Proceedings of the 1996 IEEE
International Conference on Robotics and Automation
Minneapolis, Minnesota - April 1996

Exact Dynamic Map Building for a Mobile Robot using Geometrical Primitives Produced by a 2D Range Finder

J. Vandorpe, H. Van Brussel, H. Xu

Division PMA, Department of Mechanical Engineering, Faculty of Engineering,
Katholieke Universiteit Leuven, Celestijnenlaan 300B, B-3001 Heverlee, Belgium
Tel: 32-16-322480 Fax: 32-16-322987
Email: jurgen.vandorpe@mech.kuleuven.ac.be

Abstract: *In this paper, a new mathematically exact algorithm is described for dynamic map building with geometrical primitives for a mobile robot. The dynamic map is built up using a 2D range finder mounted on the mobile robot LiAS[1] which is navigating in the environment. The dynamic map can be used for either planning or localisation purposes. The map is composed of line segments and circles. The parameters describing the geometrical primitives are provided with uncertainties which are used in the matching phase and which are necessary if the map is used for localisation. This paper describes in detail how the uncertainty on the robot position and the uncertainty on a single range measurement leads to the uncertainty on the parameters of a geometrical primitive. Promising experimental results obtained by the algorithm in real unstructured environments are presented.*

1. Introduction

An accurate perceived map of the working environment is necessary in two modules of an integrated mobile robot system. Intelligent real-time planners [1] need an up-to-date representation of the mobile robot's surroundings to find a path to the goal in a partly known environment. The map is also often used by a position estimation module where the perceived primitives in the dynamic map are compared to an a-priori known world model.

A widespread map building method is the occupancy grid based representation. The concept was introduced by Moravec and Elfes in the 80's [2],[3] and it was further enhanced and modified by various researchers [4],[5].

Occupancy grids represent the environment as a two-dimensional array of cells, each cell holding a value which represents the confidence in whether it is occupied space or free space. The low-level grid based approach proved to be very useful for map building using ultrasonic sensors. Because ultrasonic sensors have a large opening angle and their range data are seriously corrupted by reflections, reliable feature extraction for high level map building becomes very difficult. However, grid based map building algorithms proved to be very simple and quite useful for obstacle avoidance and planning purposes. In the past, also the PMA-mobile robot LiAS used an occupancy grid in a robust navigation algorithm [6],[7],[14].

We believe that the occupancy grid was developed specially for the ultrasonic sensors which are very cheap and are therefore a widespread used. With the introduction of (relative) low-cost and high-accuracy 2D range finders the occupancy grid seems no longer the best solution. Because the latest generation of range finders has a range of up to 50 meters and sample frequencies up to 1 Khz, the number of cells of which the confidence value has to be changed, is too large. Also the memory requirements for a global occupancy grid in a real (large environment) application are rather high.

The main drawback of grid based world models however, is probably the fact that it is impossible to use them directly for position estimation. In [8] Schiele and Crowley first extract line segments from the occupancy grid and match them to global segments in an a-priori model.

One of the most popular geometrical primitives is the line segment. Several researchers made efforts building line segment maps directly from the range data of 2D range finders. In [9] a method is presented where a set of short line segments approximate the shape of almost any kind of environment. A similar method was described in [10].

The here proposed method is different and more general than the methods mentioned above because it takes into account the uncertainties of the parameters before matching corresponding primitives. Doing so, the estimate of a certain primitive keeps improving while it is observed continuously. The resulting map contains a number of primitives each represented by a set of parameters with corresponding uncertainties.

[1] Leuven Intelligent Autonomous System

0-7803-2988-4/96 $4.00 © 1996 IEEE

Almost every shape can be approximated by line segments as long as the length of the lines is short enough. One major drawback of a map composed out of line segments only, is that small objects like table legs or other narrow poles are not modelled. This can cause serious problems if the map is to be used by a path-planning algorithm.

We propose a map built up by two different geometrical primitives. The first primitive is the line segment which is used to model all objects with a width exceeding 30 cm (Section 4). Measured points which lie close to each other but do not pass the criteria for line extraction are called clusters and are represented by circles. Figure 2 shows some groups of measurement points which each form a cluster and which will be represented by a small circle.

2. The Range Finder and the Mobile Robot

Figure 1 shows the PMA mobile robot LiAS. The vehicle is equipped with several sensors. Encoders and gyroscopes are used for deadreckoning.

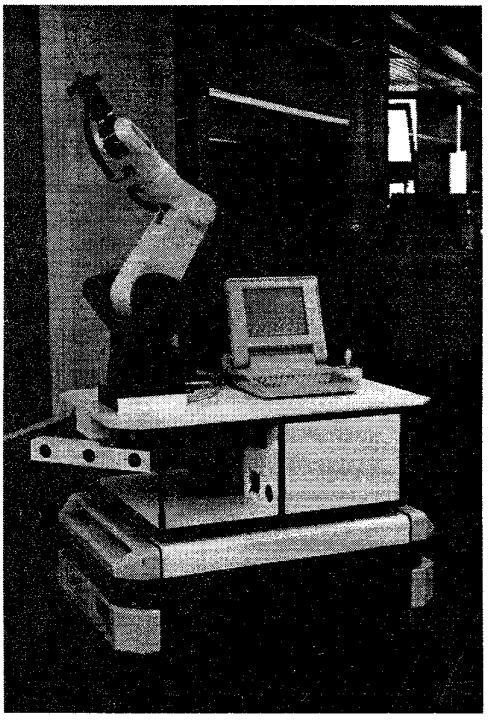

Figure 1: The mobile robot LiAS

For perception LiAS is equipped with ultrasonic sensors, a tri-aural acoustic radar and the new laser range finder which is used for the map building experiments described in this paper. A powerful on-board transputer system with 8 Transputer cards takes care of different modular tasks which are to be performed in parallel.

The 2D range finder is a commercial eye-safe laser scanner (PLS-scanner from Sick Optic Electronic). It has a scanning angle of 180 degrees with an angular resolution of 0.5 degrees and a range accuracy of less than 5 cm over a total range of 50 m. The scanning time for a complete scan of 360 range measurements takes about 600 msec. Figure 2 shows the raw sensory data from two consecutive scans of the range finder.

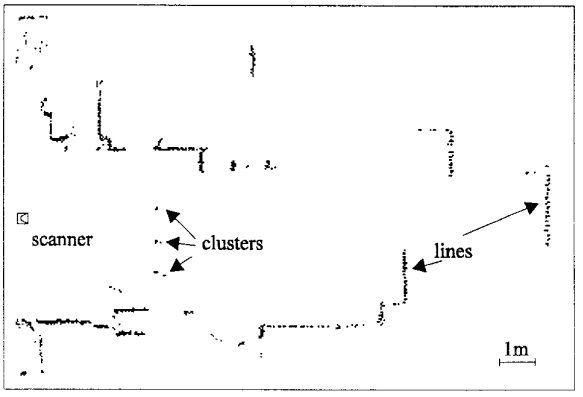

Figure 2: A double scan of the range finder

3. The Robot Position

Since the robot position has a direct influence on the parameters of the geometrical primitives, a good estimate is necessary. In [11], a sensor fusion algorithm for position estimation of the mobile robot LiAS was described. A Kalman filter combines the data from the encoders and gyroscopes with the information of beacons which are detected in the environment and compared to a known database.

The result of the Kalman filter algorithm is an accurate estimate of the robots pose vector $\mathbf{X}_r=[x,y,\theta]^T$ in a global reference frame and its corresponding covariance matrix $\mathbf{C}_{Xr}$.

4. The Geometrical Primitives

4.1. Introduction

Figure 3 shows the representation of the two basic primitives. Line segments are represented by parameters ρ and θ from following line equation:

$$y\sin\theta + x\cos\theta - \rho = 0; \quad (1)$$

by their corresponding covariances σ_ρ, σ_θ and $\sigma_{\rho\theta}$ represented by the covariance matrix $\mathbf{C}_l$;
and by their begin and end points in global cartesian coordinates: (x_b,y_b) and (x_e,y_e).

Circles are represented by the cartesian coordinates of the centre point (x_c,y_c) and by the radius R.

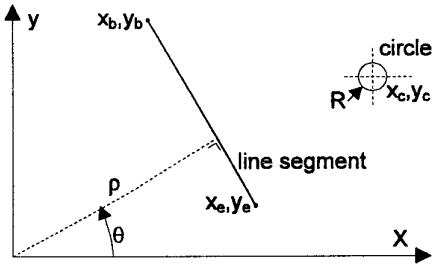

Figure 3: Parameters of a line segment and a circle

The next sections will describe how these parameters are calculated from the range measurements and the position estimate of the mobile robot in a global frame.

4.2. Conversion of measured points to the global coordinates

The range finder provides us with a range r and an angle α to a measured point, where the measurement vector $\mathbf{S}=[r,\alpha]^T$ is relative to the local reference frame of the mobile robot.

In a first step, the measured point is converted to global cartesian coordinates $\mathbf{X_p}=[x_p,y_p]^T$, using the current pose vector $\mathbf{X_r}$ and the relative position of the scanner $[\Delta x,\Delta y]^T$ on the robot with following expression:

$$\begin{aligned} x_p &= x_r + \Delta x\cos\theta_r - \Delta y\sin\theta_r + r\cos(\alpha+\theta_r), \\ y_p &= y_r + \Delta x\sin\theta_r + \Delta y\cos\theta_r + r\sin(\alpha+\theta_r); \end{aligned} \quad (2)$$

or more general with:

$$\mathbf{X_p} = f(\mathbf{X_r},\mathbf{S}). \quad (3)$$

To obtain the uncertainties on the final parameters of a line (ρ,θ), the uncertainties on each measured point $\mathbf{X_p}$ have to be calculated. The uncertainty of a measured point $\mathbf{X_p}$ is represented by a covariance matrix $\mathbf{C_p}$:

$$\mathbf{C_p} = \begin{bmatrix} \sigma_{xpxp} & \sigma_{xpyp} \\ \sigma_{ypxp} & \sigma_{ypyp} \end{bmatrix};$$

The matrix $\mathbf{C_p}$ depends on the uncertainty of the robots position represented by covariance matrix $\mathbf{C_{xr}}$ and on the uncertainty of the measurement (r,α) which is represented by the covariance matrix $\mathbf{C_S}$:

$$\mathbf{C_s} = \begin{bmatrix} \sigma_{rr} & 0 \\ 0 & \sigma_{\alpha\alpha} \end{bmatrix};$$

$\mathbf{C_S}$ is a diagonal matrix since the elements are not correlated.

The matrix $\mathbf{C_p}$ is calculated using the expression:

$$\mathbf{C_p} = \mathbf{F}\mathbf{C_{Xr}}\mathbf{F}^T + \mathbf{G}\mathbf{C_S}\mathbf{G}^T; \quad (4)$$

where $\mathbf{F}$ represents the Jacobian of f to $\mathbf{X_r}$ and $\mathbf{G}$ represents the Jacobian of f to $\mathbf{S}$.

4.3. Extraction of line segments

4.3.1. Criteria for line extraction

To add a newly measured point $\mathbf{X_p}=(x_p,y_p)$ to the current line, three conditions have to be verified:

(1) the distance of the new point to the previous point which was added to the line must be smaller than a maximum value d_{critp}.
(2) the distance of the new point to the current line under extraction must be smaller than a maximum value d_{critl}.
(3) the angular difference $\Delta\alpha$ between the new angle α_i and the angle α_{i-1} of the previous measured point must be smaller than a maximum value α_{crit}.

If any of these criteria is not met, the extraction is aborted and the parameters of the line are calculated (next sections). If the number of points belonging to the current line is less than 3, the second criterion (d_{critl}) must not be fulfilled because we state that at least three points are necessary to define a line.

Only line segments which are longer than 30 cm are transferred to the model. Shorter line segments are treated as clusters and are represented by a circle (Section 4.3.).

4.2.2. Linear regression

Although the final aim is to represent the line by general (ρ,θ)-parameters, an intermediate representation has to be calculated. The intermediate line parameters are (m,q) from the line equation $y = mx + q$ (regression y to x) or (s,t) from the line equation $x = sy + t$ (regression x to y). The choice between (m,q)- and (s,t)-parameters depends on the slope of the line in the global reference frame.

If a given measurement point $\mathbf{X_p}$ fits into the extracted line, the regression parameters R_x, R_y, R_{xx}, R_{yy} and R_{xy} are updated with this new point.

Given n points $\mathbf{X_p}$, the regression parameters are given by following expressions:

$$R_x = \sum_{i=1}^{n} x_{pi}\ ,\ R_y = \sum_{i=1}^{n} y_{pi}\ ,\ R_{xx} = \sum_{i=1}^{n} x_{pi}^2\ ,$$
$$R_{yy} = \sum_{i=1}^{n} y_{pi}^2\ ,\ R_{xy} = \sum_{i=1}^{n} x_{pi} y_{pi}\ . \qquad (5)$$

To distinguish between (m,q) or (s,t) parameters, N_1 and N_2 are defined as:

$$N_1 = R_{xx}n - R_x^2,\ N_2 = R_{yy}n - R_y^2 \qquad (6)$$

N_1 and N_2 represent the width of the cloud of the regression points along the X- and Y-axis. If N_1 is larger than N_2, the cloud of points lies more horizontally than vertically which makes regression of y to x ($y = mx + q$) favourable. If N_1 is smaller than N_2 the regression of x to y ($x = sy + t$) is selected.

With $T = R_{xy}n - R_xR_y$, the parameters (m,q) or (s,t) are given by following equations:

$$m = \frac{T}{N_1},\ q = \frac{(R_y - mR_x)}{n}, \qquad (7)$$
$$s = \frac{T}{N_2},\quad t = \frac{(R_x - sR_y)}{n}. \qquad (8)$$

4.2.3. Transition to (ρ,θ)-parameters

Transition to (ρ,θ)-parameters to represent a line is necessary because they are much more general. No distinction between more horizontal and more vertical lines has to be made. Another advantage of this representation is the fact that equation (1) is normalised. The distance d of a point (x_p,y_p) to the line is therefore simply given by evaluating equation (1) in the given point: $d = |x_p\ cos\theta + y_p\ sin\ \theta - \rho|$. This equation is used to calculate the distance d_{pl} of a new point to the current line and to compare it to d_{critl} in the second condition of section 4.2.2. Finally, (ρ,θ)-parameters with their covariances are required for the matching phase and for the calculation of a new estimate (section 5).

Following equations are used to calculate ρ and θ from (m,q) or from (s,t):

$$\rho = \left|\frac{q}{\sqrt{m^2+1}}\right| \text{ or } \rho = \left|\frac{t}{\sqrt{s^2+1}}\right|, \qquad (9)$$
$$\theta = \text{arctg}\left(\frac{q}{-q.m}\right) \text{ or } \theta = \text{arctg}\left(\frac{-t.s}{t}\right). \qquad (10)$$

4.2.4. Determination of the end points of the line

The parameters (ρ,θ) describe the equation of a boundless line. To obtain a line segment, the coordinates (x_b,y_b) and (x_e,y_e) of the begin and end point have to be determined. The first point $\mathbf{X_{p1}}$ which was added to the regression line is projected on the line resulting in the begin point. The projection of the last point leads to the end point.

4.2.5. Determination of the uncertainties on ρ and θ

After a line l has been extracted, it will be compared and identified with the lines which are already listed in the world model. If the line matches with a previously extracted line, it will be updated using the variances on both lines in a static Kalman filter.

Similar as in the calculation of ρ and θ, a two-step method is followed to calculate the covariance matrix:

$$\mathbf{C}_l = \begin{bmatrix} \sigma_{\rho\rho} & \sigma_{\rho\theta} \\ \sigma_{\rho\theta} & \sigma_{\theta\theta} \end{bmatrix}.$$

In a first step, the variances σ_{mm}, ,σ_{qq}, and σ_{mq} on m and q are calculated from the uncertainties on all the points $\mathbf{X_{pi}}$ belonging to the extracted line. In the second step the variances $\sigma_{\rho\rho}$,$\sigma_{\theta\theta}$ and $\sigma_{\rho\theta}$ on ρ and θ are calculated from σ_{mm}, σ_{qq} and σ_{mq}. Only the (m,q)-parameter case will be described. An analogue method is used for the (s,t)-parameter case.

Step 1:
The parameters m and q, represented by $\mathbf{V}=[m,q]^T$, are a function of all the measured points $\mathbf{X_{pi}}$ belonging to the line:

$$m = f_1((x_1,y_1),(x_2,y_2),...,(x_n,y_n)),$$
$$q = f_2((x_1,y_1),(x_2,y_2),...,(x_n,y_n)). \qquad (11)$$

$\mathbf{C_v} = \begin{bmatrix} \sigma_{mm} & \sigma_{mq} \\ \sigma_{mq} & \sigma_{qq} \end{bmatrix}$ is determined by the formula:

$$\mathbf{C_v} = \sum_{p=1}^{n} \mathbf{J}.\mathbf{C_p}.\mathbf{J}^T \qquad (12)$$

where **J** is the Jacobian of f_1 and f_2 to $\mathbf{X_{pi}}$ and $\mathbf{C_p}$ is the covariance matrix of point $\mathbf{X_{pi}}$.

Step 2:
$\sigma_{\rho\rho}$, $\sigma_{\theta\theta}$ and $\sigma_{\rho\theta}$ are calculated from σ_{mm}, σ_{qq} and σ_{mq}

Equation (8) can be written as:

$$\rho = f_3(m,q), \quad (13)$$

$$\theta = f_4(m). \quad (14)$$

If $\mathbf{X}_l = [\rho_l,\theta_l]^T$, then $\mathbf{C}_l$ finally is given by:

$$\mathbf{C}_l = \mathbf{H}_l.\mathbf{C}_v.\mathbf{H}_l^T \quad (15)$$

where $\mathbf{H}_l$ is the Jacobian of $\mathbf{X}_l$ to **V**.

The elements of $\mathbf{H}_l$ are given by:

$$h_{11} = \frac{\partial f_3}{\partial m} = -q.m.\frac{1}{(m^2+1)^{3/2}}, \; h_{22} = 0, \quad (16)$$

$$h_{21} = \frac{\partial f_3}{\partial q} = \frac{1}{\sqrt{m^2+1}}, \; h_{12} = \frac{\partial f_4}{\partial m} = \frac{1}{1+m^2}, \quad (17)$$

4.3. Extraction of circles

4.3.1. Criterion for circle extraction

If at least two consecutive points $\mathbf{X_{pi}}$ do not pass the criteria to be added to a line, there will be checked if these points form a cluster and can be represented by a circle. Only one condition has to be verified to add a new point $\mathbf{X_p}$ to a current cluster:

(1) the distance of the new point to the centre of the circle (equation (18)) must be smaller than a maximum value d_{critc}.

Circles which lie too close to a line segment in the map, are not modelled since they are probably noise resulting from the concerning line.

4.3.2. Determination of the circle parameters

The coordinates of the centre point (x_c,y_c) are calculated by these expressions:

$$x_c = R_x/n, \; y_c = R_y/n, \quad (18)$$

$$R = \sqrt{\sigma_x^2 + \sigma_y^2} \quad (19)$$

where R_x and R_y are similar to the ones defined by (5) and n is the number of cluster points.
σ_x and σ_y are given by:

$$\sigma_x^2 = \sum_{i=1}^{n}\frac{(x_i - x_c)^2}{n-1}, \; \sigma_y^2 = \sum_{i=1}^{n}\frac{(y_i - y_c)^2}{n-1} \quad (20)$$

5. Matching

5.1. Matching of line segments

5.1.1. Criteria for matching

After the extraction of a new line, the world model has to be updated. Two cases can be distinguished:

(1) The new line matches to a line in the model and is used to obtain a better estimate.
(2) The new line does not match to a line in the model and is transferred to the model directly.

A number of criteria have to be checked to make a distinction between the two cases.

A first eliminating test compares the angle difference $\Delta\theta$ to a threshold $\Delta\theta_{crit}$ and the distance of the end points of the new line to the lines in the model to a threshold d_{critl}.

When the first test is passed, a statistic test involving the covariance matrices of both lines is made.

Given the new line represented by $\mathbf{X}_l = [\rho_l,\theta_l]^T$ and covariance matrix $\mathbf{C}_l$ and the corresponding line in the model given by $\mathbf{X}_m = [\rho_m,\theta_m]^T$ and covariance matrix $\mathbf{C}_m$, a normalised distance is calculated using the expression:

$$\textit{normdist}(\mathbf{X}_l,\mathbf{C}_l,\mathbf{X}_m,\mathbf{C}_m) = (\mathbf{X}_l-\mathbf{X}_m)^T.(\mathbf{C}_l+\mathbf{C}_m)^{-1}.(\mathbf{X}_l-\mathbf{X}_m) \quad (21)$$

Using a χ^2-square distribution (with 2 degrees of freedom) where a significance level of approximately 90% is wanted, the condition for matching becomes:

$$\textit{normdist}(\mathbf{X}_l,\mathbf{C}_l,\mathbf{X}_m,\mathbf{C}_m) < 5, \quad (22)$$

Equation (22) means that there is a 10% chance that actual matching lines are treated as non-matching.

The previous tests only consider boundless lines. In a final test, the end points of matching lines are taken into account. A simple algorithm compares the distance along the line between the end points of the line segments to a maximum value d_{critl}.

5.1.2. Calculation of a new estimate after matching

Using two matching lines, a static Kalman filter calculates the resulting line which is represented by: $\mathbf{X}_r = [\rho_r, \theta_r]^T$ and by a covariance matrix $\mathbf{C}_r$:

$$\begin{aligned} \mathbf{K} &= \mathbf{C}_l(\mathbf{C}_l + \mathbf{C}_m)^{-1} \\ \mathbf{X}_r &= \mathbf{X}_l + \mathbf{K}(\mathbf{X}_m - \mathbf{X}_l) \\ \mathbf{C}_r &= \mathbf{C}_l - \mathbf{K}.\mathbf{C}_l \end{aligned} \quad (23)$$

The new estimate is better since the variances are smaller than the ones of either of the two lines.

Finally, the end points of the new estimate are determined. The end points of the extracted line are projected onto the new estimated line and compared to the end points of the previous estimate. The projected points replace the old end points if they make the line longer.

5.2. Matching of circles

5.2.1. Criterion for matching

Only one criterion is checked to decide if a new circle corresponds to a circle already in the world model:

(1) the distance between the centre points of both circles should be smaller than a maximum value d_{critcc}.

5.2.2. Calculation of a new estimate

Similar as in the case with line segments, matching circles are used to calculate a new estimate. If the new extracted circle is given by (x_{cn}, x_{cn}, R_n) and the matching circle in the model by (x_{cm}, y_{cm}, R_m), the new estimate (x_{cr}, y_{cr}, R_r) is given by this expression:

$$x_{cr} = (x_{cn}+x_{cm})/2,\ y_{cr} = (y_{cn}+y_{cn})/2, \quad (24)$$
$$R_r = (R_n+R_m)/2. \quad (25)$$

6. Dynamic Map Building: Deleting of Primitives

6.1. Introduction

Since the final aim is to build a dynamic map, objects which have been removed from the real world, have to be removed from the world model as well. When a new line is added to the world model or a new estimate of an existing line in the world model has been made, an algorithm checks if any primitives have to be deleted or modified. A triangular area (called wipe-triangle) between the scanner and the line segment is used to determine these primitives.

6.2. Construction of a wipe-triangle

For each new line or new estimate, the wipe-triangle is constructed as shown on figure 4. To filter out noisy lines in the map, the triangle has been made to extend slightly behind the new line.

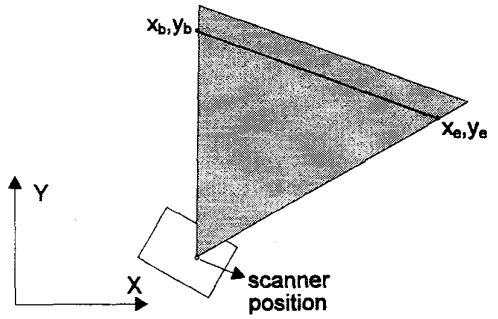

Figure 4: The wipe-triangle

6.3. Deleting or fragmentation of primitives

An algorithm checks every line in the map for intersection with the wipe-triangle. Four cases can be distinguished (Fig.5):

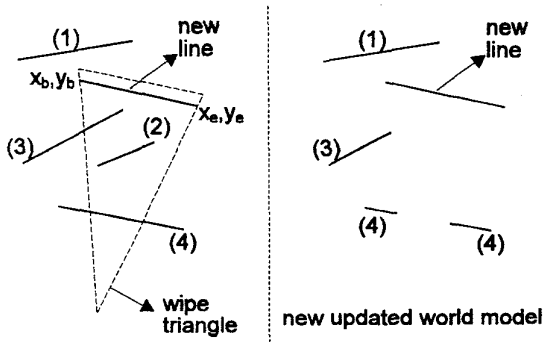

Figure 5: Deleting and fragmentation of line segments

(1) If the line segment lies completely outside the triangle no action has to be taken.
(2) If the line segment lies completely inside the triangle, it can be deleted.
(3) If the line segment has only one end point in the triangle, it has to be deleted partly.
(4) If the line segment crosses the triangle with both end points outside, the line segment has to be split up in two.

When a line segment is split up in two, the line fragments with a length smaller than 30 cm are removed from the world model. Circles are deleted as well if they lie inside the wipe-triangle.

7. Experimental Results

The map building technique described in this paper was implemented on a T800 transputer. With already 100 lines and 100 clusters in the world model, a single scan of 180 degrees is processed in less than 300 msec.

Figure 6 shows a world model representing part of the PMA-laboratory. Although the environment was extremely cluttered, only 52 line segments and 24 circles were needed to build this world model. As mentioned before, the uncertainty of the lines depends on the number of measurement points and on the number of times the line has been improved (new estimate) with a new matching line.

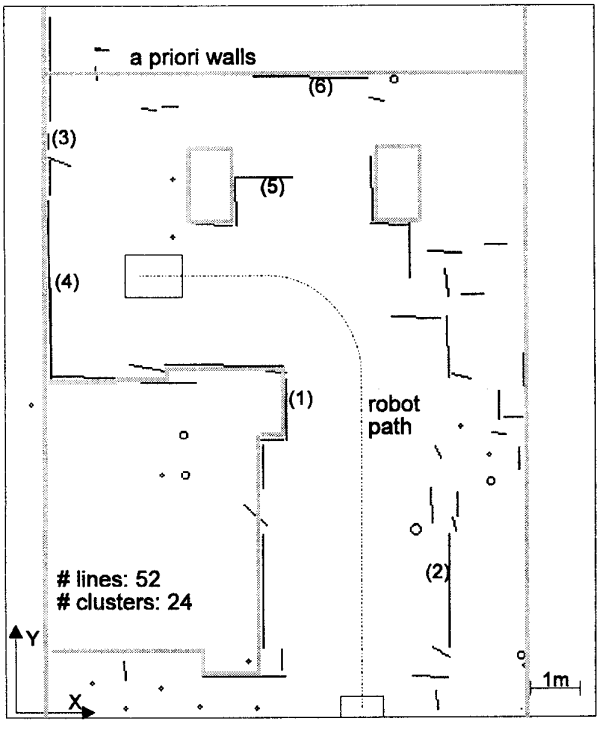

Figure 6: Resulting world model

Table I shows a number of line segments from figure 6 with their parameters and corresponding uncertainties.

Figure 7a shows a map of the environment after a robot trajectory of 20 cm during which 10 scans were taken. The map is composed out of 19 line segments and 7 circles. Figure 7b shows a new single scan. From this scan, 15 line segments and 9 circles were extracted. Only 3 lines are detected for the first time (in bold), the others match to the lines from fig.7a. From the 9 circles, 6 are newly detected. Figure 7c shows the resulting world model (22 lines and 8 circles) where the new circles which were too close to a line segment have been removed.

nr.	times detected	ρ *cm*	θ *deg*	σ_ρ *cm*	σ_θ *deg*
1	13	1155	89	99	0.64
2	12	762	90	41	0.35
3	7	1613	91	1109	4.5
4	10	1654	90	9	0.08
5	37	1714	0	28	0.27
6	38	1953	1	33	0.17

Table I: Parameters of some line segments in figure 6

8. Conclusions and Future Work

In this paper, some very good results of a map building algorithm with a 2D range finder were described. The parameters on the geometrical primitives are provided with uncertainties depending on the uncertainty of the robot position estimate and the uncertainties of all the measurements leading to this primitive.

The quality of the world model will allow the mobile robot to use the world model for localisation purposes. The localisation module can compare the perceived primitives to a known a-priori model using the corresponding uncertainties to estimate the robot position. The position estimation is based on a 'constraint' Kalman filtering technique and was already described in [11] using only ultrasonic sensors and a sensor detecting artificial beacons. Preliminary results of the position estimation with the range finder are very promising.

The world map was also used by the reactive planner which was described in [1]. Experiments which show that line segments only are not sufficient to build a complete model led to the introduction of the clusters, represented by circles.

Acknowledgements

The author would like to thank the four students who contributed to the work presented in this paper. Their work is described in detail in [12] and [13]. Further, this work was sponsored by the Flemish Institute for Scientific Research (IWT) and by the Belgian Program on Inter-

university Attraction Poles initiated by the Belgian State - Prime Minister's Office- Science Policy Programming. The scientific responsibility is assumed by its authors.

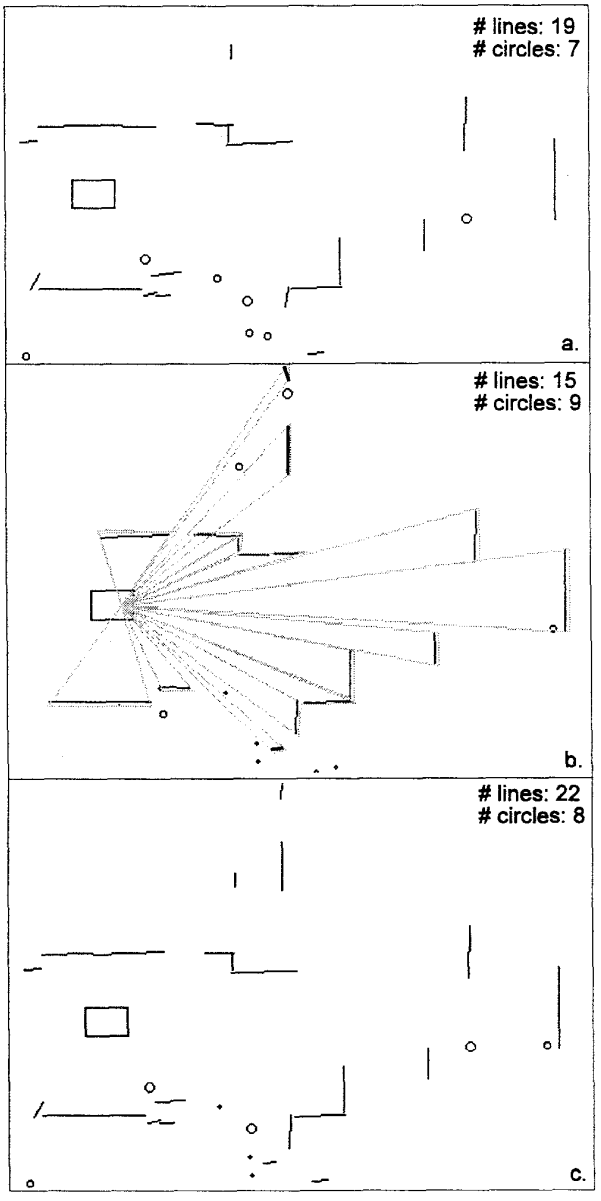

Figure 7: Updating of the world model

References

[1] Vandorpe J., Van Brussel H., 'A Reflexive Navigation Algorithm for an Autonomous Mobile Robot', Proc. Int Conf. on Multisensor Fusion and Integration of Intelligent Systems, 1995, 251-259.

[2] Elfes A., 'Using Occupancy Grids for Mobile Robot Perception and Navigation', IEEE, Computer, June 1989, 46-57.

[3] Moravec H.P., 'Sensor Fusion in Certainty Grids for Mobile Robots', AI Magazine 9 (2) 1988, 61-74.

[4] Cho D.W., Lim J.H., 'A New Certainty Grid Based Mapping and Navigation System for an Autonomous Mobile Robot', Int. J. Adv. Manuf. Technol., 1995, 139-148.

[5] Borenstein J., Koren Y., 'Histogrammic in Motion Mapping for Mobile Robot Obstacle Avoidance', IEEE Trans. on Rob. and Aut., vol 7, no 4, 1991, 535-539.

[6] Van Brussel H., Vandorpe J., Huang G.J., 'An Integrated Control System for Enhanced Autonomous Navigation of Mobile Robots', Proc. Sec. Int. Conf. on Mechatronics & Robotics, 1993, 297-318.

[7] Vandorpe J., Xu H., Van Brussel H., 'Dynamic Docking Integrated in a Navigation Archtecture for the Intelligent Mobile Robot LiAS', Proc. Int. Conf. Intellig. Autonomous Systems 4, 1995, 143-149.

[8] Schiele B., Crowley J.L., 'A Comparison of Position Estimation Techniques Using Occupancy Grids', Rob. and Auton. Systems 12, 1994, 153-171.

[9] Gonzales J., Ollero A., Reina A., 'Map Building for a Mobile Robot with a 2D Laser Rangefinder', Int. Conf. Rob. & Autom., 1994, 1904-1909.

[10] Fortarezza G., Oriolo G., Ulivi G., Vendittelli M., 'A Mobile Robot Localisation Method for Incremented Map Building and Navigation', Proc. of the 3rd Int. Symp. in Intell. Robotic Systems, 1995, 57-65.

[11] Xu H., Van Brussel H., De Schutter J., Vandorpe J. 'Sensor Fusion and Positioning of the Mobile Robot LiAS', Proc. Int. Conf. Intelligent Autonomous Systems 4, 1995, 246-253.

[12] Bogaerts D., Braeckmans W. 'Sensorfusie en Omgevingsperceptie voor Autonome Mobiele Robots', (in Dutch) Master thesis KU-Leuven, Division PMA, 94EP22, 1994.

[13] Aertbelien E., Van Gulck S. 'Sensorfusie voor Positieschatting en Kaartopbouw bij een Mobiele Robot', (in Dutch) Master thesis KU-Leuven, Division PMA, 95EP21, 1995.

[14] Vandorpe J., Van Brussel H., Xu H., 'LiAS: A Navigation Architecture for an Intelligent Mobile Robot System', To be published iin the Special Issue of IEEE Multisensor Fusion and Integration for Intelligent Systems in the Transactions on Industrial Electronics, April, 1996.