

# Reduced State Representation in Delayed-State SLAM

Viorela Ila, Josep M. Porta, and Juan Andrade-Cetto

**Abstract**—This paper introduces an approach that reduces the size of the state and maximizes the sparsity of the information matrix in exactly sparse delayed-state SLAM. We propose constant time procedures to measure the distance between a given pair of poses, the mutual information gain for a given candidate link, and the joint marginals required for both measures. Using these measures, we can readily identify non redundant poses and highly informative links and use only those to augment and to update the state, respectively. The result is a delayed-state SLAM system that reduces both the use of memory and the execution time and that delays filter inconsistency by reducing the number of linearization introduced when adding new loop closure links. We evaluate the advantage of the proposed approach using simulations and data sets collected with real robots.

## I. INTRODUCTION

The first available solutions in the SLAM literature were based on the Extended Kalman Filter (EKF) and formulated the problem as the probabilistic estimation of the robot pose and the location of static landmarks in the environment, modelled as a multivariate Gaussian parametrized with the state mean and its covariance matrix [1], [2]. However, maintaining the covariance matrix has memory and time complexities that scale quadratically with the number of landmarks, which limits this approach to relatively small environments.

This computational cost can be alleviated using the Extended Information Filter (EIF) and its alternative parametrization of Gaussian distributions based on the information vector and the information matrix. The information matrix in landmark-based SLAM is approximately sparse with very small matrix entries for distant landmarks [3]. These entries can be removed, compacting the map and speeding up the filter. If instead of only estimating the last robot pose, the whole robot trajectory is included in the state together with the landmarks, an approach typically referred to as full SLAM [4]–[6], a sparse information matrix is obtained without using approximations. Furthermore, in delayed-state SLAM [7]–[10] only the trajectory of the robot is included in the state and the landmarks are only used to derive relative motion constraints between poses. The result is an exactly sparse information matrix which grows with the number of poses and that only has non-null entries for those poses directly related by an observation. Therefore,

delayed state SLAM only requires a moderate memory use even when mapping large areas.

However, adding all robot poses to the state vector has the cost of a representation that grows even when revisiting already traversed areas. Furthermore, adding all possible loop closure links reduces the sparsity of the information matrix, slows down the execution, and contributes to produce overconfident estimates due to the linearizations introduced with each new link, which in the long run lead to filter inconsistency [11], [12].

Heuristic strategies have been used in the delayed-state SLAM approach either to restrict the number of links [8] or to reduce the size of the state by keeping only poses every few meters [10]. In the context of landmark-based SLAM, principled information-based approaches have been proposed to reduce the state representation size and to delay inconsistency incorporating only highly informative observations to the filter [13], [14]. Following this line of thought, our previous work [9] pointed out that the computational complexity of delayed-state SLAM can be reduced by considering only highly informative links between nearby poses. The problem is that computing the distance between poses and the mutual information gain for links requires access to the joint marginals, which are computationally expensive to recover from the information parameterization. Therefore, in our previous work, we only provide approximations for these two measures. However, these approximations assume the poses to be independent which is not the case in delayed state SLAM. In this paper, we show that while in open loop, when the system searches for informative links, the joint marginal covariances of a given pair of poses can be recovered in constant time. Exploiting this contribution, we propose a method to compute the distance between poses and the mutual information gain for each link in exact form. Finally, we extend the use of the distance and the information measures to identify not only relevant links but also redundant poses that are not worth to be added to the state.

The rest of the paper is structured as follows. In Section II we formalize delayed-state SLAM and define the problems to be addressed. Then, Section III describes the strategy to control the size and the fill in of the information matrix and the metrics used to compute the informative load of links and poses. Section IV details how to recover the joint marginal covariance of a given pair of poses in constant time and Section V evaluates the method using synthetic and real data sets. Finally, Section VI summarizes the contributions of this work and points out ideas for further development.

The authors are with the Institut de Robòtica i Informàtica Industrial, CSIC-UPC, Llorens Artigas 4-6, Barcelona, 08028 Spain. {vila,porta,chetto}@iri.upc.edu

This work has been partially supported by the Spanish Ministry of Science and Innovation under a Juan de la Cierva Postdoctoral Fellowship to V. Ila and the projects DPI-2007-60858, DPI-2008-06022, MIPRCV Consolider-Ingenio 2010, and the EU URUS project IST-FP6-STREP-045062.

## II. DELAYED-STATE SLAM FORMULATION

In delayed-state SLAM, the trajectory of the robot at time  $t$ ,  $\mathbf{x}_t = \{x_0, \dots, x_t\}$ , with  $x_i$  the robot pose at time  $i$ ,  $i \leq t$ , is estimated from the trajectory at time  $t-1$ , a motion command  $u_t$ , and a set of observations,  $\mathbf{y}_t$ , of relative displacements between the current robot pose and previous poses along the path

$$p(\mathbf{x}_t | u_t, \mathbf{y}_t, \mathbf{x}_{t-1}) \propto p(\mathbf{y}_t | \mathbf{x}_t) p(\mathbf{x}_t | u_t, \mathbf{x}_{t-1}). \quad (1)$$

The estimation problem in (1) corresponds to the typical SLAM operations of augmenting the state when computing  $p(\mathbf{x}_t | u_t, \mathbf{x}_{t-1})$  and updating the robot path using relative observations when computing  $p(\mathbf{y}_t | \mathbf{x}_t)$ .

Assuming Gaussian distributions, the probabilities in (1) can be parametrized either in terms of their mean and covariance  $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$  or in terms of the information vector and the information matrix  $\mathbf{x} \sim \mathcal{N}^{-1}(\boldsymbol{\eta}, \boldsymbol{\Lambda})$ , where  $\boldsymbol{\eta} = \boldsymbol{\Lambda}\boldsymbol{\mu}$ ,  $\boldsymbol{\Lambda} = \boldsymbol{\Sigma}^{-1}$ , and the estimation workhorses are the extended Kalman and information filters, respectively. As mentioned, this second case is advantageous since its state representation is exactly sparse [8].

The motion  $u_t$  is typically given by odometric sensors and it can be integrated to the filter during state augmentation.

Augmenting the state in information form introduces shared information only between the new robot pose,  $x_t$ , and the previous one,  $x_{t-1}$ , resulting in a naturally sparse information matrix,  $\boldsymbol{\Lambda}$ , with a tridiagonal block structure [8]. Assuming the state mean to be available, this operation can be performed in constant time. In the incremental form of delayed-state SLAM, adding all the poses results in a state which increases in size even when re-traversing areas. To avoid this unnecessary growth of the state size, redundant poses must be identified and not added.

Integrating the observations in  $\mathbf{y}_t$  into the filter is more problematic since it involves solving the *data association* problem. In the context of delayed-state SLAM, data association is the process of determining relative displacements between the current robot pose and previous poses in the trajectory by registering the corresponding sensor readings.

When establishing a link between pose  $t$  and pose  $i$  using the information filter, the update operation only modifies the diagonal blocks  $t$  and  $i$  of the sparse information matrix,  $\boldsymbol{\Lambda}$ , and introduces new off-diagonal blocks at locations  $t, i$  and  $i, t$ . This operation can also be executed in constant time, assuming the state mean to be available [8]. When establishing a link between an uncertain pose and a more certain one, the estimation of the entire path is revised reducing the accumulated error. But recovering the state after an update has worst case quadratic cost, dwindling the advantage of constant time predictions and updates in the information form of delayed-state SLAM. If only few loops are closed, this high cost is amortized over long periods. For this reason there must be a trade off between reducing uncertainty and keeping the number of links low.

## III. REDUCING THE STATE REPRESENTATION IN DELAYED-STATE SLAM

The strategy we propose to maintain a reduced state size and to speed up the execution of delayed-state SLAM is based on considering only highly informative links and non redundant poses.

The mutual information gain of a link measures the contribution of the link to the correction of the state. For low informative links, the state hardly changes and the linearization effects associated with the link dominate. On the contrary, for highly informative links, the linearization effects are small in comparison with the change in the state. Thus, we propose to measure the information gain for the links and to update the state using only those informative enough.

As far as poses are concerned, it seems reasonable to keep only non-redundant poses. The redundancy of a pose should be measured in the information space and not only in the Euclidean space, as it has been previously done in the literature [10]. The distance in the information space of a pose with respect to the poses already in the state can be measured from the information carried by the links established between them. If these links are not informative enough, there is no need to include the new pose in the state representation since its contribution is equivalent to poses already in the state.

The mutual information gain can be computed by actually checking the contribution of all possible observations obtained from sensor registration. If at least one of the links is very informative, both the link and the new pose are added to the state. On the other hand, if no much information is gained for any link, the pose is considered redundant and it is not added to the state.

However, the process of registering the current sensor reading with all the previous ones is slow and prone to perceptual aliasing. Since only poses that are within the sensor range are likely to be successfully registered with the current pose, we limit sensor registration only to the set of neighboring poses. Moreover, we can evaluate the expected mutual information gain before actually registering the sensor readings. Thus, we can further constrain the set of poses to be considered to those that are close enough and whose expected information gain when linked to the current pose is potentially high. When using the expected information instead of the actual one, we can keep not only the poses that close a loop but also those that are good candidate to do so.

The use of the information-based selection criterion as a previous step to sensor registration leads to restrictive data association tests that avoid computational expensive sensorial matching as much as possible. The result in a reduced state representation with the corresponding computational savings.

### A. Distance Between Poses

The relative displacement,  $d$ , from the current robot pose  $x_t \sim \mathcal{N}(\boldsymbol{\mu}_t, \boldsymbol{\Sigma}_{tt})$  to any other previous pose in the trajectory

$x_i \sim \mathcal{N}(\mu_i, \Sigma_{ii})$  can be estimated as a Gaussian with parameters

$$\mu_d = h(\mu_t, \mu_i),$$

$$\Sigma_d = [\mathbf{H}_i \ \mathbf{H}_t] \begin{bmatrix} \Sigma_{ii} & \Sigma_{it} \\ \Sigma_{it}^\top & \Sigma_{tt} \end{bmatrix} [\mathbf{H}_i \ \mathbf{H}_t]^\top$$

where  $h(x_t, x_i)$  gives the relative displacement from  $x_i$  to  $x_t$  in the frame of reference of  $x_i$ ,  $\mathbf{H}_t$  and  $\mathbf{H}_i$  are the Jacobians of  $h$  with respect to the two poses, and  $\Sigma_{it}$  is the cross correlation between poses  $i$  and  $t$ .

Marginalizing the distribution of the displacement,  $d$ , for each one of its dimensions,  $r$ , we get a one-dimensional Gaussian distribution  $\mathcal{N}(\mu_r, \sigma_r^2)$  that allows to compute the probability of pose  $x_i$  being closer than  $v_r$  to pose  $x_t$  along such dimension

$$p_r = \int_{-v_r}^{+v_r} \mathcal{N}(\mu_r, \sigma_r^2)$$

$$= \frac{1}{2} \left( \operatorname{erf} \left( \frac{v_r - \mu_r}{\sigma_r \sqrt{2}} \right) - \operatorname{erf} \left( \frac{-v_r - \mu_r}{\sigma_r \sqrt{2}} \right) \right). \quad (2)$$

If for all dimensions,  $p_r$  is above a given threshold  $s$ , then pose  $x_i$  is considered to be close enough to the current robot pose,  $x_t$ .

If we set  $v_r = 0$  for all dimensions, then  $p_r$  gives the probability of the displacement  $d$  to be null. In [9] we approximated this probability using a Mahalanobis distance and assumed independence between the two involved poses. We will see in Section IV, that the marginal covariances needed to build  $\Sigma_d$  can be recovered efficiently, allowing to exactly compute this distance in constant time for any given pair of poses.

### B. Mutual Information Gain of Candidate Links

When integrating the information of a new link between poses  $i$  and  $t$  into the state, the information matrix posterior is given by

$$\Lambda' = \Lambda + \mathbf{H}^\top \Sigma_y^{-1} \mathbf{H} \quad (3)$$

where  $\Sigma_y$  is the expected measurement noise covariance and  $\mathbf{H}$  the corresponding Jacobian

$$\mathbf{H} = [\mathbf{0} \ \dots \ \mathbf{0} \ \mathbf{H}_i \ \mathbf{0} \ \dots \ \mathbf{0} \ \mathbf{H}_t]. \quad (4)$$

For Gaussian distributions the mutual information gain produced by the candidate link is computed as [15]

$$\mathcal{I}_{it} = \frac{1}{2} \ln \frac{|\Lambda'|}{|\Lambda|}. \quad (5)$$

If (5) is above a given threshold,  $g$ , the link between the two poses is considered relevant enough to reduce the state uncertainty, and the corresponding observations are registered. When the registration succeeds we obtain a measure of the displacement between poses  $i$  and  $t$  that can be used to update the filter.

A straightforward evaluation of the global entropy reduction in (5) is computationally expensive since it requires the computation of the determinants of the prior and all possible posterior information matrices resulting from each tentative

updates. Therefore, in [9] we heuristically approximated (5) using the second term of the Bhattacharyya distance that provides a test to distinguish classes with close means but different covariances, i.e., to identify poses that are close but have significantly different uncertainty and whose linkage might result in a significant change in the state. However, this test assumes the poses to be independent and only considers the effect of the candidate link on the two directly linked poses, whereas when closing a loop, the whole state potentially changes. The test in (5) takes into account the effect of the candidate link in all the poses in the state and considers the cross-correlations between them. In the following we show that algebraic manipulation allows exact, constant-time computation of the information gain.

Using (3), (5) becomes

$$\mathcal{I}_{it} = \frac{1}{2} \ln \frac{|\Lambda + \mathbf{H}^\top \Sigma_y^{-1} \mathbf{H}|}{|\Lambda|}$$

$$= \frac{1}{2} \ln \frac{|\Sigma_y| |\Lambda + \mathbf{H}^\top \Sigma_y^{-1} \mathbf{H}|}{|\Sigma_y| |\Lambda|}.$$

Since the determinant of block matrix can be defined as

$$\begin{vmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{vmatrix} = |\mathbf{A}| |\mathbf{D} - \mathbf{C} \mathbf{A}^{-1} \mathbf{B}| = |\mathbf{D}| |\mathbf{A} - \mathbf{B} \mathbf{D}^{-1} \mathbf{C}|$$

we have that

$$\mathcal{I}_{it} = \frac{1}{2} \ln \frac{\begin{vmatrix} \Sigma_y & -\mathbf{H} \\ \mathbf{H}^\top & \Lambda \end{vmatrix}}{|\Sigma_y| |\Lambda|}$$

$$= \frac{1}{2} \ln \frac{|\Lambda| |\Sigma_y + \mathbf{H} \Lambda^{-1} \mathbf{H}^\top|}{|\Sigma_y| |\Lambda|}$$

$$= \frac{1}{2} \ln \frac{|\Sigma_y + \mathbf{H} \Lambda^{-1} \mathbf{H}^\top|}{|\Sigma_y|}$$

$$= \frac{1}{2} \ln \frac{|\Sigma_y + \mathbf{H} \Sigma \mathbf{H}^\top|}{|\Sigma_y|}$$

$$= \frac{1}{2} \ln \frac{|\mathbf{S}|}{|\Sigma_y|} \quad (6)$$

with  $\mathbf{S} = \Sigma_y + \mathbf{H} \Sigma \mathbf{H}^\top$  the Kalman innovation matrix that, taking into account (4), is

$$\mathbf{S} = \Sigma_y + [\mathbf{H}_i \ \mathbf{H}_t] \begin{bmatrix} \Sigma_{ii} & \Sigma_{it} \\ \Sigma_{it}^\top & \Sigma_{tt} \end{bmatrix} [\mathbf{H}_i \ \mathbf{H}_t]^\top.$$

The matrices involved in the final expression for  $\mathcal{I}_{it}$  have constant size; the size of the underlying pose space. Therefore, the mutual information gain can be computed in constant time independent of the size of the state, assuming the joint marginal covariance for poses  $i$  and  $t$  to be available to compute  $\mathbf{S}$ .

### IV. COMPUTATION OF JOINT MARGINALS

The proposed solution to reduce the state size relies on an efficient computation of both, the distance in (2) and the mutual information gain in (6). The measures require the state mean and the joint-marginal between the current pose,  $x_t$ , and any pose from the history,  $x_i$ . Although representing the state in information form is more memory efficient than

in covariance form, it does not offer direct access to state means and marginal covariances.

Efficient approximations of cross covariances can be computed in logarithmic time by subsampling poses and performing relaxation over multiple spatial resolutions [16], or in constant time by considering only first order relations via Markov blankets [3] or by implementing partial state updates [17]. Optimistic approximations of joint marginals increase the number of data association candidates, something that is especially sensitive after long periods of open loop traverse. Thus, exact joint marginal covariances are preferred for the accurate identification of neighboring poses. Exact joint marginals can be recovered by augmenting the sparse system of equations needed for state recovery [8] or by exploiting the sparseness of factorized forms of the information matrix with QR [6] or Cholesky factorizations [18]. These algorithms have in average linear computational complexity for band diagonal matrices, but they are worst case quadratic for matrices encoding many loops.

Capitalizing on the idea that only few links are added to the state representation, we now show how exact computation of joint marginals can be achieved in constant time during open loop.

Suppose a loop closure occurs at time  $l$ . At that point, and thanks to the sparsity of the information matrix, the state mean,  $\mu$ , and the covariance matrix,  $\Sigma$ , can be recovered either by QR or Cholesky factorization [6], [18]. In our implementation, we make use of supernodal sparse Cholesky factorization [19]. Finally, the exact marginal covariances  $\Sigma_{ii}$ , and the cross covariances  $\Sigma_{il}$  with  $1 \leq i \leq l$ , can be readily extracted from  $\Sigma$ .

After loop closure, when the robot moves to a new pose,  $x_i, i > l$ , the mean for this new pose can be computed from the previous robot pose,  $x_{i-1}$ , and the motion  $u_t$

$$\mu_i = f(\mu_{i-1}, u_t)$$

and its marginal covariance can be computed as

$$\Sigma_{ii} = \mathbf{F}_i \Sigma_{i-1} \mathbf{F}_i^\top + \mathbf{W}_i \Sigma_u \mathbf{W}_i^\top$$

with  $\mathbf{F}_i$  and  $\mathbf{W}_i$  the Jacobians of  $f$  with respect to  $x_{i-1}$  and  $u_t$ , respectively, and  $\Sigma_u$  the motion noise. These marginal covariances can be computed once and stored since they do not change until the next loop closure occurs.

The cross correlation between the last robot pose and the previously stored ones can be factorized as

$$\Sigma_{it} = \Phi_i \mathbf{F}^\top$$

with

$$\Phi_i = \begin{cases} \Sigma_{il} & i \leq l \\ \Sigma_{ii} (\mathbf{F}_{l+1}^\top \dots \mathbf{F}_i^\top)^{-1} & i > l \end{cases}$$

and where  $\mathbf{F}^\top = \mathbf{F}_{l+1}^\top \dots \mathbf{F}_t^\top$  is the accumulated Jacobian from the last loop closure to the current time slice. Observe that  $\mathbf{F}$  can be updated in constant time as the robot moves. Moreover, all the information needed to define  $\Phi_i$  is available at time slice  $i$  and can be computed in constant time since the term  $(\mathbf{F}_{l+1}^\top \dots \mathbf{F}_i^\top)^{-1}$  is the inverse of the aggregated Jacobian,  $\mathbf{F}^\top$ , at time  $i$ .

## V. EXPERIMENTS AND RESULTS

Fig. 1 shows a comparison of different strategies for delayed-state SLAM, illustrating the effect of adding only informative links and non-redundant poses. All three plots simulate a robot circling a 6 m radius circle and then circumscribing an ellipse with semiaxes 6 m and 8 m. The simulated robot has an odometric sensor whose error is 5% of the displacement in  $x$  and  $y$  and 0.00175 rad in orientation and a second sensor able to link any two poses closer than  $\pm 3$  m,  $\pm 3$  m,  $\pm 0.26$  rad, in  $x$ ,  $y$ , and orientation, respectively. This sensor has a noise covariance of  $\Sigma_y = \text{diag}(0.2^2, 0.2^2, 0.009^2)$ .

Fig. 1(a) shows the result of the simulation for the standard delayed-state SLAM that incorporates all possible poses and links to the state representation. The distance test in (2) with range  $v = (3, 3, 0.26)$  and threshold  $s = 0.1$  is used to determine neighboring poses using (2). The simulation takes about 32 s and at the end of the execution the filter includes 170 poses and 895 loop-closure links. Results correspond to a Matlab implementation running under Linux on a Intel Core 2 at 2.4 Ghz.

Fig. 1(b) shows the result of the same simulation but including only links that have a value of (6) higher than 1. In this simulation, only 36 loop-closure links are established instead of 895 and this results in a significant saving in the execution time (25 s vs. 32 s).

Finally, Fig. 1(c) shows the outcome of the experiment where only informative links and relevant poses are added to the state. As discussed in Sec. III, a pose is considered relevant only if it establishes at least one sufficiently informative link with any other pose already in the state. In this case, the simulation takes only 9 s and the resulting state includes 49 poses and 34 loop-closure links. We can conclude that the approach introduced in this paper significantly reduces the state size and, consequently, the computation time. As it can be appreciated in Fig. 1(c), since only low informative links and redundant poses are discarded, the final trajectory estimation is equivalent to that estimated by the original delayed-state SLAM approach in Fig. 1(a).

To test the proposed system on real data, a Segway robotic platform was manually driven to collect dead reckoning readings and stereo images with a Bumblebee2 stereo rig. The images are used to find constraints between the corresponding poses triangulating SIFT features [20].

The Segway dead reckoning readings and the visual pose constraints are modeled with noise covariances  $\Sigma_u = \text{diag}(0.01^2, 0.005^2, 0.03^2)$ , and  $\Sigma_y = \text{diag}(0.2^2, 0.2^2, 0.03^2)$ , respectively, and the uncertainty of the initial pose is set to  $\Sigma_{11} = \text{diag}(0.1^2, 0.1^2, 0.09^2)$ .

Experimentally, we observed that images taken in poses farther away than  $\pm 2.5$  m in  $x$ ,  $\pm 2$  m in  $y$  or  $\pm 0.26$  rad in orientation can not be safely matched and, consequently, those are the thresholds used to detect nearby poses using (2) with a neighboring probability threshold  $s = 0.1$ .

We drove the robot for 700 s for about 400 m along two loops around a couple of buildings in the Barcelona



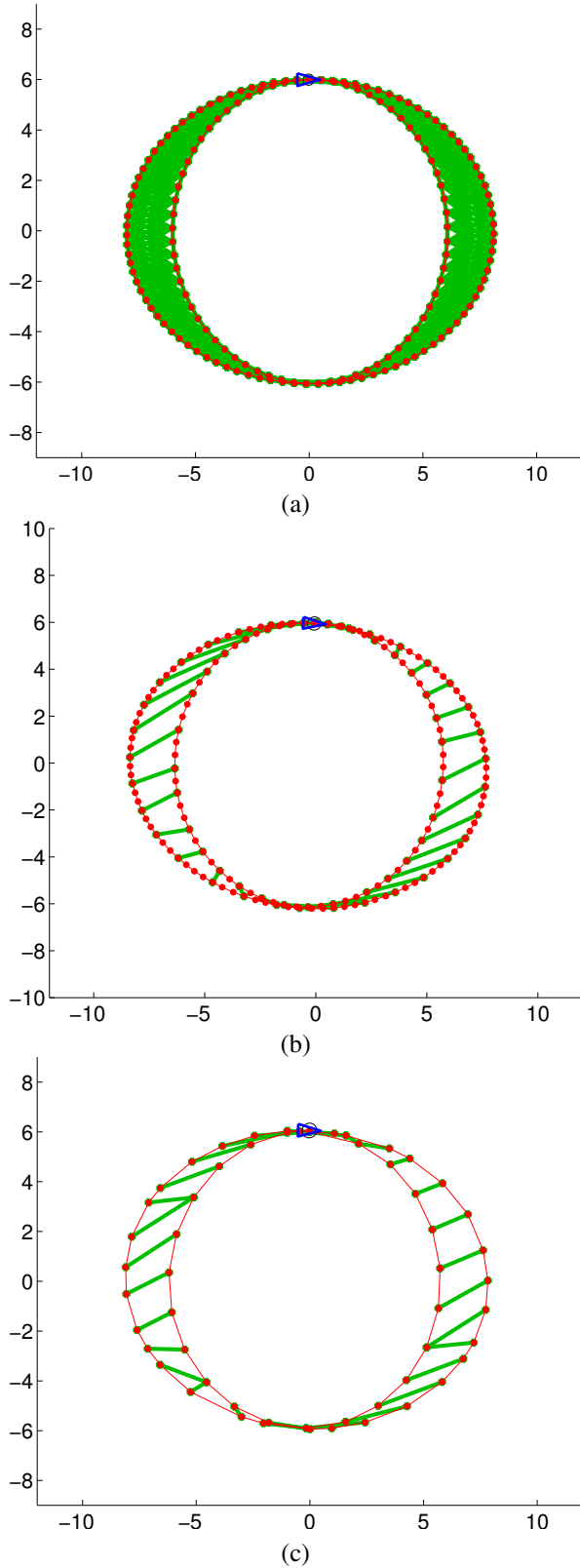


Fig. 1. Simulations to exemplify the strategies proposed in the paper a) The standard approach that incorporates all poses and all links to the filter; b) Incorporating all poses but only highly informative links; c) Incorporating only relevant poses and highly informative links.

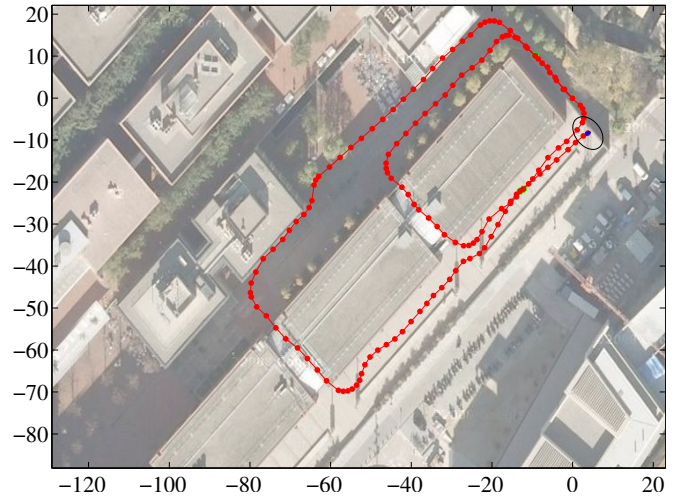


Fig. 2. Filtered trajectory (in red) using encoder and visual odometry on a dataset collected at the UPC Campus Nord. Loop closure links are displayed in green and the blue arrow indicate the final pose of the robot and the black ellipse the associated covariance at a 95% confidence level.

Robot Lab located at the UPC Campus Nord (see Fig. 2). Due to the repetitive structure of the UPC Campus Nord, this trajectory included many portions prone to perceptual aliasing. When all possible loops are closed we end up with 368 poses and 317 loop-closure links and the execution time is 296 s, without considering the vision related processes. When limiting the links to those that produce an information gain above 3, the simulation runs in 92 s and only 3 loop-closure links are established. Finally, if we only retain non-redundant poses we end up with a filter with 147 poses and 3 loop-closure links. In this case, the execution time is only 62 s, about one sixth of the original execution time.

Finally, to test the performance in a sequence with a much larger number of constraints we used the Intel dataset from [21]. The dataset includes 26915 odometry readings and 13631 laser scans. The laser scans are used to generate an alternative odometry and to assert loop closures aligning them using an ICP scan matching algorithm [7]. In this case, only poses closer than  $\pm 1\text{m}$  in  $x$  and  $y$  and  $\pm 0.35\text{rad}$  in orientation are considered reliable. The robot odometry and the laser scan match are modeled with noise covariances  $\Sigma_u = \text{diag}(0.05^2, 0.05^2, 0.03^2)$  and  $\Sigma_y = \text{diag}(0.05^2, 0.05^2, 0.009^2)$ , respectively. Finally, the covariance of the initial pose is set to  $\Sigma_{00} = \text{diag}(0.1^2, 0.1^2, 0.09^2)$ . Using the algorithm introduced in this paper with  $g = 6$ , we end up with a state including only 1218 poses and 103 links (see Fig. 3). This data set has been used in many other SLAM related works and, due to its large size, it is often pre-processed and reduced to about 1000 poses with about 3500 loop closure links [6]. The system we propose automatically selects the optimal subset of poses in the sense of the information gain and allows for a more efficient selection of the loop closure links.

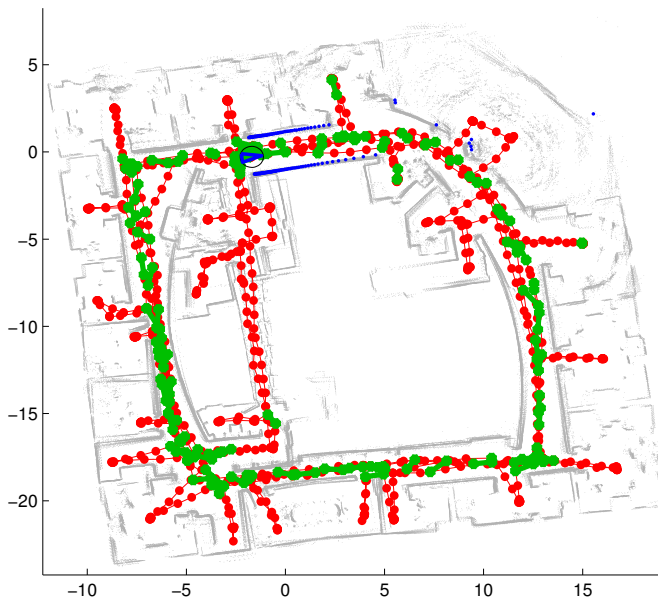


Fig. 3. Filtered trajectory using encoder and laser odometry of the Intel dataset. The blue arrow indicates the final pose of the robot and the black ellipse the associated covariance at a 95% confidence level.

## VI. CONCLUSIONS AND FUTURE WORK

This paper offers efficient solutions to reduce the state representation in delayed-state SLAM by adding only non-redundant poses and informative links. This is achieved by computing two measures, the relative distance between poses and the mutual information gain for each candidate link. In our previous work [9], we introduced these measures and methods to approximate them. In contrast, in this paper, we described how to exactly compute these two measures in an efficient way. The main problem to overcome has been that of obtaining the cross correlation between the current robot pose and previously visited poses. We have shown that these cross-correlations can be factorized and computed on demand allowing an exact, constant-time computation of both the relative distance and the information gain.

The presented experiments with real data show that the number of poses in the state can be reduced up to 10%, the number of links drops up to 1%, and the execution time is reduced up to 16% compared with the standard delayed-state SLAM algorithm, where all poses and links are included in the filter, without compromising the quality of estimation.

With the proposed strategy, the robot operates most of the time in open loop and the cost of updating the entire state after a loop closure is amortized over long periods. With this, the bottleneck for real time execution is not state recovery, but detecting neighboring poses for which feature matching is likely. Currently, this process is implemented as a linear search. In the near future, we plan to explore the possibility to organize the poses in a tree-like structure to speed up this search.

## REFERENCES

- [1] R. Smith, M. Self, and P. Cheeseman, "A stochastic map for uncertain spatial relationships," in *Proc. 4th Int. Sym. Robot. Res.*, Santa Clara, 1988, pp. 467–474.
- [2] M. W. M. G. Dissanayake, P. Newman, S. Clark, H. F. Durrant-Whyte, and M. Csorba, "A solution to the simultaneous localization and map building (SLAM) problem," *IEEE Trans. Robot. Automat.*, vol. 17, no. 3, pp. 229–241, Jun. 2001.
- [3] S. Thrun, Y. Liu, D. Koller, A. Y. Ng, Z. Ghahramani, and H. Durrant-Whyte, "Simultaneous localization and mapping with sparse extended information filters," *Int. J. Robot. Res.*, vol. 23, no. 7–8, pp. 693–716, Jul. 2004.
- [4] M. Montemerlo and S. Thrun, *FastSLAM: A Scalable Method for the Simultaneous Localization and Mapping Problem in Robotics*, ser. Springer Tracts in Advanced Robotics. Springer, 2007, vol. 27.
- [5] F. Dellaert and M. Kaess, "Square root SAM: Simultaneous localization and mapping via square root information smoothing," *Int. J. Robot. Res.*, vol. 25, no. 12, pp. 1181–1204, 2006.
- [6] M. Kaess, A. Ranganathan, and F. Dellaert, "iSAM: Incremental smoothing and mapping," *IEEE Trans. Robot.*, vol. 24, no. 6, pp. 1365–1378, 2008.
- [7] F. Lu and E. Milios, "Globally consistent range scan alignment for environment mapping," *Auton. Robot.*, vol. 4, no. 4, pp. 333–349, 1997.
- [8] R. M. Eustice, H. Singh, and J. J. Leonard, "Exactly sparse delayed-state filters for view-based SLAM," *IEEE Trans. Robot.*, vol. 22, no. 6, pp. 1100–1114, Dec. 2006.
- [9] V. Ila, J. Andrade-Cetto, R. Valencia, and A. Sanfeliu, "Vision-based loop closing for delayed state robot mapping," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, San Diego, Nov. 2007, pp. 3892–3897.
- [10] K. Konolige and M. Agrawal, "FrameSLAM: from bundle adjustment to realtime visual mapping," *IEEE Trans. Robot.*, vol. 24, no. 5, pp. 1066–1077, 2008.
- [11] S. J. Julier and J. K. Uhlmann, "A counter example to the theory of simultaneous localization and map building," in *Proc. IEEE Int. Conf. Robot. Automat.*, Seoul, May 2001, pp. 4238–4243.
- [12] T. Bailey, J. Nieto, J. Guivant, M. Stevens, and E. Nebot, "Consistency of the EKF-SLAM algorithm," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Beijing, Oct. 2006, pp. 3562–3568.
- [13] G. Dissanayake, S. B. Williams, H. Durrant-Whyte, and T. Bailey, "Map management for efficient simultaneous localization and mapping (SLAM)," *Auton. Robot.*, vol. 12, no. 3, pp. 267–286, May 2002.
- [14] W. Zhou, J. Miro, and G. Dissanayake, "Information-driven 6D SLAM based on ranging vision," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Nice, Sep. 2008, pp. 2072–2077.
- [15] T. Vidal-Calleja, A. Davison, J. Andrade-Cetto, and D. Murray, "Active control for single camera SLAM," in *Proc. IEEE Int. Conf. Robot. Automat.*, Orlando, May 2006, pp. 1930–1936.
- [16] U. Frese, P. Larsson, and T. Duckett, "A multigrid algorithm for simultaneous localization and mapping," *IEEE Trans. Robot.*, vol. 21, no. 2, pp. 1–12, 2005.
- [17] R. M. Eustice, H. Singh, J. J. Leonard, and M. R. Walter, "Visually mapping the RMS Titanic: Conservative covariance estimates for SLAM information filters," *Int. J. Robot. Res.*, vol. 25, no. 12, pp. 1223–1242, 2006.
- [18] S. Huang, Z. Wang, and G. Dissanayake, "Exact state and covariance sub-matrix recovery for submap based sparse EIF SLAM algorithm," in *Proc. IEEE Int. Conf. Robot. Automat.*, Pasadena, Apr. 2008, pp. 1868–1873.
- [19] Y. Chen, T. A. Davis, W. W. Hager, and S. Rajamanickam, "Algorithm 887: CHOLMOD, supernodal sparse Cholesky factorization and update/downdate," *ACM T. Math. Soft.*, vol. 35, no. 3, 2008.
- [20] D. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vision*, vol. 60, no. 2, pp. 91–110, 2004.
- [21] A. Howard and N. Roy, "The robotics data set repository (Radish)," <http://radish.sourceforge.net>, 2003.