Proceedings of ICRA 2006

2006 IEEE International Conference on Robotics and Automation

May 15-19, 2006

Orlando, Florida, USA

06CH37729D

ISBN: 0-7803-9506-9

Download Microsoft Media Player

Download Apple QuickTime Player

Main Menu

Search this DVD-ROM

IEEE

EXIT

Tu-PM1-02

Mobile Robot Mapping

Proceedings of the 2006 IEEE International Conference on Robotics and Automation
Orlando, Florida - May 2006

Consistent Observation Grouping for Generating Metric-Topological Maps that Improves Robot Localization*

Jose Luis Blanco, Javier Gonzalez, Juan Antonio Fernández-Madrigal
Dept. of System Engineering and Automation
University of Malaga
Malaga, Spain
{jlblanco,jgonzalez,jafma}@ctima.uma.es

Abstract – **Recently, hybrid maps that combine metric and topological world information have been proposed as a powerful representation of mobile robot environments. Among others, these maps are of special interest for efficiently managing large-scale environments, and for accurate localization. For achieving that, local geometric maps are stored in the nodes of a graph-based global map. In this paper we present a novel approach for automatically obtaining those local maps from observations. The method considers the space sensed in each observation as a node of a graph with arcs representing the space overlap between observations. The recursive partition (cut) of this graph produces groups of strongly connected nodes from which consistent local maps for accurate localization are derived. The proposed partition technique is well-grounded in the spectral graph theory of, and it is formulated for any type of sensor observation. We depict an implementation for grouping 2D laser scans, and show experimental results with real data that demonstrate the performance of the method.**

Index Terms* – *Graph partitioning, map building, mobile robots, topological maps.

I. INTRODUCTION

When a mobile robot moves in an unknown environment it must deal simultaneously with both localization and mapping, namely the SLAM problem. This topic has received a great attention by the robotics community in last years, being proposed a variety of methods to approach the problem. According to the kind of world model they use, these methods can be classified into metric ones, which use geometrical information ([6],[7],[9]), or topological ones, which represent the world with a graph whose nodes usually represent distinctive places ([1],[12]). Recently, hybrid models that combine both information types have being proposed as a promising solution to deal with larger and more complex real robot environments. Typically, these approaches attach a local geometrical map (suitable for robot localization) to the nodes of a graph-based world representation ([2],[3],[13]). A crucial point then is to decide how to distribute the mapped environment between those local maps. From the different propositions reported in the literature, the following ones are of special significance: the Atlas framework [2], where a new local map is built and updated from observations until localization performs poorly; and, more recently, the hierarchical SLAM presented in [3], where new sensed features are integrated into local maps until a given number is reached. However, none of these works provide a mathematically grounded solution for this problem. There are some interesting works aimed to achieve efficiency in SLAM by hierarchically dividing the map into local regions and subregions ([5], [10]). These works are based on the statistical dependency of landmark observations while ours is general enough to cope with any type of sensed data. In particular, in this paper we propose an implementation for dense range scans.

The method proposed here is based on the spectral graph theory for grouping together robot observations that give rise to consistent local maps for efficient and accurate pose estimation. The method considers the space sensed in each observation as a node of a graph whose arcs (edges) represent the *sensed-space overlap* (SSO) between two observations (see Fig. 1). The partition of this *observation graph* through a recursive minimum normalized cut produces groups of strongly connected nodes from which the local maps are obtained. This procedure yields near-optimal graph with respect to some goodness measure.

The observations we are referring to may come from any sensor, but they must consist of a homogeneous distribution of physical features: for example, the set of 2D points from a laser range scan (which is the particular case implemented in this work), or 3D points extracted from intensity images in some way, i.e. stereo, structure from

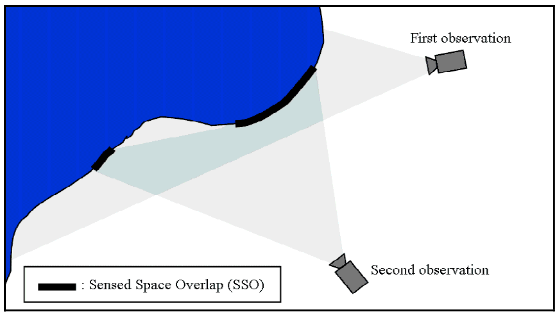

Fig.1 The sensed-space overlap (SSO) is a measurement of the common part of the environment captured by two observations.

* This work was supported by the Spanish Government under research contract DPI2005-01391.

motion, etc. The SSO of two observations is calculated through a registration process that takes into account these environmental features as well as the corresponding robot poses, as provided by a localization module (not addressed here).

The remainder of this paper is organized as follows. In section II we present the general partition problem for generic graphs. Section III describes how a set of robot observations is translated into a graph. Some implementation issues when using laser scans and experimental results are discussed in section IV. Finally, some conclusions and future work are outlined.

II. Spectral Partition of a Graph

In this section we first present the basis for the bisection of a graph using the spectral approach. Then its generalization for producing any number of subgraphs is described.

A. Spectral Bisection of a Graph

Let $G = (V, E)$ be an undirected, weighted graph, where V is the set of vertices or nodes and E the set of weighted edges or arcs, using non-negative weight values. Let $\mathbf{W}$ be a symmetric square matrix of dimension $|V|$, where each of its elements W_{ij} is the weight of the arc between nodes i and j. According to the definition introduced by Shi and Malik in [11], the *normalized cut of a graph* V into two subgraphs $A\subset V$ and $B\subset V$, with $A\cup B=V$ and $A\cap B=\emptyset$, is defined as:

$$Ncut(A,B)=\frac{cut(A,B)}{assoc(A,V)}+\frac{cut(A,B)}{assoc(B,V)} \tag{1}$$

where

$$cut(A,B)=\sum_{u\subset A,v\subset B} W_{uv} \tag{2}$$

is the cut of the graph into the subgraphs A and B, which measures their *inter-group cohesion* and

$$assoc(A,V)=\sum_{u\subset A,v\subset V} W_{uv} \tag{3}$$

is the association of the group A (similarly for B), which estimates the *intra-group cohesion,* that is, the connection "strength" of all nodes within A with the whole graph (including A).

Minimizing (2) in a graph partition (thus, denominated "*min-cut partition*") tends to generate groups of no practical usefulness for some applications, since usual resulting groups includes the least connected nodes of the graph. It is of much more interest to get subgraphs with a balance between both, the inter-group and the intra-group cohesion, which is achieved by minimizing the *Ncut*, defined in (1). Thus, the *minimum normalized cut* of a graph V (*min-Ncut*) is given by:

$$\arg\min_{A,B}\ Ncut(A,B) \tag{4}$$

The range of values of an *Ncut* can be derived from the fact that:

$$assoc(A,V)=cut(A,B)+assoc(A,A) \tag{5}$$

which implies that for the worst possible cut (the one in which nodes in a group are connected only to the other group), the values of *assoc(A,A)* and *assoc(B,B)* are zero, therefore:

$$assoc(A,V)\big|_{\min} = assoc(B,V)\big|_{\min} = cut(A,B) \tag{6}$$

and the maximum *Ncut* value becomes:

$$Ncut(A,B)\big|_{\max}=\frac{cut(A,B)}{cut(A,B)}+\frac{cut(A,B)}{cut(A,B)}=2 \tag{7}$$

On the other hand, the minimum possible value of a cut is zero when there are no connections between the two groups. Thus, the *Ncut* provides a numerically well defined measure of the goodness of a partition.

Finding the exact *min-Ncut* bisection is a computationally intractable problem, in fact a NP-complete one. Following the proposition by Shi and Malik for image segmentation [15], we will use an approximate approach based on spectral bisection of graphs, which produces near-optimal cuts. This method relies on solving a generalized eigenvalue system, as it is summarized next.

Let $\boldsymbol{x}$ be the bisection indicator vector with dimension $N=|V|$, where $x_i = -1$ if node i falls into group B, or 1 if it falls into A. Let $\boldsymbol{d}$ be the vector with the sum of adjacent arcs weights for each node, that is, $d_i = \sum_j W_{ij}$. We build a diagonal matrix $\mathbf{D}$ with $\boldsymbol{d}$ as its diagonal. It can be shown that the *min-Ncut* problem can be rewritten as:

$$\arg\min_{\mathbf{x}} \mathrm{Ncut}(\mathbf{x}) = \arg\min_{\mathbf{y}} \frac{\mathbf{y}^T(\mathbf{D}-\mathbf{W})\mathbf{y}}{\mathbf{y}^T\mathbf{D}\,\mathbf{y}} \tag{8}$$

where $\boldsymbol{y} = (\mathbf{1}+\boldsymbol{x}) - b\,(\mathbf{1}-\boldsymbol{x})$, with $\mathbf{1}$ a $N \times 1$ unity vector, and

$$b=\frac{\sum_{x_i>0} d_i}{\sum_{x_i<0} d_i} \tag{9}$$

Ideally, the elements of vector $\boldsymbol{y}$ should take just two discrete values, since x_i takes the values $\{-1,1\}$. However, if this condition is relaxed and $\boldsymbol{y}$ is allowed to be real valued, then (6) is no longer discrete and can be minimized by solving the generalized eigensystem:

$$(\mathbf{D}-\mathbf{W})\mathbf{y}=\lambda\mathbf{D}\mathbf{y} \tag{10}$$

where $\mathbf{D}-\mathbf{W}$ is a well-known term, namely the Laplacian matrix of the graph ([4],[8]). The above equation can be rewritten as a standard eigensystem using $\boldsymbol{z} = \mathbf{D}^{1/2}\boldsymbol{y}$:

$$\mathbf{D}^{-1/2}(\mathbf{D}-\mathbf{W})\mathbf{D}^{-1/2}\mathbf{z}=\lambda\mathbf{z} \tag{11}$$

It can be shown that $\boldsymbol{z}_0 = \mathbf{D}^{1/2}\mathbf{1}$ is the eigenvector corresponding to the smallest eigenvalue of (11) ("the smallest eigenvector" from now on), which is zero. Translating back this result to the original system in (10), we have that $\boldsymbol{y}_0=\mathbf{1}$ is the smallest eigenvector of (11). Since the fraction in (8) is a Rayleigh quotient [8], and its eigenvectors are orthogonal[2], then both (10) and (11) are

[2] Since the Laplacian matrix $\mathbf{D}-\mathbf{W}$ is positive semidefinite, $\mathbf{D}^{-1/2}(\mathbf{D}-\mathbf{W})\mathbf{D}^{-1/2}$ is symmetric positive semidefinite, thus its eigenvectors are orthogonal.

minimized with the next smallest eigenvector. Thus, we have that solving the *min-Ncut* expressed in (8) is equivalent to finding the second smallest eigenvector, $\boldsymbol{y_1}$, of (10).

The only approximation made on the above result is that the components of the eigenvector $\boldsymbol{y_1}$, from which we have to decide the group, will not take just two values, but any real number. Obviously, this complicates the bisection criterion; however, there still exists a clear distinction between the two groups, as it can be observed from the real examples shown in Fig. 2. Two different criteria seem plausible for assigning each node to a group: (i) look at the sign of each component of the eigenvector; and (ii) take the mean value of eigenvector as a threshold for the partition. The latest is the criterion we have used in our implementation, since it has demonstrated to give more precise partitions that the other.

B. *Partitioning a graph in k-groups*

The method presented above provides a solution to the graph bisection problem; however, for dividing a graph into a variable number of subgraphs, this method must be generalized somehow. An easy and effective way of achieving that is to recursively apply bisection to any subgraph as long as two clearly differentiated groups are obtained (as it occurred in the examples of Fig. 2). The *min-Ncut* value for a given bisection is a well-grounded measure of the goodness of the cut. The *Ncut* of a graph measures the inter-group cohesion of the resulting subgraphs, inversely scaled by the intra-group cohesions with values in the range [0,2]. Values close to zero indicate almost no connection between groups (a good partition), while values near 2 indicate that the groups are more strongly connected to each other than with themselves (not to be partitioned). Therefore, an intermediate value must be established as a threshold to decide accepting the bisection or not.

Typically, this value has been set heuristically for each application (as in [11],[14]), however, to keep the method free of any subjective ad-hoc parameter, we propose to use a fixed threshold value of one (the center of the range interval), which is supported by the following. The intra-group cohesion of a subgraph *A* is given by *assoc(A,V)* as defined in (3), while the inter-group cohesion of A with the rest of the graph, let say B, is given by *cut(A,B)* as defined in (2). When the inter-group cohesion, given by *cut(A,B)* is below certain averaged estimation of the two intra-group cohesions (*assoc(A,V)* and *assoc(B,V)*), the bisection should not be done. We propose here to use the geometric mean, which leads to a constant threshold value. Thus we set the threshold as the following value:

$$cut(\mathrm{A,B})^{\mathrm{T}} = \sqrt{assoc(\mathrm{A,A}) \cdot assoc(\mathrm{B,B})} \qquad (12)$$

which can be combined with (1), giving:

$$Ncut(\mathrm{A,B})^{\mathrm{T}} = \frac{cut(\mathrm{A,B})^{\mathrm{T}}}{assoc(\mathrm{A,V})} + \frac{cut(\mathrm{A,B})^{\mathrm{T}}}{assoc(\mathrm{B,V})} \qquad (13)$$

$$= \frac{cut(\mathrm{A,B})^{\mathrm{T}}}{cut(\mathrm{A,B})^{\mathrm{T}} + assoc(\mathrm{A,A})} + \frac{cut(\mathrm{A,B})^{\mathrm{T}}}{cut(\mathrm{A,B})^{\mathrm{T}} + assoc(\mathrm{B,B})}$$

$$= \frac{\sqrt{\alpha\beta}}{\sqrt{\alpha\beta}+\alpha} + \frac{\sqrt{\alpha\beta}}{\sqrt{\alpha\beta}+\beta} = \frac{2\alpha\cdot\beta + \sqrt{\alpha\beta}(\alpha+\beta)}{2\alpha\cdot\beta + \sqrt{\alpha\beta}(\alpha+\beta)} = 1$$

where for clarity we have used α=*assoc(A,A)* and β=*assoc(B,B)*. Hence, we obtain a value of 1 to decide whether to apply the bisection or not. The whole procedure is summarized as follows:

```
RecursivePart( G ) → { P }
begin
    SpectralBisection(G) → {A, B}, Ncut
    if (N-cut<1) then
        P = { RecursivePart(A), RecursivePart(B) }
    else
        P = G
    end-if
end
```

In the example of Fig. 3, this procedure is applied to the observations graph, which is divided into groups {G1} and {G2,G3}, in the first iteration. The latter is partitioned again because it has a minimum *Ncut* below one. The resulting groups G2 and G3 are no longer partitioned since a minimum *Ncut* over each of them gives values greater than one.

III. THE OBSERVATION GRAPH PARTITIONING PROBLEM

Let $O=\{o_1,\ldots o_N\}$ be the set of observations taken by a robot during a navigation route. These observations consist of the location of environmental features, such as points, segments, patches, etc., from which a global geometric map has been derived. The construction of this map involves the registration of these set of features and from that, the estimation of the robot poses $P=\{p_1,\ldots\ p_N\}$ in a common reference frame. When the map is very large two problems appear:

1) Robot localization becomes quite inexact due to the accumulation of errors during the map building process, including noise in the observations, erroneous features correspondence, errors in the localization algorithm, etc.
2) Dealing with an indivisible map is not efficient at all.

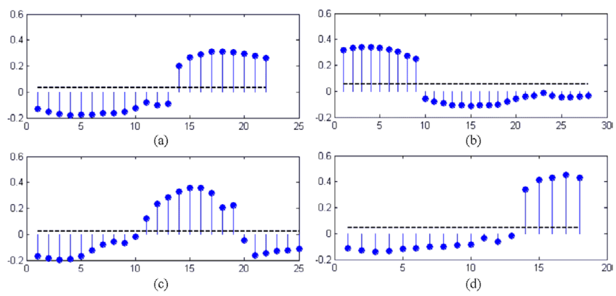

Fig.2 Some real examples for graph spectral bisections. The plots show the components of the eigenvectors which are used to choose the bisection. The length of these vectors coincides with the number of nodes in the graph. Observe that the mean of eigenvectors is a well defined threshold value for the graph bisection.

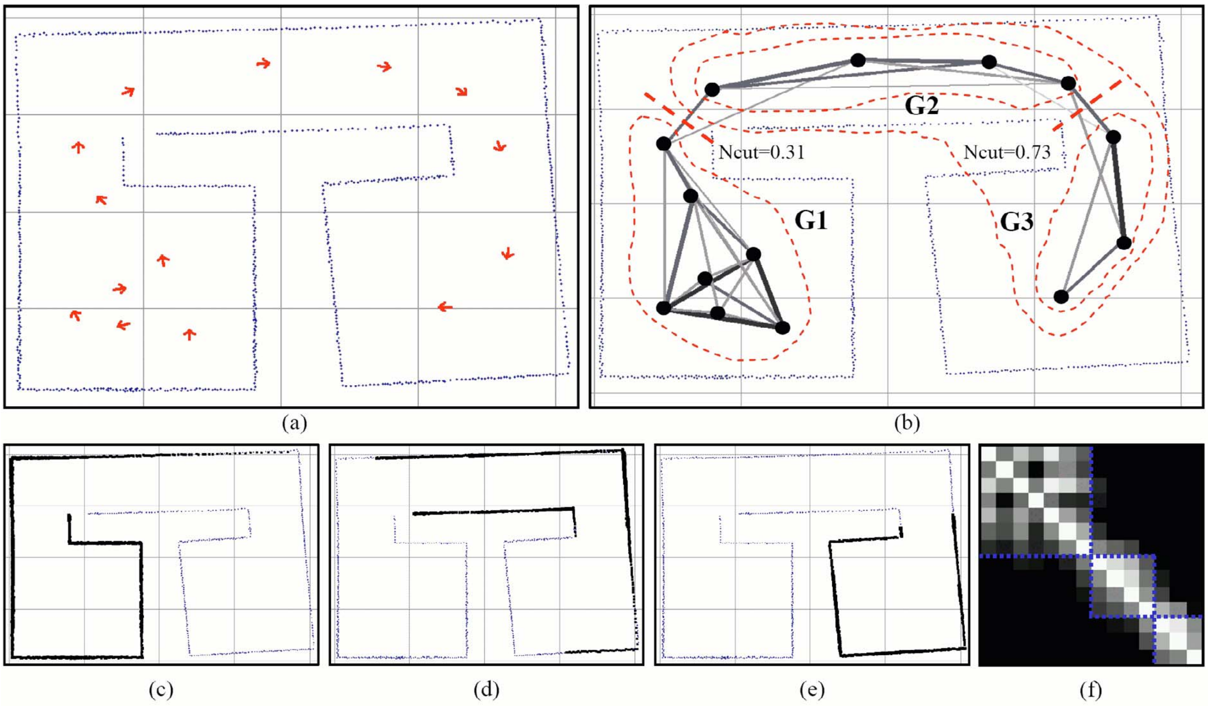

Fig.3. An illustrative example of the graph partitioning method over a 2D laser map. (a) The global map obtained from 14 observations, where arrows indicate the poses where the observations were taken from. Notice that the map presents some orientation errors. (b) The observation graph. Each node contains the sensed space data (scan), and an estimate for its pose. The darker the arc, the higher the SSO between the observations. The observation graph is recursively partitioned into three groups: firstly, it is divided into two groups {G1} and {G2,G3}, then, the latter group is partitioned again because it has a minimum *Ncut* below one. The local maps obtained from these groups are shown in (d), (c) and (e), respectively. In (f) the weights matrix of the associated graph is shown as an image with dotted squares for the three partitions.

Provided that both, the observations *O* and poses *P* are known, our objective is to group those observations into local maps accurate for robot localization. Then, these metric local maps can be arranged as nodes of a graph-based topological map, where the arcs represent the approximate relative transformation between the local maps.

We define the nodes n_i of an Observation Graph as:

$$N= \{n_1,\ldots n_N\} = \{(o_1, p_1),\ldots(o_N, p_N)\} \qquad (14)$$

It is assumed that an arc exists between any pair of nodes, and its weight represents the *sensed-space overlap* (SSO) of both observations calculated from their registration in space (using *O* and *P*), The SSO is evaluated through a normalized metric function:

$$\Psi: V \times V \rightarrow [0,1] \qquad (15)$$

which must fulfill the following two conditions:

- Reflexivity: The SSO of any observation with itself must be maximum, $\Psi(n_i,n_i) = 1$.
- Symmetry: It must be commutative for nodes order, $\Psi(n_i,n_j) = \Psi(n_j,n_i)$.

Therefore, we can define the graph weight matrix **W** simply as $W_{ij} = \Psi(n_i,n_j)$. Because of the two-above properties of Ψ, **W** becomes symmetric with a unity diagonal vector. Fig. 3 illustrates the graph partition method for a synthetic global map built from 14 2D laser scans as observations.

IV. Application to 2D laser scans

In this section we address the application of the proposed grouping observation method to laser range scans, dealing first with implementation details and then showing some experimental results.

We assume that the mobile robot has the capability to build a geometric map of the environment by applying a SLAM method, and consequently, the absolute robot poses from where the different scans were taken is available. Then, our approach consists of grouping these scans in a way that more consistent maps are obtained.

A. *Defining the SSO function Ψ for laser scans*

Let

$$C(p_i, q_j, \delta) = \begin{cases} 1 & \|p_i - q_j\| \le \delta \\ 0 & \|p_i - q_j\| > \delta \end{cases} \qquad (16)$$

be a Boolean function that, given a tolerance distance[3] δ, indicates whether two scanned points p_i and q_j from different observations o_i and o_j, respectively, are matched. To apply this function, points from both scans must be referred to a common coordinate system through the poses *P* also stored in the nodes. Then, the ratio of matched points of o_i against o_j is expressed as a function of the nodes:

[3] Please, observe that δ is not a parameter of the grouping algorithm, but of the matching process.

$$\Pi(n_i, n_j, \delta) = \frac{1}{M} \sum_{k=1}^{M} C(p_i^{\,k}, p_j^{\,k}, \delta) \qquad (17)$$

where M is the number of points in the scan o_i.

This function Π can not be directly used as a SSO function Ψ since it is not symmetric: $\Pi(n_i,n_j,\delta) \neq \Pi(n_j,n_i,\delta)$. This is caused by the variable scanning density which depends on distance: near objects are sampled denser than those distant apart. In Fig. 4 this effect is illustrated with two pairs of scans.

As a straightforward solution to this problem we have defined the SSO metric function Ψ as:

$$\Psi(n_i, n_j) = 1/2\left(\Pi(n_i, n_j, \delta) + \Pi(n_j, n_i, \delta)\right) = \Psi(n_j, n_i) \quad (18)$$

which satisfies the two required conditions.

In summary, given the set of scans *O* and poses *P*, the subsequent steps are applied:

1. By using (16), build the weight matrix **W**, with $W_{ij}=\Psi(n_i,n_j)$.
2. Build the diagonal matrix **D,** with elements $d_i = \sum_j W_{ij}$
3. Apply recursive *min-Ncut* bisection to obtain "consistent" groups of nodes. For each group, run a map building algorithm to produce a more precise local map, taking into account the scans and poses stored in the nodes.
4. Generate a metric-topological map where the above local maps are stored in the nodes of a graph and the arcs hold the relative transformation between them. This can be computed by averaging the differential poses between scans of different groups.

This process can be launched when the geometric map is considered to be not accurate enough. The most expensive stage is the map building process applied to those observations (nodes) resulting from the graph partition. This has been carried out by an external module (*map builder*) based on the method proposed by Lu and Milios in [7]. On the other hand, the evaluation of the scan overlaps through (14) is quite cheap since the poses between scans are known (provided by the *map builder*).

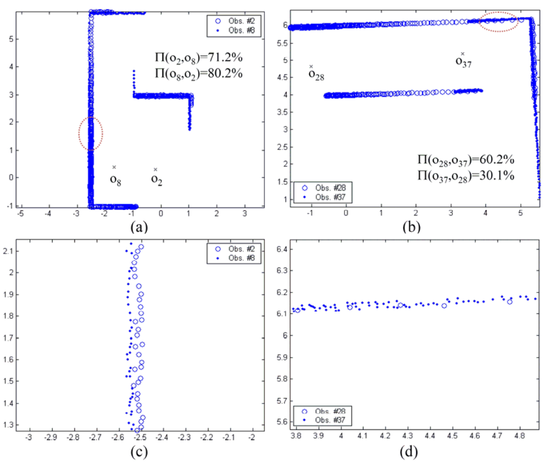

Fig.4. In (a) and (b) two pairs of scans are plotted with their matching values. The highlighted areas are detailed in (c) and (d) respectively, where the effects of the non-evenly spaced sampling are clearly visible. This explains the lack of symmetry of the *Π* function.

The hybrid map generated in this way allows the robot both to manage the space efficiently and to be localized locally more precisely. An important point that remains to be solved is how this hybrid map is updated when new observations (scans) of visited places are available. This is a tough, opened problem on which we (and many other researchers) are working currently. Nevertheless, this does not limit the applicability of the method presented in this paper, which fits well in a more comprehensive mapping-and-localization framework.

B. Experimental Results

The presented observation grouping approach has been extensively tested in a variety of real and simulated environments. For lack of space we show just one experiment in an office-like scenario, which is shown in Fig. 5. It comprises 77 laser scans taken through a route along a set of rooms and a corridor. From those observations our map builder produces the map plotted in Fig. 5(a), which presents some misaligned scans pairs: if a pair of scans has a low SSO, it is difficult to accurately align them. For example, a region in the corridor is zoomed in Fig. 5(b) from the map in Fig. 5(a), where inconsistencies can be found. The rest of maps in this figure are local maps generated by the presented approach, and its higher consistency can be seen, for example, with the zoom of Fig. 5(h) into (e). That zoomed area is the same than before, but it can be seen how most inconsistencies have disappeared. This illustrates that local maps generated with this approach are consistent maps, in the sense of no internal contradictions. It is evident at this point that more consistent local maps lead to more accurate and efficient robot localization, since just the required, precisely aligned, scans will be available in each local map.

It should be pointed out that, although the resulting local maps may look like rooms and corridors, our purpose is not detecting such specific structures but getting good maps for robot localization in a grounded way. Notice that, though dividing the space into rooms or corridors are of quite interest for interfacing with humans, it may not always be the best choice for other robotic tasks, such as localization. Finally, and to give an idea of the computational burden of the method, it can be noted that the whole procedure to generate the local maps above takes 0.24 sec. on a 2.8GHz Pentium IV processor.

V. Conclusions and Future Work

In this paper we have introduced a novel approach to cluster robot observations into groups that give rise to precise local maps. These local maps can be arranged as nodes of a graph-based topological map, where the arcs represent the approximate transformation between them. This hybrid representation of the environment enables the

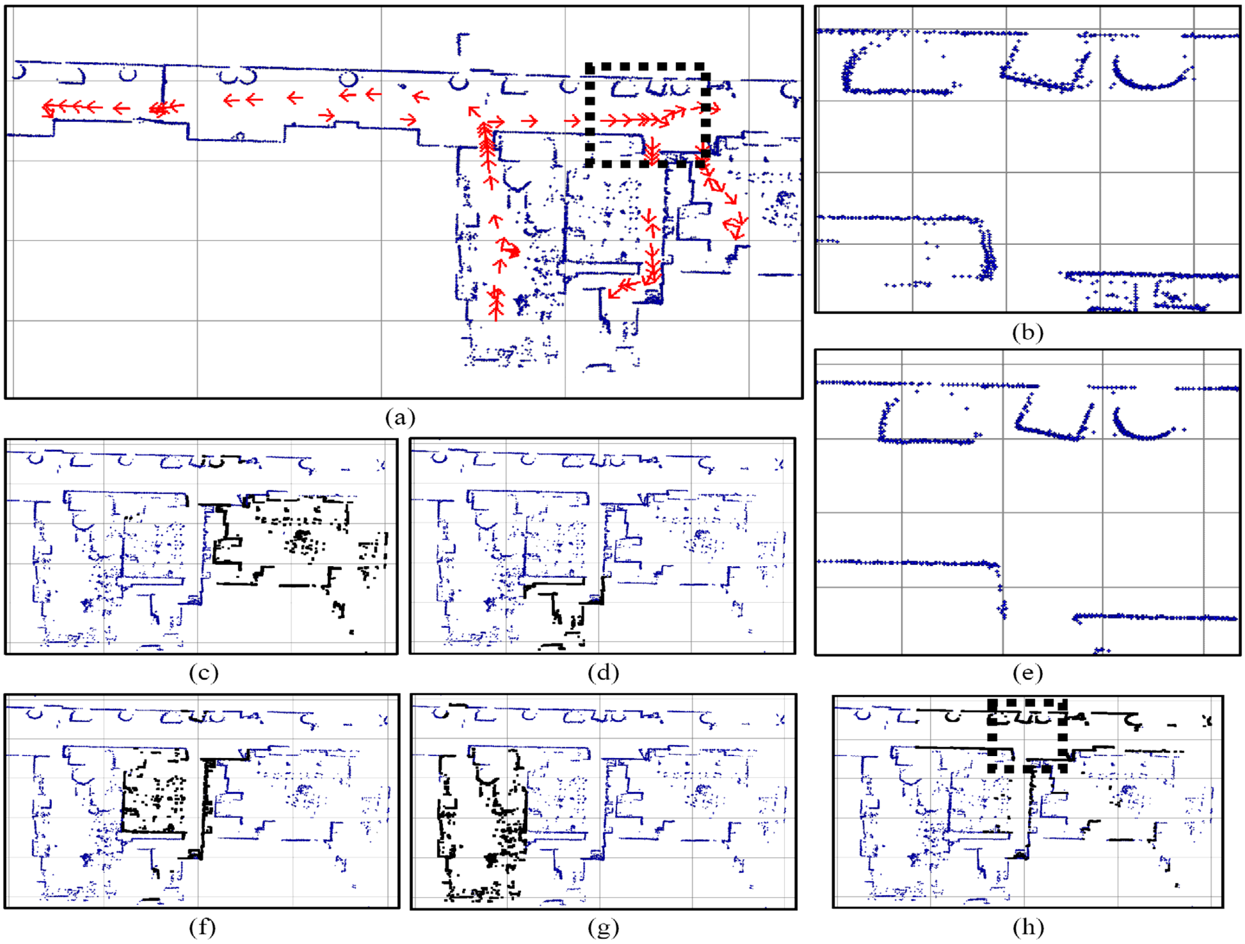

Fig.5. (a) The observations that constitute the map, whose poses are represented as arrows. The presented approach takes these 77 observations and produces eight groups, where (c),(d),(f) and (g) match exactly with the four rooms visited by the robot, while the rest are sections of the corridor, as the one in (h). Zooms for the same area are shown in (b) and (e), corresponding to the original map (a) and the generated local map (h), respectively, where it is clear that the later is more consistent. The reason for this is that only close observations, with a high *SSO* (and therefore, well-aligned) are grouped together.

robot to efficiently manage the space and to estimate its local pose more accurately. The grouping method used in this work is not new, since is a mathematical tool applied previously to other fields, but we believe that its successfully application to the problem addressed here is significant enough. This approach must be thought as part of a more comprehensive framework for localization and mapping that, among others, would manage the revisiting and updating problems, which are some of our current research goals.

References

[1] P. Beeson, N.K. Jong and B. Kuipers, "Towards Autonomous Topological Place Detection Using the Extended Voronoi Graph," *IEEE Conf. Robotics and Automation,* 2005.

[2] M. Bosse, P. Newman, J. Leonard, S. Teller, "An Atlas Framework for Scalable Mapping," *IEEE Conf. Robotics and Automation,* 2003.

[3] C. Estrada, J. Neira and J.D. Tardos, "Hierarchical SLAM: Real-Time Accurate Mapping of Large Environments," in *IEEE Transactions on Robotics*, vol. 21, pp.588-596, 2005.

[4] M. Fiedler, "Algebraic connectivity of graphs," *Czechoslovakian Mathematics Journal*, vol.23, pp.298-305, 1973.

[5] U. Frese, P. Larsson, T. Duckett, "A multilevel relaxation algorithm for simultaneous localization and mapping," in *IEEE Transactions on Robotics*, v. 21, pp. 196-207, 2005.

[6] J.S. Gutmann and K. Konolige, "Incremental Mapping of Large Cyclic Environments," *Conference on Intelligent Robots and Applications* (*CIRA*), Monterey, CA, 1999.

[7] F. Lu and E. Milios, "Globally consistent range scan alignment for environment mapping," *Autonomous Robots*, vol.4, pp.333-349, 1997.

[8] B. Mohan. "The Laplacian spectrum of graphs," *Graph Theory, Combinatorics, and Applications*, pp. 871-898, 1991.

[9] M. Montemerlo, S. Thrun, D. Koller and B. Wegbreit, "FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping Problem," *Proc. Of the National Conf. on Art. Intell. (AAAI)*, 2002.

[10]M. Paskin, "Thin Junction Tree Filters for Simultaneous Localization and Mapping," *Proc. 18th Int. Joint Conf. Artif. Intell.*, 2003.

[11]J. Shi and J. Malik, "Normalized Cuts and Image Segmentation," *IEEE Trans. Pattern Anal. Mach. Intell*, v.22, no.8, pp. 888-905, 2000.

[12]T. Soho, H. Ishiguro and T. Ishida, "Acquisition and propagation of spatial constraints based on qualitative information," *IEEE Trans. Pattern Anal. Mach. Intell*, v.23, no.3, pp.268-278, 2001.

[13]S. Thrun. "Learning Metric-Topological Maps for Indoor Mobile Robot Navigation," *Artificial Intelligence*, v.99, no.1, pp. 21-71, 1998.

[14]O. Veksler "Image Segmentation by Nested Cuts," *IEEE Conf. on Computer Vision and Pattern Recognition*, 2000.

[15]Y. Weiss, "Segmentation using eigenvectors: a unifying view," *Proc. of IEEE Int. Conf. on Computer Vision*, pp. 975–982, 1999.