluate data association for visual SLAM on data recorded during the final DARPA LAGR program demo in San Antonio, Texas. We used one of the stereo camera pairs on the LAGR platform shown in Fig. 10(a). We drove the robot for about $80m$ through the challenging environment shown in Fig. 10(b). Note that visual odometry was running in real-time on the robot (Core Duo 2GHz) and therefore only the resulting feature tracks are available. In particular, visual odometry failed over a number of frames, and only vehicle odometry is available in those places. This particular dataset was chosen for this exper-



(a) (b)

Figure 11: San Antonio sequence just before (a) and after (b) loop closing using JCBB. The vehicle internal trajectory estimate with GPS is shown in black, without GPS in magenta. The trajectory according to our visual odometry is shown in blue and the result of our visual SLAM algorithm in red. The robot is shown as rectangular outline and the gray lines represent a grid with $10m$ spacing. Note that GPS drifted before the vehicle started to move.

iment because of its bad quality. Even after extending visual odometry tracks using JCBB locally, the loop remains far from closed as shown in Figure 11(left).

Despite the large error at the end of the loop, JCBB succeeds in closing the large loop, with the result shown in Figure 11(right). Note that for these experiments we have hard coded at which frame JCBB is used to close the loop. However, JCBB successfully found a jointly compatible set of assignments that closes the loop. The results show how JCBB with our marginal covariance recovery algorithm allows for data association even under such difficult circumstances with non-descriptive features.

To show real-time performance, we have run JCBB data association on top of iSAM live on the DARPA LAGR platform with results shown in Fig. 12. The trajectory length is about $70m$ and includes a place where the robot got stuck and had to back up with significant wheel slippage. Note that the vehicle's internal state (IMU+odometry) is significantly off, while visual odometry provided a much better estimate. JCBB based on iSAM closed the loop, however, another relinearization step by iSAM would have been necessary for the trajectory to completely converge.

6. Discussion: JCBB versus RANSAC

RANSAC by Fischler and Bolles [11] is a well known algorithm suitable for data association, which does not rely on marginal covariances, so why should we not simply use RANSAC instead of JCBB? As the answer to this question is not straightforward, we decided to include some of our insights into the differences here. A detailed discussion requires several paragraphs, but the short answer is that there is typically no disadvantage to using JCBB, and it becomes necessary for data association under high uncertainty, such as large loop closings, where RANSAC fails.

RANSAC and JCBB take widely different approaches to the problem of data association. RANSAC is a probabilistic algorithm that repeatedly samples a minimum number of candidates needed to constrain the given problem and then determines their support based on the remaining candidates. The candidates are chosen from a set of putative assignments, and candidates that agree with the sampled model are called inliers. The model with the highest number of inliers is selected. The number of samples needed is adaptively determined in order to achieve a user specified confidence that the correct result is found.

The main difference between the two algorithms is that RANSAC has to be supplied with a set of putative assignments, while JCBB can explore the complete space of possible assignments. Therefore, for RANSAC, a part of the data association problem is already solved in the preprocessing step, and the whole process fails if the preprocessing does not include a sufficient number of correct assignments. This happens for example when closing large loops where the uncertainty becomes very large. Putatives are typically selected individually, and therefore their correlation is not taken into account and rather a locally optimal decision is made for each putative, for example based on distance.

JCBB in contrast is capable of exploring the complete space of possible assignments. While theoretically the same is possible with RANSAC by including all possible assignments in the set of putatives, it would require enumerating them and the probabilistic nature of RANSAC would not provide an optimal strategy to search this com-

(a)

(b)

Figure 10: (a) The DARPA LAGR mobile robot platform with two front-facing stereo camera rigs, a GPS receiver as well as wheel encoders and an inertial measurement unit (IMU). (b) Images taken along the robot path.

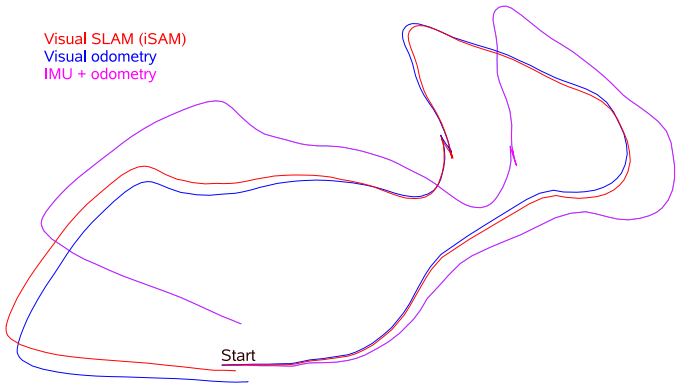


Figure 12: Robot trajectories from live demo of iSAM on the LAGR platform.

plete set. Actually, RANSAC is known to perform very poorly when only a small ratio of the putatives are inliers, as in that case many samples are required to reach a certain confidence level. JCBB on the other hand systematically explores that space and additionally prunes large parts of the search space for efficiency. And while JCBB evaluates the joint compatibility between all assignments, RANSAC at most checks for joint compatibility between the minimum set and each remaining putative individually, which should perform somewhere between individual and joint compatibility.

Finally, JCBB is preferable over RANSAC even for a small set of putatives with high inlier ratio. For some applications the set of putatives is of high quality, for example when they are based on very informative feature descriptors such as SIFT. However, while RANSAC only provides a confidence for the result, i.e. there remains a small probability that it does not find the correct solution, JCBB instead performs an exhaustive, yet efficient search. And if we have only one putative per landmark, the search tree contains only binary decisions making JCBB efficient in combination with pruning of subtrees.

Of course, as JCBB solves a NP hard problem exactly we cannot expect it to work for very large problems. However, one can typically restrict the problem to a manageable size, as we have done in the visual SLAM case. Alternatively, one can reduce the problem space for example by using informative feature descriptors as discussed above. And if the problem still remains too complex to be solved by JCBB within real-time constraints, then it is likely that either the correct solution is not contained in the putatives (otherwise we could use that small set of putatives as input to JCBB!), or we have such a low inlier ratio that we have to terminate RANSAC after a maximum number of iterations, thereby not obtaining the correct result with the targeted confidence.

7. Conclusion

We presented a dynamic programming algorithm for efficient recovery of the marginal covariances needed for data association from a square root information matrix. The square root information matrix was obtained by the incremental smoothing and mapping (iSAM) algorithm, but could for other applications also be obtained by batch matrix factorization, fixed-lag smoothing or from a filtering approach. The marginal covariances can efficiently be obtained as long as the necessary variables for data association are included and the square root information matrix is sparse. We used our algorithm to obtain the exact marginal covariances required to perform JCBB data association, thereby eliminating the need for conservative estimates. We also showed that the same quantities allow for information theoretic decisions that allow for example to reduce computational complexity by omitting uninformative measurements.

While JCBB performed well with simple point features for visual SLAM, in practice one would want to include appearance to further simplify the problem. There are also some practical issues, such as how to deal with situation in which only one landmark is visible. In that case joint compatibility reduces to individual compatibility, and wrong assignments are easily accepted. And for very large scale problems the estimation problem will have to be split into submaps, leading to the question of how to perform data association if more than one submap has to be considered.

Acknowledgments

We would like to thank U. Frese for helpful suggestions and the anonymous reviewers for their valuable comments. We also thank E. Nebot and H. Durrant-Whyte for the Victoria Park dataset. This work was partially supported by DARPA under grant number FA8650-04-C-7131.

References

[1] S. Thrun, W. Burgard, D. Fox, Probabilistic Robotics, The MIT press, Cambridge, MA, 2005. 1, 12

[2] J. Neira, J. Tardos, Data association in stochastic mapping using the joint compatibility test, IEEE Trans. Robot. Automat. 17 (6) (2001) 890–897. 1, 5, 6, 7

[3] T. Bailey, H. Durrant-Whyte, Simultaneous localisation and mapping (SLAM): Part II state of the art, Robotics & Automation Magazine. 1

[4] T. Duckett, S. Marsland, J. Shapiro, Fast, on-line learning of globally consistent maps, Autonomous Robots 12 (3) (2002) 287–300. 1

[5] M. Bosse, P. Newman, J. Leonard, S. Teller, Simultaneous localization and map building in large-scale cyclic environments using the Atlas framework, Intl. J. of Robotics Research 23 (12) (2004) 1113–1139. 1

[6] R. Eustice, H. Singh, J. Leonard, M. Walter, R. Ballard, Visually navigating the RMS titanic with SLAM information filters, in: Robotics: Science and Systems (RSS), 2005. 1

[7] E. Olson, J. Leonard, S. Teller, Spatially-adaptive learning rates for online incremental SLAM, in: Robotics: Science and Systems (RSS), 2007. 1

[8] R. Eustice, H. Singh, J. Leonard, M. Walter, Visually mapping the RMS Titanic: Conservative covariance estimates for SLAM information filters, Intl. J. of Robotics Research 25 (12) (2006) 1223–1242. 1

[9] M. Kaess, A. Ranganathan, F. Dellaert, iSAM: Incremental smoothing and mapping, IEEE Trans. Robotics 24 (6) (2008) 1365–1378. 1, 2, 3, 5, 10

[10] M. Kaess, A. Ranganathan, F. Dellaert, iSAM: Fast incremental smoothing and mapping with efficient data association, in: IEEE Intl. Conf. on Robotics and Automation (ICRA), Rome, Italy, 2007, pp. 1670–1677. 2

[11] M. Fischler, R. Bolles, Random sample consensus: a paradigm for model fitting with application to image analysis and automated cartography, Commun. Assoc. Comp. Mach. 24 (1981) 381–395. 2, 13

[12] F. Dellaert, M. Kaess, Square Root SAM: Simultaneous localization and mapping via square root information smoothing, Intl. J. of Robotics Research 25 (12) (2006) 1181–1203. 2

[13] G. Golub, R. Plemmons, Large-scale geodetic least-squares adjustment by dissection and orthogonal decomposition, Linear Algebra and Its Applications 34 (1980) 3–28. 3, 6

[14] B. Triggs, P. McLauchlan, R. Hartley, A. Fitzgibbon, Bundle adjustment – a modern synthesis, in: W. Triggs, A. Zisserman, R. Szeliski (Eds.), Vision Algorithms: Theory and Practice, LNCS, Springer Verlag, 1999, pp. 298–375. 3

[15] R. Jonker, A. Volgenant, A shortest augmenting path algorithm for dense and sparse linear assignment problems, Computing 38 (4) (1987) 325–340. 5

[16] Y. Bar-Shalom, X. Li, Estimation and Tracking: principles, techniques and software, Artech House, Boston, London, 1993. 5

[17] W. H. Press, B. P. Flannery, S. A. Teukolsky, W. T. Vetterling, Numerical Recipes: The Art of Scientific Computing, 2nd Edition, Cambridge University Press, Cambridge (UK) and New York, 1992. 6

[18] F. Dellaert, Monte Carlo EM for data association and its applications in computer vision, Ph.D. thesis, School of Computer Science, Carnegie Mellon, also available as Technical Report CMU-CS-01-153 (September 2001). 6

[19] R. Mottaghi, M. Kaess, A. Ranganathan, R. Roberts, F. Dellaert, Place recognition-based fixed-lag smoothing for environments with unreliable GPS, in: IEEE Intl. Conf. on Robotics and Automation (ICRA), Pasadena, CA, 2008. 8

[20] A. Davison, Active search for real-time vision, in: Intl. Conf. on Computer Vision (ICCV), 2005. 8, 9

[21] T. Cover, J. Thomas, Elements of Information Theory, John Wiley & Sons, New York, NY, 1991. 9

[22] T. Davis, Algorithm 849: A concise sparse cholesky factorization package, ACM Trans. Math. Softw. 31 (4) (2005) 587–591. 12

Michael Kaess received the Ph.D. degree in computer science from the Georgia Institute of Technology, Atlanta, in 2008, where he also received the M.S. degree in computer science in 2002. He did the equivalent of the B.S. degree at the University of Karlsruhe, Germany.

Currently, he is a Postdoctoral Associate at the Massachusetts Institute of Technology, Cambridge MA. His research focuses on probabilistic methods in mobile robotics and computer vision, specifically for efficient localization and large-scale mapping as well as data association.

Frank Dellaert received the Ph.D. degree in computer science from Carnegie Mellon University in 2001, an M.S. degree in computer science and engineering from Case Western Reserve University in 1995, and the equivalent of an M.S. degree in electrical engineering from the Catholic University of Leuven, Belgium, in 1989.

He is currently an Associate Professor in the College of Computing at the Georgia Institute of Technology. His research focuses on probabilistic methods in robotics and vision.

Prof. Dellaert has published more than 60 articles in journals and refereed conference proceedings, as well as several book chapters. He also serves as Associate Editor for IEEE TPAMI.