Laser Range Finder and Advanced Sonar Based Simultaneous Localization and Mapping for Mobile Robots

Albert Diosi

BE in Control Systems Engineering (1999)
Masters Degree in Control Systems Engineering (2001)
Slovak University of Technology in Bratislava, Slovakia

A thesis submitted to the

Department of Electrical and Computer Systems Engineering
Monash University
Clayton, Victoria 3800, Australia

in fulfillment of the requirements for the degree of

Doctor of Philosophy

August 2005

To the memory of Kálmán Végh,
a very good friend

Contents

Abstract

Mobile robots are likely to play an important role in the lives of humans in the future. To efficiently perform tasks, the capability of simultaneous localization and mapping (SLAM) is often necessary for mobile robots. This thesis develops fundamental building blocks for use in the SLAM process.

To perform simultaneous localization and mapping with a Kalman filter using line segments extracted from laser range finder measurements, an accurate measurement error model is necessary. A novel line fitting approach working in the native polar coordinate system of a laser range finder is described which enables simple and accurate error modeling. The line fitting approach works by minimizing the sum of square range residuals by the iterative application of linear regression on the linearized problem. To investigate the random and systematic line parameter errors, a range error model is constructed which entails the following error components: constant bias, bias growing linearly with range, bias changing with incidence angle, quantization bias and zero mean white Gaussian noise. The effects of motion and laser plane alignment error on line parameter accuracy is also investigated. The range error models are calibrated for Sick PLS and LMS laser range finders in experiments. Then the line parameter error models are experimentally evaluated. The described line fitting approach is adapted for right angle corner fitting.

The robustness of SLAM can be increased with sensor fusion. When different sensors are measuring the same percept, false negatives can be rejected and measurement precision can be improved. In case of complementary sensing, each sensor measures a different percept which complement each other. A novel sensor fusion scheme is described in which laser range finder measurements are fused with advanced sonar measurements. Advanced sonars, unlike conventional sonars can accurately measure the range and bearing of targets classified as planes, corners and edges. In the discussed fusion scheme advanced sonar aids laser segmentation, laser aids good sonar point feature selection and laser and sonar line and right angle corner measurements of the same object are fused. This sensor fusion scheme is then evaluated in SLAM experiments.

Having a sparse feature map might help with localization, but path planning is best handled using occupancy grids. It is possible to register laser scans into an occupancy grid by using

the pose of the robot provided from the SLAM approach, however motion of the SLAM map makes this approach ineffective. In a novel approach discussed in this thesis, with each laser scan the relative position of the neighboring map features are also stored. When needed, the stored laser scans can be registered into an occupancy grid by regenerating their pose with respect to the SLAM map using the stored local features. This approach allows the generation of occupancy grids which are consistent with the feature based SLAM map.

A consistent set of laser scans can be also created by performing SLAM using scan matching. In this thesis the novel Polar Scan Matching (PSM) approach is described which works in the laser scanner's polar coordinate system, therefore taking advantage of the structure of the laser measurements by eliminating the search for corresponding points. PSM belongs to the family of point-to-point matching approaches with its matching bearing association rule. The performance of PSM is evaluated in a simulated experiment, in experiments using ground truth, in experiments aimed at determining the area of convergence and in a SLAM experiment. All results are compared to results obtained using an iterated closest point (ICP) scan matching algorithm implementation.

Declaration

I declare that:

1. This thesis contains no material that has been accepted for the award of any other degree or diploma in any university or institution, and
2. To the best of my knowledge and belief, it contains no material previously published or written by another person, except when due reference is made in the text of the thesis.

.....
Albert Diosi

Acknowledgment

Without having MGS and MIRS scholarships of Monash University paying both fees and living expenses, the commencement of my PhD studies would have not been possible.

I am grateful to my supervisor Lindsay Kleeman for his guidance, ideas, support and for sharing his time with me whenever it was needed. Thanks to my associate supervisor Andy Russell for providing valuable feedback on the manuscript of this thesis. Thanks to Steve Armstrong for technical support and in keeping the mobile robot SLAMbot ever ready for experiments.

Thanks to my lab mate Geoff Taylor for being a great friend, for many fruitful discussions and for proofreading conference papers and this thesis. Thanks Geoff for making my stay in Australia so enjoyable. Special thanks to the Taylor family for the happy times I spent at their home and for letting me to be part of their family.

I appreciate the support of my mum Maria, my dad Berti, my sister Hajni and her husband Al.

Chapter 1

Introduction

1.1 Motivation

Robots have always interested me since my first encounter with them at the age of 7. I found the possibility of creating moving and one day perhaps intelligent creatures other than the "traditional way" tantalizing. Even though I find industrial robots interesting and the quick motion of their sometimes 1000kg arms amazing, mobile robots are more exciting to me. Their potential to move to different places, do different things, to interact with people and to be helpful in households made them more fun to me than industrial robots. There are flying, swimming, walking, crawling and rolling mobile robots. From all of these, for me wheeled indoor mobile robots are the most easy to experiment with, therefore they can provide the most fun. These were my personal, subjective reasons for the work discussed in this thesis.

Research laboratories are no longer the only keepers of mobile robots. Mobile robots are slowly starting to conquer households. In the year 2004, hundreds of thousands of vacuum cleaning robots of the company iRobot were sold in the world. Even though the robots emerging in households are still simple, with time they will get more complex and intelligent. The next logical step after having simple vacuum cleaning robots roaming randomly around in houses are robots which can answer the three basic questions of navigation [Leonard and Durrant-Whyte, 1991a]: "Where am I?", "Where am I going?", and "How should I get there?". This thesis addresses the first of these three questions.

The reasonable answer to the "Where am I?" question is another question "With respect to what?". Outdoor robots can use the global positioning system (GPS) to obtain their position in the Earth's reference frame. However indoor mobile robots often have to rely on their sensors to localize themselves with respect to a map. It is hard to expect that the owners of the household robots of the future will provide the robot with a map of their home. Instead the robot should map its own environment and use it to localize itself. The process when a robot builds its own map while using the map to localize itself by accounting for correlations between robot

pose and map features is called simultaneous localization and mapping (SLAM) [Leonard and Durrant-Whyte, 1991b] or concurrent localization and mapping (CML) [Feder *et al.*, 1998]. Most of the SLAM approaches take advantage of statistical representation of measurements, therefore beside having good quality measurements, it is also important to have error estimates of the measurements. The work described in this thesis does not attempt to improve an existing or devise a new SLAM algorithm, instead it contributes to the sensing side of SLAM. Fitting features to laser range data and modeling their error, the fusion of laser range finder and advanced sonar (described in chapter 3) measurements and a laser scan matching approach are investigated in this thesis together with their application to SLAM.

Simultaneous localization and mapping has been performed for example using 2D laser range finders, 3D laser range finders, conventional sonar sensors, advanced sonar sensors (described in chapter 3), monocular vision, stereo vision and omni-directional vision. The ideal sensor for mobile robot navigation classifies and recognizes all objects and provides estimates of their pose. Since such a sensor is not yet available, the choice of sensing modalities should be influenced by the operating environment of the robot and the requirements for accuracy and reliability of localization and mapping. The following thoughts played a role in the choice of the exteroceptive sensors of the mobile robot SLAMbot (more about SLAMbot in chapter 3) used in this thesis:

- In typical office environments, common 2D laser range finders such as the products of the SICK company are almost sufficient for localization and mapping tasks. The situations where navigation systems based on such laser range finders are likely to fail are buildings with glass walls which cannot be sensed reliably, or long corridors with features so small that they cannot be detected due to noise in the range measurements.

- Advanced sonar sensors as described in chapter 3 accurately measure range and bearing of targets classified as planes, corners or edges. For such sonar systems specular reflections and objects covered with sound absorbing materials such as partition walls cause problems.

- Using 3D laser range finders for mapping and localization result in higher requirements for computations due to the increased number of measurements compared to the use of 2D laser range finders. Solely for the task of localization and mapping in typical indoor environments, a 3D laser range finder with its higher material and computational cost may not be feasible.

- Vision sensors provide a lot of information, however building a system for localization and mapping which can deal with light conditions changing from darkness to direct sunlight, shadows, reflections and colors changing with changing lighting is a complex and computationally expensive task.

- Using conventional sonars measuring only range with large uncertainty in the bearing may be a relatively inexpensive solution, however the accuracy of mapping and localization is not high. This solution also fails in rooms having sound absorbing objects.

When using advanced sonars or laser range finders separately, localization and mapping likely fails in buildings containing glass walls, long corridors with few features and walls made from sound absorbing materials. When these two sensors are fused together (e.g. as described in chapter 3) the likelihood of failure can be reduced. Sound absorbing walls are visible for the laser, glass walls or doors, small features such as door jambs or moldings on the walls of long otherwise feature depleted corridors are easily detected by advanced sonars.

When fusing measurements from multiple sensors in a statistical framework, it is important to have accurate error models for correct weighting of measurements. Laser range finders are no exception either. In the literature, with the derivation of error estimates of lines fitted to laser range finder measurements, white Gaussian noise in the range and bearing measurements is often assumed to be the only source of error. Experimental validation of these assumptions and the accuracy of the resulting error models are likely missing. The failure of our first attempts at advanced sonar and SICK PLS laser range finder fusion due to the lack of satisfactory analysis of errors of lines fitted to laser range measurements provided the motivation to explore how accurately line segment measurements can be made.

Many simultaneous localization and mapping applications extract particular features from measurements and use them as landmarks. It does not matter if the features are lines, corners or something else, for the correct functioning of SLAM the chosen feature types have to be present in the environment. It is however possible to perform SLAM without the use of features. In laser scan matching the relative position of two laser scans maximizing overlap of the scans with respect to some chosen criterion are sought. In point-to-point scan matching algorithms raw laser scans are used without the need for feature extraction. However, point-to-point matching approaches are likely to be slow due to the need to perform a search for corresponding points of the two scans which are matched. A point-to-point scan matching approach which does not require such a search can be faster, resulting in a higher SLAM update rate or in the possibility of a SLAM implementation on a cheaper robot with a slower processor.

1.2 Scope of the Research

The work discussed in this thesis is limited to indoor localization and mapping for mobile robots equipped with laser range finders, advanced sonar arrays and odometry. The discussions can be divided into four topics. The first topic regards laser range finder range error modeling, line and right angle corner fitting to range data in polar coordinates, and the anal-

ysis of effects of different error sources on the accuracy of estimated line parameters. The second topic regards the fusion of laser rage finder and advanced sonar measurements with application to SLAM. The third topic regards occupancy grid generation using laser range finder measurements while performing feature based simultaneous localization and mapping. The fourth topic regards laser scan matching in the laser range finder's polar coordinate system with application to SLAM.

1.3 Thesis Outline

When measurements of sensors are used in a statistical framework such as the Kalman filter [1], accurate measurement error models are needed. Overly conservative error estimates lead to wasting information and overly optimistic error estimates can cause divergence. In **chapter 2** fitting line segments and right angle corners to laser range finder measurements, and the accuracy of line segment parameters are investigated. In chapter 2 first a range error model is described for time-of-flight laser range finders consisting of the following error components: constant bias, bias growing linearly with range, bias changing with incidence angle of laser beam and target surface normal, quantization bias and zero mean white Gaussian noise. Then a line fitting algorithm is developed which works in laser range finders native polar coordinate systems by minimizing the sum of square range residuals. This line fitting approach enables simple covariance estimation of line parameters. Since the laser range error model contains systematic error components beside the random error component used in the line parameter covariance estimation, the effects of systematic errors are also investigated. Laser range finders in this thesis are used on a mobile robot, therefore the effects of misaligned mounting of the laser range finder and the motion of the robot on line parameter accuracy are also investigated. The structure of the range error model is experimentally justified in experiments using Sick PLS and LMS laser range finders. Finally, estimated and measured line parameter random and systematic errors are experimentally compared. In chapter 2 the line fitting approach working in the polar coordinate frame of laser range finders is extended to right angle corners as well. Derivations of some of the mathematical formulas of chapter 2 are described in **appendix A**.

The corner and line features of chapter 2 are used in **chapter 3** where the synergy of laser range finder and advanced sonar array measurements are discussed. The advanced sonar arrays are unique sonar sensors in that they measure range and bearing to targets classified as planes, corners and edges while rejecting interference from other sonars by using random pulse encoding. In the devised synergy scheme, sonar measurements help to segment laser scans into lines and right angle corners, laser scans help to reject spurious sonar measurements

[1] A Kalman filter is a recursive optimal linear estimator. The extended Kalman filter is a modification for nonlinear systems. For more on Kalman filters the reader is referred to [Bar-Shalom and Li, 1993].

and to select good sonar point reflectors and then sonar and laser measurements of the same object are fused by calculating an average of the measurements weighted by their information matrix. The described sensor fusion scheme is validated using SLAM.

Sparse feature maps are good for localization, however global path planning usually requires occupancy grid maps. In chapter 3 the construction of occupancy grid maps using feature based SLAM and laser range finder measurements is also discussed. By knowing the path of the robot, building an occupancy grid from laser measurements is an easy task. Simultaneous localization and mapping can provide accurate robot pose estimates with respect to the SLAM map. Building occupancy grids by using the pose estimate of the robot from SLAM results in inaccurate grid maps since with each SLAM feature update the SLAM map may shift. The shifting of the SLAM map may result in laser scans not registering well with the occupancy grid. In the solution to this problem described in chapter 3, each laser scan is stored together with the robot's pose with respect to neighboring SLAM map features. Then in the occupancy grid building process, all stored robot poses are reconstructed with respect to the current SLAM map using a process similar to scan matching on the features stored with the pose and the features in the SLAM map. **Appendix B** contains a description of features used in the SLAM process of chapter 3.

Simultaneous localization and mapping using a laser range finder can be performed without the extraction of features from scans. In **chapter 4** a fast laser scan matching approach called polar scan matching (PSM) and its application to SLAM is described. PSM finds the pose of the current scan in the frame of the reference scan by minimizing the sum of square range residuals of the current and reference scans. Range readings of the current scan are associated with those range readings of the reference scan which share the same bearing in the reference frame. The matching bearing association rule eliminates the need for an expensive search for corresponding points. The performance of PSM is experimentally evaluated in four experiments.

1.4 Contributions

The contributions of this thesis can be summarized in the following points:

- *Approaches for fitting lines and right angle corners to laser range finder measurements directly in the laser range finder's native polar coordinate system.* Unlike other approaches, residuals of model and unmodified measurements are minimized which result in simple and accurate covariance estimation. Our publications related to line and right angle corner fitting are [Diosi and Kleeman, 2003a; 2003b; 2004].

- *Comprehensive line error model.* Unlike in other works where it is assumed that the only source of line parameter errors are random errors in laser range finder measure-

ments, here the effects of 8 error sources, most of them systematic are considered. Not neglecting systematic errors, unlike the common practice, is important since as shown in experiments, systematic line parameter errors can be larger than random ones. Having good line error models in line feature based SLAM is important, since the performance of the SLAM depends on the quality of measurement error models. Unlike in other works where the line error models are only assumed to be correct, the random and systematic components of the comprehensive line error model are tested in experiments with two laser range finders. Our publications related to line and right angle corner fitting are [Diosi and Kleeman, 2003a; 2003b].

- *Advanced sonar and laser synergy with application to SLAM.* No previous work has considered laser and advanced sonar fusion for simultaneous localization and mapping. There is one other example for fusion of advanced sonar and laser for only mapping in the literature [Vandorpe *et al.*, 1996], where only range and bearing information of advanced sonar measurements were used by transferring them together with the laser measurements into a grid map. The laser and advanced sonar synergy scheme of this chapter is much more sophisticated, and uses all the information provided by advanced sonars. The synergy scheme unites the best properties of both sensors and enables the successful performance of SLAM in environments where SLAM using only one of the sensors would fail. Our publications related to sonar and laser fusion are [Diosi and Kleeman, 2004; Diosi *et al.*, 2005].

- *Occupancy grid generation in feature based SLAM.* Unlike another approach [Bourgault *et al.*, 2002] where laser scans are registered into an occupancy grid using the pose estimate of the robot provided by feature based SLAM, in this thesis robot pose estimates together with corresponding laser scans are stored with respect to neighboring SLAM map features. When the occupancy grid is generated, the pose corresponding to each laser scan is restored in the final SLAM map using the stored local features. This approach prevents "blurring" of the occupancy grid due to the motion of the SLAM map when past robot locations are revisited and enables the construction of an occupancy grid which is consistent with the SLAM map. The consistency of the SLAM map and occupancy grid is important for using the SLAM map for localization and the occupancy grid for path planning. Our publication related to occupancy grid generation is [Diosi *et al.*, 2005].

- *Polar scan matching algorithm (PSM).* PSM works in the polar coordinate systems of laser range finders by minimizing the weighted sum of square residuals of range readings of pairs of scans being matched. Unlike other point-to-point scan matching approaches where expensive searches are applied to find corresponding points in

pairs of scans, the matching bearing association rule of PSM eliminates the need for search which results in a fast scan matching algorithm. Speed is important for on-board, on-line SLAM applications. Our publications related to polar scan matching are [Diosi and Kleeman, 2005b; 2005a]. The source code of PSM can be downloaded from www.irrc.monash.edu.au/adiosi.

Chapter 2

Laser Range Finder Features and Error Models

It is important to know the accuracy of measurements when fusing measurements from different sources in a statistical framework. Underestimating or overestimating measurement accuracy can lead to incorrect weighting of the individual measurements which results in information loss. This chapter is oriented towards fitting lines and corners to laser measurements and their error estimation which sets the stage for chapter 3 where laser range finder measurements are fused with advanced sonar measurements for mobile robot simultaneous localization and mapping.

2.1 Introduction

This chapter is concerned with fitting line segments and right angle corners to laser range finder measurements and with the modeling of their errors. Fitting geometric primitives such as lines or corners to range data with a suitable approach can ease the problem of estimating their error. Having good measurement error models is important when using the measurements in a Kalman filter context.

Laser range finder measurements can be used processed or unprocessed in mobile robot mapping and localization applications. Examples of the use of unprocessed laser scans are point-to-point scan matching approaches (see chapter 4) where there is no need to extract features from laser scans. Often laser scans are processed by extracting features from them. One of the most popular features is line segments. Line segments are abundant indoors or in structured outdoor environments. Corner features or sometimes even range extrema [Lingemann *et al.*, 2004] are also used as features. In outdoor applications one can find even tree trunks[Nieto *et al.*, 2002] as features extracted from laser scans. In this chapter line segments and right angle corner features are chosen as features of interest since they are also measured by the advanced

sonar arrays introduced in the next chapter where advanced sonar and laser lines and corners are fused. Other reasons for the use of line segments are that they are abundant, and a single observation of a line map feature can correct the orientation of a mobile robot. Observations of right angle corners used in this chapter can correct not just the orientation, but also the pose of a robot.

Line parameter error estimates depend on the line fitting methods and on range error models. There are several approaches for fitting lines to range data in the literature. In [Horn and Schmidt, 1995] Hough transformation is used to find planes representing walls in 3D laser scans. Using least squares minimization, a plane in its general form was fitted to the points of the wall. The intersection of vertical planes with the floor was calculated and the resulting line was converted into its normal form. The uncertainty of the line parameters was calculated using Taylor series expansion. However, the measured points were assumed to be affected only by nonsystematic uncorrelated noise in the bearings and in the range.

In [Nygårds and Wernersson, 1998], a local Cartesian coordinate system is placed into the center of gravity of a line segment, with the vertical axis pointing in the opposite direction, than that of the laser. The regression coefficient of the line is determined by linear regression, which is sensitive to noise for nearly vertical lines. From knowledge of the center of gravity and the regression coefficient, the angle and perpendicular distance parameters of the normal form of a line are calculated. The covariance of the angle and distance estimate of the line is derived, under the assumption of error free laser bearings.

Jensfelt [2001] uses the solution given in [Deriche *et al.*, 1992], where the angle and distance parameters of a line are estimated by minimizing the sum of square perpendicular distances of the points from the line in a Cartesian coordinate system. A simple covariance estimate is given of the line parameters, assuming uniform covariance of each point. However this assumption is only valid for short line segments if data is obtained from a laser range finder utilizing a rotating mirror.

Similarly to [Jensfelt, 2001] in [Arras and Siegwart, 1997] the authors minimized the sum of square perpendicular distances of points to a line in a Cartesian coordinate system, however their solution uses nonuniform weights of points. They also show an equivalent solution with polar coordinates, which was used to derive a line parameter covariance estimate assuming only errors in the range measurements.

Pfister et al. in [2003] also minimize the sum of square perpendicular distances of points to a line, however each point is weighted by their uncertainty which includes contributions from random range and bearing errors of laser measurements. Angle and distance parameters of the line are calculated with a maximum likelihood approach resulting in an iterative solution which is more computationally expensive than the previously described approaches. Experimental evaluation of their line error model is not presented.

Contrary to the previously described methods, in [Taylor and Probert, 1996] the authors take advantage of the description of a line in a polar coordinate system, and minimize the sum of square errors of reciprocal ranges. However, due to the use of the reciprocal of the range, their approach implicitly weights closer points more than further ones.

In all of these papers, systematic errors are neglected in line estimates, and little experimental evidence is presented to support the error models. For some lasers as shown in this chapter, systematic errors can be a significant component of the final errors in line parameters.

One of the sources of systematic errors in line parameter estimates are systematic errors in the range measurements. In the literature there are papers on laser range finder characterization covering Amplitude and Frequency Modulated Continuous Wave and time-of-flight (TOF) lasers. This chapter concentrates on TOF lasers such as the Sick PLS and LMS. In [Reina and Gonzales, 1997] and [Ye and Borenstein, 2002]) the authors investigate the accuracy of a laser range finder from Schwartz Electro-optics Inc., and a Sick LMS 200. They found that systematic errors in range changed with the time of operation, the reflectivity of the surface, and the incidence angle of laser beam and target surface. In [Reina and Gonzales, 1997] a scale factor error is also reported.

In practice, uncertainty in the exact location of the laser with respect to the robot can contribute to the error in the laser measurements as well. The problem of calibrating the laser's position with respect to the robots frame of reference is addressed in [Krotkov, 1990]. Time registration errors of the laser measurements on a moving robot can contribute to line parameter errors as well, especially on robots executing fast turns. Laser measurements are compensated for motion in [Wang and Thorpe, 2002] and [Arras, 2003].

The line fitting and error modeling approach of this chapter is extended to right angle corners as well, since this allows fusion of these corners to those classified by the advanced sonar introduced in the next chapter. Another approach to fitting a corner to points representing a right angle corner is minimizing the square sum of perpendicular distances of points from two orthogonal lines as in [Gander and Hřebíček, 1993], essentially by solving a constrained least squares problem. However when results are to be used in a Kalman filter, an accurate error model of the corner estimate is also necessary. The polar corner fitting approach described in this chapter provides such an error model in a simple way.

This chapter is organized as follows. A range error model is described in section 2.2. In this model range error is composed from bias, error increasing proportionally with range, error increasing with increasing incidence angle, quantization bias and random error components. An approach for line parameter estimation directly in polar coordinates is presented in section 2.3.2 that enables a simple line parameter covariance estimation. In the polar line estimation approach the sum of square range residuals is minimized by an iterative application of linear regression to the linearized problem. Then the effects of systematic range errors on

systematic line parameter errors are investigated. Closed form approximations are derived for some of the systematic line error estimates. The effects of laser plane misalignment and robot motion on line parameter errors is also investigated. In section 2.4 a right angle corner fitting approach working with polar coordinates is also presented. In this corner fitting approach, similarly to the line fitting approach the sum of square range residuals is minimized by the iterative application of linear regression to the linearized problem. Finally in section 2.5 first the range error model is validated using a Sick PLS and LMS laser range finder followed by the experimental evaluation of the random and systematic line parameter error models.

Note that throughout this chapter the term "closed form" is used in a stricter sense than its general meaning in mathematics. The term closed form is used in this chapter for expressions that can be evaluated by the calculation of a finite number of operations, and they do not contain sums over all measurements of line segments.

The work presented in this chapter is published in [Diosi and Kleeman, 2003a], [Diosi and Kleeman, 2003b] and [Diosi and Kleeman, 2004].

2.2 Laser Range Error Model

In general, range errors of time of flight lasers can be related to four main sources:

- errors due to varying returned signal strengths.
- errors due to change in the electrical properties of the laser's components with temperature.
- random electrical noise in the receiver electronics.
- error due to the measurement of time of flight with finite resolution which has the same effect as a truncating quantizer.

To accurately model errors from all error sources for all laser range finders is a task which is likely to be achievable only through the investment of large amount of time and effort. Since the aim of the range error model in this thesis is to provide an upper bound for the random and systematic laser line parameter errors, the construction of error models with sufficiently high accuracy for the correction of range errors is not necessary.

Next individual components of the range error model of this section are described. The total error model that includes each component is shown in (2.8). The first component in the range error model is constant bias r_b. This bias is modeled as constant for all range measurements corresponding to the measured line segment. Such bias can be present in many laser range finder models. The constant bias can be used, for example, to express such systematic errors

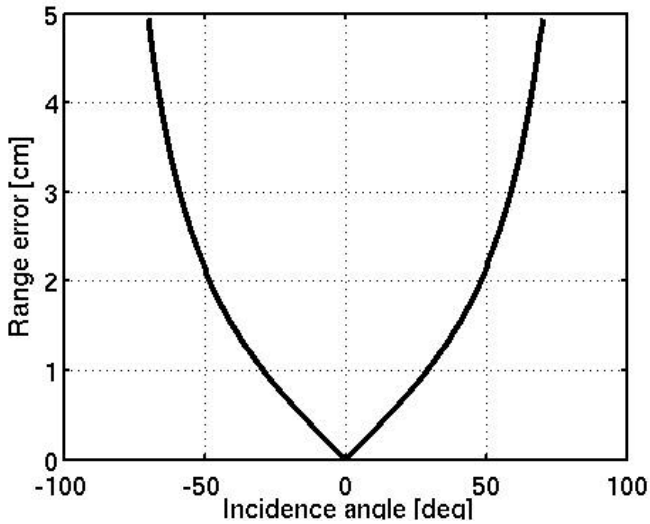


Figure 2.1: Model of range error changing with incidence angle of laser beam and target surface normal.

in laser measurements as range errors occurring during the warm-up of the sensor or the range error changing with the reflectivity of the target.

The second error component in the range error model is a linear term. The linear term $k(r' - r_{min})$ depends on the true range r' and accounts for error increasing with distance. Distance in the linear term is measured from the shortest range reading r_{min} of the line measurement. Coefficient k is determined experimentally. The linear term can be viewed for example as a linear approximation of range measurement errors which depend on distance. The source of such errors can be, for example, signal strength reducing with distance. If a laser beam illuminates a Lambertian surface, the received light intensity is assumed to decrease with range. Smaller received light intensity can result in a slower rise of the laser range finder receiver's output, which when compared to a threshold can cause longer measured time of flight. The linear error term is a good approximation if the error in the measured time of flight changes approximately linearly with the change of range in the range interval of the measured line segment.

The third term in the range error model approximates the effect of the incidence angle β of the laser beam and the target surface normal on the range error. The model is described as $w|\tan(\beta)|$ and is shown in fig. 2.1. The structure of this model has been identified from measurements with a Sick PLS, however the presence of an error with such shape in [Reina and Gonzales, 1997] suggests that such a systematic error exists in multiple laser range finder models. The likely cause of this error is that with increasing incidence angle less and less light is reflected back to the laser receiver, which can cause the receiver's output to rise slower and slower.

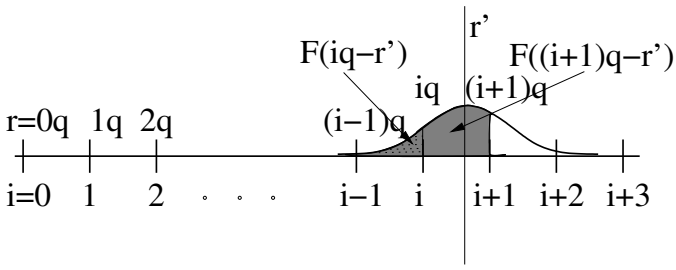


Figure 2.2: The probability of measuring $r_i = iq$ on a truncating quantizer's output with an input of r' added to random white Gaussian noise.

Laser range finders which measure time of flight by counting clock periods are also burdened with a quantization error. To model the effects of quantization, it is assumed that white noise $r_{nr} \sim N(0, \sigma_n^2)$ added to the true range r' enters an ideal truncating quantizer. The truncating quantizer is described as

$$Q(r') = \lfloor \frac{r'}{q_r} \rfloor q_r, \tag{2.1}$$

where the operator $\lfloor\ \rfloor$ finds the largest integer less than its argument and q_r is the quantization step. For the range error model, the mean and variance of the range error at the quantizer's output is sought. To calculate the mean and variance, first $P(r_i|r')$, the probability of having r_i on the quantizer's output given the true range r' needs to be found. $P(r_i|r')$ is found by calculating the probability (see fig. 2.2) of the sum of the noise and the true measurement falling between $r_i = iq_r$ and $r_{i+1} = (i+1)q_r$. Therefore $P(r_i|r')$ is calculated by subtracting the probability of the noise being smaller than $iq_r - r'$ from the probability of the noise being smaller than $(i+1)q_r - r'$:

$$P(r_i|r') = F\left((i+1)\,q_r - r'; 0; \sigma_n^2\right) - F\left(iq_r - r'; 0; \sigma_n^2\right), \tag{2.2}$$

where the notation $F(x;\mu;\sigma^2)$ denotes the cumulative distribution function of a normal random variable with mean μ and variance σ^2:

$$F(x;\mu;\sigma^2) = \int_{-\infty}^{x} N(x,\mu,\sigma^2)dx \tag{2.3}$$

Then the mean error for given a r' equals the sum of all possible errors $(iq_r - r')$ weighted by their probability $P(r_i|r')$:

$$E(Q(r' + r_{nr}) - r') = \sum_{i=-\infty}^{+\infty} \left[\left(iq_r - r'\right) P(r_i|r')\right] \tag{2.4}$$

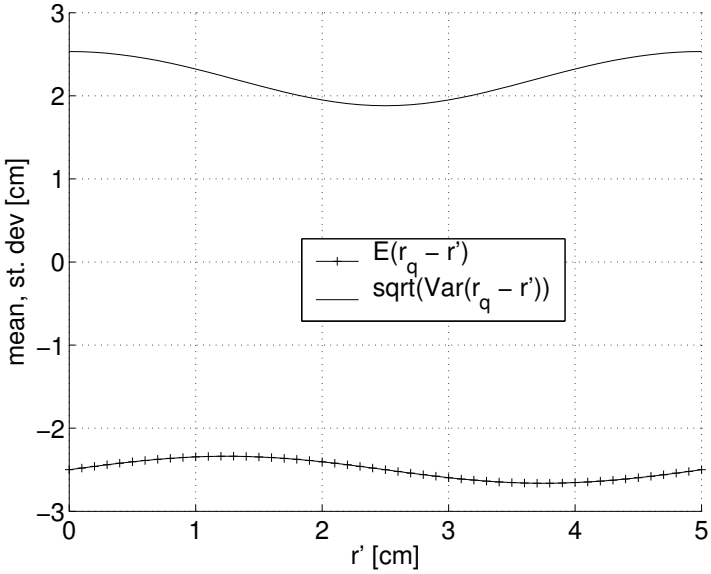


Figure 2.3: Simulated effect of quantization for random noise standard deviation $\sigma_n = 1.7cm$ and $q_r = 5cm$ quantization step for the quantization model of a laser range finder.

The variance of the error for r' is calculated using the formula $Var(X) = E(X^2) - (E(X))^2$:

$$Var(Q(r' + r_{nr}) - r') = \sum_{i=-\infty}^{+\infty} \left[\left(iq_r - r'\right)^2 P(r_i|r') \right] - \left[E(Q(r' + r_{nr}) - r') \right]^2 \tag{2.5}$$

When investigating the quantization effect in the output of a Sick PLS sensor, Jensfelt [Jensfelt, 2001] presents similar equations to (2.2), (2.5), however he assumes a rounding quantizer, and ignores the bias in the quantization error.

To visualize the mean and standard deviation of the quantization bias, (2.2)–(2.5) have been numerically calculated for the values $\sigma_n = 1.7cm$, $q_r = 5cm$ and the results are shown in fig. 2.3. $\sigma_n = 1.7cm$, $q_r = 5cm$ were chosen to describe the Sick PLS's output. Even though the calculation of (2.4)–(2.5) requires too much time to be useful in the range error model, by observing fig. 2.3 one can notice that the mean quantization error or quantization bias and the standard deviation of the noise have the shape of sinusoid functions. Equations (2.4)–(2.5) can be well approximated by functions

$$r_{qb} = E(Q(r' + r_{nr}) - r') \approx b \sin\left((r' - Q(r')) \frac{2\pi}{q_r} \right) - \frac{q_r}{2} \tag{2.6}$$

$$\sigma_r = \sqrt{Var(Q(r' + r_{nr}) - r')} \approx k_1 \cos\left((r' - Q(r')) \frac{2\pi}{q_r} \right) + k_2. \tag{2.7}$$

The approximations (2.6)–(2.7) have been tested for cases ($q_r = 1cm$, $q_r = 5cm$, $\sigma_{nr} \in < 0.5cm, 3cm >$) and it was found that residuals are negligible.

The last two terms of the range error model contain white Gaussian noise r_n and the quantization bias r_{qb} from (2.6) without the term $\frac{q_r}{2}$ which is assumed to be taken care of by the manufacturer or included into r_b. $r_n \sim N(0, \sigma_r^2)$ is the noise in the output of the quantizer. Instead of calculating σ_r from (2.7), for simplicity it is approximated as a constant.

The range error model which plays an important role in the error estimation of line fit parameters discussed in the next section is summarized in one equation as:

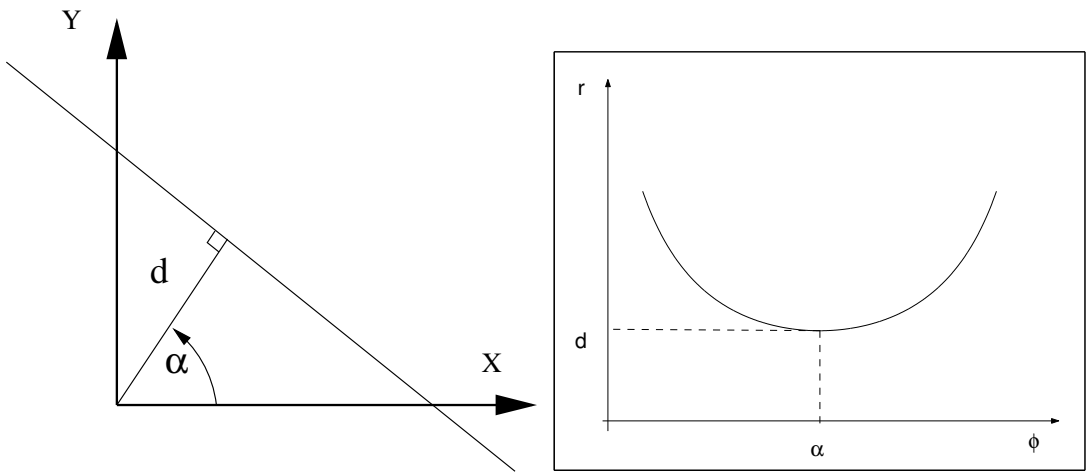
$$\Delta r \approx r_b + k(r' - r_{min}) + w\,|\tan\beta| + b\sin\left((r' - Q(r'))\frac{2\pi}{q_r}\right) + r_n, \tag{2.8}$$

where r_b is bias which is constant for all readings. The linear term depending on the true range r' accounts for error increasing with distance. r_{min} in the linear term $k(r' - r_{min})$ is the shortest range measurement corresponding to the measured line. The next term approximates the effect of the incidence angle β of the laser beam and the target surface normal on the range error. The last two terms in (2.8) account for quantization and random errors in the electronics, where $Q(r')$ is the quantized representation of the true measurement, q_r is the quantization step and r_n is zero mean white Gaussian noise with σ_r standard deviation.

2.3 Line Segments

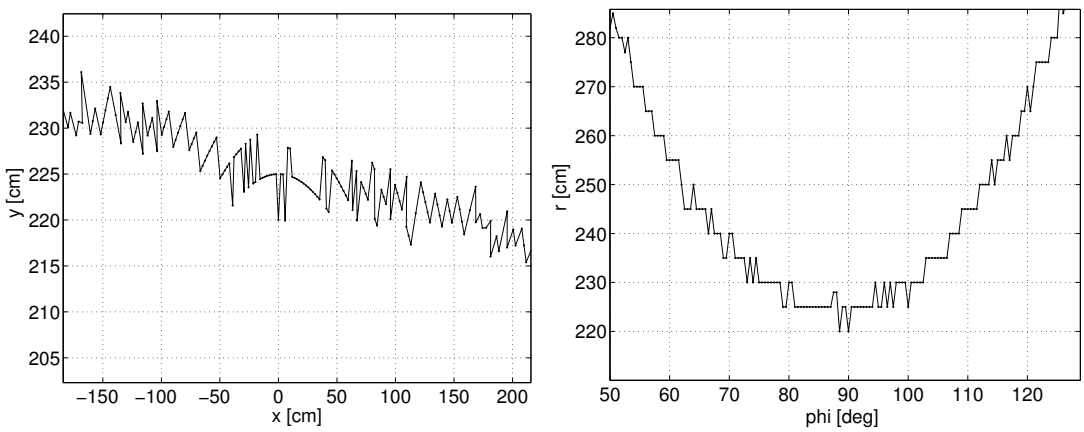
In this section line parameter and line parameter error estimation is investigated. In section 2.3.1 the normal form of line representation is discussed, followed by the description of a line fitting approach which works with laser range finder measurements in their native polar coordinate system. Then the effects of each component of the range error model of section 2.2 together with the effects of laser range finder misalignment and motion on the estimated line parameters are discussed. At the end in section 2.3.10 a concise summary of the line parameter estimation results is given together with applications.

To make line parameter systematic error estimation tractable, a simplifying assumption is made. It is assumed that systematic errors in the laser range measurements and the resulting systematic errors in the line parameter estimates are so small that the principle of superposition works. That is error contributions from different error sources can be calculated separately and the resulting error equals the sum of these errors. However, to obtain a worst case error bound, in section 2.3.10 the absolute values of errors are summed up.



(a) (b)

Figure 2.4: Line in Cartesian (X,Y) and in polar (R,Φ) coordinate system.



(a) (b)

Figure 2.5: Laser scan of a line in Cartesian (X,Y) and in polar (R,Φ) coordinate system.

2.3.1 Line Representation

The normal form representation of a line is described by the following equation in a Cartesian coordinate system (X,Y) (see fig. 2.4a):

$$x\cos\alpha + y\sin\alpha = d \tag{2.9}$$

where α is the angle between the X axis and the normal of the line, and $d \geq 0$ is the perpendicular distance of the line to the origin. However, x and y are calculated from the angle of the laser beam (ϕ) and the measured range (r). Therefore it is often more convenient to work with a line in the laser range finder's polar coordinate system (Φ,R) (see fig. 2.4b), where a line is represented by the equation (see section A.1 of the appendix for derivation):

$$r = \frac{d}{\cos(\alpha-\phi)} \tag{2.10}$$

As it can be seen from fig. 2.4b, the curve representing a line is uniquely described by the coordinates of its minimum (α,d).

An example of a real laser scan of a wall is shown in fig. 2.5 in Cartesian and polar coordinate systems.

2.3.2 Line Parameter Estimation

Several approaches can be used to identify the parameters of a line from measured data points. For example, the equations for linear regression (2.12)–(2.13) can be used to determine the parameters of a line in slope-intercept form (2.11) and then convert the resulting line into normal form using (2.14)–(2.15) (see appendix A.2). E.g:

$$y = kx + q \tag{2.11}$$

where parameters (k,q) are obtained by minimizing the cost function $\sum(y_i-\hat{y}_i)^2$:

$$k = \frac{n\sum x_i y_i - (\sum x_i)(\sum y_i)}{n\sum x_i^2 - (\sum x_i)^2} \tag{2.12}$$

$$q = \frac{\sum y_i - k\sum x_i}{n} \tag{2.13}$$

where x_i, y_i are Cartesian coordinates of the measured n line points and $\hat{y}_i$ are estimated Y coordinates. Then use

$$\alpha = \arccos\left[\frac{-k}{\sqrt{1+k^2}}\right] + (\mathrm{sign}(q)-1)\frac{\pi}{2} \tag{2.14}$$

$$d = \frac{|q|}{\sqrt{1+k^2}} \tag{2.15}$$

The drawbacks of the above mentioned approach are the following:

1. For vertical lines the results are imprecise due to numerical instability resulting from small numbers in the denominator of (2.12).

2. The cost function being minimized does not reflect the way the data points were collected. The points being processed in (X,Y) are the result of a nonlinear transformation of points from (Φ,R) what makes errors in the x and y coordinates correlated. Due to the cost function, not the sum of squared distances of points from the line is minimized, but the sum of square errors in y coordinates. Thus errors in x coordinates are not regarded.

3. The derivation of a covariance estimate for (k,q) based upon the uncertainty in (ϕ,r) is not simple.

A better approach than the previous one is to minimize the sum of square perpendicular distances of points from lines as in [Jensfelt, 2001]. However, in this section an approach is developed for finding out (α,d) with its estimated uncertainty directly in (Φ,R) coordinate system. Since the equation of a line (2.10) in a polar coordinate system is nonlinear, the proposed solution involves multiple iterations and the linearization of (2.10). Due to the iterative nature of the solution described bellow, the approach may be slower than other, non-iterative approaches.

If (2.10) is linearized around (α_0,d_0), one gets:

$$r_i - r_{0i} \approx \frac{d_0 \sin(\alpha_0-\phi_i)}{\cos^2(\alpha_0-\phi_i)}\Delta\alpha + \frac{1}{\cos(\alpha_0-\phi_i)}\Delta d \tag{2.16}$$

This is restated in vector form as

$$\Delta\mathbf{r} = \mathbf{r_m} - \mathbf{r_0} = \mathbf{H_0}\Delta\mathbf{b} + \mathbf{R} \tag{2.17}$$

Where

$$\mathbf{H_0} = \begin{bmatrix} \cdots & \cdots \\ \frac{d_0 \sin(\alpha_0-\phi_i)}{\cos^2(\alpha_0-\phi_i)} & \frac{1}{\cos(\alpha_0-\phi_i)} \\ \cdots & \cdots \end{bmatrix} \tag{2.18}$$

$$\Delta\mathbf{b} = \begin{bmatrix} \Delta\alpha & \Delta d \end{bmatrix}^T \tag{2.19}$$

$\mathbf{R}$ is a vector of measurement noise with a diagonal covariance matrix $\sigma_r^2 I$, $\mathbf{r_m}$ is a vector containing measured ranges and $\mathbf{r_0}$ is a vector representing ranges estimated using (α_0,d_0).

Using the common linear regression (see [Wetherill, 1986]) iteratively on the linearized problem (2.16), (α, d) minimizing the square sum of range residuals can be found the following way:

$$\begin{aligned}
\mathbf{r_j} &= \begin{bmatrix} r_{j1} & \dots & r_{ji} & \dots & r_{jn} \end{bmatrix}^T \\
&= \begin{bmatrix} \dots & \frac{d_j}{\cos(\alpha_j - \phi_i)} & \dots \end{bmatrix}^T && (2.20) \\
\mathbf{H_j} &= \begin{bmatrix} \dots & \dots \\ \frac{d_j \sin(\alpha_j - \phi_i)}{\cos^2(\alpha_j - \phi_i)} & \frac{1}{\cos(\alpha_j - \phi_i)} \\ \dots & \dots \end{bmatrix} && (2.21) \\
\Delta\mathbf{b} &= \left(\mathbf{H_j^T H_j}\right)^{-1} \mathbf{H}_j^T (\mathbf{r_m} - \mathbf{r_j}) && (2.22) \\
\begin{bmatrix} \alpha_{j+1} \\ d_{j+1} \end{bmatrix} &= \begin{bmatrix} \alpha_j \\ d_j \end{bmatrix} + \Delta\mathbf{b} && (2.23)
\end{aligned}$$

(2.22) yields the least squares estimate, and can be found for example in [Kay, 1993]. By initializing $\mathbf{r_j}$ with (α_0, d_0) obtained from either (2.12)–(2.15), or from minimizing the perpendicular distance of points from the line, this iterative process converges quickly. The advantage of the above mentioned approach is a simple covariance estimate described next.

2.3.3 Random Error Estimate

Assuming that range measurements r_{mi} consist of zero mean white Gaussian noise $r_{ni} \sim N(0, \sigma_r^2)$ superimposed on the true values r_i':

$$r_{mi} = r_i' + r_{ni}, \tag{2.24}$$

the line parameter covariance estimate is obtained due to the use of linear regression [Wetherill, 1986]:

$$cov(\Delta\mathbf{b}) = cov(\alpha, d) = \sigma_r^2 (\mathbf{H^T H})^{-1}. \tag{2.25}$$

The assumption of independent noise in the range measurements is examined later in the experimental results section for the Sick PLS. The implicit assumption of error free laser bearings is also justified for the Sick PLS and LMS later in the experimental results section, where measured and predicted line parameter covariances are compared. However the error free laser bearing assumption is not reasonable for moving laser scanners. The effects of motion are discussed in section 2.3.9.

2.3.4 Identical Bias in the Range Measurements

In this section the effect of laser range measurement bias on the estimated line parameters is discussed. Two cases are investigated. In the first case it is assumed that the bias in range measurements is identical for all range measurements of a scan, however the value of the bias is a random variable which changes with each laser scan. In the second case it is assumed that, the bias does not change with each laser scan. These two cases are differentiated because the error in the first case is a random variable and therefore it is modeled with a covariance matrix and averaging subsequent observations of the line can reduce the error. The error in the second case is not a random variable and its effect cannot be reduced by averaging.

Bias of the same size in each range measurement of a scan appears as a curvature in a line which normally would look straight in a Cartesian coordinate system. Positive bias deforms lines to appear concave, and negative bias deforms lines to appear convex.

In order to derive a formula which approximates the relation between bias in the range measurements and error in the angle and distance parameters of the measured line, the following assumptions are introduced:

- The measured line is horizontal, e.g. $\alpha = \frac{\pi}{2}$. It will be shown later, that this assumption has no effect on the result.
- The analysis done for datasets containing only bias error (r_b) in the range measurements is valid for datasets containing random and systematic errors as well.

As already mentioned in the beginning of section 2.3, the second assumption is justified with having such small errors in the laser measurements and in the resulting line parameter errors, that the principle of superposition is valid.

If the line being investigated is not horizontal, one needs to shift the measurement bearings using:

$$\phi_i = \phi_{im} - \hat{\alpha} + \frac{\pi}{2} \tag{2.26}$$

where $\hat{\alpha}$ is the estimated angle of the line and ϕ_{im} are the measured bearings and then set $\alpha = \frac{\pi}{2}$. In this chapter, this process is called line normalization. This line normalization has no effect on the error estimates (α_e, d_e), because it corresponds only to a shift in a polar coordinate system. The benefit of line normalization is significant, since α disappears from all equations which simplifies the systematic angle and distance error approximation.

The equations for linear regression (2.12)–(2.13) for the slope-intercept form of lines are used in the derivation of line parameter errors due to identical bias in the measurements. Then the calculated error in slope and intercept is converted into error in angle and distance. Since all calculations are performed for horizontal lines ($\alpha = \pi/2$), the denominator of (2.12) does not become small and therefore it does not cause numerical instability as in the case of vertical lines.

If $\alpha = \pi/2$ then the equation of a line in polar coordinate system (2.10) becomes

$$r = \frac{d}{\sin\phi} \tag{2.27}$$

and measured points in a Cartesian coordinate system become:

$$x_i = x'_i + x_{ei} = r'_i \cos\phi_i + r_b \cos\phi_i = d' \cot\phi_i + r_b \cos\phi_i \tag{2.28}$$

$$y_i = y'_i + y_{ei} = r'_i \sin\phi_i + r_b \sin\phi_i = d' + r_b \sin\phi_i \tag{2.29}$$

where x'_i, y'_i are the true coordinates of a point on the line, x_{ei}, y_{ei} are errors due to bias r_b in the range measurements, d' is the true distance of the line from the origin (what is assumed to be known) and r'_i is the true range. Equations (2.28)–(2.29) use (2.27) with r' and d' substituting r and d. The error in the slope k called k_e is calculated by substituting (2.28)–(2.29) into (2.12):

$$\begin{aligned} k_e &= k - k' = k - 0 = \\ &= \frac{n\sum x'_i y'_i - \sum x'_i \sum y'_i + r_b d' \left(n\sum\cos\phi_i - \sum\cot\phi_i \sum\sin\phi_i\right) + r_b^2 \left(n\sum\cos\phi_i \sin\phi_i - \sum\cos\phi_i \sum\sin\phi_i\right)}{n\sum x_i'^2 - \left(\sum x'_i\right)^2 + 2d' r_b \left(n\sum\frac{\cos^2\phi_i}{\sin\phi_i} - \sum\cot\phi_i \sum\cos\phi_i\right) + r_b^2 \left(n\sum\cos^2\phi_i - \left(\sum\cos\phi_i\right)^2\right)} \end{aligned} \tag{2.30}$$

Because the line would be horizontal without bias (i.e. $r_b = 0$), the sum of the terms containing x'_i, y'_i in the numerator is 0. In practice due to $d' \gg r_b$ the term containing r_b^2 is much smaller that the term containing $r_b d'$ in the numerator, and terms containing x'_i in the denominator are much larger than the rest of the terms in the denominator. The terms containing x'_i in the denominator are large, because $x_i'^2$ contains d'^2. To show that

$$\left|r_b d' \left(n\sum\cos\phi_i - \sum\cot\phi_i \sum\sin\phi_i\right)\right| \gg \left|r_b^2 \left(n\sum\cos\phi_i \sin\phi_i - \sum\cos\phi_i \sum\sin\phi_i\right)\right|$$

and

$$\left|n\sum x_i'^2 - \left(\sum x'_i\right)^2\right| \gg \left|2d' r_b \left(n\sum\frac{\cos^2\phi_i}{\sin\phi_i} - \sum\cot\phi_i \sum\cos\phi_i\right)\right| + \left|r_b^2 \left(n\sum\cos^2\phi_i - \left(\sum\cos\phi_i\right)^2\right)\right|$$

are true in practical situations, they are enumerated in table 2.1 for combinations $d' = 50cm$, $d' = 100cm$, $\phi_i = 10°, 11°, .., 90°$, $\phi_i = 10°, 11°, .., 170°$ and $r_b = 5cm$.

The result after the removal of small terms:

$$k_e \approx \frac{d' \left(n\sum\cos\phi_i - \sum\cot\phi_i \sum\sin\phi_i\right)}{n\sum x_i'^2 - \left(\sum x'_i\right)^2} r_b. \tag{2.31}$$

Term	$d' = 50$ $\phi_i = 10°$, $11°,..,170°$	$d' = 50$ $\phi_i = 10°$, $11°,..,90°$	$d' = 100$ $\phi_i = 10°$, $11°,..,170°$	$d' = 100$ $\phi_i = 10°$, $11°,..,90°$
$r_b d' \left(n\sum\cos\phi_i - \sum\cot\phi_i \sum\sin\phi_i\right)$	-4.9e+04	-3.1e+04	-9.8e+04	-6.1e+04
$r_b^2 \left(n\sum\cos\phi_i \sin\phi_i - \sum\cos\phi_i \sum\sin\phi_i\right)$	1.3e+03	7.4e+02	1.3e+03	7.4e+02
$n\sum x_i'^2 - \left(\sum x_i'\right)^2$	2.6e+10	9.4e+08	1.0e+11	3.8e+09
$2d' r_b (...) + r_b^2 (...)$	-1.4e+07	1.0e+06	-2.8e+07	2.1e+06

Table 2.1: Example values of terms in (2.30) for $r_b = 5$.

Instead of x_i' it is possible to substitute:

$$x_i' = r_i' \cos\phi_i = \frac{d'}{\sin\phi_i}\cos\phi_i = d'\cot\phi_i \tag{2.32}$$

after that one gets:

$$k_e \approx \frac{r_b}{d'}\frac{n\sum\cos\phi_i - \sum\cot\phi_i\sum\sin\phi_i}{n\sum\cot^2\phi_i - \left(\sum\cot\phi_i\right)^2} \tag{2.33}$$

The derivation of the error in the y-intercept q_e, (2.13) is now considered:

$$q_e = q - q' = \frac{1}{n}\left[\sum y_i - k_e\sum x_i\right] - \frac{1}{n}\left[\sum y_i' - k'\sum x_i'\right] \tag{2.34}$$

Because the line is horizontal ($k' = 0$), the term containing k' can be removed from (2.34). Furthermore (2.28)–(2.29) can be substituted into y_i and x_i to get:

$$\begin{aligned} q_e &= \frac{1}{n}\left(\sum d' + r_b\sum\sin\phi_i - k_e\sum\left[d'\cot\phi_i + r_b\cos\phi_i\right] - \sum d'\right) \\ &= \frac{r_b}{n}\left(\sum\sin\phi_i - k_e\sum\cos\phi_i\right) - \frac{d'k_e}{n}\sum\cot\phi_i \end{aligned} \tag{2.35}$$

The initial aim was to find the angle error α_e and the distance error d_e. Taking a first order Taylor expansion of a simplified version of (2.14)–(2.15):

$$\alpha = \arccos\frac{-k}{\sqrt{1+k^2}} \tag{2.36}$$

$$d = \frac{q}{\sqrt{1+k^2}} \tag{2.37}$$

for $k_e << 1$, it is shown that $\alpha_e \approx k_e$ and $d_e \approx q_e$:

$$\alpha_e \approx \left[\frac{\partial}{\partial k}\arccos\frac{-k}{\sqrt{1+k^2}}\right]_{k=0} k_e$$

$$= \left[\frac{-1}{\sqrt{1-\left(\frac{-k}{\sqrt{1+k^2}}\right)^2}} \left(-\frac{1}{\sqrt{1+k^2}} + \frac{k^2}{\sqrt{(1+k^2)^3}} \right) \right]_{k=0} k_e = k_e \tag{2.38}$$

$$d_e \approx \left[\frac{\partial d}{\partial q}\right]_{\substack{k=0\\q=0}} q_e + \left[\frac{\partial d}{\partial k}\right]_{\substack{k=0\\q=0}} k_e$$

$$= \left[\frac{1}{\sqrt{1+k^2}}\right]_{\substack{k=0\\q=0}} q_e - \left[\frac{qk}{\sqrt{(1+k^2)^3}}\right]_{\substack{k=0\\q=0}} k_e = q_e \tag{2.39}$$

The computational burden of the calculation of the angle and distance error can be reduced if all sums in (2.33), (2.35) are replaced with integrals, like:

$$\sum_{i=1}^{n} f(\phi_i) \approx \frac{n}{\Delta\phi} \int_{\phi_1}^{\phi_n} f(\phi) d\phi \tag{2.40}$$

where $\Delta\phi = \phi_n - \phi_1$. The error committed due to this approximation will be small due to the high angle resolution (for example $q_\phi = 0.5°$) of laser scanners. Note that (2.40) is similar to the numerical integration problem, except in (2.40) a sum is approximated by an integral. An upper bound for the approximation error can be obtained by modifying the upper bound rule for the Riemann sum approximation integral [Krell Institute, read in 2005]:

$$\left| \int_a^b f(x)dx - \sum f(x_i)h \right| \leq M_1 \left(\frac{b-a}{2} \right) h, \tag{2.41}$$

where a, b are the lower and upper bounds for integration, $f(x)$ is a continuous function over $< a, b >$ to be integrated, h is the numerical integration step and $M_1 = \max |f'(x)|$, $x \in < a, b >$. If (2.41) is divided by h, $\Delta\phi/n$ is substituted into h, a and b are substituted with ϕ_1 and ϕ_n and $b-a$ is replaced with $\Delta\phi$, an upper error bound for approximation (2.40) is obtained:

$$\left| \frac{n}{\Delta\phi} \int_{\phi_1}^{\phi_n} f(\phi) d\phi - \sum f(\phi_i) \right| \leq M_1 \left(\frac{\Delta\phi}{2} \right). \tag{2.42}$$

Using (2.40) as a template, the following approximations:

$$\sum \cos\phi_i \approx \frac{n}{\Delta\phi} (\sin\phi_n - \sin\phi_1) \tag{2.43}$$

$$\sum \sin\phi_i \approx \frac{n}{\Delta\phi} (\cos\phi_1 - \cos\phi_n) \tag{2.44}$$

$$\sum \cot\phi_i \approx \frac{n}{\Delta\phi} (\ln|\sin\phi_n| - \ln|\sin\phi_1|) = \frac{n}{\Delta\phi} \ln\left|\frac{\sin\phi_n}{\sin\phi_1}\right| \tag{2.45}$$

$$\sum \cot^2 \phi_i \approx \frac{n}{\Delta\phi}\left[\phi_1 + \cot\phi_1 - \phi_n - \cot\phi_n\right] = \frac{\cot\phi_1 - \cot\phi_n}{\Delta\phi} n - n \tag{2.46}$$

can be substituted into (2.33) and (2.35) to derive the following closed form approximations for the error in angle and distance due to constant range bias r_b:

$$\alpha_e \approx k_e = -\frac{r_b}{d'} \frac{\Delta\phi\left(\sin\phi_n - \sin\phi_1\right) + \ln\left|\frac{\sin\phi_n}{\sin\phi_1}\right|\left(\cos\phi_n - \cos\phi_1\right)}{\Delta\phi^2 + \Delta\phi\left(\cot\phi_n - \cot\phi_1\right) + \ln^2\left|\frac{\sin\phi_n}{\sin\phi_1}\right|} \tag{2.47}$$

$$d_e \approx q_e = \frac{r_b}{\Delta\phi}\left(\cos\phi_1 - \cos\phi_n - k_e(\sin\phi_n - \sin\phi_1) - \frac{d' k_e}{r_b} \ln\left|\frac{\sin\phi_n}{\sin\phi_1}\right|\right) \tag{2.48}$$

As already mentioned in the beginning of this chapter, the bias r_b can also be viewed as a constant for all range measurements within a scan, however randomly changing between scans. The analysis for this case is shown in section A.3 of the appendix, since these results have not been used in the experiments described later.

Note that in order to use the closed form approximations for systematic line parameter errors (2.47)–(2.48) not all laser measurement bearings need to be shifted using (2.26) in the line normalization process. Since only the bearing of the first and last range measurements are used in the closed form approximations, it is enough to shift the bearing of first and last measurement:

$$\phi_1 = \phi_{1m} - \hat{\alpha} + \frac{\pi}{2} \tag{2.49}$$

$$\phi_n = \phi_{nm} - \hat{\alpha} + \frac{\pi}{2}. \tag{2.50}$$

2.3.5 Error Changing with Incidence Angle

In this section the effect of the error component $\Delta r_{ai} = w\,|\tan\beta_i|$ of the range error model (2.8) of section 2.2 on the estimated line parameters is investigated. In $w\,|\tan\beta_i|$, w is a parameter which is identified experimentally and β_i is the angle between laser beam and surface normal. The final result represents a closed form approximation for the angle and distance systematic errors which are then used in the experimental results presented in section 2.5. The closed form approximation is obtained by propagating Δr_{ai} through one iteration of a special version of the equations for line fitting in polar coordinates (2.20)–(2.22).

When estimating the systematic error in the line parameter estimates due to error changing with incidence angle, first the bearings of the line of interest are normalized as in section 2.3.4:

$$\phi_i = \phi_{im} - \alpha' + \frac{\pi}{2} \tag{2.51}$$

to obtain a line parallel with the x axis. In (2.51) α' is the true angle of the estimated line

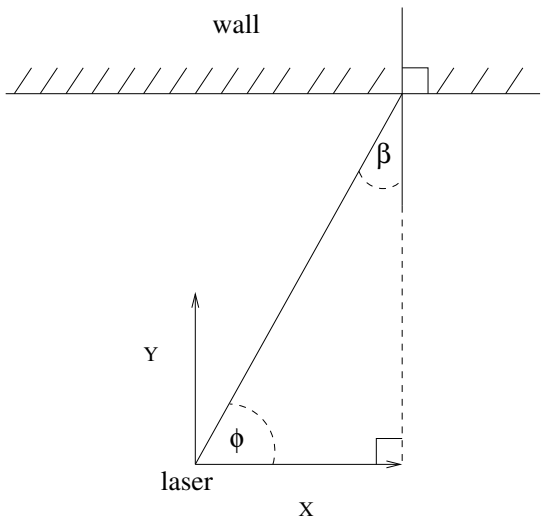


Figure 2.6: The relation of ϕ and β for horizontal lines.

and ϕ_{im} are bearings of the laser range measurements. Note that in practice the estimated line parameters are used instead of the true line parameters (α', d'). In the following, the notation α_0, d_0 will be used instead of α', d'.

The error changing with incidence angle is calculated for a horizontal line as:

$$\Delta r_{ai} = w|\tan\beta_i| = w|\tan\left(\frac{\pi}{2} - \phi_i\right)| = w|\cot\phi_i| = ws_i\cot\phi_i \tag{2.52}$$

Where $s_i = sign(\cot\phi_i)$, and the relation $\beta_i = \pi/2 - \phi_i$ (see fig. 2.6) for horizontal lines is also exploited. The absolute value operator $|\ |$ has been exchanged for multiplication with the sign s_i to ease integration in the closed form calculations described next.

The approach described in the previous section is not suitable for the derivation of a closed form approximation of the systematic line parameter error estimation due to the complex form of Δr_{ai}. Instead the simplified equations for line fitting in polar coordinates (2.20)–(2.22) are used. It is advantageous to use (2.20)–(2.22) because in each iteration they explain the range residuals between model and measurement with corrections $(\Delta\alpha, \Delta d)$ of the line parameter estimate. What equations (2.20)–(2.22) perform is need here, since we are interested in what angle and distance errors are caused by the increments Δr_{ai} of the true ranges. Equations (2.20)–(2.22) are used here in a non-vector form simplified for horizontal lines.

To derive the non-vector form of horizontal line angle and distance correction equations, first the equation of a line in polar coordinates (2.10) is linearized around the coordinates of the normalized line $(\pi/2, d_0)$. After substituting $\alpha_0 = \pi/2$ into (2.16) one gets:

$$\Delta r_i \approx \frac{\Delta d}{\sin\phi_i} + \frac{d_0\cos\phi_i}{\sin^2\phi_i}\Delta\alpha = a_i\Delta d + b_i\Delta\alpha \tag{2.53}$$

where $a_i = \frac{1}{\sin\phi_i}$ and $b_i = \frac{d_0 \cos\phi_i}{\sin^2\phi_i}$.
$(\Delta\alpha, \Delta d)$ minimizing the sum: $\sum(\Delta r_{ai} - \Delta r_i)^2$, can be calculated the following way (see appendix A.4 for derivation):

$$\Delta\alpha = \frac{\sum \Delta r_{ai} a_i \sum a_i b_i - \sum \Delta r_{ai} b_i \sum a_i^2}{(\sum b_i a_i)^2 - \sum b_i^2 \sum a_i^2} \tag{2.54}$$

$$\Delta d = \frac{\sum \Delta r_{ai} b_i \sum a_i b_i - \sum \Delta r_{ai} a_i \sum b_i^2}{(\sum b_i a_i)^2 - \sum b_i^2 \sum a_i^2} \tag{2.55}$$

Closed form approximations for the line parameter errors can be obtained by substituting (2.52) into (2.54)–(2.55) and by replacing sums with integrals as (2.40) in the previous section. The necessary derivations are shown in section A.5 of the appendix. The resulting closed form approximation for the line parameter error generated by range error changing with the incidence angle:

$$\Delta\alpha = \frac{ws}{d} \frac{EA - FC}{A^2 - \frac{1}{3}(\cot^3\phi_1 - \cot^3\phi_n)C} \tag{2.56}$$

$$\Delta d = ws \frac{FA - ED}{A^2 - \frac{1}{3}(\cot^3\phi_1 - \cot^3\phi_n)C} \tag{2.57}$$

where

$$A = \frac{1}{2}\left(\frac{1}{\sin^2\phi_1} - \frac{1}{\sin^2\phi_n}\right) \tag{2.58}$$

$$C = \cot\phi_1 - \cot\phi_n \tag{2.59}$$

$$D = \frac{1}{3}\left(\cot^3\phi_1 - \cot^3\phi_n\right) \tag{2.60}$$

$$E = -(t+1) + \frac{1}{\sin\phi_1} + \frac{t}{\sin\phi_n} \tag{2.61}$$

$$F = \frac{\cos\phi_1}{2\sin^2\phi_1} + t\frac{\cos\phi_n}{2\sin^2\phi_n} + \frac{1}{2}\ln|\tan\frac{\phi_1}{2}| + \frac{t}{2}\ln|\tan\frac{\phi_n}{2}| \tag{2.62}$$

and s and t are calculated according to:

- $s = 1,\ t = 1$ for $\phi_1 \leq \pi/2 \leq \phi_n$.
- $s = 1,\ t = -1$ for $\phi_1, \phi_n \leq \pi/2$.
- $s = -1,\ t = -1$ for $\phi_1, \phi_n \geq \pi/2$.

2.3.6 Error Due to Bias Growing with Distance

In this section the effect of laser range bias which grows linearly with distance is investigated on the systematic line parameter error. It is assumed that the range error is in the form of

$\Delta r_{ri} = k(r_i - r_{min})$ (see section 2.2), where k is a parameter describing the rate of the error growth, r_i is a range measurement corresponding to the measured line segment and r_{min} is the shortest range measurement of the line segment. Parameter k is determined experimentally. There can be laser scanners for example the Sick PLS, which have a large k, however the resulting range error does not grow too large because an internal compensation mechanism compensates errors in steps. Therefore the error of these laser scanners grows in a sawtooth pattern. In this analysis it is assumed that error grows linearly for the whole range of range readings of the measured line segment. For line segments spanning a large interval of ranges, the error grows too large without error compensation. To keep the systematic error estimate realistic for such line segments k is reduced. Not modeling the effect of error compensation for the Sick PLS is necessary since calibration parameters and raw time-of-flight measurements are not accessible.

In summary, in a simplistic approximation of the error caused by this bias, it is assumed, that the range error which is added to the true ranges grows linearly from 0 for the shortest range, up to the maximum value (but no more than max_err) for the longest range. The rate of growth is either v_r or a smaller number which results in a maximum error of max_err:

$$\begin{aligned} r_{max} &= \max(r_i),\ i = 1..n \\ r_{min} &= \min(r_i),\ i = 1..n \\ \Delta r &= \min((r_{max} - r_{min})v_r, max\_err) \\ \Delta r_{ri} &= (r_i - r_{min})\frac{\Delta r}{r_{max} - r_{min}} = (r_i - r_{min})k \end{aligned} \tag{2.63}$$

The approach used in the derivation of a closed form approximation of the systematic line parameter error is the same as in the case of error changing with incidence angle in section 2.3.5. Therefore the derivation details are described in section A.6 of the appendix.

The resulting closed form solutions are:

$$\Delta\alpha = \frac{k}{\sin\phi_{min}} \frac{BC - A\ln\left|\frac{\tan\frac{\phi_n}{2}}{\tan\frac{\phi_1}{2}}\right|}{A^2 - DC} \tag{2.64}$$

$$\Delta d = d_0 k \frac{A\left(A - \frac{B}{\sin\phi_{min}}\right) - \left[C - \frac{1}{\sin\phi_{min}}\ln\left|\frac{\tan\frac{\phi_n}{2}}{\tan\frac{\phi_1}{2}}\right|\right]D}{A^2 - DC} \tag{2.65}$$

where ϕ_{min} is the bearing corresponding to r_{min} and

$$A = \frac{1}{2\sin^2\phi_1} - \frac{1}{2\sin^2\phi_n} \tag{2.66}$$

$$B = \frac{1}{\sin\phi_1} - \frac{1}{\sin\phi_n} \tag{2.67}$$

$$C = \cot\phi_1 - \cot\phi_n \tag{2.68}$$

$$D = \frac{1}{3}\left(\cot^3\phi_1 - \cot^3\phi_n\right) \tag{2.69}$$

The application of these results is described in section 2.3.10.

2.3.7 Error Due to Quantization Bias

As it is shown in section 2.2 using simulations, the truncating quantizer introduces a bias which can be described by:

$$r_{qb} = b\sin\left((r' - Q(r'))\frac{2\pi}{q_r}\right) - \frac{q_r}{2} \tag{2.70}$$

This quantization bias superimposed on the true range can cause a systematic error in the estimated line parameters. Since the true range r' is unknown, the precise calculation of the angle and distance error of a line is probably not possible. However, it is possible to find an approximate for the maximum error. Let us assume the following:

- The $\frac{q_r}{2}$ error term in (2.70) is taken care of by the manufacturer, or its contribution to the overall error is modeled as constant bias (see section 2.3.4).
- The line parameters are reasonably accurate, therefore the estimate of r' differs from r' only by a constant, which causes a phase shift in the range bias.
- The line in question is normalized which means the line angle is set to $\pi/2$ and bearings ϕ_i are shifted as described in section 2.3.4.

Then the maximum error can be approximated by varying the superimposed sinusoid's phase and finding the maximum error. Note that by varying the phase of the sinusoid, the nonlinearity in (2.70) represented by the quantizer $Q(r')$ does not preclude the deployment of standard analysis tools to find the maximum error.

Let us rewrite (2.70) so that it contains a phase shift ε:

$$r_{qb} = b\sin\left((r' - Q(r'))\frac{2\pi}{q_r}\right) \approx b\sin\left((r - Q(r))\frac{2\pi}{q_r} + \varepsilon\right) = b\sin(\delta + \varepsilon) \tag{2.71}$$

where r is calculated using the estimated line parameters.

Since the maximum line angle error and the maximum distance error can occur at different phase shifts, these maximums are calculated separately.

To find the error in orientation caused by the quantization bias, r_{qb} of (2.71) is substituted into ξ_{mi} of (A.29):

$$\Delta\alpha = \frac{\sum\xi_{mi}a_i\sum a_ib_i - \sum\xi_{mi}b_i\sum a_i^2}{\left(\sum b_ia_i\right)^2 - \sum b_i^2\sum a_i^2} \tag{2.72}$$

$$= \frac{1}{M}\left(\sum r_{qbi}a_i \sum a_i b_i - \sum r_{qbi}b_i \sum a_i^2\right) \tag{2.73}$$

$$= \frac{1}{M}\left(\sum r_{qbi}a_i C_1 - \sum r_{qbi}b_i C_2\right) \tag{2.74}$$

$$= \frac{b}{M}\left(C_1 \sum \sin(\delta_i+\varepsilon)a_i - C_2 \sum \sin(\delta_i+\varepsilon)b_i\right) \tag{2.75}$$

where

$$C_1 = \sum a_i b_i,\ \ C_2 = \sum a_i^2,\ \ \delta_i = (r_i - Q(r_i))\frac{2\pi}{q_r} \tag{2.76}$$

$$M = \left(\sum b_i a_i\right)^2 - \sum b_i^2 \sum a_i^2,\ \ r_{qbi} = b\sin(\delta_i+\varepsilon), \tag{2.77}$$

To find out where $\Delta\alpha$ peaks, $\Delta\alpha$ is differentiated by ε and set to zero:

$$\frac{\partial \Delta\alpha}{\partial \varepsilon} = \frac{b}{M}\left(C_1 \sum \cos(\delta_i+\varepsilon)a_i - C_2 \sum \cos(\delta_i+\varepsilon)b_i\right) = 0 \tag{2.78}$$

Under the conditions that $b \neq 0$ and $M \nrightarrow \infty$, (2.78) is equivalent to

$$\begin{aligned} 0 &= C_1 \sum \cos(\delta_i+\varepsilon)a_i - C_2 \sum \cos(\delta_i+\varepsilon)b_i \\ &= \left(C_1 \sum a_i \cos\delta_i - C_2 \sum b_i \cos\delta_i\right)\cos\varepsilon - \left(C_1 \sum a_i \sin\delta_i - C_2 \sum b_i \sin\delta_i\right)\sin\varepsilon \end{aligned} \tag{2.79}$$

From (2.80), ε corresponding to an extreme of $\Delta\alpha$ can be found as:

$$\varepsilon_{m\alpha} = \arctan\frac{C_1 \sum a_i \cos\delta_i - C_2 \sum b_i \cos\delta_i}{C_1 \sum a_i \sin\delta_i - C_2 \sum b_i \sin\delta_i} \tag{2.80}$$

The maximum line orientation error $\Delta\alpha$ is then calculated by substituting $\varepsilon = \varepsilon_{m\alpha}$ back into (2.75).

To find an extreme of Δd, the above mentioned process is repeated, with the difference that (A.30) has to be differentiated with respect to ε.

Δd is expressed as:

$$\Delta d = \frac{\sum r_{qbi}b_i \sum a_i b_i - \sum r_{qbi}a_i \sum b_i^2}{\left(\sum b_i a_i\right)^2 - \sum b_i^2 \sum a_i^2} = \frac{\sum r_{qbi}b_i C_1 - \sum r_{qbi}a_i C_3}{M} \tag{2.81}$$

To make the differentiation of Δd easier, r_{qbi} is first differentiated:

$$\frac{\partial r_{qbi}}{\partial \varepsilon} = b\cos(\delta_i+\varepsilon) = b\left(\cos\delta_i\cos\varepsilon - \sin\delta_i\sin\varepsilon\right) \tag{2.82}$$

The differential of Δd is the combination of (2.81) and (2.82):

$$\frac{\partial \Delta d}{\partial \varepsilon} = \frac{b}{M} C_1 \sum (\cos\delta_i \cos\varepsilon - \sin\delta_i \sin\varepsilon)\, b_i - \frac{b}{M} C_3 \sum (\cos\delta_i \cos\varepsilon - \sin\delta_i \sin\varepsilon)\, a_i \quad (2.83)$$

The maximum distance error can be obtained by solving $\frac{\partial \Delta d}{\partial \varepsilon} = 0$, which is equivalent to:

$$\left(C_1 \sum b_i \cos\delta_i - C_3 \sum a_i \cos\delta_i\right) \cos\varepsilon - \left(C_1 \sum b_i \sin\delta_i - C_3 \sum a_i \sin\delta_i\right) \sin\varepsilon = 0 \quad (2.84)$$

From where ε generating the maximal distance error:

$$\varepsilon_{md} = \arctan \frac{C_1 \sum b_i \cos\delta_i - C_3 \sum a_i \cos\delta_i}{C_1 \sum b_i \sin\delta_i - C_3 \sum a_i \sin\delta_i} \quad (2.85)$$

where

$$C_3 = \sum b_i^2. \quad (2.86)$$

The maximum line distance error Δd is then calculated by substituting $\varepsilon = \varepsilon_{md}$ back into (2.81).

Note that the approximations presented here for systematic line segment parameter errors are not closed form in the sense of the definition of closed form in the introduction (section 2.1) since they contain sums over all laser measurements of the given line segment. These sums, unlike in previous sections are not approximated by integrals since δ_i in these sums contain quantized ranges $Q(r_i)$ which make integration difficult.

2.3.8 Error Due to Laser Plane Misalignment

When mounted on a robot, not just the laser scanner can be a source of errors, but a tilted laser plane, time registration error or imprecise knowledge of the laser's pose in the robot coordinate system can all contribute to line segment parameter errors. In this section only errors due to laser plane misalignment are discussed.

One of our robots from the lab is equipped with two pneumatic tyres and two caster wheels, one of which is suspended on springs. The laser plane orientation depends on the tyre pressures and on the orientation of the caster wheels. It was found that the laser plane pitch angle changes by 0.4° just by moving the caster wheels around. Once the roll angle was measured to be 1°, due to unequal tyre pressures.

In the analysis of the error due to laser plane misalignment, homogeneous transformations (see for example [Paul, 1981]) are used in 3D space. Let us place a world coordinate system (denoted with subscript "w") into the center of the odometry of the robot with the (X,Y) plane parallel with the floor, and Y axis pointing toward the front of the robot. Let us place a second, robot coordinate system denoted with subscript "r", into the center of odometry. This new coordinate system is obtained by rotating the world coordinate system around its X axis by the

pitch angle β, and then rotating the result around its new Y axis by the roll angle γ. Let us place a new coordinate system for the laser, denoted by "l", which has the same orientation as the robot coordinate system but it is displaced by the coordinates of the laser $(x_l, y_l, z_l, 1)^T$. Then to find out how a point measured in the misaligned laser coordinate system would appear in a well aligned laser coordinate system, one has to multiply the homogeneous coordinates of the measured point with a transformation matrix which is the product of 4 homogeneous transform matrices:

$$\begin{aligned}
\mathbf{v}_{wl} &= \mathbf{T}^{-1}\mathbf{R}_x(\beta)\mathbf{R}_y(\gamma)\mathbf{T}\mathbf{v}_{rl} = \\
&= \begin{bmatrix} 1 & 0 & 0 & x_l \\ 0 & 1 & 0 & y_l \\ 0 & 0 & 1 & z_l \\ 0 & 0 & 0 & 1 \end{bmatrix}^{-1} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & cos\beta & -sin\beta & 0 \\ 0 & sin\beta & cos\beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \\
&\quad \cdot \begin{bmatrix} cos\gamma & 0 & -sin\gamma & 0 \\ 0 & 1 & 0 & 0 \\ sin\gamma & 0 & cos\gamma & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & x_l \\ 0 & 1 & 0 & y_l \\ 0 & 0 & 1 & z_l \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_r \\ y_r \\ z_r \\ 1 \end{bmatrix}
\end{aligned}$$

Now let us assume that the (X,Y) axes of the laser are aligned with the (X,Y) axes of the robot coordinate system. Note, that in this configuration the laser is pointing forward. Let us assume that all walls are perpendicular to the floor. Then an equation of a plane representing a wall is given by the equation:

$$\omega_{rl}^T \mathbf{v}_{rl} = \begin{bmatrix} \cos\alpha & \sin\alpha & 0 & -d \end{bmatrix} \mathbf{v}_{rl} = 0 \tag{2.87}$$

where ω_{rl}^T represents the plane in the laser's frame and $\mathbf{v}_{rl}$ represents a point in the plane ω_{rl}^T expressed also in the laser's frame.

To find out the difference in the perception of a plane in case of a nonzero roll/pitch angle, the plane parameters (see [Paul, 1981]) expressed in the laser coordinate system on the robot are transformed to the laser coordinate system in the world coordinate system, which represents the nonideal case with nonzero roll and pitch angles:

$$\omega_{wl} = (\mathbf{T}^{-1}\mathbf{R}_x(\beta)\mathbf{R}_y(\gamma)\mathbf{T})^T \omega_{rl} \tag{2.88}$$

To extract the line from the result, the intersection of ω_{wl} with the (X,Y) plane is calculated, after which the equation for the line is obtained as:

$$x\cos\gamma\cos\alpha + y(\cos\beta\sin\alpha - \sin\gamma\sin\beta\cos\alpha) =$$

$$= d - (x_l \cos\gamma - y_l \sin\gamma \sin\beta - z_l \sin\gamma \cos\beta - x_l)\cos\alpha - \\ - (y_l \cos\beta - z_l \sin\beta - y_l)\sin\alpha \tag{2.89}$$

The representation of a line in (2.89) is not in normal form, because the coefficients at x and y are not normalized. Let us denote the coefficient at x as c_x the coefficient at y as c_y and the right side as c_d. Then the line in slope-intercept form:

$$y = -\frac{c_x}{c_y}x + \frac{c_d}{c_y} = kx + q \tag{2.90}$$

Finally (k,q) can be converted into (α', d') using (2.14)–(2.15). The difference between α' and α and between d' and d gives the error in the angle and distance parameters of a line when the laser plane is not parallel with the floor.

After experimenting with parameters of lines, roll, pitch angles and laser positions, it was observed that the effect of laser plane misalignment on the line parameters is insignificant if the roll and pitch angles are reasonably small, and the laser is placed close to the center of odometry. To show an example, let us assume that the laser is placed at $(0, 30cm)$ in the robots coordinate system, and that the wall to be measured can be described by $\alpha = 90°$ and $d = 200cm$. This configuration would lead to the following errors:

- $\beta = \gamma = 1° \Longrightarrow \Delta\alpha = -0.0175°$, $\Delta d = 0.0194cm$.
- $\beta = \gamma = 3° \Longrightarrow \Delta\alpha = -0.1572°$, $\Delta d = 0.1745cm$.
- $\beta = \gamma = 5° \Longrightarrow \Delta\alpha = -0.4369°$, $\Delta d = 0.4827cm$.
- $\beta = \gamma = 10° \Longrightarrow \Delta\alpha = -1.7538°$, $\Delta d = 1.8924cm$.

2.3.9 Errors Due to Motion

If a laser scanner with a rotating mirror is mounted on a robot, then there are 2 major error sources resulting from motion:

- Finite time for a mirror to rotate through a scan.
- Uncertainty in the knowledge of the time instant when the laser scan was taken.

There also can be another source of error. Some laser scanners take more scans at different times and interlace them to increase the angular resolution of the scans. In this chapter errors caused by interlaced scans will be disregarded.

Let us deal with the second error first which is due to the time registration error. In the search for a solution, the following simplifying assumptions are introduced:

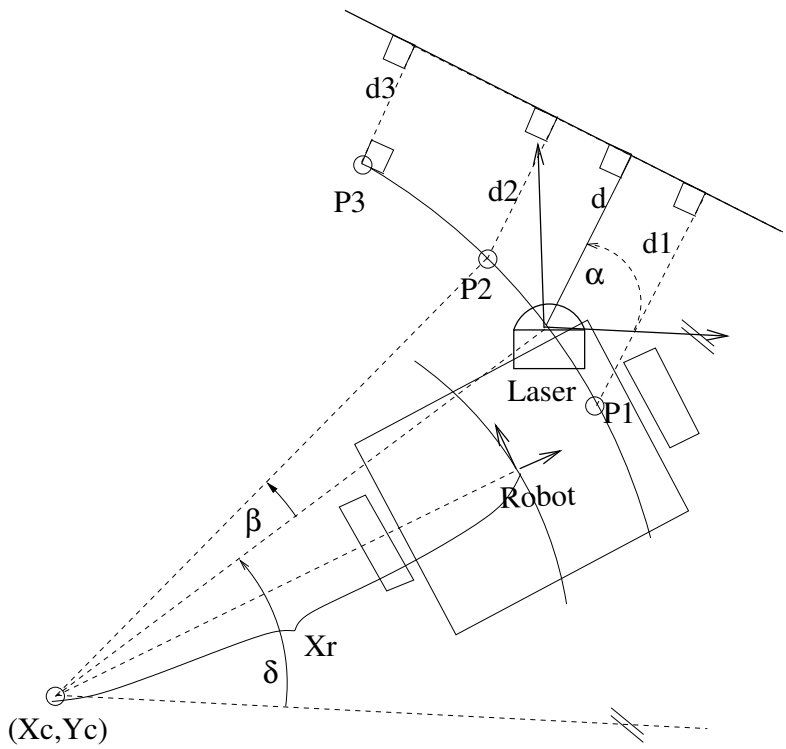


Figure 2.7: Illustration of line orientation and distance error due to time registration error.

- The time registration error is in the interval $T_e \in \langle -T, T \rangle$. T is different for different laser range finders and for different settings. The delay of a Sick PLS in 361 point measurement mode at 56Kbaud communication speed while using Linux was measured to be around 70ms.

- The differential drive robot carrying the laser range finger is moving with constant left v_l and right v_r speed, while the wheel separation D is also constant. Note that such motion results in a circular path of which a linear path is a special case.

- The measurement of the line takes place instantly and it is accurate.

- The true line parameters (α', d') at time 0 are expressed in the laser's coordinate system.

The problem of finding the maximum error $(\Delta\alpha, \Delta d)$ in the line parameters can be transformed into the geometrical problem of finding the distances of points lying on a circle from a line. The assumption of the robot moving on a circle implies that the laser scanner moves on a concentric circle (see fig. 2.7). Then the maximum line orientation error equals the orientation change of the robot, and the worst case distance error equals the distance of the furthest point of an arc on which the laser scanner is moving.

If the left and right wheels of the robot are moving with linear velocities of v_l and v_r, then the robot will be rotating around $(x_r, 0)$ expressed in the robot's reference frame:

$$x_r = \frac{D}{2}\frac{v_l + v_r}{v_l - v_r} \tag{2.91}$$

Note that if $v_l - v_r$ is small, meaning that the robot is on a near linear path, then x_r can be set to a large number to avoid division by 0. During time uncertainty T the robot and therefore the laser can change its orientation by:

$$\beta = \frac{(v_r - v_l)}{D}T \tag{2.92}$$

Since the line segment is expressed in the laser's coordinate frame let us convert the point of rotation $(x_r, 0)$ from the robot's frame into the laser's frame:

$$x_c = x_r\cos(\gamma_l) - y_l\sin(\gamma_l) - x_l\cos(\gamma_l) \tag{2.93}$$

$$y_c = -x_r\sin(\gamma_l) - y_l\cos(\gamma_l) + x_l\sin(\gamma_l) \tag{2.94}$$

where x_l, y_l, γ_l express the laser's position and orientation in the robot frame. The radius of the circle on which the laser scanner moves is then:

$$R = \sqrt{x_c^2 + y_c^2} \tag{2.95}$$

The laser's current position on the circle can be described by bearing δ when viewed from the center of the circle. δ is calculated using the four quadrant version of $\tan^{-1}$ as:

$$\delta = atan2(-y_c, -x_c) \tag{2.96}$$

when measured from an axis parallel to the X axis of the laser. From fig. 2.7 one can see that the maximum error in the angle of the line equals the maximum orientation change of the robot i.e. $\Delta\alpha = \beta$ since the laser range finder is rigidly mounted on the robot. Also from fig. 2.7 the maximum distance error is either at point P_1 corresponding to angle $\delta - \beta$, at P_2 corresponding to $\delta + \beta$ or at P_3 corresponding to the closest or furthest point of the circle to the line at angle α'. P_3 has to be considered only if it is on the shorter arc between P_1 and P_2, assuming small time registration error T with respect to the angular velocity of the robot. The error in distance can be:

$$\Delta d_1 = d' - (R\cos(\delta - \beta - \alpha') + x_c\cos(\alpha') + y_c\sin(\alpha') \tag{2.97}$$

$$\Delta d_2 = d' - (R\cos(\delta + \beta - \alpha') + x_c\cos(\alpha') + y_c\sin(\alpha') \tag{2.98}$$

$$\Delta d_3 = d' - (R\cos(\alpha' - \alpha') + x_c\cos(\alpha') + y_c\sin(\alpha')$$

$$= \quad d' - (R + x_c \cos(\alpha') + y_c \sin(\alpha')) \tag{2.99}$$

The distance errors described previously were derived by calculating the distance of the moving laser:

$$x = R\cos(\delta + \beta) + x_c \tag{2.100}$$

$$y = R\sin(\delta + \beta) + y_c \tag{2.101}$$

from the true line (α', d'):

$$\begin{aligned} \Delta d &= d' - (x\cos(\alpha') + y\sin(\alpha') \\ &= d' - ((R\cos(\delta+\beta) + x_c)\cos(\alpha') + (R\sin(\delta+\beta) + y_c)\sin(\alpha')) \\ &= d' - (R\cos(\delta+\beta-\alpha') + x_c\cos(\alpha') + y_c\sin(\alpha')) \end{aligned} \tag{2.102}$$

Once more, the estimated systematic error in line parameter due to time registration error on a moving robot can be calculated as:

$$\Delta\alpha = |\beta| \tag{2.103}$$

$$\Delta d = \max(|\Delta d_i|) \tag{2.104}$$

where $i = \{1,2,3\}$ if P_3 is on the shorter arc between P_1 and P_2, otherwise $i = \{1,2\}$.

As mentioned earlier beside the time registration error, there also can be an error due to nonzero time necessary for taking a scan. In this case given the true line parameters α', d', rotational speed of the mirror ω, and time between two range samples ΔT, first range measurements are calculated, then using the range measurement deviations from the nonmoving case, line parameters errors are estimated.

Similarly as in the case of the time registration error, the line parameters in the moving laser frame when measuring range i are:

$$\alpha_i = \alpha' - \beta_i \tag{2.105}$$

$$d_i = d' - (R\cos(\delta + \beta_i - \alpha') + x_c\cos(\alpha') + y_c\sin(\alpha')) \tag{2.106}$$

where $\beta_i = i\omega\Delta T$. Then the i-th range measurement is:

$$r_i = \frac{d_i}{\cos(\alpha_i - \phi_i)} = \frac{d' - (R\cos(\delta + \beta_i - \alpha') + x_c\cos(\alpha') + y_c\sin(\alpha'))}{\cos(\alpha' - \beta_i - \phi_i)} \tag{2.107}$$

Substituting r_i in the place of r_{mi} in (2.22) gives an estimate of line parameter errors $\Delta\alpha, \Delta d$. The investigation of the possibility of a closed form approximation is part of the future work.

2.3.10 Line Error Estimation Summary

In this section a concise summary of laser line error estimation of sections 2.3.3–2.3.9 is given together with a discussion of applications.

To make the error analysis tractable it was assumed that errors in the range and in the resulting line parameters are so small, that the principle of superposition is approximately true. In sections 2.3.3–2.3.9 line parameter errors resulting from the following error sources were discussed:

1. **Random errors in the range measurements.** Random errors in the range measurements are modeled as zero mean white Gaussian noise with σ_r standard deviation. The resulting error in the line parameters is estimated using the equation (2.25) for the estimation of the covariance matrix of linear regression results.

2. **Identical bias in the range measurements.** Assuming a constant systematic error in the range measurements, a closed form approximation describing systematic line parameter errors (2.47)–(2.48) has been derived for horizontal lines. Non-horizontal lines need to be normalized i.e. shifted left or right in a polar coordinate system using (2.26) so that range measurements represent a horizontal line.

3. **Range bias changing with incidence angle.** A closed form approximation (2.56)–(2.57) of systematic line parameter errors due to range errors caused by changing incidence angle of laser beam and target surface normal have been derived for horizontal lines. The range error change with the angle between laser beam and target surface normal has been modeled with a tangential relationship (2.52). The results derived for such range error model may have a limited application to Sick PLS laser range finders.

4. **Range bias growing with distance.** Closed form approximations (2.64), (2.65) of systematic line parameter errors have been derived for range errors modeled to grow linearly with range starting from the shortest range reading of the line segment. This approximation has been derived for horizontal lines.

5. **Range bias due to quantization.** The range bias due to quantization has been modeled with a sinusoid. Maximum line orientation error for normalized lines is found by finding that phase $\varepsilon_{m\alpha}$ (2.80) of the sinusoid which generates the maximum orientation error. $\varepsilon_{m\alpha}$ is then used in the equation for orientation error calculation (2.75) to obtain the maximum line orientation error. Similarly when estimating the maximum distance error, the phase of the sinusoid ε_{md} yielding the maximum distance error is calculated from (2.85) and the substituted in the equation describing the distance error (2.81).

6. **Laser plane misalignment.** In the analysis of the effects of nonzero pitch and roll angles of the laser scanner on line parameter errors it has been found, that errors caused by small roll and pitch angles the errors can be neglected.

7. **Finite mirror rotation rate.** On a moving robot finite mirror rotation speed of the laser scanner results in range measurements of a scan taken from a different locations. Line parameter errors are estimated by first calculating ranges (2.107) which are measured from a moving robot, then substituting them into the equation (2.22) which gives line angle and distance errors.

8. **Time registration error.** Not knowing exactly the time when a laser scan was taken on a moving robot also results in systematic line parameter errors. These errors have been approximated while assuming that the robot moves on an arc in the time interval defined by the maximum time registration error by finding the maximum line distance (2.104) and orientation (2.103) error.

There are 7 systematic $(\Delta\alpha_i, \Delta d_i)$ and one random line error components. The systematic errors can be converted into an upper bound estimate by summing up the absolute values of the individual systematic error components:

$$\Delta\alpha_s = \sum_{i=1}^{7} |\Delta\alpha_i| \tag{2.108}$$

$$\Delta d_s = \sum_{i=1}^{7} |\Delta d_i| \,. \tag{2.109}$$

How the random error estimate and the upper systematic error bound are used depends on the application. When performing localization and mapping using a set membership approach, error bounds of measurements are necessary. Therefore for such cases the sum of absolute errors increased by the standard deviations of the random error estimate can be used. In the next chapter where the fusion of laser and advanced sonar measurements for simultaneous localization and mapping using a Kalman filter framework is described, covariance estimates of line parameter errors are necessary. In the advanced sonar and laser fusion case, the squared upper systematic error bounds are added to the diagonal elements of the random error covariance matrix. This way the systematic line angle and distance errors are modeled as independent random errors with standard deviations equal to the upper systematic error bounds.

When modeling systematic errors as random errors in localization and mapping in a Kalman filter context, care has to be taken not to use measurements taken of the same line from the same position multiple times. If a line feature is observed multiple times from a stationary robot, then the constant systematic error components cause the errors to be correlated which contradicts the white Gaussian measurement noise assumption of Kalman filters. Since subse-

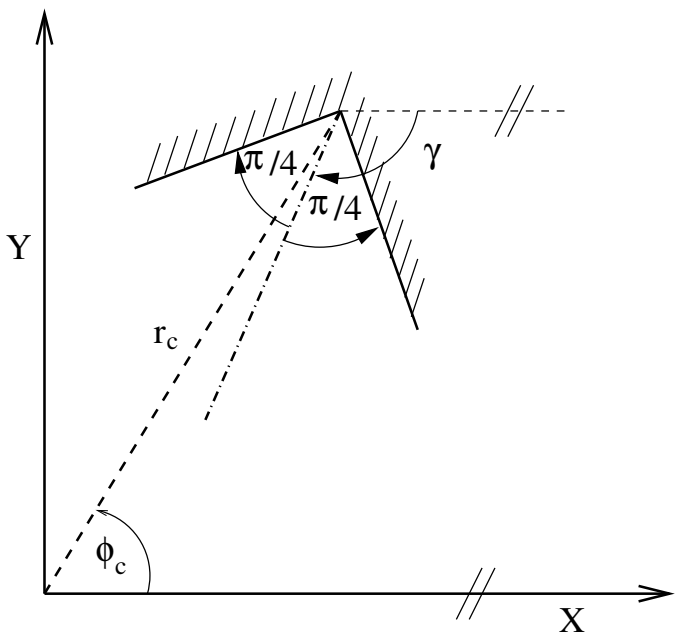


Figure 2.8: Meaning of corner parameters.

quent measurements from a stationary robot contain less information than modeled, the state error estimate of the Kalman filter becomes overoptimistic.

2.4 Right Angle Corner Fitting

The polar line fitting approach in this chapter can be extended to right angle corners. This extension is necessary, so that right angle corners measured by laser and advanced sonar can be fused in the next chapter. In [Gander and Hřebíček, 1993] a method for fitting orthogonal lines is described, where the association of points to lines is assumed to be known. A singular value decomposition is applied to minimize the sum of square distances of the points from the fitted straight line. Since the laser scanner operates fundamentally in polar coordinates by measuring ranges to objects at discrete bearings, the errors in Cartesian coordinates are correlated. Moreover, the covariance matrix of each point is different, which makes an accurate error estimation for the method described in [Gander and Hřebíček, 1993] complicated. To make the error estimation simpler and more accurately model the fundamental range error mechanism, a corner fitting approach is developed here which minimizes the sum of square range residuals.

As can be seen from fig. 2.8 a right angle corner is described by the range r_c and bearing ϕ_c of its center and the angle γ of its bisecting line with the X axis. Then the angles of the normals of the lines constituting the corner are $\gamma + \frac{\pi}{4}$ and $\gamma - \frac{\pi}{4}$. The lines themselves can be

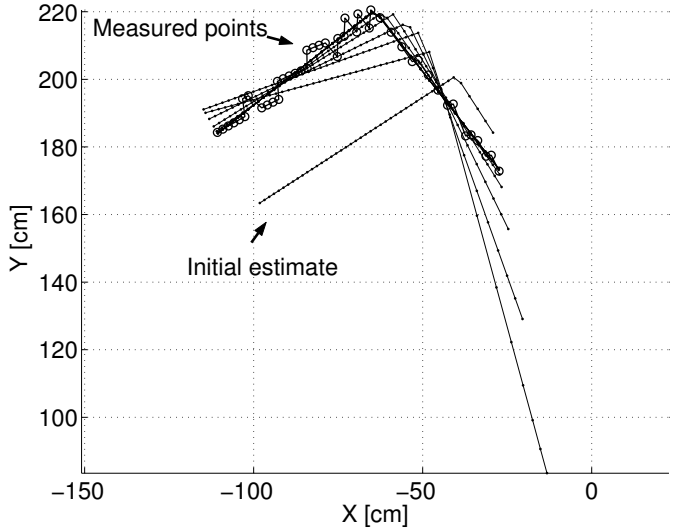


Figure 2.9: Corner estimation depicted in Cartesian coordinates.

described by the equation of a line in polar coordinates:

$$d = r\cos(\gamma \pm \frac{\pi}{4} - \phi) \tag{2.110}$$

where ϕ is the bearing of range reading r and d is the distance of the line from the origin. Since both lines go through (ϕ_c, r_c), (2.110) can be written as:

$$d = r_c \cos(\gamma + sign(\phi_i - \phi_c)\frac{\pi}{4} - \phi_c) \tag{2.111}$$

where to distinguish between the lines, an observation was used according to which measured laser bearings ϕ_i belonging to the first line are smaller than ϕ_c, and bearings belonging to the second line are larger than ϕ_c. Then measured ranges can be modeled as:

$$r_i = \frac{d}{\cos(\gamma + sign(\phi_i - \phi_c)\frac{\pi}{4} - \phi_i)} = \frac{r_c \cos(\gamma + sign(\phi_i - \phi_c)\frac{\pi}{4} - \phi_c)}{\cos(\gamma + sign(\phi_i - \phi_c)\frac{\pi}{4} - \phi_i)} \tag{2.112}$$

Using linear regression iteratively on the linearized version of (2.112):

$$r_i - r_{0i} \approx \frac{\partial r_i}{\partial \phi_c}\Delta\phi_c + \frac{\partial r_i}{\partial r_c}\Delta r_c + \frac{\partial r_i}{\partial \gamma}\Delta\gamma \tag{2.113}$$

where

$$\frac{\partial r_i}{\partial \phi_c} = \frac{r_{c0}\sin(\gamma_0 + sign(\phi_i - \phi_{c0})\frac{\pi}{4} - \phi_{c0})}{\cos(\gamma_0 + sign(\phi_i - \phi_{c0})\frac{\pi}{4} - \phi_i)} \tag{2.114}$$

$$\frac{\partial r_i}{\partial r_c} = 1 \tag{2.115}$$

$$\frac{\partial r_i}{\partial \gamma} = \frac{r_{c0}\sin(\phi_{c0}-\phi_i)}{\cos^2(\gamma_0 + sign(\phi_i-\phi_{c0})\frac{\pi}{4}-\phi_i)}, \tag{2.116}$$

can provide an estimate on the corner parameters. Equation (2.113) can be restated in vector form as

$$\Delta\mathbf{r} = \mathbf{r_m} - \mathbf{r_0} = \mathbf{H_0}\Delta\mathbf{b} + \mathbf{R} \tag{2.117}$$

Where

$$\mathbf{H_0} = \begin{bmatrix} \dots & \dots & \dots \\ \frac{\partial r_i}{\partial \phi_c} & \frac{\partial r_i}{\partial r_c} & \frac{\partial r_i}{\partial \gamma} \\ \dots & \dots & \dots \end{bmatrix} \tag{2.118}$$

$$\Delta\mathbf{b} = [\Delta\phi_c \;\; \Delta r_c \;\; \Delta\gamma]^T \tag{2.119}$$

$\mathbf{R}$ is a vector of measurement noise with a covariance matrix $\sigma_r^2 I$ and $\mathbf{r_m}$ is a vector containing measured ranges.

Using linear regression [Wetherill, 1986] iteratively on the linearized problem (2.113), one can find (ϕ_c, r_c, γ) which minimizes the sum of square range residuals $\sum(r_{mi}-r_{ji})^2$, the following way:

$$\begin{aligned} \mathbf{r_j} &= \begin{bmatrix} r_{j1} & \dots & r_{ji} & \dots & r_{jn} \end{bmatrix}^T \\ &= \begin{bmatrix} \dots & \dfrac{r_{cj}\cos(\gamma_j + sign(\phi_i-\phi_{cj})\frac{\pi}{4}-\phi_{cj})}{\cos(\gamma_j + sign(\phi_i-\phi_{cj})\frac{\pi}{4}-\phi_i)} & \dots \end{bmatrix}^T \\ \mathbf{H_j} &= \begin{bmatrix} \dots & \dots & \dots \\ \frac{\partial r_i}{\partial \phi_c} & 1 & \frac{\partial r_i}{\partial \gamma} \\ \dots & \dots & \dots \end{bmatrix} \qquad (2.120) \\ \Delta\mathbf{b} &= (\mathbf{H_j^T H_j})^{-1}\mathbf{H_j^T}(\mathbf{r_m} - \mathbf{r_j}) \qquad (2.121) \\ \begin{bmatrix} \phi_{cj+1} \\ r_{cj+1} \\ \gamma_{j+1} \end{bmatrix} &= \begin{bmatrix} \phi_{cj} \\ r_{cj} \\ \gamma_j \end{bmatrix} + \Delta\mathbf{b} \qquad (2.122) \end{aligned}$$

Equation (2.121) yields the least squares estimate, and can be found for example in [Kay, 1993]. The advantage of the above mentioned approach is that the corner point is implicitly chosen, and a simple covariance estimate is obtained due to the use of linear regression [Wetherill, 1986]:

$$cov(\Delta\mathbf{b}) = cov([\phi_c \; r_c \; \gamma]^T) = \sigma_r^2(\mathbf{H^T H})^{-1} \tag{2.123}$$

As discussed in the previous section, laser range finder output can be influenced by systematic

errors. To gain a worst case systematic error estimate, similarly to the previous section, the contribution from each range error source is calculated using (2.121) by replacing $(\mathbf{r_m} - \mathbf{r_j})$ with the estimated range systematic errors.

The disadvantage of this corner fitting approach is the slight possibility of divergence when the initial choice of parameters has a large error. To improve robustness, in each iterations those r_{ji} are removed from $\mathbf{r}_j$ which are either negative or they are larger than twice of the measured range r_{mi}. A practical way to obtain the initial parameters is to choose the point furthest from the first and last point of the corner in the scan as the corner point, and choose the bifurcating angle of the (first point, corner point, last point) triangle as the orientation.

An example of the corner fitting algorithm is shown in fig. 2.9 where the convergence from a deliberately poor initial corner estimate is shown. In practice at most 10 iterations are necessary. The computational load is small compared to SLAM.

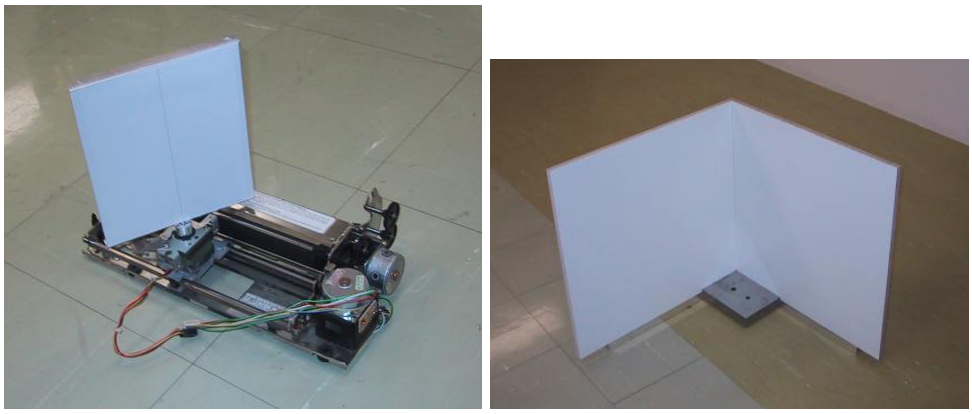
2.5 Experimental Work

The laser line error model and the line fitting approach are applied to two laser scanners: the Sick PLS [Erwin Sick GmbH, 1995] and LMS [SICK AG, 2002]. In this section, first the range error models are tuned to range measurements, then laser line random and systematic errors excluding errors due to motion and laser plane misalignment are tested. Errors due to motion are not tested due to difficulties associated with the accurate realization of such tests. Errors due to laser plane misalignment are not tested experimentally since their effects in typical indoor applications are negligible. In the tests two calibration tools were used that are described in the next section.

2.5.1 Laser Calibration Tools

For evaluation of the line error models and for testing of the Sick PLS and LMS, two tools were used. The first tool resembles a perfect right angle corner with 60cm long arms (see fig. 2.10b). The angle of the corner was estimated as $89.85° \pm 0.3°$ in the following way: A tape measure was used to measure the length of the arms and the hypotenuse. Then the cosine law was applied to calculate the opening angle of the corner. The error bounds were obtained by assuming $\pm$1mm error in the measurements. The surface of the corner has a rather shiny finish with visible specular reflections.

The inspiration for the second tool used came from [Ye and Borenstein, 2002], where the authors used a 4m linear motion table with a rotating target plane mounted on it. A similar, but inexpensive setup (see fig. 2.10a) was created, by recycling the head moving mechanism of a discarded printer. The printer head was replaced with a 16x16cm target plane rotated by a stepper motor from a 5.25” floppy drive. The target plane’s surface was covered with thick,



(a) (b)

Figure 2.10: Laser calibration tool made from old printer a), “perfect” corner b).

white, non-glossy paper. The mechanism was controlled by a PC running Linux. The achieved resolution in distance and angle was 0.3mm and 1.8°.

2.5.2 Specifications of the SICK PLS Laser Sensor

Probably two of the most common laser scanners mounted on mobile robots are the SICK PLS (see fig. 2.11) and the newer LMS. This section describes the PLS 101-112.

A rotating mirror inside the PLS sensor deflects an infrared laser beam in the range from 0° to 180° [Erwin Sick GmbH, 1995]. Distance is determined by measuring the time of flight of the emitted laser pulses. The resolution of the sensor is 5cm in the distance, and maximally 0.5° in the angles. The worst case error according to [Erwin Sick GmbH, 1995] is 94mm at the distance of 2m and 131mm at 4m. One 180° scan takes 40ms to complete. The maximum range of the sensor is 50m. Measurements from the sensor are transmitted to a PC through a serial line.

The patent [Wetteborn, 1993] of the Deutches Patent Amt describes the operation of a laser range finder, which is very similar to the PLS. Therefore it is assumed here, that the PLS and the patent [Wetteborn, 1993] are related. The laser range finder described in [Wetteborn, 1993] works the following way: A laser source sends out a pulse of 3.5ns duration and a counter of 330ps resolution is started. The 330ps time resolution results in 5cm distance resolution. The returned pulse is detected by a photo receiver. The output from the receiver is fed into a comparator. When the received signal is 7 times larger than the average noise level of the photo receiver, then the output of the comparator stops the counter. Comparing the output of the photo receiver with multiples of the average noise level measured on the photo receiver helps to keep false detection rate low.

Returning light pulses with different light intensities generate signals on the photo detector

Figure 2.11: SICK PLS101-112 laser range finder.

with different rise times. In an example from [Wetteborn, 1993], a change in the rise time can generate a 20cm error in the range measurement. To correct for this error, a peak detector consisting of ECL (Emitter Coupled Logic) comparators is employed to discriminate between 6 levels. The output of the peak detector is fed into a microprocessor, where the time of flight is compensated for by the rise time error. However as it will be shown later, the 6 level resolution results in easily detectable errors.

2.5.3 Specifications of the SICK LMS Laser Sensor

The Sick LMS [SICK AG, 2002] sensor is an improved version of the PLS. Its angular resolution has been reduced to 0.25° through interlacing 4 measurements taken at 1° resolution. Its distance resolution has been reduced to 1cm. The mirror speed has been increased to 75Hz, therefore it can take one 180° scan at 1° resolution each 13ms. According to its manual [SICK AG, 2002], the Sick LMS 200 has a systematic error of $\pm 15mm$ in mm mode and $\pm 4cm$ in cm mode and a typical random error standard deviation is 5mm in mm mode.

2.5.4 Warm-Up Experiments and Identical Bias

Similar experiments were conducted to the ones found in [Ye and Borenstein, 2002] with the PLS and LMS sensors. In the warm-up experiment with the PLS sensor a point on the wall was measured for over 4 hours. A 3cm drift was observed in the readings (see fig. 2.12). In order to make the change in the measured range clearly visible, the raw readings were filtered as follows: $output(i) = 0.995output(i+1) + 0.005input(i)$, which is equivalent to a first order linear filter with 40s time constant. However, to be able to initialize this filter correctly, the filter was applied in reverse time (i.e. last value first, first value last). This enabled the correct initialization of the filter by setting the initial value equal to the average of the last 100 measurements. The large number of samples used for initialization was necessary due to the large quantization step of 5cm of the PLS. Had the filter been applied in forward direction, it would not have been possible to initialize the filter correctly due to the quick change of measurements at the beginning.

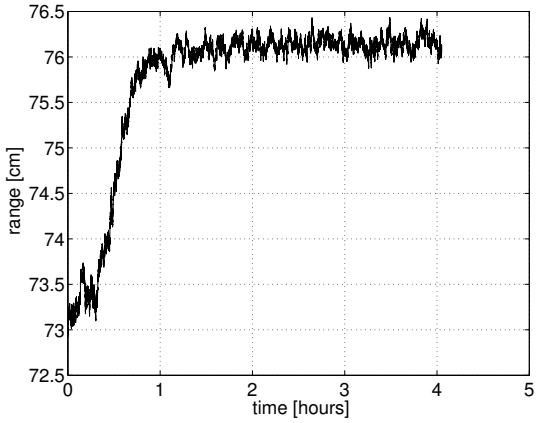


Figure 2.12: Change in filtered range readings during warm-up.

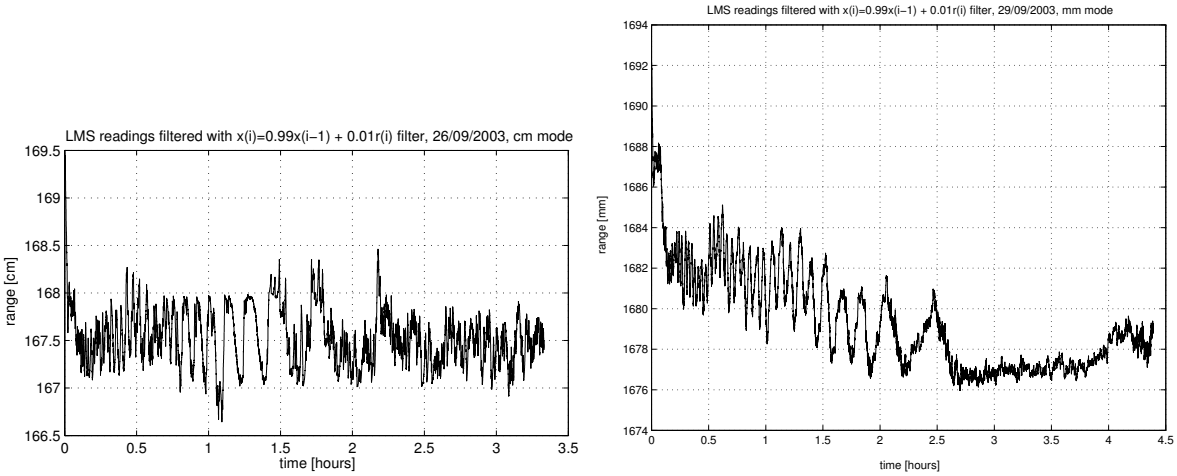


Figure 2.13: Change in filtered LMS range readings during 2 warm-up periods, where the left figure LMS is in cm mode, and the right is LMS in mm mode.

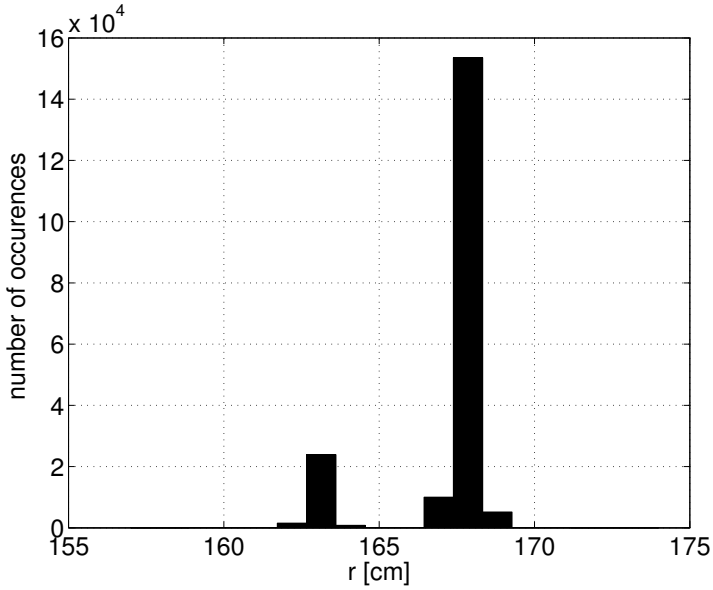


Figure 2.14: Distribution of PLS laser measurements of one point of a wall.

The filtered results of the LMS experiment are visible in fig. 2.13. The filter applied was $output(i) = 0.99output(i+1) + 0.01input(i)$. On the left the laser was in cm mode, where the range changes 2.5cm. On the right the laser was in mm measurement mode, and the range changed by 16mm.

Bias due to warm-up is not the only source of bias, there are other errors creating bias as well. For example in the output of the PLS from time to time, all range measurements shift randomly by $\pm$1cm for the duration of one scan. This is depicted on fig. 2.14 which shows a distribution of 200k samples taken of the same point. The small bars on both sides of the main peaks demonstrate this 1cm shift. Judging from the results of other experiments, it seems, that the 1cm shift occurs with roughly the same probability in both directions.

It is expected for the PLS, that surfaces with different reflectivity result in different range bias, since different reflectivity causes different amount of light to be reflected, therefore causing different times necessary for the output of the photo receiver to reach the threshold.

The LMS range reading accuracy depends on the surface properties as well. In [Ye and Borenstein, 2002] the LMS accuracy was tested object of different material and color. The bias depending on the surface properties was up to 1cm large. Since the LMS laser was not obtained until much later in this PhD, it was not tested for this effect.

2.5.5 Experiment with Calibration Target Moved Away from Laser

Using the laser calibration tool, the dependence of bias and standard deviation of the Sick PLS output on distance was measured. In the experiment the laser calibration tool's longitudinal

axis was aligned with a laser beam. The target plane was moved by 20cm further away from the laser in 2.5mm steps. After each step, the plane stayed in position for 10 minutes to enable the collection of 3000 sample points. The PLS was used in 361 point mode, and the on board averaging of measurements was turned off. Because the precise distance of the Sick PLS and the laser calibration tool was unknown, the change of bias was being measured instead of the absolute value.

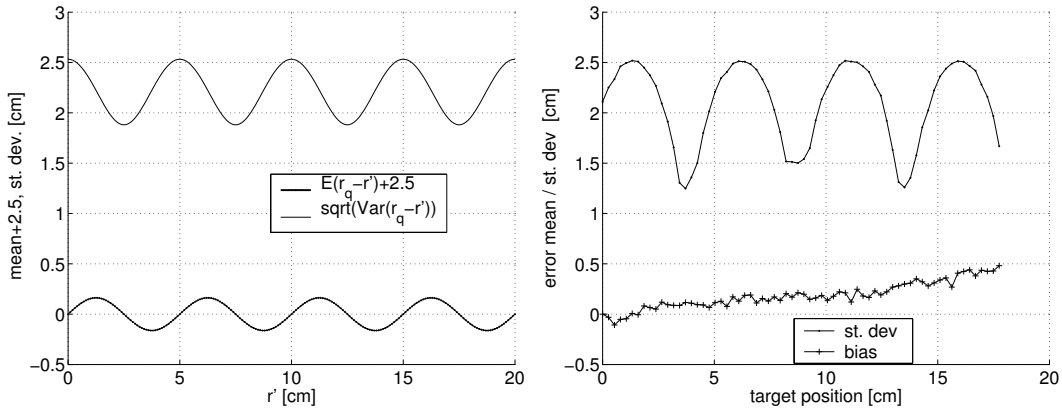
The result of the experiment is depicted in fig. 2.15b. The standard deviation is a periodic function with a period of the quantization step. In half of the period the standard deviation looks like a sinusoid, which peaks at 2.5cm. For the rest of the time it resembles a V shape. Figure 2.15a was generated by assuming white Gaussian noise of $\sigma_n = 1.7cm$ entering a truncating quantizer with 5cm quantization level. 1.7cm were chosen to generate modeled range noise standard deviation in fig. 2.15a close to the measured standard deviation. Note that the quantization bias parameter corresponding to $\sigma_n = 1.7cm$ is $b = 1.6$.

The change in the bias, can be approximated by a linear function of distance with a slope of $0.5/18 \approx 0.03$. It is likely that just as in fig. 2.15a the bias has a periodic component, however it is hard to detect due to noise. It is unlikely that the rise of the bias was caused by a change in the temperature, since the PLS was allowed a 3 hour warm-up time prior the commencement of the experiment. Furthermore the range of a stationary point was also recorded during the experiment, and only about one mm drift was observed.

A possible explanation for the rise in the bias is the following: because the PLS uses laser as a light source, the illumination of the target surface does not depend on range (disregarding the divergence of the laser beam, and the attenuation of the air). However, if a Lambertian distribution of the reflected light is assumed, then the amount of reflected light reaching the photo receiver decreases with increasing range. Less light causes a slower slope of the output signal of the photo receiver, which causes the object to appear further back. It is possible that the bias appears as linear only because the measurement interval was too short. It is assumed that the error gets reset by the compensation mechanism of the laser whenever the received signals peak leave one band and enters another band (see section 2.5.2).

In this chapter the standard deviation of range measurements for the PLS was approximated by a constant $\sigma_r = 2.4cm$ when estimating line segment uncertainty due to random errors. A constant was chosen for two reasons: firstly the variation in the standard deviation is not large compared to its mean (see fig. 2.15a). Secondly, if only one scan is taken of a plane, one cannot know which measurement has what standard deviation, therefore the worst case is assumed.

The same experiment was repeated with the Sick LMS. The LMS was operated in its mm mode. The result of the experiment is in fig. 2.16b. The corresponding simulation result shown in fig. 2.16a was generated using (2.4) and (2.5) with $\sigma_n = 1.7mm$. The standard deviation is a sinusoid function with a mean around 2.8mm, an amplitude around 1.8mm and a period of



(a) (b)

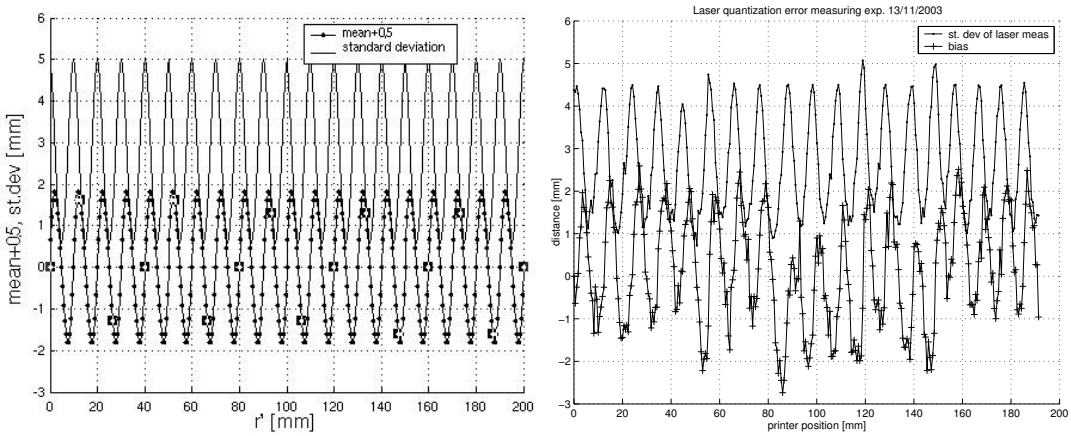
Figure 2.15: Effect of quantization on the PLS output: a) simulation, b) measurement

10mm. Note that in the line segment error modeling experiments described later, the Sick LMS is used in its cm mode since this mode is used also in chapters 3 and 4 for SLAM using advanced sonar fused with laser range finder measurements and with scan matching. Centimeter mode is used in chapters 3 and 4, because then the Sick LMS measurements can be used without modifying the existing software created for the Sick PLS which can measure only in cm mode, and also because the Sick LMS has a much longer measurement range in its cm mode than in its mm mode. The experiment performed with the LMS in its cm mode by moving the target plane away from the laser proved to be unusable, therefore the results are not shown here. However the range standard deviation of the LMS measurements were found to be approximately $\sigma_{rl} = 1cm$. This value is used later in the line parameter random error experiments.

The bias error in fig. 2.16b for the LMS in mm mode appears to be a sinusoid function superimposed on a function of undetermined character. The sinusoid function is as predicted with a quantization bias. Its phase is delayed by half the 10mm period with respect to the standard deviation. Unlike in the case of the PLS a linearly increasing trend in the bias is not observable. The amplitude of the quantization bias b_{lms} is around 1.9mm. As mentioned in the previous paragraph, in the rest of this thesis the Sick LMS is used in its cm mode. Since the quantization step and the noise entering into the quantizer is the same in the cm and in the mm mode of the LMS and only the internal error compensation is different, the same $b_{lms} = 1.9mm$ is used in the systematic line error experiments described later.

2.5.6 Experiment with Target Plane Rotated

In paper [Ye and Borenstein, 2002] the authors measured the error in the range measurements of the LMS laser scanner. They found that the error in the range depends among others on



(a) (b)

Figure 2.16: Effect of quantization on the LMS output: a) simulation, b) measurement

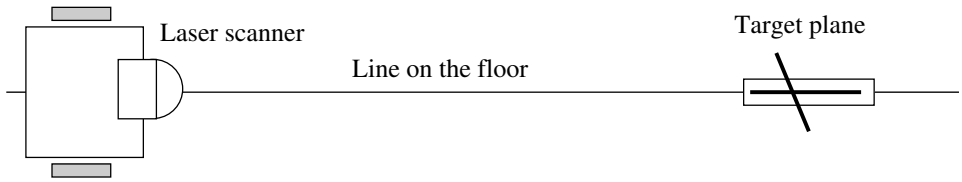
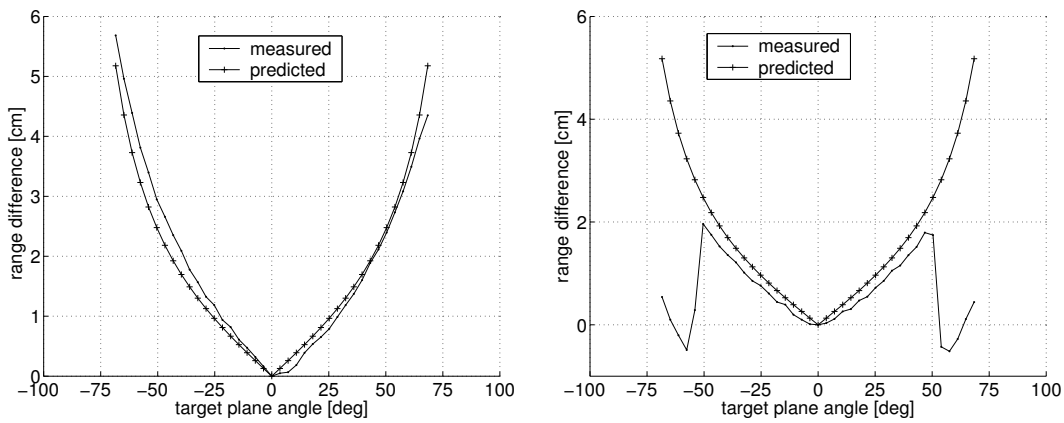


Figure 2.17: The alignment of laser scanner, line on the floor and calibration tool target plane in the experiment aimed to measure range bias changing with incidence angle.



(a) (b)

Figure 2.18: Measured range change depending on target plane angle

the incidence angle of the beam and the plane. However, they provided no explanation for the source of this error, or any quantitative approximation of the error. In this section the results of similar but more elaborate experiments are described which were performed with the PLS. The method deployed for measuring the error change with incidence angle was the following: The laser calibration tool's longitudinal axis was aligned to be parallel with a line running on the floor of the lab (see fig. 2.17). The axis of rotation of the target plane was adjusted to be above the line. Then the differential drive robot carrying the PLS laser was moved so that the odometry center would be roughly above the line as well. By observing the laser's output, the robot's position was tuned so that the target plane would be represented with an odd number of points, while the center point had a bearing of $90°$. This way one could ensure that there is a laser spot close to the center of rotation, and that the angle between the laser beam of interest and the target plane is about $90° \pm 2°$. In the next step range readings were collected, while moving the target plane from $0°$ (at this orientation the angle between plane and beam is $90°$) to $+50°$ and to $-50°$. If there was a difference between the average of range readings taken at $+50°$ and $-50°$ then we moved the laser calibration tool slightly to the left or right, depending on the sign of the difference. The amount of movement depended on the magnitude of the difference. This process was repeated iteratively until the ranges measured at $+50°$ and $-50°$ were equal.

After ensuring that the center of the laser beam of interest coincided with the center of rotation, the experiments were started. In the experiments the target plane was rotated from $+70°$ to $-70°$ by $1.8°$ steps each 10min. To avoid bias from quantization to influence the measurements, the measurements were repeated after moving the target plane away from the laser by half the quantization step, eg. by 2.5cm. In the next step the average range for each angle was computed. To allow bias due to quantization to be canceled, the average of the averages for the same angles, but different distances were calculated. Then from all the obtained results, the range value for $0°$ was subtracted.

Two examples of the results are shown in fig. 2.18, whereas the approximated ("predicted") error was calculated as $\Delta r_a = 2|\tan\beta|$, where β is the angle of the target plane. The tangential relationship and the constant 2 in Δr_a have been identified experimentally from figures such as fig. 2.18. In fig. 2.18a the measured error is close to the approximated, however in fig. 2.18b it seems as if at angle $\pm 50°$, 2.5cm was subtracted from the readings. A possible explanation for that is the following: as the target plane was rotated, less and less light came back to the laser scanner causing longer and longer slope of the output signal of the photo detector. It is suspected, that at $\pm 50°$ the signal level got into a different band (see section 2.5.2), and the laser used different compensation values. In the experiments the jump in the range readings occurred at different angles for different laser to target plane distances. Also note that the $\Delta r_a = 2|\tan\beta|$ approximation was tested only with one target plane, which was coated with

angle [deg]	-60	-40	-20	-10	0	10	20	40	60
range [index]	47.2	46.14	45.184	45.08	44.660	45.040	45.240	46.240	47.776

Table 2.2: Range measurements depending on the target orientation from [Reina and Gonzales, 1997].

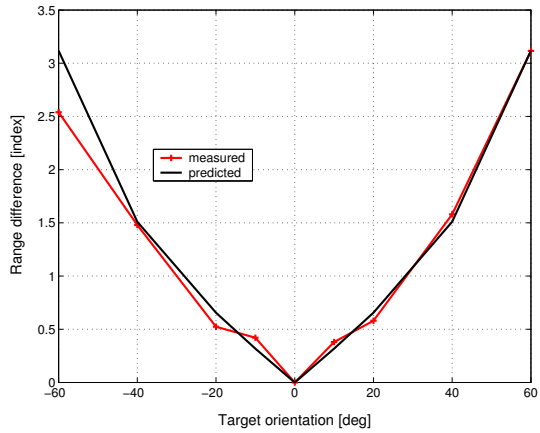


Figure 2.19: Measured range change depending on target plane angle for the Explorer laser scanner [Reina and Gonzales, 1997].

a non-glossy white paper, probably giving a good Lambertian reflection. It is possible that surfaces with more specularity would give different error characteristics.

Due to technical difficulties this experiment was not repeated with the LMS laser scanner. However the applicability of a tangent error model is not limited to the PLS only. In [Reina and Gonzales, 1997] a time of flight laser range finder is characterized. The name of the laser range finder is Explorer and it was manufactured by Schwartz Electro Optics. In the paper the authors also measured the dependence of range on the incidence angle. Their measurements are reproduced in table 2.2. After subtracting the range reading corresponding to 0° from all range readings, the measurements are shown in fig. 2.19. In the figure the modeled error $\Delta r_a = 1.8|\tan\beta|$ is also depicted. The resemblance of model and measurements clearly indicates that the tangential relationship between target orientation and measured range difference and therefore the line error model is applicable to laser scanner other than PLS.

2.5.7 Line Error Experiments

The described random and systematic line parameter error models were validated by using the "perfect"corner tool and a mobile robot. For the systematic line parameter errors only indirect tests of the orientation error were performed because precise measurement of line parameters is complicated and/or time consuming. In the experiments the "perfect" corner tool was used, whose opening angle was estimated as $89.85^\circ \pm 0.3^\circ$. In the experiments a mobile robot was

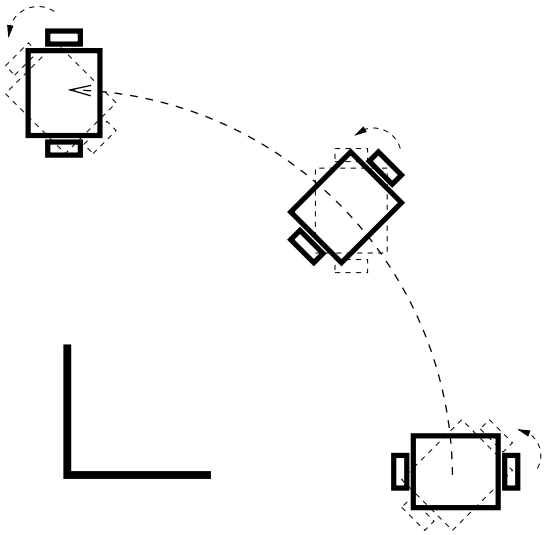


Figure 2.20: Robot moving around a perfect corner in the line parameter error experiment.

	PLS	LMS
Range standard deviation: σ_r	2.4cm	1cm
Identical bias parameter: r_b	5cm	3cm
Incidence angle error parameter: w	2	0
Error growing with distance: v_r	0.03 $(max\_err = 13cm)$	0
Quantization bias: b	0.17cm	0.18cm

Table 2.3: Parameters used in scan matching during the line segment experiments.

moving on a path resembling an arc around the "perfect" corner tool as can be seen in fig. 2.20. The motivation of the experiment is to observe the corner from different angles. On its course the robot stopped each few degrees, turned toward the corner and took thousands of scans. The places the robot stopped are referred to as positions in the following text. At each position scan points belonging to the corner were segmented into two lines. Line segments were fitted to the segmented scan points of each arm. During the experiment, the LMS was used in its cm mode. The cm mode was used because the range of the LMS is much longer in this mode and a longer range is important for SLAM in the next chapter.

The error parameters used in the experiments with the PLS and LMS are shown in table 2.3.

There were 3 experiments conducted for both the PLS and LMS. In experiment 1, the robot was moving on an arc of radius $R = 1.5m$ around the "perfect" corner, and stopped to collect scans at each $\Theta = 5°$. A sample scan taken with PLS at each position is shown in fig. 2.31. Each sub-figure of fig. 2.31 is numbered as the corresponding robot measurement position.

In experiment 2 the robot moved on an arc of $R = 2m$ radius and took scans each $\Theta = 10°$. Sample corner scans from PLS can be seen in fig. 2.33.

In experiment 3 the robot moved on an arc of $R = 3m$ radius and took scans each $\Theta = 5°$. Sample corner scans from PLS can be seen in fig. 2.35. Data from experiment 3 were used only in the testing of the systematic line error models.

When processing the measurements, in both systematic and nonsystematic tests, positions, where less then 1000 valid lines were fitted were ignored. Also measurements of line segments consisting of less than 10 points were also ignored. Measurements with few points occurred when a corner arm was pointing towards the laser.

In the segmentation of the arms of the corner, failing to identify the correct corner point in the data can lead to large errors especially, when one of the lines consists of only a few points. A presumably often used method for corner point detection consists of finding the furthest point from a line defined by the two non-touching endpoints of the lines (see [Gutmann, 2000]). Due to the noisy nature of the Sick PLS sensor's output, this method often misses the right corner point. This problem can be solved by sacrificing some computing power. The choice of corner point was refined by finding that point in the neighborhood of the initial corner point, which gives the minimum sum of error variances of both lines. Under error variance of a line the variance of perpendicular distances of points from the line calculated using principal component analysis (PCA) [Forsyth and Ponce, 2003] is meant. The calculation of variances was performed by calculating the eigenvalues of the covariance matrix of the x and y coordinates of points, and choosing the smaller eigenvalue. Since the eigenvector of the largest eigenvalue points to the direction of the largest variance, therefore the eigenvalue associated with the other eigenvector which is orthogonal to the first one represents the variance of points in the direction perpendicular to the line.

Since the Sick LMS laser scanner's range readings are not so noisy as the PLS's, identifying the furthest point in the corner measurement from the beginning and endpoint of the corner gives a good estimate of the corner point. To further reduce the effect of misidentification of the corner point, 3 points left and right of the corner point and the corner point are discarded.

Nonsystematic Line Errors

In the testing of the nonsystematic errors, only datasets from experiment 1 $(R = 1.5m,\ \theta = 5°)$ and experiment 2 $(R = 2m,\ \theta = 10°)$ were used. Note that experiment 3 is used only in the systematic line error testing described in the next section. Even though there were many different line segments in the experiments, the whole possible parameter space of line segments has not been covered. For experiment 2 the line parameters are distributed in the line parameter space as depicted in fig. 2.22. Each line which is represented by more than 1000 measurements is numbered for identification purposes. For experiment 1 the line parameter distribution is similar but more dense.

The covariance matrix of line segment range measurements representing the left or right

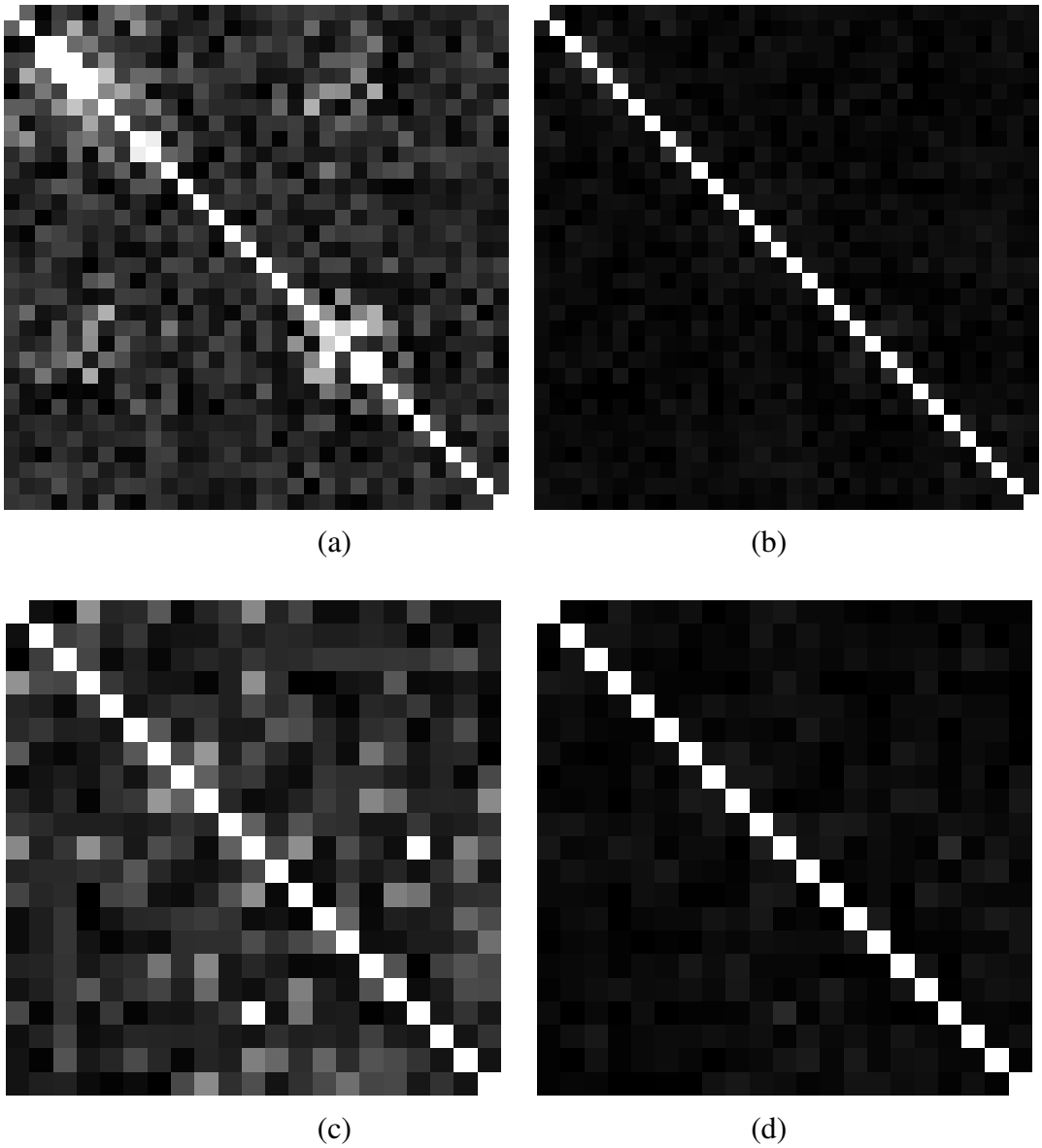


Figure 2.21: Covariance matrix a) and correlation coefficient matrix b) of the range readings for line 1, respectively for line 13 (c and d) for the PLS. Darker color means lower value. The corresponding covariance and correlation matrices are shown in section A.7 of the appendix.

arm of the corner was modeled as diagonal, because one range measurement is assumed to be uncorrelated with the value of other range measurements. Therefore (2.25) was used to determine the covariance of estimated line parameters. To show, that diagonal approximation of the measurement covariance matrix is not unrealistic, in fig. 2.21 the covariance and correlation coefficient matrices for the first couple of points of lines 1 and 13 of fig. 2.23 for the PLS are depicted. More details on the lines can be read in the next paragraphs. The correlation matrices can be viewed in their usual form in section A.7 of the appendix.

The measured line parameter covariances were obtained by calculating the covariance of measurements. In the results measured and estimated covariance matrices are represented as single standard deviation error ellipses. The probability of a point falling inside an ellipse is about 40%.

In the results, lines are numbered the following way: right arm from the first position is assigned one, right arm from the second position is assigned two and right arm from the last n-th position is assigned n. Left arm from the first position is assigned $n+1$, and left arm from the last position is assigned $n+n=2n$.

Results for the PLS can be seen in fig. 2.23–2.26. Figure 2.23 depicts segmented measurements of the arms of the right angle corner tool of the $R = 2m$; $\Delta\theta = 5^\circ$ experiment. Measured and estimated line parameter covariances are shown in fig. 2.24 with solid and dotted lines respectively. The error models are good enough most of the time, however in the case of line 1,2,19 of the first experiment, the measured covariances are slightly larger than the predicted. The reason for this deviation is unknown. It is suspected that the assumption of uncorrelated errors is violated since as it can be seen on fig. 2.23 the points of line 1,2 and 19 span through too few quantization levels. Therefore the information content of the measured points is smaller than modeled which results in optimistic covariance estimates. For lines 10, 20 the error ellipses are missing, because while being processed together, line 10 did not have a sufficient number of points.

The results of the experiment with $R = 1.5m$; $\Delta\theta = 5^\circ$ using the PLS (fig. 2.25 and fig. 2.26) are equally good as that of the first one except for those few cases where the measured line segment was pointing into the laser. For line number 18, the line was so well aligned with the laser, that only very few points represented the line segment, which resulted in a big difference between the measured and the predicted covariance matrix.

The experimental results with the LMS are shown in figures 2.27, 2.28 for the $R = 2m$; $\Delta\theta = 5^\circ$ experiment and in figures 2.29, 2.30 for the $R = 1.5m$; $\Delta\theta = 5^\circ$ experiment. Missing subfigures are due to the ignored line segments containing insufficient number of points or there were not enough line samples. The LMS results are worse than those for the PLS, however they are still sufficiently close...

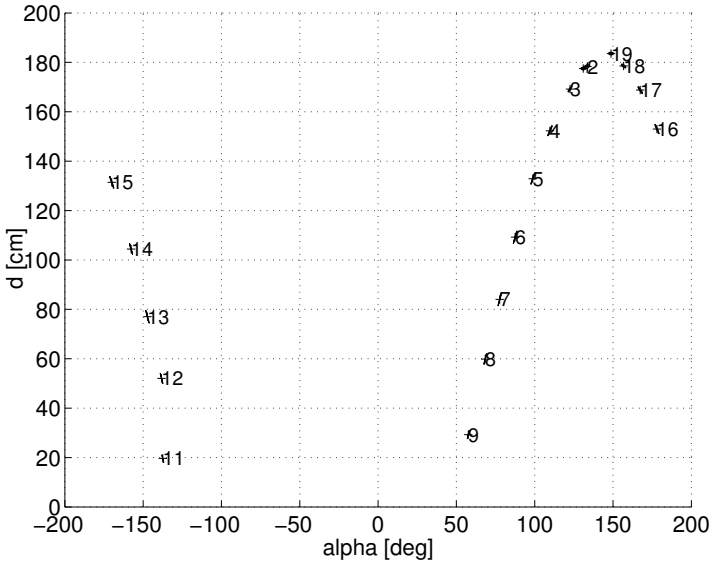


Figure 2.22: Parameters of lines taking part of the random noise model evaluation. $R = 2m$, $\Delta\theta = 10°$.

Systematic Line Errors

Lines were fitted to the arms of the tool in the scans, and the average angle difference was evaluated for each position which should ideally be 90°. It is assumed that the averages were influenced only by systematic errors, because each average was calculated from about 3000 samples for the PLS and more than 1000 samples for the LMS. The opening angle of the corner was estimated by calculating the difference of the estimated angles of the corner arms.

Systematic line parameter error estimates were calculated only for the first scan taken at each position. In the error calculation, the average line orientation and distance parameters were used. The absolute values of the errors from all 4 error sources for both lines were summed up to create a worst case estimate.

Experimental results for the PLS are shown in figures 2.31–2.36. Sample scans taken at each position are shown in figures 2.31, 2.33 and 2.35. Each sub-figure showing sample corners is numbered the same way as measured opening angles in figures 2.32, 2.34 and 2.36 where the estimated error bounds are marked with "*" and "+".

Modeling line parameter systematic errors for the PLS can be important, since the orientation systematic error can be larger than the orientation error. For example at position 9 in fig. 2.32 the corner opening error is almost 4°. However the orientation standard deviation for the corresponding line segments 9 and 9+18=27 of fig. 2.26 are only around 1°.

At the evaluation of the systematic error models, one needs to know that in case of a perfect systematic error model, either the upper, or the lower error mark would lie on the −90° mark of figures 2.32, 2.34 and 2.36. However as can be seen, the errors are most of the time larger.

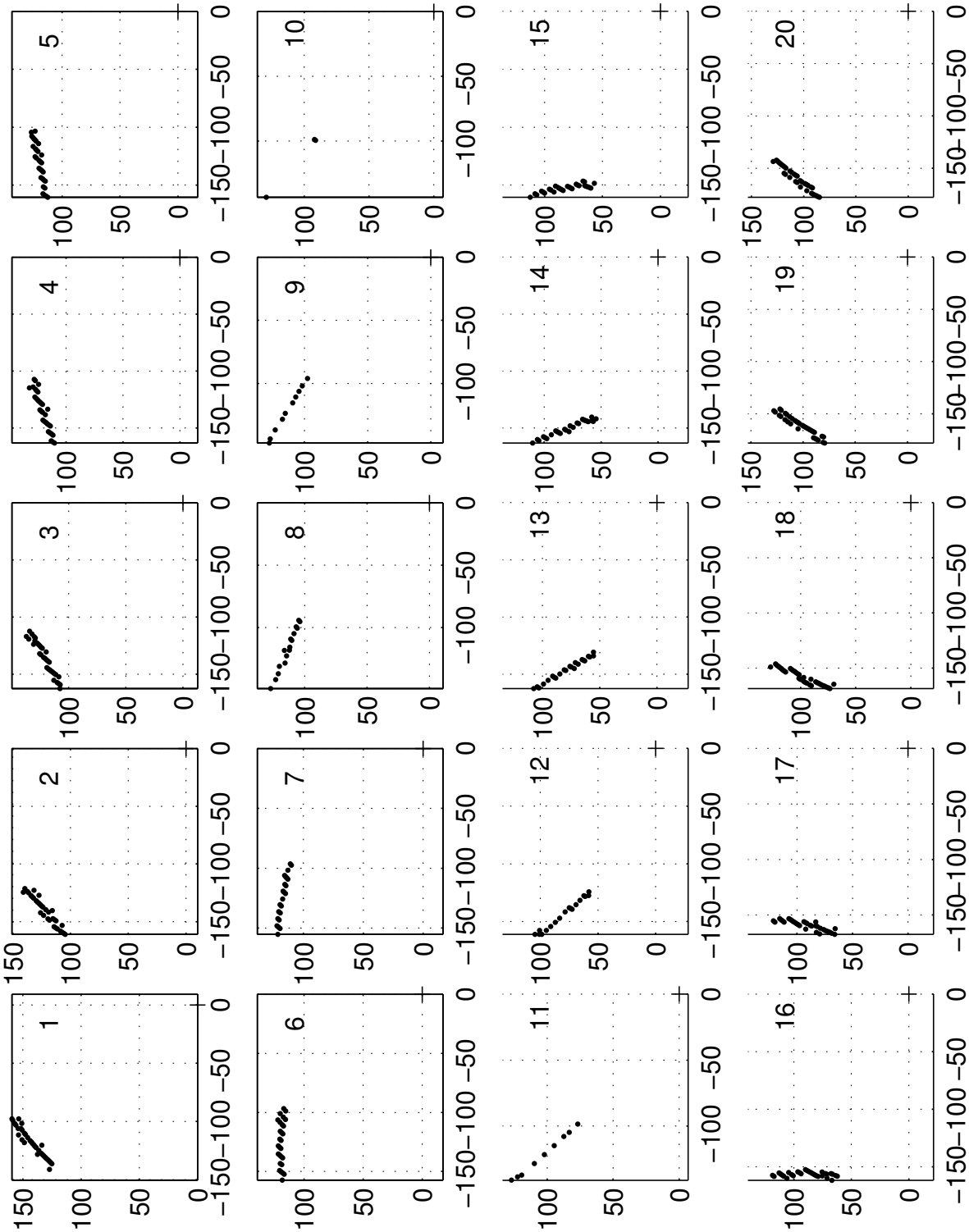


Figure 2.23: PLS raw data in Cartesian coordinate system to which lines were fitted. A "+" sign denotes the position of the laser. $R = 2m$, $\Delta\theta = 10°$.

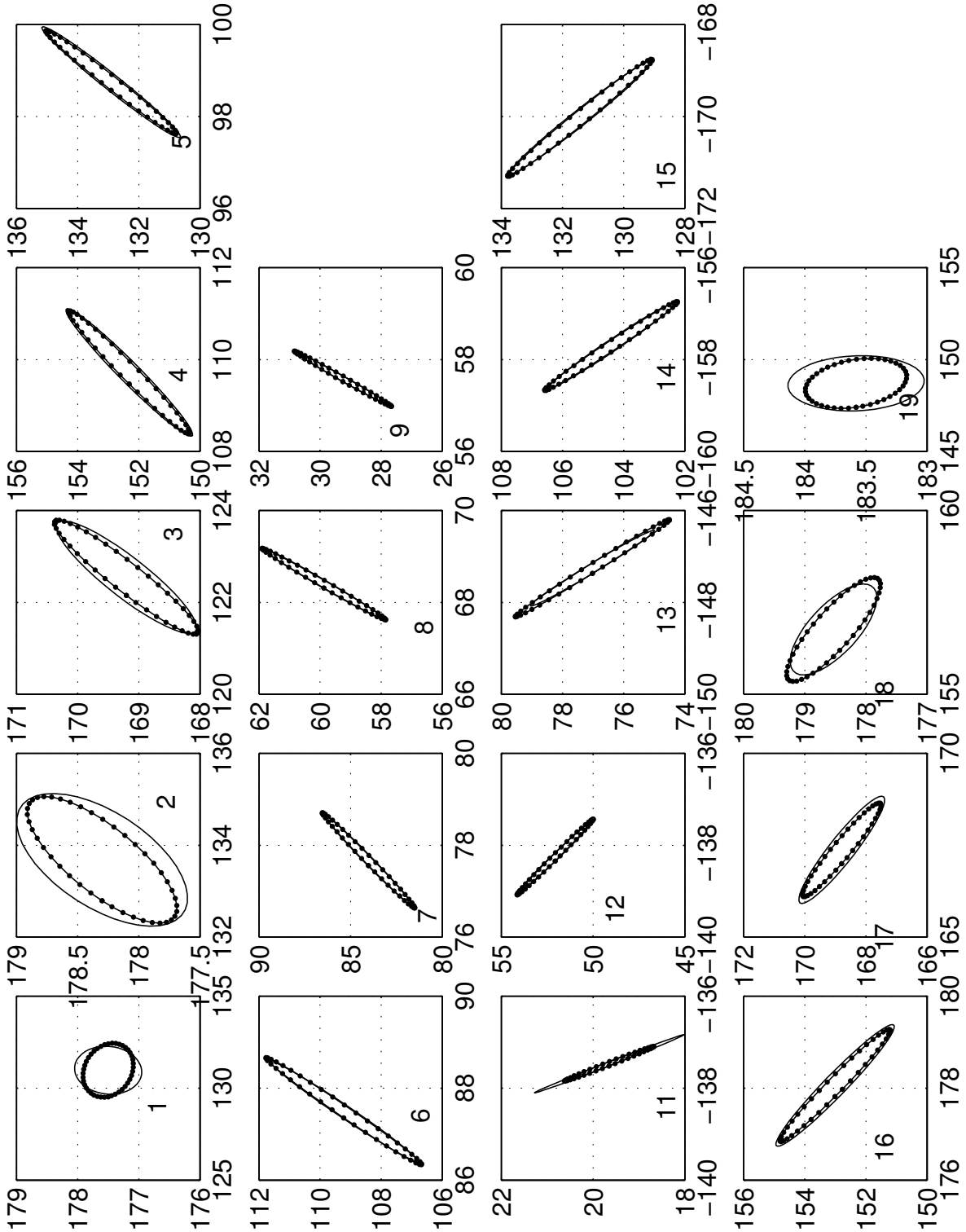


Figure 2.24: PLS error ellipses of the measured (solid line) and estimated (dotted line) line parameter covariances plotted in line parameter space. Horizontal axis: angles, vertical axis: distances. $R = 2m$, $\Delta\theta = 10°$.

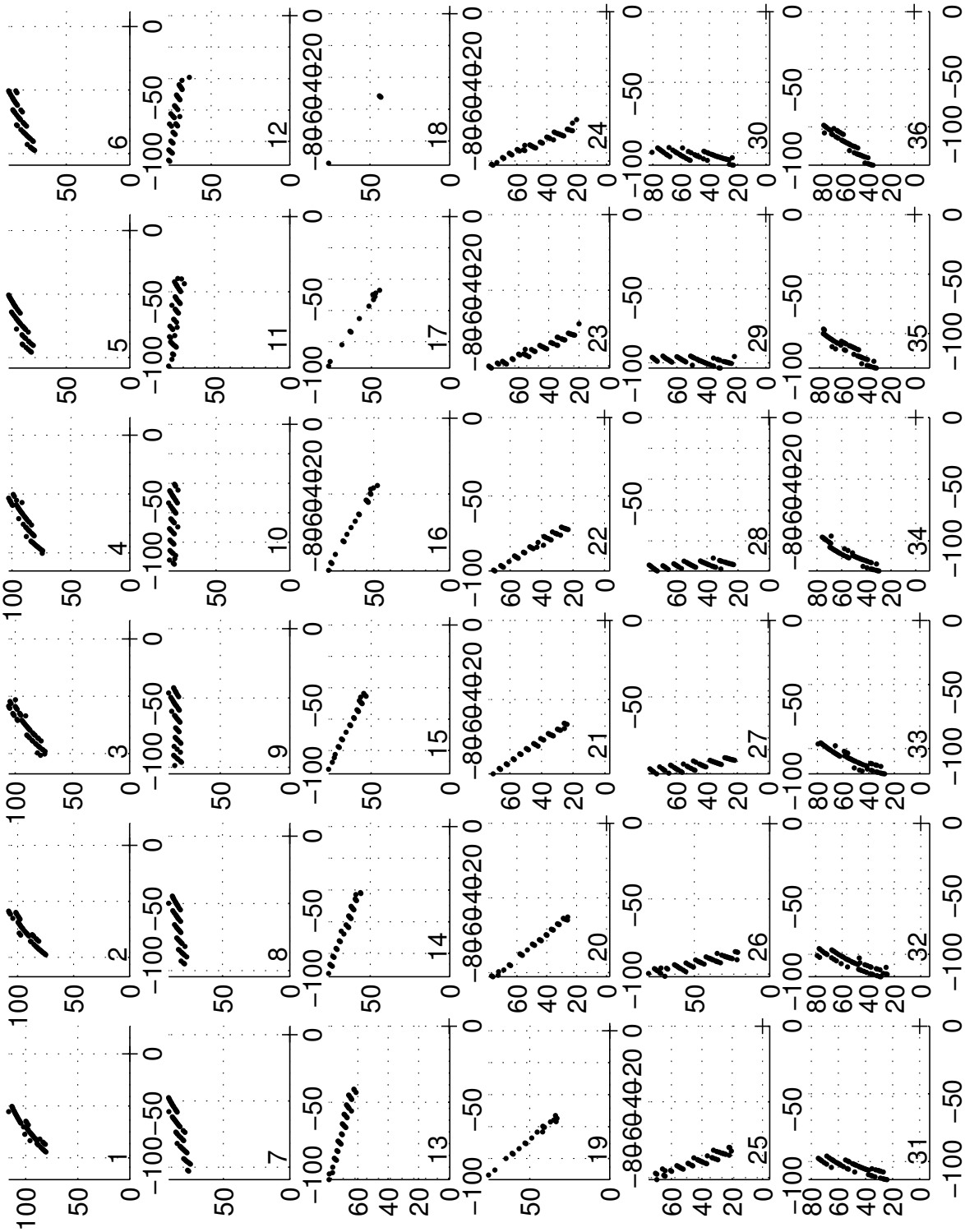


Figure 2.25: PLS raw data in Cartesian coordinate system to which lines were fitted. A "+" sign denotes the position of the laser. $R = 1.5m$, $\Delta\theta = 5°$.

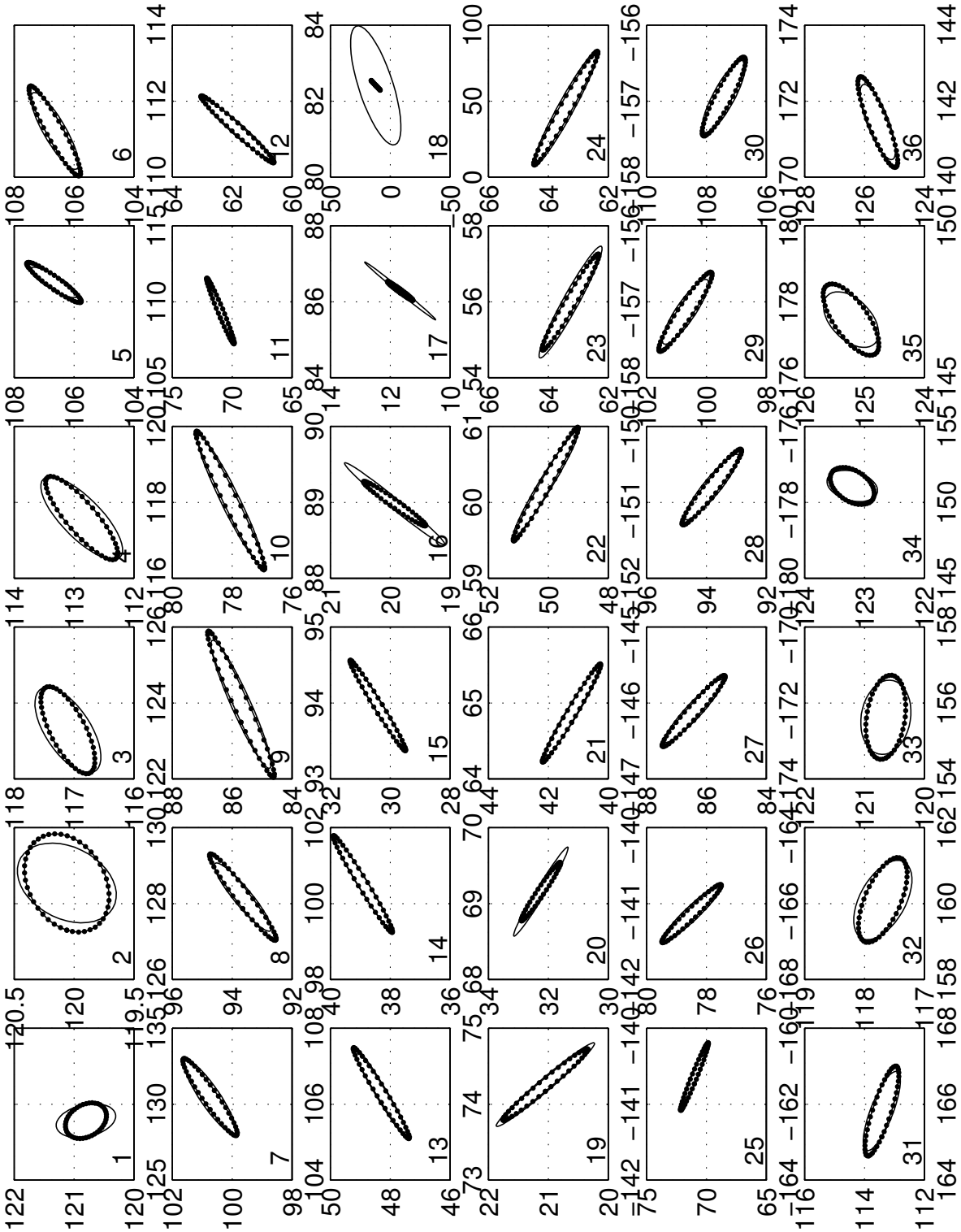


Figure 2.26: PLS error ellipses of the measured (solid line) and estimated (dotted line) line parameter covariances plotted in line parameter space. Horizontal axis: angles, vertical axis: distances. $R = 1.5m$, $\Delta\theta = 5°$.

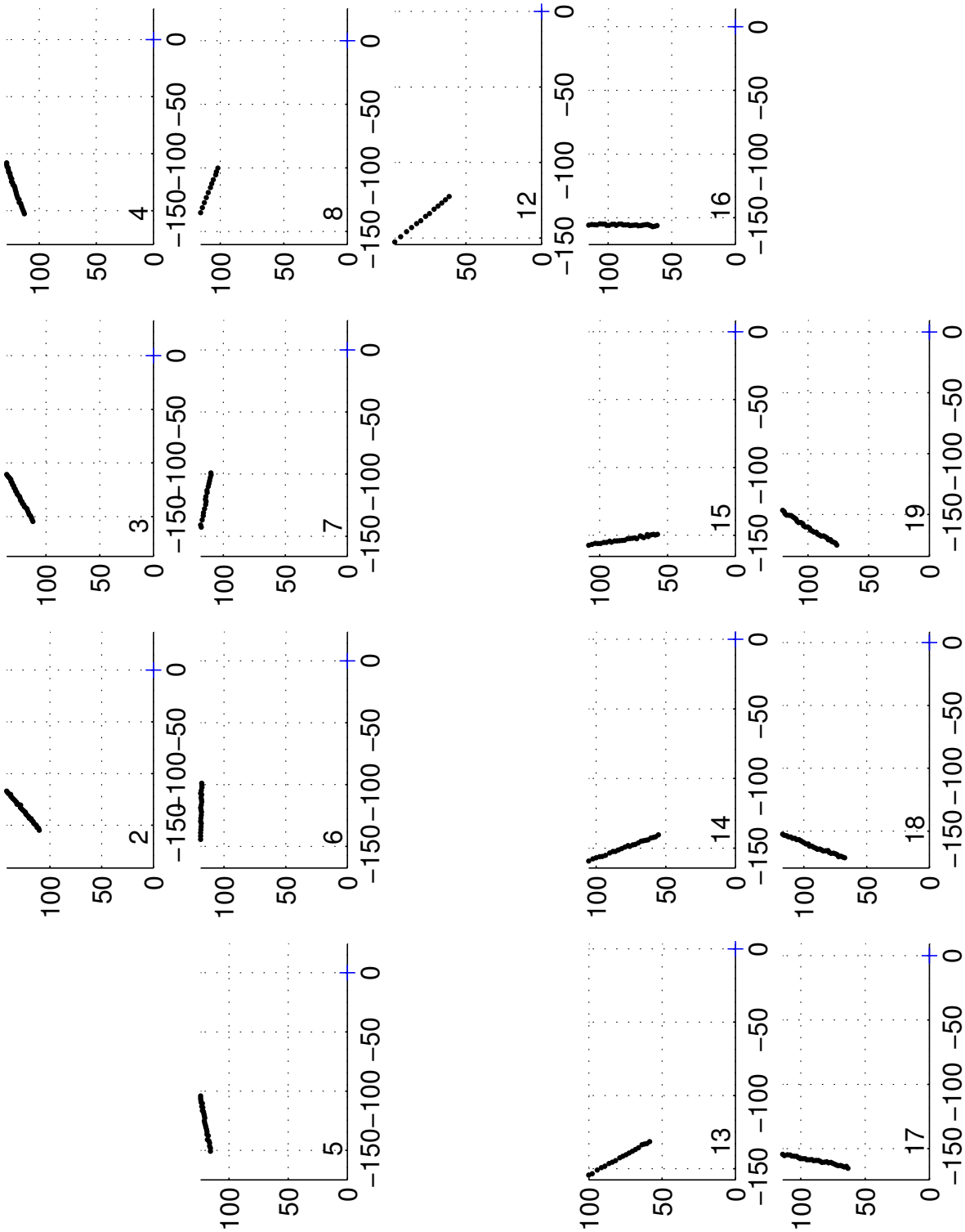


Figure 2.27: Raw LMS data in Cartesian coordinate system to which lines were fitted. A "+" sign denotes the position of the laser. $R = 2m$, $\Delta\theta = 10°$.

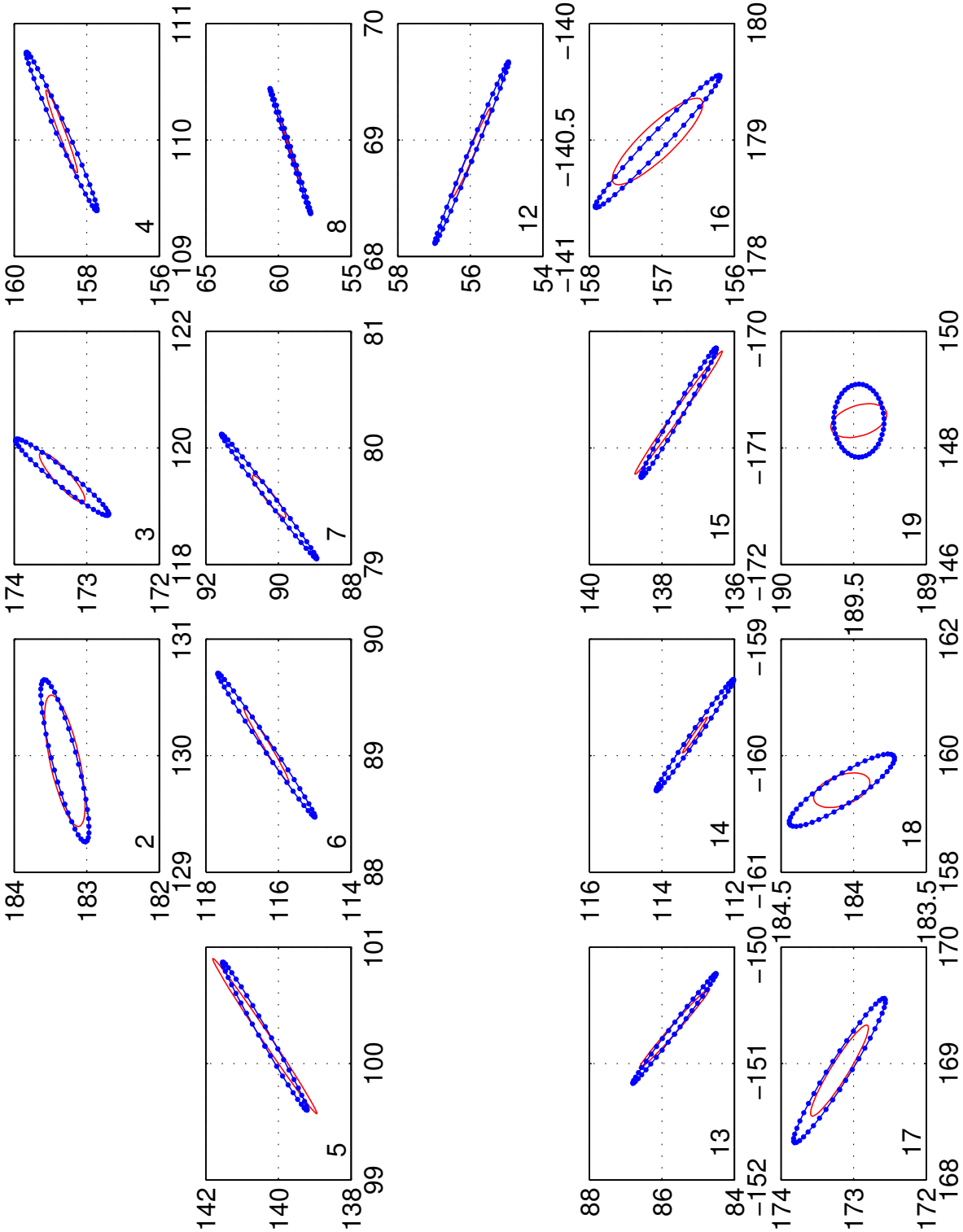


Figure 2.28: LMS error ellipses of the measured (solid line) and estimated (dotted line) line parameter covariances plotted in line parameter space. Horizontal axis: angles, vertical axis: distances. $R = 2m$, $\Delta\theta = 10°$.

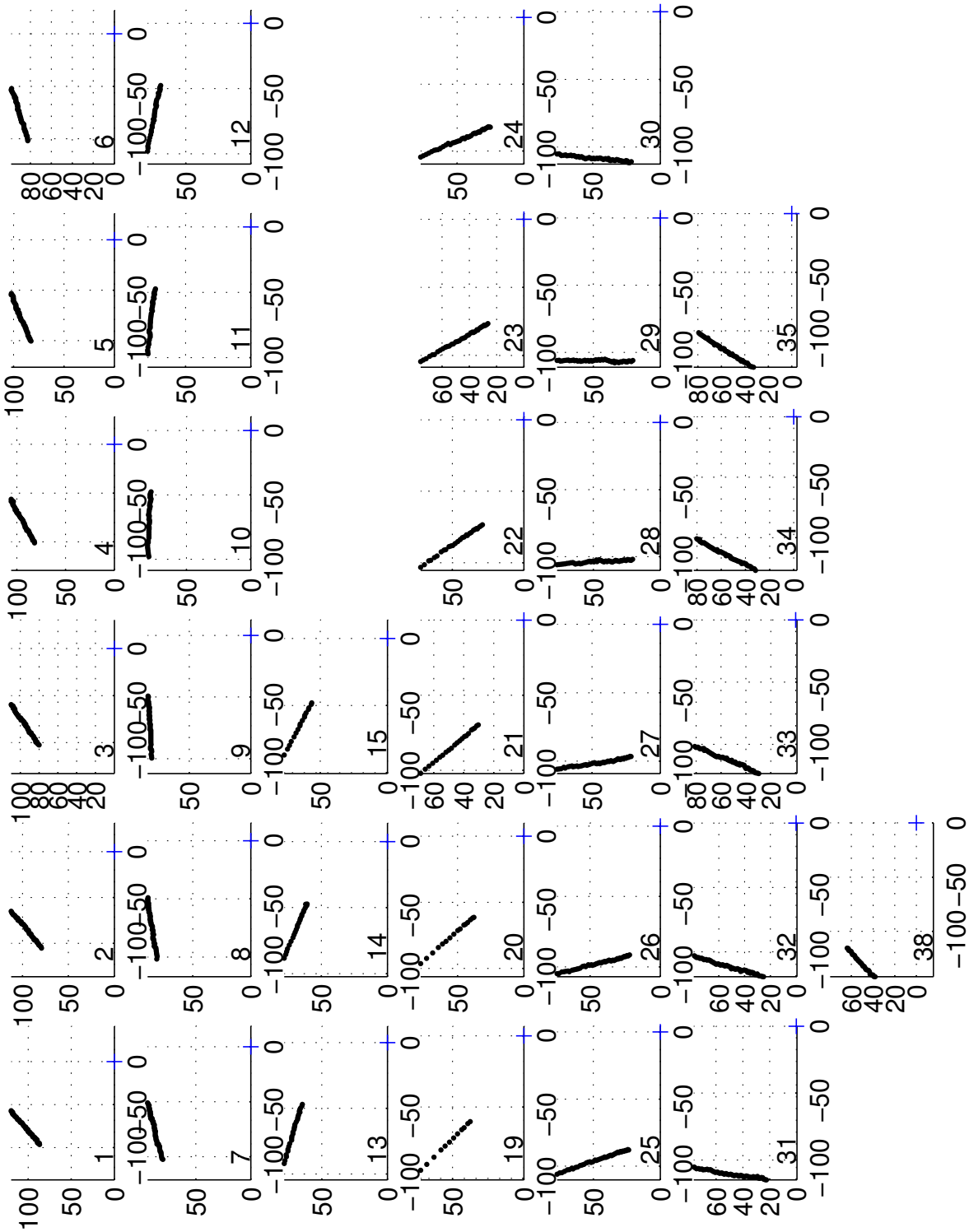


Figure 2.29: Raw LMS data in Cartesian coordinate system to which lines were fitted. A "+" sign denotes the position of the laser. $R = 1.5m$, $\Delta\theta = 5°$.

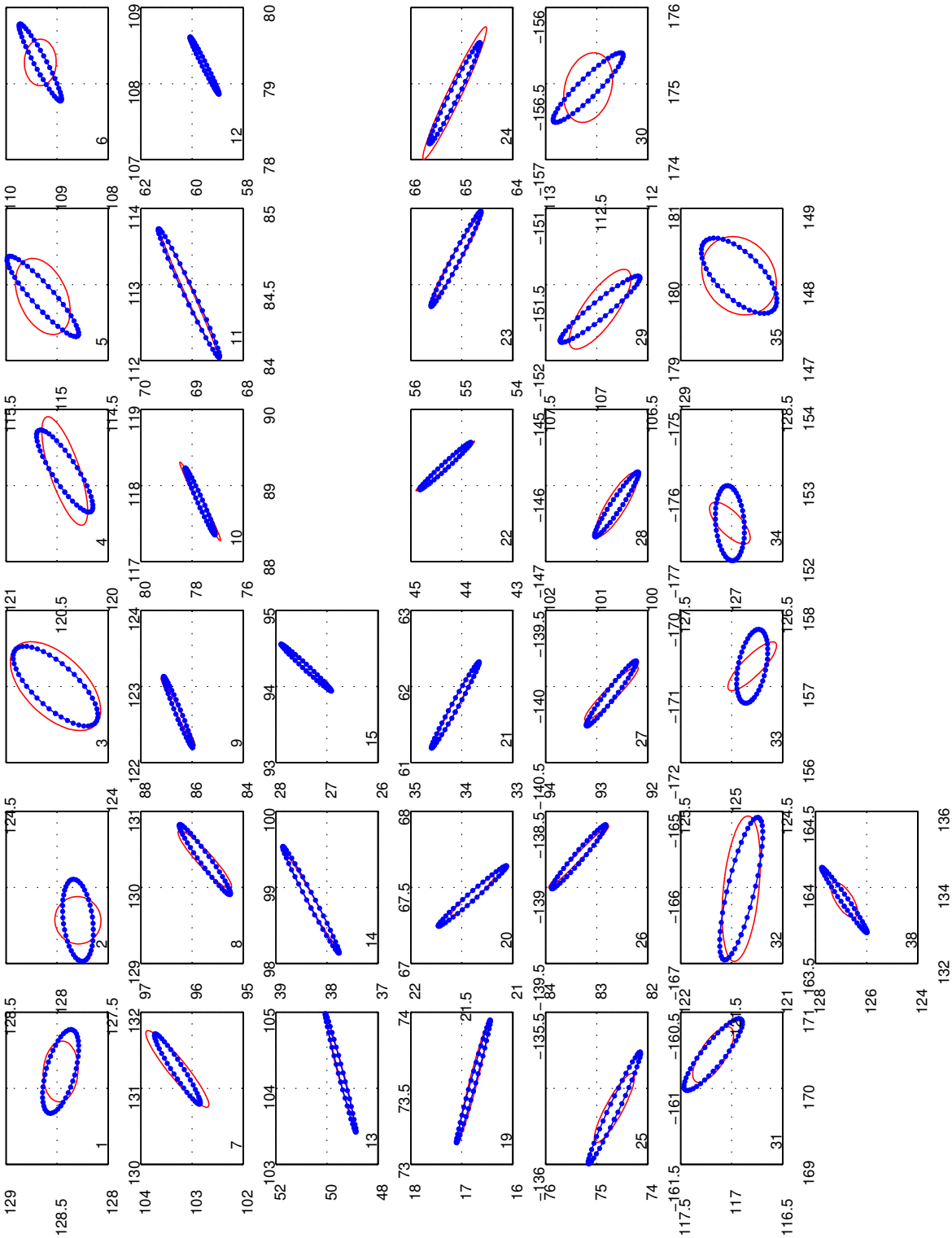


Figure 2.30: LMS error ellipses of the measured (solid line) and estimated (dotted line) line parameter covariances plotted in line parameter space. Horizontal axis: angles, vertical axis: distances. $R = 1.5m$, $\Delta\theta = 5°$.

This is because the absolute values of angle errors from each error source were added up, thus creating a worst case estimate. In reality, these errors some times cancel out each other.

In the error models the compensation mechanism of the PLS laser were not considered either, which corrects readings by measuring the peak of a returned signal. The reason for this is that one cannot access any of the internal parameters including the measured signal strength. Compensation can reduce the line parameter errors, or it can increase them depending on where it happens.

In some instances the estimated error was smaller than the measured one. It is more than likely than there are other error sources involved which were not modeled. One such error source is the error caused by wrong segmentation of the arms of the right angle calibration tool. Because range measurements are noisy, the determination of where one arm ends and the second arm starts is hard. Errors in the segmentation can cause systematic errors in some line parameters because points associated with the wrong arm can lie in some cased always on one side of the arm.

The results for the line segment systematic error experiments for the LMS can be seen in figures 2.37–2.42. The sum of systematic line orientation errors is more than twice smaller than that of the PLS. However, it is still important to model systematic errors, since they can be larger than the random ones for the LMS as well. For example the systematic error in fig. 2.38, position 3 is around 1.2°. The orientation standard deviations of the corresponding lines 3 and 21 of fig. 2.30 are only about 0.5°. The performance of the systematic error model for the LMS is satisfactory. In the three experiments the systematic orientation error was underestimated only once and that is depicted in fig. 2.38, position 3.

2.6 Conclusions

In this chapter, the results of laser error modeling efforts are presented. Firstly range error models have been developed, then tested and adapted to a Sick PLS and LMS laser range finders. The range errors include constant bias, bias increasing with distance, bias increasing with increasing incidence angle, quantization bias and random range error.

An approach for line parameter estimation is described in which parameters of a line are estimated directly in the laser's polar coordinate system, without the conversion of measurements into a Cartesian coordinate system (except for getting an initial estimate). This line parameter estimation method together with the range error model allows a satisfactory line parameter uncertainty estimation for the LMS and PLS, which has been verified experimentally. The line parameter error models were derived while assuming no errors in the laser beam bearings. Also the effects of tilted laser plane on line parameter errors have been investigated, together with errors occurring when the laser range finder is in motion. The line parameter

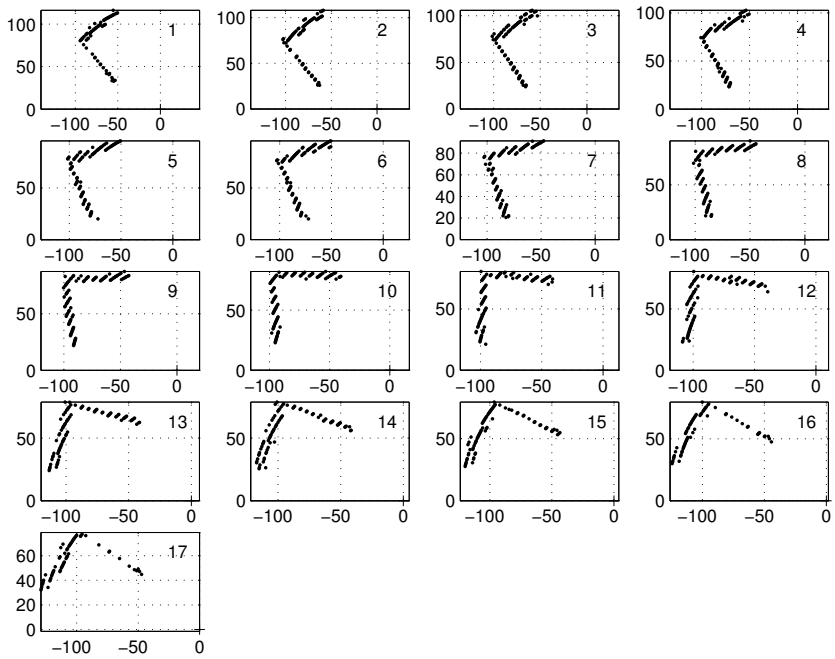


Figure 2.31: Samples of PLS scans of corners in Cartesian coordinate system for the experiment where $R = 1.5m$, $\Delta\Theta = 5^\circ$. Laser scanner position is at $(0,0)$ in all of the sub-figures.

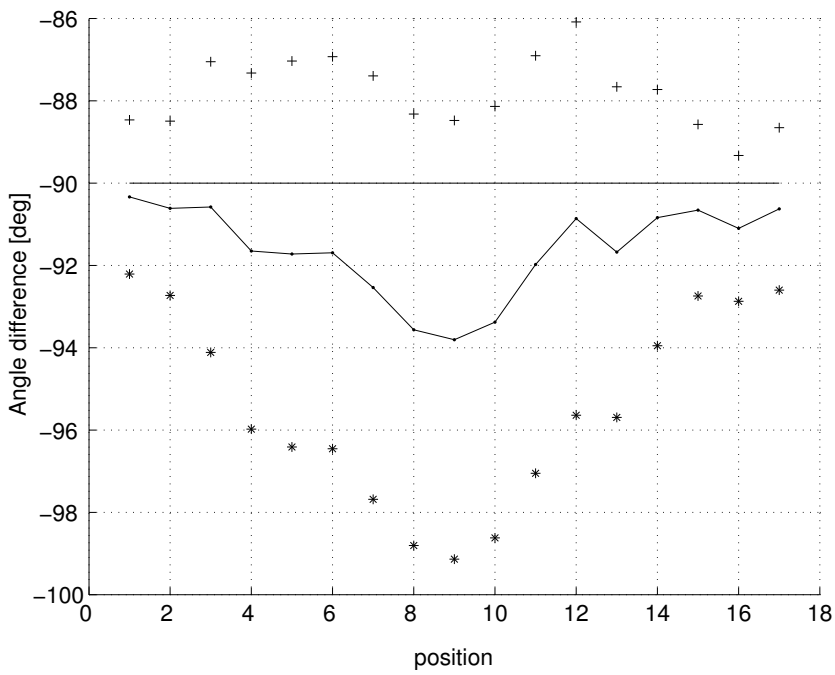


Figure 2.32: Systematic error model test ($R = 1.5m$, $\Delta\Theta = 5^\circ$) for PLS: measured opening angle of a right angle corner and error upper and lower bound estimates.

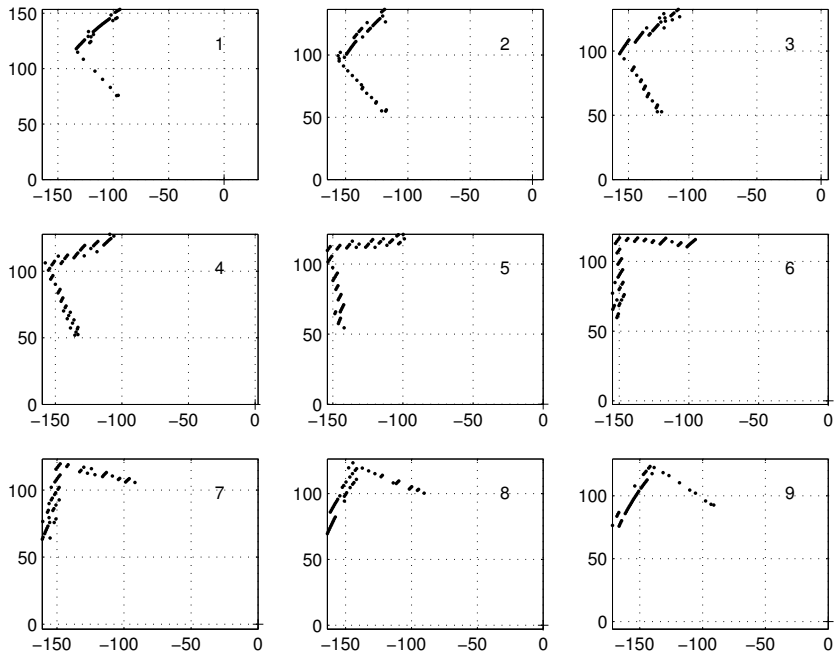


Figure 2.33: Samples of PLS scans of corners in Cartesian coordinate system for the experiment where $R = 2m$, $\Delta\Theta = 10°$. Laser scanner position is at $(0,0)$ in all of the sub-figures.

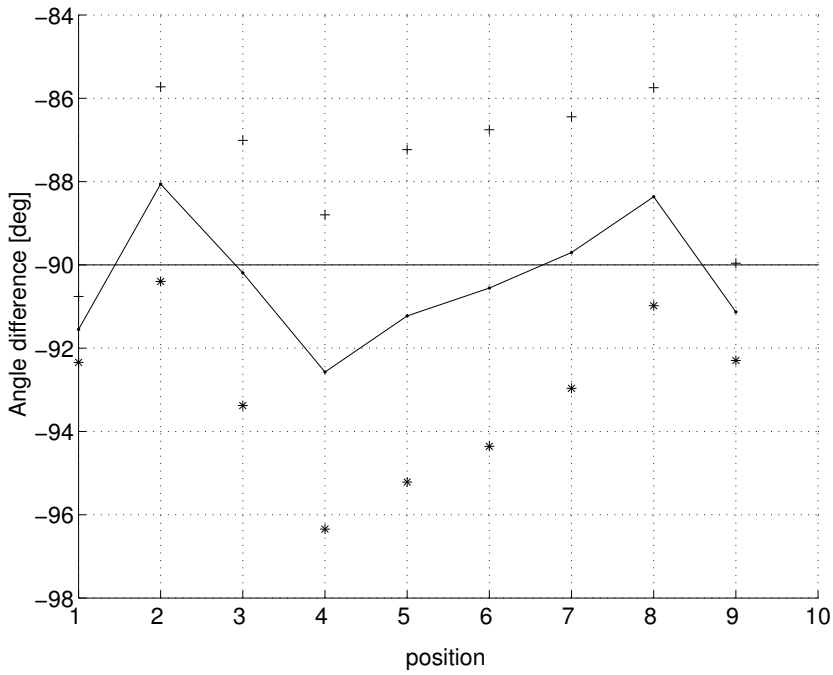


Figure 2.34: Systematic error model test ($R = 2m$, $\Delta\Theta = 10°$) for PLS: measured opening angle of a right angle corner and error upper and lower bound estimates.

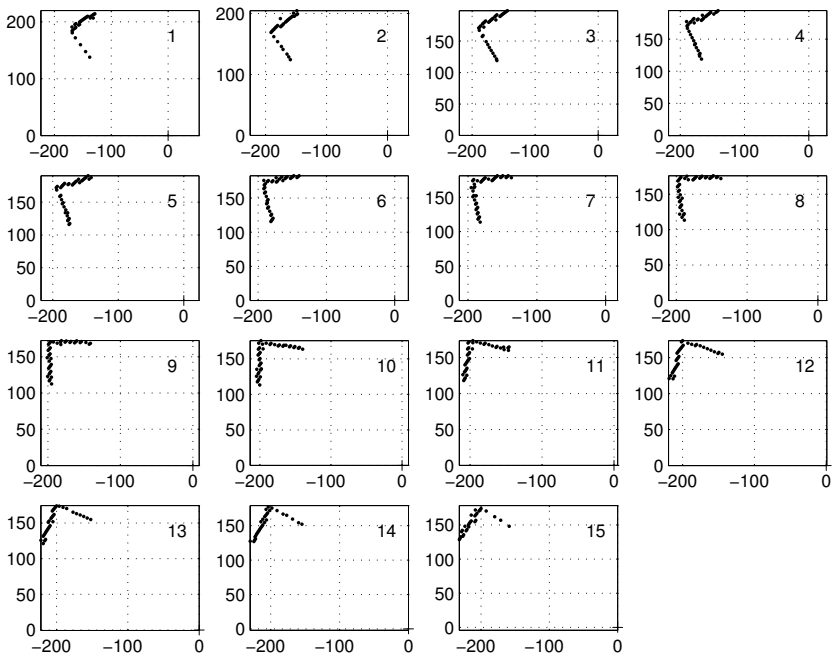


Figure 2.35: Samples of PLS scans of corners in Cartesian coordinate system for the experiment where $R = 3m,\ \Delta\Theta = 5°$. Laser scanner position is at $(0,0)$ in all of the sub-figures.

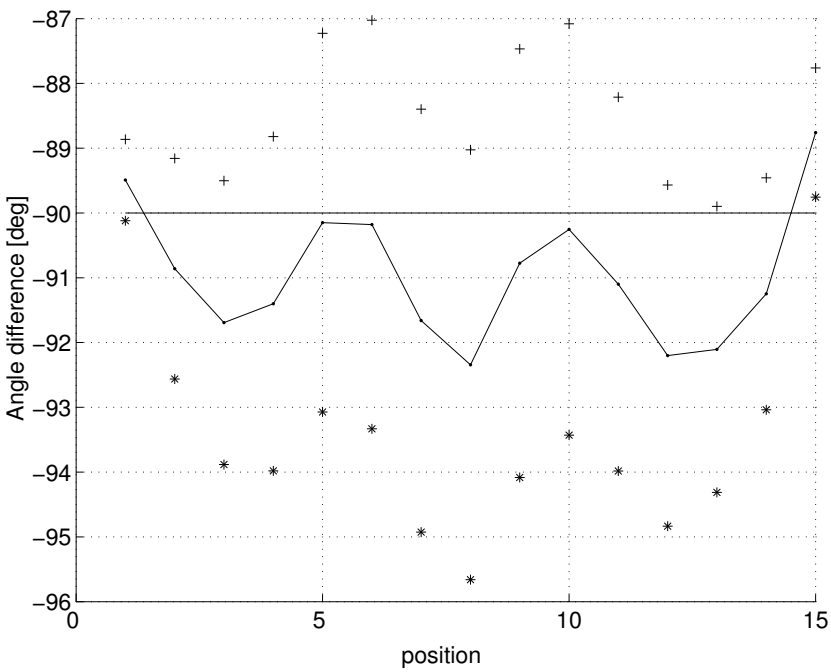


Figure 2.36: Systematic error model test ($R = 3m,\ \Delta\Theta = 5°$) for PLS: measured opening angle of a right angle corner and error upper and lower bound estimates.

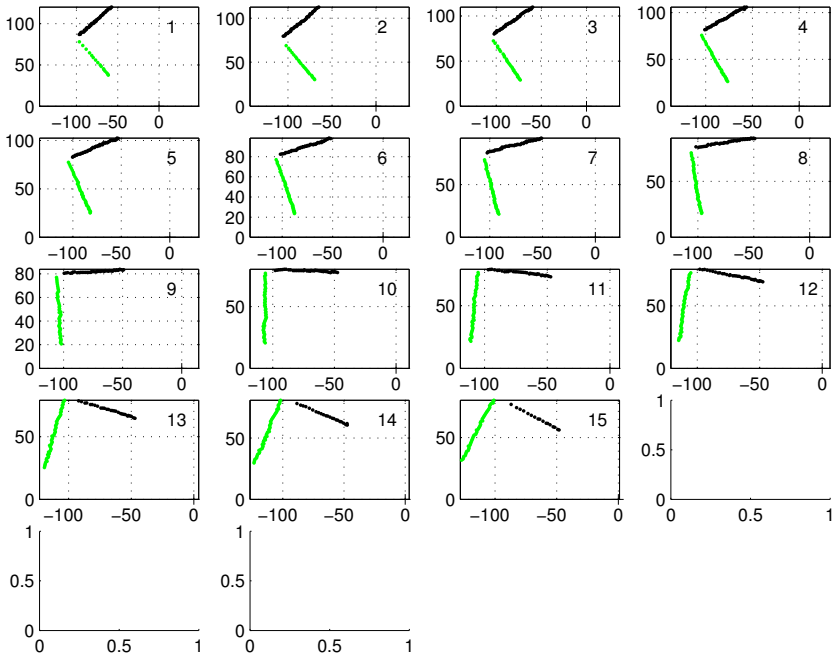


Figure 2.37: Samples of LMS scans of corners in Cartesian coordinate system for the experiment where $R = 1.5m$, $\Delta\Theta = 5^\circ$. Laser scanner position is at $(0,0)$ in all of the sub-figures.

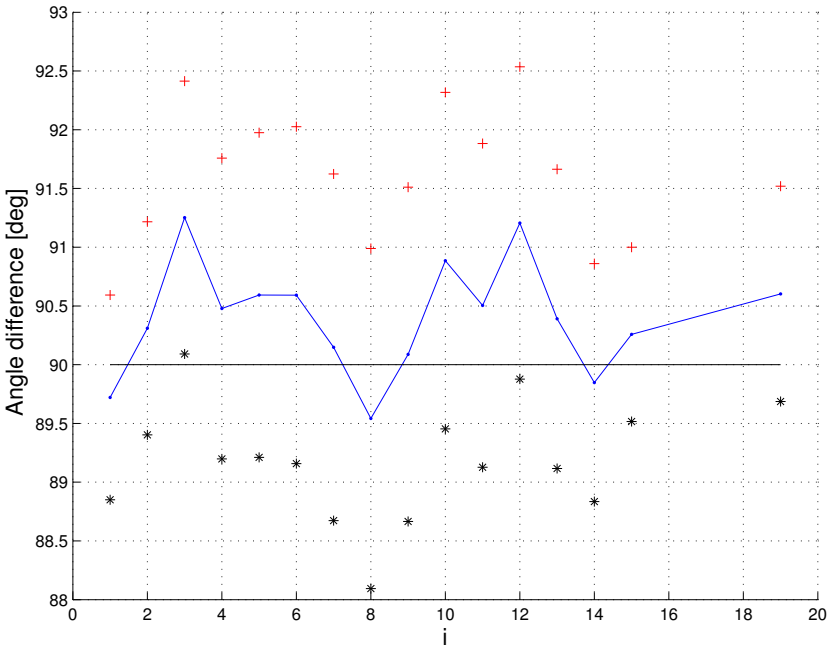


Figure 2.38: LMS Systematic error model test ($R = 1.5m$, $\Delta\Theta = 5^\circ$): measured opening angle of a right angle corner and error upper and lower bound estimates.

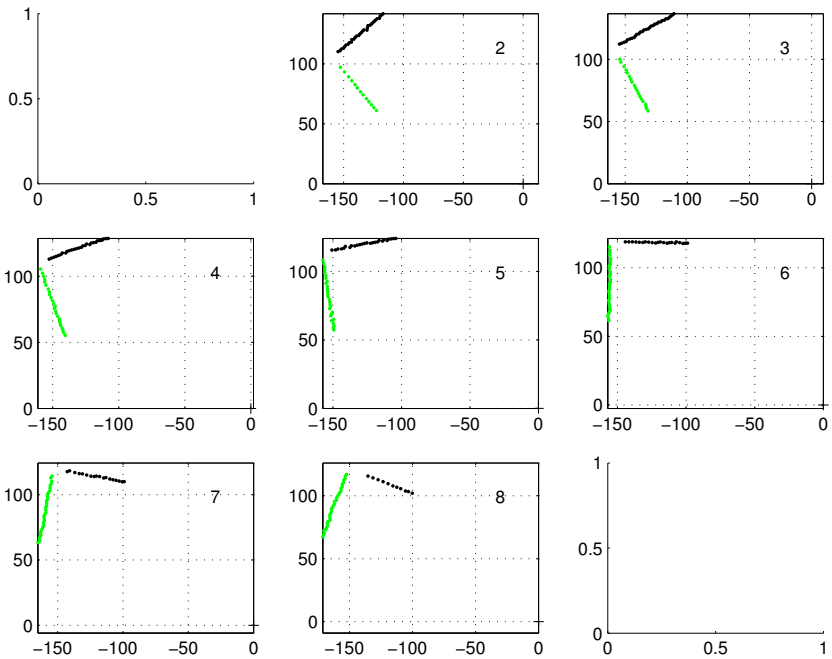


Figure 2.39: Samples of LMS scans of corners in Cartesian coordinate system for the experiment where $R = 2m$, $\Delta\Theta = 10°$. Laser scanner position is at $(0,0)$ in all of the sub-figures.

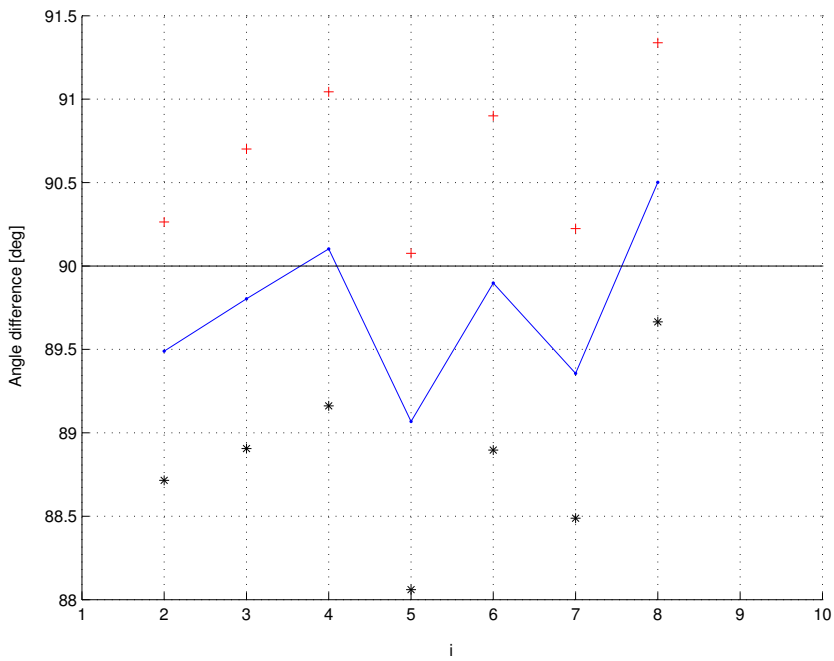


Figure 2.40: LMS systematic error model test ($R = 2m$, $\Delta\Theta = 10°$): measured opening angle of a right angle corner and error upper and lower bound estimates.

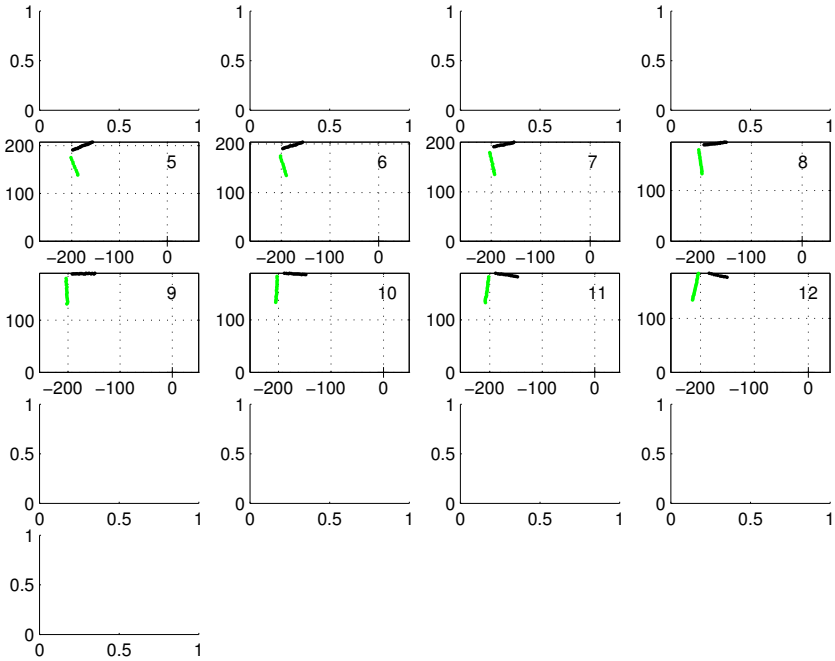


Figure 2.41: Samples of LMS scans of corners in Cartesian coordinate system for the experiment where $R = 3m$, $\Delta\Theta = 5^\circ$. Laser scanner position is at $(0,0)$ in all of the sub-figures.

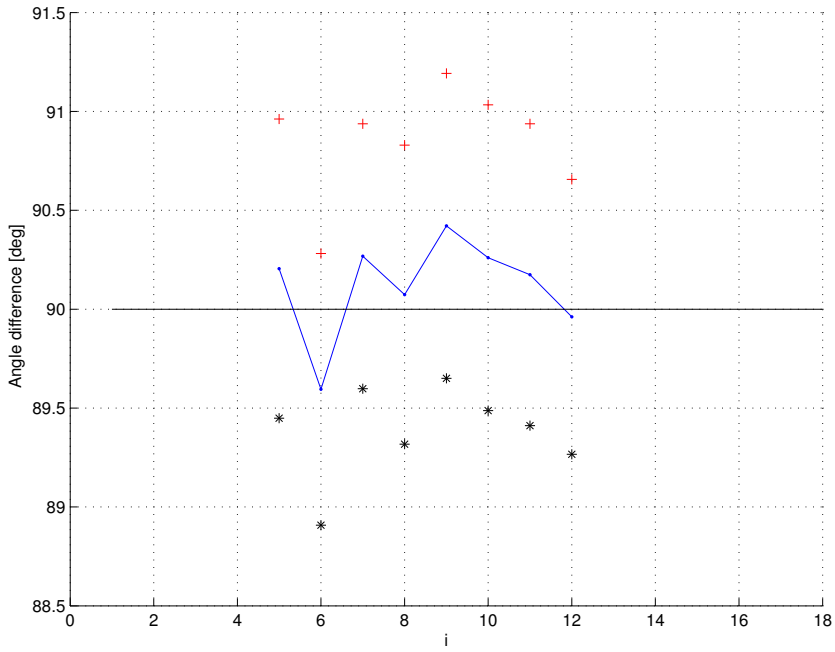


Figure 2.42: LMS systematic error model test ($R = 3m$, $\Delta\Theta = 5^\circ$): measured opening angle of a right angle corner and error upper and lower bound estimates.

estimation approach has been extended to the estimation of position and orientation of right angle corners.

It has been found that errors in the estimated line parameters can have a substantial systematic error component. As shown the systematic error can be even larger than random errors for both PLS and LMS. Experiments using a right angle corner have shown the systematic line parameter model to be reasonably accurate but on rare occasions, underestimates the angle error. Systematic error estimates were not used for error compensation since the upper bound was estimated only.

The applicability of the systematic error models is restricted to laser scanners with similar errors to the Sick PLS and LMS. One example is the laser scanner described in [Reina and Gonzales, 1997]. Even if the whole systematic error model is not applicable to a particular laser scanner, parts of it could be used. For example, many lasers have a bias in their range readings which changes during warm-up.

The contents of this chapter play an important role in the next chapter where laser line and corner measurements are fused with advanced sonar measurements for the performance of simultaneous localization and mapping.

Chapter 3

Advanced Sonar and Laser Fusion for SLAM

Mobile robots that build their own maps whilst using them for localization represent an important step towards creating useful autonomous mobile robots. For localization and mapping it is often an advantage if the sensor measurements used in the process are accurate and provide adequate information of the robot's environment. To satisfy these requirements often different sensor modalities have to be fused together. There has been a large variety of sensors used while performing simultaneous localization and mapping (SLAM): sonar (measuring only range) [Feder *et al.*, 1999], advanced sonar measuring range and bearing to planes, corners and edges [Chong and Kleeman, 1999; Kleeman, 2003], laser range finders [Jensfelt, 2001], monocular vision [Rybski *et al.*, 2003] and stereo vision [Davison, 1998]. Often a combination of sensors is used to improve performance such as vision with laser [Castellanos and Tardós, 1999].

In this chapter the fusion of laser range finder measurements with a special sonar sensor called an advanced sonar array is discussed together with its application to SLAM. 2D laser range finders can be characterized with good accuracy, long range and an excellent angular resolution. However, acquiring measurements in a plane results in missing those objects which are above or underneath the sensing plane. In certain environments such as long corridors the range resolution of laser range finders might not allow the detection of features which can constrain the robot's position in all directions. Transparent or highly reflective surfaces might not get reliably detected by a laser range finder. The advanced sonar array has a shorter range compared to a laser range finder, phantom objects can appear due to multiple specular reflections and smooth walls can be sensed from roughly 90° only. The wide beam width of the advanced sonar arrays mean that objects are not sensed in a plane, but in a cone, which results in more robust obstacle detection. Instead of rough measurements, advanced sonar arrays provide accurate range and bearing of automatically extracted features such as corners,

planes and edges. Even though the advanced sonar arrays measure range and bearing of planes, information about the length of planes can be gathered only indirectly through the motion of the robot. Through the synergy of advanced sonar and laser, the best of both sensors can be utilized while eliminating the negative characteristics. Laser measurements can help to remove specular reflections from advanced sonar measurements. Advanced sonars can detect small objects such as wall moldings, door jambs which are otherwise undetectable with laser range finders commonly mounted on mobile robots. The detection of such small objects by the advanced sonar array may allow a mobile robot to localize itself on long corridors with doors as the only features. Accurate plane measurements of the advanced sonar can be fused with laser measurements containing also the endpoints of the plane.

In this chapter the fusion of laser range finder and advanced sonar measurements is based on a statistical framework in which access to accurate measurement error models is crucial. The laser feature fitting and error modeling approaches of the previous chapter play an important role in this chapter, since over or underestimating errors in the measurements can lead to nonoptimal use (or waste) of information or even the divergence of the SLAM process.

Having a sparse feature map might help with localization, but path planning is best handled using occupancy grids. It is possible to register laser scans into an occupancy grid by using the pose of the robot provided from the SLAM approach, however shifting of the SLAM map makes this approach ineffective. In the approach discussed in this chapter, with each laser scan the relative position of the neighboring map features are also stored. When needed, the stored laser scans can be registered into an occupancy grid by regenerating their pose with respect to the SLAM map using the stored local features. This approach allows the generation of occupancy grids which are consistent with the feature based SLAM map. In the next chapter another approach for generating a consistent map of laser scan is described, where SLAM is performed using laser scan matching.

3.1 Introduction

There are several reasons for sensor fusion. When different sensors are measuring the same percept, false negatives can be rejected and measurement precision can be improved. In the case of complementary sensing, each sensor measures a different percept which complement each other.

Sensor fusion can be done on different levels [Kam *et al.*, 1997]. In this chapter only low level fusion is discussed which entails direct integration of sensory data. The other category for sensor fusion is high level fusion where an indirect integration of data occurs on higher levels such as through behaviors [Murphy, 2000].

A common sensor modality on robots is sonar. In this thesis the word sonar is reserved

for simple sonar systems which measure range only, based on thresholding the received signal and where the bearing of the target is known within the beamwidth of the transducer. In general, the main drawbacks of simple sonar systems are specularity, wide beam width and crosstalk [Jörg, 1995]. Wide beamwidth causes high uncertainty when determining the bearing of an object. Smooth walls act like mirrors due to relatively low wavelength of sonars used [Kim and Cho, 2001], causing little acoustic energy to be returned to a receiver in case of incidence angles differing much from 90°. Consequently, only walls enclosing nearly a right angle with the acoustic axis of the sonar are detected. Objects having good sound insulation properties like cloth covered partitions, might not appear in sonar measurements at all [Murphy, 2000]. Contrary to all the drawbacks, sonars are still being used on mobile robots, partly because of their low price and partly because of their wide beam width. The positive side of wide beam width is that it enables sonars to detect objects on a different height to the sonar, which is important for obstacle avoidance. In addition, sonars have no problem in detecting glass doors and mirrors [Enderle *et al.*, 1999], unlike laser scanners.

Laser scanners like the SICK series, on the other hand can be characterized with reasonable accuracy over a wide range (20cm to 50m) and an excellent angular resolution of 0.5 degree and more. However, there are several problems [Terrien *et al.*, 2000]. Because all the measurements are taken in a horizontal plane, obstacles above or underneath this plane will be missed. It is quite usual, that tables appear on the laser scan as 4 legs, which can be dangerous from an obstacle avoidance point of view. To alleviate this problem some researchers are mounting their laser scanners pointing slightly up-wards [Murphy, 2000]. Smoke and steam can cause erroneous measurements, just as glare. The detection of mirrors and glass doors can be a problem as well.

As we see, laser and sonar are complementing each other, therefore the question is not "Why to fuse sonar with laser?", but "How to fuse sonar with laser?". There are several approaches in the literature about fusion of laser and sonar, usually depending on the sensor suite used on the robot and on the purpose of the fusion (e.g. obstacle avoidance, path planing or map making).

In the literature only one case of advanced sonar and laser range finder fusion was found. In [Vandorpe *et al.*, 1996] a "tri-aural" sensor capable of measuring range and bearing to planes, corners and edges was used with a laser scanner and a conventional sonars. Laser and conventional sonar measurements were combined into a polar sector map by choosing the shorter measurement from sonar and laser for each bearing. If laser and sonar measurements were roughly the same, the sector map was updated by the average of the measurements. This sector map was then used to update a global world occupancy grid map. The measurements of the 'tri-aural' sensor were transferred directly into the global map. This approach used the range and bearing information of the 'tri-aural' sensor but discarded the information about the

type of the object.

In the following two paragraphs work regarding fusion of simple sonar and laser is reviewed.

Fusion of laser and sonar has been done for different purposes. In [Jörg, 1995] sonar and laser readings were combined into a sector map for obstacle avoidance. Point features extracted from sonar readings which were recorded at different positions are used for localization in [Jensfelt, 2001] together with line segments and door features from a laser range finder. Fusion of sonar and laser measurements took place in a Kalman filter when updating the estimated robot location with the measurements of landmarks.

When mapping, laser measurements can be used to filter out spurious sonar measurements [Jörg, 1995; Vandorpe *et al.*, 1996; Dudek *et al.*, 1996] or vice versa [Kim and Cho, 2001]. Then measurements can be stored in an occupancy map [Vandorpe *et al.*, 1996; Enderle *et al.*, 1999], or features such as lines and corners can be extracted and combined as in [Dudek *et al.*, 1996]. Alternatively, belief in the existence of line features stemming from laser measurements can be reinforced using sonar measurements as in [Kim and Cho, 2001]. Line features can also be extracted from an occupancy map generated using both sensing modalities [Jörg, 1995].

This chapter presents results of simultaneous localization and mapping using advanced sonar and laser range measurements. The advanced sonar arrays are capable of measuring range and bearing with a standard error of 0.1° and 0.2mm respectively, and classifying sensed targets into *planes*, *right angle corners* and *edges* [Kleeman, 2002]. The advanced sonar aids laser line segmentation by detecting small edge and corner features not seen by the laser due to the larger range quantization and error of the laser. For example doors that are not flush with a corridor are often merged with the corridor line segment with laser alone, however doorjambs are easily detectable to the advanced sonar as edges and small corner features, thus providing a cue for line segmentation in laser measurements. Planes and corners measured by laser and sonar are fused, achieving greater robustness and accuracy. Typically the advanced sonar range and bearing measurements are more accurate within a 5 meter range, whilst laser provides measurements along a line segment rather than just at the normal point as is the case for sonar. The line and corner fitting approaches and error models developed in chapter 2 are used in this chapter in the fusion process.

The sparse feature map created by SLAM provides accurate localization, but global path planning and obstacle avoidance usually require an accurate occupancy grid representation of the environment. Many occupancy grid based map building and localization schemes using laser range measurements have been proposed in previous work. A common approach is to first generate a laser scan map by matching individual scans collected during map building [Lu, 1995; Gutmann and Konolige, 1999]. The laser scan map is easily converted into a grid map for path planning, and localization is again provided by laser scan matching. An important

Figure 3.1: The advanced sonar array with panning mechanism.

issue in the scan matching approach to map building is sensitivity to people and other transient moving objects. This can been addressed by identifying moving objects and removing them from the scan [Hähnel *et al.*, 2003b].

Bourgault *et al.* [2002] use a similar mapping scheme described in this chapter, involving two maps: a feature map for accurate localization generated using SLAM, and an occupancy grid for planning and exploration. The occupancy grid is generated by aligning each new laser scan using the current pose of the robot estimated by SLAM. However, the main drawback of this approach is that the SLAM feature map is continuously corrected as features are re-observed, causing the previously estimated path of the robot to become invalid. The result is a smearing of the occupancy grid with reduced certainty about the occupancy of each cell. In this chapter a solution to this problem is proposed by storing the robot path and laser scans with respect to neighboring SLAM features and delaying the generation of the occupancy grid until it is needed. With this approach, the generated occupancy grid will always be consistent with the current SLAM map.

The work presented in this chapter is published as [Diosi and Kleeman, 2004] and [Diosi *et al.*, 2005].

3.2 Advanced Sonar Array

The advanced sonar array developed by Kleeman [2002; 2001] is shown in fig. 3.1. It employs 4 Polaroid transducers as sensing elements. From the 4 transducers the top 2 are used as transmitters and the bottom 2 as receivers. The sonar operates as follows. The top right

transducer fires a short pulse which is followed by a pulse from the left transducer. The time between the two pulses changes randomly with each measurement. Since different advanced sonar arrays likely use different time separation at any given moment, crosstalk rejection is a matter of ignoring echoes with unmatched time separations. After amplification, the signals from the bottom receivers are fed into an analog to digital converter. Outputs from the analog to digital converter are processed in a digital signal processor which sends its time stamped advanced sonar measurements through a serial port to a robot's PC.

The 2 pulses emitted by the transmitters are received as 4 echoes in the receivers. The arrival times of the echoes are determined by matched filtering where the echoes are cross correlated with templates for different arrival angles. From the 4 arrival times and from the known transmitter and receiver geometry, bearings of the returns corresponding to the transmitted pulse from the left and from the right transmitter are calculated. Classification of targets is based on these 2 bearings. If the pulse from the left transmitter is observed on the left and the pulse from the right transmitter is observed on the right then the target is classified as a plane. If the left pulse is observed on the right and vice versa, then the target is classified as a right angle corner. If both transmitted pulses are observed at the same bearing then the target is classified as an edge. Classification of targets works up to 5m.

The firing rate of the advanced sonar arrays is about 25 to 30 cycles per second. The standard deviation of bearing measurements was measured to be $\sigma_{bearing} \approx 0.1^\circ$ in laboratory settings. The standard deviation of range measurements was measured to be $\sigma_{range} \approx 0.2mm$ in laboratory settings. The sensitivity of the sonar sensor is best characterized by the fact that walls are often represented three times in the measurements where the furthest two are due to multiple reflections of the transmitted pulse between the wall and the robot. Small right angled reflectors formed from 1cm moldings on the walls of the corridors are classified as edge reflectors. Two advanced sonar arrays are mounted on the robot on panning mechanisms which are swept back and forth continuously.

The error model adopted for the advanced sonar arrays is derived simply from random white Gaussian noise added to bearing measurements and to the speed of sound. The standard deviations of these errors are taken as 2° for bearing and 2% of the speed of sound and were chosen to account for changes in air temperature, air turbulence, unmodeled robot vibrations and association errors. Note that even if errors caused by changes in the air temperature and incorrect associations are not random, they may be modeled as such if measurements are taken from random positions.

3.3 Sonar and Laser Synergy

The benefits of combining advanced sonar and laser measurements are as follows:

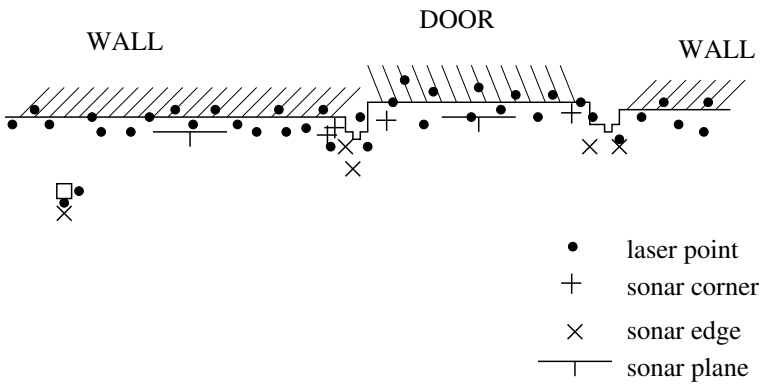


Figure 3.2: Laser scan segmentation aided by sonar. Note the sonar measurements are offset from the wall for clarity only.

- *Increase in the accuracy of measurements.* The greater the number of measurements fused, usually the more accurate are the results.

- *Enhancement in measurement reliability.* If the same feature is observed by two sensing modalities, the existence of that feature is more likely.

- *More robust laser segmentation.* Due to the limited range and bearing resolution of lasers (especially the Sick PLS), measurements from different features can be erroneously segmented into one line segment. However by using advanced sonar observations, the correctness of segmentation can be improved. This can avoid some systematic measurement errors due to incorrect laser associations.

- *Removal of specular reflections.* Using information from laser scans, phantom multiple reflection sonar measurements can be removed more easily compared to using sonar alone.

The following scheme is chosen for sonar and laser synergy: sonar aids laser segmentation; laser helps to remove spurious sonar measurements and to select good sonar point features. Laser and sonar features of the same object are fused and statistics are kept for each feature as to their sensor type contribution.

While combining laser and sonar measurements, it is assumed that sonar and laser are observing the same features, the speed of sound used in the calculations is reasonably accurate and the robot short term odometry is reasonably accurate.

3.3.1 Segmentation

For each laser scan, sonar readings 6 second before and 6 seconds after a laser scan are transformed into the laser coordinate frame using odometry information. This time window is

chosen to include one full sweep of each sonar and therefore provide sonar coverage for the whole laser scan.

In the next step sonar readings are clustered. Clusters distant from laser readings are tagged as invisible and are not processed further with this scan.

In the laser segmentation step, each point not already part of a line forms the start of new line segment. The segment grows until any of the following conditions is satisfied:

- There is a discontinuity in the scan.
- The standard deviation of point distances from the line exceeds a set threshold.
- The distance of the next 3 points from the line exceeds a threshold derived from a range error model.
- The range of a point is further than a threshold. This is to stop fitting lines to measurements which are too far from the robot.
- The end of the 180° scan is reached.
- There is a cluster of sonar readings containing corners or edges in the vicinity.

A completed line segment containing more than a set number of points is accepted as a valid line segment and the conditions under which it was completed are stored. The start condition of the next line inherits the terminating condition of the previous line candidate . Line segment end conditions play an important role in the SLAM implementation described later.

Sonar clusters containing corners and edges that terminate line segments are good candidates for point features. They are especially good if they are grouped close together, and if they are close to the line. Sonar readings closest to the centers of a such clusters are selected as good point features. Such point features ease association, and reduce the number of features in the map by discarding isolated targets such as chair legs.

In the next phase, extracted laser lines that are separated by a sonar corner are checked if they constitute a right angle corner. If so, a corner is fitted to the raw data points with the approach described in section 2.4. The number of sonar corners are then stored with the laser corner and vice versa.

The number of sonar features confirming the existence of each laser line and corner feature is kept to provide a measure of reliability of features. At the end of the scan processing, all laser lines, corners and their uncertainties are transformed into the robot's reference frame.

An example for sonar laser segmentation is depicted in fig. 3.2, where there is a wall, a door and table leg shown as a square. The line growing starts on the righthand side and stops at the right door frame due to sonar edge and corner returns. This line is rejected since it consists of too few line points. A new line is grown between the right and left door frames and is accepted

since it contains enough points. The next valid line is grown from the left frame until the table leg where there is a sonar edge and a range discontinuity. Other segmenting methods such as Split and Merge [Ballard and C. M, 1982] or RANSAC [Fischler and Bolles, 1981] would probably have segmented each scan point except those associated with the table leg as one segment. Our segmentation approach is conservative since it assumes that each line segment belongs to a different object. This assumption reduces the chance of systematic measurement errors in SLAM due to incorrect segmentation. For example let us assume in the map there is a line feature entailing both parts of walls and an indented door. If an observation contains due to obstruction only the door and part of one wall, then the estimated line will have a different angle since the ratio of wall points and indented door points is changed. The different line angle results in a systematic error component in the innovation error.

This segmentation approach can result in a large number of map features, however line merging is allowed in the SLAM implementation as described later and the table leg in fig. 3.2 will not have a lasting effect on the map.

3.3.2 Fusion

A laser line segment or corner having a corresponding sonar line or corner are fused if the sonar reading's time stamp is closest to that particular laser scan's time stamp. Fusion is done in a Kalman filter fashion, where each measurement is weighted by its certainty:

$$\mathbf{z_f} = \mathbf{A}\mathbf{x_l} + \mathbf{B}\mathbf{x_s} \tag{3.1}$$

where $\mathbf{z_f}$ is the fused measurement, $\mathbf{x_l}$, $\mathbf{C_l}$ are the laser measurement and covariance matrix, $\mathbf{x_s}$, $\mathbf{C_s}$ are the sonar measurement and covariance matrix and

$$\mathbf{A} = \frac{\mathbf{C_l^{-1}}}{\mathbf{C_l^{-1}} + \mathbf{C_s^{-1}}}, \quad \mathbf{B} = \frac{\mathbf{C_s^{-1}}}{\mathbf{C_l^{-1}} + \mathbf{C_s^{-1}}}. \tag{3.2}$$

The estimated covariance matrix of the fused feature $\mathbf{x_f}$ is calculated as:

$$\mathbf{C_f} = \mathbf{A}\mathbf{C_l}\mathbf{A^T} + \mathbf{B}\mathbf{C_s}\mathbf{B^T} \tag{3.3}$$

Note that unlike the corners extracted from the laser measurements, the advanced sonar array corner measurements do not contain orientation information. The problem of laser corner measurements having an extra dimension compared to sonar corner measurements is solved by appending the average corner orientation of $-\frac{\pi}{2}$ to the sonar corner measurements. The sonar corner covariance matrix is also extended to 3x3 with a large diagonal element. Because the covariance of the virtual orientation of the sonar corner is much larger than the covariance of the orientation of the laser corner, in the fusion process the sonar corner orientation is

neglected. Another approach for the fusion of laser and sonar corners would be to fuse only the range and bearing of corner and then append orientation to the result afterward. The advantage of the implemented approach over the second approach is that with the implemented approach, the correlations of the laser corner orientation with respect to range and bearing are not discarded.

3.4 Odometry

Mounted on each of SLAMbot's (see section 3.8.1) driving wheels are optical encoders with a resolutions of 2000 steps per revolution that are sampled every 10ms. The following equations describe the state update of odometry $\mathbf{x} = [x\ y\ \theta]^T$:

$$x_{k+1} = x_k - \frac{L_r + L_l}{2}\sin(\theta_k + \frac{L_r - L_l}{2D}) \tag{3.4}$$

$$y_{k+1} = y_k + \frac{L_r + L_l}{2}\cos(\theta_k + \frac{L_r - L_l}{2D}) \tag{3.5}$$

$$\theta_{k+1} = \theta_k + \frac{L_r - L_l}{D} \tag{3.6}$$

where $L_r,\ L_l$ are distances traveled by the right and left wheel and D denotes the wheelbase. In this error model as in [Kleeman, 2002], it is assumed, that the sources of odometry errors are random white nose added to the wheel separation and to distances traveled by the wheels. The odometry covariance matrix $\mathbf{P}_k$ is propagated in the conventional way:

$$\mathbf{P_{k+1}} = \frac{\partial \mathbf{x}_{k+1}}{\partial \mathbf{x}_k}\mathbf{P_k}\left(\frac{\partial \mathbf{x}_{k+1}}{\partial \mathbf{x}_k}\right)^T + \frac{\partial \mathbf{x}_{k+1}}{\partial \mathbf{u}_{k+1}}\mathbf{G_{k+1}}\left(\frac{\partial \mathbf{x}_{k+1}}{\partial \mathbf{u}_{k+1}}\right)^T \tag{3.7}$$

where $\mathbf{u_{k+1}} = [L_r\ L_l\ D]^T$, and

$$\mathbf{G_{k+1}} = \begin{bmatrix} K_R^2 L_r^2 & 0 & 0 \\ 0 & K_L^2 L_l^2 & 0 \\ 0 & 0 & K_D^2 D^2 \end{bmatrix} \tag{3.8}$$

Model parameters $K_R,\ K_L,\ K_D$ were chosen as $0.1, 0.1$ and 0.2. These parameters were chosen to generate an orientation error with 10° standard deviation, an along-path error with 4cm standard deviation and a cross path error with 60cm standard deviation after the robot travels 10m on a straight path.

3.5 EKF SLAM in General

Mapping whilst using the map being created for localization in the literature is referred to as simultaneous localization and mapping (SLAM). There are a number of ways for performing SLAM such as using an extended Kalman filter (EKF) [Leonard *et al.*, 1992], a sparse extended information filter (EIF) [Thrun *et al.*, 2002], particle filters for example FastSLAM [Montemerlo, 2003], set membership approach [Marco *et al.*, 2001], expectation maximization [Thrun *et al.*, 1998] or a biologically inspired framework (RatSLAM) [Milford *et al.*, 2004]. Since this thesis contributes only to the application of SLAM and not to the SLAM algorithm itself, a further discussion of the different SLAM algorithms will be omitted. In this thesis, from all the SLAM variants the extended Kalman filter version is chosen for its simplicity and because it is well documented.

The following equations as written in [Davison, 1998] describe the extended Kalman filter. In the prediction step the state $\hat{\mathbf{x}}(k+1|k)$ and covariance estimate $\mathbf{P}(k+1|k)$ at step $k+1$ is calculated based on the state and state covariance estimate at step k $\hat{\mathbf{x}}(k|k)$ resp. $\mathbf{P}(k|k)$, the dynamics of the system $\mathbf{f}(\hat{\mathbf{x}}(k|k),\mathbf{u}(k))$ and the input $\mathbf{u}(k)$:

$$\hat{\mathbf{x}}(k+1|k) = \mathbf{f}(\hat{\mathbf{x}}(k|k),\mathbf{u}(k)) \tag{3.9}$$

$$\mathbf{P}(k+1|k) = \frac{\partial \mathbf{f}}{\partial \mathbf{x}}(k|k)\mathbf{P}(k|k)\frac{\partial \mathbf{f}}{\partial \mathbf{x}}^T(k|k)+\mathbf{Q}(k) \tag{3.10}$$

where $\mathbf{Q}(k)$ is the covariance matrix of the process noise. The process noise is assumed to be zero mean white Gaussian noise representing errors due to uncertainty in $\mathbf{u}(k)$, in the system parameters, unmodeled dynamics etc.

When a measurement containing information about the state is available, then the state $\hat{\mathbf{x}}(k+1|k)$ and state covariance estimate $\mathbf{P}(k+1|k)$ can be updated as:

$$\hat{\mathbf{x}}(k+1|k+1) = \hat{\mathbf{x}}(k+1|k)+\mathbf{W}(k+1)\nu(k+1) \tag{3.11}$$

$$\mathbf{P}(k+1|k+1) = \mathbf{P}(k+1|k)-\mathbf{W}(k+1)\mathbf{S}(k+1)\mathbf{W}^T(k+1) \tag{3.12}$$

Where the state vector is updated by the Kalman gain $\mathbf{W}(k+1)$ multiplied by the innovation $\nu(k+1)$. The state covariance is reduced by the product of the Kalman gain and innovation covariance $\mathbf{S}(k+1)$. The innovation $\nu(k+1)$ is the difference between measurement $\mathbf{z}(k+1)$ (having zero mean white Gaussian noise with covariance $\mathbf{R}(k)$) and the predicted measurement $\mathbf{h}(\hat{\mathbf{x}}(k+1|k)$ based on the state estimate $\hat{\mathbf{x}}(k+1|k)$. The Kalman gain $\mathbf{W}(k+1)$ is calculated as:

$$\mathbf{W}(k+1) = \mathbf{P}(k+1|k)\frac{\partial \mathbf{h}^T}{\partial \mathbf{x}}(k|k)\mathbf{S}^{-1}(k+1) \tag{3.13}$$

The innovation covariance $\mathbf{S}(k+1)$ expressing the measurement and measurement prediction

residual covariance is calculated as:

$$\mathbf{S}(k+1) = \frac{\partial \mathbf{h}}{\partial \mathbf{x}}(k|k)\mathbf{P}(k+1|k)\frac{\partial \mathbf{h}^T}{\partial \mathbf{x}}(k|k) + \mathbf{R}(k) \tag{3.14}$$

For a deeper understanding of Kalman filters the reader is referred to [Bar-Shalom and Li, 1993].

The EKF SLAM described in [Davison, 1998] takes advantage of the structure of the SLAM problem to simplify some of the equations. The state vector $\hat{\mathbf{x}}$ consists of vector $\hat{\mathbf{x}}_v$ defining the robot pose $(x, y, \theta)^T$ and $\hat{\mathbf{y}}_i$ representing map features:

$$\hat{\mathbf{x}} = \begin{bmatrix} \hat{\mathbf{x}}_v \\ \hat{\mathbf{y}}_1 \\ \hat{\mathbf{y}}_2 \\ \vdots \end{bmatrix} \tag{3.15}$$

Given the structure of $\hat{\mathbf{x}}$, the state covariance $\mathbf{P}$ is then partitioned as:

$$\mathbf{P} = \begin{bmatrix} \mathbf{P}_{xx} & \mathbf{P}_{xy_1} & \mathbf{P}_{xy_2} & \cdots \\ \mathbf{P}_{y_1x} & \mathbf{P}_{y_1y_1} & \mathbf{P}_{y_1y_2} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \tag{3.16}$$

In the prediction step, the map features stay unchanged, therefore the new state vector is calculated as:

$$\hat{\mathbf{x}}(k+1|k) = \begin{bmatrix} \mathbf{f}_v(\hat{\mathbf{x}}_v(k|k), \mathbf{u}(k)) \\ \hat{\mathbf{y}}_1(k|k) \\ \hat{\mathbf{y}}_2(k|k) \\ \vdots \end{bmatrix} \tag{3.17}$$

where $\mathbf{f}_v$ typically represent the odometry equations and $\mathbf{u}$ represent the wheel encoder outputs for a mobile robot. Just like the state vector, not all the elements of the covariance matrix need to be changed:

$$\mathbf{P}(k+1|k) = \begin{bmatrix} \frac{\partial \mathbf{f}_v}{\partial \mathbf{x}_v}\mathbf{P}_{xx}(k|k)\frac{\partial \mathbf{f}_v}{\partial \mathbf{x}_v}^T + \mathbf{Q}(k) & \frac{\partial \mathbf{f}_v}{\partial \mathbf{x}_v}\mathbf{P}_{xy_1}(k|k) & \frac{\partial \mathbf{f}_v}{\partial \mathbf{x}_v}\mathbf{P}_{xy_2}(k|k) & \cdots \\ \mathbf{P}_{y_1x}(k|k)\frac{\partial \mathbf{f}_v}{\partial \mathbf{x}_v}^T & \mathbf{P}_{y_1y_1}(k|k) & \mathbf{P}_{y_1y_2}(k|k) & \cdots \\ \mathbf{P}_{y_2x}(k|k)\frac{\partial \mathbf{f}_v}{\partial \mathbf{x}_v}^T & \mathbf{P}_{y_2y_1}(k|k) & \mathbf{P}_{y_2y_2}(k|k) & \cdots \\ \vdots & \vdots & \vdots & \end{bmatrix} \tag{3.18}$$

where $\mathbf{Q}(k)$ is the estimated odometry error increment.

When a landmark is observed, one has to check first if it is a re-observation of an already mapped landmark or a new landmark. A measurement can be accepted as an observation of a

map landmark if the innovation normalized by the innovation covariance is smaller than a so called validation gate. The innovation covariance can be calculated as [Davison, 1998]:

$$\mathbf{S} = \frac{\partial \mathbf{h}}{\partial \mathbf{x}_v}\mathbf{P}_{xx}\frac{\partial \mathbf{h}}{\partial \mathbf{x}_v}^T + \frac{\partial \mathbf{h}}{\partial \mathbf{x}_v}\mathbf{P}_{xy_i}\frac{\partial \mathbf{h}}{\partial \mathbf{y}_i}^T + \frac{\partial \mathbf{h}}{\partial \mathbf{y}_i}\mathbf{P}_{y_i x}\frac{\partial \mathbf{h}}{\partial \mathbf{x}_v}^T + \frac{\partial \mathbf{h}}{\partial \mathbf{y}_i}\mathbf{P}_{y_i y_i}\frac{\partial \mathbf{h}}{\partial \mathbf{y}_i}^T + \mathbf{R} \tag{3.19}$$

then the normalized innovation, called Mahalanobis distance is calculated as:

$$d = \nu^T \mathbf{S}^{-1} \nu \tag{3.20}$$

Note that the normalized innovation is χ^2 distributed with degrees of freedom equaling the number of elements is ν. Consequently if a validation gate of 6 is chosen for an innovation vector consisting of 2 elements, then 95% of otherwise correctly associated measurements will be accepted as correct associations.

If the measurement is associated with an existing map feature (or landmark), then the state and covariance can be updated as:

$$\hat{\mathbf{x}}_{new} = \hat{\mathbf{x}}_{old} + \mathbf{W}(\mathbf{z} - \mathbf{h}) \tag{3.21}$$

$$\mathbf{P}_{new} = \mathbf{P}_{old} - \mathbf{W}\mathbf{S}\mathbf{W}^T \tag{3.22}$$

If the measurement is accepted as a new feature $\mathbf{y}_i$, then the state vector and covariance matrix are modified as:

$$\mathbf{x}_{new} = \begin{bmatrix} \mathbf{x}_v \\ \mathbf{y}_1 \\ \mathbf{y}_2 \\ \vdots \\ \mathbf{y}_i \end{bmatrix} \tag{3.23}$$

$$\mathbf{P}_{new} = \begin{bmatrix} \mathbf{P}_{xx} & \mathbf{P}_{xy_1} & \cdots & \mathbf{P}_{xx}\frac{\partial \mathbf{y}_i}{\partial \mathbf{x}_v}^T \\ \mathbf{P}_{y_1 x} & \mathbf{P}_{y_1 y_1} & \cdots & \mathbf{P}_{y_1 x}\frac{\partial \mathbf{y}_i}{\partial \mathbf{x}_v}^T \\ \mathbf{P}_{y_2 x} & \mathbf{P}_{y_2 y_1} & \cdots & \mathbf{P}_{y_2 x}\frac{\partial \mathbf{y}_i}{\partial \mathbf{x}_v}^T \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial \mathbf{y}_i}{\partial \mathbf{x}_v}\mathbf{P}_{xx} & \frac{\partial \mathbf{y}_i}{\partial \mathbf{x}_v}\mathbf{P}_{xy_1} & \cdots & \frac{\partial \mathbf{y}_i}{\partial \mathbf{x}_v}\mathbf{P}_{xx}\frac{\partial \mathbf{y}_i}{\partial \mathbf{x}_v}^T + \frac{\partial \mathbf{y}_i}{\partial \mathbf{h}}\mathbf{R}\frac{\partial \mathbf{y}_i}{\partial \mathbf{h}}^T \end{bmatrix} \tag{3.24}$$

The limitations of using such a basic EKF implementation of SLAM are the following:

- The computational and memory requirements are proportional to the square of the map feature count. This problem can be alleviated through the use of local maps as in [Chong and Kleeman, 1999], or using the ATLAS framework of [Bosse *et al.*, 2004], or a com-

pressed Kalman filter [Guivant and Nebot, 2001]. The computational requirements can also be alleviated if one particular feature is updated many times sequentially by an approach described in [Davison, 1998].

- Kalman filter inconsistency problems due to linearizations can appear sooner than problems due to computational and memory requirements [Castellanos *et al.*, 2004]. The inconsistency problems manifest themselves in optimistic state error estimates. Large error in a robot's orientation estimate can speed up the appearance of such problems.

However, in the SLAM implementation described next, it was assumed that the number of landmarks is small enough so that the computational and memory issues do not manifest themselves and that the robot's orientation error can be kept so small that consistency problems do not cause failure of the SLAM process.

3.6 SLAM using Sonar and Laser

The SLAM algorithm is implemented in Matlab and in C++ using a simple extended Kalman filter version similar to that in [Davison, 1998]. For simplicity speed of sound and odometry parameters are not included into the filter and only line segments, corners and point features are used. In the current implementation all line segment features used either originate from the laser only or are the result of laser and sonar line fusion.

The core of equations needed for SLAM using lines segment, point and right angle features are summarized in appendix B.

3.6.1 Association

Association is implemented in a simple way using a validation gate. For line segments, the length of overlap is also checked. If there are two possible map feature candidates for a line segment measurement, with the measurement covering at least 40cm of both candidates, then line segment features are merged in the update process. If there are two or more possible candidates without sufficient overlap, the measurement is discarded. If no feature can be associated with the measurement, then the measurement is inserted into the map as new feature.

For point and corner measurements, the map features are checked first for possible candidates, based on the distance between measurement and map feature. The matched candidates are then checked using a validation gate. Similarly to line features if there are two or more possible candidates, the measurement is discarded. If no feature can be associated with the measurement, then the measurement is inserted into the map as new feature.

3.6.2 A Remark on Line Segment Endpoints

Line segment endpoints pose a problem in the implemented SLAM. At each SLAM update, the map can change its shape, rotate and translate. Unlike the line orientation and distance parameters, line endpoints are not included into the SLAM map, therefore their values are not changed automatically with each update. One solution would be to put all line segment ends into the SLAM map as well and set the correlation coefficients with the corresponding line parameters to 1. This would however increase the memory and computational requirements of SLAM.

An alternative that is computationally more efficient is presented here. Line segment endpoints are stored separately in Cartesian coordinates, and their position is changed upon re-observation. The observations endpoints are projected onto the corresponding line feature from the map. Then the line feature endpoints are:

- moved towards the observation endpoints in a large step if it is likely that the line ends there, e.g. the line is terminated with sonar edges or corners.
- moved towards the observation endpoints in a small step if the observation's endpoint is unreliable, for example due to termination of the line with poor fitting laser points and no sonar information.
- unchanged if measurement endpoint carries no information. For example if measurement endpoint is inside the map line segment, but was terminated because line went beyond the lasers 180° field of view.

Line segment endpoints are also changed with each state update. Each map line (α_i, d_i) is checked if the rotational component of the corresponding correction $(\Delta\alpha, \Delta d_i)$ is larger than a threshold. The value of this threshold was 0.05° in the results. For line segments undergoing larger orientation corrections, the intersection of line (α_i, d_i) with $(\alpha_i + \Delta\alpha, d_i + \Delta d_i)$ is sought:

$$x_c = \frac{d_i \sin(\alpha_i + \Delta\alpha_i) - (d_i + \Delta d_i)\sin\alpha_i}{\sin\Delta\alpha_i} \tag{3.25}$$

$$y_c = \frac{-d_i \cos(\alpha_i + \Delta\alpha_i) + (d_i + \Delta d_i)\cos\alpha_i}{\sin\Delta\alpha_i} \tag{3.26}$$

Then endpoints (x_e, y_e) are rotated around the point of intersection:

$$x_e = x_c + (x_e - x_c)\cos\Delta\alpha_i - (y_e - y_c)\sin\Delta\alpha_i \tag{3.27}$$

$$y_e = y_c + (x_e - x_c)\sin\Delta\alpha_i + (y_e - y_c)\cos\Delta\alpha_i \tag{3.28}$$

The equations for intersecting point (x_c, y_c) calculation become ill conditioned when the orientation corrections are small. The correction corresponding to such cases is the translation of

a line and its endpoints:

$$x_e = x_e + \Delta d_i \cos \alpha_i \tag{3.29}$$

$$y_e = y_e + \Delta d_i \sin \alpha_i \tag{3.30}$$

All line endpoints are at the end back projected onto the corrected line to constrain the effect of numerical errors:

$$\begin{bmatrix} x_e \\ y_e \end{bmatrix} = \begin{bmatrix} x_e \\ y_e \end{bmatrix} + \left(d_i + \Delta d_i - \begin{bmatrix} \cos(\alpha_i + \Delta\alpha_i) \\ \sin(\alpha_i + \Delta\alpha_i) \end{bmatrix}^T \begin{bmatrix} x_e \\ y_e \end{bmatrix} \right) \begin{bmatrix} \cos(\alpha_i + \Delta\alpha_i) \\ \sin(\alpha_i + \Delta\alpha_i) \end{bmatrix} \tag{3.31}$$

Note that the line endpoint corrections do not work when the map is corrected in the direction of the line. Since endpoints are corrected with each observation as well, the described shortcoming usually does not have a long lasting effect.

An another approach is to use neighboring features to track line segment endpoint movements [Kleeman, 2004], however this depends on the availability of sufficiently many close map features. The same idea is used in the next section for path tracking.

3.7 Occupancy Grid Generation and Path Tracking

3.7.1 Path Tracking

The sparse feature map created by SLAM provides accurate localization, but global path planning and obstacle avoidance (using distance transform based algorithms) require an accurate occupancy grid representation of the environment. If the pose of the robot in a static environment is accurately known at all times, generating an occupancy grid using laser scans without scan matching is a trivial problem. Next an approach for the accurate recovery of the path of the robot to generate such a grid is described.

Registration of laser scans into an occupancy grid using SLAM based on-the-fly estimation of the robot position was shown in [Bourgault *et al.*, 2002]. As noted in the introduction, problems with this approach arise when features are re-observed and the entire map is updated. For example, consider a robot travelling to the end of a long corridor and returning while suffering from odometry drift. As the robot re-observes map features near the initial position, the robot pose, feature location, and location of all correlated features in the map are updated. For a corridor of 40 meters in length, small angular corrections by the robot at one end may cause correlated features at the other end to be corrected by a shift of several meters. This can cause serious divergence between the feature map and occupancy grid.

To address this problem, the approach presented in this chapter involves the periodical

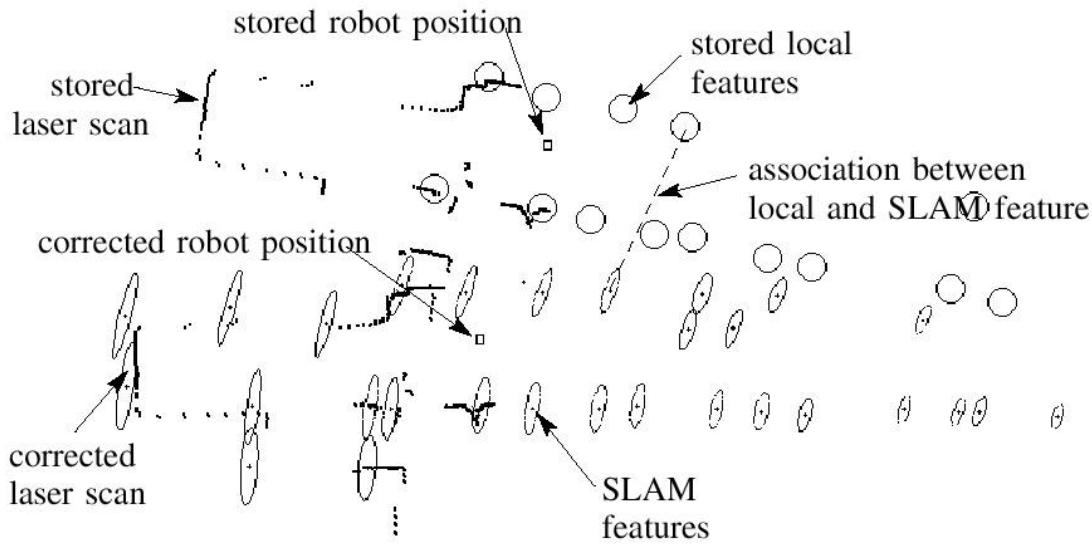


Figure 3.3: Path correction using scan matching with local features. Courtesy to Geoff Taylor for putting the annotation in the figure.

recording of robot poses and associated laser scans during mapping and delaying grid generation until the SLAM map has sufficiently converged so that the entire path can be corrected. One way to achieve this is to periodically store the robot pose in the Kalman filter state vector, so that the stored robot path is automatically corrected each time the map is updated [Garcia *et al.*, 2002]. However, increasing the size of the state vector has the undesirable effect of increasing the filter update time.

In this chapter, an alternative solution is proposed (illustrated in fig. 3.3) which was partially inspired by scan matching and by how Kleeman[2004] keeps track of line segment ends as described earlier. Based on the reasonable assumption that the SLAM map is always locally correct, the periodically stored robot poses are associated with local point features. Note that robot poses, laser scans and local point features are stored outside the SLAM state vector. As the map is updated, the path is also corrected provided the local configuration of point features does not change significantly. The scheme is implemented as follows: for every 1 meter of travel or change in orientation of the robot by 30°, the most recent laser scan is stored together with the position of point features in the local neighborhood, expressed in the local robot frame. Information about feature correspondence is also stored to ease association in the reconstruction stage. To recover the stored position of the robot at a later time, a process identical to scan matching is adopted. Assuming the correspondence between two or more local features (shown as circles in fig. 3.3) and map points (shown as error ellipses) is known, the equations described in [Lu and Milios, 1997] (Appendix C) are applied to recover the pose

of the robot in the SLAM map:

$$\theta_m = \arctan \frac{S_{xy'} - S_{yx'}}{S_{xx'} + S_{yy'}} \tag{3.32}$$

$$x_m = \bar{x}' - (\bar{x}\cos\theta_m - \bar{y}\sin\theta_m) \tag{3.33}$$

$$y_m = \bar{y}' - (\bar{x}\sin\theta_m + \bar{y}\cos\theta_m) \tag{3.34}$$

where (x_m, y_m, θ_m) is the recovered pose of the robot in the SLAM map and

$$\begin{array}{cc} \bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i, & \bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i \\ \bar{x}' = \frac{1}{n}\sum_{i=1}^{n} x'_i, & \bar{y}' = \frac{1}{n}\sum_{i=1}^{n} y'_i \\ S_{xx'} = \sum_{i=1}^{n}(x_i - \bar{x})(x'_i - \bar{x}'), & S_{yy'} = \sum_{i=1}^{n}(y_i - \bar{y})(y'_i - \bar{y}'), \\ S_{xy'} = \sum_{i=1}^{n}(x_i - \bar{x})(y'_i - \bar{y}'), & S_{yx'} = \sum_{i=1}^{n}(y_i - \bar{y})(x'_i - \bar{x}'). \end{array} \tag{3.35}$$

In (3.35) (x_i, y_i) express point features stored in the robot's coordinate frame and (x'_i, y'_i) express the corresponding SLAM map features in the SLAM coordinate frame. The accuracy of the result increases as more features are used, since the local configuration of points is likely to change between initially storing the path and the final SLAM map. In the results shown later the closest 10-20 landmarks were used.

3.7.2 Occupancy Grid Generation

After correcting the complete path of the robot in the SLAM map by applying the process described above, the registered laser scans can be assembled into an occupancy grid. The first step in grid generation is to find a suitable orientation of the SLAM map to minimize the required number of grid cells. This reduces both the memory requirements of the occupancy grid and the computational expense of segmentation and path planning. The approximate orientation is found numerically, by rotating the path from 0 to 90° in 5° steps and choosing the orientation that minimizes the area of a rectangular bounding box. This orientation defines the transformation from the SLAM map to the occupancy grid, which is then applied to the stored robot path and registered laser scans.

The occupancy grid is generated using a ray-casting method similar to the algorithm presented in [Veeck and Burgard, 2004]. All grid cells are initially set to zero (occupied), and free space is "carved" out by incrementing the count in each cell for every intersecting laser beam. To minimize the effect of orientation error, laser beams are truncated at a range of 7 meters. After processing all scans, cells with sufficiently high count (using a fixed threshold) are labelled as unoccupied. This approach to grid generation is computationally efficient: path correction and occupancy map generation for about 700 path points/laser scans requires only 1.4 seconds processing time on a P4 2.8GHz PC. This suggests the possibility of generating

on-the-fly grid maps during exploration.

As noted earlier, the main advantage of the described occupancy grid generation approach is to store the laser scans with respect to local SLAM features, and delay grid generation until it is needed. Thus, when the occupancy grid is eventually generated, it will always be consistent with the current SLAM map.

3.8 Experimental Results

All the experimental results presented next were performed using the robot SLAMbot.

3.8.1 SLAMbot

The testbed for all experiments described next is an in-house built differential drive robot called SLAMbot (see fig. 3.4). Sensing capabilities include odometry, two advanced sonar arrays and a Sick LMS laser range finder. The advanced sonar sensors are capable of measuring range and bearing with a standard error of 0.1° and 0.2mm respectively, and classifying sensed targets into *planes*, *right angle corners* and *edges* [Kleeman, 2002]. The sensors are mounted on panning mechanisms which are continuously swept back and forth in a 270° arc. The Sick LMS200 generates time-of-flight laser scans in a horizontal plane with 0.5° angular resolution and 1cm range resolution at 36Hz. Odometry is based on 2000 count wheel encoders on both driven wheels, which are counted on a custom-programmed Xilinx field programmable gate array (FPGA). Time-stamped encoder counts are reported at precise 10ms intervals. The FPGA also appends time-stamps to the Sick serial packets, which resolves time registration issues and eliminates the need for a hard real-time operating system. This FPGA based time stamping unit is an important part of SLAMbot since time registration errors can render otherwise accurate measurements inaccurate. The laptop in fig. 3.4 mounted on the robot is used only when performing interactive SLAM [Diosi *et al.*, 2005] where a tour guide shows the robot around.

The results of 3 experiments which were performed with SLAMbot are shown next. Note only line segments longer than 70cm are used in the SLAM results.

3.8.2 Experiment #1

Only in this experiment, SLAMbot was equipped with an old Sick PLS and the time stamping unit was not mounted on the robot yet. The robot was driven around the sonar robotics lab with a joystick, then out into the corridor. The robot was then driven from one end of the corridor to the other and back to the lab. Odometry, sonar and laser measurements were logged during the more than 150 meters travelled by the robot. During the experiment, half a dozen people walked past the robot, however no changes to the environment occurred such as closing a door.

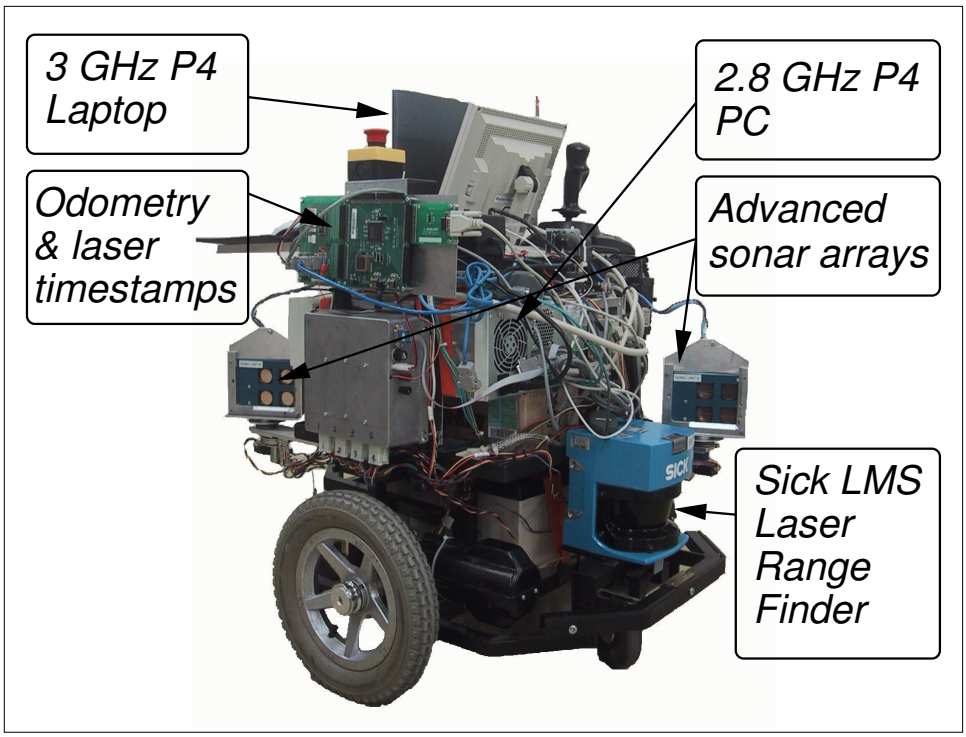


Figure 3.4: SLAMbot.

Only in this experiment, all calculations were done off-line in Matlab. In the rest of the experiments a C++ implementation of SLAM was used. The resulting 240 feature map is shown in fig 3.5. Dots in the figure represent corners and points. The robot managed to keep track of its position throughout the whole experiment. However the created map is not error free. At around (15m, 6m), due to a violation of the flat-floor assumption a line was included in the map across the corridor. The corridor is sloped at one place which caused the laser beams to reflect back from the floor.

For comparison, SLAM was run with features from the laser only using the same dataset, but the robot accumulated a large enough error in the direction of the corridor to get lost (see the double walls of the lab in fig. 3.6). Even though there are features on the corridor allowing localization in the direction of the corridor, these features were observed only when the robot was moving in one particular direction due to the 180° field of view of the laser. Therefore the sonar fusion with laser has prevented the divergence of SLAM.

SLAM was also run with the same dataset using only point landmarks from sonar and the robot kept track of its position. However the quality of the map measured in the curvature of the corridor was worse. It was found that for the robot to produce a map of the corridor using advanced sonar point features only with the same straightness as with SLAM using advanced sonar and laser, the robot had to traverse the corridor three time as many times.

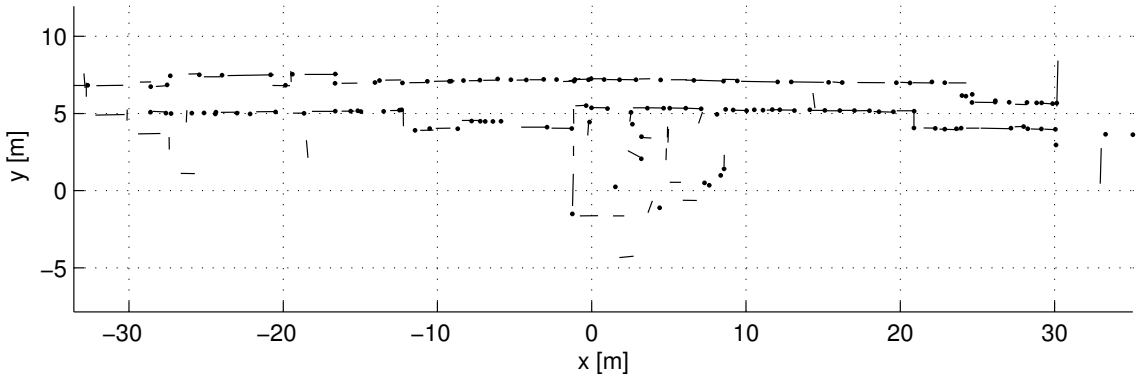


Figure 3.5: SLAM results on the corridor of building 36 using Sick PLS and advanced sonar arrays. Grid squares represent 10x10 meters.

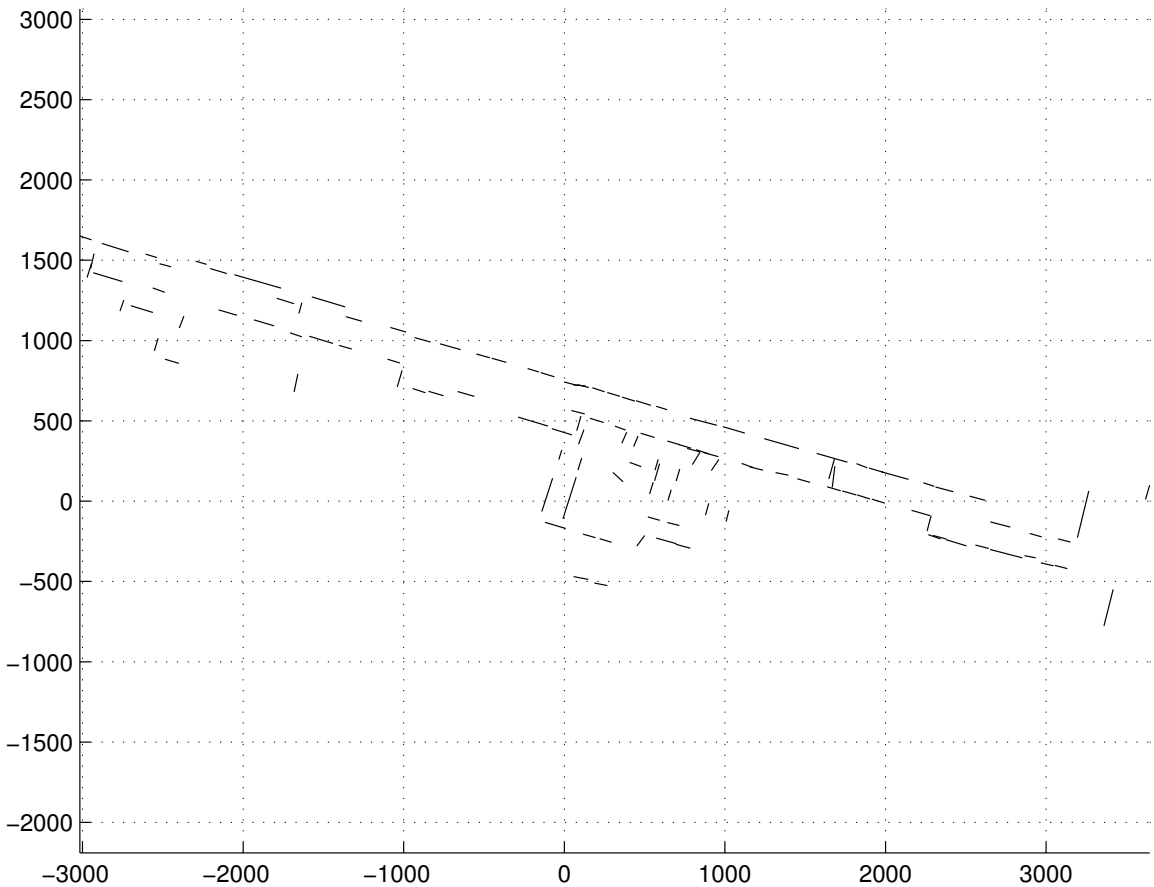


Figure 3.6: SLAM results on the corridor of building 36. Only lines segment from a Sick PLS were used therefore SLAM diverged which can be seen from the double walls of the lab. Grid squares represent 10x10 meters.

3.8.3 Experiment #2

Experiment #2 was conducted in collaboration with Geoffrey Taylor. More details about the experiment than described next are given in[Diosi *et al.*, 2005].

In this experiment, the newer Sick LMS 200 laser scanner was mounted on the robot instead of the older Sick PLS. The time stamping unit described earlier in section 3.8.1 was also in use. Instead of the Matlab code, a C++ implementation of the mapping software was running in real time on the robot.

The purpose of experiment #2 was to demonstrate the viability of interactive SLAM. The idea of interactive SLAM can be found in [Jensfelt, 2001] where Jenfelt describes a possible robot teaching scenario. In this scenario a human tour guide shows a mobile robot around in a building and gives the robot some information about each room. The first implementation of the idea is described in [Althaus and Christensen, 2003] where the robot does a simple topological mapping with no free space representation and the tour-guide communicates with the robot through a keyboard.

In our implementation, the process of teaching the robot can be described as follows. The tour-guide walks up to the robot, and gives the robot the command "Robot follow me!" into a microphone. Then as the guide walks, the robot follows the guide by tracking the guide's legs in the laser range finder measurements. While in motion the robot builds a SLAM map of its environment using the advanced sonar and laser SLAM described earlier. As the guide reaches different rooms or areas of interests, he gives the robot information about the name of the location for example "Robot we are in the lab" or "Robot we are in the corridor", etc. Each time the robot receives such information from the guide, it stores the guide's location with respect to the SLAM map. These locations are later used in the occupancy grid segmentation process. When the guide intends to finish the tour, he issues the robot with the command "Robot, mapping complete". At this moment the robot switches from SLAM into localization mode, finalizes the occupancy grid by using the path tracking algorithm described earlier, segments the free space of the occupancy grid into coherent areas based on the commentary of the guide during the tour and gets ready to receive further commands. After the mapping is finished, users can send the robot to the center of previously visited rooms by issuing commands such as "Robot, go to the lab", "Robot, go to the office".

The implemented competencies of the robot were grouped into three programs or layers. In the lowest layer, sensor measurements are collected and distributed to other programs. Also a 10x10m local map of the robot's environment is maintained for local path planning purpose. Other programs can connect to the lowest layer and request the robot to move to positions within the local map. Path planning on this lowest level is performed using distance transform [Jarvis, 1985] at 2Hz. The speed of the robot is adjusted in a loop running at 5Hz so that time to collision with other objects is kept longer than 3s.

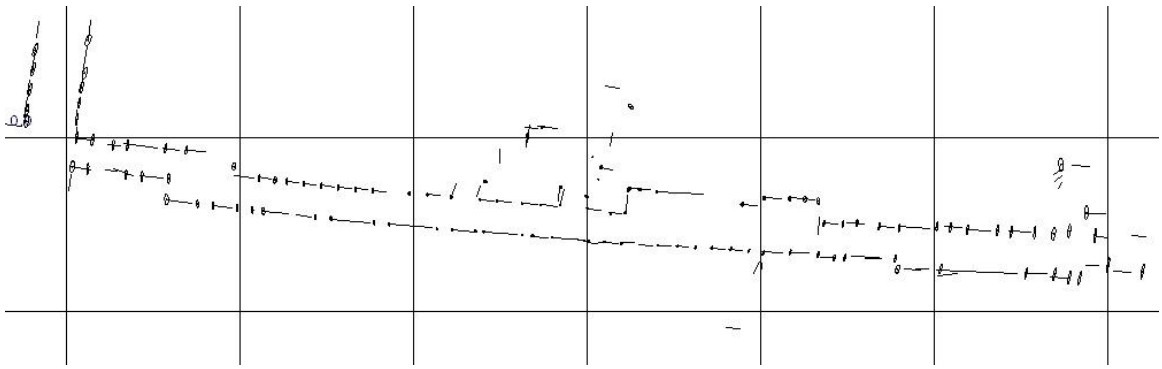


Figure 3.7: Slam results on a corridor of building 36, using Sick LMS and advanced sonar arrays. Grid squares represent 10x10 meters.

The middle level is represented with the described SLAM program which receives sensor measurements from the lowest level and provides robot pose updates or occupancy grids to requesting programs.

The highest level has the following competencies: recognition and execution of the guide's voice commands, person tracking and providing the lowest level with the coordinates of the person to follow, segmentation of the occupancy grid into rooms and global path planning using path transform [Zelinsky, 1994]. The highest level receives sensor measurements from the lowest level, robot positions and occupancy grids from the middle level and sends local goals to the lowest level.

In experiment #2 the robot executed a similar path to experiment #1, except it did not drive in a loop around the sonar robotics lab located in the middle of the resulting map, because it was physically partitioned into two rooms.

The features in the final SLAM map are shown in fig. 3.7; the slight curvature in the corridor is most likely due to systematic odometry errors. Figure 3.8 plots the stored laser scans as recorded at the robot positions provided by SLAM during the experiment, overlaid on the final SLAM map. Clearly, the laser scans no longer align with the SLAM map, as the SLAM map changes when features are re-observed. If the robot had visited each location more than only once, the walls would therefore appear in the laser data multiple times. Figure 3.8 clearly demonstrates the need for path correction.

Figure 3.9 shows the laser scans overlaid on the corrected robot path (using the matching scheme described in section 3.7.1) which demonstrates a significant improvement over fig. 3.8. Importantly, the laser scans are now consistent with the final SLAM map. Finally, fig. 3.10 shows the occupancy grid segmented into rooms with a path generated for the request of the guide, which was successfully executed.

In the collaborative experiment, the author of this thesis was not involved in voice recognition, person tracking and occupancy grid segmentation.

Figure 3.8: Laser scans overlaid on the map from fig. 3.7 using robot pose estimate from SLAM. Grid squares represent 10x10 meters.

Figure 3.9: Laser scans overlaid on the map from fig. 3.7 using robot pose stored with respect to SLAM map. Grid squares represent 10x10 meters.

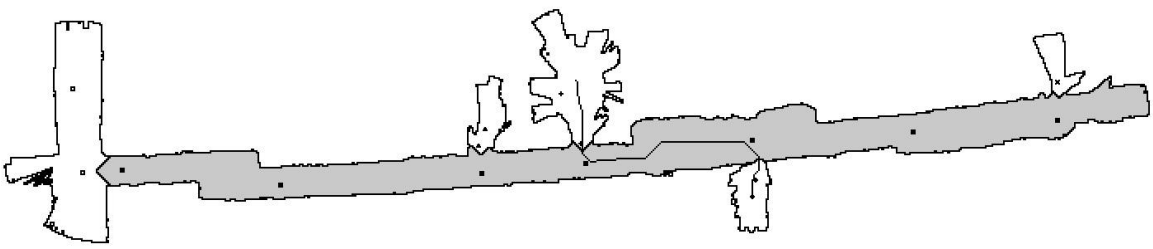


Figure 3.10: Segmented occupancy grid generated from the overlaid scans of fig. 3.9. A planned path is also shown in the figure. Courtesy to Geoff Taylor for the segmented rendering of the occupancy grid.

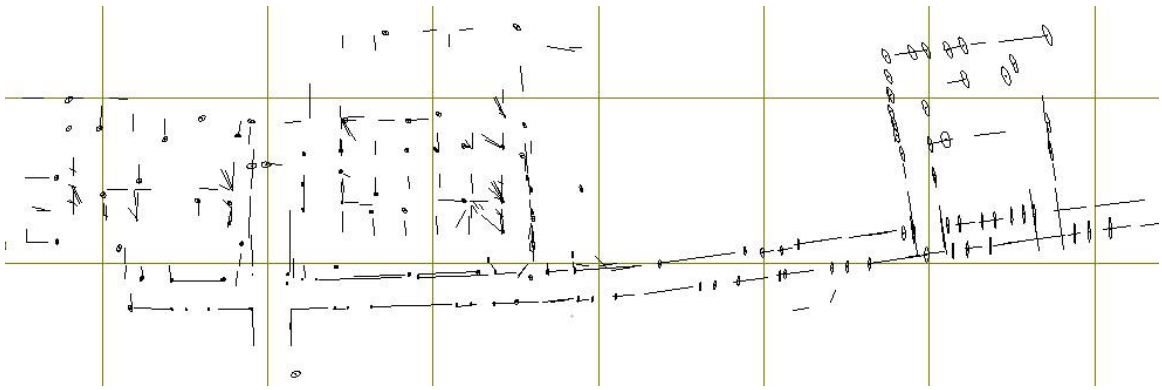


Figure 3.11: SLAM results on a corridor of building 72, using Sick LMS and advanced sonar arrays. Grid squares represent 10x10 meters.

3.8.4 Experiment #3

The environment where the robot was joysticked around in the third experiment is best described using fig. 3.13. The structures in the middle of the two rooms on the left are office cubicles. The third room is a seminar room filled with tables and chairs. The third room was not fully mapped because the tables and chairs prevented the robot from accessing the whole room. In fig. 3.13 the legs of chairs and tables are represented with a dense laser scan point cloud. The robot started from the corridor intersection between the 2 rooms on the left. It visited the left room, and after one loop, it proceeded through the corridor to the middle room where it performed a large and a small loop and continues to the third (seminar) room. In the third room the robot was twice driven over a 1.5cm high cable protector on the floor at 40cm/s and at 20cm/s speed. After the visit to the third room the robot returned to its initial location from which it traveled to the far end of the corridor, went around a loop and came back. During the traversal of the environment, no less than 10 people walked in the view of the laser scanner and some doors were opened and closed. Considering the presence of walking people, repetitive cubicles, long corridors and the 2 collisions with an obstacle on the floor, this dataset is not the most ideal for mapping.

The SLAM results from playing back logged data are shown in fig. 3.11. The bend in the corridor was likely caused by the orientation error introduced when bumping over the cable protector. Figure 3.13 shows the laser scans overlaid on the corrected robot path (using the matching scheme described in section 3.7.1) which demonstrates a significant improvement over fig. 3.12.

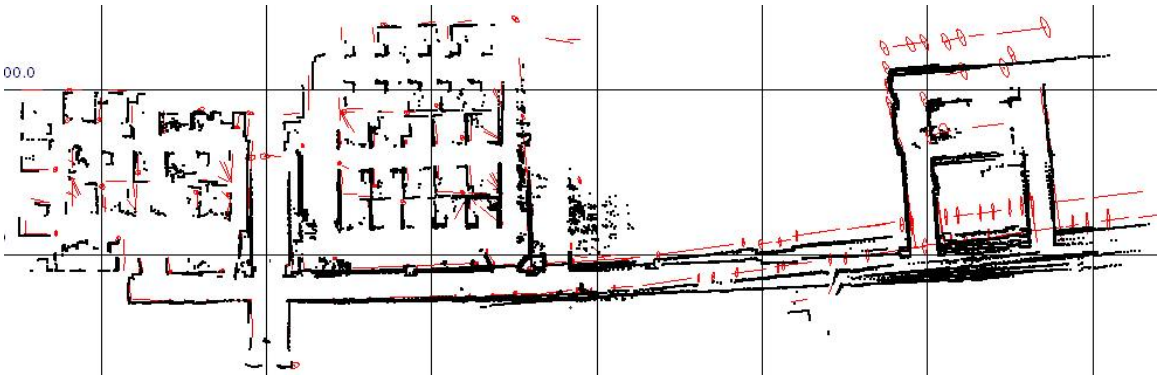


Figure 3.12: Laser scans overlaid on the map from fig. 3.11 using robot pose estimate from SLAM. Grid squares represent 10x10 meters.

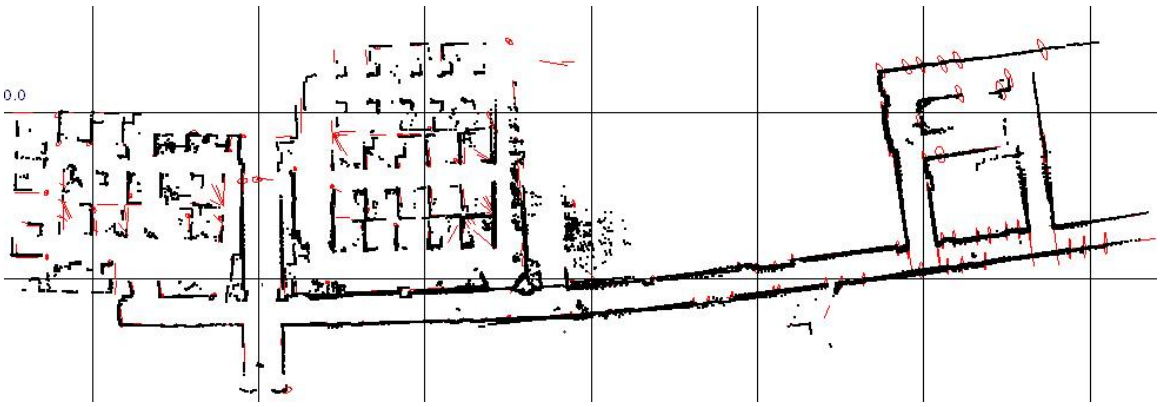


Figure 3.13: Laser scans overlaid on the map from fig. 3.11 using robot pose stored with respect to SLAM map. Grid squares represent 10x10 meters.

3.9 Conclusions

This chapter presents SLAM using advanced sonar with a laser range finder. In this chapter the successful fusion of lines and corners measured with sonar and laser is also demonstrated. The synergistic properties of sonar and laser measurements have been exploited in this work to aid laser line and corner segmentation with advanced sonar readings. Laser measurements, on the other hand simplify and improve the selection of reliable sonar point features and assist in removing multiple reflection sonar phantom features. It has been shown, that SLAM with fused sonar and laser measurement can succeed there where SLAM with laser line segments only diverges.

In this chapter the use of feature based SLAM for occupancy grid generation is also discussed. During SLAM, the location of the robot and an associated laser scan are periodically recorded, along with several local features in the current SLAM map. The path of the robot can be recovered later by matching the stored local features to points in the final SLAM map using a modification of the laser scan matching algorithm. An occupancy grid consistent with the SLAM map is recovered by overlaying the laser scans on the corrected robot path. Experimental results have demonstrated the necessity of path correction, and verify that the described approach generates accurate occupancy grids.

The approach described in this chapter is not the only way to generate consistent occupancy grids for path planning. As described in the next chapter consistent laser scan maps used in occupancy grid generation processes can be also constructed using SLAM with scan matching. Features in this case are the robot poses themselves, which are stored with corresponding laser scans. To perform SLAM with scan matching efficiently, a fast scan matching method is necessary. The polar scan matching approach of the next chapter fulfills this requirement.

Chapter 4

Scan Matching in Polar Coordinates

In the previous chapter line segments and right angle corners were extracted from laser measurements followed by their fusion with advanced sonar measurements. These line and corner features with additional advanced sonar point features were used in Kalman filter SLAM. In this chapter however SLAM is performed without the extraction of any features from the laser measurements. Instead of feature extraction, laser measurements are used together with scan matching to create maps. In the previous chapter path tracking was introduced to enable the generation of a consistent set of laser scans for occupancy grid generation during the feature based SLAM process. In this chapter, since landmarks in the map are previous robot poses, a consistent set of laser scans is given for free as the by-product of the SLAM process.

However the main focus of this chapter is not on SLAM but on laser scan matching itself. Laser scan matching approaches which match individual points of two scans are attractive since they do not require any interpretation of the laser measurements. Understanding and taking advantage of the structure of laser scans is important because it enables reduction of computations necessary for scan matching. The fast scan matching approach described in this chapter works in polar coordinate frames of 2D laser scanners utilizing a rotating mirror and harvests the benefits of the native structure of laser scans.

4.1 Introduction

Localization and map making is an important function of mobile robots. One possible way to assist with this functionality is to use laser scan matching. In laser scan matching, the position and orientation or pose of the current scan is sought with respect to a reference laser scan by adjusting the pose of the current scan until the best overlap with the reference scan is achieved. In the literature there are methods for 2D and 3D scan matching. This chapter restricts discussion to 2D laser scan matching.

Scan matching approaches can be local [Lu and Milios, 1997] or global [Tomono, 2004].

When performing local scan matching, two scans are matched while starting from an initial pose estimate. When performing global scan matching the current scan is aligned with respect to a map or a database of scans without the need to supply an initial pose estimate. Scan matching approaches also can be categorized based on their association method such as feature to feature, point to feature and point to point. In feature to feature matching approaches, features such as line segments [Gutmann, 2000], corners or range extrema [Lingemann *et al.*, 2004] are extracted from laser scans, and then matched. Such approaches interpret laser scans and require the presence of chosen features in the environment. In point to feature approaches, such as one of the earliest by Cox [1991], the points of a scan are matched to features such as lines. The line features can be part of a predefined map. Features can be more abstract as in [Biber and Straßer, 2003], where features are Gaussian distributions with their mean and variance calculated from scan points falling into cells of a grid. Point to point matching approaches such as the approach presented in this chapter, do not require the environment to be structured or contain predefined features.

Examples of point to point matching approaches are the following: iterative closest point (ICP), iterative matching range point (IMRP) and the popular iterative dual correspondence (IDC). Besl and Mac Kay [1992] proposed ICP, where for each point of the current scan, the point with the smallest Euclidean distance in the reference scan is selected. IMPR was proposed by Lu and Milios [1997], where corresponding points are selected by choosing a point which has the matching range from the center of the reference scan's coordinate system. IDC, also proposed by Lu and Milios [1997] combines ICP and IMRP by using the ICP to calculate translation and IMPR to calculate rotation. The mentioned point to point methods can find the correct pose of the current scan in one step provided the correct associations are chosen. Since the correct associations are unknown, several iterations are performed. Matching may not always converge to the correct pose, since they can get stuck in a local minima. Due to the applied association rules, matching points have to be searched across 2 scans, resulting in $O(n^2)$ complexity[1], where n is the number of scan points. All three approaches operate in a Cartesian coordinate frame and therefore do not take advantage of the native polar coordinate system of a laser scan. However, as shown later in this chapter, a scan matching algorithm working in the polar coordinate system of a laser scanner can eliminate the search for corresponding points thereby achieving $O(n)$ computational complexity for translation estimation. $O(n)$ computational complexity is also achievable for orientation estimation if a limited orientation estimation accuracy is acceptable.

These point to point matching algorithms apply a so called projection filter [Gutmann,

[1]Lu and Milios [1997] claim that the IDC is of $O(n)$ complexity if the search for corresponding points is restricted to a window consisting of a fixed number of points. However if the angular resolution of laser scans is increased then the use of such a search window results in a decrease in the performance of the IDC due to the shrinking size of the search window expressed in angles. Therefore as it is correctly pointed out in [Gutmann, 2000] the computational complexity of IDC is $O(n^2)$.

2000] prior to matching. The objective of this filter is to remove those points from the reference and current scan not likely to have a corresponding point. The computational complexity of this filter is also $O(n^2)$.

There are other scan matching approaches such as the method of Weiss and Puttkamer [1995]. Here for both reference and current scans, an angle-histogram of the orientation of line segments connecting consecutive points is generated. The orientation of the current scan with respect to the reference scan is obtained by finding the phase with the maximum cross correlation of the 2 angle histograms. The translation is found similarly by calculating x and y histograms, and calculating cross correlations. In scan matching, not all approaches use only that information in a scan, which describes where objects are located. Thrun et al. [2000] in their scan matching method utilize the idea, that free space in a scan is unlikely to be occupied in future scans.

In scan matching another important task, apart from finding the current scans pose, is the estimation of the quality of the match. Lu and Milios [1997] calculate the uncertainty of the match results by assuming white Gaussian noise in the x,y coordinates of scan points. This implicitly assumes that correct associations are made that results in optimistic error estimates, especially in corridors. Bengtsson and Baerveldt in [2001] developed more realistic approaches. In their first approach the pose covariance matrix is estimated from the Hessian of the scan matching error function. In their second approach, the covariance matrix is estimated off-line by simulating current scans and matching them to the reference scan.

Mapping with scan matching has been done for example by minimizing an energy function [Lu, 1995], using a combination of maximum likelihood with posterior estimation [Thrun *et al.*, 2000], using local registration and global correlation [Gutmann, 2000] and using FastSLAM [Hähnel *et al.*, 2003a]. A Kalman filter implementation can be found in [Bosse *et al.*, 2004].

In this chapter the Polar Scan Matching (PSM) approach is described which works in the laser scanner's polar coordinate system, therefore taking advantage of the structure of the laser measurements by eliminating the search for corresponding points. It is assumed that in the 2D laser measurements range readings are ordered by their bearings. Laser range measurements of current and reference scans are associated with each other using the matching bearing rule, which makes translation estimation of the PSM approach $O(n)$ complexity unlike IDC's $O(n^2)$. The orientation estimation's computational complexity is also $O(n)$ if limited accuracy is acceptable, otherwise $O(kn)$, where k is proportional to the number of range readings per unit angle i.e. to the angular resolution of the scan. Note that k is introduced to differentiate between increasing the number of scan points by increasing the field of view or the angular resolution of the laser range finder. An $O(mn)$ complexity scan projection algorithm working in polar coordinates is also described in this chapter. The variable m equals to one added to the

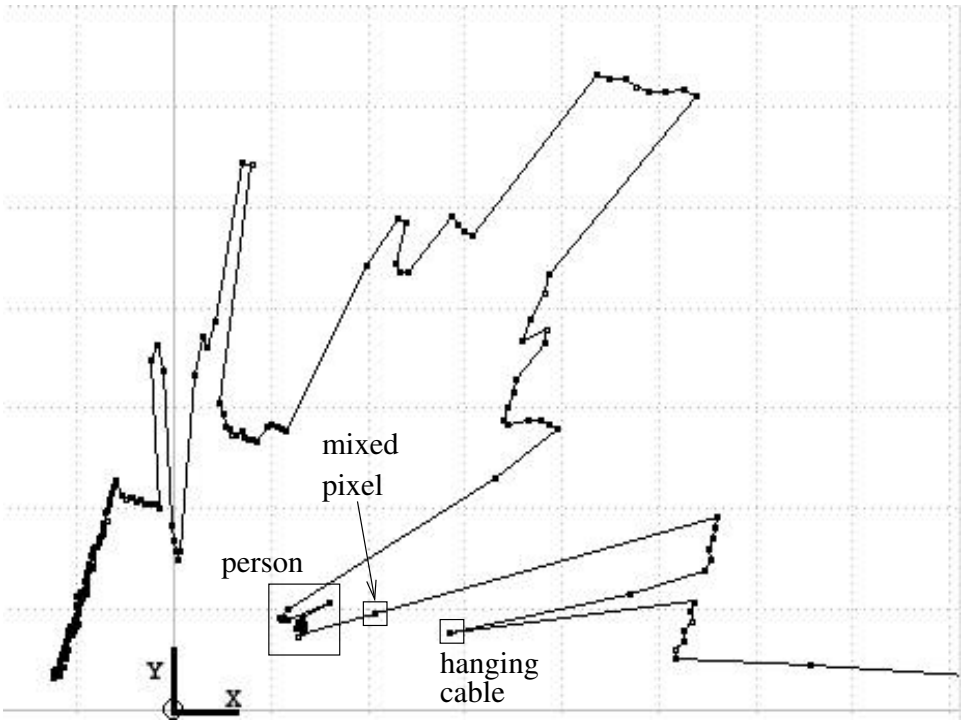


Figure 4.1: Laser scan in a Cartesian coordinate frame. Grid is 1m.

maximum number of objects occluding each other in the current scan viewed from the reference scan's pose. However this projection filter is of $O(n)$ complexity if no occlusions occur in the scan, therefore being more efficient than that of [Gutmann, 2000].

The rest of the chapter is organized as follows; first scan preprocessing steps, followed by the PSM algorithm is described. A heuristic scan match error model is presented next followed by a Kalman filter SLAM implementation utilizing our scan matching approach. Details of experimental results follow that include simulation, ground truth measurements and an implementation of SLAM. Finally conclusions and future work are presented.

The work presented in this chapter is published as [Diosi and Kleeman, 2005a; 2005b].

4.2 Scan Preprocessing

Prior to matching, the current and the reference scans are preprocessed. Preprocessing helps to remove erroneous measurements, clutter or to group measurements of the same object to increase the accuracy and robustness of scan matching. In fig. 4.1 a laser scan is depicted in a Cartesian coordinate system. Corresponding raw range measurements are shown in fig. 4.2. Laser scans can have points which are not suitable matching. Such points are:

- Points representing moving objects such as the legs of a person in fig 4.1. Table and chair legs are also such points, since they are less likely to be static in the long term.

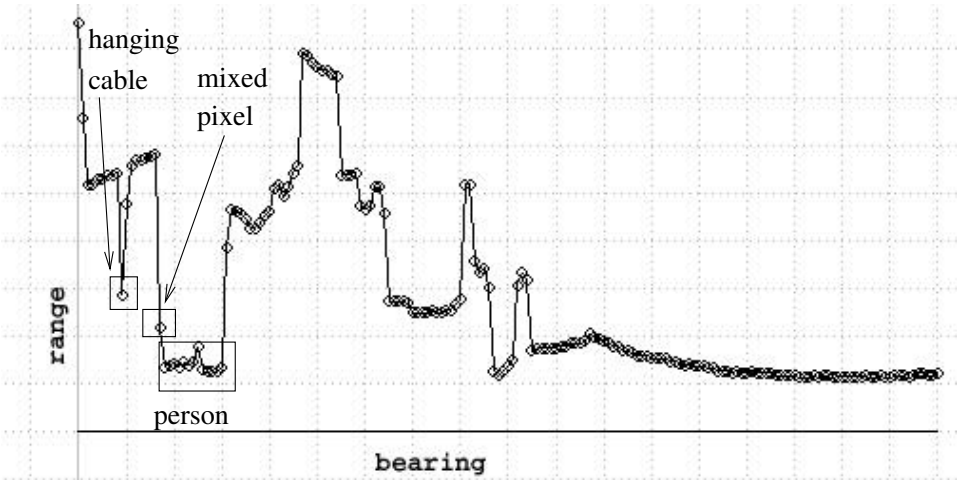


Figure 4.2: Scan of fig. 4.1 in the laser's polar coordinate frame. Horizontal grid size is 10°, vertical grid size is 1m.

- Mixed pixels. At range discontinuities laser scanners often generate measurements which are located in the free space between two objects [Ye and Borenstein, 2002].

- Measurements with maximum range. Such readings are returned, when there is no object withing the range of the scanner. Also some surfaces (for example clean clear glass) do not illuminate well and show a laser spot, therefore they can appear as measurements with maximum range.

Instead of only removing range readings which are out of the range of the sensor, it was found useful to artificially restrict the sensor to a distance

$$PM\_MAX\_RANGE = 10m \tag{4.1}$$

and disregard (tag) any more distant readings. When having a sensor with 1° angular resolution, the minimum distance between two readings at 10 meters is 17cm. A large distance between neighboring points complicates the segmentation of scans, since if the distance between measured points is large it is hard to decide if the points belong to the same object. Interpolating between two neighboring points belonging to 2 different objects can be a source of error. In addition by artificially restricting the range of the sensor, difficulties may be introduced in large rooms with a lot of open space.

In the following subsections we will focus on how to exclude these unwanted points from the scan matching process.

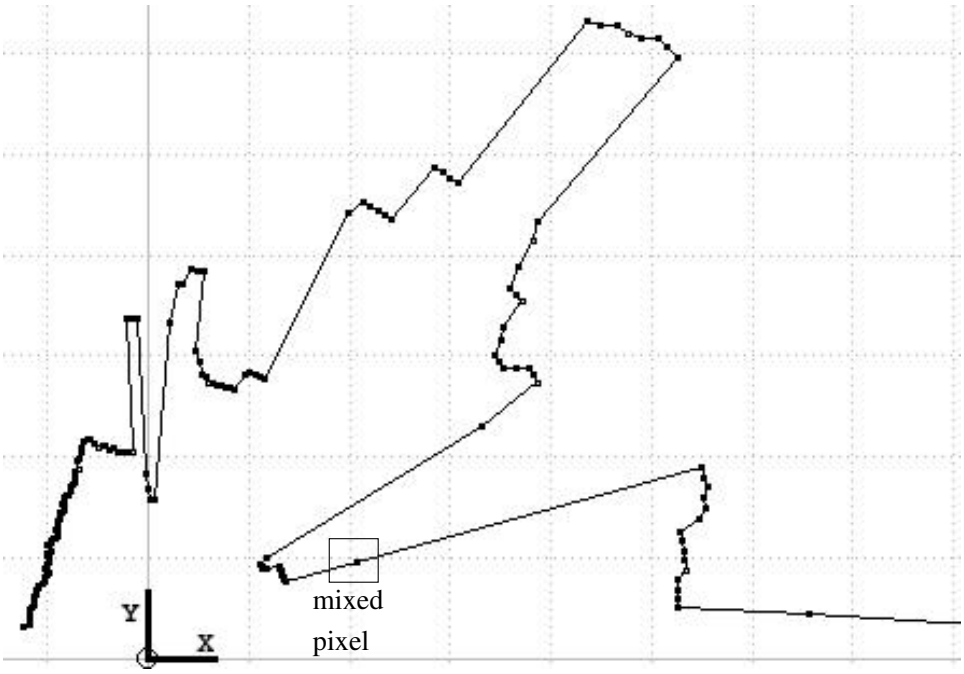


Figure 4.3: Laser scan of fig. 4.1 in a Cartesian coordinate frame after median filtering. Grid is 1m.

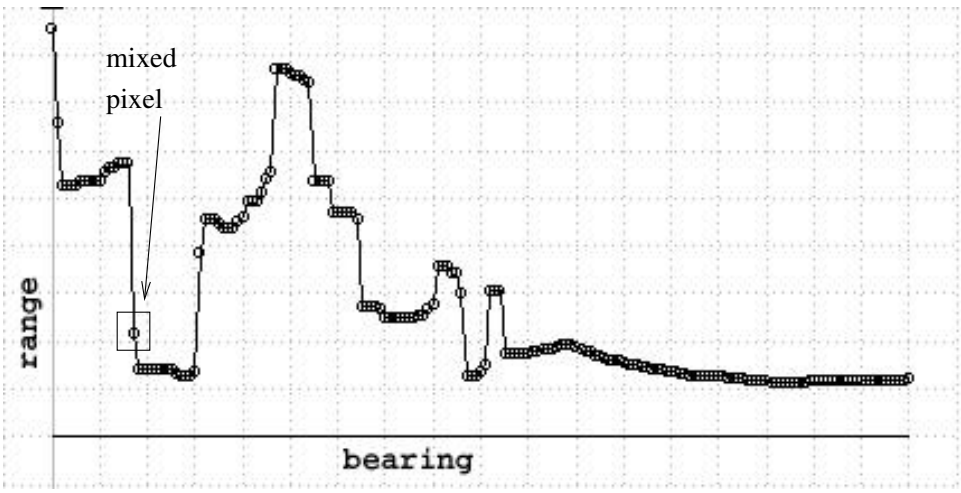


Figure 4.4: Scan of fig. 4.1 in the laser's polar coordinate frame after median filtering. Horizontal grid size is 10°, vertical grid size is 1m.

4.2.1 Median Filtering

Median filters are used to replace outliers with suitable measurements [Gutmann, 2000]. After the application of a median filter to the range readings, objects such as chair and table legs are likely to be removed. Similarly to [Gutmann, 2000], a window size of

$$PM\_MEDIAN\_WINDOW = 5 \tag{4.2}$$

for the median filter was found satisfactory since it can replace at most 2 neighborings outliers. Mixed pixels are unlikely to be replaced by a median filter. The result after applying median filter to the scan of fig. 4.1 can be seen in figures 4.3 and 4.4. From fig. 4.4 it is clear, that at least 4 spikes were replaced, but all the mixed pixels still remain.

4.2.2 Long Range Measurements

After the application of a median filter all points further than a threshold *PM_MAX_RANGE* are tagged. These tagged points are used only in segmentation described next and not in scan matching. Range measurements larger than *PM_MAX_RANGE* are not used in the scan matching because distance between such measurements is large, which makes it hard to decide if they belong to the same object or not.

4.2.3 Segmentation

Segmenting range measurements can have two advantages. The first advantage is that interpolation between 2 separate objects can be avoided if one knows that the objects are separate. Such interpolation is useful when one wants to know how a scan would look from a different location (scan projection). The second advantage is that if laser scans are segmented and the segments are tracked in consecutive scans then certain types of moving objects can be identified. Tracking moving objects can make scan matching more robust.

Two criteria are used in the segmentation process. According to the first criterion, a range reading, not differing more than

$$PM\_MAX\_DIFF = 20cm \tag{4.3}$$

from the previous range reading, belongs to the same segment. This criterion fails to correctly segment out points which are for example on a wall oriented towards the laser. Therefore a second criterion is also applied according to which if 3 consecutive range readings lie approximately on the same polar line, then they belong to the same segment. Note that a mixed pixel can only then connect two objects if the distance between first object and mixed pixel and

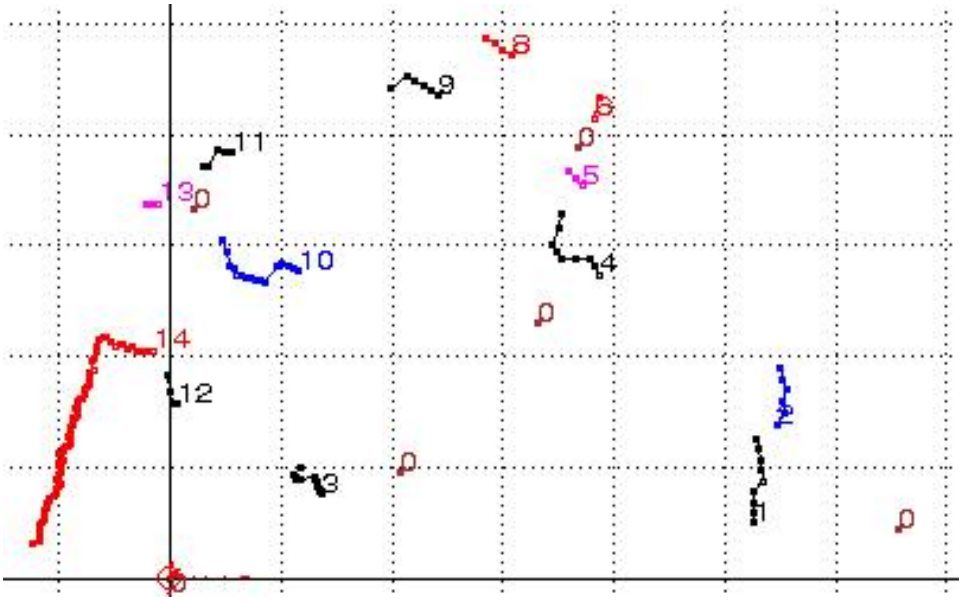


Figure 4.5: Laser scan of fig. 4.3 in a Cartesian coordinate frame after segmentation. Grid is 1m. Segments are assigned numbers. 0 is assigned to segments having only one point.

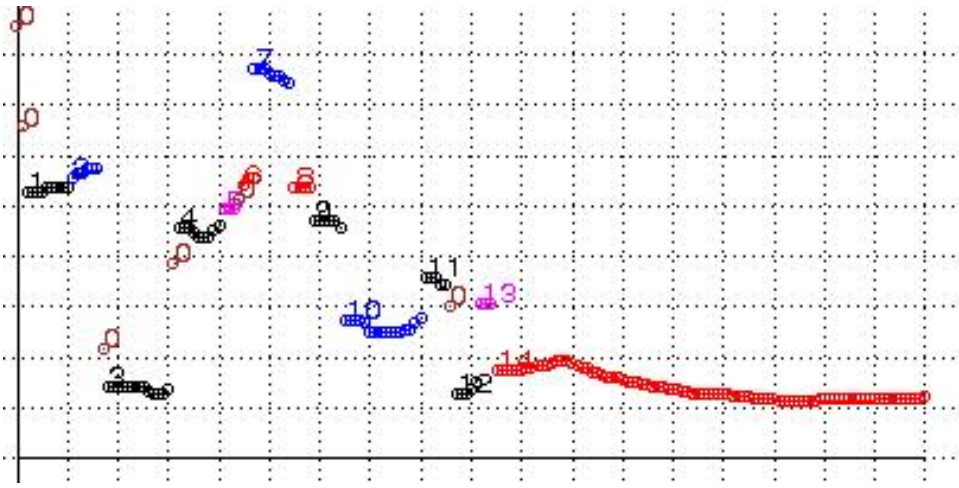


Figure 4.6: Scan of fig. 4.3 in the laser's polar coordinate frame after segmentation. Horizontal grid size is 10°, vertical grid size is 1m. Segments are assigned numbers. 0 is assigned to segments having only one point.

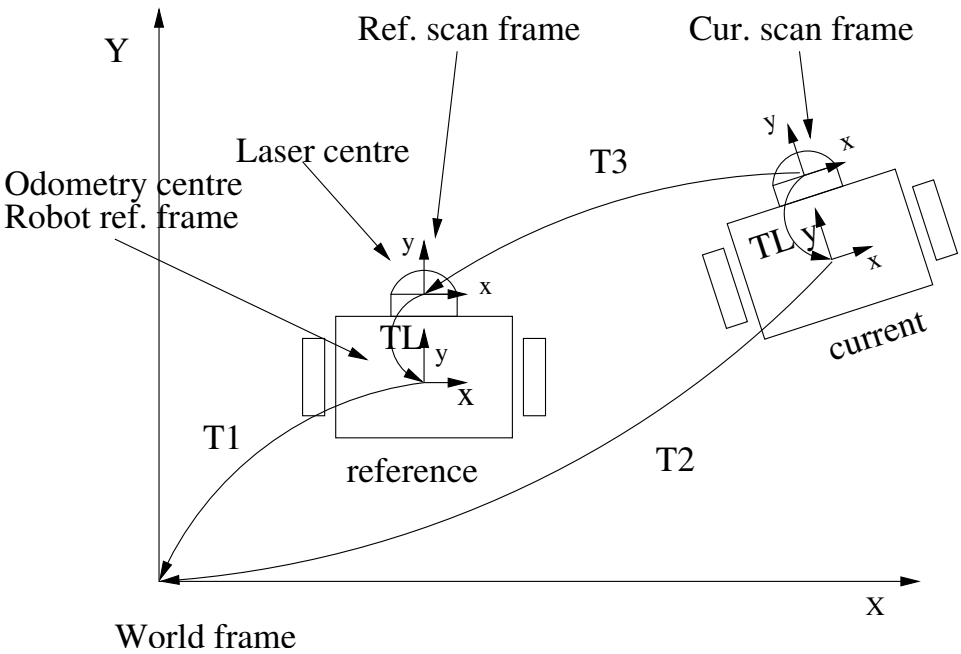


Figure 4.7: Reference and current coordinate frames.

second object and mixed pixel is less than PM_MAX_DIFF. Tagged range readings also break segments.

Segmentation results can be seen in fig. 4.5 and 4.6. Different segments are assigned different numbers, except 0, which is assigned to segments consisting of only one point. Segments assigned 0 are also tagged, therefore they are not used in the scan matching process. Note that most of the mixed pixels get assigned 0.

Note that this simple segmentation is of $O(n)$ complexity.

4.2.4 Motion Tracking

One of the problems of point to point scan matching is that moving objects can cause wrong associations thus reducing the accuracy of the scan matching results, or can even cause divergence. It is therefore advisable to track and tag moving objects in laser scans. Motion tracking constitutes future work and is beyond the scope of this chapter.

4.2.5 Current Scan Pose in Reference Scan Coordinate Frame

To make scan matching simpler, let us set the goal of scan matching to find the position and orientation (or pose) of the current laser scan's coordinate frame with respect to the reference scan's coordinate frame (see fig. 4.7). Otherwise the introduction of a world frame in which to relate the current and reference frame would make the equations describing the relation of

current and reference scan points more complicated. The reference scan's coordinate frame is the coordinate frame of the laser scanner at the reference location.

To find the position and orientation of the current frame with respect to the reference frame, from fig. 4.7 we can write:

$$\mathbf{T_1 T_L T_3} = \mathbf{T_2 T_L} \tag{4.4}$$

where $\mathbf{T}_1$ is the homogeneous transformation matrix from robot frame at reference location to world frame, $\mathbf{T}_2$ is the transformation from robot frame at current location to world frame, $\mathbf{T}_3$ is the transformation from current laser scan frame to reference laser scan frame and $\mathbf{T}_L$ is the transformation from laser frame into robot frame. With the multiplication of the matrices it is assumed that coordinates of points are stored in column vectors therefore the transformations are applied from right to left. From (4.4), the transformation from current into reference frame is:

$$\mathbf{T_3} = \mathbf{T_L^{-1} T_1^{-1} T_2 T_L}. \tag{4.5}$$

If $(x_{rr}, y_{rr}, \theta_{rr})$ describes the robot's pose at the reference location expressed in world frame, $(x_{cr}, y_{cr}, \theta_{cr})$ describes the robot pose the current location expressed in world frame, (x_c, y_c, θ_c) describes the laser scanner's pose at the current location expressed in the reference frame, and if the laser scanners pose is described with (x_l, y_l, θ_l) in the robots frame, then:

$$\begin{aligned} T_3 &= \begin{bmatrix} \cos\theta_c & -\sin\theta_c & x_c \\ \sin\theta_c & \cos\theta_c & y_c \\ 0 & 0 & 1 \end{bmatrix} \quad T_1 = \begin{bmatrix} \cos\theta_{rr} & -\sin\theta_{rr} & x_{rr} \\ \sin\theta_{rr} & \cos\theta_{rr} & y_{rr} \\ 0 & 0 & 1 \end{bmatrix} \\ T_L &= \begin{bmatrix} \cos\theta_l & -\sin\theta_l & x_l \\ \sin\theta_l & \cos\theta_l & y_l \\ 0 & 0 & 1 \end{bmatrix} \quad T_2 = \begin{bmatrix} \cos\theta_{cr} & -\sin\theta_{cr} & x_{cr} \\ \sin\theta_{cr} & \cos\theta_{cr} & y_{cr} \\ 0 & 0 & 1 \end{bmatrix} \end{aligned} \tag{4.6}$$

By substituting (4.6) into (4.5) and comparing the left and right sides, the current pose (x_c, y_c, θ_c) expressed in the reference frame can be determined as:

$$\theta_c = \theta_{cr} - \theta_{rr} \tag{4.7}$$

$$x_c = x_l(\cos\beta - \cos\theta_l) + y_l(\sin\beta - \sin\theta_l) + (x_{cr} - x_{rr})\cos\gamma + (y_{cr} - y_{rr})\sin\gamma \tag{4.8}$$

$$y_c = -x_l(\sin\beta - \sin\theta_l) + y_l(\cos\beta - \cos\theta_l) - (x_{cr} - x_{rr})\sin\gamma + (y_{cr} - y_{rr})\cos\gamma \tag{4.9}$$

where $\beta = \theta_l + \theta_{rr} - \theta_{cr}$ and $\gamma = \theta_l + \theta_{rr}$.

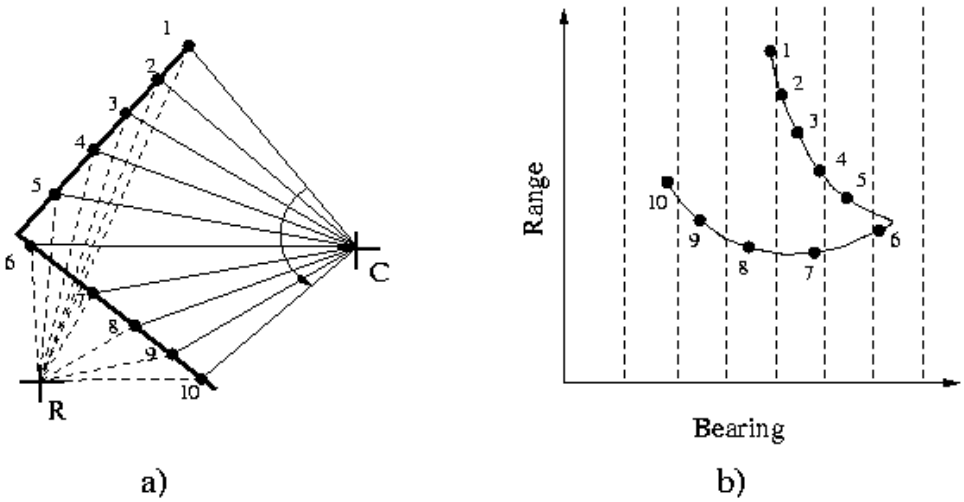


Figure 4.8: a) projection of measured points taken at C to location R. b) points projected to R shown in polar coordinates. Dashed lines represent bearings which the scanner would have sampled.

4.3 Scan Matching

The laser scan matching method described next aligns the current scan with respect to the reference scan so that the sum of square range residuals is minimized. It is assumed that an initial pose of the current scan is given, expressed in the coordinate frame of the reference scan. The coordinate frame of a laser scan is centered at the point of rotation of the mirror of a laser scanner. The X axis or zero angle of the laser's Cartesian coordinate system coincides with the direction of the first reported range measurement. The current scan is described as $C = (x_c, y_c, \theta_c, \{r_{ci}, \phi_{ci}\}_{i=1}^{n})$, where x_c, y_c, θ_c describe position and orientation, $\{r_{ci}, \phi_{ci}\}_{i=1}^{n}$ describe n range measurements r_{ci} at bearings ϕ_{ci}, expressed in the current scan's coordinate system. $\{r_{ci}, \phi_{ci}\}_{i=1}^{n}$ are ordered by the bearings in ascending order as they are received from a SICK laser scanner. The reference scan is described as $R = \{r_{ri}, \phi_{ri}\}_{i=1}^{n}$. Note that if bearings where range measurements are taken are unchanged in current and reference scans then $\phi_{ri} = \phi_{ci}$. The scan matching works as follows: after preprocessing the scans, scan projection followed by a translation estimation or orientation estimation are iterated. In the polar scan matching (PSM) of this chapter, one orientation step is followed by one translation step. More details on these steps are given in the following subsections.

4.3.1 Scan Projection

An important step in scan matching is finding out how the current scan would look if it were taken from the reference position. For example in fig. 4.8, the current scan is taken at location C and the reference scan is taken at position R. The range and bearings of the points from point R (see fig. 4.8b) are calculated:

$$r'_{ci} = \sqrt{(r_{ci}\cos(\theta_c+\phi_{ci})+x_c)^2+(r_{ci}\sin(\theta_c+\phi_{ci})+y_c)^2} \tag{4.10}$$

$$\phi'_{ci} = atan2(r_{ci}\sin(\theta_c+\phi_{ci})+y_c,\ r_{ci}\cos(\theta_c+\phi_{ci})+x_c) \tag{4.11}$$

where $atan2$ is the four quadrant version of arctan.

In fig. 4.8b the dashed vertical lines represent sampling bearings (ϕ_{ri}) of the laser at position R in fig. 4.8a. Since the association rule is to match bearings of points, next ranges r''_{ci} at the reference scan bearings ϕ_{ri} are calculated using interpolation. The aim is to estimate what the laser scanner would measure from pose R. This resampling step consists of checking (r'_{ci}, ϕ'_{ci}) (i.e. 1,2,..10 in fig. 4.8b) of each segment if there are one or more sample bearings between 2 consecutive points (i.e. between 1 and 2 there is one, between 6 and 7 there are 2). By linear interpolation a range value is calculated for each sample bearing. If a range value is smaller than an already stored range value at the same bearing, then the stored range is overwritten with the new one to handle occlusion. As in [Lu and Milios, 1997] a new range value is tagged as invisible if the bearings of the 2 segment points are in decreasing order.

A pseudo code implementation of the described scan projection is shown in fig. 4.9. Note that unlike the equations in this chapter, the indexes of vector elements in fig. 4.9 start from 0. Also note that this particular implementation assumes that laser scans have 1° bearing resolution. However this assumption is used only in the transformation of bearings from radians to indexes into range arrays and can be easy changed to work with scans of arbitrary resolution. The pseudo code on lines 00-09 transforms the current scan readings (ϕ_i, r_{ci}) into the reference scan's coordinate frame, while using the current frame pose (x_c, y_c, θ_c) expressed in the reference frame. Since the projected current scan (ϕ'_{ci}, r'_{ci}) is resampled next at the sample bearings ϕ_i of the reference scan, the data structures associated with the resampled current scan are also initialized. Status registers $tagged''_{ci}$ contain flags describing if resampled range readings r''_{ci} have been tagged or if they contain a range reading. All flags of the status registers are cleared except the flag PM_EMPTY which indicates that no range reading has been resampled into the particular position of the range array r''_c. Resampled current scan range readings r''_{ci} are set to a value which is larger than the maximum range of the laser scanner.

The resampling of the projected current scan readings (ϕ'_{ci}, r'_{ci}) takes place on lines 09-43 in a loop which goes through neighboring pairs of (ϕ'_{ci}, r'_{ci}). Pairs of measurements are only resampled if they belong to the same segment and none of them are tagged. Next, on lines 14-

```
/*************Scan Projection******************/
00 //Transform current measurements into reference frame
01 for i = 0 → number_of_points-1 {
02     x = r_ci cos(θ_c + φ_i) + x_c
03     y = r_ci sin(θ_c + φ_i) + y_c
04     r'_ci = √(x^2 + y^2)
05     φ'_ci = atan2(y,x)
06     tagged''_ci = PM_EMPTY
07     r''_ci = LARGE_VALUE
08 }
09 //Given the projected measurements (r'_ci, φ'_ci), calculate what would have been
10 measured with the laser scanner at the reference pose.
11 for i = 1 → number_of_points-1 {
12     if segment_ci ≠ 0 & segment_ci = segment_ci−1
13         !tagged_ci & !tagged_ci−1 & φ'_ci > 0 & φ'_ci−1 > 0 {
14         if φ'_ci > φ'_ci−1 { //Is it visible?
15             occluded = false
16             a_0 = φ'_ci−1
17             a_1 = φ'_ci
18             φ_0 = ceil(φ'_ci−1 180/π)
19             φ_1 = floor(φ'_ci 180/π)
20             r_0 = r'_ci−1
21             r_1 = r_ci
22         }else{
23             occluded = true
24             a_0 = φ'_ci
25             a_1 = φ'_ci−1
26             φ_0 = ceil(φ'_ci 180/π)
27             φ_1 = floor(φ'_ci−1 180/π)
28             r_0 = r'_ci
29             r_1 = r_ci−1
30         }
31         while φ_0 ≤ φ_1{
32             r = (r_1−r_0)/(a1−a0) (φ_0 π/180 − a_0) + r_0
33             ifφ_0 ≠ 0 & φ_0 < number_of_points & r''_cφ0 > r {
34                 r''_cφ0 = r
35                 tagged''_cφ0 & =∼ PM_EMPTY
36                 if occluded
37                     tagged''_cφ0 | = PM_OCCLUDED
38                 else
39                     tagged''_cφ0 & =∼ PM_OCCLUDED
40             φ_0 = φ_0 + 1
41         }//while
42     }//if
43 }//for
```

Figure 4.9: Scan projection pseudo code for 1° resolution.

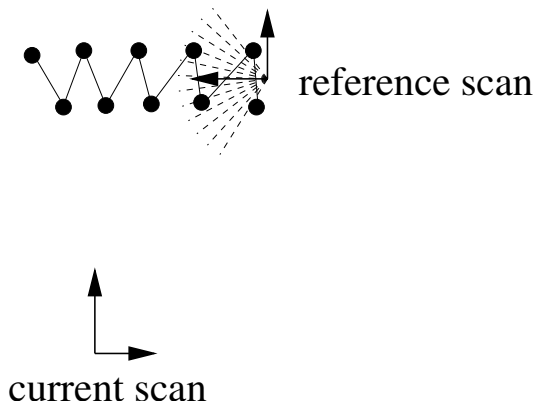


Figure 4.10: Example for the worst case scenario for scan projection.

30 the measurement pair is checked if it is viewed from behind by testing if $\phi'_{ci} > \phi'_{ci-1}$. Then depending on their order, ϕ'_{ci} and ϕ'_{ci-1} are converted into indexes ϕ_0, ϕ_1 into the resampled ranges array, so that $\phi_0 <= \phi_1$. This is done to simplify the following interpolation step where the resampled ranges r''_c are calculated in a while loop (lines 31-43) at index ϕ_0 which is incremented until it reaches ϕ_1. In the while loop first range r corresponding to ϕ_0 is calculated using linear interpolation. Then if ϕ_0 is within the bounds of the array r''_c and if r is smaller than the value already stored at $r''_{c\phi_0}$ then the empty flag of $tagged''_{c\phi_0}$ is cleared and $r''_{c\phi_0}$ is overwritten by r. This last step filters out those projected current scan readings which are occluded by other parts of the current scan. Finally the occluded flag of $tagged''_{c\phi_0}$ is cleared or set, depending on if ϕ'_{ci} was greater than ϕ'_{ci-1}, and ϕ_0 is incremented.

The body of the while loop (lines 32-40) of the pseudo code is executed at most $2n$ times for scans with no occlusion, where n is the number of points. However it is easy to contrive a scenario where the inside of the while loop would execute at most n^2 times. For example fig. 4.10 depicts a situation where the noise in the current scan readings (drawn with connected circles) is large and the scan readings are aligned with the reference scan's frame so that most of the reference scan's laser beams go through in between the points of the current scan. In a such case for each pair of current scan points the while loop would execute almost n times resulting in a total number of executions between $2n$ and n^2. The computational complexity of this projection filter is $O(mn)$ where m is the maximum number of objects occluding each other in the current scan viewed from the reference scan's pose incremented by one. For example if there is no occlusion then m is 1. If there is at least one object which occludes another object, while the occluded object does not occlude any other object, then m is 2. If there are objects A, B and C where A occludes B and B occludes C then m is 3.

The scan projection filter described in [Gutmann, 2000] is of $O(n^2)$ complexity, because a double loop is employed to check for occlusion. That occlusion check consists of checking whether any current scan point in XY coordinates is obscured by any other pair of consecutive

current or reference scan points. Since the scan projection implementation in fig. 4.9 is of $O(n)$ complexity when there are no occlusions in the current scan, it is reasonable to believe that under normal circumstances it is more efficient than that described in [Gutmann, 2000]. Due to its efficiency the projection filter of fig. 4.9 is applied in each iteration of the PSM scan matching algorithm.

Note that the Cartesian projection filter in [Gutmann, 2000] removes all current scan points which are further than one meter from all reference scan points and vice versa. In PSM associated current and reference scan measurements with a residual larger than a preset threshold are ignored in the position estimation process and not in the projection filter. This eliminates the need for performing the computationally expensive removal of points without correspondence in the projection filter.

4.3.2 Translation Estimation

After scan projection, for each bearing ϕ_{ri} there is at most one r''_{ci} from the projected current scan and a corresponding r_{ri} from the reference scan. The aim is to find (x_c, y_c) which minimizes $\sum w_i(r_{ri} - r''_{ci})^2$, where w_i is a weight used to reduce the weighting[2] of bad matches. To minimize the weighted sum of square residuals linear regression was applied to the linearized (4.10):

$$\Delta r_i \approx \frac{\partial r''_{ci}}{\partial x_c}\Delta x_c + \frac{\partial r''_{ci}}{\partial y_c}\Delta y_c = \cos(\phi_{ri})\Delta x_c + \sin(\phi_{ri})\Delta y_c \tag{4.12}$$

$\frac{\partial r''_{ci}}{\partial x_c} = \cos(\phi_{ri})$ has been derived from (4.10) the following way:

$$\begin{aligned} \frac{\partial r''_{ci}}{\partial x_c} &= \frac{1}{2}\frac{2(r_{cj}\cos(\theta_c+\phi_{cj})+x_c)}{\sqrt{(r_{cj}\cos(\theta_c+\phi_{cj})+x_c)^2+(r_{cj}\sin(\theta_c+\phi_{cj})+y_c)^2}} \\ &= \frac{(r_{cj}\cos(\theta_c+\phi_{cj})+x_c)}{r''_{ci}} = \frac{r''_{ci}\cos\phi_{ri}}{r''_{ci}} = \cos\phi_{ri} \end{aligned} \tag{4.13}$$

Where ϕ_{cj}, r_{cj} is a virtual, unprojected reading which would correspond to an uninterpolated ϕ_{ri}, r''_{ci}. The derivation of $\frac{\partial r''_{ci}}{\partial y_c}$ is analogous to the derivation of $\frac{\partial r''_{ci}}{\partial x_c}$.

If range differences between projected current range and reference range readings are modeled as

$$(\mathbf{r''_c} - \mathbf{r_r}) = \mathbf{H}\begin{bmatrix} \Delta x_c \\ \Delta y_c \end{bmatrix} + \mathbf{v} \tag{4.14}$$

where $\mathbf{v}$ is the noise vector and

$$\mathbf{H} = \begin{bmatrix} \frac{\partial r''_{c1}}{\partial x_c} & \frac{\partial r''_{c1}}{\partial y_c} \\ \frac{\partial r''_{c2}}{\partial x_c} & \frac{\partial r''_{c2}}{\partial y_c} \\ \dots & \dots \end{bmatrix}, \tag{4.15}$$

[2]Note that there is also an implicit weighting of closer objects, since they cover a larger angle.

then the position correction Δx_c, Δy_c of the current scan is calculated by minimizing the sum of weighted range residuals $\sum w_i(r_{ri} - r''_{ci})^2$ using the well known equation for weighted least squares [Kay, 1993]:

$$\begin{bmatrix} \Delta x_c \\ \Delta y_c \end{bmatrix} = (\mathbf{H^T W H})^{-1}\mathbf{H^T W}(\mathbf{r_r} - \mathbf{r''_c}) \tag{4.16}$$

where $\mathbf{r''_c}, \mathbf{r_r}$ are vectors containing r''_{ci} and r_{ri} and $\mathbf{W}$ is a diagonal matrix of weights. The elements of $\mathbf{W}$ are calculated according to the recommendations of Dudek and Jenkin in [2000]:

$$w_i = 1 - \frac{d_i^m}{d_i^m + c^m} \tag{4.17}$$

where $d_i = r''_{ci} - r_{ri}$ is the error between projected current scan range measurements and reference scan range measurements and c is a constant. Equation (4.17) describes a sigmoid function with weight 1 at $d_i = 0$ and a small weight for large d_i. Parameter c determines where the sigmoid changes from 1 to 0, and m determines how quickly the sigmoid function changes from 1 to 0. In [Dudek and Jenkin, 2000] (4.17) was used to weight the distance of a laser scan point to a line in a point-to-feature scan matching method.

To reduce the effects of association errors in the implementation of equation (4.16), only those visible measurements are taken into consideration which are not tagged (see section 4.2). Also the errors between reference and current scan range measurements have to be smaller than a preset threshold *PM_MAX_ERROR* to be included.

An example implementation for one step of the translation estimation can be seen in fig. 4.11. In the implementation first $\mathbf{H^T W H}$ and $\mathbf{H^T W \Delta r}$ are calculated for untagged associated reference and current scan measurements, which are closer to each other than a threshold. Note that elements h1, h2 of the Jacobian matrix $\mathbf{H}$ on lines 05-06 have to be calculated only once, since ϕ_{ri} depends only on the type of laser scanner. Matrix $\mathbf{H^T W H}$ is inverted on lines 16-20 followed by the calculation of pose corrections. As one can see from fig. 4.11, translation estimation is of $O(n)$ complexity. The translation estimation step of IDC and ICP is of $O(n^2)$ complexity.

Note that the equation used in other point-to-point scan matching methods which operate in XY coordinate systems such as ICP or IDC find the correct translation and rotation of the current scan in one step if the correct associations are given. The PSM approach, due to the use of linearization, requires multiple iterations. Since the correct associations are in general not known multiple iterations are necessary for the other methods as well. Also note that the PSM approach for translation estimation is most accurate if the correct orientation of the current scan is known. Estimating the orientation of the current scan is described in section 4.3.3.

A negative property of this translation estimation approach is apparent when matching scans which were taken of long featureless corridors - the position error along the corridor

```
/**************Polar Translation Estimation****************/
00 //Matrix multiplications for linearized least squares
01 for i = 0 → number_of_points-1 {
02     Δr = r_ri − r''_ci
03     if !tagged''_ci & !tagged_ri &| Δr| < PM_MAX_ERROR {
04         w = C/(Δr^2+C) //weight calculation
05         h1 = cos φ_ri
06         h2 = sin φ_ri
07         //calculating H^T W Δr
08         hwr1 = hwr1 + w * h1Δr
09         hwr2 = hwr2 + w * h2Δr
10         hwh11 = hwh11 + w * h1^2//calculating H^T WH
11         hwh12 = hwh12 + w * h1 * h2
12         hwh21 = hwh21 + w * h1 * h2
13         hwh22 = hwh22 + w * h2^2
14     }//if
15 }//for
16 D = hwh11 * hwh22 − hwh12 * hwh21
17 inv11 = hwh22/D
18 inv12 = −hwh12/D
19 inv21 = −hwh21/D
20 inv22 = hwh11/D
21 Δx = inv11 * hw1 + inv12 * hw2
22 Δy = inv21 * hw1 + inv22 * hw2
23 x_c = x_c + Δx
24 y_c = y_c + Δy
```

Figure 4.11: Pseudo code for translation estimation in polar coordinates.

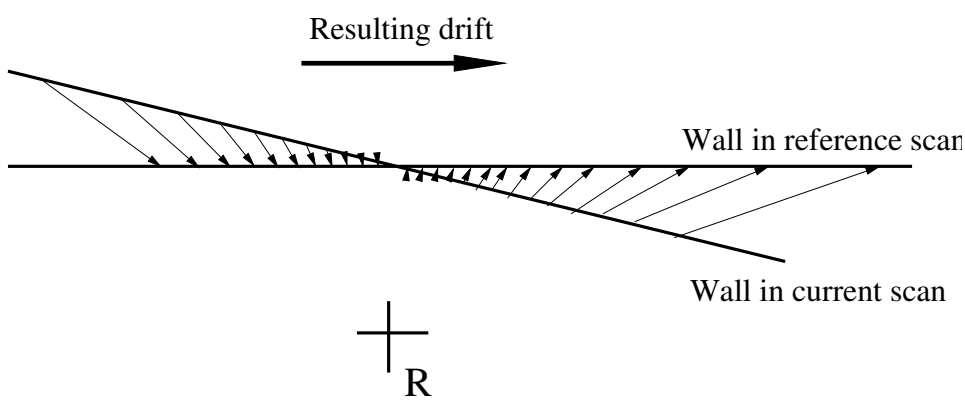


Figure 4.12: Cause of drift in for translation estimation in corridor like environments.

can drift. In fig. 4.12 the reference and current scan contain only a wall. The associations are depicted with an arrow pointing from the current scan point to the reference scan point. The direction of the arrows also coincide with the corresponding Jacobians which project into the x and y corrections. From fig. 4.12 it can be observed, that all the arrows have a positive x component, therefore the translation correction will drift to the right.

There are two reasons why polar scan matching estimates translation separately from orientation. First reason: if the partial derivatives $\frac{\partial r''_{ci}}{\partial \theta_c} = y_c \cos\phi_{ri} - x_c \sin\phi_{ri}$ are appended to matrix $\mathbf{H}$ (4.15), matrix $\mathbf{H^T W H}$ can become ill-conditioned and the estimation process can diverge. The cause of ill-conditioning lies in the structure of $\mathbf{H}$:

$$\mathbf{H} = \begin{bmatrix} \vdots & \vdots & \vdots \\ \cos\phi_{ri} & \sin\phi_{ri} & y_c \cos\phi_{ri} - x_c \sin\phi_{ri} \\ \vdots & \vdots & \vdots \end{bmatrix}, \tag{4.18}$$

where two columns contain small numbers in the range of $\langle -1, 1 \rangle$ and the third column contains potentially large numbers depending on the value of x_c and y_c. As an example let us assume that $x_c = 100,\ y_c = 100,\ \phi_{ri} = 0°, 1°, 2°, ..., 180°$ and $\mathbf{W}$ is a diagonal matrix with 1's on the diagonal. Then the largest eigenvalue of $\mathbf{H^T W H}$ is about $2x10^6$ and the smallest eigenvalue is about $3x10^{-33}$ which means the matrix $\mathbf{H^T W H}$ is ill-conditioned and will likely cause numerical instability. On the other hand if x_c and y_c are 0, then the right column of $\mathbf{H}$ will consist of 0's and $\mathbf{H^T W H}$ will have 0 determinant and will not have an inverse which is necessary for the computation of (4.16). The second reason why polar scan matching estimates translation separately from orientation is that estimating orientation as described later is much more efficient.

Note, that if uniform weights were used, and all measurements were used in each scan matching, then matrix $(\mathbf{H^T W H})$ is a constant matrix and as such it has to be calculated only once.

It is interesting to investigate how the matching bearing association rule performs with the pose estimation equations described in Lu and Milios [1997]. The details are given next.

Pose Estimation in Cartesian Frame

Lu and Milios in [1997] minimize the sum of square distance between current and actual scan points. To increase robustness it is recommended in [Gutmann, 2000], that only the best 80% of matches take part in the estimation process. Here instead of sorting the matches, each match is weighted based on its "goodness", as in the previous subsection. The original objective

function in [Lu and Milios, 1997] expressed using the notation used in this chapter is:

$$E = \sum_{i=1}^{n} (x''_{ci} \cos\Delta\theta_c - y''_{ci} \sin\Delta\theta_c + \Delta x_c - x_{ri})^2 + (x''_{ci} \sin\Delta\theta_c + y''_{ci} \cos\Delta\theta_c + \Delta y_c - y_{ri})^2 \quad (4.19)$$

Where (x''_{ci}, y''_{ci}) correspond to the projected and interpolated current scan's (ϕ_{ri}, r''_{ci}) in Cartesian coordinate frame. (x_{ri}, y_{ri}) corresponds to (ϕ_{ri}, r''_{ri}) of the reference scan. The weighted version used in this chapter:

$$E = \sum_{i=1}^{n} w_i \left[(x''_{ci} \cos\Delta\theta_c - y''_{ci} \sin\Delta\theta_c + \Delta x_c - x_{ri})^2 + (x''_{ci} \sin\Delta\theta_c + y''_{ci} \cos\Delta\theta_c + \Delta y_c - y_{ri})^2 \right] \quad (4.20)$$

Since (x_{ci}, y_{ci}) belong to the same bearing as (x_{ri}, y_{ri}), (4.20) is equivalent to the sum of weighted square range residuals $\sum w_i (r_{ri} - r''_{ci})^2$ used in the previous subsection. A solution to (4.20) can be obtained by solving $\frac{\partial E}{\partial x_c} = 0$, $\frac{\partial E}{\partial y_c} = 0$ and $\frac{\partial E}{\partial \theta_c} = 0$:

$$\Delta\theta_c = atan2\left(\bar{x}_r \bar{y}''_c - \bar{x}''_c \bar{y}_r + W(S_{y_r x''_c} - S_{x_r y''_c}), -\bar{y}_r \bar{y}''_c - \bar{x}''_c \bar{x}_r + W(S_{x_r x''_c} - S_{y_r y''_c})\right) \quad (4.21)$$

$$\Delta x_c = \frac{\bar{x}_r - \bar{x}''_c \cos\Delta\theta_c + \bar{y}''_c \sin\Delta\theta_c}{W} \quad (4.22)$$

$$\Delta y_c = \frac{\bar{y}_r - \bar{x}''_c \sin\Delta\theta_c - \bar{y}''_c \cos\Delta\theta_c}{W} \quad (4.23)$$

where

$$\begin{aligned} \bar{x}_r &= \sum w_i x_{ri}, & \bar{y}_r &= \sum w_i y_{ri} \\ \bar{x}''_c &= \sum w_i x''_{ci}, & \bar{y}_c &= \sum w_i y''_{ci} \\ S_{x_r y''_c} &= \sum w_i x_{ri} y''_{ci}, & S_{x''_c y_r} &= \sum w_i x''_{ci} y_{ri} \\ S_{x_r x''_c} &= \sum w_i x_{ri} x''_{ci}, & S_{y''_c y_r} &= \sum w_i y''_{ci} y_{ri} \\ W &= \sum w_i \end{aligned} \quad (4.24)$$

Even though the objective function here is equivalent to the objective function in the previous subsection, the solutions are not equivalent. In the previous subsection, one iteration returns an approximate solution for x_c, y_c. Linearization was necessary due to the square root in (4.10). Here on the other hand a solution is calculated without linearization and without the need for multiple iterations (assuming known associations), which also contains θ_c and not just x_c, y_c. In experiments it was found that if only (4.21)–(4.23) are used to estimate pose, then the convergence speed is unsatisfactory, and the estimation process is more likely to get stuck in a local minima. Therefore just as in the previous subsection, it is best to interleave the described way of estimating x_c, y_c, θ_c with the orientation estimation described in the following subsection.

Note that the advantage of using (4.21)–(4.23) for calculating a solution of $\sum w_i (r_{ri} - r''_{ci})^2$

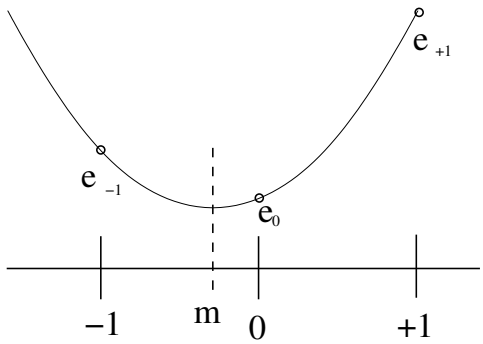


Figure 4.13: Orientation estimate improvement by interpolation.

in one step opposed to the multiple iteration needed when using (4.16) is not important since the unknown associations of the reference and current scan points require an iterative pose estimation process. Also note that from now on using (4.21)–(4.23) together with the orientation estimation approach described next will be called PSM-C.

4.3.3 Orientation Estimation

Change of orientation of the current scan is represented in a polar coordinate system by a left or right shift of the range measurements. Therefore assuming that the correct location of the current scan is known and the reference and current scans contain measurements of the same static objects, the correct orientation of the current scan can be found by shifting the projected current scan (r''_{ci}, ϕ_{ri}) until it covers the reference scan. A $\pm 20^\circ$ shift was implemented at 1° intervals of the projected current scan, and for each shift angle the average absolute range residual is calculated. Orientation correction is estimated by fitting a parabola to the 3 closest points to the smallest average absolute error, and calculating the abscissa of the minimum.

The calculation of the abscissa of the minimum is performed as follows. Assume that the 3 points of the error function are $(-1, e_{-1})$, $(0, e_0)$ and $(+1, e_{+1})$ (see fig. 4.13). Then the abscissa m of the minimum e_m of the parabola described as $e = at^2 + bt + c$ is sought. Given the equation of the parabola, the abscissa of the minimum can be found at:

$$\frac{\partial e}{\partial t} = 0 = 2am + b = 0 \Rightarrow m = -\frac{b}{2a} \tag{4.25}$$

To find a, b let us substitute the 3 known points into the equation of the parabola:

$$a - b + c = e_{-1} \tag{4.26}$$

$$c = e_0 \tag{4.27}$$

$$a + b + c = e_{+1} \tag{4.28}$$

By substituting (4.27) into (4.26) and (4.28), and adding (4.26) and (4.28), one gets:

$$2a + 2e_0 = e_{-1} + e_{+1} \Rightarrow a = \frac{e_{-1} + e_{+1} - 2e_0}{2} \tag{4.29}$$

Similarly b can be calculated by subtracting (4.26) from (4.28):

$$2b = e_{+1} + e_{-1} \Rightarrow b = \frac{e_{+1} - e_{-1}}{2} \tag{4.30}$$

Then the abscissa of the minimum is:

$$m = -\frac{b}{2a} = -\frac{\frac{e_{+1}-e_{-1}}{2}}{2\frac{e_{-1}+e_{+1}-2e_0}{2}} = \frac{e_{+1} - e_{-1}}{2(2e_0 - e_{-1} - e_{+1})} \tag{4.31}$$

Assuming the orientation correction corresponding to 0 in fig. 4.13 is $\Delta\theta_1$, the distance between 0 and 1 in fig. 4.13 is $\Delta\phi$, then the estimated orientation correction will be

$$\Delta\theta_c = \Delta\theta_1 + m\Delta\phi \tag{4.32}$$

A simple pseudo code implementation of the orientation estimation is shown in fig 4.14. In fig 4.14 on lines 00-19 average absolute range residuals are calculated while shifting the reference range readings left and right by Δi. The value of Δi changes in the range of $\pm WINDOW$. The value of $WINDOW$ is chosen so, that the range of shift is around $\pm 20^\circ$. On lines 02-07 those indexes into the current range readings array are calculated which overlap with the shifted reference range array. In a for loop average absolute range residuals are calculated only for untagged range readings. The average range residuals for the corresponding shift values are then stored in $error_k$ and in β_k. Then the minimum error and the corresponding shift value is found on lines 21-25, which is improved by fitting a parabola on line 26-27. $\Delta\phi$ on line 27 is the angle corresponding to changes of Δi.

The computational complexity of this orientation estimation approach depends on how the increments of Δi are chosen. If the reference scan is shifted by constant increments for example by 1° then the computational complexity is $O(n)$. The justifications for using constant increments, opposed to the smallest possible increment which is the angular resolution of the scan are the following:

- The orientation estimates are improved by quadratic interpolation.
- When performing scan matching in real environments the error in orientation due to fixed Δi increments will likely to be much smaller than errors caused by incorrect associations.

If the increments of Δi are chosen to be equal to the bearing resolution of the scans, then assuming constant size of search window in angles, the orientation estimation will be of $O(kn)$

```
/*************Orientation estimation****************/
00 for Δi = −WINDOW → +WINDOW {
01     n = 0, e = 0
02     if Δi <= 0
03         min_i = −Δi
04         max_i = number_of_points
05     else
06         min_i = 0
07         max_i = number_of_points − Δi
08     for i = min_i → max_i − 1 {
09         if !tagged''_ci & !tagged_{ri+Δi}
10             e = e + |r''_ci − r_{ri+Δi}|
11             n = n + 1
12     }
13     if n > 0
14         error_k = e/n
15     else
16         error_k = 1000
17     β_k = Δi
18     k = k + 1
19 }
20 e_min = LARGE_VALUE
21 for i = 0 → k-1 {
22     if error_i < e_min
23         e_min = error_i
24         i_min = i
25 }
26 m = (e_{i_min+1} − e_{i_min−1}) / (2(2e_{i_min} − e_{i_min−1} − e_{i_min+1}))
27 θ_c = θ_c + (β_{i_min} + m)Δφ
```

Figure 4.14: Pseudo code for orientation estimation in polar coordinates.

complexity, where k is proportional to the number of range measurements per unit angle, i.e. to the angular resolution of the scan.

The last possibility discussed here in the choice of the increments of Δi is when one starts from a coarse increment of Δi and iteratively reduce Δi together with the size of the search window. In this case the computational complexity of $O(n \log n)$ may be achieved.

4.4 Error Estimation

4.4.1 Covariance Estimate of Weighted Least Squares

If correct associations are assumed, then the covariance estimate for the translation estimate of the polar scan matching algorithm is the same as the covariance estimate for weighted least squares [Kay, 1993]:

$$\mathbf{C} = \sigma_r^2 (\mathbf{H^T W H})^{-1}, \tag{4.33}$$

where σ_r^2 is estimated range error variance. σ_r^2 can be estimated based on the range residuals similarly to [Cox, 1991] as

$$\sigma_r^2 = \frac{[\mathbf{r}_c'' - \mathbf{r}_r]^T [\mathbf{r}_c'' - \mathbf{r}_r]}{n-4} \tag{4.34}$$

Unfortunately even if the current and reference scan were taken of the same scene, there can always be incorrect associations for example due to moving objects, or due to objects which appear differently from different location (e.g. vertically nonuniform objects observed from a slightly tilted laser scanner). For this reason a heuristic error estimation usually yields more accurate results.

4.4.2 Heuristic Covariance Estimation

A simple heuristic error estimation approach was chosen to circumvent overoptimistic error estimates arising from incorrect associations. A preset covariance $\mathbf{C}_0$ matrix is scaled with the square of average absolute range residual from which an offset S_0 is subtracted:

$$\mathbf{C} = max\left((\frac{1}{n}\sum |\Delta r|)^2 - S_0, 1 \right) \mathbf{C}_0 \tag{4.35}$$

If the mean error is smaller than S_0 then $\mathbf{C} = \mathbf{C}_0$ to ensure, that the covariance estimate does not get too small. It is assumed that smaller range residuals are the result of better association and better scan matching results. However on a featureless corridors, one can have a small mean absolute range residual error, and a large along corridor error. Therefore different $\mathbf{C}_0$ are chosen for corridor like areas and for non-corridor like areas. For scans of non-corridor like areas, a diagonal covariance matrix is chosen. A non-diagonal covariance matrix is chosen for corridors which expresses the larger along corridor error. Classification of scans into corridors is done by calculating the variance of orientations (measured as angles of the normals) of line segments obtained by connecting neighboring points. If this variance is smaller than a threshold, then the scan is classified as a corridor. The orientation of the corridor necessary for the covariance matrix generation is estimated by calculating an angle histogram [Weiss and Puttkamer, 1995] from the line segment orientations. The angle perpendicular to the location

of the maximum of the histogram will correspond to the corridor orientation.

4.5 SLAM using Polar Scan Matching

A simple implementation of Kalman filter SLAM was programmed in C++ to evaluate the practical usability of the described scan matching method. As in [Bosse *et al.*, 2004] laser scanner poses are used as landmarks. With each landmark the associated laser scan is also stored. Each time the robot gets to a position which is further than one meter from the closest landmark, a new landmark is created. Each time the robot gets closer than 50cm and 15° to a landmark not updated in the previous step, an update of the landmark is attempted. Note that consecutive scans are not matched. This is because it is assumed that short term odometry of the robot when traveling on flat floor is much more accurate than scan matching.

When updating a landmark, the observation is obtained by scan matching. The laser measurement is passed to scan matching as the reference scan, and the scan stored with the landmark is passed as the current scan. The result of scan matching is the position of the landmark expressed in the laser's coordinate system at the robot's current position.

A few details follow next. Following the notation and equations of Davison [Davison, 1998], a new feature is appended to the state as

$$\mathbf{x}_{new} = \begin{bmatrix} \mathbf{x}_v \\ \mathbf{y}_1 \\ \vdots \\ \mathbf{y}_i \end{bmatrix} \tag{4.36}$$

$$\mathbf{P}_{new} = \begin{bmatrix} \mathbf{P}_{xx} & \mathbf{P}_{xy_1} & \cdots & \mathbf{P}_{xx}\frac{\partial \mathbf{y}_i}{\partial \mathbf{x}_v}^T \\ \mathbf{P}_{y_1x} & \mathbf{P}_{y_1y_1} & \cdots & \mathbf{P}_{y_1x}\frac{\partial \mathbf{y}_i}{\partial \mathbf{x}_v}^T \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}_{y_{i-1}x} & \mathbf{P}_{y_{i-1}y_1} & \cdots & \mathbf{P}_{y_{i-1}x}\frac{\partial \mathbf{y}_i}{\partial \mathbf{x}_v}^T \\ \frac{\partial \mathbf{y}_i}{\partial \mathbf{x}_v}\mathbf{P}_{xx} & \frac{\partial \mathbf{y}_i}{\partial \mathbf{x}_v}\mathbf{P}_{xy_1} & \cdots & \frac{\partial \mathbf{y}_i}{\partial \mathbf{x}_v}\mathbf{P}_{xx}\frac{\partial \mathbf{y}_i}{\partial \mathbf{x}_v}^T + \frac{\partial \mathbf{y}_i}{\partial \mathbf{h}}\mathbf{R}\frac{\partial \mathbf{y}_i}{\partial \mathbf{h}}^T \end{bmatrix} \tag{4.37}$$

Where $\mathbf{x}_v$ is the vehicle pose, $\mathbf{y}_n = [x_n, y_n, \theta_n]$ is the n-th landmark, y_i is new landmark in world frame, $\mathbf{h}$ is the measurement of the new landmark expressed in robot frame and $\mathbf{R}$ is the estimated covariance of $\mathbf{h}$. Since the measurement $\mathbf{h}$ corresponding to the new landmark $\mathbf{y}_i$ is simply the accurately known pose of the laser on the mobile robot, the measurement noise covariance $\mathbf{R}$ is a null matrix. The transformation of the measurement into world frame is the following:

$$x_i = x_v - y_L \sin(\theta_v) \tag{4.38}$$

$$y_i = y_v + y_L \cos(\theta_v) \tag{4.39}$$
$$\theta_i = \theta_v \tag{4.40}$$

where it is assumed that the laser scanner's X axis is parallel to the robot's X axis and that the laser scanner's center is placed to $(0, y_L)$ on the robot. The Jacobian $\frac{\partial y_i}{\partial x_v}$ is then the following:

$$\frac{\partial y_i}{\partial x_v} = \begin{bmatrix} 1 & 0 & -\cos(\theta_v) y_L \\ 0 & 1 & -\sin(\theta_v) y_L \\ 0 & 0 & 1 \end{bmatrix} \tag{4.41}$$

Next the specifics for update are described. The prediction $h_i = [x_{hi}, y_{hi}, \theta_i]^T$ of the i-th landmark, i.e. the i-th landmark expressed in the laser's frame is calculated as:

$$x_{hi} = (x_i - x_v)\cos(\theta_v) + (y_i - y_v)\sin(\theta_v) \tag{4.42}$$
$$y_{hi} = -(x_i - x_v)\sin(\theta_v) + (y_i - y_v)\cos(\theta_v) - y_L \tag{4.43}$$
$$\theta_{hi} = \theta_i - \theta_v \tag{4.44}$$

Then the Jacobian of h_i necessary for the update is:

$$\frac{\partial h_i}{\partial x} = \begin{bmatrix} \frac{\partial h_i}{\partial x_v} & 0^T & \cdots & 0^T & \frac{\partial h_i}{\partial y_i} & 0^T & \cdots \end{bmatrix} =$$
$$= \begin{bmatrix} -\cos\theta_v & -\sin\theta_v & -\sin(\theta_v)(x_i - x_v) + \cos(\theta_v)(y_i - y_v) & \cdots & \cos\theta_v & \sin\theta_v & 0 & \cdots \\ \sin\theta_v & -\cos\theta_v & -\cos(\theta_v)(x_i - x_v) - \sin(\theta_v)(y_i - y_v) & \cdots & -\sin\theta_v & \cos\theta_v & 0 & \cdots \\ 0 & 0 & -1 & \cdots & 0 & 0 & 1 & \cdots \end{bmatrix} \tag{4.45}$$

4.6 Experimental Results

The results of 4 experiments are presented where the performance of PSM, PSM-C (polar scan matching using Cartesian coordinates) and a simple implementation of ICP are compared. In the first experiment simulated laser scans are matched and evaluated. The remaining experiments use a SICK LMS 200 laser range finder at a 1° bearing resolution in indoor environments. In the second experiment, laser scan measurements are matched at 10 different scenes by positioning the laser manually in known relative poses and the results are compared with the known relative poses. In the third experiment, the areas of convergence for particular pairs of scans are investigated. The scan matching algorithms are evaluated in a SLAM experiment in the fourth experiment. The parameters used in all scan matching experiments are shown in tab. 4.1.

Every scan matching variant was stopped and divergence declared if the number of matches

PM_MAX_ERROR	100cm
PM_MAX_RANGE	1000cm
PM_MAX_ITER	30
PM_MIN_VALID_POINTS	40
PM_MAX_DIFF	20cm
C	70cm reduced to 10cm after 10 iterations

Table 4.1: Parameters used in scan matching during the experiment.

sank bellow PM_MIN_VALID_POINTS. The terminating condition for PSM and PSM-C was that either in 4 consecutive iterations

$$\varepsilon = |\Delta x_c[cm]| + |\Delta y_c[cm]| + |\Delta \theta_c[^\circ]| \tag{4.46}$$

was smaller than 1 or the maximum number of iterations PM_MAX_ITER has been reached. The need for a hard limit on the number of iterations is necessary, since PSM position estimate often drifts along corridors. Another reason for a hard limit is, that both PSM and PSM-C can enter a limit cycle. This limit cycle is not a problem however if the terminating condition is chosen as 1. In the case of ICP, terminating condition had to be chosen as $\varepsilon < 0.1$, because of the low convergence speed of ICP. In the case of $\varepsilon < 1$, ICP often terminated with a too large error. Due to the slow convergence speed, the maximum number of iterations was chosen 60 for ICP, which is twice as much as that for PSM.

In PSM one position estimated step was followed by one orientation estimation step. These 2 steps are considered as 2 iterations. In PSM-C 3 pose estimation steps are followed by 1 orientation step. These 4 steps were considered as 4 iterations. Note that this counting of iterations is different to [Lu and Milios, 1997] where one position estimation step followed by an orientation estimation step was considered as one iteration for IDC.

In the following results, all the run times were measured on a 900Mhz Celeron laptop.

4.6.1 Simple Implementation of ICP

In the ICP implementation, initially a median filter is applied to the range readings of the reference and current scan. Then in each iteration the projection of the current scan follows similarly to [Lu and Milios, 1997]. First each current scan point is transformed into the reference scan's Cartesian coordinate system. Current scan points are then checked if they are visible from the reference position by checking the order of points. This is followed by checking if two neighboring (in a bearing sense) reference or current scan points occlude the current scan point being checked. Occluded current scan points are then removed, if they are at least one meter further back than their interpolated reference counterparts. Current scan points not

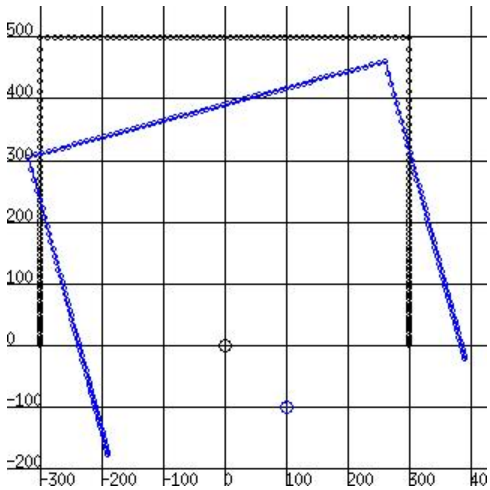


Figure 4.15: Current and reference scan prior to matching. Grid size in 1x1m.

in the field of view of the laser at the reference location are also removed. Note that none of the reference scan points are removed like in the projection filter in [Gutmann, 2000]. Reference scan points are not searched in this projection filter implementation, therefore this implementation is simpler but faster than of [Gutmann, 2000].

After scan projection, the implementation of the closest point association rule follows. For each remaining current scan point the closest reference scan point is sought in a $\pm 20°$ window. Unlike in [Lu and Milios, 1997], there is no interpolation between neighboring reference scan points, which increases speed, but reduces accuracy. Associated points with larger than PM_MAX_ERROR distance are ignored. Then the worst 20% percent of associations are found and excluded. From the remaining associated point pairs pose corrections are calculated using equations from [Lu and Milios, 1997] and the current pose is updated.

The ICP algorithm is simpler than that described in [Lu and Milios, 1997] since at the corresponding point search there is no interpolation between two neighboring reference scan points and the search window size is not reduced exponentially with the number of iterations. However, projection of the current scan with occlusion testing has been implemented without expensive searches and therefore it has been included at the beginning of each iteration. Performing an occlusion check in each iteration opposed to once at the beginning can increase the accuracy of the results in the case of large initial errors where many visible points may be removed incorrectly or many invisible points are left in the scan incorrectly.

4.6.2 Simulated Room

Figure 4.15 shows two simulated scans of a room. The scans were taken of the same location, but the x and y position of the current scan was altered by 100cm. Orientation was altered by 15°. Figure 4.16 shows the results after scan matching using PSM, PSM-C and ICP. Fig-

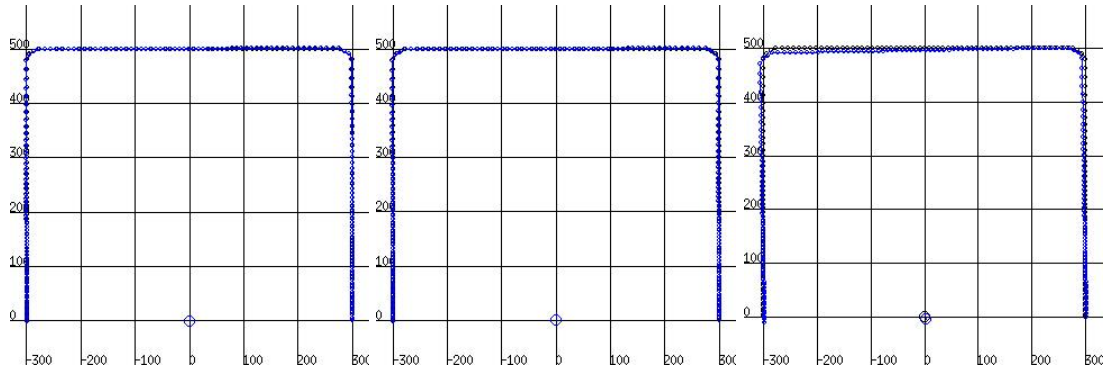
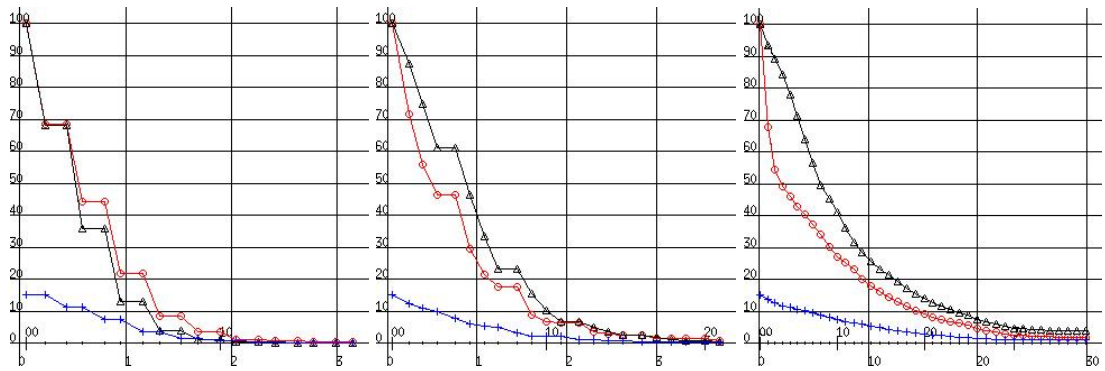


Figure 4.16: PSM, PSM-C and ICP results in the simulated experiment.Grid size in 1x1m.



a) b) c)

Figure 4.17: Evolution of x (circles),y (triangles) and orientation (crosses) error expressed in $[cm]$ and $[°]$, respectively of PSM, PSM-C and ICP in the simulated experiment. Grid size is 1ms x 10cm and 1ms x 10°, respectively. The horizontal resolution of the grid for ICP is 10ms. Iterations are marked with small vertical lines on the horizontal axis. Each 10-th iteration is marked with a longer vertical line.

	iterations	time [ms]	$\lvert\Delta x\rvert$[cm]	$\lvert\Delta y\rvert$[cm]	$\lvert\Delta\theta\rvert$ [°]
PSM	17	3.1	0.4	0.005	0.16
PSM-C	21	3.5	0.8	0.5	0.29
ICP	38	13.7	1.9	3.9	1

Table 4.2: Scan matching results of the simulated room.

ure 4.17 shows the evolution of errors. The final errors can be see in tab. 4.2.

From fig. 4.17 and tab. 4.2 it is clear that the PSM algorithm reached the most accurate result in the shortest time, and ICP was the slowest and least accurate.

4.6.3 Ground Truth Experiment

To determine how the polar scan matching algorithm variants cope with different types of environments, an experiment with ground truth information was conducted. On 4 corners of a 60x90cm plastic sheet, 4 Sick LMS 200 laser scanner outlines were drawn with different orientations. This sheet was then placed into different scenes ranging from rooms with different degrees of clutter to corridors. At each scene, laser scans were recorded from all 4 corners of the sheet, and matched against each other with initial positions and orientations deliberately set to 0 in the iterative procedure. The combinations of scans taken at corners which take part in the scan matching are shown in tab. 4.3. Ground truth values have been determined by first measuring the left bottom corners of each outline with respect to an accurate grid printed on the plastic sheet. Then the measurement of outline orientations followed using a ruler and the inverse tangent relationship. The determined orientations were also checked using a protractor. Finally a Matlab script was written which calculated the ground truth values by utilizing the relationship of the laser range finder's left bottom corner and the center of rotation of the mirror determined from the technical drawings of the Sick LMS's manual [SICK AG, 2002]. The carefully determined ground truth values for current scan poses in reference scan frames which also correspond to the initial errors are also displayed in tab. 4.3. From tab. 4.3 one can see, that the initial errors were up to 80cm displacement and up to 27° orientation. During the experiments the environment remained static.

A photo of each scene numbered 0-9 is shown in fig. 4.18. A matched current and reference scan from each scene is displayed in fig. 4.19 for PSM, fig. 4.21 for PSM-C and fig. 4.23 for ICP. The evolution of pose error is shown in fig. 4.20 for PSM, fig. 4.22 for PSM-C and fig. 4.24 for ICP. The displayed scans have all undergone median filtering. Only results for match 3 for each scene are displayed because match 3 contains a large initial error in displacement (77cm) and a large initial error in orientation ($-27°$) as can be seen from tab. 4.3. Absolute residual between ground truth and match results together with number of iterations and runtime are

shown in tables 4.5–4.7. There are 6 error vectors corresponding to each match for each scene. In tab. 4.5 “ERROR” denotes a situation, when scan matching stopped due to the lack of corresponding points and divergence was declared.

Scene 0 is a room with a small degree of clutter. Current and reference scans were quite similar, and the matching results are good. Scene 1 is in a more cluttered room where laser scans from different locations look different as one can see in fig. 4.19. The reason why the current scan differs from the reference scan so much is not clear. Perhaps the objects in the room were not uniform in the vertical direction and the laser beam is not a small spot or the laser was slightly tilted. The results for scene 1 (see tables 4.5–4.7, row 1) are not good for all 3 implementations, but they are still usable for example in a Kalman filter with an appropriate error estimate. In scene 2 the sheet was placed in front of a door to a corridor. The results are excellent. Scene 3 is a corridor without features. While the orientation error and the error in the cross corridor direction were quite small, the along corridor errors are large. PSM has the largest along corridor error of all, since the solution can drift in the direction of the corridor. With a proper error model (small orientation and cross corridor error, large along corridor error) the results are still useful when used with a Kalman filter. Scenes 4, 5 and 6 are similar to 3. In scene 4 PSM diverged once. When observing the results for 4, 5 and 6 in fig. 4.19, there are phantom readings appearing at the corridor ends, even though the real corridor ends were 30 meters away. The likely reason for the phantom readings is a slight tilt of the laser beams causing and readings from the floor to be obtained. Scene 7 is situated on the border of a room and a corridor. The results are good for all 3 scan matching methods. Scenes 8 and 9 were situated in a room. The results are quite good except those of ICP.

To compare the 3 scan matching approaches average of errors, number of iterations and run times were calculated and shown in tab 4.4. Average orientation error, iteration and run time were calculated for all scenes except for the scenes 4, 5, 6 with the large phantom objects. In the average displacement error calculation, all corridor like environments (3, 4, 5, 6) were left out, due to the large along corridor errors.

In the ground truth experiment, the implemented PSM and PSM-C clearly outperformed the implemented ICP. According to tab. 4.4 the performance of PSM and PSM-C are almost the same, with PSM being slightly more accurate but slower.

4.6.4 Convergence Map

The purpose of this experiment is to find those initial poses from which scan matching converges to an acceptable solution. Ideally one varies the initial position and orientation of the current scan in a systematic way and observes if the found solution is close enough to the true solution. Areas of convergence can be visualized by drawing the trajectory of the current scan into an image. To make visualization simpler just like in [Dudek and Jenkin, 2000] only the

match number	ref. scan recorded at corner	current scan recorded at corner	x [cm]	y [cm]	θ [°]
0	0	1	39.41	2.12	13
1	0	2	2.02	66.55	-14
2	0	3	38.84	66.99	12
3	1	2	-21.94	68.33	-27
4	1	3	14.04	68.33	-1
5	2	3	35.62	9.33	26

Table 4.3: Combinations of scans taken at different corners (numbered 0-3) of the plastic sheet for the ground truth experiment. These combinations marked as match number 0-5 were used for each scene. The pose of current scan with respect to reference scan is also shown. These poses correspond to the initial errors as well.

	iterations	time [ms]	orientation err. [°]	displacement err. [cm]
PSM	18.57	3.35	0.86	3.8
PSM-C	15.78	2.66	1.04	4.36
ICP	30.8	12.66	4.1	15.3

Table 4.4: Summary of average scan matching results in the ground truth experiment.

0	(0.9, 1.5, 0.3) 14, 2.7	(0.7, 0.4, 1.3) 16, 2.9	(1.1, 0.2, 0.1) 18, 3.3	(1.5, 0.4, 2.4) 20, 3.6	(0.6, 7.4, 0.2) 12, 2.2	(3.6, 0.1, 1.3) 12, 2.2
1	(5.1, 17.3, 5.5) 20, 3.7	(7.7, 13.1, 5.8) 28, 5.1	(0.4, 24.8, 8.3) 30, 5.5	(1.0, 1.5, 0.4) 27, 4.8	(0.3, 5.0, 0.6) 24, 4.3	(2.4, 0.5, 0.6) 12, 2.3
2	(0.4, 0.3, 0.3) 8, 1.5	(0.2, 0.6, 0.1) 24, 4.2	(0.5, 1.0, 0.3) 20, 3.6	(0.2, 0.9, 0.3) 20, 3.5	(0.2, 4.8, 0.3) 28, 4.9	(1.0, 2.7, 0.3) 12, 2.2
3	(9.7, 5.0, 0.2) 18, 3.2	(51.8, 25.3, 0.2) 12, 2.2	(22.1, 11.1, 0.3) 20, 3.6	(90.8, 19.4, 0.0) 30, 5.2	(24.3, 11.2, 0.1) 16, 2.9	(55.3, 46.3, 0.1) 30, 5.2
4	ERROR	(4.2, 47.9, 1.3) 20, 3.7	(0.9, 4.3, 0.0) 30, 5.1	(61.6, 160.3, 4.7) 30, 5.2	(73.8, 210.5, 1.5) 30, 4.9	(1.0, 6.1, 0.1) 30, 5.1
5	(0.6, 20.3, 0.4) 24, 4.4	(0.2, 24.0, 0.4) 30, 5.5	(1.3, 10.7, 0.4) 30, 5.4	(12.6, 49.3, 0.5) 30, 5.4	(3.6, 6.6, 0.9) 21, 3.8	(1.6, 4.6, 1.5) 19, 3.4
6	(1.4, 30.7, 0.3) 18, 3.3	(2.0, 63.0, 0.1) 16, 2.9	(2.7, 79.1, 0.2) 16, 3.0	(23.0, 85.2, 0.4) 30, 5.2	(21.8, 86.6, 0.3) 16, 3.0	(0.8, 4.7, 0.0) 9, 1.7
7	(0.2, 0.1, 0.0) 26, 4.7	(1.5, 0.2, 0.2) 16, 2.9	(0.1, 0.3, 0.1) 20, 3.6	(0.8, 2.6, 0.3) 18, 3.2	(0.9, 4.9, 0.1) 16, 2.9	(0.0, 0.6, 0.3) 15, 2.7
8	(0.7, 0.0, 0.0) 11, 2.0	(1.3, 2.1, 0.1) 22, 3.9	(0.1, 0.4, 0.3) 19, 3.3	(0.6, 0.6, 1.9) 22, 3.9	(0.0, 5.6, 0.9) 23, 4.0	(0.6, 0.4, 0.1) 12, 2.2
9	(3.7, 1.7, 0.8) 10, 1.9	(2.0, 0.4, 0.4) 18, 3.3	(1.4, 0.9, 0.2) 18, 3.3	(2.8, 3.0, 0.5) 18, 3.3	(1.6, 9.5, 0.7) 12, 2.2	(1.1, 1.6, 0.2) 13, 2.4

Table 4.5: Absolute errors in x[cm], y[cm], θ[°], number of iterations and runtime [ms] of the PSM algorithm in the experiments with ground truth.

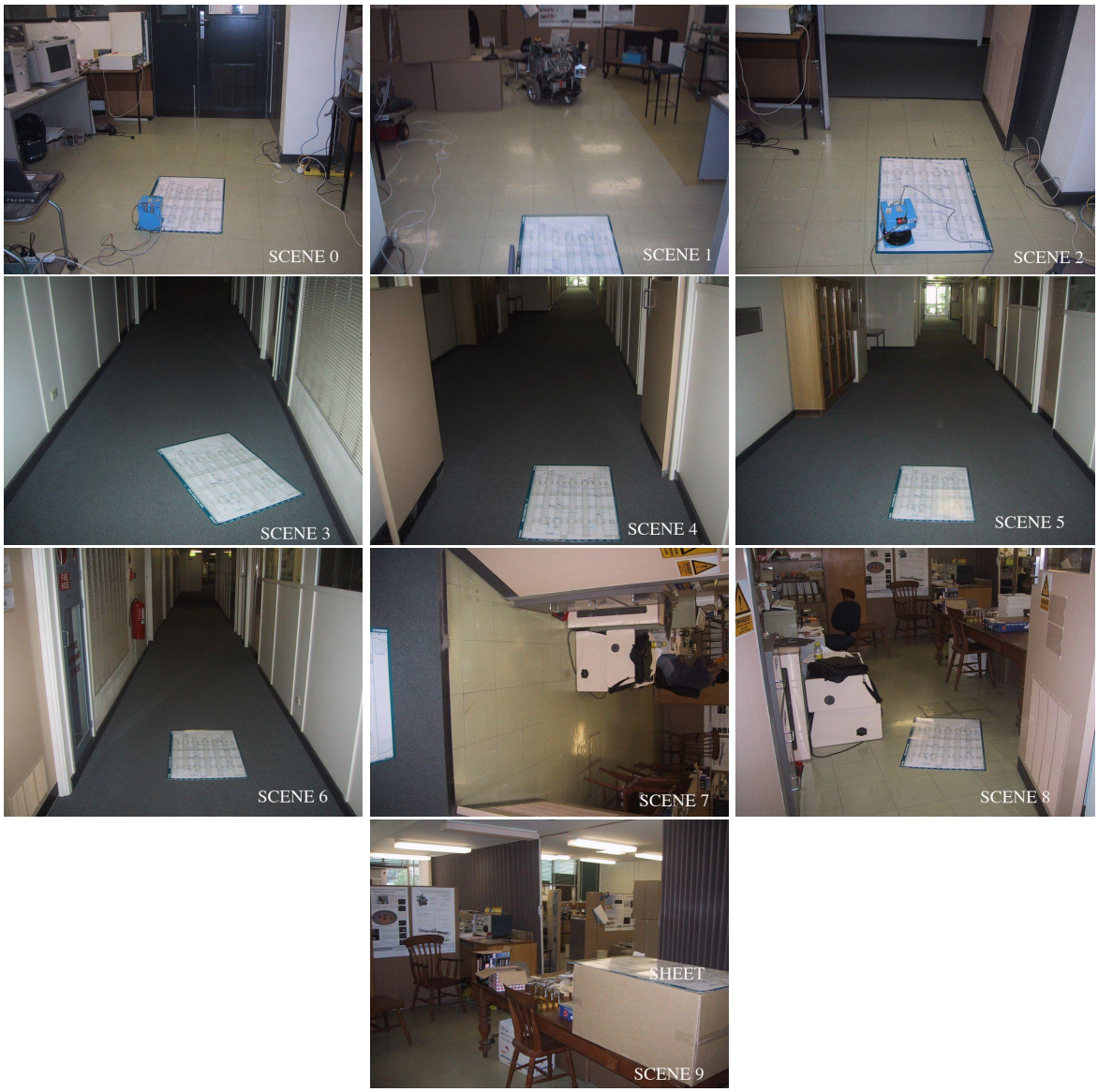


Figure 4.18: A photo of each scene with the plastic sheet in the foreground.

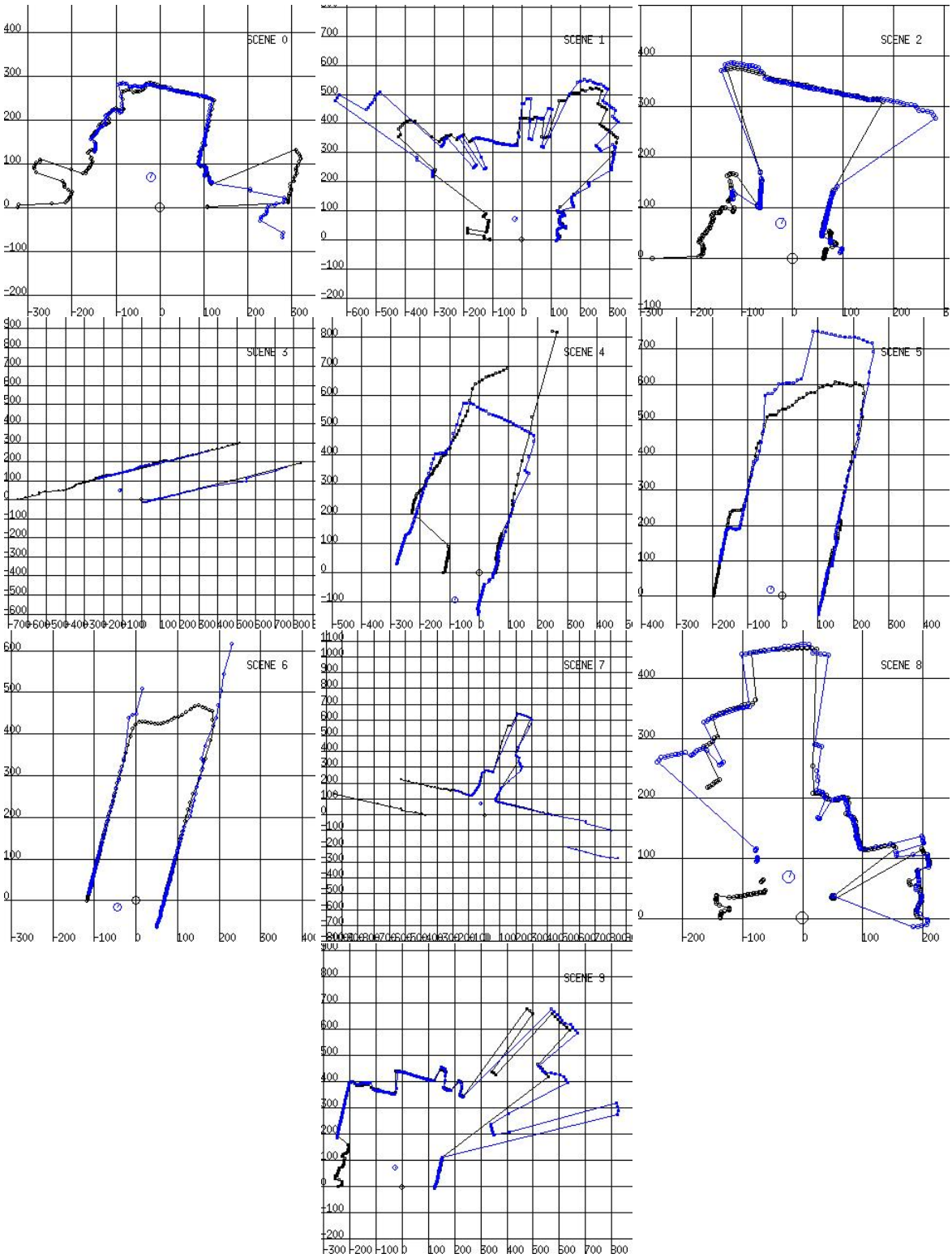


Figure 4.19: Scan match result for each scene for match number 3 in the experiment with ground truth using PSM.

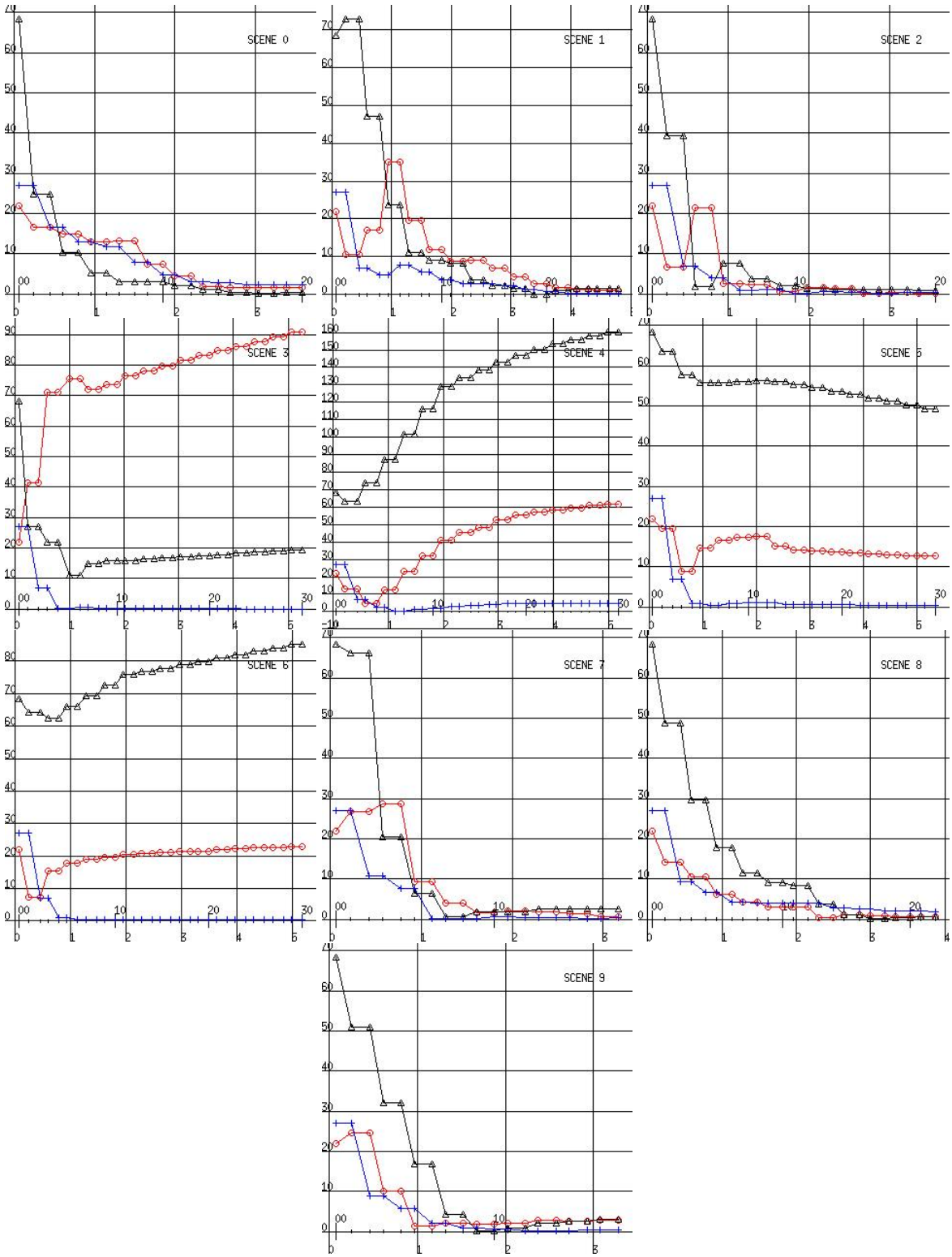


Figure 4.20: Match 3 scan match error evolution for each scene in the experiment with ground truth using PSM. Error in x (circles), y (triangles) and orientation (crosses) are expressed in $[cm]$ and $[^\circ]$, respectively. Grid size is 1ms x 10cm and 1ms x 10°, respectively. Iterations are marked with small vertical lines on the horizontal axis. Each 10-th iteration is marked with a longer vertical line.

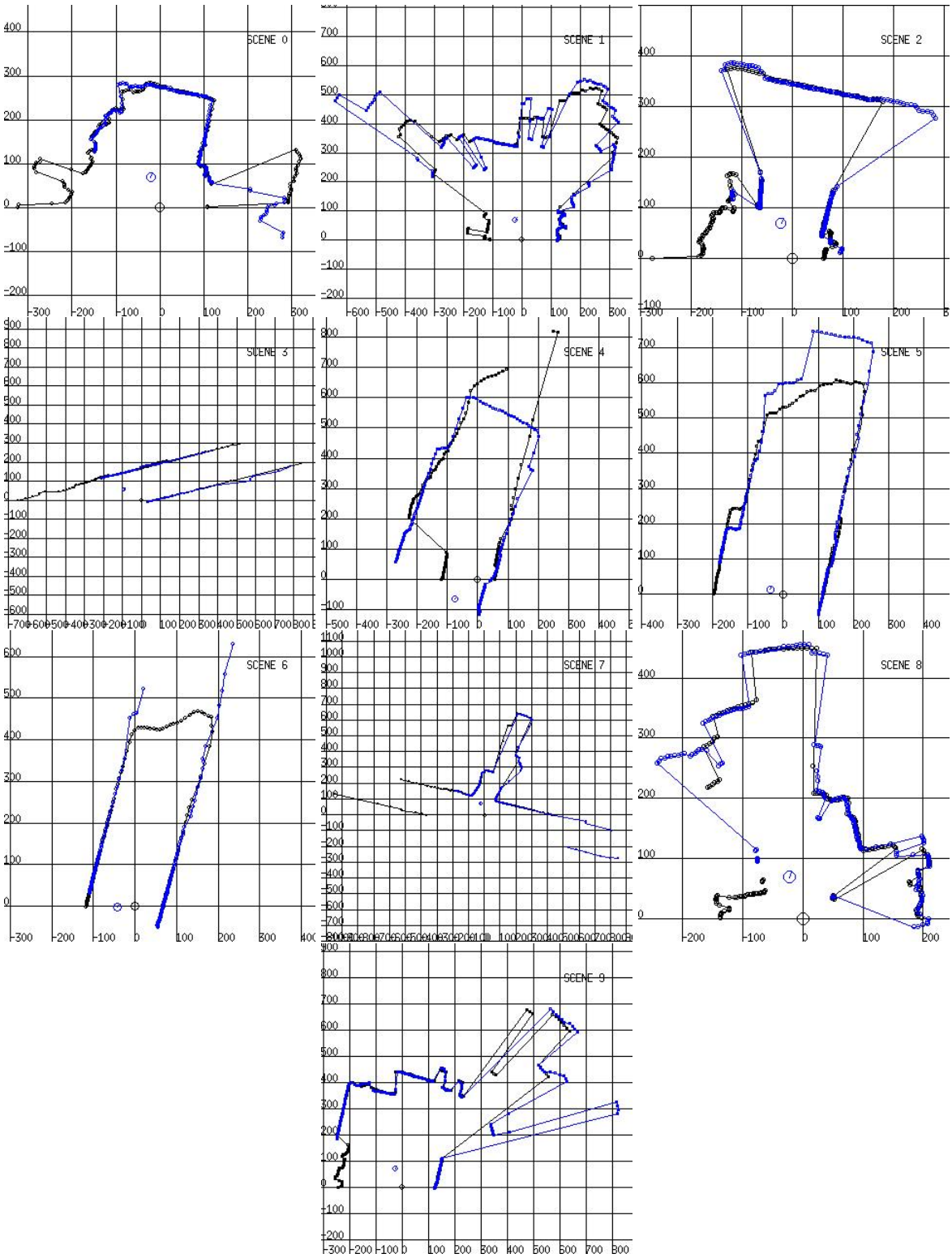


Figure 4.21: Scan match result for each scene for match number 3 in the experiment with ground truth using PSM-C.

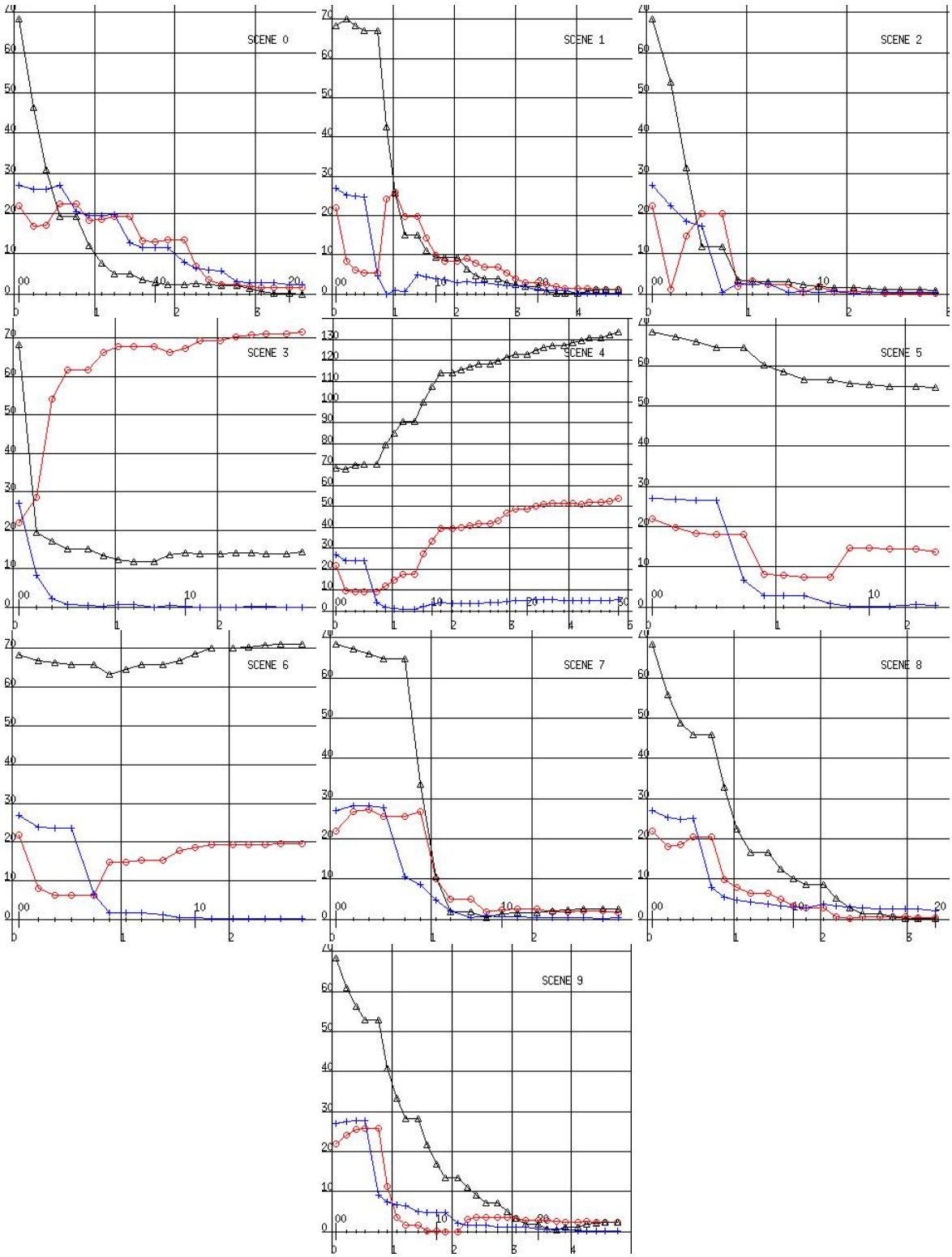


Figure 4.22: Match 3 scan match error evolution for each scene in the experiment with ground truth using PSM-C. Error in x (circles), y (triangles) and orientation (crosses) are expressed in $[cm]$ and $[^\circ]$, respectively. Grid size is 1ms x 10cm and 1ms x 10°, respectively. Iterations are marked with small vertical lines on the horizontal axis. Each 10-th iteration is marked with a longer vertical line.

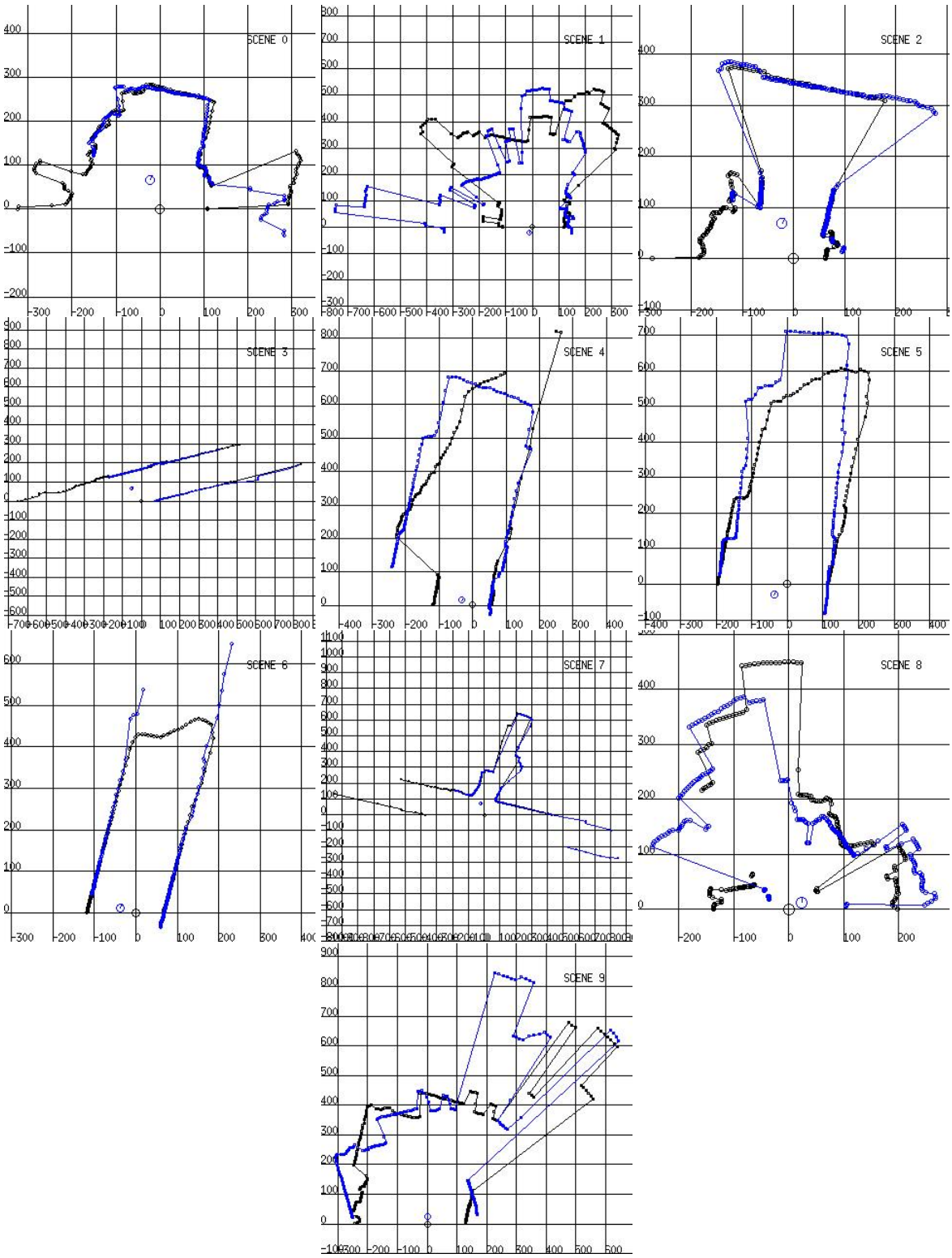


Figure 4.23: Scan match result for each scene for match number 3 in the experiment with ground truth using ICP.

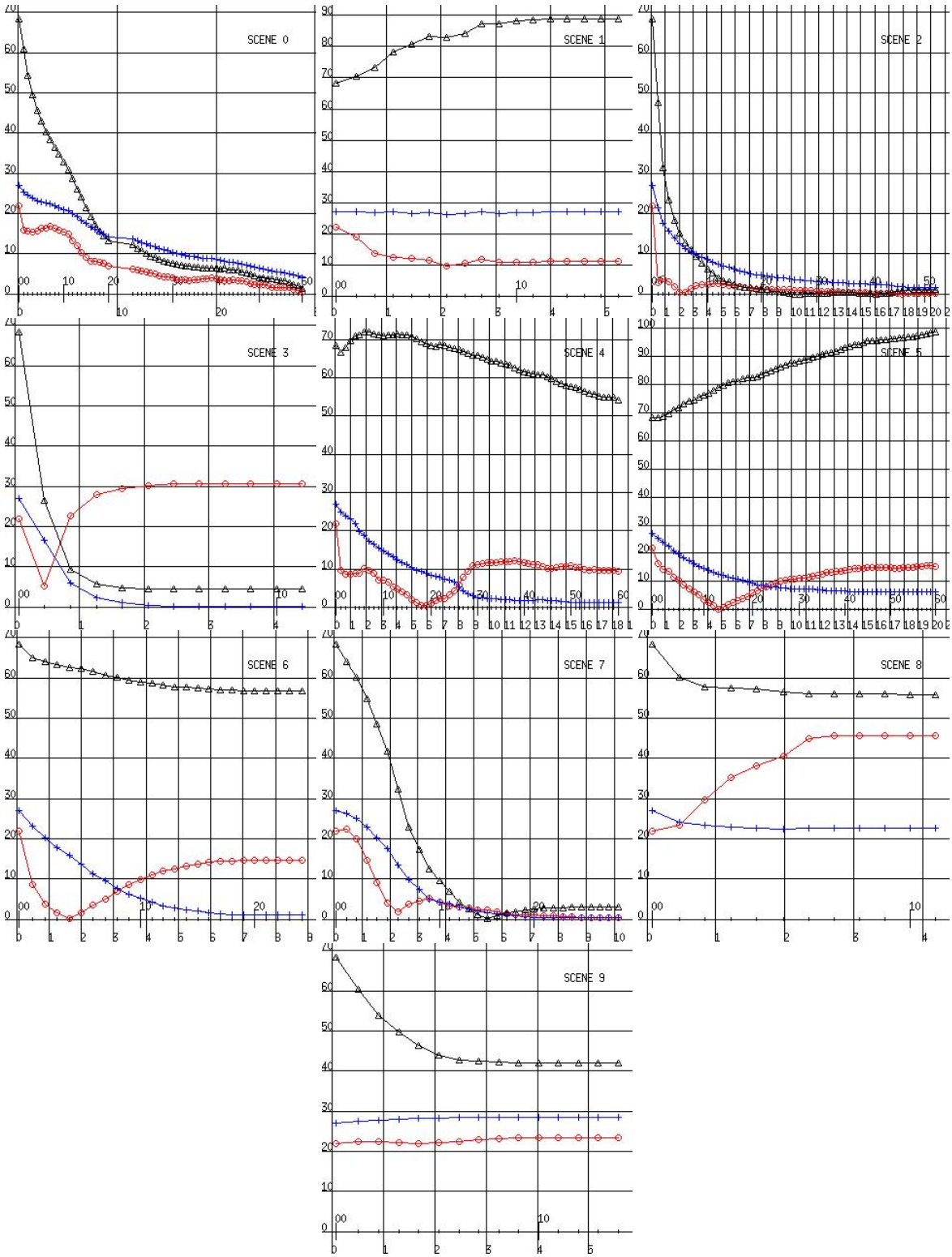


Figure 4.24: Match 3 scan match error evolution for each scene in the experiment with ground truth using ICP. Error in x (circles), y (triangles) and orientation (crosses) are expressed in $[cm]$ and $[^\circ]$, respectively. Grid size is 1ms x 10cm and 1ms x 10°, respectively. The horizontal resolution of the grid for scene 0 is 10ms. Iterations are marked with small vertical lines on the horizontal axis. Each 10-th iteration is marked with a longer vertical line.

0	(0.7, 1.4, 0.2) 13, 2.3	(0.8, 0.4, 1.3) 20, 3.5	(1.0, 0.3, 0.2) 17, 2.9	(1.6, 0.0, 2.4) 21, 3.6	(0.8, 6.5, 0.4) 14, 2.4	(3.0, 0.3, 1.1) 13, 2.3
1	(7.7, 19.3, 5.4) 12, 2.1	(7.4, 15.0, 6.2) 30, 5.0	(1.4, 26.7, 8.5) 30, 5.0	(1.2, 1.3, 0.3) 28, 4.6	(4.9, 16.3, 2.8) 13, 2.2	(1.0, 0.1, 0.4) 13, 2.2
2	(0.4, 0.1, 0.3) 9, 1.6	(0.8, 0.5, 0.1) 13, 2.2	(0.3, 0.7, 0.4) 13, 2.2	(0.3, 1.0, 0.3) 17, 2.8	(0.2, 4.6, 0.4) 13, 2.2	(1.0, 2.5, 0.3) 13, 2.2
3	(13.2, 6.9, 0.3) 9, 1.5	(49.4, 24.0, 0.2) 5, 0.9	(34.1, 17.2, 0.2) 6, 1.0	(71.5, 14.5, 0.0) 17, 2.8	(23.2, 10.9, 0.0) 6, 1.0	(29.1, 24.7, 0.1) 9, 1.5
4	(2.6, 44.4, 2.5) 17, 2.8	(6.0, 57.3, 1.6) 18, 3.1	(0.9, 1.5, 0.0) 30, 4.8	(53.8, 133.5, 5.3) 30, 4.9	(65.4, 169.9, 5.8) 30, 4.7	(6.1, 23.8, 0.2) 16, 2.6
5	(1.2, 17.8, 0.6) 17, 2.9	(0.1, 39.7, 0.5) 17, 2.9	(0.9, 25.8, 0.4) 21, 3.7	(13.8, 54.6, 0.4) 13, 2.2	(3.2, 2.8, 1.3) 26, 4.4	(0.9, 0.5, 0.7) 17, 2.9
6	(1.3, 24.1, 0.4) 17, 3.0	(1.5, 58.1, 0.4) 17, 2.8	(2.7, 74.1, 0.0) 13, 2.3	(19.6, 71.1, 0.3) 16, 2.6	(20.3, 79.9, 0.1) 17, 2.9	(1.3, 4.3, 0.3) 9, 1.6
7	(0.6, 0.1, 0.1) 17, 2.8	(0.9, 0.3, 0.1) 13, 2.2	(0.6, 0.0, 0.1) 14, 2.3	(1.5, 2.5, 0.3) 17, 2.9	(1.2, 4.9, 0.2) 13, 2.2	(0.2, 0.8, 0.1) 16, 2.7
8	(0.6, 0.7, 0.1) 13, 2.2	(1.1, 2.0, 0.3) 21, 3.4	(0.1, 0.3, 0.2) 20, 3.3	(0.5, 0.1, 2.2) 20, 3.3	(8.5, 13.3, 6.6) 13, 2.2	(0.3, 0.1, 0.0) 17, 2.9
9	(1.5, 1.4, 0.6) 13, 2.3	(1.5, 1.1, 0.2) 24, 4.1	(1.3, 2.5, 0.0) 13, 2.3	(2.3, 2.4, 0.1) 28, 4.7	(0.2, 5.2, 0.5) 20, 3.4	(0.8, 1.4, 0.2) 17, 2.9

Table 4.6: Absolute errors in x[cm], y[cm], $\theta[^\circ]$, number of iterations and runtime [ms] of the PSM-C algorithm in the experiments with ground truth.

0	(0.4, 0.1, 0.5) 56, 24.3	(0.2, 0.5, 1.3) 60, 28.1	(0.9, 0.2, 0.4) 22, 10.5	(0.4, 1.5, 4.1) 60, 27.3	(0.7, 5.3, 0.7) 43, 20.2	(1.4, 0.8, 1.0) 60, 25.9
1	(7.2, 16.5, 6.4) 18, 7.5	(2.1, 1.6, 1.0) 38, 14.3	(13.0, 29.9, 9.6) 13, 5.0	(11.0, 88.9, 27.0) 16, 5.3	(13.5, 53.3, 0.6) 16, 6.3	(24.7, 5.5, 22.6) 60, 21.4
2	(0.2, 1.1, 1.3) 25, 10.9	(0.1, 0.6, 0.4) 55, 22.2	(1.2, 1.7, 1.8) 42, 17.4	(0.1, 1.1, 1.6) 52, 20.7	(1.3, 5.4, 1.1) 38, 15.9	(2.1, 4.4, 4.6) 35, 14.1
3	(3.0, 1.3, 0.2) 8, 3.3	(40.5, 19.9, 0.1) 21, 9.1	(37.8, 19.6, 0.3) 9, 3.9	(30.7, 4.5, 0.2) 11, 4.6	(30.5, 12.8, 0.0) 8, 3.6	(6.9, 5.7, 0.1) 23, 9.2
4	(0.5, 2.7, 1.3) 21, 8.3	(5.5, 61.3, 1.5) 30, 10.1	(6.2, 55.5, 1.5) 20, 7.0	(9.4, 54.3, 1.4) 60, 18.6	(10.7, 70.1, 1.3) 60, 20.8	(5.8, 22.4, 0.1) 36, 13.1
5	(0.1, 1.4, 0.1) 35, 13.7	(7.8, 89.4, 4.3) 60, 22.2	(0.6, 47.4, 0.4) 52, 21.9	(15.4, 98.7, 6.3) 60, 20.7	(14.6, 67.7, 0.0) 9, 4.1	(0.3, 2.2, 0.1) 47, 17.5
6	(0.1, 1.4, 0.1) 23, 9.9	(1.6, 65.4, 0.5) 35, 13.3	(2.4, 63.7, 0.1) 19, 7.6	(14.7, 56.7, 1.1) 24, 9.0	(17.0, 68.6, 0.2) 12, 5.2	(0.6, 0.5, 0.4) 27, 10.1
7	(0.2, 0.1, 0.0) 19, 7.4	(0.4, 0.0, 0.1) 23, 8.7	(0.1, 0.2, 0.3) 20, 7.9	(0.5, 3.1, 0.3) 29, 10.3	(1.6, 6.5, 3.0) 16, 6.8	(0.0, 0.2, 0.6) 32, 11.7
8	(1.9, 0.3, 2.0) 28, 10.9	(0.2, 0.1, 0.5) 23, 8.8	(28.0, 89.1, 11.6) 18, 6.5	(45.5, 55.9, 22.6) 11, 4.3	(2.1, 7.2, 1.9) 37, 15.2	(0.3, 0.4, 0.7) 56, 23.1
9	(1.1, 0.8, 0.8) 43, 17.9	(19.5, 55.6, 11.8) 21, 9.4	(1.0, 0.4, 0.6) 36, 14.6	(23.4, 42.0, 28.6) 14, 5.7	(0.5, 4.8, 0.4) 32, 13.7	(0.4, 1.8, 1.4) 48, 17.7

Table 4.7: Absolute errors in x[cm], y[cm], $\theta[^\circ]$, number of iterations and runtime [ms] of the ICP algorithm in the experiments with ground truth.

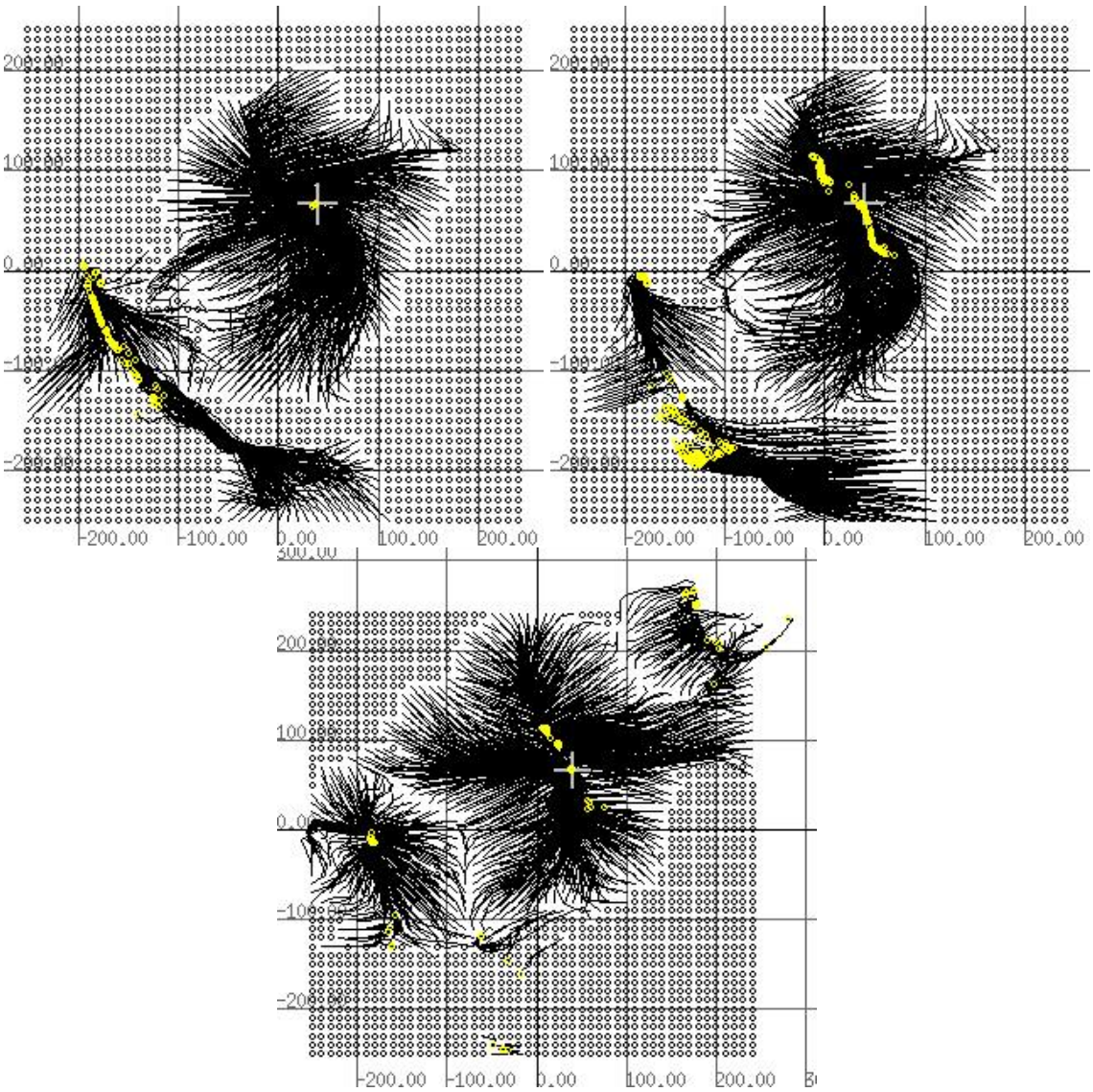


Figure 4.25: PSM, PSM-C and ICP convergence maps.

scene	PSM $[m^2]$	PSM-C $[m^2]$	ICP $[m^2]$
0	6.68	6.53	**7.18**
1	0.00	0.23	**2.05**
2	**4.38**	**4.38**	4.34
3	**0.52**	0.42	0.42
4	**0.81**	0.56	0.66
5	**3.10**	2.05	1.57
6	0.10	0.26	**0.36**
7	**3.71**	3.06	2.92
8	**3.60**	3.59	3.45
9	**7.01**	5.81	5.09
average	**2.99**	2.69	2.80

Table 4.8: Convergence areas for match 2 of all scenes for PSM, PSM-C and ICP.

initial pose was changed.

Scan pairs from scenes 0-9, match 2 were selected for the convergence map experiment. Match 2 was selected in this experiment because the point of view for the current and reference scan differed the most from all matches. The initial orientation error was always 12°. The initial position varied from -250cm to 250cm for x and y in 10cm increments. The resulting convergence plots for scene 9 can be seen in fig. 4.25. Dark circles represent initial positions where the scan matching algorithms failed for the lack of associated points. Light colored circles represent final positions. Black lines correspond to the trajectories of the scans. Gray crosses mark the correct current scan positions.

When examining fig. 4.25, one has to keep in mind that in all scan matching implementations, associations having a larger error than one meter were discarded. To get an objective value for the performance of the implementations, the total number of matches and the number of successful matches were counted. Successful matches were matches with less than 10cm error in position and 2° in orientation. The total number of scan matching trials was 2500 which corresponds to an area of $25m^2$. PSM had 701 successful matches which means that it converged to the correct solution from around a $7m^2$ area. PSM-C had 581, giving $5.8m^2$. ICP had 509 correct ones which corresponds to $5.09m^2$.

A non-graphical representation of all the results can be seen in tab. 4.8. From this table one can observe that PSM has the largest average area of convergence followed by ICP and PSM-C. However the differences of the averages are small. One can also observe, that the area of convergence of PSM was in 7 cases larger than that of ICP. Note that the small areas of convergence from scenes 3-6 were due to the corridor like character of the scenes, where the lack of features limit the accuracy of scan matching results in the along corridor direction.

From this experiment one can conclude, that on average the implemented PSM converged

from slightly larger area than ICP when using the dataset for scene 0-9, match 2. PSM-C performed slightly worse than ICP.

4.6.5 SLAM

The raw data set used in the Kalman filter SLAM is shown in fig. 4.26. The structures in the middle of the two rooms on the left are office cubicles. The third room is a seminar room filled with tables and chairs. The table and chair legs are represented by a randomly distributed point cloud. The robot was equipped with one SICK LMS 200 and odometry and started from the corridor intersection between the 2 rooms on the left. It visited the left room, and after one loop, it proceeded through the corridor to the middle room where it performed a large and a small loop and continued to the third (seminar) room. In the third room the robot was twice driven over a 1.5cm high cable protector on the floor at 40cm/s and at 20cm/s speed. After the visit to the third room the robot returned to its initial location from which it traveled to the far end of the corridor, went around a loop and came back. During the traversal of the environment, no less than 10 people walked in the view of the laser scanner and some doors were opened and closed. Considering the presence of walking people, repetitive cubicles, long corridors and 2 collisions with an obstacle on the floor, this dataset is not the most ideal for mapping. This experiment is the same as the one described in section 3.8.4 except the processing is done using SLAM with scan matching.

The SLAM results are shown in fig. 4.26. The SLAM results are significantly better than those from odometry only (fig. 4.26, top). Consecutive laser scans were not matched against each other, since it was assumed that over short distances odometry is more accurate than scan matching. PSM performed best on this data set followed by PSM-C, and worst with the ICP implementation. Note that the odometry of the robot was reasonably calibrated. This was necessary to be able to perform loop closing in the repetitive cubicle environment of the second room without the implementation of special loop closing algorithms.

In the C++ implementation of scan matching and SLAM, the 20 minutes worth of data consisting of 46×10^3 scans and 12×10^4 odometry readings took about 2 minutes to process on a 900MHz Celeron laptop for all variants. There were 100 successful (no divergence) scan matches for PSM SLAM variant with an average of 3.1ms scan matching time. There were 100 scan matches with PSM-C, with 2.1ms average time. ICP was successfully used 65 times with an average of 12.6ms.

The dataset used in this SLAM experiment was used also in chapter 3 for SLAM using laser and advanced sonar. When comparing the map created using SLAM with PSM (fig. 4.26) with the results in fig. 3.13 from SLAM with fusion of laser and advanced sonar, one can notice that laser scans are better aligned in fig. 3.13, however SLAM with sonar and laser could not cope well with the error introduced at the collision of the robot with the cable protector which

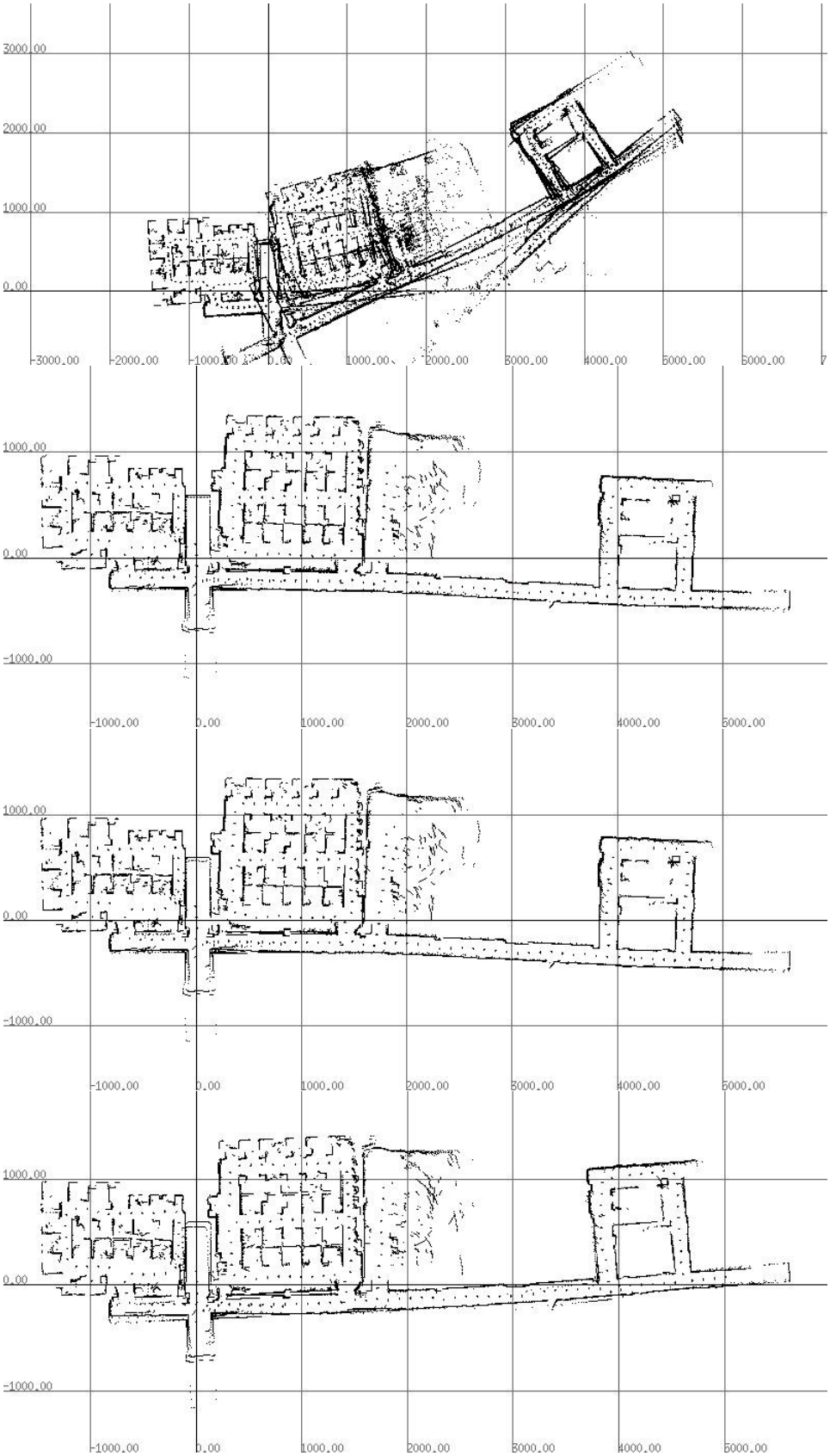


Figure 4.26: Maps resulting from odometry only, SLAM with PSM, PSM-C and ICP. Grid size is 10x10m.

resulted in a curved corridor.

The advantages of performing SLAM with PSM over SLAM with fusion of advanced sonar and laser measurements are the following:

- Since laser scans are in general more distinguishable than line segments or point features, association can be made more robust due to easier detection of failure.
- Scan matching may converge from an initial error in the order of meters and tens of degrees. When performing SLAM with advanced sonar and laser, such errors in the relative pose of the robot with respect to neighboring landmarks would likely result in an invalid association or in a robot unable to update its pose due to many features falling into measurement validation gates. SLAM with scan matching is more robust with respect to error in the robot pose.
- Since the robot is allowed to have a larger pose estimate error with respect to neighboring landmarks, the SLAM filter can be run at lower update rate which makes scan matching SLAM more computationally efficient.
- Since there is no need to extract features, SLAM with scan matching may work in environments lacking lines, points and corners necessary for the functioning of SLAM with advanced sonar and laser.
- Since SLAM with scan matching is implemented using only a laser scanner and an odometry sensor, in environments where a laser scanner is sufficient for navigation a robot performing SLAM with scan matching is cheaper. Less sensors in this case also mean easier coding and debugging. SLAM with advanced sonar and laser requires two extra advanced sonar arrays which make a robot more expensive and require more complicated coding and debugging.

The disadvantages of performing SLAM with PSM over SLAM with fusion of advanced sonar and laser measurements are the following:

- Scan matching can get stuck in local minima.
- Unlike in SLAM with advanced sonar and laser, moving objects may have adverse effects on the scan matching results.
- In corridors, scan matching may correct the robot's pose in the cross corridor direction. This can result in a robot getting lost in long corridors. In SLAM using laser and advanced sonar, the existence of small features on the corridor such as doorjambs or wall moldings are sufficient for correct operation. Therefore a robot navigating with SLAM using scan matching may get lost while a robot with SLAM using advanced sonar and laser still operates properly.

- SLAM with scan matching works best using 360° scans. If a robot with a laser scanner providing only 180° coverage heads down a corridor, then on the way back it will not be able to update the landmarks created in the way there due to the lack of sufficient overlap. If the robot uses 360° scans, the lack of overlap problem does not occur.

4.7 Extensions to 3D

3D scan matching has gained popularity in recent years. Therefore it would be interesting to know if PSM could be adapted to 3D. The projection filter and position estimation would be still of $O(mn)$ and $O(n)$ complexities, respectively. Estimation of the 3 orientation angles if done sequentially would still result in $O(kn)$ complexity, where k is proportional to the number of range readings per unit angle. Even though 3D scan matching with a modified PSM is an exciting problem, due to the lack of time we have to consider it as possible future work.

4.8 Conclusions and Future Work

In this chapter a laser scan matching method is proposed which works with the laser measurements in their native, polar form. The polar scan matching (PSM) approach belongs to the class of point to point matching algorithms. PSM takes advantage of the structure of laser scanner measurements by functioning in the laser scanner's polar coordinate system. The direct use of range and bearing measurements coupled with a matching bearing association rule and a weighted range residual minimization, results in an $O(n)$ complexity pose estimation approach and an $O(kn)$ complexity orientation estimation approach, opposed to the $O(n^2)$ complexity of the popular IDC algorithm by Lu and Milios [1997]. In $O(kn)$ of the orientation estimation approach, k is proportional to the angular resolution of the laser scans. Opposed to the $O(n^2)$ projection filter of [Gutmann, 2000], preprocessing of scans is done also with $O(n)$ complexity if there are no occlusions in the current scan when viewed from the reference scan's position. A variant of PSM, PSM-C is also introduced where the translation estimation step of PSM is replaced with a weighted variant of the pose estimation equations from [Lu and Milios, 1997]. In PSM-C due to the use of the matching bearing rule, equations from [Lu and Milios, 1997] also minimize the sum of square range residuals. For comparison, a basic ICP has also been implemented.

Simulation of matching scans in a room demonstrates that the current scan pose error decreases more quickly with PSM and PSM-C to a small value, than with ICP. Scan matching experiments were also performed with a SICK LMS 200 in a variety of environments. Comparison of the results with ground truth revealed that in the tests, the performance of PSM and PSM-C surpasses that of ICP in speed and accuracy. However when matching corridors, a

position drift in the direction of the corridor has been observed with PSM. This drift was not observed when using PSM-C or ICP. A comparison of areas of convergence for PSM, PSM-C and ICP were also performed. It was found, that PSM converged to the correct solution from slightly larger area than PSM-C and ICP.

A simple heuristic scan matching error model has been proposed where first scans are classified into rooms and corridors. For rooms a diagonal covariance matrix is scaled by the sum of square errors. For corridors, the xy sub-matrix (of the covariance matrix) with a large error in the x direction and small error in the y direction is rotated first, so that the large error aligns with the direction of the corridor. Then the resulting covariance matrix is scaled by the average square error.

The usability of the proposed scan matching approaches has been tested by performing Kalman filter SLAM with scan matching in a static environment. The maps created by PSM and PSM-C are better than that of ICP as shown in fig. 4.26. In fig. 4.26, the quality of the maps can be judged by the straightness of the corridor and by the presence of walls with multiple representations.

A map resulting from SLAM with PSM has been compared to a map resulting from SLAM using laser and advanced sonar fusion and path tracking. Laser scans of the map resulting from SLAM with PSM were less aligned than those from SLAM using advanced sonar and laser. In general SLAM with PSM can experience problems when there is significant motion in the scans and also on corridors with not enough features to correct the along corridor error of the robot. On the other hand SLAM with PSM compared to SLAM with advanced sonar and laser can handle larger robot errors relative to neighboring landmarks, needs less frequent landmark updates, is simpler and it does not extract features from the environment.

As future work, the tracking and tagging of moving objects could be considered. Also the real advantage of the efficient PSM over $O(n^2)$ methods becomes more apparent when the number of points is large. One such case is in 3D scan matching. The modification of PSM for 3D scan matching is also considered as future work.

Chapter 5

Conclusions and Future Work

This thesis has presented the application of laser range finders and advanced sonar arrays to mobile robot navigation, specifically in the area of simultaneous localization and mapping. A range of topics has been discussed including feature extraction and error modeling for laser scans, laser scan matching, laser and advanced sonar fusion and simultaneous localization and mapping using scan matching and fused laser and sonar measurements. In the following, the conclusions and recommendations for future work are structured according to the contributions section of the introduction (chapter 1):

- *Approaches for fitting lines and right angle corners to laser range finder measurements directly in the laser range finder's native polar coordinate system.* When measurements of sensors are used in a statistical framework such as the Kalman filter, accurate measurement error models are needed. An error estimate that is too conservative can lead to a waste of information and overly optimistic error estimates can cause divergence. In chapter 2 fitting line segments and right angle corners to laser range finder measurements have been investigated. The formulation of the corner fitting approach implicitly associates measurements to the two sides of corners. The corner and line fitting approaches utilize an iterative minimization of sum of square range residuals which result in simple and accurate random error estimates.

  As future work, the reduction of computation time of the line fitting algorithm can be considered. Equation (2.22) provides line parameter estimates and contains the term $\mathbf{H}^{\mathbf{T}}\mathbf{H}$, where $\mathbf{H}$ is a $n \times 2$ matrix. It is possible to speed up the calculation of $\mathbf{H}^{\mathbf{T}}\mathbf{H}$ by approximating sums with the solutions of definite integrals as in section 2.3.4. Given that the line fitting approach is an iterative process, final line parameter errors introduced by the approximation would be negligible.

- *Comprehensive line error model.* Unlike in other works where it is assumed that the only source of line parameter errors are random errors in laser range finder measurements,

here the effects of 8 error sources, most of them systematic have been considered. The error sources are: constant range bias, range bias changing linearly with range, range bias changing with incidence angle of laser beam and target surface normal, range bias due to quantization, white Gaussian noise in the range measurements, laser plane misalignment, finite time for a mirror to rotate through a scan and finally time registration error. The line error models have been experimentally evaluated using Sick PLS and LMS laser range finders. Before testing the line error models, range error models for the PLS and LMS have been experimentally calibrated. In the random error model evaluation, thousands of scans have been taken of lines from different ranges and distances. Estimated covariances of lines have then been compared to the measured ones. An excellent match of measured and estimated line parameter covariances has been found in the case of the PLS indicating correctness of the error models. In the case of the LMS, the match has been satisfactory. Systematic line orientation errors have been indirectly evaluated through taking thousands of measurements of a right angle corner from different positions and orientations. Lines have been fitted to the arms of corner and the average line orientation estimates have been compared to 90°. The systematic line orientation error estimates have been proved satisfactory for both PLS and LMS. Most researchers model line parameter errors from laser range measurements as random quantities. However, in our experiments there have been cases where systematic orientation errors have been observed that are larger than the standard deviation of random orientation errors. This observation justified the efforts on range and line systematic error modeling.

As future work the development of a closed form approximation for line parameter errors due to finite time necessary for a mirror to rotate through a scan should be considered. The solution should be closed form in the sense, that the approximation does not contain sums over all measurements. In the future, it would be interesting to extend the evaluation of the error models to laser range finders other than the common Sick PLS and LMS. Good candidates are for example the range finders of the Japanese company HOKUYO. These sensors have recently started gaining popularity for their small size, low energy consumption and low price. Even though they do not measure time of flight directly using timers, there are likely some components of the error model which can be applied.

- *Advanced sonar and laser synergy with application to SLAM.* Having proved line and corner fitting approaches with the corresponding error models could make one think that one is ready for a Kalman filter implementation of simultaneous localization and mapping using a laser range finder. However there are environments where laser line and corner features are not enough for SLAM to be successful for example long corridors where there are not enough features to correct the robot's along-corridor error, and

perhaps environments with a lot of glass walls where the laser measurements are not reliable. These problems lead to the work behind chapter 3 where the synergy of laser range finder and advanced sonar array measurements are discussed. The advanced sonar arrays are unique sonar sensors in that they measure range and bearing to targets classified as planes, corners and edges. In the synergy scheme devised in this project sonar measurements help to segment laser scans into lines and right angle corners, laser scans help to reject spurious sonar measurements and to select good sonar point reflectors. Sonar and laser measurements of the same object are then fused by calculating an average of the measurements weighted by their information matrix. Segmenting laser measurements by using sonar readings is helpful for example when different objects such as a door and a wall cannot be separated because of noise in the laser measurements. However by using sonar reflections from the border of the two objects, in our case from the door jamb, correct segmentation of the line segments can be achieved while increasing accuracy. The sonar recognizes only right angle corners as corner targets, this information is also used in the laser segmentation. Clusters of sonar point measurements terminating a laser line are chosen as good sonar point reflectors. The implemented advanced sonar and laser fusion scheme has been tested using a simple implementation of Kalman filter SLAM. It has been shown, that with the fusion scheme the mapping process did not diverge on long corridors, where laser only SLAM implementation failed due to the lack of features for along-corridor error correction. Note that no other work has considered laser and advanced sonar fusion for simultaneous localization and mapping. There is one other example for fusion of advanced sonar and laser for mapping in the literature [Vandorpe *et al.*, 1996], where only range and bearing information of advanced sonar measurements were used by transferring them together with the laser measurements into a grid map.

As future work implementation issues of SLAM should be first improved. Currently new line map features can be created only if the line segments are measured by the laser. It would be useful to allow the creation of line map features based only on advanced sonar measurements to enable mapping of glass walls. In the future it would also be useful to mount the advanced sonar sensor and laser range finder onto a smaller robot which could be transported to different environments for mapping purposes. Performing simultaneous localization and mapping experiments in different environments such as shopping centers, homes and sport halls would provide constant motivation to improve the laser and advanced sonar synergy and SLAM implementations. For environments where SLAM using laser and advanced sonar fails, the addition of vision can be considered. For example, vision could solve ambiguous situations which sometimes arise when trying to close large loops.

From the general SLAM point of view it would be interesting to investigate how to

handle situations where SLAM diverges. The interesting question is if it is possible to repair incorrect maps without accessing past measurements.

- *Occupancy grid generation in feature based SLAM.* Having a sparse feature map might help with localization, but for path planning a denser grid map is preferable. By knowing the path of the robot, building an occupancy grid from laser measurements is an easy task. Simultaneous localization and mapping can provide accurate robot pose estimates with respect to the SLAM map. Building occupancy grids by using the pose estimate of the robot from SLAM results in inaccurate grid maps since with each SLAM feature update the SLAM map may move. In chapter 3 a solution to this problem is described where with each scan the robot pose with respect to neighboring SLAM map features is stored instead of the robot pose estimate from SLAM. Then in the occupancy grid building process, all stored robot poses are reconstructed with respect to the current SLAM map using a process similar to scan matching on the features stored with the pose and the features in the SLAM map.

- *Polar scan matching algorithm (PSM).* For building good quality occupancy grids, the previously described approach of storing laser scans together with previous robot poses with respect to a SLAM map is not the only possibility. The previous robot poses can be the map features which are updated by scan matching. In chapter 4, a fast laser scan matching approach called polar scan matching (PSM) has been described. PSM finds the pose of the current scan in the frame of the reference scan by minimizing the sum of square range residuals of the current and reference scan. Range readings of the current scan are associated with those range readings of the reference scan which share the same bearing in the reference frame. In polar scan matching an iteration step consisting of the application of a projection filter followed by a position estimation step is alternated with an iteration step consisting of the application of a projection filter followed by an orientation estimation step. The devised projection filter works in the polar coordinate frame of the laser scanner, just as the rest of PSM and does not need expensive searches to eliminate occluded points. The computational complexity of the projection filter is $O(mn)$ where n is the number of scan points and m is the highest depth of occlusion in the current scan. The depth of occlusion is one if there are no occlusions, 2 if there are occluding objects which are not themselves occluded and 3 if there are occluding objects which are also occluded. The $O(n)$ translation estimation step minimizes the sum of square range residuals by solving the linearized least squares problem. If two scans were taken from the same pose, but one of them is rotated, this rotation manifests itself as a shift of the scan in a polar coordinate system. This is used in the orientation estimation step by shifting the current scan within a window and calculating the average absolute range residual for each shift. The orientation correction estimate corresponding

to the minimum is then improved by quadratic interpolation. If the shift increments are fixed to a value (for example 1°), then the orientation estimation step is of $O(n)$ complexity. However if maximum accuracy is required, and the shift increments are equal to the angular resolution of the scan, then the computational complexity is $O(kn)$, where k is the number of range readings per degree.

The performance of PSM has been evaluated and compared to ICP's in four experiments. A simulation experiment has been followed by an experiment where scans have been taken at known locations in different scenes, and then the scan matching results have been compared with ground truth. Convergence maps, that is maps showing where scan matching converges from different initial positions have also been generated, and the areas from where the scan matching algorithms converged to correct solutions have been measured. In the last experiment SLAM has been performed using scan matching and the resulting maps have been compared. PSM performed much better in all experiments compared to ICP except in the experiment where the area of convergence had to be measured. In this case PSM performed only slightly better.

As future work an important task is to compare PSM with Jens-Steffen Gutmann's implementation of IDC. It would be wise to create a public database of scans and scan matching algorithms so authors could objectively compare the performance of their algorithm with others. Also in the future tracking and tagging of moving objects should be implemented to make PSM more robust in populated environments. The fast nature of PSM would be an important advantage in 3D scan matching. If PSM could be adapted to 3D laser scans, the lack of search for corresponding points would result in large time savings due to the large number of points.

Appendix A

Laser Line Fitting

A.1 Line in Polar Coordinate System

If the equations for transformation of a point from polar to Cartesian frame

$$x = r\cos\phi \tag{A.1}$$

$$y = r\sin\phi \tag{A.2}$$

are substituted into the normal equation of a line:

$$x\cos(\alpha) + y\sin(\alpha) = d \tag{A.3}$$

one gets

$$r\cos(\alpha)\cos(\phi) + r\sin(\alpha)\sin(\phi) = d \tag{A.4}$$

Which can be simplified to

$$r\cos(\alpha - \phi) = d \tag{A.5}$$

A.2 Conversion of Lines from Slope-Intercept Form to Normal Form

Line expressed in normal form:

$$x\cos\alpha + y\sin\alpha = d \tag{A.6}$$

Line expressed in slope-intercept form:

$$y = kx + q \tag{A.7}$$

Let us rewrite (A.7) so, that the coefficients at coordinates x, y constitute a unit vector:

$$\frac{-k}{\sqrt{1+k^2}}x+\frac{1}{\sqrt{1+k^2}}y=\frac{q}{\sqrt{1+k^2}} \tag{A.8}$$

When comparing (A.8) and (A.6), it is clear, that they are equivalent only if:

$$\frac{-k}{\sqrt{1+k^2}} = \cos\alpha \tag{A.9}$$

$$\frac{1}{\sqrt{1+k^2}} = \sin\alpha \tag{A.10}$$

$$\frac{q}{\sqrt{1+k^2}} = d \tag{A.11}$$

Upon using (A.10), α would be always larger then 0 and smaller than $\pi/2$, because of the restricted domain of definition of inverse sine. Therefore (A.9) is used to determine α. However, the inverse of cosine returns values in the $<0,\pi>$ interval, and d has to be larger than 0, therefore the result has to be modified to get:

$$\alpha = \arccos\frac{-k}{\sqrt{1+k^2}}+(sign(q)-1)\frac{\pi}{2} \tag{A.12}$$

$$d = \frac{|q|}{\sqrt{1+k^2}} \tag{A.13}$$

A.3 Covariance Estimate of Line Parameter Errors for Bias Changing with Each Scan

In this section the results of section 2.3.4 are extended. It is shown how to derive a covariance estimate of the angle and distance error if range bias r_b is fixed for the range measurements of a line segment in a scan, but changes as zero mean white Gaussian noise with each scan. For the sake of simplicity, let us introduce the following substitutions from equations (2.33) and (2.35):

$$k_e = Ar_b \tag{A.14}$$

$$\begin{aligned} q_e &= Br_b - Cr_bk_e - k_eD \\ &= Br_b - ACr_b^2 - ADr_b \end{aligned} \tag{A.15}$$

where $A=\frac{1}{d}\frac{n\sum\cos\phi_i-\sum\cot\phi_i\sum\sin\phi_i}{n\sum\cot^2\phi_i-(\sum\cot\phi_i)^2}$, $B=\frac{1}{n}\sum\sin\phi_i$, $C=\frac{1}{n}\sum\cos\phi_i$ and $D=\frac{1}{n}\sum\cot\phi_i$. Prior calculating the covariance matrix, the expectations for k_e and q_e are

calculated:

$$\begin{aligned}
\bar{k}_e &= \mathrm{E}(k_e) = \mathrm{E}(Ar_b) = A\mathrm{E}(r_b) = A\bar{r}_b = 0 && \text{(A.16)}\\
\bar{q}_e &= \mathrm{E}(q_e) = \mathrm{E}(Br_b - ACr_b^2 - ADr_b) \\
&= (B-AD)\mathrm{E}(r_b) - AC\mathrm{E}(r_b^2) = 0 - AC\sigma_{r_b}^2 \approx 0 && \text{(A.17)}
\end{aligned}$$

In (A.17) a simplifying assumption is made of ACr_b^2 and therefore of $AC\sigma_{r_b}^2$ being so small that it can be approximated with 0. Then the elements of the covariance matrix are calculated as following:

$$\begin{aligned}
\sigma_{k_e}^2 &= \mathrm{E}(k_e^2) - (\mathrm{E}(k_e))^2 = \mathrm{E}(A^2 r_b^2) = A^2\sigma_{r_b}^2 && \text{(A.18)}\\
\sigma_{q_e}^2 &= \mathrm{E}(q_e^2) - (\mathrm{E}(q_e))^2 = \mathrm{E}((B-AD)^2 r_b^2) = (B-AD)^2\sigma_{r_b}^2 && \text{(A.19)}\\
\sigma_{k_e q_e} &= \mathrm{E}\left[(k_e - \bar{k}_e)(q_e - \bar{q}_e)\right] \\
&= \mathrm{E}(k_e q_e) = \mathrm{E}(Ar_b(B-AD)r_b) = A(B-AD)\sigma_{r_b}^2 && \text{(A.20)}
\end{aligned}$$

Then the resulting covariance matrix due to bias changing randomly but fixed for the whole line segment will be:

$$C_b = cov(k_e, q_e) = cov(\alpha_e, d_e) = \begin{bmatrix} A^2 & (B-AD)A \\ (B-AD)A & (B-AD)^2 \end{bmatrix} \sigma_{r_b}^2 \qquad \text{(A.21)}$$

A.4 Derivation of (2.54)–(2.55)

The objective is to find out how line parameters change if the true ranges of horizontal line are perturbed by systematic range errors ξ_i. Let us assume, we have a line with parameters (α, d). Then this line can be described as:

$$r_i = \frac{d}{\cos(\alpha - \phi_i)} \qquad \text{(A.22)}$$

If we assume to have a horizontal line with the true parameters $(\alpha_0 = \frac{\pi}{2}, d_0)$ and if we linearize (A.22) around (α_0, d_0) we get:

$$\xi_i = r_i - r_{0i} \approx \frac{\Delta d}{\sin\phi_i} + \frac{d_0 \cos\phi_i}{\sin^2\phi_i} = a_i \Delta d + b_i \Delta\alpha \qquad \text{(A.23)}$$

where $a_i = \frac{1}{\sin\phi_i}$ and $b_i = \frac{d_0\cos\phi_i}{\sin^2\phi_i}$.

We can interpret ξ_i as the range difference we get if we add $(\Delta\alpha, \Delta d)$ to (α_0, d_0). Therefore if we know that a bias of ξ_{mi} is added to each true range, then we can get an estimate of $\Delta\alpha$, Δd if we minimize sum of square deviations between the bias ξ_{mi} and the estimated range

difference ξ_i:

$$E = \sum (\xi_i - \xi_{mi})^2 = \sum (a_i \Delta d + b_i \Delta\alpha - \xi_{mi})^2 \tag{A.24}$$

To find $\Delta\alpha$, Δd which minimize E, we need to differentiate (A.24) with respect to $\Delta\alpha$ and Δd, and find out where are they equal to 0:

$$\frac{\partial E}{\partial \Delta\alpha} = 2\sum (a_i \Delta d + b_i \Delta\alpha - \xi_{mi})\, a_i = 0 \tag{A.25}$$

$$\frac{\partial E}{\partial \Delta d} = 2\sum (a_i \Delta d + b_i \Delta\alpha - \xi_{mi})\, b_i = 0 \tag{A.26}$$

Expanding the previous equations and dividing by 2 results in:

$$\Delta d \sum a_i^2 + \Delta\alpha \sum b_i a_i = \sum \xi_{mi} a_i \tag{A.27}$$

$$\Delta d \sum a_i b_i + \Delta\alpha \sum b_i^2 = \sum \xi_{mi} b_i \tag{A.28}$$

of which the solution is:

$$\Delta\alpha = \frac{\sum \xi_{mi} a_i \sum a_i b_i - \sum \xi_{mi} b_i \sum a_i^2}{(\sum b_i a_i)^2 - \sum b_i^2 \sum a_i^2} \tag{A.29}$$

$$\Delta d = \frac{\sum \xi_{mi} b_i \sum a_i b_i - \sum \xi_{mi} a_i \sum b_i^2}{(\sum b_i a_i)^2 - \sum b_i^2 \sum a_i^2} \tag{A.30}$$

The last two equations are a special case of the approach shown in section 2.3.2 for estimating lines in a polar coordinate system. The difference is that only one iteration is made (small range deviations are assumed) and the true line must be horizontal.

A.5 Closed Form Error Calculation for Systematic Error Caused by Error Changing with Incidence Angle

This section contains the details omitted from section 2.3.5 of the closed form derivation of line parameter error estimates due to error changing with incidence angle.

After substituting (2.52), $a_i = \frac{1}{\sin\phi_i}$ and $b_i = \frac{d_0 \cos\phi_i}{\sin^2\phi_i}$ into (2.54)–(2.55) one gets:

$$\begin{aligned} numerator(\Delta\alpha) &= \sum \Delta r_{ai} a_i \sum a_i b_i - \sum \Delta r_{ai} b_i \sum a_i^2 \\ &= \sum ws_i \cot\phi_i \frac{1}{\sin\phi_i} \sum d_0 \frac{\cos\phi_i}{\sin^3\phi_i} - \sum ws_i \cot\phi_i d_0 \frac{\cos\phi_i}{\sin^2\phi_i} \sum \frac{1}{\sin^2\phi_i} \\ &= wd_0 \left(\sum s_i \frac{\cos\phi_i}{\sin^2\phi_i} \sum \frac{\cos\phi_i}{\sin^3\phi_i} - \sum s_i \frac{\cos^2\phi_i}{\sin^3\phi_i} \sum \frac{1}{\sin^2\phi_i} \right) \end{aligned} \tag{A.31}$$

$$
\begin{aligned}
denom.(\Delta\alpha,\Delta d) &= d_0^2\left[\left(\sum\frac{\cot\phi_i}{\sin^2\phi_i}\right)^2-\sum\frac{\cot^2\phi_i}{\sin^2\phi_i}\sum\frac{1}{\sin^2\phi_i}\right] && \text{(A.32)}\\
numerator(\Delta d) &= \sum\Delta r_{ai}b_i\sum a_ib_i-\sum\Delta r_{ai}a_i\sum b_i^2\\
&= \sum ws_i\cot\phi_i d_0\frac{\cos\phi_1}{\sin^2\phi_i}\sum d_0\frac{\cos\phi_i}{\sin^3\phi_i}-\sum ws_i\cot\phi_i d_0\frac{1}{\sin\phi_i}\sum d_0^2\frac{\cos^2\phi_i}{\sin^4\phi_i}\\
&= wd_0^2\left(\sum s_i\frac{\cos^2\phi_i}{\sin^3\phi_i}\sum\frac{\cos\phi_i}{\sin^3\phi_i}-\sum s_i\frac{\cos\phi_i}{\sin^2\phi_i}\sum\frac{\cos^2\phi_i}{\sin^4\phi_i}\right) && \text{(A.33)}
\end{aligned}
$$

To get a closed form solution, the following approximations are used, whereas the solutions for the integrals are taken from [Jeffrey, 2000]:

$$
\begin{aligned}
\sum\frac{1}{\sin^2\phi_i} &\approx \frac{n}{\Delta\phi}\int_{\phi_1}^{\phi_n}\frac{1}{\sin^2\phi_i}=\frac{n}{\Delta\phi}(-\cot\phi_n+\cot\phi_1) && \text{(A.34)}\\
\sum\frac{\cos\phi_i}{\sin^2\phi_i} &\approx \frac{n}{\Delta\phi}\int_{\phi_1}^{\phi_n}\frac{\cos\phi_i}{\sin^2\phi_i}=\frac{n}{\Delta\phi}\left(-\frac{1}{\sin\phi_n}+\frac{1}{\sin\phi_1}\right) && \text{(A.35)}\\
\sum\frac{\cos\phi_i}{\sin^3\phi_i} &\approx \frac{n}{\Delta\phi}\int_{\phi_1}^{\phi_n}\frac{\cos\phi_i}{\sin^3\phi_i}=\frac{n}{\Delta\phi}\left(-\frac{1}{2\sin^2\phi_n}+\frac{1}{2\sin^2\phi_1}\right) && \text{(A.36)}\\
\sum\frac{\cos^2\phi_i}{\sin^4\phi_i} &\approx \frac{n}{\Delta\phi}\int_{\phi_1}^{\phi_n}\frac{\cos^2\phi_i}{\sin^4\phi_i}=\frac{n}{\Delta\phi}\left(-\frac{1}{3}\cot^3\phi_n+\frac{1}{3}\cot^3\phi_1\right) && \text{(A.37)}\\
\sum\frac{\cos^2\phi_i}{\sin^3\phi_i} &\approx \frac{n}{\Delta\phi}\int_{\phi_1}^{\phi_n}\frac{\cos^2\phi_i}{\sin^3\phi_i}=\\
&= \frac{n}{\Delta\phi}\left(\frac{\cos\phi_1}{2\sin^2\phi_1}-\frac{\cos\phi_n}{2\sin^2\phi_n}+\frac{1}{2}\ln\left|\tan\frac{\phi_1}{2}\right|-\frac{1}{2}\ln\left|\tan\frac{\phi_n}{2}\right|\right) && \text{(A.38)}
\end{aligned}
$$

However as one can see, s_i is left out from (A.35), (A.38). Because $s_i = sign(\cot\phi_i)$, $s_i = 1$ for $\phi \in (0,\pi/2)$ and $s_i = -1$ for $\phi \in (\pi/2,\pi)$. This means that in case of $\phi_1,\phi_n \leq \pi/2$ or $\phi_1,\phi_n \geq \pi/2$, $s_i = s$ can be moved in front of the sums:

$$
\begin{aligned}
\sum s_i\frac{\cos^2\phi_i}{\sin^3\phi_i} &\approx \frac{sn}{\Delta\phi}\left(\frac{\cos\phi_i}{2\sin^2\phi_i}+\frac{\cos\phi_n}{2\sin^2\phi_n}+\frac{1}{2}\ln\left|\tan\frac{\phi_1}{2}\tan\frac{\phi_n}{2}\right|\right) && \text{(A.39)}\\
\sum s_i\frac{\cos\phi_i}{\sin^2\phi_i} &\approx \frac{sn}{\Delta\phi}\left(\frac{1}{\sin\phi_1}-\frac{1}{\sin\phi_n}\right) && \text{(A.40)}
\end{aligned}
$$

In case where $\phi_1 \leq \pi/2 \leq \phi_n$, the integrals in (A.35), (A.38) have to be broken down into two parts such as an integral from ϕ_1 to $\pi/2$ with $s_i = s = 1$ and an integral from $\pi/2$ to ϕ_n with $s_i = s = -1$:

$$
\begin{aligned}
\sum s_i\frac{\cos^2\phi_i}{\sin^3\phi_i} &\approx \frac{n}{\Delta\phi}\left[\int_{\phi_1}^{\frac{\pi}{2}}\frac{\cos^2\phi}{\sin^3\phi}d\phi-\int_{\frac{\pi}{2}}^{\phi_n}\frac{\cos^2\phi}{\sin^3\phi}d\phi\right]\\
&= \frac{n}{\Delta\phi}\left(\frac{\cos\phi_1}{2\sin^2\phi_1}+\frac{\cos\phi_n}{2\sin^2\phi_n}+\frac{1}{2}\ln\left|\tan\frac{\phi_1}{2}\tan\frac{\phi_n}{2}\right|\right) && \text{(A.41)}
\end{aligned}
$$

$$\begin{aligned}
\sum s_i \frac{\cos\phi_i}{\sin^2\phi_i} &\approx \frac{n}{\Delta\phi}\int_{\phi_1}^{\phi_n}\frac{\cos\phi_i}{\sin^2\phi_i}d\phi \\
&= \frac{n}{\Delta\phi}\left[\int_{\phi_1}^{\frac{\pi}{2}}\frac{\cos\phi}{\sin^2\phi}d\phi - \int_{\frac{\pi}{2}}^{\phi_n}\frac{\cos\phi}{\sin^2\phi}d\phi\right] \\
&= \frac{n}{\Delta\phi}\left[-\frac{1}{\sin\frac{\pi}{2}}+\frac{1}{\sin\phi_1}+\frac{1}{\sin\phi_n}-\frac{1}{\sin\frac{\pi}{2}}\right] \\
&= \frac{n}{\Delta\phi}\left[-2+\frac{1}{\sin\phi_1}+\frac{1}{\sin\phi_n}\right] \qquad \text{(A.42)}
\end{aligned}$$

(A.39) with (A.41) and (A.40) with (A.42) can be merged together the following way:

$$\begin{aligned}
\sum s_i \frac{\cos^2\phi_i}{\sin^3\phi_i} &\approx \frac{ns}{\Delta\phi}F = \\
&= \frac{ns}{\Delta\phi}\left(\frac{\cos\phi_1}{2\sin^2\phi_1}+t\frac{\cos\phi_n}{2\sin^2\phi_n}+\frac{1}{2}\ln\left|\tan\frac{\phi_1}{2}\right|+\frac{t}{2}\ln\left|\tan\frac{\phi_n}{2}\right|\right) \qquad \text{(A.43)} \\
\sum s_i \frac{\cos\phi_i}{\sin^2\phi_i} &\approx \frac{ns}{\Delta\phi}\left(-(t+1)+\frac{1}{\sin\phi_1}+\frac{t}{\sin\phi_n}\right) = \frac{nsE}{\Delta\phi} \qquad \text{(A.44)}
\end{aligned}$$

where

- $s=1,\ t=1$ for $\phi_1 \le \pi/2 \le \phi_n$.
- $s=1,\ t=-1$ for $\phi_1, \phi_n \le \pi/2$.
- $s=-1,\ t=-1$ for $\phi_1, \phi_n \ge \pi/2$.

Thus the closed form solution for the line parameter error generated by range error changing with the incidence angle:

$$\begin{aligned}
\Delta\alpha &= \frac{ws}{d}\frac{EA-FC}{A^2-\frac{1}{3}(\cot^3\phi_1-\cot^3\phi_n)C} \qquad \text{(A.45)} \\
\Delta d &= ws\frac{FA-ED}{A^2-\frac{1}{3}(\cot^3\phi_1-\cot^3\phi_n)C} \qquad \text{(A.46)}
\end{aligned}$$

where

$$\begin{aligned}
A &= \frac{1}{2}\left(\frac{1}{\sin^2\phi_1}-\frac{1}{\sin^2\phi_n}\right) \qquad \text{(A.47)} \\
C &= \cot\phi_1-\cot\phi_n \qquad \text{(A.48)} \\
D &= \frac{1}{3}\left(\cot^3\phi_1-\cot^3\phi_n\right) \qquad \text{(A.49)} \\
E &= -(t+1)+\frac{1}{\sin\phi_1}+\frac{t}{\sin\phi_n} \qquad \text{(A.50)} \\
F &= \frac{\cos\phi_1}{2\sin^2\phi_1}+t\frac{\cos\phi_n}{2\sin^2\phi_n}+\frac{1}{2}\ln|\tan\frac{\phi_1}{2}|+\frac{t}{2}\ln|\tan\frac{\phi_n}{2}| \qquad \text{(A.51)}
\end{aligned}$$

A.6 Closed Form Error Calculation for Systematic Error Caused by Error Growing with Distance

This section describes the omitted derivation details of section 2.3.6.

To derive the closed form error estimates (2.27) is substituted into (2.63) and the result is substituted into (2.54)–(2.55):

$$\begin{aligned}
numerator(\Delta\alpha) &= \sum \Delta r_{ri} a_i \sum a_i b_i - \sum \Delta r_{ri} b_i \sum a_i^2 \\
&= \sum r_{ei} \frac{1}{\sin\phi_i} \sum d_0 \frac{\cos\phi_i}{\sin^3\phi_i} - \sum r_{ei} d_0 \frac{\cos\phi_i}{\sin^2\phi_i} \sum \frac{1}{\sin^2\phi_i} \\
&= \sum d_0 k \left(\frac{1}{\sin(\phi_i)} - \frac{1}{\sin\phi_{min}} \right) \frac{1}{\sin\phi_i} \sum d_0 \frac{\cos\phi_i}{\sin^3\phi_i} - \\
&\quad - \sum d_0 k \left(\frac{1}{\sin(\phi_i)} - \frac{1}{\sin\phi_{min}} \right) d_0 \frac{\cos\phi_i}{\sin^2\phi_i} \sum \frac{1}{\sin^2\phi_i} \\
&= \frac{d_0^2 k}{\sin\phi_{min}} \left(\sum \frac{1}{\sin^2\phi_i} \sum \frac{\cos\phi_i}{\sin^2\phi_i} - \sum \frac{1}{\sin\phi_i} \sum \frac{\cos\phi_i}{\sin^3\phi_i} \right) \qquad \text{(A.52)} \\
numerator(\Delta d) &= \sum \Delta r_{ri} b_i \sum a_i b_i - \sum \Delta r_{ri} a_i \sum b_i^2 \\
&= \sum r_{ei} \frac{d_0 \cos\phi_i}{\sin^2\phi_i} \sum d_0 \frac{\cos\phi_i}{\sin^3\phi_i} - \sum r_{ei} \frac{1}{\sin\phi_i} \sum \frac{d_0^2 \cos^2\phi_i}{\sin^4\phi_i} \\
&= d_0^3 k \left(\sum \left(\frac{1}{\sin(\phi_i)} - \frac{1}{\sin\phi_{min}} \right) \frac{\cos\phi_i}{\sin^2\phi_i} \sum \frac{\cos\phi_i}{\sin^3\phi_i} \right) - \\
&\quad - d_0^3 k \left(\sum \left(\frac{1}{\sin(\phi_i)} - \frac{1}{\sin\phi_{min}} \right) \frac{1}{\sin\phi_i} \sum \frac{\cos^2\phi_i}{\sin^4\phi_i} \right) \\
&= d_0^3 k \left(\sum \frac{\cos\phi_i}{\sin^3\phi_i} - \frac{1}{\sin\phi_{min}} \sum \frac{\cos\phi_i}{\sin^2\phi_i} \right) \sum \frac{\cos\phi_i}{\sin^3\phi_i} \\
&\quad - d_0^3 k \left(\sum \frac{1}{\sin^2\phi_i} - \frac{1}{\sin\phi_{min}} \sum \frac{1}{\sin\phi_i} \right) \sum \frac{\cos^2\phi_i}{\sin^4\phi_i} \qquad \text{(A.53)} \\
denominator(\Delta\alpha, \Delta d) &= d_0^2 \left[\left(\sum \frac{\cot\phi_i}{\sin^2\phi_i} \right)^2 - \sum \frac{\cot^2\phi_i}{\sin^2\phi_i} \sum \frac{1}{\sin^2\phi_i} \right] \qquad \text{(A.54)}
\end{aligned}$$

Then the following approximations are used, whereas the solutions for the integrals are taken from [Jeffrey, 2000]:

$$\sum \frac{1}{\sin\phi_i} \approx \frac{n}{\Delta\phi} \int_{\phi_1}^{\phi_n} \frac{1}{\sin\phi_i} = \frac{n}{\Delta\phi} \ln \left| \frac{\tan\frac{\phi_n}{2}}{\tan\frac{\phi_1}{2}} \right| \qquad \text{(A.55)}$$

$$\sum \frac{1}{\sin^2\phi_i} \approx \frac{n}{\Delta\phi} \int_{\phi_1}^{\phi_n} \frac{1}{\sin^2\phi_i} = \frac{n}{\Delta\phi} (-\cot\phi_n + \cot\phi_1) \qquad \text{(A.56)}$$

$$\sum \frac{\cos\phi_i}{\sin^2\phi_i} \approx \frac{n}{\Delta\phi} \int_{\phi_1}^{\phi_n} \frac{\cos\phi_i}{\sin^2\phi_i} = \frac{n}{\Delta\phi} \left(-\frac{1}{\sin\phi_n} + \frac{1}{\sin\phi_1} \right) \qquad \text{(A.57)}$$

$$\sum \frac{\cos\phi_i}{\sin^3\phi_i} \approx \frac{n}{\Delta\phi}\int_{\phi_1}^{\phi_n}\frac{\cos\phi_i}{\sin^3\phi_i} = \frac{n}{\Delta\phi}\left(-\frac{1}{2\sin^2\phi_n}+\frac{1}{2\sin^2\phi_1}\right) \tag{A.58}$$

$$\sum \frac{\cos^2\phi_i}{\sin^4\phi_i} \approx \frac{n}{\Delta\phi}\int_{\phi_1}^{\phi_n}\frac{\cos^2\phi_i}{\sin^4\phi_i} = \frac{n}{\Delta\phi}\left(-\frac{1}{3}\cot^3\phi_n+\frac{1}{3}\cot^3\phi_1\right) \tag{A.59}$$

$$\tag{A.60}$$

The resulting closed form solutions are:

$$\Delta\alpha = \frac{k}{\sin\phi_{min}}\frac{BC - A\ln\left|\frac{\tan\frac{\phi_n}{2}}{\tan\frac{\phi_1}{2}}\right|}{A^2-DC} \tag{A.61}$$

$$\Delta d = d_0 k\frac{A\left(A-\frac{B}{\sin\phi_{min}}\right)-\left[C-\frac{1}{\sin\phi_{min}}\ln\left|\frac{\tan\frac{\phi_n}{2}}{\tan\frac{\phi_1}{2}}\right|\right]D}{A^2-DC} \tag{A.62}$$

where ϕ_{min} is the bearing corresponding to r_{min} and

$$A = \frac{1}{2\sin^2\phi_1}-\frac{1}{2\sin^2\phi_n} \tag{A.63}$$

$$B = \frac{1}{\sin\phi_1}-\frac{1}{\sin\phi_n} \tag{A.64}$$

$$C = \cot\phi_1-\cot\phi_n \tag{A.65}$$

$$D = \frac{1}{3}\left(\cot^3\phi_1-\cot^3\phi_n\right) \tag{A.66}$$

A.7 Covariance and Correlation Coefficient Matrices of Range Readings

To support our assumption, that estimating the range measurement covariance matrix as a diagonal matrix is sufficient, we show the first 7x7 sub-matrices of the covariance and correlation coefficient matrices of range readings for line 1 and 13 (see fig. 2.23). All matrices in graphical representation are shown in fig. 2.21. For the calculation of these matrices, about 3000 samples were used.

$$cov_1 = \begin{bmatrix} 3.502 & 0.385 & -0.016 & 0.242 & 0.173 & 0.001 & 0.364 \\ 0.385 & 4.704 & -0.051 & 0.266 & 0.673 & 0.200 & 0.367 \\ -0.016 & -0.051 & 5.335 & 1.080 & 0.422 & 0.471 & 0.235 \\ 0.242 & 0.266 & 1.080 & 6.186 & 1.232 & 0.253 & 0.558 \\ 0.173 & 0.673 & 0.422 & 1.232 & 7.357 & 1.308 & 0.769 \\ 0.001 & 0.200 & 0.471 & 0.253 & 1.308 & 8.809 & 0.528 \\ 0.364 & 0.367 & 0.235 & 0.558 & 0.769 & 0.528 & 7.954 \end{bmatrix} \tag{A.67}$$

$$corr_1 = \begin{bmatrix} 1.000 & 0.095 & -0.004 & 0.052 & 0.034 & 0.000 & 0.069 \\ 0.095 & 1.000 & -0.010 & 0.049 & 0.114 & 0.031 & 0.060 \\ -0.004 & -0.010 & 1.000 & 0.188 & 0.067 & 0.069 & 0.036 \\ 0.052 & 0.049 & 0.188 & 1.000 & 0.183 & 0.034 & 0.079 \\ 0.034 & 0.114 & 0.067 & 0.183 & 1.000 & 0.163 & 0.101 \\ 0.000 & 0.031 & 0.069 & 0.034 & 0.163 & 1.000 & 0.063 \\ 0.069 & 0.060 & 0.036 & 0.079 & 0.101 & 0.063 & 1.000 \end{bmatrix} \tag{A.68}$$

$$cov_{13} = \begin{bmatrix} 6.008 & 0.062 & -0.003 & -0.576 & 0.145 & -0.165 & 0.344 \\ 0.062 & 2.430 & 0.242 & 0.293 & 0.119 & 0.211 & 0.066 \\ -0.003 & 0.242 & 4.128 & 0.272 & -0.042 & 0.124 & 0.034 \\ -0.576 & 0.293 & 0.272 & 6.329 & 0.216 & 0.131 & -0.086 \\ 0.145 & 0.119 & -0.042 & 0.216 & 2.579 & 0.077 & 0.140 \\ -0.165 & 0.211 & 0.124 & 0.131 & 0.077 & 5.362 & 0.343 \\ 0.344 & 0.066 & 0.034 & -0.086 & 0.140 & 0.343 & 5.562 \end{bmatrix} \tag{A.69}$$

$$corr_{13} = \begin{bmatrix} 1.000 & 0.016 & -0.001 & -0.093 & 0.037 & -0.029 & 0.060 \\ 0.016 & 1.000 & 0.076 & 0.075 & 0.048 & 0.058 & 0.018 \\ -0.001 & 0.076 & 1.000 & 0.053 & -0.013 & 0.026 & 0.007 \\ -0.093 & 0.075 & 0.053 & 1.000 & 0.054 & 0.022 & -0.014 \\ 0.037 & 0.048 & -0.013 & 0.054 & 1.000 & 0.021 & 0.037 \\ -0.029 & 0.058 & 0.026 & 0.022 & 0.021 & 1.000 & 0.063 \\ 0.060 & 0.018 & 0.007 & -0.014 & 0.037 & 0.063 & 1.000 \end{bmatrix} \tag{A.70}$$

As it can be seen from (A.68), (A.70), the correlation coefficients are most of the time quite small.

Appendix B

More Equations for SLAM

In the following table some of the equations necessary for EKF SLAM with line segment, right angle corner and point features are described. In the table the function "norm" normalizes angles by converting them into the interval $(-\pi, \pi >$. x, y, θ express the robot's pose from the state vector $\mathbf{x}$.

	Line segments	Point features	Corner Features
Representation in robot frame	α_r orientation d_r distance	(x_r, y_r)	ϕ_r bearing, r_r range of center γ_r orientation
Representation in SLAM map	α_m orientation d_m distance	(x_m, y_m)	x_m, y_m of center γ_m orientation
Transf. from robot to map	$\alpha_m = norm(\theta + \alpha_r)$ $d_m = d_r + x\cos(\alpha) + y\sin(\alpha)$ $s = 1$ $d_m < 0 \Rightarrow d_m = -d_m;\ s = -1$ $\alpha_m = norm(\alpha_m + \pi);$	$x_m = x_r\cos\theta - y_r\sin\theta + x$ $x_m = x_r\sin\theta + y_r\cos\theta + y$	$x_r = r_r\cos\phi_r$ $y_r = r_r\sin\phi_r$ $x_m = x_r\cos\theta - y_r\sin\theta + x$ $x_m = x_r\sin\theta + y_r\cos\theta + y$ $\gamma_m = norm(\gamma_r + \theta)$
Transf. from map to robot	$\alpha_r = norm(\alpha_m - \theta)$ $d_r = d_m - x\cos(\alpha) - y\sin(\alpha)$ $s = 1$ $d_r < 0 \Rightarrow d_r = -d_r;\ s = -1;$ $\alpha_r = norm(\alpha_r + \pi)$	$x_r = x_m\cos\theta + y_r\sin\theta - x\cos\theta - y\sin\theta$ $y_r = -y_m\sin\theta + y_r\cos\theta + x\sin\theta - y\cos\theta$	$x_r = x_m\cos\theta + y_r\sin\theta - x\cos\theta - y\sin\theta$ $y_r = -y_m\sin\theta + y_r\cos\theta + x\sin\theta - y\cos\theta$ $\phi_r = \arctan 2(y_r, x_r)$ $r - r = \sqrt{x - r^2 + y_r^2}$ $\gamma_r = norm(\gamma_m - \theta)$
$\frac{\partial \mathbf{y}_i}{\partial x_v}$	$\begin{bmatrix} 0 & 0 & 1 \\ s\cos\alpha_m & s\sin\alpha_m & -sx\sin\alpha_m + sy\cos\alpha_m \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & -x_r\sin\theta - x_r\cos\theta \\ 0 & 1 & x_r\cos\theta - y_r\sin\theta \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & -r_r\sin(\theta+\phi_r) \\ 0 & 1 & r_r\cos(\theta+\phi_r) \\ 0 & 0 & 1 \end{bmatrix}$
$\frac{\partial \mathbf{y}_i}{\partial h}$	$\begin{bmatrix} 1 & 0 \\ -sx\sin\alpha_m + sy\cos\alpha_m & s \end{bmatrix}$	$\begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$	$\begin{bmatrix} -r_r\sin(\theta+\phi_r) & \cos(\theta+\phi_r) & 0 \\ r_r\cos(\theta+\phi_r) & \sin(\theta+\phi_r) & 0 \\ 0 & 0 & 1 \end{bmatrix}$
$\frac{\partial h}{\partial \mathbf{y}_i}$	$\begin{bmatrix} 0 & 0 & -1 \\ -s\cos\alpha_m & -s\sin\alpha_m & 0 \end{bmatrix}$	$\begin{bmatrix} -\cos\theta & -\sin\theta & (x-x_m)\sin\theta - (y-y_m)\cos\theta \\ \sin\theta & -\cos\theta & (x-x_m)\cos\theta + (y-y_m)\sin\theta \end{bmatrix}$	$t = (x-x_m)^2 + (y-y_m)^2$ $\begin{bmatrix} -(y-y_m)/t & (x-x_m)/t & -1 \\ (x-x_m)/\sqrt{t} & (y-y_m)/\sqrt{t} & 0 \\ 0 & 0 & -1 \end{bmatrix}$
$\frac{\partial h}{\partial \mathbf{x}_v}$	$\begin{bmatrix} 1 & 0 \\ sx\sin\alpha_m - sy\cos\alpha_m & s \end{bmatrix}$	$\begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}$	$\begin{bmatrix} (y-y_m)/t & -(x-x_m)/t & 0 \\ (x-x_m)/\sqrt{t} & (y-y_m)/\sqrt{t} & \\ 0 & 0 & 1 \end{bmatrix}$

Bibliography

[Althaus and Christensen, 2003] P. Althaus and H. I. Christensen. Automatic map acquisition for navigation in domestic environments. In *Proc. of the 2003 IEEE Int. Conf. on Robotics and Aut.*, pages 1551–1556. IEEE, 2003.

[Arras and Siegwart, 1997] K. O. Arras and R. Siegwart. Feature extraction and scene interpretation for map-based navigation and map building. In *Proceedings of SPIE, Mobile Robotics XII*, pages 42–53, 1997.

[Arras, 2003] K. O. Arras. *Feature-Based Robot Navigation in Known and Unknown Environments*. PhD thesis, École Polytechnique Fédérale de Lausanne, 2003.

[Ballard and C. M, 1982] D. H. Ballard and Brown C. M. *Computer Vision*. Prentice Hall, New Jersey, 1982.

[Bar-Shalom and Li, 1993] Y. Bar-Shalom and X. Li. *Estimation and Tracking: Principles, Techniques, and Software*. Artech House, Boston, 1993.

[Bengtsson and Baerveldt, 2001] O. Bengtsson and A.-J. Baerveldt. Localization by matching of range scans - certain or uncertain? In *Eurobot'01 - Fourth European Workshop on Advanced Mobile Robots*, pages 49–56, Lund, Sweden, Sep. 2001.

[Besl and McKay, 1992] P. J. Besl and N. D. McKay. A method for registration of 3D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992.

[Biber and Straßer, 2003] P. Biber and W. Straßer. The normal distributions transform: A new approach to laser scan matching. In *IROS'03*, volume 3, pages 2743–2748. IEEE, 2003.

[Bosse *et al.*, 2004] Michael Bosse, Paul Newman, John Leonard, and Seth Teller. Simultaneous localization and map building in large-scale cyclic environments using the Atlas framework. *The International Journal of Robotics Research*, 23(12):1113–1139, 2004.

[Bourgault *et al.*, 2002] F. Bourgault, A. A. Makarenko, S. B. Williams, B. Grocholsky, and H.F. Durrant-Whyte. Information based adaptive robotic exploration. In *Proceedings of Int. Rob. and Syst.*, volume 1, pages 540–545. IEEE, 2002.

[Castellanos and Tardós, 1999] J. A. Castellanos and J. D. Tardós. *Mobile Robot Localization and Map Building, A Multisensor Fusion Approach*. Kluwer Academic Publisher, Norwell, Massachusetts, 1999.

[Castellanos *et al.*, 2004] J.A. Castellanos, J. Neira, and J.D. Tardos. Limits to the consistency of EKF-based SLAM. In *5th IFAC Symposium on Intelligent Autonomous Vehicles*, Lisbon, July 2004.

[Chong and Kleeman, 1999] K. S. Chong and L. Kleeman. Feature-based mapping in real, large scale environments using an ultrasonic array. *IJRR*, 18, No. 1:3–19, Jan 1999.

[Cox, 1991] I. J. Cox. Blanche–an experiment in guidance and navigation of an autonomous robot vehicle. *IEEE Transactions on Robotics and Automation*, 7(2):193–203, april 1991.

[Davison, 1998] A. Davison. *Mobile Robot Navigation Using Active Vision*. PhD thesis, University of Oxford, 1998.

[Deriche *et al.*, 1992] R. Deriche, R. Vaillant, and O. Faugeras. *From Noisy Edges Points to 3D Reconstruction of a Scene : A Robust Approach and Its Uncertainty Analysis*, volume 2, pages 71–79. World Scientific, 1992. Series in Machine Perception and Artificial Intelligence.

[Diosi and Kleeman, 2003a] A. Diosi and L. Kleeman. Uncertainty of line segments extracted from static SICK PLS laser scans. Technical Report MECSE-26-2003, Intelligent Robotics Research Centre, Monash University, 2003. Available: http://www.ds.eng.monash.edu.au/techrep/reports/.

[Diosi and Kleeman, 2003b] A. Diosi and L. Kleeman. Uncertainty of line segments extracted from static SICK PLS laser scans. In *Proceedings of the 2003 Australasian Conference on Robotics and Automation*, 2003.

[Diosi and Kleeman, 2004] A. Diosi and L. Kleeman. Advanced sonar and laser range finder fusion for simultaneous localization and mapping. In *Proc. of IROS'04*, volume 2, pages 1854–1859. IEEE, 2004.

[Diosi and Kleeman, 2005a] A. Diosi and L. Kleeman. Laser scan matching in polar coordinates with application to SLAM. Accepted for publication in the Proc. of IROS'05, 2005.

[Diosi and Kleeman, 2005b] A. Diosi and L. Kleeman. Scan matching in polar coordinates with application to SLAM. Technical Report MECSE-29-2005, Department of Electrical and Computer Systems Eng., Monash University, 2005. Available: http://www.ds.eng.monash.edu.au/techrep/reports/.

[Diosi *et al.*, 2005] A. Diosi, G. Taylor, and L. Kleeman. Interactive SLAM using laser and advanced sonar. In *Proc. of ICRA'05*. IEEE, 2005.

[Dudek and Jenkin, 2000] G. Dudek and M. Jenkin. *Computational Principles of Mobile Robotics*. Cambridge University Press, Cambridge, 2000.

[Dudek *et al.*, 1996] G. Dudek, P. Freedman, and I. M. Rekleitis. Just–in–time sensing: efficiently combining sonar and laser range data for exploring unknown worlds. In *Proc. of the 1996 IEEE Int. Conf. on Robotics & Automation*, pages 667–671, Minneapolis, Minnesota, April 1996. IEEE.

[Enderle *et al.*, 1999] S. Enderle, G. Kraetzschmar, S. Sablatnög, and G. Palm. Sonar interpretation learned from laser data. In *Proceedings of the Third European Workshop on Advanced Mobile Robots (EUROBOT'99)*, pages 121–126, Zurich, Switzerland, 1999.

[Erwin Sick GmbH, 1995] *Operating Instructions, PLS and PLS User Software Laser Scanner*. Waldkirch, 1995.

[Feder *et al.*, 1998] H.J.S. Feder, J.J. Leonard, and C.M. Smith. Adaptive concurrent mapping and localization using sonar. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 2, pages 892–898, Victoria, Oct. 1998. IEEE.

[Feder *et al.*, 1999] H. J. S. Feder, J. J. Leonard, and C. M. Smith. Adaptive mobile robot navigation and mapping. *IJRR*, pages 650–668, July 1999.

[Fischler and Bolles, 1981] M. Fischler and R. Bolles. Random sampling consensus: a paradigm for model fitting with application to image analysis and automated cartography. *Commun. of the ACM*, 24(6):381–395, 1981.

[Forsyth and Ponce, 2003] D. A. Forsyth and J. Ponce. *Computer Vision: A Modern Approach*. Prentice Hall, 2003.

[Gander and Hřebíček, 1993] W. Gander and J. Hřebíček. *Solving Problems in Scientific Computing Using Maple and MATLAB*. Springer–Verlag, Berlin, Heidelberg, 1993.

[Garcia *et al.*, 2002] R. Garcia, J. Puig, P. Ridao, and X. Cufi. Augmented state Kalman filtering for AUV navigation. In *Proc. of ICRA'02*, volume 4, pages 4010–4015. IEEE, 2002.

[Guivant and Nebot, 2001] Jose E. Guivant and Eduardo M. Nebot. Optimization of the simultaneous localization and map-building algorithm for real-time implementation. *IEEE Transactions on Robotics and Automation*, 17, no. 3:242–257, June 2001.

[Gutmann and Konolige, 1999] J.-S. Gutmann and K. Konolige. Incremental mapping of large cyclic environments. In *International Symposium on Computational Intelligence in Robotics and Automation (CIRA'99)*, pages 318–325, Monterey, November 1999.

[Gutmann, 2000] J.-S. Gutmann. *Robuste Navigation autonomer mobiler Systeme*. PhD thesis, Albert-Ludwigs-Universität Freiburg, 2000.

[Hähnel *et al.*, 2003a] D. Hähnel, W. Burgard, D. Fox, and S. Thrun. An efficient fastSLAM algorithm for generating maps of large-scale cyclic environments from raw laser range measurements. In *IROS'03*, volume 1, pages 206–211. IEEE, 2003.

[Hähnel *et al.*, 2003b] D. Hähnel, R. Triebel, W. Burgard, and S. Thrun. Map building with mobile robots in dynamic environments. In *ICRA*, volume 2, pages 1557–1563. IEEE, 2003.

[Horn and Schmidt, 1995] J. Horn and G. Schmidt. Continuous localization of a mobile robot based on 3D-laser-range-data, predicted sensor images, and dead-reckogning. *Robotics and Autonomous Systems*, 14:99–118, 1995.

[Jarvis, 1985] R. A. Jarvis. Collision-free trajectory planning using distance transforms. *Mechanical Engineering Transactions*, 10:187–191, 1985.

[Jeffrey, 2000] A. Jeffrey. *Handbook of Mathematical Formulas and Integrals*. Academic Press, London, second edition, 2000.

[Jensfelt, 2001] P. Jensfelt. *Approaches to Mobile Robot Localization in Indoor Environments*. PhD thesis, KTH, 2001.

[Jörg, 1995] K.-L. Jörg. World modeling for an autonomous mobile robot using heterogenous sensor information. *Robotics and Autonomous Systems*, 14:159–170, 1995.

[Kam *et al.*, 1997] M. Kam, X. Zhu, and Paul Kalata. Sensor fusion for mobile robot navigation. *Proceedings of the IEEE*, 85:108–119, January 1997.

[Kay, 1993] Steven M. Kay. *Fundamentals of Statistical Signal Processing*, volume 2. Estimation Theory. Prentice Hall, New Jersey, 1993.

[Kim and Cho, 2001] K.-H. Kim and H. S. Cho. Range and contour fused environment recognition for mobile robot. In *Proceedings on International Conference on Multisensor Fusion and Integration for Intelligent Systems*, pages 183–188, Dusseldorf, Germany, Aug. 2001. IEEE.

[Kleeman, 2001] L. Kleeman. Advanced sonar sensing. In *Proceedings 10th International Symposium Robotics Research*, pages 286–296, Lorne Victoria Australia, November 2001.

[Kleeman, 2002] L. Kleeman. On-the-fly classifying sonar with accurate range and bearing estimation. In *IEEE/RSJ Int. Conf. on Intelligent Robots & Systems*, pages 178–183. IEEE, 2002.

[Kleeman, 2003] L. Kleeman. Advanced sonar and odometry error modeling for simultaneous localization and map building. In *Proc. of the 2003 IEEE/RSJ Intl. Conf. on Intelligent Robots & Systems*, pages 699–704, Las Vegas, Nevada, October 2003. IEEE.

[Kleeman, 2004] Lindsay Kleeman. private communication, 2004.

[Krell Institute, read in 2005] Numerical integration and the mean value theorem. web page, read in 2005. http://www.krellinst.org/UCES/archive/classes/CNA/dir2.2/uces2.2.html, accessed 26.06.2005.

[Krotkov, 1990] E. Krotkov. Laser rangefinder calibration for a walking robot. Technical Report CMU-RI-TR-90-30, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, December 1990.

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

[Leonard and Durrant-Whyte, 1991b] J. J. Leonard and H. F. Durrant-Whyte. Simultaneous map building and localization for an autonomous mobile robot. In *IEEE/RSJ International Workshop on Intelligent Robots and Systems IROS'91*, pages 1442–1447, Osaka, Japan, November 1991. IEEE.

[Leonard *et al.*, 1992] J. Leonard, H. Durrant-Whyte, and I. Cox. Dynamic map building for an autonomous mobile robot. *International Journal of Robotics Research*, 11(4):286–298, August 1992.

[Lingemann *et al.*, 2004] K. Lingemann, H. Surmann, A. Nüchter, and J. Hertzberg. Indor and outdoor localization for fast mobile robots. In *IROS'04*, volume 3, pages 2185–2190. IEEE, 2004.

[Lu and Milios, 1997] F. Lu and E. Milios. Robot pose estimation in unknown environments by matching 2D range scans. *J. of Intelligent and Robotic Systems*, 20:249–275, 1997.

[Lu, 1995] Feng Lu. *Shape Registration Using Optimization for Mobile Robot Navigation*. PhD thesis, University of Toronto, 1995.

[Marco *et al.*, 2001] M.DI Marco, A.Garulli, S.Lacroix, and A.Vicino. Set membership localization and mapping for autonomous navigation. *International Journal of Robust and NonLinear Control*, 11-7:709–734, June 2001.

[Milford *et al.*, 2004] M. J. Milford, G. Wyeth, and D. Prasser. RatSLAM: A hippocampal model for simultaneous localization and mapping. In *International Conference on Robotics and Automatio*, volume 1, pages 730–735. IEEE, 2004.

[Montemerlo, 2003] M. Montemerlo. *FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping Problem With Unknown Data Association*. PhD thesis, School of Computer Science, Carnegie Mellon University, 2003.

[Murphy, 2000] R. Murphy. *Introduction to AI Robotics*. The MIT Press, Cambridge, Massachusetts, 2000.

[Nieto *et al.*, 2002] J. Nieto, J. Guivant, and E. Nebot. FastSLAM: Real time implementation in outdoor environments. In *Proc. of the Australasian Conf. on Robotics & Automation*, Auckland, 2002. ARAA.

[Nygårds and Wernersson, 1998] Jonas Nygårds and Åke Wernersson. On covariances for fusing laser rangers and vision with sensors onboard a moving robot. *IEEE International Conference on Intelligent Robots and Systems*, 2:1053 – 1059, 1998.

[Paul, 1981] R. P. Paul. *Robot manipulators : mathematics, programming, and control : the computer control of robot manipulators*. MIT Press, Cambridge, Mass., 1981.

[Pfister *et al.*, 2003] S. T. Pfister, S. I. Roumeliotis, and J. W. Burdick. Weighted line fitting algorithms for mobile robot map building and efficient data representation. In *Proc. 2003 IEEE Int. Conf. on Robotics and Automation*, volume 1, pages 1304–1311, Taipei, Taiwan, May 2003. IEEE.

[Reina and Gonzales, 1997] Antonio Reina and Javier Gonzales. Characterization of a radial laser scanner for mobile robot navigation. In *IROS'97*, pages 579–585. IEEE, 1997.

[Rybski *et al.*, 2003] P. E. Rybski, S. I. Roumeliotis, M. Gini, and N. Papanikolopoulos. Appearance-based minimalistic metric SLAM. In *Proc. of the 2003 IEEE/RSJ Intl. Conf. on Intelligent Robots & Systems*, pages 194–199, Las Vegas, Nevada, 2003. IEEE.

[SICK AG, 2002] SICK AG, Germany. *LMS 200 / LMS 211 / LMS 220 / LMS 221 / MS 291 Laser Measurement Systems*, 2002.

[Taylor and Probert, 1996] R. M. Taylor and P. J. Probert. Range finding and feature extraction by segmentation of images for mobile robot navigation. In *Proceedings of the 1996*

IEEE International Conference on Robotics and Automation, pages 95–100, Minneapolis, Minnesota, April 1996. IEEE.

[Terrien *et al.*, 2000] G. Terrien, T. Fong, Ch. Thorpe, and Ch. Baur. Remote driving with a multisensor user interface. In *SAE 30th ICES*, Tolouse, France, July 2000.

[Thrun *et al.*, 1998] S. Thrun, W. Burgard, and D. Fox. A probabilistic approach to concurrent mapping and localization for mobile robots. *Machine Learning and Autonomous Robots*, 31/5:1–25, 1998.

[Thrun *et al.*, 2000] S. Thrun, W. Burgard, and D. Fox. A real-time algorithm for mobile robot mapping with applications to multi-robot and 3D mapping. In *ICRA'00*, volume 1, pages 321–328. IEEE, 2000.

[Thrun *et al.*, 2002] S. Thrun, D. Koller, Z. Ghahramani, H. Durrant-Whyte, and Ng. A.Y. Simultaneous mapping and localization with sparse extended information filters. In *Proceedings of the Fifth International Workshop on Algorithmic Foundations of Robotics*, Nice, France, 2002.

[Tomono, 2004] M. Tomono. A scan matching method using Euclidean invariant signature for global localization and map building. In *ICRA'04*, volume 1, pages 886–871. IEEE, 2004.

[Vandorpe *et al.*, 1996] J. Vandorpe, H. V. Brussel, and H. Xu. Lias: A reflexive navigation architecture for an intelligent mobile robot system. *IEEE Transactions on Industrial Electronics*, 43:432–44, June 1996.

[Veeck and Burgard, 2004] M. Veeck and W. Burgard. Learning polyline maps from range scan data acquired with mobile robots. In *IROS'04*, volume 2, pages 1065–1070. IEEE, 2004.

[Wang and Thorpe, 2002] Chieh-Chih Wang and Charles Thorpe. Simultaneous localization and mapping with detection and tracking of moving objects. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, volume 3, pages 2918–2924, Washington, DC, May 2002.

[Weiss and Puttkamer, 1995] G. Weiss and E. Puttkamer. A map based on laserscans without geometric interpretation. In *Intelligent Autonomous Systems - 4*, pages 403–407, Germany, 1995.

[Wetherill, 1986] G. B. Wetherill. *Regression analysis with applications*. Chapman and Hall, London, 1986.

[Wetteborn, 1993] H. Wetteborn. Laserabstandsermittlungsvorrichtung, 1993. German patent no. DE4340756A1 (November 30, 1993).

[Ye and Borenstein, 2002] C. Ye and J. Borenstein. Characterization of a 2-D laser scanner for mobile robot obstacle negotiation. In *Proceedings of the 2002 IEEE International Conference on Robotics and Automation*, pages 2512–2518, Washington DC, May 2002. IEEE.

[Zelinsky, 1994] A. Zelinsky. Using path transforms to guide the search for findpath in 2D. *I. J. Robotic Res. 13(4)*, pages 315–325, 1994.