A Discussion of Simultaneous Localization and Mapping

Udo Frese (`ufrese@informatik.uni-bremen.de`) *
Dept. of Computer Science, Universität Bremen
`http://www.informatik.uni-bremen.de/agbkb/`

Abstract. This paper aims at a discussion of the structure of the SLAM problem. The analysis is not strictly formal but based both on informal studies and mathematical derivation. The first part highlights the structure of uncertainty of an estimated map with the key result being "Certainty of Relations despite Uncertainty of Positions". A formal proof for approximate sparsity of so-called information matrices occurring in SLAM is sketched. It supports the above mentioned characterization and provides a foundation for algorithms based on sparse information matrices.

Further, issues of nonlinearity and the duality between information and covariance matrices are discussed and related to common methods for solving SLAM.

Finally, three requirements concerning map quality, storage space and computation time an ideal SLAM solution should have are proposed. The current state of the art is discussed with respect to these requirements including a formal specification of the term "map quality".

Keywords: mobile robot, navigation, simultaneous localization and mapping, SLAM, estimation, uncertainty, information matrix

1. Introduction

Navigation is the "*science of getting ships, aircraft, or spacecraft from place to place*" (Merriam-Webster's Collegiate Dictionary). It is also the science of getting mobile robots from place to place, a problem that is central to mobile robotics and has been subject to extensive research. According to Leonard and Durrant-Whyte (1992) this involves answering the three questions "*Where am I*", "*Where am I going?*" and "*How do I get there?*".

SLAM[1], i.e., the Simultaneous Localization and Mapping problem, aims at a fully autonomous answer to the question "Where am I?" by providing an autonomously built map. While moving through an environment the robot is required to derive a map from its perceptions and simultaneously determine its own position in this map. From the late eighties this problem has been explored and in recent years has received enormous attention.

This paper aims at discussing SLAM by pointing out what makes it special as an estimation problem. The first sections §2 to §5 as

* This article is based on research conducted during the author's Ph.D. studies at the German Aerospace Center (DLR) in Oberpfaffenhofen.

[1] Also called Concurrent Mapping and Localization (CML).

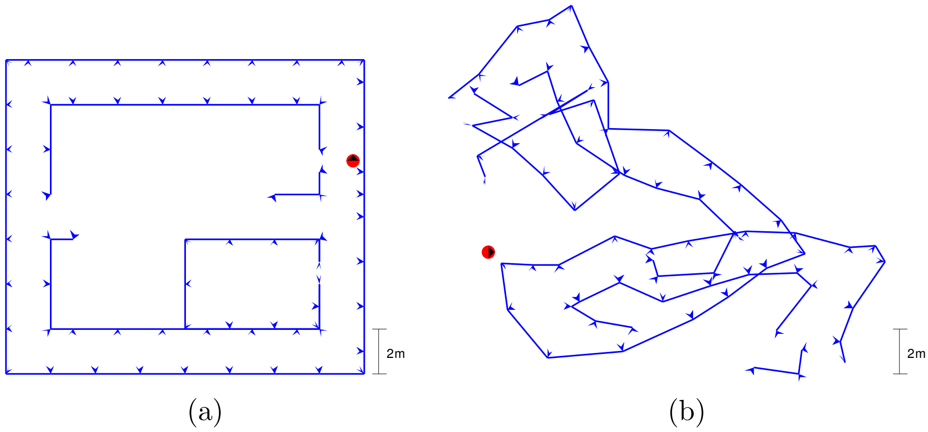


Figure 1. (a) Exact map of the building used as an example. (b) Map derived from raw odometry data. All maps are based on simulations in which the measurements are perturbed by comparatively large artificial noise and bias.

well as §12 characterize the structure of the inherent uncertainty of an estimated map with key findings being paraphrased as "*Certainty of Relations despite Uncertainty of Positions*". Formally, the central topic is the so called information matrix of all landmarks. Its structure corresponds to the discussed uncertainty structure. Section 11 sketches[2] a formal proof that the information matrix is approximately sparse, i.e., most entries are very small. This result substantiates the previous discussion and has been exploited by several algorithms.

Furthermore, the linearization error incurred in SLAM will be analyzed by discussing its sources and structure. The duality between information and covariance based representations will be explained and related to the "textbook" methods for solving SLAM, i.e., maximum likelihood, least squares, and Extended Kalman Filter (§6-§10).

In section 13 the SLAM problem is addressed from a different perspective intentionally taking an inexpert view, disregarding its known difficulty. This leads to proposal of a set of three requirements which an ideal SLAM algorithm should satisfy. These requirements concern map quality, storage space, and computation time. The discussion includes a mathematical formalization of the term "map quality". Section 14 briefly reviews the current state of the art in SLAM with respect to the proposed requirements, finding that since first publication of the requirements in (Frese and Hirzinger, 2001) SLAM algorithms have improved impressively, basically meeting all of them at present. Figure 1 shows the simulated example used as an illustration.

[2] Details published in (Frese, 2005b).

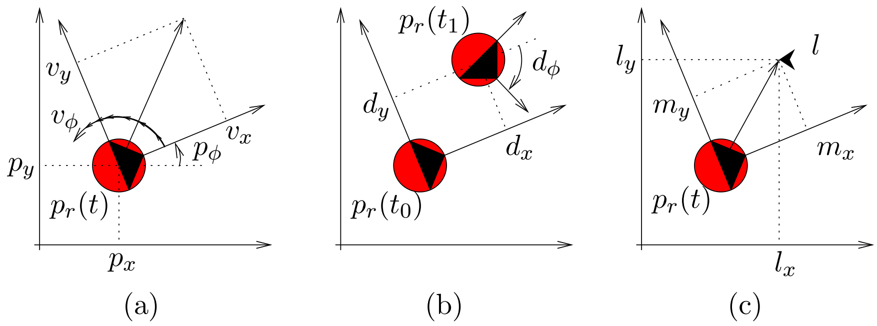


Figure 2. Coordinates describing the robot state $p_r = (p_x, p_y, p_\phi)^T$ and measurements: (a) Continuous odometry measurement: velocity in robot coordinates $v_r = (v_x, v_y, v_\phi)^T$ (b) Discrete odometry measurement: relative movement $d_r = (d_x, d_y, d_\phi)^T$ from $p_r(t_0)$ to $p_r(t_1)$ (c) Landmark measurement: relative position $m = (m_x, m_y)^T$ of landmark at $l = (l_x, l_y)^T$

2. Measurement Equations

When moving through a planar or nearly planar environment, the state of the mobile robot, its *pose*, is described by three variables, two for the robot position and one for the robot orientation (Fig. 2a). Similarly, a landmark is described by two variables for its position (Fig. 2c). For the purpose of this discussion a map is a state vector of landmark positions and robot poses. Usually the vector represents only the most recent robot pose but depending on context all poses or even no pose may be represented. Since SLAM is an estimation theoretic problem, the uncertainty of the measurements and map estimates is important. The uncertainty is usually described by a covariance matrix or alternatively by a so-called information matrix. Both are symmetric positive definite matrices, in which each row and column corresponds to one variable of the state vector.

In general, the uncertainty for an estimate is derived from an a-priori model for the measurement and measurement uncertainty. The measurement is defined by a measurement function that maps the system state, i.e., map and robot pose to the measurement.

In order to make the discussion independent from specific sensors it is assumed that the landmark sensor provides a landmark position and odometry provides the robot's velocity in robot coordinates with their corresponding covariance matrices. An alternative approach is to match sensor readings taken at two robot poses, e.g. laser scans, sonar or vision data to derive an estimate for their relative pose. Then, since the robot poses are reference frames for the sensor readings, they are treated as landmarks (Lu and Milios, 1997). The resulting equations are similar and the general discussion is still valid. Odometry measurements are continuous, occurring at every point in time[3]. This requires them to be integrated for some time, and then a discrete measurement of the path travelled to be derived as input for the SLAM algorithm.

Continuous Odometry Equations

The robot state is represented by 3 variables as $p_r = (p_x, p_y, p_\phi)^T$. Odometry provides an estimate of the robot's velocity $\hat{v}_r = (\hat{v}_x, \hat{v}_y, \hat{v}_\phi)^T$ in robot coordinates (Fig. 2a) with white Gaussian noise (covariance C_v). By rotating v_r by an angle of p_ϕ, v_r is transformed into world coordinates and the kinematic equation of the system is derived as

$$c := \cos p_\phi, \quad s := \sin p_\phi \tag{1}$$

$$\dot{p}_r = \begin{pmatrix} \dot{p_x} \\ \dot{p_y} \\ \dot{p_\phi} \end{pmatrix} = \begin{pmatrix} c & -s & 0 \\ s & c & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} v_x \\ v_y \\ v_\phi \end{pmatrix}. \tag{2}$$

A differential equation for the error covariance C_p of the odometry can be derived (Kelly, 2000; Frese, 2004) as

$$\dot{C}_p = B(p_\phi)\, C_v\, B(p_\phi)^T + A(p_\phi, v_x, v_y)\, C_p + C_p\, A(p_\phi, v_x, v_y)^T, \tag{3}$$

$$\text{with} \quad A(p_\phi, v_x, v_y) := \begin{pmatrix} 0 & 0 & -cv_y - sv_x \\ 0 & 0 & -sv_y + cv_x \\ 0 & 0 & 0 \end{pmatrix}, \quad B(p_\phi) := \begin{pmatrix} c & -s & 0 \\ s & c & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{4}$$

Equations (2) and (3) are computed[4] in the control loop of the mobile robot exploiting the high sensor rate of motor encoders.

Discrete Odometry Equations

Between two landmark observations a SLAM algorithm simply updates the robot's position estimate according to odometry. When the next landmark observation becomes available a more complex update is necessary. Thus, the odometry data between two landmark observations at time t_0 and t_1 is accumulated. It is then passed to the SLAM algorithm in the form of a discrete step $\hat{d}_r = (\hat{d}_x, \hat{d}_y, \hat{d}_\phi)^T$ with corresponding covariance C_d (Fig. 2b). The robot controller continuously integrates (2) providing $\hat{p}_r(t)$ and (3) providing $C_p(t)$. So $\hat{d}_r$ must be computed

[3] From the perspective of control theory. Technically the sensor, i.e., wheel encoder is usually sampled at the rate of the control loop.

[4] Numerically integrating (3) for a time step Δt by Euler integration may lead to non-positive definite results later in (8). It is advisable to integrate a small discrete step $\Delta t\, v$ using (5) and (7) instead.

from the values $\hat{p}_r(t_0), \hat{C}_p(t_0)$ and $\hat{p}_r(t_1), C_p(t_1)$ of the odometry by solving the odometry measurement equation

$$\begin{pmatrix} p_x(t_1) \\ p_y(t_1) \\ p_\phi(t_1) \end{pmatrix} = \begin{pmatrix} p_x(t_0) \\ p_y(t_0) \\ p_\phi(t_0) \end{pmatrix} + \begin{pmatrix} c & -s & 0 \\ s & c & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} d_x \\ d_y \\ d_\phi \end{pmatrix} \tag{5}$$

for d and applying to the estimates. The Jacobians with respect to $p_r(t_0)$ and d_r are

$$J_1 := \frac{\partial p_r(t_1)}{\partial p_r(t_0)} = \begin{pmatrix} 1 & 0 & -sd_x - cd_y \\ 0 & 1 & cd_x - sd_y \\ 0 & 0 & 1 \end{pmatrix}, \quad J_2 := \frac{\partial p_r(t_1)}{\partial d_r} = \begin{pmatrix} c & -s & 0 \\ s & c & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{6}$$

The corresponding equation for the covariance C_d of $\hat{d}_r$ follows from the white noise assumption for $\hat{v}_r$. Both $\hat{p}_r(t_0)$ and $\hat{d}_r$ are independent and the covariance for $\hat{p}_r(t_1)$ can be expressed as

$$C_p(t_1) = J_1 C_p(t_0) J_1^T + J_2 C_d J_2^T. \tag{7}$$

Using $J_2^{-1} = J_2^T$ (5) is solved for $\hat{d}_r$ and (7) for C_d yielding the step

$$\hat{d} = J_2^T \left(\hat{p}_r(t_1) - \hat{p}_r(t_0)\right), \quad C_d = J_2^T \left(C_p(t_1) - J_1 C_p(t_0) J_1^T\right) J_2 \tag{8}$$

passed to the SLAM algorithm. Equation (5) expresses odometry as a dynamic equation which maps the old state $p_r(t_0)$ and the measurement d_r to the new state $p_r(t_1)$. The corresponding measurement equation which maps old and new state to measurement is

$$\begin{pmatrix} d_x \\ d_y \\ d_\phi \end{pmatrix} = \begin{pmatrix} c & s & 0 \\ -s & c & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} p_x(t_1) - p_x(t_0) \\ p_y(t_1) - p_y(t_0) \\ p_\phi(t_1) - p_\phi(t_0) \end{pmatrix} \tag{9}$$

$$J_3 := \frac{\partial d_r}{\partial p_r(t_0)} = \begin{pmatrix} -c & -s & d_y \\ s & -c & -d_x \\ 0 & 0 & -1 \end{pmatrix}, \quad J_4 := \frac{\partial d_r}{\partial p_r(t_1)} = \begin{pmatrix} c & s & 0 \\ -s & c & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{10}$$

Landmark observation

A huge amount of different landmarks and landmark sensors have been proposed in the literature (Cox and Wilfong, 1990). Here discussion will be restricted to point landmarks in the plane $(l_x, l_y)^T$ and sensors which locate the landmark relative to the robot $(m_x, m_y)^T$ (Fig. 2c) with covariance C_m. At the moment it is assumed, that the landmarks can be identified (see §15). The measurement equation and Jacobians are

$$\begin{pmatrix} m_x \\ m_y \end{pmatrix} = \begin{pmatrix} c & s \\ -s & c \end{pmatrix} \begin{pmatrix} l_x - p_x \\ l_y - p_y \end{pmatrix} \tag{11}$$

$$J_5 := \frac{\partial m}{\partial p_r} = \begin{pmatrix} -c & -s & m_y \\ s & -c & -m_x \end{pmatrix}, \quad J_6 := \frac{\partial m}{\partial l} = \begin{pmatrix} c & s \\ -s & c \end{pmatrix}. \tag{12}$$

In (10) and (12) the measured quantity m_x, m_y and d_x, d_y, i.e. the position relative to the robot appears. However this is not the actual value measured but rather the corresponding relative position in the state chosen as linearization point (cf. §7).

3. Error Accumulation

If we consider the the robot moving through a known environment i.e. by using an a-priori map, or in a region already mapped by SLAM then the uncertainty of the robot's poses (position) remains bounded. This is because each observation of two landmarks essentially reduces the uncertainty down to the landmark's uncertainty plus the uncertainty of the observation.

However, if the robot moves through an unknown region the uncertainty of its pose in absolute coordinates will get arbitrarily large because the odometry error accumulates over time (Fig. 1b). The uncertainty can be greatly reduced by fusing odometry with several measurements of a new landmark as the landmark is passed by (Fig. 3a). For most sensors this produces much better results than just using odometry (Thrun et al., 1998). Nevertheless, estimating the robot's position after traveling a long distance is still subject to accumulated error: due to the limited sensor range the position is derived from a chain of several relations between successive landmarks. Sometimes a compass can help to reduce the problem, although this is unreliable in many buildings.

The fact that errors may accumulate to arbitrarily high values distinguishes SLAM from many other estimation problems and gives rise to the problems discussed in §4 and §9.

4. Representation of Relativity

The author believes that the dominant aspect of SLAM is modeling

Certainty of Relations despite Uncertainty of Positions.

This may be called *ability* to *represent relativity*. In the example scenario, for instance, position and orientation of the room will be quite uncertain, while its shape will be highly certain (Fig. 3a).

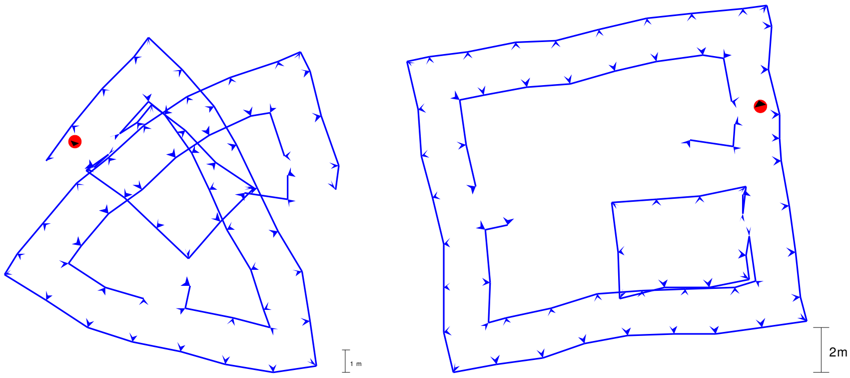
If the robot moves through an unknown region and observes a sequence of landmarks, the uncertainty of relative positions of the landmarks only depends on the measurement errors of the landmarks by the robot and on the odometry error between those measurements. So the most precisely known relations are those concerning the relative location of adjacent landmarks.

The uncertainty of the absolute robot pose before observing the first landmark however increases the uncertainty of the absolute position of all landmarks, acting as an unknown rigid-body transformation on the whole set of observed landmarks. As the absolute robot pose is subject to error accumulation, the common situation is that relations are fairly certain, whereas absolute positions can be arbitrarily uncertain. In large maps this effect can appear at different scales: the relative positions of some landmarks in a room are much more precisely known than the position of the room in the building, which seen as a relative position with respect to other rooms is in turn much more precisely known than the absolute position of the building.

Thus, a SLAM system should be able to represent the certainty of relations between landmarks despite large uncertainty in the absolute position of the landmarks. In particular, assigning a single uncertainty value to each landmark only is insufficient (Uhlmann et al., 1997).

In the theory of SLAM it is an extremely important insight (Newman, 1999) that the uncertainty of any relation converges to zero when repeatedly moving through the same environment. This theorem clarifies the uncertainty structure in the limit, being of theoretical interest, but in general this approach is probably neither practical nor necessary. Most applications can be based exclusively on relative information: When navigating, for instance, path planning will result in a sequence of way-points. The location of each way-point will be known relative to the surrounding landmarks. So the robot, knowing its own pose relative to those landmarks, will be able to navigate from one way-point to the next. A path defined this way will even remain valid when the map changes significantly while the robot is moving.

So it is important to focus on the behavior when moving through the building for the first time. The structure of uncertainty is still complex and a single measurement may have a significant effect on the estimate. This effect is most prominent and probably the most important test case in general when closing large loops.



(a) (b)

Figure 3. Closing the loop: (a) before (b) after integration. An animation of several random outcomes of this mapping process can be downloaded from the author's web site (Frese, 2005a).

5. Implications of Closing the Loop

Assume the robot moves along a closed loop and returns to the beginning of the loop but has not yet re-identified any landmark, so this is not known to the robot. Typically, the loop is not closed in the map due to the error accumulated along the loop (Fig. 3a).

At the beginning of the loop a landmark is re-identified and the corresponding measurement is integrated into the map causing the loop to become closed. To achieve this, the SLAM system has to "deform" the whole loop to incorporate the information of a connection between both ends of the loop without introducing a break somewhere else (Fig. 3b).

This goal is sometimes referred to as the map being "topologically consistent" or "globally consistent" (Lu and Milios, 1997; Duckett et al., 2002), meaning that two parts of the map are represented to be adjacent if and only if this was observed by the robot. Within a landmark based approach adjacency is not explicitly modeled. Topological consistency has to be interpreted in the sense that two landmarks are represented as being near to each other (the distance being low with low uncertainty), if and only if this was observed by some measurement.

It has to be emphasized that correct treatment of uncertainty contained in the measurements will implicitly yield the necessary deformation. More specifically, the precisely known relative location of each landmark with respect to adjacent landmarks prevents any break in the loop: if there was a break the relative positions of the landmarks on both sides of the break would be highly incorrect, thus being inconsistent with the measurements made in this vicinity. So the map estimate which is consistent with all measurements automatically deforms smoothly when closing large loops.

An important insight is that any representation of the uncertainty of a map estimate must be able to “represent relativity” in order to achieve this kind of behavior.

In a landmark based approach two corridors can overlap (Fig. 3a) if the map is sufficiently uncertain. This is because negative information, i.e. a landmark is not seen although it should be seen is not included in such a framework in contrast to dense approaches modeling the map as an evidence grid (Murphy, 1999; Hähnel et al., 2003; Grisetti et al., 2005). The situation can be resolved by looking at the uncertainty information. If the robot is in one corridor the landmarks of that corridor are represented to be nearby with low uncertainty. The landmarks of the other corridor are represented also as nearby but with high uncertainty.

6. Maximum Likelihood Estimation

At the moment let us assume that the measurements are disturbed by independent Gaussian measurement errors with a-priori known covariance, that data association, i.e. the identity of an observed landmark is known and that computation time is no issue. In this case there is a theoretically thorough optimal solution, namely the maximum likelihood (ML) solution (Press et al., 1992, §15.1). It dates back to Gauss (1821) who invented the Gaussian distribution, least square estimation, and the Gaussian algorithm for solving the resulting equation. He used these results, in some sense for SLAM, to survey the kingdom of Hanover from 1818 until 1826.

The ML estimate is based on the a-priori known probability distribution for the measurement given the map, i.e. on the conditional probability distribution of the measurement for a fixed map. After the measurement has been made, this distribution is interpreted as a likelihood distribution for the maps given the measurement. The map with the largest likelihood is the ML estimate. Of all maps it has the largest probability of causing the observed measurements and thus is optimal in this sense. It is also the map with the largest probability if the true map is assumed to be drawn from a uniform a-priori distribution. By definition of Gaussian errors the likelihood for a map given a single measurement y_i is

$$p_i(x) \propto e^{-\frac{1}{2}q_i(x)}, \quad q_i(x) := (y_i - f_i(x))^T C_i^{-1}(y_i - f_i(x)). \tag{13}$$

Assume that the data under consideration consist of n landmarks, p robot poses, and m measurements, the landmark positions and different robot poses form the parameter vector x having size $2n+3p$ in the equation. The vector y_i is the i-th measurement, C_i its covariance, and $f_i(x)$ is the corresponding measurement equation, i.e., the value the measurement should have if the landmark and robot poses were x.

The likelihood of x given all measurements is the product of the individual likelihoods, due to the assumption of stochastic independence:

$$p(x) \propto \prod_{i=1}^{m} e^{-\frac{1}{2}q_i(x)} = e^{-\frac{1}{2}\chi^2(x)}, \quad \text{with} \quad \chi^2(x) := \sum_{i=1}^{m} q_i(x) \tag{14}$$

$$\hat{x}_{\mathrm{ML}} = \arg\max_x p(x) = \arg\min_x \chi^2(x). \tag{15}$$

In order to find the numerical minimum, a least squares nonlinear model fitting algorithm such as for instance Levenberg-Marquardt can be employed (Press et al., 1992, §15.5), by iteratively linearizing the measurement equations. The linearized equations yield a quadratic approximation to χ^2, the minimum of which can be found by solving a large linear equations system. This approach requires all old robot poses to be represented in the equations system, which consequently has $O(n+p)$ equations and variables. Solving such a system needs $O((n+p)^3)$ computation time. So this approach is not a practical solution for SLAM in real time. Its invaluable benefit, however, lies in the fact that it can provide a gold-standard for discussion and for comparison with efficient approaches. Figure 3 has been computed this way.

7. Linear Least Squares

If the measurement functions f_i are linearized at some point x_i^0 with Jacobian J_{f_i}, the χ^2 function becomes quadratic:

$$f_i^{lin}(x) = f_i(x_i^0) + J_{f_i}(x_i^0)(x - x_i^0) \tag{16}$$

$$\chi^2_{lin} = \textstyle\sum_{i=1}^{m} \left(y_i - f_i^{lin}(x)\right)^T C_i^{-1} \left(y_i - f_i^{lin}(x)\right) \tag{17}$$

$$= x^T \underbrace{\left(\sum_{i=1}^{m} J_{f_i}(x_i^0)^T C_i^{-1} J_{f_i}(x_i^0)\right)}_{A} x + x^T \underbrace{\left(2\sum_{i=1}^{m} J_{f_i}(x_i^0)^T C_i^{-1}\left(y_i - f_i(x_i^0) + J_{f_i}(x_i^0)x_i^0\right)\right)}_{b} + \text{const}\,. \tag{18}$$

Such a quadratic function can always be represented as $x^T A x + x^T b +$ const, with a symmetric positive definite (SPD) matrix A and a vector

b. The linearized least squares (LLS) estimate $\hat{x}_{\mathrm{LLS}}$ being the same as the linearized maximum likelihood estimate, can be computed by

$$\hat{x}_{\mathrm{LLS}} = \arg\min_{x} \chi^2_{lin}(x) = A^{-1}b/2. \tag{19}$$

The matrix A is called information matrix. Its inverse A^{-1} is the error covariance of the estimate $\hat{x}_{\mathrm{LLS}}$ (Press et al., 1992, §15.6). High entries in A correspond to precisely known relations. This matrix is sparse and has an important structure that is discussed in §11.

The quality of a linearized least squares estimate depends on the points of linearization chosen: if all measurements are *always* linearized at the latest estimate and the whole process is iterated until convergence, the limit is the ML estimate. Actually this is the way nonlinear least squares algorithms such as Levenberg - Marquardt work. However this approach involves re-evaluating all Jacobians and thus storing all measurements.

Another approach is to linearize each measurement once and forever at the estimate in the moment of measurement. There is a subtle difference whether the estimate before or after integrating the measurement is used. Due to error accumulation the prior estimate for a relative landmark position $(d_x, d_y)^T$ can be arbitrarily bad and much worse than the actual value measured. Using the posterior estimate is best but requires two iterations so an alternative is to use the prior for recently observed landmarks and the measured value otherwise.

When the linearization point is not changed after integrating a measurent, the measurements can be accumulated in a matrix A and vector b. Each measurement involves only few variables, so J_{f_i} is sparse and accumulation can be performed in $O(1)$. Nevertheless, to provide an estimate $A\hat{x} = b/2$ has to be solved, which takes $O((n+p)^3)$ or $O((n+p)^2)$ exploiting sparsity. The estimate is subject to linearization error depending on the error in the estimates used for linearization (§9).

It is important to note that the information matrix A represents all landmark positions and all robot poses. Thus, the size of the representation still grows even when moving through an already mapped area. This problem can be avoided by marginalizing out, i.e. removing, old robot poses by computing the so-called Schur complement. The resulting information matrix P' of all landmarks and the actual robot pose is no longer sparse any more. It is no longer sparse in the strict sense that most entries are exactly $= 0$. However as Thrun et al. (2002) observed, it is still approximately sparse, i.e. most entries are very small, so ≈ 0. In §11 a proof for approximate sparsity of P' will be sketched. By this theorem P' can be replaced by a sparse approximation, so the matrix has only $O(n)$ entries and equation solving can be performed in $O(n^2)$.

8. Extended Kalman Filter

The Extended Kalman Filter (EKF) (Gelb, 1974) is the tool most often applied to SLAM (Smith et al., 1988; Hebert et al., 1995; Castellanos et al., 1999) using the same measurement equations as for ML estimation. The EKF integrates all measurements into a covariance matrix C_{EKF} of the landmark positions and the actual robot pose without any measurements to be stored afterwards. The estimate $\hat{x}_{\mathrm{EKF}}$ provided is the same as for linearized least squares, so EKF suffers the same linearization problems as will be discussed in §9.

Since the EKF maintains a covariance matrix instead of an information matrix, marginalization of old robot poses is no problem but can simply be done by removing corresponding rows and columns. In fact, most implementations simply replace the old robot pose with the new one whenever integrating an odometry measurement.

The key problem is to update the covariance matrix C_{EKF} after a landmark observation. From (18) it can be seen that a single term $J^T C_i^{-1} J$ is added to the information matrix A. Since C_i is a 2×2 matrix, $J^T C_i^{-1} J$ has rank 2 and the corresponding change in $C = A^{-1}$ and $\hat{x} = A^{-1}b/2$ can be efficiently computed via the Woodbury formula (Press et al., 1992, §2.7) resulting in the well known EKF update equation.

Compared to $O((n+p)^3)$ for linearized least squares, EKF is moderately efficient taking $O(n^2)$ computation time. But still this is so much that EKF can only be used for small environments ($n \lessapprox 100$).

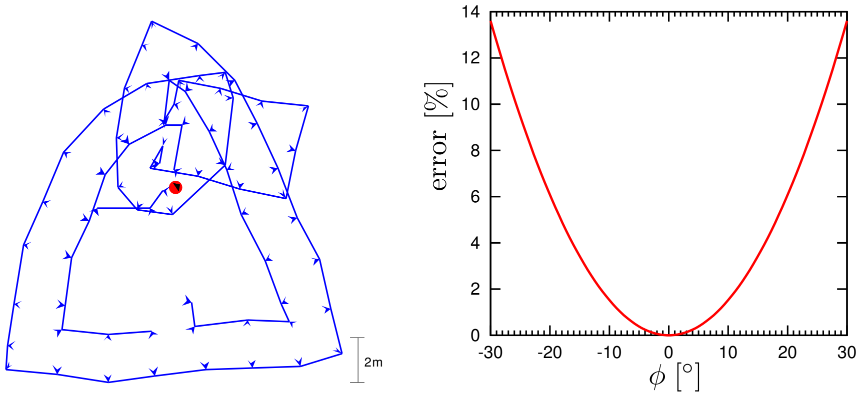
9. Linearization Error

There are two sources for a linearization error: the error of the robot's orientation estimate $\hat{p}_\phi$ (*orientation error*) and the error of the measured quantities d_x, d_y and m_x, m_y. This important fact can be seen when transforming the Jacobians $J_1 \dots J_6$ of the measurements into robot coordinates:

$$R_2 := \begin{pmatrix} c & -s \\ s & c \end{pmatrix}, \qquad R_3 := \begin{pmatrix} c & -s & 0 \\ s & c & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{20}$$

$$J_1 = R_3 \begin{pmatrix} 1 & 0 & -d_y \\ 0 & 1 & d_x \\ 0 & 0 & 1 \end{pmatrix} R_3^T, \qquad J_2 = R_3 \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \tag{21}$$

$$J_3 = \begin{pmatrix} -1 & 0 & d_y \\ 0 & -1 & -d_x \\ 0 & 0 & -1 \end{pmatrix} R_3^T, \qquad J_4 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} R_3^T, \tag{22}$$



(a) (b)

Figure 4. Linearization error: (a) closing the loop with EKF / LLS (b) linearization error of linearized rotation $\left|\begin{pmatrix}\cos\phi\\ \sin\phi\end{pmatrix} - \begin{pmatrix}1\\ \phi\end{pmatrix}\right|$ as a function of angular error ϕ

$$J_5 = \begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix} R_2^T, \;\; J_6 = \begin{pmatrix}-1 & 0 & m_y\\ 0 & -1 & -m_x\end{pmatrix} R_3^T. \qquad (23)$$

The variables d_x, d_y, m_x, m_y involved in the transformed Jacobians are directly measurable. So their errors do not accumulate and are often rather small. Still Julier and Uhlmann (2001) showed that the EKF can diverge because it linearizes different measurements with inconsistent m_x, m_y estimates. Linearization is even more difficult for the rotation matrices R_2, R_3 depending on p_ϕ. When moving through an unmapped area the orientation error accumulates. In practical settings errors of 45° may easily be exceeded rendering all linearizations of sine and cosine useless. The effect of processing the example scenario with EKF instead of using ML estimation is disastrous (Fig. 4a). The beginning and end of the loop do not match and, even worse, the room although precisely known gets significantly larger than before. The reason for this is that EKF would have to move and rotate the room implicitly to make the map consistent. Instead, a rotation linearizing the angle at 0 is performed:

$$\mathrm{Rot}(\phi) := \begin{pmatrix}\cos\phi & -\sin\phi\\ \sin\phi & \cos\phi\end{pmatrix} \xrightarrow{\phi\approx 0} \begin{pmatrix}1 & -\phi\\ \phi & 1\end{pmatrix} = \sqrt{1+\phi^2}\cdot \mathrm{Rot}(\arctan\phi)$$

The consequence is that the room is larger than before and rotated by too small an angle (Fig. 4b).

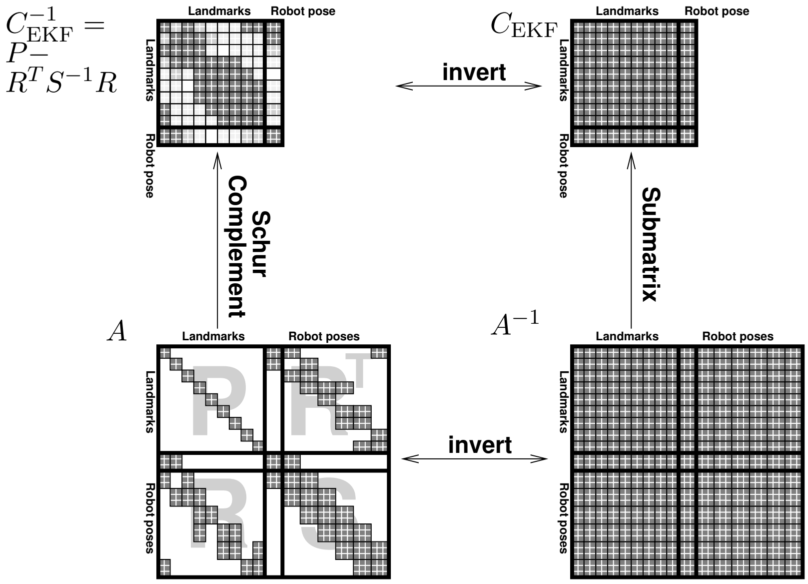


Figure 5. Relation between the LLS information matrix $A = \left(\begin{smallmatrix} P & R^T \\ R & S \end{smallmatrix}\right)$ which represents all robot poses and the covariance matrix C_{EKF} used by EKF only representing the actual robot pose. A^{-1} is the covariance matrix corresponding to A. C_{EKF} is derived from A^{-1} as a submatrix marginalizing out old robot poses . Accordingly C_{EKF}^{-1} is the information matrix corresponding to C_{EKF}. It can be directly derived from A via Schur complement. So on the whole, taking a submatrix of a covariance matrix is equivalent to applying Schur complement to an information matrix.

10. Covariance vs. Information Matrices

Covariance and information matrices are complementary representations of uncertainty, since one is the inverse of the other. This duality extends to the operation of taking a submatrix, which is equivalent to applying Schur complement in the inverse (Press et al., 1992, §2.7) (Fig. 5). Of particular interest is the decomposition $A = \left(\begin{smallmatrix} P & R^T \\ R & S \end{smallmatrix}\right)$ with rows and columns of the first block corresponding to landmarks (maybe including the current robot pose) and rows and columns of the second block corresponding to (old) robot poses. In this case $P' = (P - R^T S^{-1} R)^{-1}$ is the covariance matrix of all landmarks (and the current robot pose) as used by the EKF.

The Schur complement $P - R^T S^{-1} R$ equals the corresponding submatrix P minus a correction term $R^T S^{-1} R$. This term can be thought of as somehow "transferring" the effect of S into the realm of P via a mapping provided by the off-diagonal block R^T.

Taking a submatrix of the information matrix or applying Schur complement to the covariance matrix corresponds to conditioning, i.e.

random variables (landmark positions, robot poses) in the removed rows and columns being exactly known. Conversely taking a submatrix of the covariance matrix or applying Schur complement to the information matrix corresponds to marginalization. This means random variables in the removed rows and columns being unknown, i.e., all information about them is discarded.

The main difference between information and covariance matrix lies in the representation of indirect relations. Assume the robot is at pose P1 observing landmark L1 and moves to P2 observing L2. The measurements directly define relations P1-L1, P1-P2, P2-L2, indirectly constituting a relation L1-L2. The covariance matrix explicitly stores this relation in the off-diagonal entries corresponding to L1-L2, whereas the information matrix does not.

Thus the information matrix $A = \left(\begin{smallmatrix} P & R^T \\ R & S \end{smallmatrix}\right)$ used by LLS is sparse, having non-zero entries only for random variables involved in a common measurement. Although A is sparse the Schur complement $P - R^T S^{-1} R$ is dense, because S^{-1} is dense. However it turns out that it is *approximately sparse* (Fig. 5) with an off-diagonal entry $(P - R^T S^{-1} R)_{l_1 l_2}$ corresponding to two landmarks l_1, l_2 decaying exponentially with the distance traveled between observation of l_1 and l_2.

11. Sparsity of SLAM Information Matrices

In this section the central result for the SLAM uncertainty structure is derived, by stating that the information matrix appearing in SLAM is approximately sparse:

> *In the SLAM information matrix off-diagonal entries corresponding to two landmarks decay exponentially with the distance traveled between observation of the first and second landmarks.*

This result is important both for computation and analysis. First, the approach of saving space and computation time by making the information matrix sparse is being confirmed. This approach has been proposed by Frese and Hirzinger (2001) and is the basis of the algorithm presented in (Frese, 2004). It is also utilized in the well known work of Thrun et al. (2002) on sparse extended information filters (SEIF). Second the result implies that the large scale uncertainty of a map estimate is generated by local uncertainties composed along the path the robot has been traveling. Thus, in contrast to the local uncertainty structure, it is simple and dominated by the map's geometry (§12).

Proof Outline

Due to lack of space, only the formal theorem and a sketch of the proof will be given here, the complete derivation can be found in (Frese, 2004; Frese, 2005b).

First, the structure of $A = \left(\begin{smallmatrix} P & R^T \\ R & S \end{smallmatrix}\right)$ is analyzed. It is a block matrix with the first block row / column corresponding to the landmarks and the second corresponding to the different robot poses. This is the same situation as shown in figure 5 but with the current robot pose included in S. As discussed in the previous section, the diagonal blocks P and S are information matrices of two related subproblems: P is the information matrix of the mapping subproblem, describing the uncertainty of the landmarks, if the robot poses were known. Conversely, S is the information matrix of the localization subproblem, describing the uncertainty of the robot poses, if the landmarks were known. Both matrices are extremely sparse: P is block diagonal and S is block tridiagonal.

The matrix under investigation will be the information matrix of the landmarks only, i.e., without robot poses. It is $P' = P - R^T S^{-1} R$ by Schur complement. The role of R^T in this formula is to provide a mapping from robot poses to landmarks. It creates an off-diagonal entry between two landmarks, whose magnitude depends on the entry in S^{-1} corresponding to the two robot poses these landmarks have been observed from. S^{-1} is the covariance of all robot poses given the position of all landmarks. Hence the magnitude of an off-diagonal entry corresponding to two landmarks depends on the covariance the corresponding robot poses had if all landmark positions were known.

This covariance is well understood because it is the covariance of different robot poses when localizing in an a-priori map. Consider what happens when localizing for instance with an EKF: in each step the pose estimate is replaced by a weighted sum of the old estimate and the measurements of observed landmarks. The covariance with a fixed old robot pose is thereby reduced by a constant factor. So the covariance between two robot poses decreases exponentially with the number of steps traveled in between leading to small off-diagonal entries in S^{-1}.

So overall the off-diagonal entry between two landmarks, in P' decreases exponentially in the number of steps between observations of those two landmarks making the matrix approximately sparse. Below the formal theorem is stated.

THEOREM 1 (Information Matrix Sparsity). *Consider a sequence of odometry and landmark observations with parameter ω. Then the resulting SLAM information matrix of all landmarks P' is approximately* *sparse. The off-diagonal block $P'_{l_1 l_2}$ corresponding to two landmarks $l_1 \neq l_2$ decays exponentially with the smallest number of steps $d_{l_1 l_2}$ traveled between observation of l_1 and l_2.*

$$\frac{\|P'_{l_1 l_2}\|}{\min\left\{\|P'_{l_1 l_1}\|, \|P'_{l_2 l_2}\|\right\}} = O\left(\left(1 + \frac{4}{3}\omega\right)^{-d_{l_1 l_2}}\right) \tag{24}$$

Proof. Published in (Frese, 2005b) including a formal definition of the parameter ω.

The parameter ω describes the ratio between information gained from landmark observations and transported from the last robot pose via odometry. It is a sensor characteristic and remains bounded even if the map grows. It governs the rate of exponential decay and thereby how sparse "approximately sparse" means. An important insight is that the more precise landmark observation is compared to odometry, the smaller are the resulting off-diagonal entries. In the limit of no odometry at all the matrix is exactly sparse (Walter et al., 2005). This corresponds well to intuition, because with precise odometry spatial information can be transported over large distances creating long range coupling entries, whereas with imprecise odometry information about landmarks observed long ago quickly vanishes.

12. Local vs. Global Uncertainty

It can be observed that there is a qualitative difference between local and global structures of SLAM, i.e., between relations of neighboring and of distant landmarks. Roughly speaking, the local uncertainty is small but complex and depends on actual observations, whereas the global uncertainty is large, rather simple and dominated by the map's geometry. This is a consequence of the preceding theorem:

The measurements themselves define independent relations between landmarks and robot poses. For most sensors the uncertainty depends on the distance (laser scanner, stereo vision) or is even infinite in one dimension (mono vision). The information provided by the set of landmark observations from a single robot pose contains a highly coupled uncertainty originating from the uncertainty of the robot pose. From successive robot poses similar but different sets of landmarks are usually observed. So the parts of the information corresponding to different robot poses are highly coupled, but are always coupling different sets of landmarks. As a result the overall information on a local scale is also

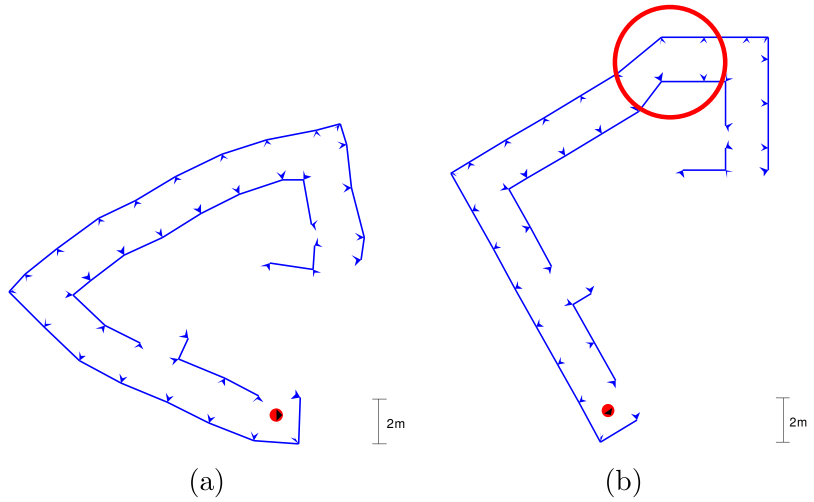


Figure 6. (a) measurement uncertainty everywhere (b) measurement uncertainty only in the encircled region.

highly coupled and very complex. This corresponds to the entries $P'_{l_1 l_2}$ in the information matrix being high for landmarks l_1, l_2 that are near to each other.

On a global scale the structure of the information is governed by theorem 1. The coupling entry $P'_{l_1 l_2}$ between distant landmarks is very low. So the uncertainty of the relation between them is approximately the composition of local uncertainties along the path from l_1 to l_2:

Consider the information matrix resulting from the integration of several local pieces of information, for instance, the distance of each landmark to any other landmark nearby. By (18), this matrix is the sum of the information matrices for each piece of information. Each of them has non-zero coupling entries only for the landmarks involved. So the overall information matrix is sparse with all coupling entries being zero, except those of adjacent landmarks.

Thus, as local information corresponds to a sparse information matrix, an approximately sparse information matrix corresponds to information that can be approximately viewed as being the integration of local information. To appreciate the uncertainty structure of such information, imagine that measurement noise applies only to the measurements in a small region (Fig. 6b). The noise corrupts the robot position and orientation estimate when the robot moves through the region. Thus the part of the map built afterward is affected by an uncertain translation and rotation relative to the part before. The effect of the rotation around the region grows linearly with the distance to the region. So globally it is much larger than the uncertain translation. The magnitude of the rotation angle depends on the local uncertainty in the region. Its structure, however, solely depends on the distances of the different landmarks to the region, i.e., on the map's geometry.

If all measurements are uncertain, the global effect is approximately the sum of an uncertain rotation for each local region. The resulting uncertainty structure can best be described as an *uncertain bending* of the map (Fig. 6a). Compared to local uncertainty it is much larger, but simpler because the maps geometry dominates its structure. From the author's web site (Frese, 2005a) an animation visualizing this uncertain bending can be downloaded. It animates several random outcomes of the mapping process before and after closing the loop (Fig. 3). The "certainty of relation despite uncertainty of position" principle can also be seen. Every uncertain aspect of the map, e.g. the room's pose, moves a lot whereas every aspect that is certain, e.g. the room's shape, moves only a little in the animation.

13. Requirements for an Ideal Solution

In this section some requirements, which an ideal SLAM solution should fulfill, are postulated. They are based on an intentionally inexpert view of the problem disregarding its apparent difficulty, but asking how mapping should work when based on a common sense understanding of maps. These requirements were first proposed by Frese and Hirzinger in 2001, at which time they were largely unfulfilled by existing algorithms. Since then a lot of new efficient algorithms have been proposed that will be discussed with respect to these requirements in §14.

Quality, Storage Space and Computation Time

> **(R1) Bounded Uncertainty** *The uncertainty of any aspect of the map should not be much larger than the minimal uncertainty that could be theoretically derived from the measurements.*

This requirement is quite general saying all that can be known from the measurements should at least be roughly represented in the map. Consistently approximating some relations for the sake of efficiency is acceptable to the extent that relations get slightly less precise, but without losing all or almost all information on certain relations. Since many relations can be known precisely from the measurements, not representing one would violate the principle stated and hence (R1) implies the ability to represent relativity and hence to close large loops achieving topologically consistent maps.

> **(R2) Linear Storage Space** *The storage space of a map covering a large area should be linear in the number of landmarks ($O(n)$).*

The soundness of this requirement can be seen from the following example: imagine a building consisting of two parts, A and B, being connected by a few doorways. Then the map of the whole building consists of the map of both parts plus some information concerning the connections and should thus have a size only slightly larger than the size of a map of A plus the size of a map of B.

Simply storing all measurements will not meet (R2), since the storage space is proportional to the number of measurements m, not to the number of landmarks n. Thus, the map's size would grow during motion even when repeatedly traveling through the same area.

> **(R3) Linear Update Cost** *Incorporating a measurement into a map covering a large area should have a computational cost at most linear in the number of landmarks ($O(n)$).*

Establishing this requirement is more difficult than the preceding one: Let us assume that the setting above holds with a measurement made in A. At first the measurement has to be incorporated into the map of A, taking into account the known effect of A on the connection between A and B. Then, the effect of these connections onto B must be computed. This is equivalent to incorporating *several* measurements concerning the connections into the map of B. However computation can be deferred until the robot actually enters B, sharing the computational cost with all other measurements that have generated effects on the connections. As the number of landmarks in the connections is small, this should not take more time than incorporating the original measurement into the map of A.

So the total cost for integrating a measurement into a map of A and B should not be larger than the cost of integration into A plus the cost of integration into B, thus being linear in the number of landmarks.

(R1) states that the map shall represent nearly all information contained in the measurements, thus binding the map to reality. (R2) and (R3) regard efficiency, requiring linear space and time consumption.

Requirement (R2) and (R3) concerning storage space and computation time refer to criteria which are canonically applied to any algorithm. However requirement (R1) must still be formalized appropriately:

DEFINITION 1 (Aspect). *An* aspect *of a map is a function f mapping the landmark positions to a real number $f(x)$*

$$f : \mathbb{R}^{2n} \longrightarrow \mathbb{R}. \tag{25}$$

Examples for aspects of a map are: a landmark's $x-$ or $y-$ coordinate, the distance between two landmarks, the angle between three landmarks or any linear combination of landmark coordinates. Considering the SLAM uncertainty structure ("Certainty of Relations despite Uncertainty of Positions") a relation is consequently an aspect of the map, being invariant under rigid-body transformation of the whole map

$$f_{\mathrm{rel}}(\mathrm{Rot}_\alpha\, x + \mathrm{Trans}_d) = f_{\mathrm{rel}}(x) \quad \forall \alpha, d, \tag{26}$$

where Rot_α is a rotation matrix, rotating the whole map by α and Trans_d is a vector translating the whole map by d.

The uncertainty of an aspect f of a map estimate $\hat{x}$ is its standard deviation $\sqrt{\mathrm{cov}(f(\hat{x}))}$. In terms of (R1) the "minimal uncertainty that could be theoretically derived from the measurements" is the corresponding standard deviation of the optimal maximum likelihood estimate $\sqrt{\mathrm{cov}(f(\hat{x}_{\mathrm{ML}}))}$. The ratio between those uncertainties indicates how much error is induced by the estimation algorithm and how much error is caused by the sensors. So the *relative uncertainty*

$$\mathrm{ruc}(f) := \frac{\sqrt{\mathrm{cov}(f(\hat{x}))}}{\sqrt{\mathrm{cov}(f(\hat{x}_{\mathrm{ML}}))}} \tag{27}$$

of an aspect f indicates the quality of the estimation algorithm for a particular map and aspect. It can be computed from $C := \mathrm{cov}(\hat{x})$, $C_{\mathrm{ML}} := \mathrm{cov}(\hat{x}_{\mathrm{ML}})$ and g the gradient of f as

$$\mathrm{ruc}(f) = \frac{\sqrt{\mathrm{cov}(f(\hat{x}))}}{\sqrt{\mathrm{cov}(f(\hat{x}_{\mathrm{ML}}))}} = \sqrt{\frac{\mathrm{cov}(f(\hat{x}))}{\mathrm{cov}(f(\hat{x}_{\mathrm{ML}}))}} \approx \sqrt{\frac{g^T C g}{g^T C_{\mathrm{ML}} g}}. \tag{28}$$

The last equation is the usual first order approximation for propagating covariances through functions. The term "any aspect of the map" in (R1) formally stands for "any function f". By virtue of (28) this can be replaced by a conceptually much more convenient expression involving "any vector g". In order to systematically characterize the values for different choices of g, the so-called generalized eigenvalues λ_i and eigenvectors v_i defined by

$$Cv_i = \lambda_i C_{\mathrm{ML}} v_i \tag{29}$$

are useful. They correspond to independent directions both with respect to C and C_{ML}. Their properties are

$$v_i^T C v_i = \lambda_i \ \forall i, \quad v_i^T C_{\mathrm{ML}} v_i = 1 \ \forall i, \tag{30}$$
$$v_i^T C v_j = 0 \ \forall i \neq j \quad \text{and} \quad v_i^T C_{\mathrm{ML}} v_j = 0 \ \forall i \neq j. \tag{31}$$

If g happens to be the i-th eigenvector v_i, the relative error $\mathrm{ruc}(g)$ in this aspect is the square root of the i-th eigenvalue

$$\mathrm{ruc}(g) = \mathrm{ruc}(v_i) = \sqrt{\frac{v_i^T C v_i}{v_i^T C_{\mathrm{ML}} v_i}} = \sqrt{\frac{\lambda_i}{1}} = \sqrt{\lambda_i}. \tag{32}$$

For an arbitrary aspect g, $\mathrm{ruc}(g)$ is the square root of a convex combination of the different eigenvalues. The weights of the convex combination are just the squares of the coefficients μ_i used in expressing g as a linear combination $g = \sum_i \mu_i v_i$ of the eigenvectors

$$\mathrm{ruc}(g) = \mathrm{ruc}\left(\sum_i \mu_i v_i\right) = \sqrt{\frac{\sum_i \mu_i^2 (v_i^T C v_i)}{\sum_i \mu_i^2 (v_i^T C_{\mathrm{ML}} v_i)}} = \sqrt{\frac{\sum_i \mu_i^2 \lambda_i}{\sum_i \mu_i^2}}. \tag{33}$$

So the generalized eigenvalues of C and C_{ML} characterize the relative error compared to the optimal maximum likelihood solution. Each eigenvalue corresponds to an independent aspect of the map in which the relative error is just the square root of the corresponding eigenvalue. In particular the square root of the largest eigenvalue bounds the maximum relative error in any aspect of the map. So in order to meet requirement (R1) formally, the largest eigenvalue must be a small constant $O(1)$. Analytically bounding these eigenvalues appears to be extremely hard but for a specific map they can be determined by Monte Carlo simulations (Frese, 2004). The eigenvalue spectrum allows a much more thorough assessment of an algorithms estimation quality than the usually used absolute (or rms) errors in the estimated landmarks' positions. Absolute errors are dominated by the largest error component, namely the "uncertainty of positions". So an estimation algorithm could significantly increase errors in some precisely known relations. This would not affect the landmarks absolute errors much but be clearly visible in the eigenvalue spectrum.

14. State of the Art

In this section a brief overview of the current state-of-the-art is given. The different SLAM algorithms established in the literature are compiled and analyzed with respect to the requirements mentioned above. The discussion focuses on the core estimation algorithm, a broader survey is given by Thrun (2002) and in a recent textbook by Thrun et al. (2005).

Algorithms

The evolution of SLAM algorithms can be divided into three phases:

In the first phase from the mid-eighties to the early nineties, the mathematical formulation of SLAM was still open and the special uncertainty structure discussed above was not yet fully recognized.

First approaches to building a map were based on so-called *evidence grids* introduced by Moravec and Elfes (1985). They divide the map into a regular grid with square cells of fixed size (typically $\approx$ 5cm). Each cell stores a real number $[0 \dots 1]$ representing the accumulated evidence of this cell containing an obstacle. Evidence grids are well suited to integrating the noisy low resolution information provided by ultrasonic sensors. However, they cannot represent robot pose uncertainty and thus are unable to perform SLAM.

Other authors followed a feature based approach as proposed by Brooks (1985). They represent the map as a graph of uncertain metrical relations between objects (Chatila and Laumond, 1985; Cheeseman and Smith, 1986; Durrant-Whyte, 1988; Faugeras, 1989). These approaches can incorporate uncertainty in the robot pose and led to an estimation theoretic formulation of SLAM.

The second phase of SLAM development was initiated by the influential paper of Smith, Self, and Cheeseman (1988) who first formulated SLAM systematically as a probabilistic estimation problem. They realized that landmark estimates are highly correlated because of the accumulated error in the robot pose and proposed representing all landmark positions and the robot pose in a joined state vector in combination with a full covariance matrix. This representation is called a *stochastic map* and is basically an EKF.

The stochastic map has been widely used and extended by several authors (Tardós, 1992; Castellanos et al., 1999; Hebert et al., 1995; Newman, 1999). Later, Durrant-Whyte et al. introduced the name *simultaneous localization and mapping* (1995). Surprisingly at that point the field wasn't aware of the original results by Gauss (1821) and later work in the field of surveying and photogrammetry (Triggs et al., 2000, for a modern overview) which are much older than the Kalman Filter (Kalman, 1960). In contrast to the EKF, most of these approaches are based on an information matrix representation partly exploiting the information matrix' sparsity. So for over a decade the main problem of large computation time remained. The most time consuming part of the computation is namely to update the covariance matrix, taking $O(n^2)$ time for n landmarks. This limited the use of SLAM to small environments ($n \lessapprox 100$ landmarks).

Recently, interest in SLAM has increased drastically and several, more efficient algorithms have been developed. In contrast to the EKF based approaches, most of these algorithms are efficient enough to be used in medium sized environments ($n \approx 500$ landmarks). Some very fast approaches can even be used for large environments ($n \gtrapprox 10000$ landmarks), but for these algorithms there are some limitations regarding the quality of the estimated map in certain situations.

Most approaches exploit the fact that the field of view of the involved sensors is limited. Thus, at any point in the environments, only few landmarks in the vicinity of the robot are observable and can be involved in measurements. The number k of these landmarks influences the computation time of the algorithm. It depends on the sensor and the density of landmarks but does not grow when the map gets larger. So it is small, practically $k \approx 10$, and often considered constant $k = O(1)$.

Guivant and Nebot (2001, 2003) developed a modification of the EKF called *Compressed EKF* (*CEKF*) that allows the accumulation of measurements in a local region with k landmarks at cost $O(k^2)$ independent from the overall map size n. When the robot leaves this region, the accumulated result must be propagated to the full EKF (*global update*) at cost $O(kn^2)$. Landmarks are grouped into *constellations* and their coordinates expressed relative to some reference landmark for each constellation. Thereby correlations between landmarks of distant constellations become negligible and an approximate global update can be performed in $O(kn^{\frac{3}{2}})$ with $O(n^{\frac{3}{2}})$ storage space needed.

Duckett et al. (2002) employ an iterative equation solver called *relaxation* to the linear equation system appearing in maximum likelihood estimation. The idea is to find the optimum for a chosen robot pose (or landmark) keeping all other landmarks and robot poses fixed. They apply one iteration updating each robot pose (and landmark) after each measurement with computation time $O(kn)$ and $O(kn)$ storage space. After closing a loop, more iterations are necessary leading to $O(kn^2)$ computation time in the worst case. This problem was recently solved by Frese and Duckett (2004) by a method called *Multilevel Relaxation*

(MLR). They employ a multilevel approach similar to the multigrid methods used in the numerical solution of partial differential equations. So computation time could be reduced to $O(kn)$ even when closing large loops.

Montemerlo et al. (2002) derived an algorithm called *fastSLAM* from the observation that the landmark estimates are conditionally independent given the robot pose. Basically, the algorithm is a particle filter in which every particle represents a sampled robot trajectory and associated Gaussian distributions of the different landmark positions. The conditional distribution of the different landmarks given the robot poses are independent, so n small covariance matrices suffice instead of one large matrix. The computation time for integrating a measurement is $O(M \log n)$ for M particles with $O(Mn)$ storage space needed. Of the discussed algorithms it is the only one that can handle uncertain data association (Nieto et al., 2003), which is an important advantage. The efficiency crucially depends on M being not too large. On the other hand a large number of particles is needed to close a loop with large error: A particle filter integrates measurements by choosing a subset of particles that is compatible with the measurements from the set of already existing particles (re-sampling) neither modifying the robot trajectory represented by the particles chosen, nor back-propagating the error along the loop. So at least one particle of the set must already close that loop by chance, and either many particles are needed or there will be gaps in a loop already closed.

Later Eliazar and Parr (2003), Hähnel et al. (2003) as well as Grisetti et al. (2005) extended the framework to using plain evidence grids as particles. The first appearance of this idea can be traced back to Murphy (1999). This way maps can be constructed in difficult situations without any sensor preprocessing such as landmark extraction or scan matching.

Thrun et al. (2002) presented a constant time algorithm called Sparse Extended Information Filter (SEIF). They followed a similar idea, also proposed by Frese and Hirzinger (2001) and use an information matrix instead of a covariance matrix to represent uncertainty. The algorithm exploits the fact that the information matrix is approximately sparse requiring $O(kn)$ storage space as shown in §11. The representation allows integration of a measurement in $O(k^2)$ computation time, but to give an estimate a system of n linear equations must be solved. Similar to the approach of Duckett et al. this is done by relaxation. Thrun et al. propose not to relax all n landmarks, but only $O(k)$, thereby formally obtaining an $O(k^2)$ algorithm. This technique is called *amortization* since the computation time for solving the equation system is spread over several update steps. However, such an amortized algorithm no longer complies with (R1) because after closing the loop the map's error will be much larger than in the optimal estimate for a long time. In the numerical literature relaxation is reputed to need $O(n)$ iterations with $O(kn^2)$ time for reducing the equation error by a constant factor (Press et al., 1992, §19.5). SEIF will accordingly need about $O(kn^2/k^2) = O(n^2/k)$ update steps or $O(n/k)$ passes along the loop until the effect of closing the loop is mostly incorporated into the estimate.

It is interesting to note, that nevertheless even such an aggressive amortization strategy does not slow down asymptotic convergence of the map estimate. When passing through the environment t times, i.e. repeating the same sequence of measurements t times, the error of any relation in the map will decrease $O(t^{-\frac{1}{2}})$. The calculation above shows that the equation error will decrease $e^{O(\frac{kt}{n})}$, since it decreases by a constant factor every $O(n/k)$ passes. Since the latter is asymptotically faster, in the limit $t \to \infty$ the stochastic error will be dominant.

The SEIF algorithm can be modified (SEIF w. full update) to update all landmarks after each measurement and perform $O(n)$ iterations after closing a loop. Then the algorithm needs $O(kn)$ computation time per measurement complying with (R1)-(R3) with the exception of closing a loop, where $O(kn^2)$ computation time is needed.

Paskin (2003) phrases the SLAM problem as a Gaussian graphical model in his Thin Junction Tree Filter (TJTF). It maintains a tree where the overall posterior distribution is represented as the product of a low-dimensional Gaussian at each node. With this representation an update can be performed in $O(k^3n)$ by passing marginalized distributions along the edges of the tree. When a node involves too many landmarks, TJTF sparsifies, i.e. further approximates the represented distribution to maintain efficiency.

Frese (2004) has independently proposed a similar algorithm based on a so-called tree map that works by hierarchically dividing the map into local regions and subregions. At each level of the hierarchy each region stores a matrix representing the landmarks at the region's border. Thereby it effectively decomposes the large sparsely approximated information matrix into a sum of small matrices exploiting the special topology encountered in typical buildings. In a similar manner to CEKF a measurement is integrated into a local subregion using $O(k^2)$ computation time. The global update necessary when moving to a different subregion requires only $O(k^3 \log n)$ computation time by virtue of the hierarchical decomposition. Computing an estimate for the whole map takes $O(kn)$ time (tree map w. global map). Furthermore

Table I. Performance of different SLAM algorithms with n landmarks, m measurements, p robot poses and k landmarks local to the robot: *Maximum Likelihood (ML), Extended Kalman Filter (EKF), Compressed Extended Kalman Filter (CEKF), Relaxation, Multi-Level Relaxation (MLR), FastSLAM, Sparse Extended Information Filter (SEIF), Thin Junction Tree Filter (TJTF), Tree Map.* The tree map algorithm assumes a topologically suitable building. FastSLAM is a particle filter approach (M particles). SEIF/full upd. denotes the SEIF algorithm but updating all landmarks after each measurement. UDA stands for 'Uncertain Data Association' meaning that the algorithm can handle landmarks with uncertain identity.

		(R1)		**(R2)**		**(R3)**	
	UDA	non-linear	loop quality	memory	update	global update	loop
ML		√	√	m		.... $(n+p)^3$	
EKF			√	n^2		.... n^2	
CEKF			√	$n^{\frac{3}{2}}$	k^2	.... $kn^{\frac{3}{2}}$	
Relaxation		√	√	kn	.... kn		kn^2
MLR		√	√	kn		.... kn	
FastSLAM	√	√	see §14	Mn		.... $M \log n$	
SEIF				kn		.... k^2	
w. full update			√	kn	.... kn		kn^2
TJTF		√	√	$k^2 n$	k^3	..... $k^3 n$	
Tree map		√	√	kn	k^2	... $k^3 \log n$ ...	
w. global map		√	√	kn		.... kn	

it solves linearization problems by "nonlinear rotation" of individual regions (see also §9).

Comparison

Table I shows an overview of the performance of the algorithms discussed. It can be seen that only multi-level relaxation and the tree map based algorithm strictly fulfill all three requirements.

Requirement (R1) is completely fulfilled by ML, single or multi-level relaxation, TJTF, the tree map algorithm, and also additionally by EKF, CEKF, and SEIF w. full update, if the orientation error is small enough to allow linearization. When closing a loop SEIF and fastSLAM do not fulfill (R1) due to the problems mentioned before.

Requirement (R2) is met by relaxation, TJTF, MLR, the tree map algorithm and fastSLAM (for $M = O(1)$) and SEIF.

Requirement (R3) is fulfilled by fastSLAM and SEIF, giving an estimation that does not always meet (R1). Relaxation, SEIF w. full update, and CEKF come very close to meeting (R3). Relaxation and SEIF w. full update need linear computation time except when closing a large loop, while CEKF only needs $O(n^{\frac{3}{2}})$ even when closing loops. The additional advantage of CEKF is that this computation is not performed after each measurement, but only when the robot leaves a local area of the map. Multi-level relaxation, TJTF, and, the tree map algorithm fulfill (R3) completely.

What is the reason for the large progress in performance of recently developed algorithms?

Least squares estimation and incremental least squares estimation in general lead to linear equations systems, which is an established and thoroughly studied area of numerical mathematics. So it is very unlikely that a general solution that is faster than EKF with $O(n^2)$ will be found. From the author's perspective the key point is to identify a property distinguishing SLAM from a general estimation problem. Indeed, all faster approaches exploit such a property: relaxation, multi-level relaxation, and SEIF exploit sparsity, fastSLAM exploits a special factorization of the involved probability distribution and CEKF exploits some property of the correlation of distant landmarks. TJTF and the tree map based algorithm exploit the fact that typical buildings can be recursively divided into two parts, with very few landmarks of one part being observable from the other part. This property is stronger than general sparsity allowing extremely efficient updates $(O(k^3 \log n))$ but it is also more restrictive being, for instance, not met in general cross country navigation.

15. Data Association

The task of data association or landmark identification is to recognize a detected landmark as a landmark already represented in the map. In other words, landmark observations are matched with landmarks in the map including the decision to define unmatched landmarks as new. In general, *not associating* a landmark observation is a harmless error only resulting in a duplicate landmark being introduced in the map. This may possibly lead to some problems when using the map but does not affect the map in general. In contrast *false association* of a landmark, i.e., confusing it with another landmark, usually ruins the map completely since wrong information such as two different places being the same is integrated.

Up to now the discussion has assumed that all observed landmarks have been identified before the measurements are passed to the SLAM algorithm. However this task is far from being trivial.

Three types of information can be used by a data association algorithm: appearance of landmarks, layout of a group of landmarks, and bounds on the error in robot pose accumulated after the last observation of a landmark. It is obvious that the difficulty of data association strongly depends on landmarks, sensors, and environment. In order of increasing capability possible approaches are *nearest-neighbor association*, matching a set of landmark observations simultaneously using *Mahalanobis distance*, matching a local *map patch*, *multi-hypothesis tracking* and *lazy data-association*.

In some settings the distance between confusable landmarks is larger than the robot's pose uncertainty accumulated after last observation of the landmarks (Castellanos et al., 1999; Guivant and Nebot, 2001). Then each landmark can be matched to the nearest neighbor according to observation and current robot pose estimate.

A more complex approach is needed when there are several possible matching candidates for each individual observation in the map. In this case a whole set of observations is matched simultaneously by iterating through all matching combinations that are compatible with the bounds on the accumulated error in the robot's pose. The plausibility of a possible match can be evaluated by summing up the squared distance between matched observation and landmark to which the observation is matched. A much better approach is to use Mahalanobis distance based on the covariance matrix of the considered landmarks provided by the SLAM algorithm (Neira and Tardós, 2000). Unlike simple squared distance, Mahalanobis distance takes into account that some relations between the landmarks are much more precisely known than other relations so these relations affect likelihood of a match much more.

Again, sensor noise, map size, and arrangement of landmarks decide whether observations from a single robot pose can be uniquely matched to the map. If not, as an alternative a local map patch around the robot can be matched with the remaining map (Gutmann and Konolige, 1999; Frese, 2004).

In even more difficult situations it might be impossible to determine a landmark's identity at the moment it is observed. Then algorithms are needed that can defer the decision about association of that landmarks until further observations provide enough evidence. One approach is multi-hypothesis tracking generating several copies of the uncertain map whenever the identity of a landmark is ambiguous. This can be done in a particle filter framework (Montemerlo and Thrun, 2003). Both the discrete uncertainty of landmark identification and the continuous uncertainty generated by measurement noise are represented by the set of particles, i.e. samples drawn from the resulting multi-modal distribution. So each particle corresponds to a map without uncertainty. Alternatively one could use a mixture-of-Gaussian model where each Gaussian represents a topological hypothesis, i.e. a map with fixed data association but still retaining the continuous uncertainty generated by measurement noise.

If the map contains several independent ambiguities, a particle filter would need an exponentially increasing number of particles to cover all possible combinations. Then lazy data-association is needed, i.e. the algorithm must be able to revise a past data association decision when new evidence suggests that it has been wrong. Hähnel et al. (2003) have realized this idea by incrementally building and pruning the data-association decision-tree. The tree has one level for each landmark observation with the different children of each node corresponding to different decisions on the landmark's identity. The likelihood of the data association corresponding to a node is evaluated using LLS and is monotonically decreasing along the tree. Thus whenever a node is less likely than the current most likely leaf, the node can be pruned and need not be expanded. The algorithm always finds the most likely solution which is an advantage but also a disadvantage since in the worst case the tree can grow exponentially. Ranganathan and Dellaert (2004) avoid the computational limitations of finding the most likely solution by employing a Markov Chain Monte Carlo (MCMC) framework (Neal, 1993) for searching through the space of possible data associations, i.e. topologies. Duckett (2003) searches through the space of possible robot trajectories using a genetic algorithm. It works on a population of trajectories which are modified using mutation and cross-over. Trajectories are selected according to how consistently the observed sensor data can be integrated into a map assuming the robot poses defined by the trajectory.

To conclude the discussion: Within a least square SLAM algorithm determining the landmarks identity and estimating a map based on this data association are two relatively independent subtasks. This is due to the fact that the underlying Gaussian distributions are unimodal, i.e., they cannot represent the discrete uncertainty of not knowing which of the landmarks in the map corresponds to the observation. Thus there are two approaches. Either data association must be determined before integrating a measurement. Uncertainty information from the SLAM algorithm can greatly facilitate this task (Mahalanobis distance). Or least square SLAM must be used as a core engine within a larger framework. Then the framework generates different hypotheses by tentatively integrating a measurement. The likelihood of each hypothesis is in turn evaluated using the core SLAM algorithm.

16. Summary

Three exceptional features distinguish SLAM from many other estimation problems:

1. *Error accumulation*:
   When moving through an unknown area the error of the robot pose and, consequently, the global error of landmarks nearby can grow arbitrarily high. Nevertheless, relative properties of these landmarks such as distances or angles are known much more precisely with an uncertainty independent from the overall uncertainty of the robot pose. This gives rise to a highly specific uncertainty structure, called the *Certainty of Relations despite Uncertainty of Positions.* Relations between nearby landmarks are known precisely although the absolute position of the landmarks is highly uncertain. On a global scale, uncertainty is mainly a composition of local orientation uncertainties along the path traveled. The error effects an "uncertain bending" of the map, showing a simple geometrically determined structure despite its magnitude. Conversely, on a local scale the uncertainty is much smaller and more complex.

2. *High dimensionality*:
   After each measurement a SLAM algorithm has to estimate the robot pose (3 DOF) and the whole map ($2n$ DOF for n landmarks). So the overall dimension $3 + 2n$ of the estimation problem is very large (> 500) and continues to grow. Therefore, common estimation algorithms are not efficient enough for SLAM. However there has been an enormous progress in recently published algorithms which now allows both linear update time and storage space.

   A theorem has been sketched that assures that the information matrix resulting from marginalization of old robot poses is approximately sparse, a property exploited by several efficient algorithms. The theorem further theoretically substantiates the analysis of the global uncertainty described above.

3. *Nonlinearity*:

   It is common for estimation problems to have nonlinear measurement equations. For the SLAM estimation problem, the equations are nonlinear in the robot's orientation, ane the estimation error of the robot's orientation can grow unboundedly. So for a sufficiently large map linearization is not suitable. This is a particular problem of SLAM not often encountered in other estimation tasks. A critical consequence is the distortion of distances between landmarks even though the distances are well known from the measurements.

Due to nonlinearities in the measurement equations, the Jacobians of the equations are non-constant. However, when transforming them into robot coordinates by factoring out rotation by robot orientation, they are nearly constant. This shows that mainly non-linearity of orientation is of relevance.

SLAM evolved over a decade of research and has reached a level of by now that it can be used in medium or even large environments. Undoubtedly, in the future the focus will shift towards applying SLAM as a component in larger systems and handling the challenges of uncertain data association, very large, dynamical, and outdoor environments. This poses new problems, both concerning the core algorithm and possible applications.

References

Brooks, R.: 1985, 'Visual map making for a mobile robot'. In: *Proceedings of the IEEE International Conference on Robotics and Automation, St. Louis*. pp. 824 – 829.

Castellanos, J., J. Montiel, J. Neira, and J. Tardós: 1999, 'The SPmap: A Probablistic Framework for Simultaneous Localization and Map Building'. *IEEE Transactions on Robotics and Automation* **15**(5), 948 – 952.

Chatila, R. and J. Laumond: 1985, 'Position referencing and consistent world modeling for mobile robots'. In: *Proceedings of the IEEE International Conference on Robotics and Automation, St. Louis*. pp. 138 – 145.

Cheeseman, R. and P. Smith: 1986, 'On the representation and estimation of spatial uncertainty'. *International Journal of Robotics* **5**, 56 – 68.

Cox, I. J. and G. T. Wilfong (eds.): 1990, *Autonomous Robot Vehicles*. Springer Verlag, New York.

Duckett, T.: 2003, 'A Genetic Algorithm for Simultaneous Localization and Mapping'. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'2003)*. Taipei, pp. 434–439.

Duckett, T., S. Marsland, and J. Shapiro: 2002, 'Fast, On-line Learning of Globally Consistent Maps'. *Autonomous Robots* **12**(3), 287 – 300.

Durrant-Whyte, H.: 1988, 'Uncertain geometry in robotics'. *IEEE Transactions on Robotics and Automation* **4**(1), 23 – 31.

Durrant-Whyte, H., D. Rye, and E. Nebot: 1995, 'Localization of Autonomous Guided Vehicles'. In: G. Hirzinger and G. Giralt (eds.): *Proceedings of the 8th International Symposium on Robotics Research*. pp. 613 – 625, Springer Verlag, New York.

Eliazar, A. and R. Parr: 2003, 'DP-SLAM: Fast, Robust Simulataneous Localization and Mapping without Predetermined Landmarks'. In: *Proceedings of the International Joint Conference on Artificial Intelligence, Acapulco*. pp. 1135–1142.

Faugeras, O.: 1989, 'Maintaining representations of the environment of a mobile robot'. *IEEE Transactions on Robotics and Automation* **5**(6), 804–819.

Frese, U.: 2004, 'An $O(\log n)$ Algorithm for Simulateneous Localization and Mapping of Mobile Robots in Indoor Environments'. Ph.D. thesis, University of Erlangen-Nürnberg. `http://www.opus.ub.uni-erlangen.de/opus/volltexte/2004/70/`.

Frese, U.: 2005a, 'Animation illustrating SLAM uncertainty structure.'. `http://www.informatik.uni-bremen.de/~ufrese/slamdiscussion_e.html`.

Frese, U.: 2005b, 'A Proof for the Approximate Sparsity of SLAM Information Matrices'. In: *Proceedings of the IEEE International Conference on Robotics and Automation, Barcelona*. pp. 331–337.

Frese, U. and G. Hirzinger: 2001, 'Simultaneous Localization and Mapping - A Discussion'. In: *Proceedings of the IJCAI Workshop on Reasoning with Uncertainty in Robotics, Seattle*. pp. 17 – 26.

Frese, U., P. Larsson, and T. Duckett: 2004, 'A Multigrid Algorithm for Simultaneous Localization and Mapping'. *IEEE Transactions on Robotics* **21**(2), 1–12.

Gauss, C.: 1821, 'Theoria combinationis observationum erroribus minimis obnoxiae'. *Commentationes societatis regiae scientiarum Gottingensis recentiores* **5**, 6–93.

Gelb, A. (ed.): 1974, *Applied Optimal Estimation*. MIT Press, Cambridge.

Grisetti, G., C. Stachniss, and W. Burgard: 2005, 'Improving Grid-based SLAM with Rao-Blackwellized Particle Filters by Adaptive Proposals and Selective Resampling'. In: *Proceedings of the IEEE International Conference on Robotics and Automation, Barcelona*. pp. 667–672.

Guivant, J. and E. Nebot: 2001, 'Optimization of the Simultaneous Localization and Map-Building Algorithm for Real-Time Implementation'. *IEEE Transactions on Robotics and Automation* **17**(3), 242 – 257.

Guivant, J. and E. Nebot: 2003, 'Solving computational and memory requirements of feature-based simultaneous localization and mapping algorithms'. *IEEE Transactions on Robotics and Automation* **19**(4), 749–755.

Gutmann, J. and K. Konolige: 1999, 'Incremental mapping of large cyclic environments'. In: *Proceedings of the IEEE International Symposium on Computational Intelligence in Robotics and Automation, Monterey*. pp. 318–325.

Hähnel, D., W. Burgard, D. Fox, and S. Thrun: 2003, 'An Efficient FastSLAM Algorithm for Generating Maps of Large-Scale Cyclic Environments from Raw Laser Range Measurements'. In: *Proceedings of the International Conference on Intelligent Robots and Systems, Las Vegas*. pp. 206–211.

Hähnel, D., W. Burgard, B. Wegbreit, and S. Thrun: 2003, 'Towards lazy data association in SLAM'. In: *In Proceedings of the 10th International Symposium of Robotics Research*.

Hebert, P., S. Betge-Brezetz, and R. Chatila: 1995, 'Probabilistic Map Learning, Necessity and Difficulty'. In: *Proceedings of the International Workshop Reasoning with Uncertainty in Robotics*. pp. 307 – 320.

Julier, S. J. and J. K. Uhlmann: 2001, 'A counter examplee to the theory of simultaneous localization and map building'. In: *Proceedings of the IEEE International Conference on Robotics and Automation*. pp. 4238–4243.

Kalman, R.: 1960, 'A New Approach to Linear Filtering and Prediction Problems'. *Transaction of the ASME Journal of Basic Engineering* pp. 35–45.

Kelly, A.: 2000, 'Some Useful Results for Closed-Form Propagation of Error in Vehicle Odometry'. Technical Report CMU-RI-TR-00-20, Carnegie Mellon University.

Leonard, J. and H. Durrant-Whyte: 1992, 'Dynamic Map Building for an autonomous mobile robot'. *The International Journal on Robotics Research* **11**(4), 286 – 298.

Lu, F. and E. Milios: 1997, 'Globally Consistent Range Scan Alignment for Environment Mapping'. *Autonomous Robots* **4**, 333 – 349.

Montemerlo, M. and S. Thrun: 2003, 'Simultaneous Localization and Mapping with Unknown Data Association using FastSLAM'. In: *Proceedings of the International Conference on Robotics and Automation, Taipei*. pp. 1985–1991.

Montemerlo, M., S. Thrun, D. Koller, and B. Wegbreit: 2002, 'FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping Problem'. In: *Proceedings of the AAAI National Conference on Artificial Intelligence, Edmonton*. pp. 593–598.

Moravec, H. and A. Elfes: 1985, 'High resolution maps from wide angle sonar'. In: *Proceedings of the IEEE International Conference Robotics and Automation, St. Louis*. pp. 116 – 121.

Murphy, K.: 1999, 'Bayesian Map Learning in Dynamic Environments'. In: *Advances in Neural Information Processing Systems (NIPS), Denver*, Vol. 12. pp. 1015–1021.

Neal, R.: 1993, 'Probabilistic Inference Using Markov Chain Monte Carlo Methods'. Technical Report CRG-TR-93-1, Department of Computer Science, University of Toronto.

Neira, J. and J. Tardós: 2000, 'Robust and feasible Data Asssociation for Simultaneous Localization and Map Building'. In: *ICRA Workshop SLAM, San Francisco*.

Newman, P.: 1999, 'On the structure and Solution of the Simultaneous Localisation and Map Building Problem'. Ph.D. thesis, Deptartment of Mechanical and Mechatronic Engineering, Sydney.

Nieto, J., J. Guivant, E. Nebot, and S. Thrun: 2003, 'Real Time Data Association for fastSLAM'. In: *Proceedings of the IEEE Conference on Robotics and Autonomation, Taipeh*. pp. 412–418.

Paskin, M.: 2003, 'Thin Junction Tree Filters for Simultaneous Localization and Mapping'. In: *Proceedings of the 18th International Joint Conference on Artificial Intelligence*. San Francisco, pp. 1157–1164.

Press, W., S. Teukolsky, W. Vetterling, and B. Flannery: 1992, *Numerical Recipes, Second Edition*. Cambridge University Press, Cambridge.

Ranganathan, A. and F. Dellaert: 2004, 'Inference In The Space Of Topological Maps: An MCMC-based Approach'. In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Sendai*. pp. 2518–2523.

Smith, R., M. Self, and P. Cheeseman: 1988, 'Estimating Uncertain Spatial Relationships in Robotics'. In: I. Cox and G. Wilfong (eds.): *Autonomous Robot Vehicles*. Springer Verlag, New York, pp. 167 – 193.

Tardós, J.: 1992, 'Representing partial and uncertain sensorial information using the theory of symmetries'. In: *Proceedings of the IEEE International Conference on Robotics and Automation, Nice*. pp. 1799 – 1804.

Thrun, S.: 2002, 'Robotic Mapping: A Survey'. In: G. Lakemeyer and B. Nebel (eds.): *Exploring Artificial Intelligence in the New Millenium*. Morgan Kaufmann. §1.

Thrun, S., W. Burgard, and D. Fox: 1998, 'A Probabilistic Approach to Concurrent Mapping and Localization for Mobile Robot'. *Machine Learning* **31**(5), 1 – 25.

Thrun, S., W. Burgard, and D. Fox: 2005, *Probabilistic Robotics*. MIT Press.

Thrun, S., D. Koller, Z. Ghahramani, H. Durrant-Whyte, and N. A.Y.: 2002, 'Simultaneous Mapping and Localization With Sparse Extended Information Filters: Theory and Initial Results'. In: *Proceedings of the Fifth International Workshop on Algorithmic Foundations of Robotics, Nice*.

Triggs, W., P. McLauchlan, R. Hartley, and A. Fitzgibbon: 2000, 'Bundle Adjustment – A Modern Synthesis'. In: W. Triggs, A. Zisserman, and R. Szeliski (eds.): *Vision Algorithms: Theory and Practice*, LNCS. Springer Verlag, pp. 298–375.

Uhlmann, J., S. Julier, and M. Csorba: 1997, 'Nondivergent simultaneous map building and localization using covariance intersection'. In: *Proceedings of the SPIE Conference on Navigation and Control Technologies for Unmanned Systems II*, Vol. 3087. pp. 2 – 11.

Walter, M., R. Eustice, and J. Leonard: 2005, 'A Provably Consistent Method for Imposing Exact Sparsity in Feature-based SLAM Information Filters'. In: *Proceedings of the 12th International Symposium of Robotics Research*.

Author's Vitae

Udo Frese was born in Minden, Germany in 1972. He received the Diploma degree in computer science from the University of Paderborn in 1997. From 1998 to 2003 he was a Ph.D. student at the German Aerospace Center in Oberpfaffenhofen. In 2004 he joined University of Bremen. His research interests are mobile robotic, simultaneous localization and mapping and computer vision.