Improvement of Map Building during the Exploration of Polygonal Environments using the Range Data

Sandor Iles* Marija Seder* Ivan Petrovic*
*Department of Control and Computer Engineering, Faculty of Electrical Engineering and Computing, University of Zagreb, Croatia

***Abstract*— In this paper we consider problem of exploration and mapping of unknown polygonal environments. To construct a map of unknown environment we first must have exploration algorithm, and we have to choose a map representation method. Unknown environment needed to be explored is an indoor office environment. We use line map representation method since it is easy to represent office environment using line segments and it is significantly less memory consuming than occupancy grid map representation method. We combined two algorithms for line map building: 1) weighted line fitting algorithm developed by Pfister *et al.* [9], which incorporates noise models of the range sensor and robot's pose uncertainty and 2) exploration algorithm developed by Ekman *et al.* [2], which explores polygonal environments using ideal range data with no uncertainty and dealing with no positional uncertainty. We combined and improved both methods to derive complete exploration of polygonal environments and statistically sound mapping solution. The algorithms were tested using Pioneer 3DX mobile robot equipped with a laser range finder.**

***Index Terms*— exploration, mapping, line extraction, line fitting, mobile robotics**

I. Introduction

The exploration problem is the part of the more general problem of robot motion planning. The robot equipped with range sensor is required to autonomously navigate in the finite unknown environment with the purpose of building the complete map. During the exploration, the robot is expected to traverse as small a distance as possible. Therefore, path planning algorithm must be used for calculating the optimal paths to the destination points of the exploration algorithm using the newly explored information. Also, path following algorithm must be used, which directly controls the robot's motion obeying its kinematic and dynamic constraints and avoiding obstacles in the path. There are different possible exploration methods and map representations.

The most exploration methods generate a line maps since they require significantly less memory than occupancy grid maps. Furthermore, they are more accurate since they do not suffer from discretization problems. A variety of techniques for learning line maps from range data have been developed. Common task for all techniques is to find a minimum number of line segments that best approximate a given set of range data. Techniques used in exploration problem must operate online, i.e., while the robot is exploring its environment. A number of algorithms apply the well-known iterative end-point fit or split-and-merge algorithm [1], [11] but do not incorporate noise models of the range data. Therefore, the fitted lines do not have a sound statistical interpretation. A Kalman-Filter based approach for extracting line segments [10] allows only for uniform weighting of the point fitting contributions. Pfister *et al.* [9] consider how to accurately fit a line segment to a set of uncertain points. Their fitting procedure weights each point's influence on the overall fit according to its uncertainty, where the point's uncertainty is derived from sensor noise models. They provide closed-form formulas for the covariance of the line fit.

The requirement for the exploration problem commonly used in literature is that obstacles must be algebraic manifolds. In [5] exploration strategy for environments containing convex polygonal objects is obtained by repeatedly calling a path planning algorithm with randomly selected destination points. This strategy generates a faithful map with probability 1 but lacks a termination criterion. Ekman *et al.* [2] present exploration strategy for arbitrary polygonal environments, which assumes a range sensor of finite angular resolution and thus provides sampled version of the visibility polygon [6] instead of imposing further restrictions on the environment. They derive conditions, resembling the Shannon sampling theorem, under which it is possible to generate a faithful map from the range data. The exploration strategy is based on the presence of discontinuities in the range data called ”jump edges” as a possibly unexplored environmental regions. The measurement positions are in the front of jump edges and their selection is defined by the exploration strategy. The exploration strategy is proved to completely explore a finite unknown environment in finite time. However, proposed mapping strategy in [2] uses robot with no positional uncertainty and with ideal laser range sensor.

In this paper we introduce a complete solution for exploration and mapping of unknown environment with some constraints on the environment. We use exploration strategy presented in [2] and weighted line fitting algorithm presented in [9], which introduces sensor noise model and uncertainty. We combined and improved both methods to derive complete exploration of polygonal environments and statistically sound mapping solution. We speed up the exploration convergence by changing the selection criterion of the measurement positions. Line map is produced with weighted line fitting algorithm presented in [9] with certain improvements. Instead of using

Standard Hough Transform (SHT)[3], we used Probabilistic Hough Transform (PHT) with postprocessing. PHT uses only a fraction of points, thus minimising amount of computation needed to detect lines, while giving less false positive and false negatives than SHT, as stated in [4]. Post processing is used to retrieve finite line segments instead of infinite lines. Finite line segments retrieved from PHT, are used as initial guess for weighted line fitting algorithm. Points are grouped around estimated line segment if distance between point and line segment is smaller or equal to $3\sigma_d$ value. However this method can lead to undesired effects due to PHT line segment estimation if too fine discretization is chosen. It is then possible and highly likely that estimation will result in too many line segments. We propose methods to eliminate impossible line estimations in single range scan. When we detect such impossible lines we simply group points around them and proceed to linefit. Furthermore, we propose a method for separating the line segments, which share a common wall but are separated by a doorway. Since range data have no information on data between points, range data measurement prediction is used to ensure proper line splitting when necessary, giving correct data representation.

This paper is structured as follows. Section 2 describes polygonal environment exploration algorithm. Section 3 describes initial guess and methods for eliminating impossible line estimations. Section 4 describes line fitting algorithm along with methods for line splitting and merging. The experiments in Section 5 demonstrate achieved results.

II. Polygonal environment exploration algorithm

When working with indoor office environment, it is easy to use line segments for representing the environment. Therefore, we choose the exploration algorithm of polygonal environment since line segments can easily be connected to a polygon. A planar polygon P is a sequence of at least three line segments $\overline{v_0v_1}$, $\overline{v_1v_2}$, ..., $\overline{v_{n-1}v_0}$, where the points v_0, v_1, ..., v_{n-1} are coplanar.

We used polygonal environment exploration algorithm described in [2]. Algorithm is written using robot with no localisation uncertainty and with ideal laser sensor. Strategy is based on the presence of discontinuities in the range data. Such discontinuities are called jump edges. Jump edges indicates possibly unexplored regions. When robot measures in front of jump edges, new regions are explored. Lack of jump edges indicates that environment had been explored. Map is represented by polygonal representation, and at the higher level graph representation is used for path planning, called exploration graph. A candidate measurement position is defined in front of each jump edge. Exploration graph is composed of nodes, which correspond to measurement positions, and edges which are defined between visible nodes[1].

In [2] a criterion is proposed that satisfies completely exploration of an environment within a finite number of measurements, with no claim of being optimal. Since environment is *a priori* unknown, heuristics is used to predict the amount of new information obtained from a measurement. Next measurement position is chosen from the exploration graph according to the selection criterion, which is defined as:

$$C(n) = \frac{g_n}{c_n}, \tag{1}$$

where g_n is the gain value of node n defined by all jump edges (je) visible form the node n and c_n is the length of the path between current measurement position and the node n. The gain is defined as:

$$g_n = \sum_i \alpha_{ni}, \tag{2}$$

where α_{ni} is the angle at the node n in the triangle defined by ith jump edge (je_i) and the node n. This is illustrated in Fig. 1. The node with the highest C value is selected as the new measurement position. Let's consider an environment configuration described in Fig. 2, where laser range scan is shown with shaded light blue colour and jump edges are shown with red lines. According to original selection criterion, it can happen that further candidate measurement position will be selected rather than closer candidate measurement position to the robot's position and the part of the polygonal environment will still remain unexplored. In such indoor environments with a lot of doors, desks and passages, this situation could happen

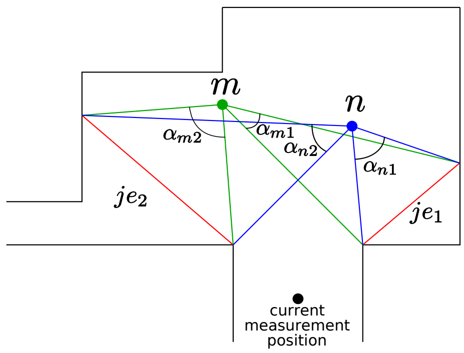

Fig. 1. Selection criterion based on the gain values

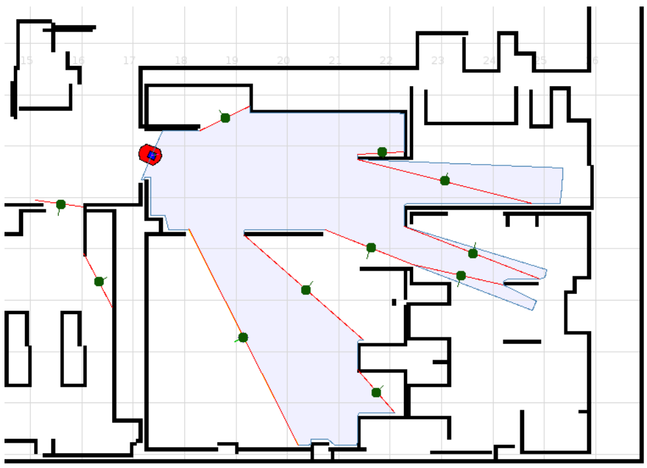

Fig. 2. Jump edges and candidate measurement positions

[1]Two points in the space are said to be visible to each other, if the line segment that connects them does not intersect any obstacles

frequently, which will result in unsatisfactory too long robot's motion. The main reason for this occasion is the use of the ratio between the gain value and the length of the path in the selection criterion.

To prevent such situations, we changed the selection criterion. Instead of using the ratio, we use the sum of two functions. Since the gain value g_n can maximally be $180°$ (see Fig. 1) and placing the measurement position should not be too far from the jump edge due to the limit of the sensor range, we have chosen to place measurement position exactly at the midpoint of the jump edge with orientation perpendicular to the jump edge. Therefore, there is no need of using the gain value in the selection criterion. Instead, we incorporate the length of the jump edge in the selection criterion since it greatly determines the size of unexplored environment. Fig. 2 shows candidate measurement positions (green dots) in order to scan possible unexplored parts of the environment. Orientations of the candidate measurement positions are noted with small lines. The selection criterion is defined as:

$$C(n) = \frac{l_n}{R_{max}} + \frac{R_{max}}{c_n}, \tag{3}$$

where l_n is the length of the jump edge at the node n and R_{max} is maximal sensor range. This form of the selection criterion prefers equally longer jump edges and shorter paths to the candidate measurement positions. Fig. 3 shows the robot taking the next measurement scan from the best measurement position according to the selection criterion and configuration presented in Fig. 2. The following best measurement position

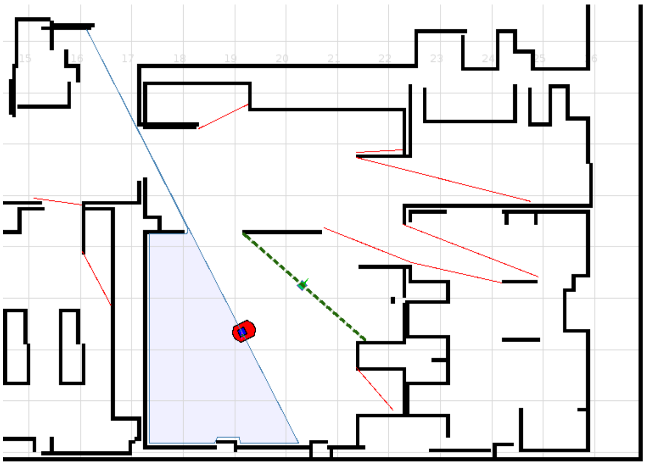

Fig. 3. The robot's next scan from the best measurement position

is shown with green dashed line.

At every measurement position the algorithm transforms range scan into sampled version of the visibility polygon [6]. Measurement polygon is created iteratively by adding new polygons from the following measurement positions. Extended exploration polygon is created by adding the jump edges to the measurement polygon. Jump edge at the measurement position is deleted from the extended exploration polygon. If extended exploration polygon contains no jump edges, then it is equal to faithful polygonal description P_F.

In Fig. 4 simulation results of the exploration algorithm is shown. Algorithm is implemented in *Player/Stage*[2]. Range

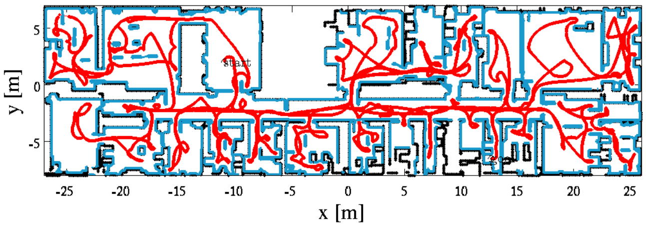

Fig. 4. Simulation results of polygonal environment exploration algorithm

sensor and position are almost ideal and, therefore, polygonal map (blue lines) is completely aligned with the real map (black dots) used in the simulator. The robot explores all parts of environment it could reach in finite time. Its trajectory is presented with red colour.

The used exploration algorithm assumes ideal laser range sensor and no positional uncertainty. When faithful polygonal description P_F is obtained, the map consists of almost all the raw sensor data samples that have been gathered. Our contribution is in reducing noise effects on map representation and reducing number of lines.

III. Initial line segment estimation

This step is initial estimation of the line segment and it is necessary for further iterative solution of non-linear optimisation problem. The range data from a measurement position is first sorted into subsets of roughly collinear points using the Hough Transform.

We use weighted line fitting algorithm based on Hough transform described in [8] and [9] with certain improvements. We use probabilistic Hough transform (PHT) with post processing, instead of standard Hough transform (SHT). Only a fraction of points is used by PHT, thus amount of computation time needed to detect lines is minimised. PHT also estimates less false positive and false negative lines than SHT, as stated in [4]. However PHT does not take noise and uncertainty into account when estimating the line parameters. As a consequence it is possible to get higher number of line segments than necessary from single scan e.g (two partially overlapping line segments instead of one line segment). It is also possible to get impossible line positions (e.g intersecting line segments or partially overlapping line segments) if too fine initial discretization is chosen. Various methods for merging partially overlapping and fully overlapping segments were proposed in [12]. To maintain statistically sound method, we detect such lines and group points around them, as seen in Fig. 5. It is then possible to detect dominant line and proceed with statistically sound linefit algorithm using points grouped around that line. Afterwards we remove grouped points, and proceed with new set of points as input to PHT.

[2] a free software tool for robot and sensor applications, `www.playerstage.sourceforge.net`

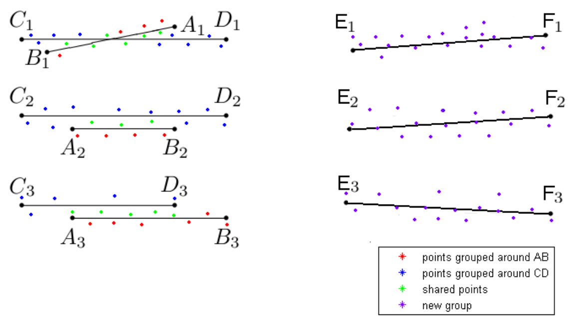

Fig. 5. Examples of candidate lines to be grouped together

IV. Weighted line fitting algorithm

As mentioned before, line fitting algorithm and its derivation is fully described in [7],[8],[9]. Here we will only introduce laser range sensor model and line representation, and provide formulas for line estimation, line covariance estimation, and line merging using chi-squared test.

We use laser range sensor that measures range in 361 directions from the $\frac{-\pi}{2}$ to $\frac{\pi}{2}$. Let the range measurement $\hat{d}_k$ be comprised of the true range d_k and an additive noise term ϵ_d:

$$\hat{d}_k = d_k + \epsilon_d. \tag{4}$$

Also, let the scan angle $\hat{\Theta}_k$ be comprised of the true scan angle Θ_k and an additive noise term ϵ_Θ:

$$\hat{\Theta}_k = \Theta_k + \epsilon_\Theta. \tag{5}$$

The set of n points from the single measurement position is denoted as $\hat{u}_k$, where $k = 1...n$. Therefore, kth measurement point in the robot's local reference frame is given as:

$$\hat{u}_k = (d_k + \epsilon_d)\begin{bmatrix}\cos(\Theta_k + \epsilon_\Theta)\\ \sin(\Theta_k + \epsilon_\Theta)\end{bmatrix}. \tag{6}$$

Let assume that all noise terms are zero-mean Gaussian random variable. There, ϵ_d has variance σ_d^2 and ϵ_Θ has variance σ_Θ^2. For practical computation, we can use $\hat{\Theta}_k$ and $\hat{d}_k$ as a good estimates for the quantities Θ_k and d_k.

Infinite line in polar form is represented as follows:

$$L = \begin{bmatrix}\rho\\ \alpha\end{bmatrix} \tag{7}$$

Line parameters are obtained by well known Weighted Least Square algorithm:

$$\rho = P_{\rho\rho}\left(\sum_k^n P_k^{-1}\right) \tag{8}$$

Iterative solution for α is suggested as $\alpha = \hat{\alpha} + \delta\alpha$:

$$\delta\alpha = -\frac{\sum_{k=1}^n \left(\frac{P_k a_k' - a_k b_k'}{(P_k)^2}\right)}{\sum_{k=1}^n \frac{(a_k'' P_k - a_k b_k'')P_k - 2(a_k' P_k - a_k b_k')b_k'}{(P_k)^3}}, \tag{9}$$

with:

$$\begin{aligned} s_k &= \sin(\hat{\alpha} - \Theta_k)\\ c_k &= \cos(\hat{\alpha} - \Theta_k)\\[1em] a_k &= (d_k c_k - \hat{\rho})^2\\ a_k' &= -2 d_k s_k (d_k c_k - \hat{\rho})\\ a_k'' &= 2 d_k^2 s_k^2 - 2 d_k c_k (d_k c_k - \hat{\rho})\\ b_k' &= 2(d_k^2 \sigma_\Theta^2 - \sigma_d^2)(c_k s_k)\\ b_k'' &= 2(d_k^2 \sigma_\Theta^2 - \sigma_d^2)(c_k^2 - s_k^2), \end{aligned} \tag{10}$$

where P_k is described with Eq. (15). The covariance of the line position is:

$$P_L = \begin{bmatrix} E\left[\epsilon_\alpha^2\right] & E\left[\epsilon_\alpha \epsilon_\rho\right]\\ E\left[\epsilon_\alpha \epsilon_\rho\right] & E\left[\epsilon_\rho^2\right]\end{bmatrix} = \begin{bmatrix} P_{\rho\rho} & P_{\rho\alpha}\\ P_{\alpha\rho} & P_{\alpha\alpha}\end{bmatrix}, \tag{11}$$

with:

$$P_{\rho\rho} = \frac{1}{\sum_{k=1}^N P_k^{-1}}, \tag{12}$$

$$P_{\alpha\alpha} = \frac{1}{\sum_{k=1}^N \frac{\delta\Psi_k^2}{P_k}}, \tag{13}$$

$$P_{\rho\alpha} = -P_{\alpha\alpha} P_{\rho\rho} \sum_{k=1}^N \frac{\delta\Psi_k}{P_k}, \tag{14}$$

with P_k and $\delta\Psi_k$ defined as follows:

$$P_k = \sigma_d^2 \cos^2(\hat{\alpha} - \Theta_k) + \sigma_\Theta^2 d_k^2 \sin^2(\hat{\alpha} - \Theta_k), \tag{15}$$

$$\delta\Psi_k = d_k \sin(\hat{\alpha} - \hat{\Theta}_k) - \frac{\sum_{k=1}^n \frac{\Psi_k}{P_k}}{\sum_{k=1}^n \frac{1}{P_k}}. \tag{16}$$

After detecting infinite lines, line segments are retrieved by trimming infinite lines at extreme endpoints. Line segment representation with common infinite line and arbitrary even number of points is used. Extreme endpoints are detected if distance between two points is larger than some threshold. Our contribution is to dynamically calculate that threshold (l_1) using measurement prediction. Measurement prediction and line segment representation mentioned above is used to detect doorways and similar features, as shown in Fig. 6. Furthermore we introduce another threeshold (l_2) for eliminating points that belongs to some other line segment.

Method for estimating distance between two measurement points is proposed in [10]. However we take into account modeled laser range noise when predicting next measurement:

$$\begin{aligned} \tilde{\Theta}_k &= \Theta_k \pm 5\sigma_\Theta\\ \hat{x} &= \frac{(\rho \pm 5\sigma_{d_k})}{\cos(\alpha) + \sin(\alpha)\tan(\tilde{\Theta}_k)}\\ \hat{y} &= x\tan(\tilde{\Theta}_k), \end{aligned} \tag{17}$$

where $\hat{x}, \hat{y}$ are Cartesian coordinates of estimated measurement.

Ideal prediction is made along with modeled noise prediction for all combinations of $\pm$ sign to get maximal and minimal expected distance between two measurements. Minimal allowed distance allows detecting points that do not belong to

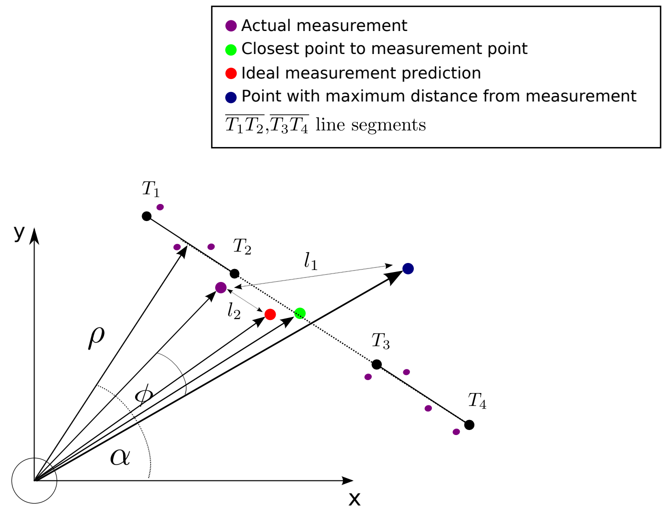

Fig. 6. Line segments sharing infinite line separated by measurement prediction

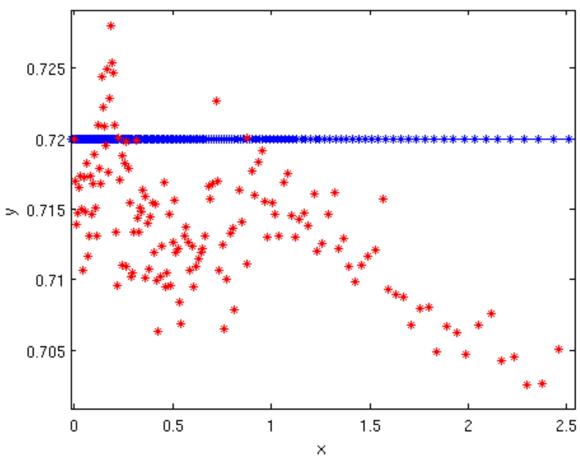

Fig. 7. Measurement prediction based on actual measurement

the proposed line. Ideal measurement prediction is shown in Fig. 7.

After estimating line segments it is possible to further compress data by merging sufficiently similar line segments. If two lines are not taken from the same pose, it is first necessary to transform lines and covariance matrices as described in the following.

Lets consider line $L_i = [\alpha_i \ \rho_i]$ taken from local reference frame i represented by $g_i = (x_i, y_i, \phi_i)$. It is possible to transform line into global reference frame g_0. Line in global reference frame is represented as L_0 :

$$L_0 = \begin{bmatrix} \alpha_0 \\ \rho_0 \end{bmatrix} = \begin{bmatrix} \alpha_i + \phi_i \\ \rho_i + \delta\rho_i \end{bmatrix}, \tag{18}$$

where :

$$\delta\rho_i = x_i \cos(\alpha_i + \phi_i) + y_i \sin(\alpha_i + \phi_i). \tag{19}$$

Let's consider line L_i and its covariance matrix P_{Li} taken at pose g_i with covariance matrix P_{g_i}. It is possible to transform covariance matrix as follows:

$$P_{L0} = K_i P_{Li} K_i^T + H_i P_{g_i} H_i^T, \tag{20}$$

where:

$$K_i = \begin{bmatrix} 0 & 0 & 1 \\ \cos(\alpha_i + \phi) & \sin(\alpha_i + \phi) & 0 \end{bmatrix}, \tag{21}$$

$$H = \begin{bmatrix} 1 & 0 \\ \delta\Psi & 1 \end{bmatrix}, \tag{22}$$

$$\delta\Psi = y_i \cos(\alpha_i + \phi_i) - x_i \sin(\alpha_i + \phi_i). \tag{23}$$

To determine wether given pair of lines are sufficiently similar to merge we use merge criterion based on Chi-Squared test :

$$X^2 = (\delta L)^T (P_{L1}^i + P_{L2}^i)^{-1} (\delta L) < 3, \tag{24}$$

where

$$\delta L = L_1^i - L_2^i. \tag{25}$$

If lines are sufficiently similar for merging, we compute new underlying line estimate as follows:

$$L_m^i = P_{Lm}^i ((P_{L1}^i)^{-1} {L_1}^i + (P_{L2}^i)^{-1} {L_2}^i), \tag{26}$$

$$P_{Lm}^i = ((P_{L1}^i)^{-1} + (P_{L2}^i)^{-1})^{-1}. \tag{27}$$

V. Results

Algorithms are tested on mobile robot Pioneer 3DX in a small part of the environment. Mapping algorithms are compared under the same conditions. In Fig. 8 results obtained from single range scan data are presented. In Fig. 9 results from ten poses are brought to common reference frame.

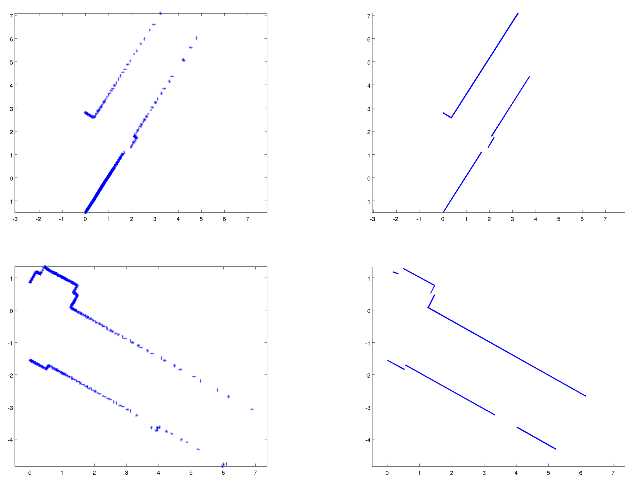

Fig. 8. Two pose single scan results.

In Fig. 10 the result of the exploration algorithm is shown. The robot explores all parts of environment it could reach in finite time. Its trajectory is presented with red colour. Polygonal map is presented with blue lines and partially

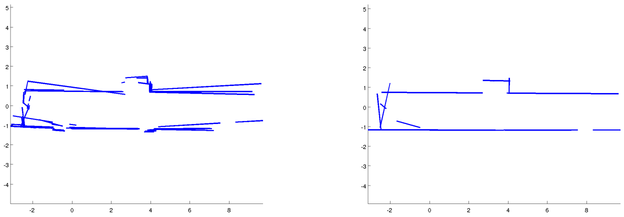

Fig. 9. Fit lines from several poses in common reference frame. Nonmodeled uncertainty (left) and modeled uncertainty (right)

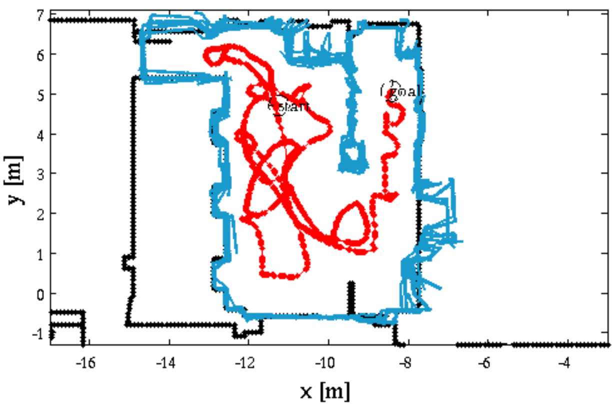

Fig. 10. Experimental results of polygonal environment exploration algorithm - Polygonal map representation

known map used for evaluation of the results is presented with black dots. The map consists of almost all the raw sensor data samples that have been gathered. It is a noisy map representation due to real sensor noise and localisation uncertainty.

Improvement in reducing noise effects on map representation and reducing number of lines is obvious in Fig. 11. After

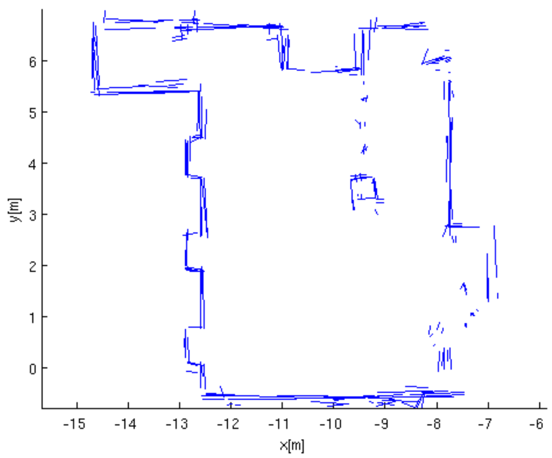

Fig. 11. Experimental results of polygonal environment exploration algorithm - Line fitted map representation

linefit algorithm without grouping points around similar lines and merging part we retrieved 886 lines from 21960 raw data points. After grouping points around similar lines we retrieved 416 fit lines. After merging similar lines we retrieved 545 lines, and by combining these two methods we retrieved 324 lines. We achieved compression of 97.05%. Further compression is possible but it would result in loosing information of empty space between objects which we prevented by measurement prediction. However, it is worth mentioning that further compression could probably been achieved if environment had less nonpolygonal obstacles. There was a table, a flower, some boxes and some chairs to block access to stairways which are not drawn on map on simulator.

VI. Conclusion and Future Work

In this paper we combined two methods to achieve complete exploration and mapping solution: Ekman's exploration strategy and Pfister's weighted line fit algorithm. We improved convergence of the exploration strategy and produce accurately fitted line map based on modelled sensor and position uncertainty. The algorithms were experimentally tested and compared under the same conditions. In future work we will consider further simplifying line fit by doing measurement prediction at scan positions to eliminate "invisible" lines. By connecting endpoints as a final step we would turn line fit map into polygonal map for exploration algorithm.

Acknowledgement

This research has been supported by the Ministry of Science, Education and Sports of the Republic of Croatia under grant No. $036-0363078-3018$.

References

[1] G.A. Borges and M.J. Aldon. A split-and-merge segmentation algorithm for line extraction in 2Drange images. In *Pattern Recognition, 2000. Proceedings. 15th International Conference on*, volume 1, 2000.

[2] A. Ekman, A. Torne, and D. Stromberg. Exploration of polygonal environments using range data. *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, 27(2):250–255, 1997.

[3] J. Illingworth and J. Kittler. A survey of the hough transform. *Computer Vision, Graphics, and Image Processing*, 44(1):87 – 116, 1988.

[4] J. Matas, C. Galambos, and J. Kittler. Progressive probabilistic hough transform. In *Proc British Machine Vision Conference BMVC98*, pages 256–265, 1998.

[5] B. Oommen, S. Iyengar, N. Rao, and R. Kashyap. Robot navigation in unknown terrains using learned visibility graphs. part i: The disjoint convex obstacle case. *Robotics and Automation, IEEE Journal of*, 3(6):672–681, December 1987.

[6] Joseph O'Rourke, Heather Booth, and Richard Washington. Connect-the-dots: A new heuristic. *Computer Vision, Graphics, and Image Processing*, 39(2):258 – 266, 1987.

[7] S.T. Pfister. Weighted line fitting and merging. *California Institute of Technology, Tech. Rep*, 2002.

[8] S.T. Pfister. *Algorithms for mobile robot localization and mapping, incorporating detailed noise modeling and multi-scale feature extraction*. PhD thesis, California Institute of Technology, 2006.

[9] S.T. Pfister, SI Roumeliotis, and JW Burdick. Weighted line fitting algorithms for mobile robot map building and efficient data representation. In *IEEE International Conference on Robotics and Automation, 2003. Proceedings. ICRA'03*, volume 1, 2003.

[10] S.I. Roumeliotis and G.A. Bekey. SEGMENTS: a layered, dual-Kalman filter algorithm for indoorfeature extraction. In *2000 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2000.(IROS 2000). Proceedings*, volume 1, 2000.

[11] A. Siadat, A. Kaske, S. Klausmann, M. Dufaut, and R. Husson. An optimized segmentation method for a 2D laser-scanner applied to mobile robot navigation. In *Proceedings of the 3rd IFAC symposium on intelligent components and instruments for control applications*, 1997.

[12] J.M.R.S. Tavares and A.J. Padilha. A new approach for merging edge line segments. In *RecPad'95-7th Portuguese Conference on Pattern Recognition*, 1995.