Consistent Mapping of Multistory Buildings by Introducing Global Constraints to Graph-based SLAM

Michael Karg[1] Kai M. Wurm[2] Cyrill Stachniss[2] Klaus Dietmayer[1] Wolfram Burgard[2]

***Abstract*— In the past, there has been a tremendous advance in the area of simultaneous localization and mapping (SLAM). However, there are relatively few approaches for incorporating prior information or knowledge about structural similarities into the mapping process. Consider, for example, office buildings in which most of the offices have an identical geometric layout. The same typically holds for the individual stories of buildings. In this paper, we propose an approach for generating alignment constraints between different floors of the same building in the context of graph-based SLAM. This is done under the assumption that the individual floors of a building share at least some structural properties. To identify such areas, we apply a particle filter-based localization approach using maps and observations from different floors. We evaluate our system using several real datasets as well as in simulation. The results demonstrate that our approach is able to correctly align multiple floors and allows the robot to generate consistent models of multi-story buildings.**

I. Introduction

Learning maps of environments, commonly denoted as simultaneous localization and mapping (SLAM), is a substantially researched area in mobile robotics. A large variety of solutions to this problem have been proposed in the past. Efficient and robust solutions exist especially for mapping environments where a planar 2D map is suitable for robot navigation. However, the question of how to incorporate global constraints or prior knowledge within SLAM algorithms is to a large extend unsolved. Consider, as an example, multistory buildings in which typically a considerable amount of similarities can be found. Obviously, symmetries or repetitive structures introduce additional constraints to the mapping problem. While state-of-the-art mapping algorithms may be able to consistently map each floor of such a building, the correct alignment of the individual floor maps can typically not be derived such that the overall map is globally consistent. A typical example of this problem is illustrated in Fig. 1.

In this paper, we present an approach to mapping multistory building and to identifying global constraints within maps that help to learn more consistent maps. Our approach is based on the assumption that most real-world buildings show structural similarities and we especially focus on similarities between the individual stories of multistory buildings. We propose an approach to identify such similarities based on global localization and describe how the resulting constraints can be incorporated within a graph-based formulation of the SLAM problem. In particular, the contribution of this paper is a novel approach to SLAM for multi-floor buildings. Our method generates inter-floor constraints within one building based on a localization approach. In our current implementation, these constraints are generated using a global localization method based on the Monte-Carlo localization (MCL) approach. We evaluate our method using several real-world datasets as well as in simulation.

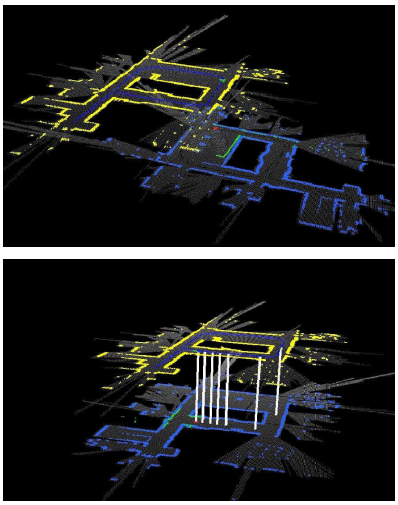

Fig. 1. With standard SLAM techniques the correct alignment of separate floor maps can not be derived in general (top). Our approach, in contrast, generates constraints between the individual floors of a building that result in a correct alignment of the individual floors (bottom).

II. Related Work

There is a large variety of SLAM approaches available in the robotics community. Common techniques apply extended and unscented Kalman filters [12], [14], sparse extended information filters [2], particle filters [16], and graph-based, least square error minimization approaches [5], [9], [15], [17].

A graph-based formulation of SLAM has been introduced by Lu and Milios [15]. In recent years, a variety of algorithms have been proposed to efficiently solve constraint networks in the context of SLAM. Frese *et al.* presented the treemap approach which employs multilevel relaxation [4] similar to Paskin's TJTF method [18]. A stochastic gradient descent method is proposed by Olson *et al.* [17] which further has been extended by Grisetti *et al.* [7] using an efficient tree parameterization. These general optimization frameworks

This work has partly been supported by the DFG under SFB/TR-8 as well as by the European Commission under FP7-231888-EUROPA.

[1] University of Ulm, Institute of Measurement, Control and Microtechnology, 89081 Ulm, Germany

[2] University of Freiburg, Department of Computer Science, 79110 Freiburg, Germany

have also been successfully applied to datasets containing simulated multi-level buildings [6], [7]. Please note, however, that in these datasets constraints between floors have not been generated from sensor measurements but have been generated from the simulated ground truth.

So far, there exist only very few methods that explicitly model the relation between separate floors of an environment. One such approach has been presented by Iocchi *et al.* [11]. A set of separate 2D floor maps is used to model the environment. In contrast to the approach presented in this paper, the floor maps need to be generated by a single robot which is able to derive the floor alignment by employing a precise visual odometry method. Consequently, this approach cannot be applied in a multi-robot scenario and will fail to provide a correct alignment if floor maps are not mapped consecutively or if the transition of floors is not visually salient.

The generation of constraints between different but similar floors is related to the problem of loop-closing within one floor and map merging in multi-robot systems. Howard *et al.* [10] present a graph-based SLAM approach that is able to merge the maps of several exploring robots once they meet and recognize each other. In [13], a multi-robot SLAM approach is presented. The relative positions of the robots are initially unknown and estimated by localizing each robot in the maps of the other robots. Hypotheses are actively verified by employing a rendezvous strategy. This approach is similar to the approach presented in this paper in that it uses a localization method to generate constraints. In contrast to our approach, however, it does not address the problem of multistory buildings. Furthermore, a physical verification can not be performed in such a scenario.

III. Graph-based SLAM

The graph-based formulation of the SLAM problem models the poses of the robot as nodes in a graph (the pose graph). Spatial constraints between poses resulting from observations or from odometry are encoded in the edges between the corresponding nodes.

Solving the SLAM problem in this formulation can be seen as a two-step process. The so-called SLAM front-end interprets the sensor data of a robot and computes constraints between poses. The SLAM back-end then typically applies optimization techniques to estimate the most-likely configuration of the nodes given these constraints. The contribution of our approach is an extension to existing SLAM front-ends in that it generates constraints between different floors of a building to improve the overall consistency of the resulting map.

A. SLAM Back-end

The aim of the SLAM back-end is to find the configuration of the nodes that maximizes the likelihood of the observations. Let $x = (x_1 \ \cdots \ x_n)^T$ be a vector of parameters which describes a configuration of the nodes. Let furthermore δ_{ji} and Ω_{ji} be the mean and the information matrix of an observation of node j seen from node i. Finally, let $f_{ji}(x)$ be a function that computes a zero noise observation according to the current configuration of the nodes j and i. Given a constraint between node j and node i, we can define the *error* e_{ji} introduced by the constraint as

$$e_{ji}(x) \quad = \quad f_{ji}(x) - \delta_{ji} \tag{1}$$

as well as the residual $r_{ji} = -e_{ji}(x)$. Let $\mathcal{C}$ be the set of pairs of indices for which a constraint exists. The goal of a ML approach is to find the configuration of the nodes that minimizes the negative log likelihood of the observations. Assuming the constraints to be independent, this can be written as

$$x^* \quad = \quad \underset{x}{\operatorname{argmin}} \sum_{\langle j,i \rangle \in \mathcal{C}} r_{ji}(x)^T \Omega_{ji} r_{ji}(x). \tag{2}$$

To solve (2), different techniques can be applied. Our work applies the approach of Grisetti *et al.* [8] which itself can be seen as an extension of the approach of Olson *et al.* [17]. It applies a stochastic gradient descent method to compute the most likely configuration of the nodes in the network. The approach minimizes (2) by iteratively selecting a constraint and modifying the configuration of nodes in the pose graph in order to decrease the error introduced by the selected constraint. For more details, we refer the reader to [8].

B. SLAM Front-end

The SLAM back-end focuses on correcting a pose graph *given* all constraints. In contrast to that, the SLAM front-end aims at extracting constraints from sensor data. In this paper, we apply a rather standard approach for obtaining constraints. First, we incrementally align the laser range scans using a gradient descent-based scan matcher. The information matrix Ω_{ji} is determined using the covariance matrix returned by the scan matching method. In addition to that, we seek for loop closures with a similar procedure considering the pose uncertainty of the robot.

The contribution of this paper is an extension to this standard SLAM front-end that allows for identifying constraints between different floors of a building. Under the assumption that different floors share at least some common structural parts (e.g., stairways, elevators, or support shafts) our system can identify similar areas and in this way align different floors and perform map corrections as we will demonstrate in the experiments.

IV. Mapping of Multi-Floor Buildings

The algorithm presented in this paper solves the problem of mapping buildings with several floors where the alignment of the floors is initially noisy or unknown. We assume that different floors share at least some structural elements such as stairways, elevators, or corridors. Once such similarities are identified, a set of inter-floor constraints is generated. These constraints are used in a graph formulation of the SLAM problem to estimate a consistent model of the complete building.

To identify structurally similar areas among the individual floor maps, we use a global localization method based on

the Monte-Carlo localization (MCL) approach [1]. MCL estimates the distribution of the robot pose x_t in map m at timestep t given all previous observations $z_{1:t}$ and actions $u_{1:t-1}$:

$$p(x_t \mid z_{1:t}, u_{1:t-1}, m) = \eta\, p(z_t \mid x_t, m) \int p(x_t \mid u_{t-1}, x_{t-1}) \; p(x_{t-1} \mid z_{1:t-1}, u_{1:t-2}, m)\, dx_{t-1} \tag{3}$$

Here, the term $p(x_t \mid u_{t-1}, x_{t-1})$ is usually referred to as the motion model and $p(z_t \mid x_t, m)$ as the sensor model while η is a normalizer which ensures that $p(x_t \mid z_{1:t}, u_{1:t-1}, m)$ sums up to one. For all quantities, we use the standard models described in [20].

As a central data structure for representing the map and the perceived data, our approach uses a pose graph g in which each node stores a pose of the robot and the corresponding laser observation. Let n_k be a node in such a graph with $n_k = \langle p_k, z_k \rangle$, where p_k is the pose of the robot during data acquisition and z_k the corresponding laser observation. This data structure is attractive since it allows us to render a grid map using mapping with known poses while maintaining all sensor observations at the same time. In this way, the laser readings recorded in one floor can be localized in another floor using the same data structure.

Our method to generate inter-floor constraints is summarized in Algorithm 1. The algorithm is initialized by choosing a reference floor g_{ref} randomly among the individual floors. A global localization is performed using the map computed from g_{ref} and the sensor observations recorded at the remaining floors. To start the global localization, the particles are initialized uniformly in the freespace of the reference map.

The algorithm then replays the observations from each floor i using the data stored in graph g_i and updates the particle filter accordingly. In every step of the algorithm, we check whether the filter has converged to an unimodal distribution. This is achieved by verifying if 99% of the probability mass is inside a small predefined area (in our current implementation set to a circle with a radius of 0.5 m). Whenever the filter has converged to a pose estimate x, we obtain a hypothesis of a correspondence between floors.

Note that a pose estimate x, obtained by localizing a sequence of d observations starting from z_k recorded in floor i within the map of the reference floor, will not be equivalent to a pose corresponding to a node in g_{ref}. Therefore, we select the pose q that corresponds to a node in g_{ref} with the smallest distance to the pose estimate x computed via MCL. Using the pose q and the pose p_k that corresponds to z_k in g_i together with an given vertical offset Δ_z to account for the height difference between both floors, we compute an inter-floor constraint as

$$c = (q\, p_k^{-1}) + (0, 0, \Delta_z)^T. \tag{4}$$

In principle, one could now add these constraints to the pose-graph connecting the so-far unconnected graphs representing the individual floors. The described procedure, however, is likely to generate outliers. Adding such outliers to the SLAM optimization procedure used to compute the global map is likely to lead to a catastrophic failure. In contrast to that, it is typically not critical to omit a correctly identified constraint between floors. Therefore, we introduce two further tests in order to reduce the risk of adding wrong constraints.

The first test considers the sensor measurement likelihoods obtained when evaluating the sensor model. We compute the average observation likelihood l_t of a measurement z_t over all N particles as

$$l_t = \frac{1}{N} \sum_{j=1}^{N} p(z_t \mid x_t^{(j)}, m), \tag{5}$$

where $x_t^{(j)}$ denotes the pose hypothesis of particle j.

High values of l_t indicate that the measurement can be explained well by the map m while low values may indicate that the particles have actually converged to a position in the environment that does not correspond to a likely pose in the map. Thus, we use a heuristic threshold criterion and discard the hypothesis if $l_t < l^{min}$. The threshold l^{min} can be learned by evaluating the sensor model for localization while building the map of the current floor (the map corresponds to the floor from which the observations are taken). If l_t exceeds the threshold, the corresponding hypothesis c is added to a candidate set $\mathcal{C}_i$. Then, the filter is reset and the procedure is restarted.

In addition to this heuristics, we apply RANSAC [3] on the candidate sets $\mathcal{C}_i$ to eliminate outliers. RANSAC is an iterative algorithm that estimate parameters of a model from a set of observed data which contains outliers. In our system, the model corresponds to an alignment of floor maps which can be described by a planar translation and rotation around the vertial axis. Using the RANSAC notation, the alignment hypotheses in $\mathcal{C}_i$ serve as observations of this alignment. In this way, RANSAC will determine the maximum set of consistent inter-floor constraints.

Algorithm 1 Constraint generation via MCL

Require: $\{g_0..g_n\}$, set of floor graphs
1: $ref \leftarrow$ random $(0..n)$ *// select reference floor*
2: **for all** $g_i \in \{g_0..g_m\} \backslash \{g_{ref}\}$ **do**
3: $\mathcal{C}_i \leftarrow \emptyset$
4: **for** $k = 1$ to $sizeof(g_i)$ **do**
5: **if** $x \leftarrow$ MCL $(g_{ref}, z_{k:k+d})$ **then**
6: $k \leftarrow k + d$
7: $q \leftarrow$ findNearestNeighbor (g_{ref}, x)
8: $c \leftarrow$ computeConstraint (q, p_k, Δ_z)
9: $\mathcal{C}_i = \mathcal{C}_i \cup \{c\}$
10: **end if**
11: **end for**
12: $\mathcal{C}_i \leftarrow$ RANSAC $(\mathcal{C}_i)$
13: **end for**
14: mergeGraphs($\{g_0..g_n\}$, $\{\mathcal{C}_0..\mathcal{C}_n\}$)

After adding the set of constraints verified with the

RANSAC, the graph-based optimization approach will align the individual floors and thus lead to a consistent model of the building with high likelihood.

Note that the minimal amount of inter-floor constraints that is necessary to correctly map a building depends on the input data. In theory and under the assumption that individual floors are consistently mapped, a single constraint will be sufficient to align two floors maps and RANSAC would not have to be used. In practice, however, floor graphs contain erroneous local constraints due to measurement noise and odometry errors. Thus, the overall model of the building will be improved by multiple inter-floor constraints. For this reason, our current implementation requires at least three constraints to be found for each floor.

An Illustration of the overall approach can be found in the multimedia attachment of this paper.

V. Limitations of the Approach

As we will illustrate in the experimental section, the approach works well in typical real world buildings. Even if different floors share only few common structures, our approach is able to align them.

Buildings with strong symmetries may, however, pose a problem to the described approach. In such cases, the global localization method will most likely converge to a multi-modal distribution and thus no constraints can be generated.

Our approach will furthermore not be successful for a building whose floors do not share any similarities. These building occur rarely in real world but do exist. One such building is the Stata Center at MIT. Here, our approach was unable to find constraints between the third and the eighth floor. This comes as no surprise, since even a manual alignment of the floor maps is nearly impossible without background knowledge.

In both cases described above, our system behaves like a standard graph-based SLAM approach.

VI. Experiments

Our approach was evaluated using several real world datasets as well as simulated data. The experiments are designed to investigate whether the proposed approach is able to correctly model real world buildings. Furthermore, we also evaluate whether the overall mapping error is reduced and to what extend mapping errors in one of the floors can be corrected using the maps of other floors. All real world datasets were recorded using a Pioneer2 robot equipped with a SICK laser scanner.

A. Typical Office Building with four Floors

This dataset was recorded in building 106 on the computer science campus in Freiburg (see Fig. 2). The building consists of four floors that share a similar structure, most prominently in the area around the elevator. The floors differ mainly in the area around the staircase. Additional differences are introduced by open doors and furniture.

The floor maps aligned by our approach can be seen in Fig. 3. By visual inspection of an overlay of all four maps (see Fig. 4) there is no apparent alignment error.

Fig. 2. Building 106 on Freiburg Campus

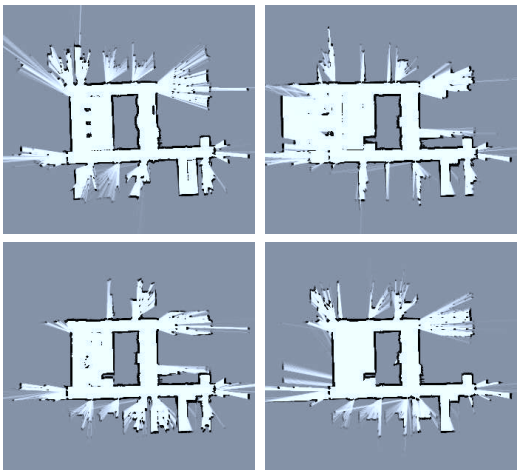

Fig. 3. Aligned floors of building 106 on Freiburg Campus

B. Building with Long Corridors

A more challenging dataset was recorded in building 051 on the computer science campus in Freiburg. The floors of this three-story building essentially consist of two long corridors meeting at an elevator in the middle of the building.

With all office doors closed the elevator area and a staircase on either end of the building are the main structural features. Using the proposed approach, constraints were found in those salient areas and a correct alignment was obtained. In Fig. 5 two of the three already aligned floors are depicted in an three-dimensional illustration.

C. Building with Few Structural Similarities

Building 101 on the computer science campus in Freiburg features a modern architecture with large glass constructions (see Fig. 6). Its three floors do not share a lot of structural

Fig. 4. Overlay of the four floors of building 106 on Freiburg Campus

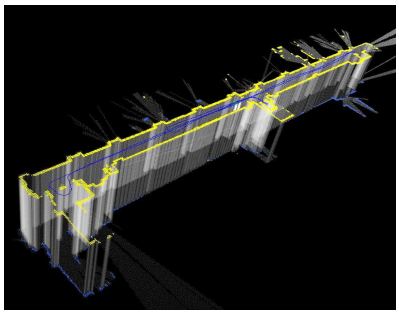

Fig. 5. Two correctly aligned floors of building 051. To illustrate the alignment, occupied cells found in both floor maps have been connected.

Fig. 6. Building 101 on Freiburg Campus

features, the only exception being an elevator shaft and a short corridor leading to the restrooms.

The MCL-based approach successfully generated inter-floor constraints in the similar areas around the elevator but failed to generate further constraints in other parts of the building. For this reason, the alignment achieved for this dataset is not as good as in the previous experiments. A small but noticeable rotational error remains and can be seen in the overlay of two of the resulting floor maps depicted in Fig. 7.

D. Quantitative Evaluation Using a Simulated Building with Ten Floors

To quantitatively evaluate the alignment obtained using our approach, we created a virtual dataset consisting of ten floors

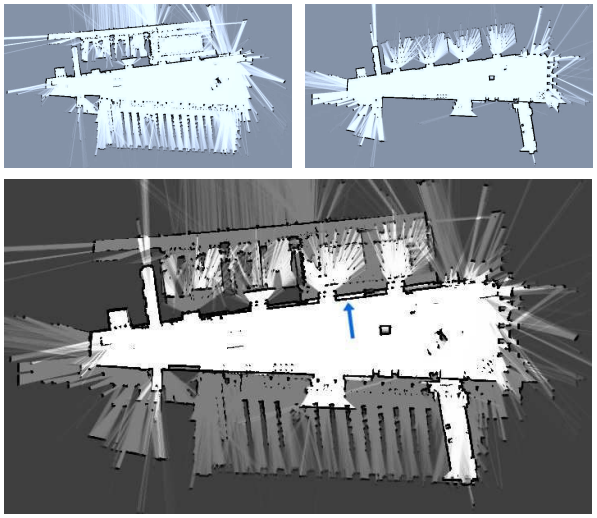

Fig. 7. Top: two floor maps of building 101 with few similarities. Bottom: overlay of the floors with alignment found by our approach. The blue arrow indicates a minor inconsistency.

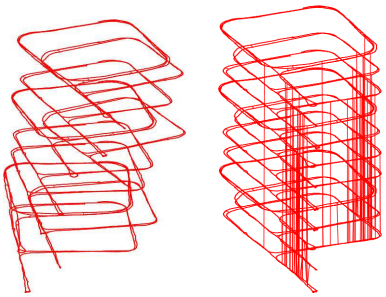

Fig. 8. Left: Floor graphs of a simulated building with ten floors before alignment. Right: a merged graph has been computed and each of the floor graphs has been correctly aligned against the reference floor.

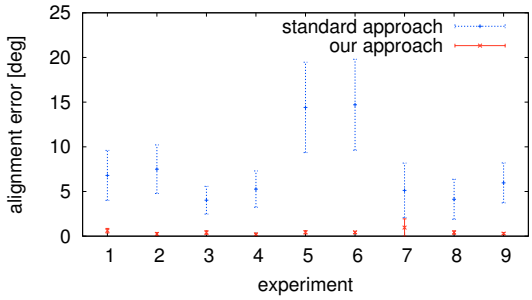

Fig. 9. Average alignment error with confidence intervals according to a t-test with 95% confidence computed for 9 different experiments with 10 floors each.

taken from a real world dataset. To this end, we extended the dataset of building 106. The top three floors were duplicated twice while the first floor was used once only. A Gaussian rotational error was added to each floor graph to simulate pose uncertainty across floors. The variance of this error relates to the rotational error observed in experiments in which a robot entered an elevator, turned on the spot, and left the elevator again. The graphs of the individual floor without connecting constraints as well as the corrected and merged graph generated by our approach are shown in Fig. 8.

Figure 9 depicts a statistical analysis of the alignment error. For this experiment, we manually selected the first floor as the reference floor and aligned each of the following floor graphs against it using our approach. By choosing the reference floor in this way, an unrealistic overfitting is avoided since matchings between duplicated (virtual) floors are not considered. The graph optimization was then carried out with and without the inter-floor constraints found by our system. We manually determined the correct alignment of the floors and computed the deviation of this to the alignment obtained by the SLAM approaches. This was done in nine different experiments with ten floors. As can be seen from the average alignment error and the confidence intervals, using inter-floor constraints significantly reduces the alignment error of the floors.

E. Correction of Systematic Mapping Errors

In this experiment, we evaluate to what extend the presented approach can be used to correct for systematic sensor errors in one of the floors. Such errors may, for example, be induced by lighting conditions in the case of vision sensors

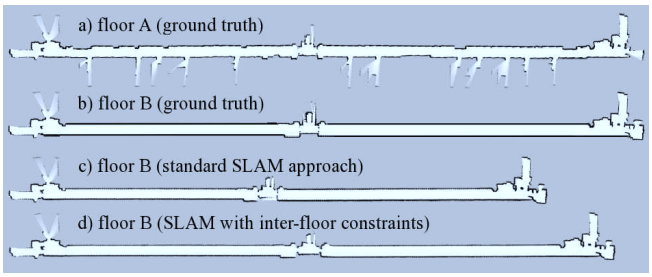

Fig. 10. Simulated environment with long corridors. a) simulated floor A. b) simulated floor B with few structural features. c) floor B mapped using the standard approach without inter-floor constraints. There is an average error in length of 14.2%. d) floor B after applying our approach. The average remaining error in length is 5.8%.

or by long, featureless corridors in the case of laser scanners. To evoke such an error, we simulated an environment based on building 051 that consists of two different corridors with a length of 130 m as depicted in Fig. 10. Floor A features a similar geometrical structure as the original building but was extended by copying parts of the corridors. Floor B is identical to floor A with the exception that it does not contain any significant structure within the corridors. However, both floors still share the elevator area and the two staircases also present in the original map.

The standard SLAM front-end was able to generate a consistent map of floor A. The featureless corridor of floor B, however, posed a problem to the scan matcher. The maximum measurement range of 30 m was not sufficient to measure structurally salient regions or the limiting walls at the ends of the corridor at all times. For this reason, the corridor of floor B was shortened by the scan matching leading to a shortened corridor in the resulting map (see Fig. 10 c). This is a typical effect of short range proximity sensors [19]. We measured the error in the corridor length over ten runs of the experiment. On average, this shortening amounted to 14.2% of the original length (corresponding to 18.5 m). The results in significant according to a t-test performed with a confidence level of 5%.

The proposed MCL-based approach was able to find inter-floor constraints at the structurally identical staircases at both ends of the simulated floors. Using those constraints and the floor graphs of floor A and floor B, our approach was able to generate a merged graph of both floors. The resulting map of floor B can be seen in Fig. 10 d. The average residual error in length was 5.8% (corresponding to 7.5 m).

VII. Conclusion

This paper presented an approach to simultaneous localization and mapping that extracts alignment constraints between different floors of buildings and utilizes these constraint to generate more accurate maps of multistory buildings. Our approach uses Monte Carlo localization and performs a global pose estimation to seek for potential constraints between the individual floors. The selected constraints are then integrated into a graph-based optimization approach to address the SLAM problem. We evaluated our system using different real world datasets. The results indicate that the overall approach is quite robust and allows for the generation of multistory maps that are more accurate than those obtained with an approach not considering global constraints between the individual floors.

References

[1] F. Dellaert, D. Fox, W. Burgard, and S. Thrun. Monte carlo localization for mobile robots. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, Leuven, Belgium, 1998.

[2] R. Eustice, H. Singh, and J.J. Leonard. Exactly sparse delayed-state filters. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, pages 2428–2435, Barcelona, Spain, 2005.

[3] M.A. Fischler and R.C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Graphics and Image Processing*, 1981.

[4] U. Frese. Treemap: An $o(logn)$ algorithm for indoor simultaneous localization and mapping. *Journal of Autonomous Robots*, 21(2):103–122, 2006.

[5] U. Frese, P. Larsson, and T. Duckett. A multilevel relaxation algorithm for simultaneous localisation and mapping. *IEEE Transactions on Robotics*, 21(2):1–12, 2005.

[6] U. Frese and L. Schroder. Closing a million-landmarks loop. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 5032–5039, 2006.

[7] G. Grisetti, S. Grzonka, C. Stachniss, P. Pfaff, and W. Burgard. Efficient estimation of accurate maximum likelihood maps in 3d. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, San Diego, CA, USA, 2007.

[8] G. Grisetti, C. Stachniss, and W. Burgard. Non-linear constraint network optimization for efficient map learning. *IEEE Transactions on Intelligent Transportation Systems*, 10(3), 2009.

[9] G. Grisetti, C. Stachniss, S. Grzonka, and W. Burgard. A tree parameterization for efficiently computing maximum likelihood maps using gradient descent. In *Proc. of Robotics: Science and Systems (RSS)*, Atlanta, GA, USA, 2007.

[10] A. Howard. Multi-robot simultaneous localization and mapping using particle filters. In *Robotics: Science and Systems*, pages 201–208, Cambridge, MA, USA, 2005.

[11] L. Iocchi, S. Pellegrini, and G. D. Tipaldi. Building multi-level planar maps integrating LRF, stereo vision and IMU sensors. In *Proc. of IEEE International Workshop on Safety, Security and Rescue Robotics (SSRR'07)*, Rome, Italy, 2007.

[12] S. Julier, J. Uhlmann, and H. Durrant-Whyte. A new approach for filtering nonlinear systems. In *Proc. of the American Control Conference*, pages 1628–1632, Seattle, WA, USA, 1995.

[13] J. Ko, B. Stewart, D. Fox, K. Konolige, and B. Limketkai. A practical, decision-theoretic approach to multi-robot mapping and exploration. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 3232–3238, Las Vegas, NV, USA, 2003.

[14] J.J. Leonard and H.F. Durrant-Whyte. Mobile robot localization by tracking geometric beacons. *IEEE Transactions on Robotics and Automation*, 7(4):376–382, 1991.

[15] F. Lu and E. Milios. Globally consistent range scan alignment for environment mapping. *Journal of Autonomous Robots*, 4:333–349, 1997.

[16] M. Montemerlo and S. Thrun. Simultaneous localization and mapping with unknown data association using FastSLAM. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, pages 1985–1991, Taipei, Taiwan, 2003.

[17] E. Olson, J. Leonard, and S. Teller. Fast iterative optimization of pose graphs with poor initial estimates. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, pages 2262–2269, 2006.

[18] M.A. Paskin. Thin junction tree filters for simultaneous localization and mapping. In *Proc. of the Int. Conf. on Artificial Intelligence (IJCAI)*, pages 1157–1164, Acapulco, Mexico, 2003.

[19] C. Stachniss, M. Bennewitz, G. Grisetti, S. Behnke, and W. Burgard. How to learn accurate grid maps with a humanoid. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, 2008.

[20] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics*. MIT Press, 2005.