

UNIVERSITAT POLITÈCNICA DE CATALUNYA

Doctoral Programme:

AUTOMÀTICA, ROBÒTICA I VISIÓ

Research Plan:

**Information Gain Metrics  
for Localization and Mapping**

Joan Vallvé Navarro

Advisor: Juan Andrade-Cetto

June 10, 2015



## Contents

<b>1</b>	<b>Introduction and motivation</b>	<b>1</b>
<b>2</b>	<b>Objectives</b>	<b>1</b>
<b>3</b>	<b>Expected contribution</b>	<b>2</b>
<b>4</b>	<b>State of the art</b>	<b>2</b>
<b>5</b>	<b>Preliminary work</b>	<b>11</b>
<b>6</b>	<b>Work plan</b>	<b>14</b>
<b>7</b>	<b>Resources</b>	<b>17</b>
<b>8</b>	<b>Publications</b>	<b>17</b>



# 1 Introduction and motivation

Three of the most researched topics in mobile robotics are localization, mapping and path planning. We can define mapping as the problem of generating a representation of the environment (a map) from the sensory output of a robot that is following a given trajectory. Localization is the problem of estimating the pose of such robot within a given a map whilst executing a given trajectory. And, path planning is the problem of computing a feasible trajectory from an initial configuration to a final one, given a map and full certainty about the robot location within it.

The intersection of these three problems generates four new problems. Combining localization and mapping, we define simultaneous localization and mapping as the problem of generating map and localizing the robot in it while it is executing a given trajectory. Alternatively, the problem of planning the trajectory that will localize better the robot given a map, i.e. the combination of localization and path planning, is called active localization. Exploration, the combination of mapping and path planning, is the problem of driving a (localized) robot to build a map of an environment. Finally, we call the combination of all these problems as Active Simultaneous Localization and Mapping (Active SLAM), and we define it as the problem of finding the best path to drive a robot to build a map of a previously unknown environment and simultaneously localizing itself in it.

The problem grows as time increases, receiving more sensor data, increasing the trajectory of the robot to be estimated, enlarging the map, etc. Hence finding scalable solutions to the combined problem poses big challenges.

Nevertheless, even though mapping, localization and planning are widely studied topics in mobile robotics, they are not the ultimate tasks we want the robots to do. They should be background non-costly processes that provide the robot with basic capabilities to do more sophisticated tasks such as fetching goods, guiding a human, surveying an area, etc.

Significant advances have been made in all these areas. We can find now state of the art algorithms that accurately localize a robot in real time for a restricted number of domains, but new application demands require for this to happen in systems with limited computational power (cell phones, UAVs, embedded computing systems, etc.) So finding scalable solutions to the problem is always a valid concern for researchers. This poses a compromise between computation speed and accuracy; and to tackle the problem, some researchers opt to reduce the problem size (dropping measurements, marginalizing states, subsampling, pruning the graph of constraints, etc.), or renounce to build it from scratch (relinearizing part of the problem), all these using most of the time heuristics. In this thesis we will derive principled ways to tackle such variants of problem reduction using information metrics to evaluate the effect of them.

## 2 Objectives

The main objective of the thesis is to propose and formalize information gain metrics that can be used to tackle the scalability concern for the SLAM and Active SLAM problems, the amount of information at stake.

As explained before, in SLAM there will always be a compromise between computation time and accuracy and several decisions will have to be taken in that sense. We will exploit various information metrics to provide criteria to make those decisions. By measuring the amount of information that is ignored or taken into account when relinearizing, pruning, subsampling or even solving only parts of the problem will help us make principled decision to balance accuracy with computational cost.

In the case of Active SLAM, the decision of which paths to take have a direct influence on both the total travelled distance by the robot, hence time, and on the quality of the resulting

map. The information gain on a path estimate and the ensuing map would allow us to evaluate different path candidates and even give some clues on how to create the path candidate set.

Sumarizing, our objectives are to efficiently evaluate the information gain for the various parts of the SLAM and Active SLAM problems:

- Develop and formalize information metric-based formulations to be used in existing state-of-art SLAM methods to ease the time and space computational burden of the algorithms (i.e. graph pruning to discard observations and/or variables from the problem; partial relinearization and partial problem solving; etc.).
- Develop information gain-based Active SLAM methods, choosing the most infomative path to efficiently and correctly map the environment.

### 3 Expected contribution

To accomplish the abovementioned objectives, we sill study the state of the art and provide a comprehensive view on the use of information metrics for the solution of the SLAM and Active SLAM problems, and contribute with novel alternatives.

For the case of SLAM, the ultimate goal is to make the problem tractable, using information metrics to make decisions about the use of sensory data even before it is fed to any state estimation or optimization machinery. Furthermore, the effect of missing sensory data (including odometry) will be systematically analyzed with respect to the loss of accuracy.

In consequence, we expect to provide means for the systematic pruning of measurements and variables; and an information-hierarchic representation of the problem that allow solving only the most informative part using state-of-art SLAM methods, hence reducing computational burden. The effect of relinearization (i.e. rebuilding the nonlinear problem at a new linearization point) on the accuracy and computational cost of the solution will also be studied.

This introduces a new perspective to the problem, the time saved by those processes that renounce to accuracy (prunning state variables or measurements) should be balanced with that consumed by those processes that improve the accuracy of the solution (relinearization).

Furthermore, since Active SLAM is driven by opposing forces (the exploration vs. exploitation paradigm), a joint metric that evaluates the combined effect of path and map entropy decrease will be proposed.

This project is developed in the framewok of the FP7-SST EU project Cargo-ANTs, dedicated to the autonomous navigation of trucks and AGVs in cargo ports and terminals. Hence, a SLAM front-end, compatible with the latest SLAM back-end solutions that makes use of the proposed solutions to scalability will be provided.

## 4 State of the art

### Simultaneous localization and mapping

Several different approaches to solve the SLAM problem exist depending on the environment representation choosen, on what is estimated , or on how the solution is computed.

Localizing the robot means estimating the current robot pose. For this reason, initial solutions to the SLAM problem estimated some environment landmarks and only the current robot pose [6]. However, the concept of SLAM has evolved to include the full history of tobot poses in the state, as well as the landmarks. This is kown as full SLAM [7]. And, when the landmarks are

marginalized and only the robot poses are retained, we may refer to the Pose SLAM solution to the problem [8].

These first SLAM methods [9, 10] used an Extended Kalman Filter (EKF) to solve the problem. In the EKF the state  $\mathbf{x}$  is estimated as a multi-variate Gaussian distribution  $\mathbf{p}(\mathbf{x}) \sim \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ , with its mean  $\boldsymbol{\mu}$  and its covariance  $\boldsymbol{\Sigma}$ .

The filter updates the estimate of  $\mathbf{x}$  with observations  $\mathbf{z}_k$  from a variety of sensors. Observations<sup>1</sup> come either from proprioceptive or exteroceptive sensors and take the form

$$\mathbf{z}_k = h_k(\mathbf{x}) + \mathbf{v}_k, \quad \mathbf{v}_k \sim \mathcal{N}(0, \boldsymbol{\Sigma}_{z_k}) \quad (1)$$

being  $\boldsymbol{\Sigma}_{z_k}$  the measurement noise covariance of the  $k$ -th measurement, and the error between the expected observation and the one actually measured is

$$\mathbf{e}_k(\mathbf{x}) = h_k(\mathbf{x}) - \mathbf{z}_k. \quad (2)$$

Whenever an observation  $\mathbf{z}_k$  produces a new variable (whether a new landmark or a new robot pose) eq.2 cannot be computed, instead the problem is augmented with a new state variable  $\mathbf{x}_n$ , which obviously requires augmenting the sizes of the mean vector  $\boldsymbol{\mu}$  and the covariance matrix  $\boldsymbol{\Sigma}$ . The new state estimate is computed<sup>2</sup> propagating the sensor uncertainty with

$$\boldsymbol{\mu} \leftarrow \begin{bmatrix} \boldsymbol{\mu} \\ g_k(\boldsymbol{\mu}_i, \mathbf{z}_k) \end{bmatrix} \quad (3)$$

$$\boldsymbol{\Sigma} \leftarrow \begin{bmatrix} \boldsymbol{\Sigma} & \boldsymbol{\Sigma}_{:,i} \mathbf{G}_{k_x}^\top \\ \mathbf{G}_{k_x} \boldsymbol{\Sigma}_{i,:} & \mathbf{G}_{k_x} \boldsymbol{\Sigma}_{i,i} \mathbf{G}_{k_x}^\top + \mathbf{G}_{k_z} \boldsymbol{\Sigma}_{z_k} \mathbf{G}_{k_z}^\top \end{bmatrix} \quad (4)$$

where  $\mathbf{x}_n = g_k(\mathbf{x}_i, \mathbf{z}_k)$  is the inverse sensor model of the observation  $\mathbf{z}_k$ . It returns the new state variable  $\mathbf{x}_n$  given the observation  $\mathbf{z}_k$  and the pose at which was taken  $\mathbf{x}_i$ .  $\mathbf{G}_{k_x}$  and  $\mathbf{G}_{k_z}$  are the Jacobians of  $g_k(\mathbf{x}, \mathbf{z}_k)$  with respect to  $\mathbf{x}$  and  $\mathbf{z}_k$ , respectively, evaluated at  $\boldsymbol{\mu}$ .

Since state augmentation implies a multiplication of a covariance block-column by the Jacobian  $\mathbf{G}_{k_x}$ , its cost is linear with respect to the problem size  $\mathcal{O}(n)$ .

However, to update the filter with a new observation  $\mathbf{z}_k$  between existing variables of the state, the state update is computed with

$$\Delta \boldsymbol{\mu} = \boldsymbol{\Sigma} \mathbf{J}_k \mathbf{S}_k (\mathbf{z}_k - h_k(\boldsymbol{\mu})) \quad (5)$$

$$\Delta \boldsymbol{\Sigma} = -\boldsymbol{\Sigma} \mathbf{J}_k^\top \mathbf{S}_k^{-1} \mathbf{J}_k \boldsymbol{\Sigma} \quad (6)$$

being  $\mathbf{J}_k$  the Jacobian of  $h_k(\mathbf{x})$  with respect to  $\mathbf{x}$ , evaluated at  $\boldsymbol{\mu}$ , and  $\mathbf{S}_k$  the innovation matrix

$$\mathbf{S}_k = \mathbf{J}_k \boldsymbol{\Sigma} \mathbf{J}_k^\top + \boldsymbol{\Sigma}_{z_k}. \quad (7)$$

$\mathbf{S}_k$  has the size of the variables involved in the observation, hence its inversion does not imply a big cost as long as the size of the measurement is small. However, the multiplication of the entire covariance matrix has quadratic cost with respect to the problem size  $\mathcal{O}(n^2)$ , and in SLAM,  $\boldsymbol{\Sigma}$  encodes all metric correlations between variables, ultimately becoming fully correlated. For this reason, the naïve solution of the EKF SLAM problem becomes computationally intractable even for problems of moderate size.

<sup>1</sup> To cope with the two-stage prediction and update steps of the traditional Kalman filter, observations are typically divided in the literature as those coming from the motion model, and those coming from the measurement model. However, motion can be rewritten analogously as an observation.

<sup>2</sup>The left arrow is used to depict an update on the content of the state. The subindex denotes row and column blocks. For instance,  $\Sigma_{3,3}$  refers to the marginal covariance block of the 3rd block of variables. For block column or row, we use ':' meaning all blocks.

To avoid that, one can resort to the Extended Information Filter (EIF) [11] which represent the state using the information form  $\mathbf{p}(\mathbf{x}) \sim \mathcal{N}^{-1}(\mathbf{x}; \boldsymbol{\eta}, \boldsymbol{\Lambda})$ , where the information matrix  $\boldsymbol{\Lambda}$  and the information vector  $\boldsymbol{\eta}$  are related with the mean  $\boldsymbol{\mu}$  and covariance  $\boldsymbol{\Sigma}$  by

$$\boldsymbol{\Lambda} = \boldsymbol{\Sigma}^{-1}, \quad \boldsymbol{\eta} = \boldsymbol{\Lambda}\boldsymbol{\mu}. \quad (8)$$

Taking this representation, the information matrix become sparse because it only encodes the conditionals between variables instead of correlations. Then, the filter update with a new observation  $\mathbf{z}$  becomes

$$\Delta\boldsymbol{\eta} = \mathbf{J}_k^\top \boldsymbol{\Sigma}_{z_k}^{-1} (\mathbf{z}_k - h_k(\boldsymbol{\mu}) + \mathbf{J}_k \boldsymbol{\mu}) \quad (9)$$

$$\Delta\boldsymbol{\Lambda} = \mathbf{J}_k^\top \boldsymbol{\Sigma}_{z_k}^{-1} \mathbf{J}_k \quad (10)$$

Augmenting the problem is equivalent to updating after resizing the information matrix and vector with zeros. However, despite that the computational cost of augmenting and updating the state is near constant since  $\mathbf{J}_k$  is full of zeros except for the variables involved in the observation, the solution of the problem is not yet available. It is mandatory to have a solution of the mean to evaluate Jacobians, and sometimes of the covariance of the estimation. Then, a state recovery method is needed and the naïve implementation would have cubical cost  $O(n^3)$  since it implies inverting the information matrix (eq. 8), being even more expensive than the naïve EKF.

Ila et al. [8] used the EIF to estimate only the robot trajectory using the sensor data to establish correspondences between robot poses. In such case, an efficient state recovery solution provides the mean, the marginal covariances of all the poses, and the last covariance block column with constant time after an augment step, and almost in linear time with respect to the state size after an update step.

Both actions are critical in terms of computational cost. Adding a large number of robot poses will significantly increase the computational cost of a future loop closure, and closing a large number of loops implies also an important cost. In [8], the information gain is defined as the entropy change of the state estimation, and it is used to make decisions about when to add new robot poses to the state or when to close loops. Adding only those poses that add “significant amounts” of information to the state and closing only the “sufficiently informative” loops allows for a much smaller representation of the problem without renouncing to the main information of it. It meant, solving the problem faster without significantly renouncing to the accuracy.

But whatever representation is used, filtering linearizes each observation over the mean  $\boldsymbol{\mu}$  when it has arrived, and no relinearization is done even if the solution has importantly changed. As a consequence, filtering is overconfident in measuring the error and can end up producing an ill-defined covariance or information matrix.

The SLAM problem can be represented by a graph such as factor graph (fig. 1). Graph-based SLAM methods are those that use graphs for representing and managing the problem. Several graph-based SLAM methods tackle the problem as a non-linear least squares problem, i.e., as the minimization of the squared Mahalanobis distance of the measurement error

$$\Delta\boldsymbol{\mu}^* = \arg \min_{\Delta\boldsymbol{\mu}} \sum_k \|\mathbf{e}_k(\boldsymbol{\mu} + \Delta\boldsymbol{\mu})\|_{\boldsymbol{\Sigma}_k}^2. \quad (11)$$



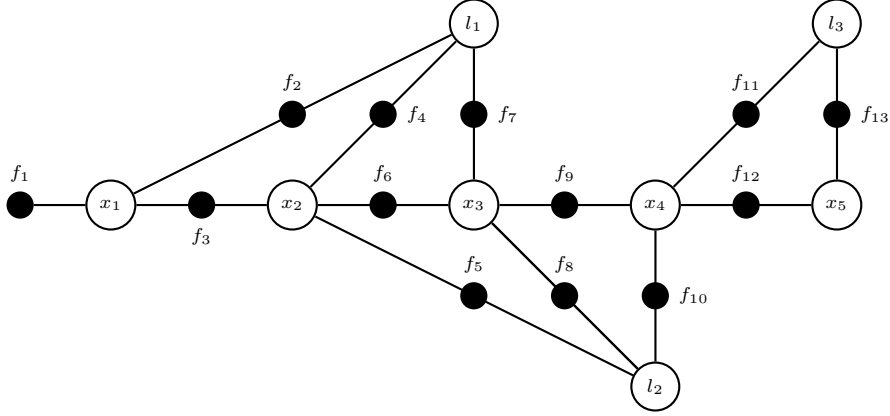


Figure 1: Factor graph example. White nodes represent the state variables  $\mathbf{x}$  including the robot trajectory ( $x_1, x_2, x_3, x_4, x_5$ ) and the set of landmarks ( $l_1, l_2, l_3$ ). Black nodes (factors) represent observations  $\mathbf{z}$ , i.e. constraints, including odometry constraints ( $f_3, f_6, f_9, f_{12}$ ), landmark observations ( $f_2, f_4, f_5, f_7, f_8, f_{10}, f_{11}, f_{13}$ ) and a prior for the first robot pose  $f_1$ .

Linearizing the error,

$$\begin{aligned}
\mathbf{e}_k(\boldsymbol{\mu} + \Delta\boldsymbol{\mu}) &= \underbrace{\mathbf{e}_k(\boldsymbol{\mu})}_{\check{\mathbf{e}}_k} + \left. \frac{\partial \mathbf{e}_k(\mathbf{x})}{\partial \mathbf{x}} \right|_{\boldsymbol{\mu}} \Delta\boldsymbol{\mu} \\
&= \check{\mathbf{e}}_k + \left. \frac{(\partial h_k(\mathbf{x}) - \mathbf{z})}{\partial \mathbf{x}} \right|_{\boldsymbol{\mu}} \Delta\boldsymbol{\mu} \\
&= \check{\mathbf{e}}_k + \left( \left. \frac{\partial h_k(\mathbf{x})}{\partial \mathbf{x}} \right|_{\boldsymbol{\mu}} - \cancel{\left. \frac{\partial \mathbf{z}}{\partial \mathbf{x}} \right|_{\boldsymbol{\mu}}} \right) \Delta\boldsymbol{\mu} \\
&= \check{\mathbf{e}}_k + \mathbf{J}_k \Delta\boldsymbol{\mu}
\end{aligned} \tag{12}$$

and Eq. 11 becomes

$$\Delta\boldsymbol{\mu}^* = \arg \min_{\Delta\boldsymbol{\mu}} \|\mathbf{A}\Delta\boldsymbol{\mu} + \mathbf{b}\|^2, \tag{13}$$

being  $\mathbf{A} = [\boldsymbol{\Sigma}_1^{-\top/2} \mathbf{J}_1 \dots \boldsymbol{\Sigma}_k^{-\top/2} \mathbf{J}_k]^\top$  and  $\mathbf{b} = [\boldsymbol{\Sigma}_1^{-\top/2} \check{\mathbf{e}}_1 \dots \boldsymbol{\Sigma}_k^{-\top/2} \check{\mathbf{e}}_k]^\top$ .

One important thing to point out is that  $\mathbf{A}$  is sparse since every measurement depends only on a few state variables and thus  $\mathbf{J}_k$  is full of zeros except for the blocks corresponding to the variables involved. From this point, two main approaches have been exploited to solve eq. 13. The first approach [12–14] uses QR factorization of  $\mathbf{A}$

$$\mathbf{A} = \mathbf{Q} \begin{bmatrix} \mathbf{R} \\ \mathbf{0} \end{bmatrix} \tag{14}$$

where  $\mathbf{Q}$  is a rotation matrix and  $\mathbf{R}$  is an upper triangular matrix. Applying the QR factorization, we obtain a solution of eq. 13

$$\mathbf{R}\Delta\boldsymbol{\mu}^* = -\mathbf{d}, \quad \text{where} \quad \begin{bmatrix} \mathbf{d} \\ \mathbf{c} \end{bmatrix} = \mathbf{Q}^\top \mathbf{b}. \tag{15}$$

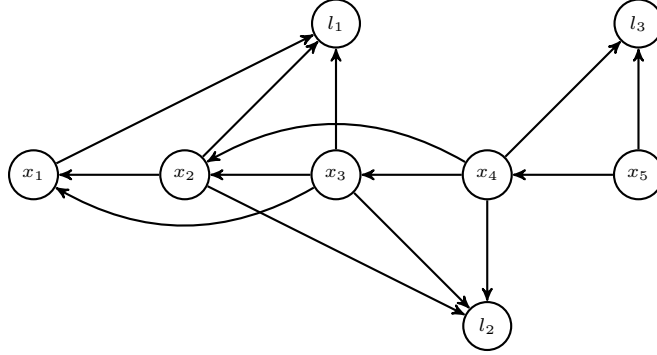


Figure 2: Bayes net after variable elimination of the factor graph in fig. 1 with the elimination order:  $l_1, l_2, l_3, x_1, x_2, x_3, x_4, x_5$ . Each directed edge represents a conditional probability.

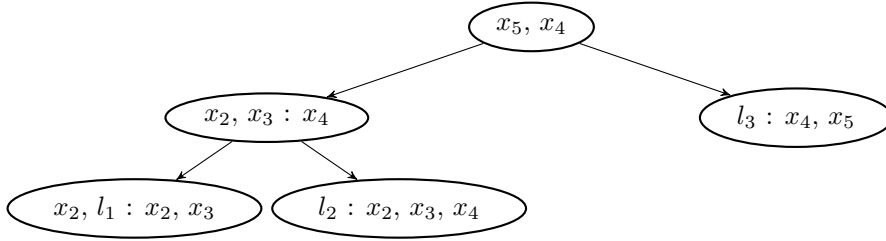


Figure 3: Bayes tree representation of the Bayes net fig. 2 introduced by Dellaert et al. in iSAM2. Built in the reverse elimination order of the Bayes net discovering its cliques.

Therefore, the optimal solution for the update  $\Delta\mu^*$  can be computed by back-substitution with a computational cost quadratic to the size of the problem in the worst case (when  $\mathbf{R}$  is dense). However, given the sparsity of  $\mathbf{A}$ , using some variable (column) ordering algorithms such as COLAMD [15], the resulting matrix  $\mathbf{R}$  can also be made sparse, significantly decreasing the computational cost.

In [12], Dellaert et al. presented  $\sqrt{\text{SAM}}$  using QR factorization for solving the smoothing problem. In  $\sqrt{\text{SAM}}$ , once a new observation arrives, all the errors are relinearized and new  $\mathbf{A}$  and  $\mathbf{b}$  are built, then the QR factorization is made from scratch and the problem is solved by back-substitution. The incremental version, iSAM [13], updates the matrix  $\mathbf{R}$  and vector  $\mathbf{d}$  while new measurements arrive. It implies renouncing to the relinearization of the error of previously added measurements and to the variable reordering, so batch relinearization (rebuilding of  $\mathbf{A}$  and  $\mathbf{b}$ ) and variable reordering is periodically done providing a new matrix  $\mathbf{R}$  and a new vector  $\mathbf{d}$  for computing the solution.

Kaess et al. improved this incremental smoothing and mapping approach with iSAM2 [14] making use of a new graph structure, the Bayes Tree. Performing variable elimination from the factor graph (fig. 1) a Bayes net is obtained (fig. 2). This process is equivalent to the QR factorization being the variable order equivalent to the elimination order. Finally, the Bayes Tree (see figure 3) encodes the cliques of the Bayes net, storing all non-zero entries of the equivalent  $\mathbf{R}$  matrix. Methods to incrementally update the problem and relinearize specific variables within the Bayes Tree were provided overperforming their previous method iSAM both in computational time and accuracy of results.

Besides QR factorization, the problem can be solved in an alternative way. Derivating the problem in eq. 11 with respect to  $\Delta\boldsymbol{\mu}$  and equalling it to zero, considering the linearized error (eq. 12), we obtain

$$\Delta\boldsymbol{\mu}^* = \arg \min_{\Delta\boldsymbol{\mu}} \|\mathbf{A}\Delta\boldsymbol{\mu} + \boldsymbol{\nu}\|^2, \quad (16)$$

with

$$\mathbf{A} = \sum_k \mathbf{J}_k^\top \boldsymbol{\Sigma}_k \mathbf{J}_k = \mathbf{A}^\top \mathbf{A} \quad (17)$$

$$\boldsymbol{\nu} = \sum_k \mathbf{J}_k^\top \boldsymbol{\Sigma}_k \check{\mathbf{e}}_k = \mathbf{A}^\top \mathbf{b}, \quad (18)$$

which is equivalent to premultiplying Eq. 13 by  $\mathbf{A}^\top$ . Then the solution of the problem,  $\Delta\boldsymbol{\mu}^*$  is given by

$$\mathbf{A}\Delta\boldsymbol{\mu} = -\boldsymbol{\nu}. \quad (19)$$

This is the Gauss-Newton optimization step of the least squares approach of SLAM. It is important to note that the information matrix  $\mathbf{A}$  is the same as in the EIF, and thus, it is symmetric and sparse.

Using the Cholesky factorization of  $\mathbf{A} = \mathbf{R}^\top \mathbf{R}$  being  $\mathbf{R}$  an upper triangular matrix <sup>3</sup> we obtain

$$\mathbf{R}^\top \mathbf{R} \Delta\boldsymbol{\mu} = -\boldsymbol{\nu}, \quad (20)$$

and using a forward-substitution and then a back-substitution, the solution is obtained

$$\mathbf{R}^\top \mathbf{y} = -\boldsymbol{\nu} \quad (21)$$

$$\mathbf{R} \Delta\boldsymbol{\mu} = \mathbf{y}. \quad (22)$$

For the same problem and variable ordering, we see that  $\mathbf{R}$  is the same for the QR factorization of  $\mathbf{A}$  and for the Cholesky factorization of  $\mathbf{A}$  since  $\mathbf{A} = \mathbf{A}^\top \mathbf{A} = (\mathbf{QR})^\top \mathbf{QR} = \mathbf{RQ}^\top \mathbf{QR} = \mathbf{R}^\top \mathbf{R}$ . Then, also for the Cholesky factorization, the sparsity of  $\mathbf{R}$  can be increased by variable ordering in  $\mathbf{A}$ .

Grisetti et al. presented a hierarchical optimization scheme called HOG-Man [16], solving the Gauss-Newton optimization using the Cholesky factorization over the quaternion manifold. In order not to recompute the solution of the whole problem each iteration, HOG-Man works with some graphs representing the problem in different hierarchical levels. The entire problem is level 0 and grouping the graph nodes depending on graph-distance, higher hierarchy graphs are built. After solving the highest hierarchy graph, only lower graphs will be solved if the solution changed more than a certain Euclidean distance threshold.

Polok et al. [17] analogously to iSAM2, presented the incremental version of the Cholesky factorization of  $\mathbf{A}$ . After taking a new measurement,  $\mathbf{R}$  is updated directly and variable ordering and relinearization is also done partially when needed.

Altering eq. 19

$$(\mathbf{A} + \lambda \mathbf{I}) \Delta\boldsymbol{\mu} = -\boldsymbol{\nu}, \quad (23)$$

we obtain the Levenberg-Marquardt (LM) method. It controls the convergence of the optimization by dynamically changing the damping factor  $\lambda$  at each iteration depending on how the error changes. When a bigger error is obtained after solving, the damping factor  $\lambda$  is increased giving a step closer to the gradient descent direction. On the contrary, for smaller error, the value of  $\lambda$  is decreased, bringing the algorithm closer to the original Gauss-Newton in eq. 19.

---

<sup>3</sup>or equivalently,  $\mathbf{A} = \mathbf{L}\mathbf{L}^\top$  being  $\mathbf{L}$  a lower triangular matrix:  $\mathbf{L} = \mathbf{R}^\top$

Konolige et al. proposed Sparse Pose Adjuntment (SPA) [18], which implements a LM optimization for solving 2D SLAM problems. Kummerle et al. presented g<sup>2</sup>o [19] also using LM for solving more general SLAM problems. Specifically, g<sup>2</sup>o exploits the structure of the landmark-based SLAM problem as in [20]. Consider that the robot poses  $\mathbf{x}_P$  and landmarks  $\mathbf{x}_L$  are ordered in blocks, then eq. 23 can be rewritten as

$$\begin{bmatrix} \mathbf{\Lambda}_{PP} + \lambda \mathbf{I} & \mathbf{\Lambda}_{PL} \\ \mathbf{\Lambda}_{PP}^\top & \mathbf{\Lambda}_{LL} + \lambda \mathbf{I} \end{bmatrix} \begin{bmatrix} \Delta \mu_P \\ \Delta \mu_L \end{bmatrix} = - \begin{bmatrix} \nu_P \\ \nu_L \end{bmatrix} \quad (24)$$

and if we consider the two separate equations

$$(\mathbf{\Lambda}_{PP} + \lambda \mathbf{I}) \Delta \mu_P + \mathbf{\Lambda}_{PL} \Delta \mu_L = -\nu_P \quad (25)$$

$$\mathbf{\Lambda}_{PP}^\top \Delta \mu_P + (\mathbf{\Lambda}_{LL} + \lambda \mathbf{I}) \Delta \mu_L = -\nu_L, \quad (26)$$

solving for  $\mu_L$  in eq. 26 and substituting in eq. 25 we obtain a new smaller linear problem

$$(\mathbf{\Lambda}_{PP} + \lambda \mathbf{I} - \mathbf{\Lambda}_{PL}(\mathbf{\Lambda}_{LL} + \lambda \mathbf{I})^{-1} \mathbf{\Lambda}_{PL}^\top) \Delta \mu_P = -\nu_P + \mathbf{\Lambda}_{PL}(\mathbf{\Lambda}_{LL} + \lambda \mathbf{I})^{-1} \nu_L. \quad (27)$$

To build it, only the inversion of  $\mathbf{\Lambda}_{LL} + \lambda \mathbf{I}$  has to be computed and it only has linear cost, since the information matrix part relating to the landmark variables  $\mathbf{\Lambda}_{LL}$  is block diagonal. Then, solving this linear system can be done by Cholesky factorization of a matrix significantly smaller than  $\mathbf{\Lambda} + \lambda \mathbf{I}$ . Once  $\Delta \mu_P$  is obtained,  $\Delta \mu_L$  is taken from eq. 26 with no need to invert more matrices.

The emergence of these least squares minimization solutions outperformed significantly the accuracy and consistency of the filter-based methods by allowing the relinearization of the problem. Whatever method is used to solve it, the computational complexity is related with the variable ordering. To improve the time consumption of batch solution of SLAM (linearizing, factorizing and solving) incremental approaches update the problem with new measurements renouncing to relinearize and/or reorder (hence rebuild) the entire problem.

Although  $\sqrt{\text{SAM}}$ , iSAM, iSAM2, g<sup>2</sup>o and incremental block Cholesky factorization do not focus on that, the problem size (i.e. the amount of variables) and density (i.e. the amount of observations) are equally important. Different alternatives are chosen in order to face it from discarding to marginalizing.

One alternative to the increasing size of SLAM could be to consider a (temporal, spatial, etc) window and discarding or marginalizing the landmarks and/or poses that rest out of this window. This is the case of visual and visual-inertial odometry methods [21, 22]. While marginalizing ends up causing dense information matrices, discarding landmarks implies renouncing to large loop closures.

In Pose SLAM [8], some new observations and new robot poses are decided not to be added to the problem according to their information content, defined as the full multi-variate Gaussian distribution entropy change

$$\mathcal{I}_k = \Delta H(\mathbf{x}) = \frac{1}{2} \ln \frac{|\mathbf{\Lambda}|}{|\mathbf{\Lambda} + \Delta \mathbf{\Lambda}|} = \frac{1}{2} \ln \frac{|\mathbf{S}_k|}{|\mathbf{\Sigma}_{z_k}|}. \quad (28)$$

While redundant odometries are marginalized while they are integrated, poorly-informative loop closures are discarded. In contrast, in HOG-Man, the hierarchical structure allows to maintain the size of the problem bounded. The upper graphs represent marginalized subgraphs and if the error is low enough, the optimization is done for the upper graph levels. However, all observations are added to the lowest level graph.

Several publications approached the problem of graph pruning and sparsifying for SLAM. Kretzschmar and Stachniss [23] presented an information-theoretic compression method for pose graph SLAM by selecting the most informative laser scans with respect to the map. They defined the problem as finding the subset of measurements  $\mathbf{z}^*$  containing at most  $K$  measurements which maximize the mutual information of the map  $\mathbf{m}$  for that subset

$$\mathbf{z}^* = \arg \max_{\mathbf{z}^* \subseteq \mathbf{z}, |\mathbf{z}^*| \leq K} H(\mathbf{m}|\mathbf{z}) - H(\mathbf{m}|\mathbf{z}^*). \quad (29)$$

Then, the measurements left are approximately marginalized based in Chow-Liu trees [24] in order to maintain sparsity. Johansson et al. proposed a reduced pose graph that only grows with the size of the environment mapped by marginalizing new poses close to previous poses.

Wang et al. [25] use Kullback-Liebler divergence (KLD) of the map  $\mathbf{m}$  for a subset of poses  $\mathbf{x}^*$  and the observations  $\mathbf{z}^*$  recorded at those poses

$$KLD = \int p(\mathbf{m}|\mathbf{x}, \mathbf{z}) \log \frac{p(\mathbf{m}|\mathbf{x}, \mathbf{z})}{p(\mathbf{m}|\mathbf{x}^*, \mathbf{z}^*)} d\mathbf{m}. \quad (30)$$

They proposed two alternative mechanisms: keyframe selection and the pruning of previous pose nodes, approximating the KLD of the original graph to that of the pruned graph.

Mazuran et al. [26] formulated sparsification as a convex minimization problem and proposed a Chow-Liu tree-based sparsification of the graph. Chouldhary et al. [27] proposed an information-based reduced landmark SLAM. Their method minimizes an objective function based on memory consumption and the elimination of the least informative landmarks, defining the information of a landmark as the estimation entropy change as in eq. 28.

To summarize, three time-critical decisions in solving SLAM problems have been identified: variable ordering, relinearization and problem size and sparsity. While the first is only related to time consumption, the second and third relate also to its accuracy. Therefore the decisions, regarding the inclusion of observations and the amount of relinearization, present a trade-off between time consumption and accuracy of the results. A principled solution should weight both consequences simultaneously.

The fastest state-of-art SLAM methods, iSAM2 and incremental Cholesky factorization, approach the relinearization in a different manner. While the second rebuilds the entire problem when the solution falls further than a certain threshold from the linearization point, iSAM2 evaluates all nodes independently and only relinearizes some specific nodes while rebuilding part of the problem at the incremental reordering routine. However, both methods use Euclidean distance thresholds without taking into account the the quality of the estimate (i.e. its covariance). For instance, improving 0.1m the estimation after a relinearization is worth depending on how large the covariance of the estimation is. For large covariances, solving the problem with new observations would bring more information and perhaps be more beneficial than spending resources in relinearizing.

Relating to problem size and sparsity, several information-based methods exist for batch and incremental pruning the graph. Many of them propose to marginalize or approximately marginalize graph nodes or constraints, however, no relinearization is possible for those nodes once they are removed and correlation fill-in is produced. The use of hierarchical graphs could be combined with the idea marginalization in order not to definitively discard or marginalize nodes but instead to optimize at different levels of abstraction.

A coherent metric is needed in order to compare the contribution of relinearization vs. pruning. This metric will allow us to make informed decisions on the trade-offs between computational cost and accuracy.

## Exploration and Active SLAM

As defined above, exploration is the problem of planning trajectories in order to map the environment. Most typical approaches to exploration are frontier-based methods, defining the frontier as the boundary between free space and unknown space. Using an occupancy grid representation, Yamauchi presented a seminal frontier-based exploration method [28] that drives the robot to the nearest frontier. Since then, several frontier-based exploration methods implemented new frontier selection policies depending on some utility and cost [29, 30], using greedy strategies seeking coverage [31], or using potential fields to drive the robot to frontiers [32].

Some other methods seek coverage without driving the robot to frontiers. For instance, Amigoni and Caglioti [33] evaluate a set of random reachable goals weighting the expected map information gain and the distance to reach them. The integrated exploration method [34] takes into account also the localization uncertainty.

A principled way to consider exploration and localization uncertainty at the same time is by minimizing two independent terms, the map entropy and the path entropy. Feder et al. [35], propose a metric to evaluate uncertainty reduction as the sum of these two independent entropies, but only for the last robot pose and a limited set of landmarks with a one step exploration horizon. Bourgault et al. [36] alternatively propose a utility function that computes independently the potential reduction of vehicle localization uncertainty from a feature-based map, and the information gained over an occupancy grid. To consider joint map and path entropy reduction, Vidal et al. [37] tackled the issue taking into account robot and map cross correlations for the Visual SLAM EKF case. Torabi et al. [38] jointly computed the entropy reduction directly in configuration space but for a limited number of configurations.

Whereas the effect of actions can be evaluated with respect to both map entropy reduction or path entropy reduction, there is still the question on how to generate the action set to be evaluated. Action evaluation is a costly endeavour, and most exploration methods are designed to evaluate the effect of only limited set of heuristically chosen actions (path candidates) [39, 40].

Formally, the joint map  $\mathbf{m}$  and path  $\mathbf{x}$  entropy, given all motions and observations  $\mathbf{z}^4$  is given by the sum of the path entropy given  $\mathbf{z}$  and the average over the entropy of the resulting map for each of the paths in the path probability distribution

$$H(\mathbf{x}, \mathbf{m}|\mathbf{z}) = H(\mathbf{x}|\mathbf{z}) + \int_{\mathbf{x}} p(\mathbf{x}|\mathbf{z}) H(\mathbf{m}|\mathbf{x}, \mathbf{z}) d\mathbf{x}. \quad (31)$$

In [39], a particle filter is used, and the weighted integral is approximated as a sum for all weighted particles. In [40] instead, the joint entropy is approximated with

$$H(\mathbf{x}, \mathbf{m}|\mathbf{z}) \approx H(\mathbf{x}|\mathbf{z}) + H(\mathbf{m}|\boldsymbol{\mu}, \mathbf{z}), \quad (32)$$

where instead of averaging over all paths in the probability distribution, the second term evaluates the entropy of the map  $\mathbf{m}$  given full confidence on the path mean path  $\boldsymbol{\mu}$ .

For an occupancy map with size cell  $l$ , its entropy can be computed as the scalar sum

$$H(\mathbf{m}|\mathbf{z}) = -l^2 \sum_{c \in \mathbf{m}} (p_c \log p_c + (1 - p_c) \log(1 - p_c)), \quad (33)$$

where  $p_c$  is the classification probability of cell  $c$ .

The path entropy  $H(\mathbf{x}|\mathbf{z})$ , i.e. the entropy of a multivariate Gaussian of size  $n$  is given by

$$H(\mathbf{x}|\mathbf{z}) = \ln((2\pi e)^{(n/2)} |\boldsymbol{\Sigma}|). \quad (34)$$

---

<sup>4</sup>As in the previous subsection, the motion commands  $\mathbf{u}$  are conceptually equivalent to observations, and can be combined with measurements in an aggregated vector  $\mathbf{z}$ , hence simplifying the notation.

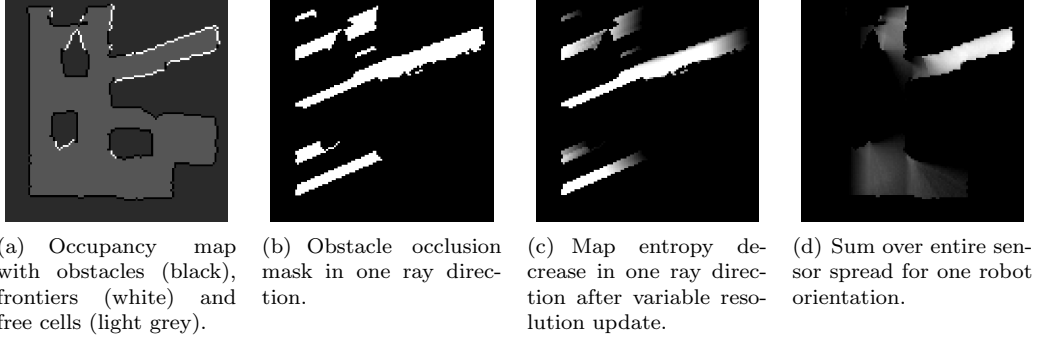


Figure 4: Computation of map entropy change for one particular robot orientation.

Unfortunately,  $\Sigma$  in Eq. 34 can easily become ill-defined when two poses become fully correlated. To compute an estimate, the entropy of the path could be approximated without taking into account correlations between poses [40], by averaging over the  $N$  individual pose marginals

$$H(\mathbf{x}|\mathbf{z}) \approx \frac{1}{N} \sum_{i=1}^N \ln((2\pi e)^{(dim/2)} |\Sigma_{ii}|). \quad (35)$$

where  $dim$  is the dimension of the poses.

As said previously, in previous approaches [39, 40], only a heuristically chosen set of action candidates were evaluated, frontier-based goals and loop closing or revisiting configurations in the traversed path.

## 5 Preliminary work

The firsts results of our research on using information metrics in Active SLAM are reported below.

### Information gain over the configuration space

Departing from the work of Valencia et al [40], we explored a new approach [2, 5] to evaluate the effect of not only a few path candidates, but of a dense sampling of all the possible robot configurations of the known environment.

This means, computing a dense estimate of joint entropy reduction for the entire configuration space. To do so, we compute directly the map and path entropy reduction estimates for each possible robot configuration using frontier-visibility. Using the fact that sensor beams repeat for many orientations at the same position, and using very efficient kernel convolutions, we are able to compute these estimates very fast and densely. To illustrate the idea, fig. 4 shows the steps of the visibility estimation process. By estimating the entropy decrease densely, i.e., for all robot configurations, we can now evaluate motion policies not only for a few candidate configurations [28, 39, 40], but for the entire C-space (see Fig. 5.a). The method uses PoseSLAM as its estimation backbone and hence performs path entropy reduction during state update when poses are revisited.

In addition, instead of the joint entropy approximation used in [40], a new factor in the second summand is added relating to the probability weights

$$H(\mathbf{x}, \mathbf{m}|\mathbf{z}) \approx H(\mathbf{x}|\mathbf{z}) + \alpha(p(\mathbf{x}|\mathbf{z}))H(\mathbf{m}|\boldsymbol{\mu}, \mathbf{z}). \quad (36)$$

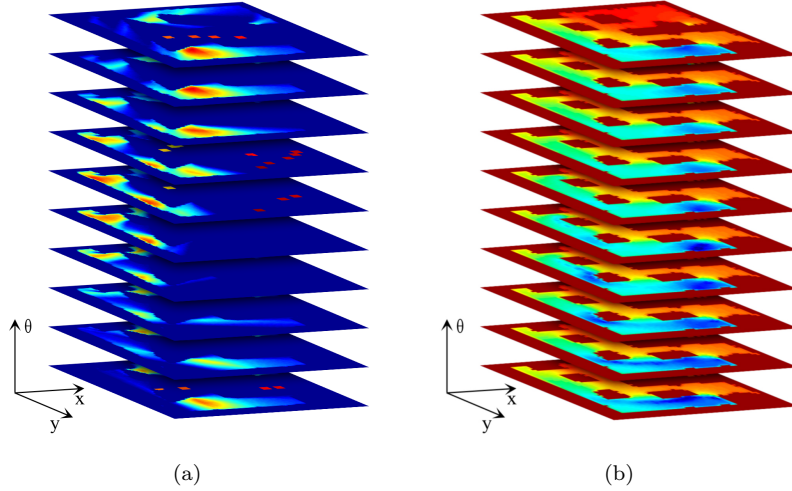


Figure 5: (a) Dense joint entropy decrease and (b) Information-based potential field in two different instances of time in the cave-like map.

This factor  $\alpha(p(\mathbf{x}|\mathbf{z}))$  has an intuitive meaning, exploratory trajectories that depart from well localized priors produce more accurate maps than explorations that depart from uncertain locations. In fact, sensor readings coming from robot poses with large marginal covariance may spoil the map adding bad cell classifications, i.e., adding entropy. Our approach is to weight the entire entropy reduction map with the inverse of the determinant of the marginal covariance at the current configuration  $\alpha(p(\mathbf{x}|\mathbf{z})) = |\Sigma_{nn}|^{-1}$ .

Exploratory trajectories that depart from uncertain configurations will be weighted giving predominance to the path entropy reduction term and viceversa. In this way, we modulate the importance of the exploratory and relocalization behavior as shown in [5], improving the strong sensibility with respect to motion noise.

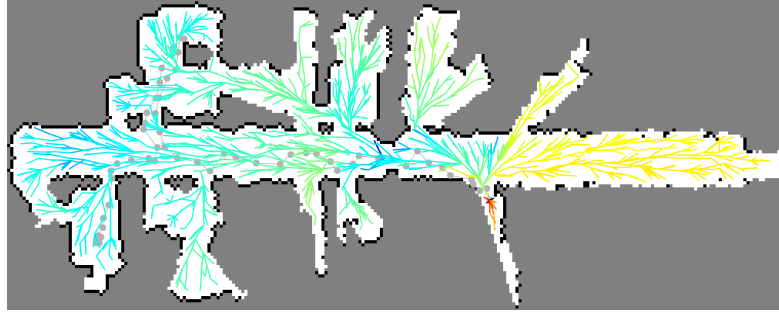
Two exploration strategies that make use of this densely computed entropy reduction estimate in the whole C-space are provided. Both strategies have as objective to drive the robot to minimize both map and path entropies, i.e., maximizing coverage while maintaining the robot well localized. The first strategy converts the entropy decrease estimate to an information-based potential field, and searches its minima using a gradient descent method. The second approach chooses the maximum informative configurations as goals and plans a trajectory to them through the free space with the aid of an RRT\* tree.

Making use of harmonic functions, a potential field is obtained from the joint entropy decrease estimation (see Fig. 5.b). Since the path planned is the result of a gradient descent over it, not only the last pose of the path is a (local) minima (maxima in terms of information) but also the path taken to reach such pose goes thru the most informative configurations.

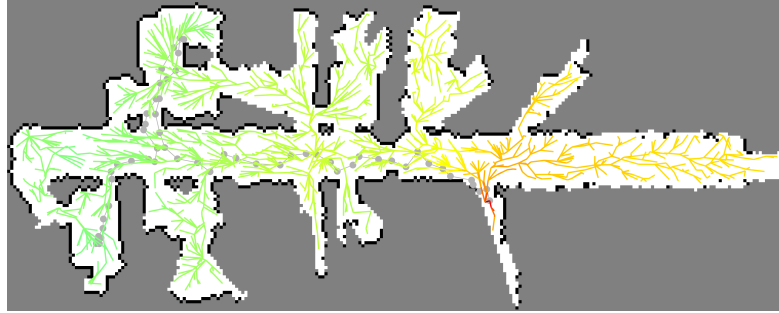
## Information gain over the action space

But evaluating joint map and path entropy decrease (i.e. information gain) estimates densely for the entire configuration space finely discretized does not take into account the changes in entropy induced during path execution and only the estimated change at the final robot configuration. Not taking into account the effect of the whole path implies renouncing to optimality. It would make more sense to evaluate the effect of the whole path and not just at the goals.





(a)  $dRRT^*$



(b)  $eRRT^*$

Figure 6: Two examples of the resulting path candidates using the two different cost functions proposed. Color of branches represent the value of the information gain per meter travelled (Eq. 37) of the path that ends in the corresponding node. Red paths indicate better candidates. In gray, the Pose SLAM nodes.

In our last work [4], we add to the equation the distance traveled, so that joint entropy minimization is not independent of the path length. In this way we can compare path candidates  $a = \{u_{n+1}, \dots, u_N\}$  of different length when searching for the optimal entropy reduction action set and also minimize the distance traveled. Consequently, we propose the minimization of the joint entropy change divided by the length of the path candidate, maximizing information gain per meter traveled

$$a^* = \operatorname{argmin} \frac{H(\mathbf{x}_{1:N}, \mathbf{m}_N | \mathbf{z}_{1:N}) - H(\mathbf{x}_{1:n}, \mathbf{m}_n | \mathbf{z}_{1:n})}{\operatorname{dist}(a)}. \quad (37)$$

The estimation should not overcount the joint entropy decrease of a path, i.e., a map cell discovered in previous poses of the path candidate should not be accounted for twice. This is taken into account in our computation of map entropy reduction. To that end, we extend our previous work with an efficient method to accumulate joint entropy decrease estimation over a travelled path. Using this estimation, two alternatives to search for path candidates are presented. First, it is used to evaluate the path candidates generated by a standart RRT\* ( $dRRT^*$ ), and second, it is used also as a cost function of the RRT\* algorithm in order to make it grow looking for the optimal paths ( $eRRT^*$ ).

Figure 6 shows two examples of both alternatives. While  $dRRT^*$  provides distance-optimal paths to reach several free poses,  $eRRT^*$  provides several information gain per meter travelled optimal paths. The evaluation of all the candidates generated in terms of information gain per

meter travelled are depicted in the same color scale showing that obviously the  $eRRT^*$  action set is better. However, the price to pay is about an order of magnitude higher with respect to computational cost.

## 6 Work plan

We now present the various tasks that need to be carried out to accomplish the objectives of this research plan, and an intended Gantt chart for their execution.

### Task 1: Literature review

This task is meant to gain a general overview on the state-of-art of SLAM and Active SLAM. The SLAM literature review will focus on least squares minimization methods, pruning and sparsification, and information theory applied to those problems. Also, the latest hierarchical SLAM representations will be reviewed. The Active SLAM literature review will include those publications related to exploration strategies, information-based active SLAM and planning.

### Task 2: SLAM front-end implementation: WOLF

Since the main objective of the thesis is not to implement a new SLAM back-end but instead, to use information gain metrics with state-of-art SLAM back-ends; the implementation of a SLAM front-end capable of managing several available SLAM back-ends is a principal task of the thesis.

### Task 3: Information gain in SLAM

#### Task 3.1: Pruning measurements

This task entails the formulation of information gain metrics to evaluate the information content of any given measurement, i.e, the contribution of such specific measurement to the solution of the whole estimation problem. This metric will be used to discard those measurements that least contribution to the solution of the problem making it more sparse, and hence with a lower computational burden.

#### Task 3.2: Pruning variables

The quantification of the information content encoded in each SLAM variable, and in the correlation between variables will be tackled in this task. Both robot poses and landmarks will be taken into consideration. Identifying the most redundant variables, and marginalizing or removing them, will allow us to reduce the size of the problem, and hence the computational cost of its solution.

#### Task 3.3: Information-hierarchical SLAM

Once information-theoretic decisions for the pruning of measurements and variables are in place, we will develop a hierarchical approach in which these discardable measurements and variables can be encoded in lower layers of the hierarchy. In this way, their contribution to the solution of the problem is not completely lost, but instead deferred within the estimation pipeline.

This hierarchy will also allow us to gauge the level of information content to be included in the problem, according to the computational resources that can be committed to its solution. In

this way, suboptimal solutions with smaller computational cost can still be obtained focusing on different parts of the problem hierarchy.

#### **Task 3.4: Relinearization**

In this task we will evaluate the effect of relinearization on the accuracy of the solution. On one side we have filtering, with only one linearization step, which is known to be suboptimal. On the other extreme we have iterative Gauss Newton providing a full nonlinear least squares solution which has better accuracy but is significantly more expensive computationally, and all heuristics and techniques in between described in the state of the art, which linearize only part of the problem and upon demand. We will provide principled ways to balance this linearization gauge together with the other time consuming tasks of variable and measurement pruning.

#### **Task 3.5: Information gain rate**

The above-mentioned techniques, observation dropping and variable pruning, produce faster solutions of the problem at the expense of sub-optimality. In contrast, relinearization is an expensive process that can increase the accuracy of the solution. Hence, the information content that is lost in each of latter steps and the improvement in accuracy coming from the the former can be related to the increase or decrease in speed in reaching a solution. We call this relations information gain rate, and will provide formal ways to represent it and to exploit optimally in the solution of SLAM and Active SLAM problems.

### **Task 4: Information gain in Active SLAM**

#### **Task 4.1: Estimating the information gain over configuration space**

This task includes the development of an efficient method to densely approximate the map and localization entropy change over the configuration space and some planning strategies to exploit that.

#### **Task 4.2: Estimating the information gain over action space**

This task entails the development of an Active SLAM method approaching it as an action selection problem. It allows to evaluate not only the final pose but all the intermediate poses of all candidate paths.

It includes the modification of the efficient approximation of the localization and map entropy change developed in the previous task to allow evaluating several path candidates to optimally build the exploration set.

### **Task 5: Elaboration of the dissertation**

The last task of this research is dedicated to the elaboration of the dissertation and the preparation of its public defense.

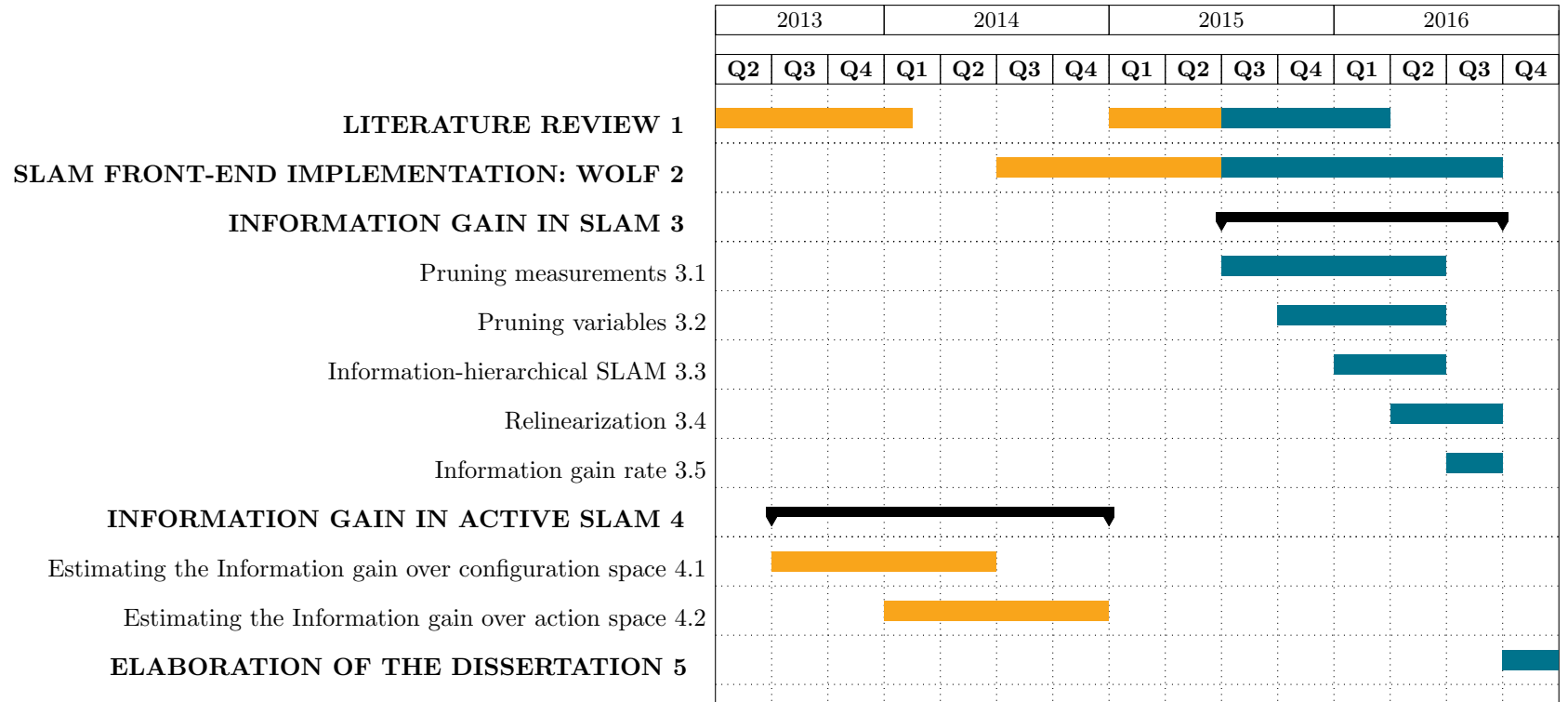


Figure 7: Work plan of the proposed work



## References

- [6] 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. Robotics Autom.*, vol. 17, no. 3, pp. 229–241, Jun. 2001.
- [7] 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. Robotics Res.*, vol. 23, no. 7-8, pp. 693–716, Jul. 2004.
- [8] V. Ila, J. M. Porta, and J. Andrade-Cetto, “Information-based compact Pose SLAM,” *IEEE Trans. Robotics*, vol. 26, no. 1, pp. 78–93, Feb. 2010.
- [9] R. Smith, M. Self, and P. Cheeseman, “A stochastic map for uncertain spatial relationships,” in *Proc. 4th Int. Sym. Robotics Res.*, Santa Cruz, 1987, pp. 467–474.
- [10] J. J. Leonard, H. F. Durrant-Whyte, and I. J. Cox, “Dynamic map building for an autonomous mobile robot,” *Int. J. Robotics Res.*, vol. 11, no. 4, pp. 286–292, 1992.
- [11] R. M. Eustice, H. Singh, and J. J. Leonard, “Exactly sparse delayed-state filters for view-based SLAM,” *IEEE Trans. Robotics*, vol. 22, no. 6, pp. 1100–1114, Dec. 2006.
- [12] F. Dellaert and M. Kaess, “Square root SAM: Simultaneous localization and mapping via square root information smoothing,” *Int. J. Robotics Res.*, vol. 25, no. 12, pp. 1181–1204, 2006.
- [13] M. Kaess, A. Ranganathan, and F. Dellaert, “iSAM: Incremental smoothing and mapping,” *IEEE Trans. Robotics*, vol. 24, no. 6, pp. 1365–1378, 2008.
- [14] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. J. Leonard, and F. Dellaert, “iSAM2: Incremental smoothing and mapping using the bayes tree,” *Int. J. Robotics Res.*, vol. 31, no. 2, pp. 216–235, 2011.
- [15] T. Davis, J. Gilbert, S. Larimore, and E. Ng, “A column approximate minimum degree ordering algorithm,” *ACM T. Math. Software*, vol. 30, no. 3, pp. 353–376, 2004.
- [16] G. Grisetti, R. Kummerle, C. Stachniss, U. Frese, and C. Hertzberg, “Hierarchical optimization on manifolds for online 2D and 3D mapping,” in *Proc. IEEE Int. Conf. Robotics Autom.*, Anchorage, May 2010, pp. 273–287.
- [17] L. Polok, V. Ila, M. Solony, P. Smrz, and P. Zemcik, “Incremental block Cholesky factorization for nonlinear least squares in robotics,” in *Robotics: Science and Systems*, Berlin, Jun. 2013.
- [18] K. Konolige, G. Grisetti, R. Kummerle, B. Limketkai, and R. Vincent, “Efficient sparse pose adjustment for 2D mapping,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Taipei, Oct. 2010, pp. 22–29.
- [19] R. Kummerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard, “g<sup>2</sup>o: A general framework for graph optimization,” in *Proc. IEEE Int. Conf. Robotics Autom.*, Shanghai, May 2011, pp. 3607–3613.
- [20] K. Konolige, “Sparse sparse bundle adjustment,” in *Proc. British Mach. Vis. Conf.*, Dundee, Aug. 2011, pp. 102.1–102.11.

- [21] S. Leutenegger, S. Lynen, M. Bosse, R. Siegwart, and P. Furgale, “Keyframe-based visual-inertial odometry using nonlinear optimization,” *Int. J. Robotics Res.*, vol. 34, no. 3, pp. 314–334, 2015.
- [22] M. Li and A. I. Mourikis, “High-precision, consistent EKF-based visual-inertial odometry,” *Int. J. Robotics Res.*, vol. 32, no. 6, pp. 690–711, 2013.
- [23] H. Kretzschmar and C. Stachniss, “Information-theoretic compression of pose graphs for laser-based SLAM,” *Int. J. Robotics Res.*, vol. 31, no. 11, pp. 1219–1230, 2012.
- [24] C. Chow and C. Liu, “Approximating discrete probability distributions with dependence trees,” *IEEE Trans. Inform. Theory*, vol. 14, no. 3, pp. 462–467, 1968.
- [25] Y. Wang, R. Xiong, Q. Li, and S. Huang, “Kullback-Leibler divergence based graph pruning in robotic feature mapping,” in *Proc. Eur. Conf. Mobile Robots*, Barcelona, Sep. 2013, pp. 32–37.
- [26] M. Mazuran, G. D. Tipaldi, L. Spinello, and W. Burgard, “Nonlinear graph sparsification for SLAM,” in *Robotics: Science and Systems*, Berkeley, Jul. 2014, pp. 1–8.
- [27] S. Choudhary, V. Indelman, H. Christensen, and F. Dellaert, “Information-based reduced landmark SLAM,” in *Proc. IEEE Int. Conf. Robotics Autom.*, Seattle, May 2015.
- [28] B. Yamauchi, “A frontier-based approach for autonomous exploration,” in *IEEE Int. Sym. Computational Intell. Robot. Automat.*, Monterrey, 1997, pp. 146–151.
- [29] W. Burgard, M. Moors, D. Fox, R. Simmons, and S. Thrun, “Collaborative multi-robot exploration,” in *Proc. IEEE Int. Conf. Robotics Autom.*, San Francisco, Apr. 2000, pp. 476–481.
- [30] J. Faigl and M. Kulich, “On determination of goal candidates in frontier-based multi-robot exploration,” in *Proc. Eur. Conf. Mobile Robots*, Barcelona, Sep. 2013, pp. 210–215.
- [31] L. Freda and G. Oriolo, “Frontier-based probabilistic strategies for sensor-based exploration,” in *Proc. IEEE Int. Conf. Robotics Autom.*, Barcelona, Apr. 2005, pp. 3881–3887.
- [32] R. Shade and P. Newman, “Choosing where to go: Complete 3D exploration with stereo,” in *Proc. IEEE Int. Conf. Robotics Autom.*, Shanghai, May 2011, pp. 2806–2811.
- [33] F. Amigoni and V. Caglioti, “An information-based exploration strategy for environment mapping with mobile robots,” *Robotics Auton. Syst.*, vol. 58, no. 5, pp. 684–699, 2010.
- [34] A. Makarenko, S. B. Williams, F. Bourgault, and H. F. Durrant-Whyte, “An experiment in integrated exploration,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Lausanne, Oct. 2002, pp. 534–539.
- [35] H. J. S. Feder, J. J. Leonard, and C. M. Smith, “Adaptive mobile robot navigation and mapping,” *Int. J. Robotics Res.*, vol. 18, pp. 650–668, 1999.
- [36] F. Bourgault, A. Makarenko, S. Williams, B. Grocholsky, and H. Durrant-Whyte, “Information based adaptative robotic exploration,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Lausanne, Oct. 2002, pp. 540–545.
- [37] T. Vidal-Calleja, A. Sanfeliu, and J. Andrade-Cetto, “Action selection for single camera SLAM,” *IEEE Trans. Syst., Man, Cybern. B*, vol. 40, no. 6, pp. 1567–1581, Dec. 2010.

- [38] L. Torabi, M. Kazemi, and K. Gupta, “Configuration space based efficient view planning and exploration with occupancy grids,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, San Diego, Nov. 2007, pp. 2827–2832.
- [39] C. Stachniss, G. Grisetti, and W. Burgard, “Information gain-based exploration using Rao-Blackwellized particle filters,” in *Robotics: Science and Systems I*, Cambridge, Jun. 2005, pp. 65–72.
- [40] R. Valencia, J. Valls Miró, G. Dissanayake, and J. Andrade-Cetto, “Active Pose SLAM,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Vilamoura, Oct. 2012, pp. 1885–1891.