2008 American Control Conference
Westin Seattle Hotel, Seattle, Washington, USA
June 11-13, 2008

A Control Theoretic Formulation of the Generalized SLAM Problem in Robotics

Magnus Egerstedt and Patric Jensfelt

***Abstract*— Simultaneous Localization and Mapping (SLAM) has emerged as a key capability for autonomous mobile robots navigating in unknown environments. The basic idea behind SLAM is to concurrently obtain a map of the environment and an estimate of where the robot is placed within this map. In other words, the map and the robot's pose have to be estimated at the same time, given the same data set. This paper revisits this problem from a control theoretic vantage point by reformulating the SLAM problem as a problem of simultaneously estimating the state and the output map of a controlled, dynamical system. What is different with this formulation is that the map is contained in the output map and not, as previously done, in the state of the system.**

I. Introduction

SLAM, or Simultaneous Localization and Mapping, is the process of concurrently building a map of the environment and estimating the pose of the robot in this environment. This problem has been referred to as the "holy grail" in mobile robotics [1] and as such has received considerable attention. For an overview of the history and basics of SLAM, as well as the state-of-the-art, see e.g. [2], [3].

Arguably, the progress made in the field of SLAM has lead to a point where the problem in practical terms can be considered solved in small enough environments, and with range-based sensors. As a result, as the field has matured, some of the attention has shifted towards a more theoretical analysis of the basic properties of the SLAM problem itself, and it is in this context that this paper is to be understood. Most of the previous analysis of the SLAM problem has been devoted to the EKF formulation and to linear examples. For instance, convergence properties for a linear SLAM formulation was presented in [1]. Several examples of consistency analysis ARE available [4], [5], [6], and in [7] the observability and controllability of SLAM is discussed for certain classes of systems, such as switched linear systems.

This paper continuous along this latter trend of papers and tries to take a step back and approach the SLAM problems from a slightly more abstract perspective. In fact, we will argue that SLAM should be understood as a problem involving the simultaneous estimation of the output map and state of a dynamic system rather than a (sometimes unnatural) incorporation of the map into the state of the system. This formulation moreover lends itself to be directly extended to the so-called SPLAM [8] problem (Simultaneous Planning, Localization, and Mapping) in that the control input becomes an integral part of the problem formulation.

This work was partially sponsored by the SSF through its Centre for Autonomous Systems (CAS) and the EU as part of the project CoSy IST-2004-004450. The support is gratefully acknowledged.

M. Egerstedt is with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA magnus@ece.gatech.edu

P. Jensfelt is with the Centre for Autonomous Systems, Royal Institute of Technology, 10044 Stockholm, Sweden patric@csc.kth.se

II. Simultaneous Localization and Mapping

A. Background

Loosely speaking, the SLAM problem has traditionally been formulated as finding the joint probability distribution $P(x_k, \mathcal{M} \mid u_{[0,k]}, y_{[0,k]}, x_0)$, where x_k is the pose of the robot at time k, $\mathcal{M}$ is a (static) map of the environment, $u_{[0,k]}$ and $y_{[0,k]}$ are the control and measurement history respectively, and x_0 is an initial estimate of the pose.

One of the first key insights into the SLAM problem was the understanding that in order to estimate the robot pose and the map, the correlation between these two individual components is very important and needs to be correctly maintained in order for the SLAM problem to be solvable. These correlations and the fact that the size of the problem often is large typically results in computationally demanding problems. In this light it is not surprising that a substantial amount of work has been carried out to tackle the complexity issue of SLAM (see e.g. CEKF [9], DSM [10], Atlas [11], treemap [12], SEIF [13], Graphical SLAM [14], M-Space [15]).

One of the central issues in solving the SLAM problem is that of representation, both in terms of the map and in terms of the probability distribution itself. In fact, the issue of representation is one of the main differences between the many proposed approached to SLAM, the two most common ones being Extended Kalman Filters (EKF-SLAM) [16] and Rao-Blackwellized particle filters (FastSLAM) [17].

In terms of map representations, the two most widely used representations use so-called *features* and *occupancy grids* [18]. In a feature-based representation, the environment is modelled as a set of geometric features such as points, lines, planes, etc, while an occupancy grid representation uses a discrete approximation of the world into grid cells, typically with a 2D assumption.

The view advocated in this paper is that the environment map should be thought of as an output mapping, which is somewhat in line with the ideas advocated in [19] in that the world is its own best representation, using raw or only slightly processed sensor data as the model. An example of this is to use laser scans acquired as the robot moves around in the environment, with each component in the map being given by a tuple $\{x_k, scan_k\}$, where $scan_k$ is the laser scan reading at time k. (These scans can in fact be seen as generalized features.)

In this paper, we will formalize the SLAM problem at a level of abstraction in which entities such as "robot positions", "maps", "landmarks", "particle filters", "Kalman filters", and "range sensors" are all delegated to particular instantiations of the SLAM problem. We will in fact do this by formulating a more general, control-theoretic problem, namely the SSAOME (Simultaneous State And Observation Map Estimation) problem but we acknowledge that this acronym will never catch on, so we will keep referring to it as the SLAM problem.

B. A Trivial Yet Illustrative Example

In order to distill away the particulars of robot-based map building and localization from the SLAM problem to arrive at a control-theoretic formulation, we start by considering a highly trivial scenario. (And, it should be pointed out already at this point that the approach taken in the next few paragraphs is certainly not the most effective.) Here a robot is to navigate a 2×2 grid, with cells $(1,1), (1,2), (2,1), (2,2)$. Moreover, each of the grid cells is colored with one of two different colors, namely white and gray. These colors are the measurements that the robot have access to, and we thus let the output set be given by $Y = \{\texttt{white},\texttt{gray}\}$.

The robot can move in four different directions, i.e. the input set is $U = \{\uparrow, \downarrow, \leftarrow, \rightarrow\}$ and we assume that, at the boundary of the grid, a move that would result in the robot leaving the grid simply leaves the robot in the cell from which the move originated. In fact, if we let the grid be given by $X = \{(1,1), (1,2), (2,1), (2,2)\}$ and let x_k be the cell that the robot is in at time k, the dynamics of the system can be defined through the transition function $f : X \times U \rightarrow X$.

Now, given that the robot is traversing through the grid, using a string of moves, $u_0, u_1, \ldots$, the result is that a string of colors is observed, $y_0, y_1, \ldots$. And, the SLAM problem is precisely that of figuring out how the different cells in the grid are colored at the same time as we should know where in the grid the robot is located.

As an example, assume that the input string is $u_0 = \uparrow$, $u_1 = \leftarrow, u_2 = \downarrow, u_3 = \rightarrow, u_4 = \uparrow$, with the corresponding output string being $y_0 = \texttt{white}$, $y_1 = \texttt{white}$, $y_2 = \texttt{white}$, $y_3 = \texttt{gray}$, $y_4 = \texttt{white}$, $y_5 = \texttt{white}$. Is it possible to solve the SLAM problem using this information? The answer is yes, and the solution is shown in Figure 1.

At time $k = 0$, the robot can be anywhere, but since $y_0 = \texttt{white}$, each hypothesis about the robot's location corresponds to a particular map with only one cell having a known color. (In the figure, the robot's location is denoted by $\star$, while a ? in the cell means that the color of that cell is unknown.) At time $k = 1$, there are still four possible colorings of the grid, but we now know that the robot is in one of two possible locations, namely (1,2) or (2,2). The localization part of the SLAM problem is in this case actually solved already at time $k = 2$, where we know that $x_2 = (1,2)$. However, there are still four possible grid colorings that are consistent with the input-output string. In

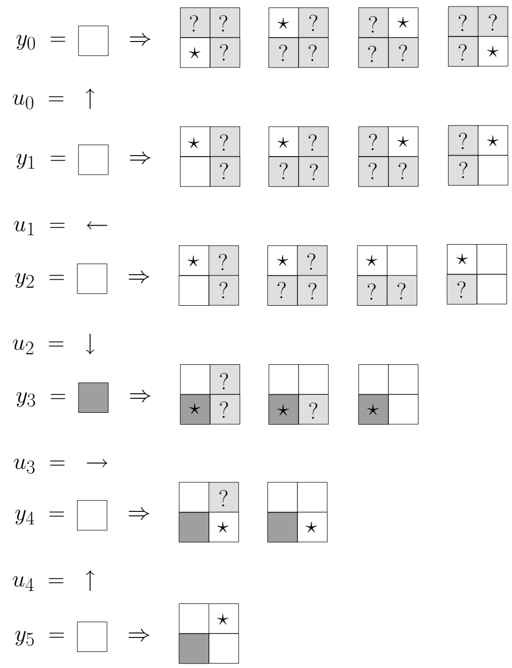

Fig. 1. A simple example problem is shown in which a robot negotiates a 2×2 grid, in which each cell is colored with one of two possible colors.

fact, using this particular input-output string, it takes until $k = 5$ until a unique coloring of the grid has been obtained, and the SLAM problem has been completely solved.

C. A Control-Theoretic Approach

The example described in the previous paragraphs is certainly not overly complicated. And the solution was not very hard to come by. But, what was actually the problem that we solved? If we associate an output map to each grid cell $h : X \rightarrow Y$ it is clear that knowing h is equivalent to knowing how the grid was colored. In other words, the SLAM problem in the previous paragraphs can be formulated as finding x and h from the input-output string.

In fact, it is our claim that this formulation captures the SLAM problem precisely. Given a general, dynamic system

$$x_{k+1} = f(x_k, u_k)$$
$$y_k = h(x_k),$$

together with input-output strings $u_0, u_1, \ldots$ and $y_0, y_1, \ldots$, find estimates $\hat{x}$ and $\hat{h}$ such that $\|x_k - \hat{x}_k\| \rightarrow 0$ and $\|h - \hat{h}_k\| \rightarrow 0$ as $t \rightarrow \infty$, in some appropriate (possibly functional) norms. The central claim to this paper is thus that *SLAM is a particular instantiation of the problem of simultaneously estimating the state and the output map of a dynamical system.*

Before we start actually solving some SLAM problems, a few words should be mentioned about what this problem formulation actually entails. It really states that knowing the output map is the same as knowing the map of the world. This formulation, innocent as it may look, is in fact the main contribution of this paper.

III. Linear SLAM: A Bilinear Estimation Problem

Consider the linear system

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k \\ y_k &= Cx_k, \end{aligned}$$

where $x \in \mathbb{R}^n,\ u \in \mathbb{R}^m,\ y \in \mathbb{R}^p$, and A, B, C are constant matrices of compatible dimensions. As the SLAM problem is precisely that of estimating C and x from input-output strings, we see that the output equation above is bilinear in these two terms. In other words, SLAM for linear systems inevitably involves solving an estimation problem for linear systems with bilinear output equations. And, as the output equation becomes nonlinear, it is clear that observability alone might not suffice to ensure that the SLAM problem has a unique solution, as will be seen in the following paragraphs.

A. Non-Uniqueness In the Autonomous Case

First, consider the situation in which there is no control term. Assume moreover that the output string $y_0, y_1, \ldots$ was observed and that we were able to find $\hat{x}_k,\ k = 0, 1, \ldots$ and $\hat{C}$ that perfectly reproduced this output string, i.e. $\hat{C}\hat{x}_0 = y_0,\ \hat{C}\hat{x}_1 = y_1, \ldots$. Does this mean that we have recovered x and C? The answer to this is no, as can be directly seen through the following construction:

Let

$$\hat{x}_k = \alpha x_k,\ \hat{C} = \frac{1}{\alpha} C$$

for any non-zero $\alpha \in \mathbb{R}$. With this choice, we have that $\hat{C}\hat{x}_k = Cx_k = y_k,\ \forall k \geq 0$ and as such we have reproduced the output string perfectly without having the correct state and output matrix estimates (as long as $C \neq 0$). If, in fact, $C = 0$ then by letting $\hat{C} = 0$ and $\hat{x}$ be arbitrary, the correct (all zeros) output string is also reproduced, and we have thus shown the following result:

Theorem 3.1: For linear, autonomous systems $x_{k+1} = Ax_k,\ y_k = Cx_k$, the SLAM problem can not be uniquely solved.

B. A Small Computation

So, in light of the previous result, one is tempted to abandon all hope of being able to solve the SLAM problem for linear systems. However, it will turn out that this is not necessarily the case if one ensures sufficient excitation of the system. This can, as we will see in the next paragraphs, be achieved by allowing feedback in the system. In other words, if we use the state estimate to drive the true system, the state estimate will affect the output, and in that way, uniqueness can be obtained.

We illustrate this fact informally by considering the following scalar system

$$\begin{aligned} x_{k+1} &= ax_k + u_k \\ y_k &= cx_k, \end{aligned}$$

where $x, u, y, a, b, c \in \mathbb{R}$. Now, let, as before, $\hat{c}$ and $\hat{x}_k,\ k = 0, 1, \ldots$ be state and output matrix estimates and assume that we use the estimate to define the control signal through the feedback law

$$u = -k\hat{x},$$

with the result that $\hat{x}_{k+1} = ax_k - bk\hat{x}_k$. Moreover, assume that the estimates are such that they reproduce the output string perfectly, i.e. $\hat{c}\hat{x}_0 = y_0,\ \hat{c}\hat{x}_1 = y_1, \ldots$. If we assume that $\hat{x}_0, a \neq 0$ and that the dynamics for the state estimate is $\hat{x}_{k+1} = (a - bk)\hat{x}_k$, this implies that

$$\begin{aligned} &\hat{c}\hat{x}_0 = cx_0 \ \Rightarrow\ \hat{c} = \tfrac{cx_0}{\hat{x}_0} \\ &\hat{c}\hat{x}_1 = \hat{c}(a - bk)\hat{x}_0 = cx_1 = cax_0 - cbk\hat{x}_0, \end{aligned}$$

which implies that $\hat{c}a\hat{x}_0 = cax_0$. But, since $\hat{c} = cx_0/\hat{x}_0$ this gives that $\hat{x}_0 = x_0$ and thus also that $\hat{c} = c$.

What this computation shows is that if it is possible to find $\hat{x}_0$ and $\hat{c}$ such that the output string is perfectly reproduced under the dynamics $\hat{x}_{k+1} = (a - bk)\hat{x}_k$, we have in fact solved the SLAM problem uniquely. However, it does not tell us when this is possible or (even less) how one would go about finding $\hat{c}$ and $\hat{x}_0$. However, it does give us hope that the SLAM problem should be solvable under at least some assumptions.

C. Deadbeat SLAM: Uniqueness Through Feedback

We will now return to the general, linear case with state estimate feedback, and we let

$$\begin{aligned} x_{k+1} &= Ax_k - BK\hat{x}_k \\ y_k &= Cx_k, \end{aligned}$$

where $x \in \mathbb{R}^n, u \in \mathbb{R}^m, y \in \mathbb{R}^p$, and where $\hat{x}_k \in \mathbb{R}^n$ is the state estimate at time k.

We note that

$$\begin{aligned} y_0 &= Cx_0 \\ y_1 &= Cx_1 = CAx_0 - CBK\hat{x}_0 \\ y_2 &= Cx_2 = CAx_1 - CBK\hat{x}_1 = \\ &= CA^2x_0 - CABK\hat{x}_0 - CBK\hat{x}_1 \\ &\ \vdots \\ y_n &= CA^n x_0 - \textstyle\sum_{k=0}^{n-1} CA^k BK\hat{x}_{n-1-k}. \end{aligned}$$

The Cayley-Hamilton Theorem states that

$$A^n = \sum_{k=0}^{n-1} \alpha_k A^k,$$

for some real coefficients $\alpha_0, \ldots, \alpha_{n-1}$ defined through the characteristic polynomial of A. What this tells us is that if we let

$$\begin{aligned} \mathcal{Y}_n = \Big[& y_n - \textstyle\sum_{k=0}^{n-1} \alpha_k y_k,\ y_{n+1} - \sum_{k=0}^{n-1} \alpha_k y_{k+1} \\ & \cdots y_{2n-1} - \textstyle\sum_{k=0}^{n-1} \alpha_k y_{k+n-1} \Big] \in \mathbb{R}^{p \times n}, \end{aligned}$$

we have that

$$\mathcal{Y}_n = C\mathcal{M}\mathbf{K}\hat{\mathbf{X}}_0,$$

where

$$\hat{\mathbf{X}}_0 = \begin{bmatrix} \hat{x}_0 & \hat{x}_1 & \cdots & \hat{x}_{n-1} \\ \hat{x}_1 & \hat{x}_2 & \cdots & \hat{x}_n \\ \vdots & \vdots & & \vdots \\ \hat{x}_{n-1} & \hat{x}_n & \cdots & \hat{x}_{2n-2} \end{bmatrix} \in \mathbb{R}^{n^2 \times n}$$

$$\mathbf{K} = \begin{bmatrix} K & 0 & \cdots & 0 \\ 0 & K & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & K \end{bmatrix} \in \mathbb{R}^{nm \times n^2},$$

and

$$\mathcal{M} = \Big[-A^{n-1}B + \sum_{k=1}^{n-1} \alpha_k A^{n-1-k}B,\\ - A^{n-2}B + \sum_{k=2}^{n-1} \alpha_k A^{n-1-k}B,\\ ,\cdots, -AB + \alpha_{n-1}B,\ -B\Big] \in \mathbb{R}^{n\times nm}.$$

What we would like to do is thus to establish conditions under which $\mathcal{M}\mathbf{K}\hat{\mathbf{X}}_0$ has full rank and then simply set

$$\hat{C} = \mathcal{Y}_n \left(\mathcal{M}\mathbf{K}\hat{\mathbf{X}}_0\right)^{-1} \in \mathbb{R}^{p\times n},$$

which would uniquely give us the correct output matrix. Then, as long as the system is observable, finding the state estimate is simply the standard linear estimation problem.

But, before we can establish this result, some notation is needed. Given a $p \times q$ matrix D, we let $\mathcal{R}(D) \subseteq \mathbb{R}^p$ denote the range space of D, given by $\{\eta \in \mathbb{R}^p \mid \exists \mu \in \mathbb{R}^q \text{ s.t. } \eta = D\mu\}$. Similarly, the null space $\{\mu \in \mathbb{R}^q \mid D\mu = 0\} \subseteq \mathbb{R}^q$ is denoted by $\mathcal{N}(D)$. Finally, given a subspace $S \subset \mathbb{R}^p$, by $S^\perp \subset \mathbb{R}^p$ we understand the orthogonal complement of S given by $\{v \in \mathbb{R}^p \mid v^T s = 0\ \forall s \in S\}$.

Lemma 3.1: If (A, B) is a controllable pair, $\mathcal{R}(B^T) \cap \mathcal{N}(K^T) = \{0\}$, and $\mathcal{R}(K^T\mathcal{M}^T) \subseteq \mathcal{R}(\hat{\mathbf{X}}_0)$, then $\text{rank}(\mathcal{M}\mathbf{K}\hat{\mathbf{X}}_0) = n$.

Proof: Let $e_1, \dots, e_n$ be the unit vectors, i.e. the standard, orthonormal basis for $\mathbb{R}^n$. Since (A, B) is a controllable pair, for each e_i, there exists a $k_i \in \{0, \dots, n-1\}$ such that $e_i^T A^{k_i} B \neq 0$. Let $k_{i^\star}$ be the smallest such k_i. This in turn implies that

$$e_i^T\mathcal{M} = \begin{bmatrix} \star & \cdots & \star & -e_{i^\star}^T A^{k_{i^\star}} B & 0 & \cdots & 0 \end{bmatrix} \neq 0,$$

for $i = 1, \dots, n$, and hence $\text{rank}(\mathcal{M}) = n$.

Given an arbitrary $z \in \mathbb{R}^n$, let

$$z^T\mathcal{M} = \begin{bmatrix} \omega_1^T & \omega_2^T & \dots & \omega_n^T \end{bmatrix},$$

with $\omega_i \in \mathbb{R}^m$. Since $\mathcal{M}$ has full rank, at least one $\omega_i \neq 0$. Moreover, the structure of $\mathcal{M}$ directly gives that $\omega_i \in \mathcal{R}(B^T)$, and multiplying together $z^T\mathcal{M}\mathbf{K}$ gives

$$z^T\mathcal{M}\mathbf{K} = \begin{bmatrix} \omega_1^T K & \cdots & \omega_i^T K & \cdots & \omega_n^T K \end{bmatrix}.$$

As $\omega_i \in \mathcal{R}(B^T)$ and thus, per assumption, not in $\mathcal{N}(K^T)$, a direct consequence of this is that $\omega_i^T K = (K^T\omega_i)^T \neq 0$ as long as $\omega_i \neq 0$, $i = 1, \dots, n$, and hence $z^T\mathcal{M}\mathbf{K} \neq 0$.

Now, let $z \in \mathbb{R}^n$ be arbitrary and non-zero. Then $z^T\mathcal{M}K = \xi^T \neq 0$, with $\xi \in \mathcal{R}(K^T\mathcal{M}^T)$. But, if $\mathcal{R}(K^T\mathcal{M}^T) \subseteq \mathcal{R}(\hat{\mathbf{X}}_0)$ this in turn implies that $\xi \in \mathcal{R}(\hat{\mathbf{X}}_0)$ or, more importantly, $\xi \notin \mathcal{R}(\hat{\mathbf{X}}_0)^\perp$. As such, ξ can not be orthogonal to all of $\hat{\mathbf{X}}_0$'s columns, i.e. if we let e_i, $i = 1, \dots, n$ denote the unit vectors in $\mathbb{R}^n$, $\xi^T(\hat{\mathbf{X}}_0 e_i) \neq 0$ for at least one i, which completes the proof. ∎

Now, in order to establish the final result, some additional notation is needed. Let $\mathbf{Y}_p^N = (y_p^T, y_{p+1}^T, \dots, y_{p+N}^T)^T$, and let $\mathcal{N}(C, \hat{x}_p, \dots, \hat{x}_{p+N-1})$ be given by

$$\begin{bmatrix} 0 \\ -CBK\hat{x}_p \\ \vdots \\ -\sum_{k=0}^{N-1} CA^kBK\hat{x}_{p+N-1-k} \end{bmatrix}.$$

Moreover, let the standard observability matrix be denoted by $\mathcal{O}(C)$, where we have added an explicit dependency on C in order to be able to use the estimated output matrix rather than the actual one. Using this notation, it is straightforward to establish the following relation

$$\mathbf{Y}_n^{n-1} = \mathcal{O}(C)x_n - \mathcal{N}(C, \hat{x}_n, \dots, \hat{x}_{2n-1}).$$

Moreover, if $\text{rank}(\mathcal{O}(C)) = n$, i.e. the system is completely observable, then x_n can be recovered from the output sequence as

$$x_n = \mathcal{O}(C)^\dagger(\mathbf{Y}_n^{n-1} + \mathcal{N}(C, \hat{x}_n, \dots, \hat{x}_{2n-1})),$$

where $\mathcal{O}(C)^\dagger$ is the Moore-Penrose pseudo inverse. And, we have thus shown the following, main theorem of this section:

Theorem 3.2: Given the linear system

$$x_{k+1} = Ax_k + Bu_k\\ y_k = Cx_k,$$

where $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$, $y \in \mathbb{R}^p$. If the system is completely controllable and completely observable, $u = -K\hat{x}$, where the $m \times n$ gain matrix K satisfies $\mathcal{R}(B^T) \cap \mathcal{N}(K^T) = \{0\}$ and $\hat{x}$ is the state estimate, then the SLAM problem can be solved uniquely in $2n$ steps as long as it is possible to choose the first $2n-1$ state estimates $\hat{x}_0, \dots, \hat{x}_{2n-2}$ such that $\mathcal{R}(K^T\mathcal{M}^T) \subseteq \mathcal{R}(\hat{\mathbf{X}}_0)$. With these initial state estimates, after observing the first $2n$ outputs, $y_0, \dots, y_{2n-1}$, the output matrix is uniquely given by

$$\hat{C} = \mathcal{Y}_n \left(\mathcal{M}\mathbf{K}\hat{\mathbf{X}}_0\right)^{-1},$$

which involves neither C nor x for its computation. Moreover, $\hat{x}_{2n-1}$ is uniquely given by

$$\hat{x}_{2n-1} = A^{n-1}\mathcal{O}(\hat{C})^\dagger(\mathbf{Y}_n^{n-1} + \mathcal{N}(\hat{C}, \hat{x}_n, \dots, \hat{x}_{2n-1})) - \\ - \sum_{k=0}^{n-2} A^kBK\hat{x}_{2n-2-k},$$

where $\hat{C}$ is given above.

D. Examples

Unicycles and Walls: What is potentially overly restrictive, from a robotics point-of-view, with the assumptions leading up to the previous result is not that the system dynamics are linear, but that the output map is. For example, consider a unicycle robot moving in a fixed (albeit unknown) direction. Moreover, assume that the robot is equipped with a range sensor that measures the distance to a straight wall in the direction perpendicular to the robot's movement.

The problem of finding the robot's position as well as the orientation of the wall obviously seems ill-posed since any simultaneous rotation and translation of the robot and the wall would result in the same measurements. However, by insisting that the output equation is linear, we have in fact already assumed that the origin is the point at which the robot hits the wall, i.e. the translation is already taken cared of. Secondly, the output equation is only linear in the distance to this (arbitrary) origin, and as such, the system is assumed to have already been rotated so that it lines up with the direction in which the robot moves. As such, a linear formulation of this problem has, in effect, already assumed away the rigid transformation of the system that would otherwise cause the solution to be non-unique.

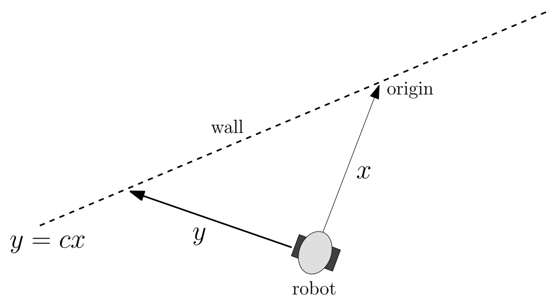

Fig. 2. A robot is approaching a wall with an unknown location and orientation.

Having said that, we will still solve this problem, just to show how the deadbeat SLAM solution developed in the previous paragraphs can be put to use. The setup is shown in Figure 2, where $x \in \mathbb{R}$ is the distance to the point on the wall where the robot's trajectory intersects the wall. (Note, we have thus assumed that the wall is in fact not parallel to the movement of the robot.) Moreover, if $y \in \mathbb{R}$ is the distance to the wall along the direction perpendicular to the movement of the robot, then $y_k = cx_k$, where $1/c$ is the slope of the wall relative to the axis perpendicular to the movement of the robot, as shown in the figure. Assuming we can control the velocity of the robot, the system becomes

$$\begin{aligned} x_{k+1} &= x_k + u_k \\ y_k &= cx_k. \end{aligned}$$

Now, by choosing an arbitrary, non-zero $\hat{x}_0 \in \mathbb{R}$ and using the feedback law $u = -k\hat{x}$, where k is an arbitrary (for stability, we need that $|k| < 1$ but stability is not necessary for the estimator), non-zero scalar we can apply the deadbeat SLAM derived in Theorem 3.2. And, after only one step, we get

$$\hat{c} = \frac{y_0 - y_1}{k\hat{x}_0},$$

which is well-defined as long as $\hat{x}_0, k \neq 0$.

We moreover have that $y_1 \neq y_0$ as long as the wall and the movement of the robot are not parallel, which in turn gives

$$y_1 = cx_1 = \hat{c}\hat{x}_1 \;\Rightarrow\; \hat{x}_1 = \frac{y_1 k}{y_0 - y_1}\hat{x}_0.$$

For general $a, b, c \neq 0$, the one-step deadbeat solution to the scalar, linear SLAM problem is

$$\begin{aligned} \hat{c} &= \tfrac{ay_0 - y_1}{bk\hat{x}_0} \\ \hat{x}_1 &= \tfrac{y_1 bk}{ay_0 - y_1}\hat{x}_0 \\ \hat{x}_{k+1} &= (a - bk)\hat{x}_k, \;\; k = 1, \ldots \end{aligned}$$

A Two Dimensional Example: Now consider the case when $n = 2$ and

$$A = \begin{bmatrix} 0.7 & -0.5 \\ 0.5 & 0.4 \end{bmatrix}, \; B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$
$$K = \begin{bmatrix} 0.1 & 0.1 \end{bmatrix}, \; C = \begin{bmatrix} 1 & 0 \end{bmatrix}.$$

As

$$\text{rank}\left(\begin{bmatrix} B & AB \end{bmatrix}\right) = 2, \;\; \text{rank}\left(\begin{bmatrix} C \\ CA \end{bmatrix}\right) = 2,$$

the system is both controllable and observable. Moreover, as $\mathcal{N}(K^T) = \{0\}$, the condition that $\mathcal{R}(B^T) \cap \mathcal{N}(K^T) = \{0\}$ is trivially satisfied.

The final condition states that $\hat{x}_0, \hat{x}_1, \hat{x}_2$ should be chosen in such a way that $\mathcal{R}(\mathbf{K}^T\mathcal{M}^T) \subseteq \mathcal{R}(\hat{\mathbf{X}}_0)$. And, in our case

$$\mathcal{M}\mathbf{K} = \begin{bmatrix} \alpha_1 B - AB & -B \end{bmatrix} \begin{bmatrix} K & 0 \\ 0 & K \end{bmatrix} =$$
$$= \begin{bmatrix} 0.05 & 0.05 & 0 & 0 \\ 0.07 & 0.07 & -0.1 & -0.1 \end{bmatrix},$$

which implies that

$$\mathcal{R}(\mathbf{K}^T\mathcal{M}^T) = \text{span}\left(\begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}\right).$$

By letting $\hat{x}_0 = (1,1)^T$, $\hat{x}_1 = (0,0)^T$, $\hat{x}_2 = (1,1)^T$, the conditions for Theorem 3.2 are satisfied.

An example of using the deadbeat SLAM solution is shown in Figure 3.

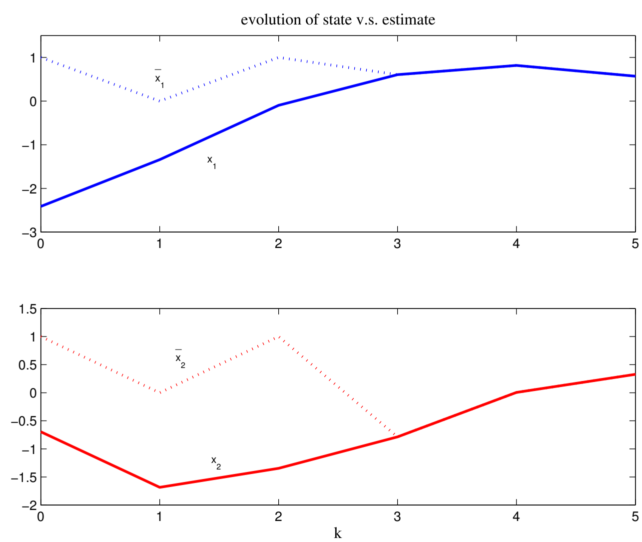

Fig. 3. Dipicted is the true state x (dotted) and the estimate $\hat{x}$ (solid) as function of time.

In Higher Dimensions: One might be tempted to draw the conclusion from the previous example that the conditions under which $\text{rank}(\mathcal{M}\mathbf{K}\hat{\mathbf{X}}_0) = n$ are pathological in the sense that the system parameters must be carefully selected for this to be true. However, based on a large number of simulations, in which the dimension, the system matrices, and the initial state estimates were all selected randomly, $\mathcal{M}\mathbf{K}\hat{\mathbf{X}}_0$ maintained full rank. As such, it seems that the condition for SLAM to be solvable for linear systems with state feedback are generically satisfied, which indicates that teh sufficient conditions in Theorem 3.2 might be overly restrictive. However, a more thorough study of this topic is left to the future.

IV. NONLINEAR SLAM

If the system dynamics and output equations are nonlinear, i.e.

$$\begin{aligned} x_{k+1} &= f(x_k, u_k) \\ y_k &= h(x_k) \end{aligned}$$

things get significantly more complicated. What facilitated the solution in the linear case was that the output gain matrix C could be thought of a state of the system (albeit a constant one), resulting in a linear dynamical system with bilinear output equation. Similarly, if is possible to parameterize the output equation as $y_k = h(\alpha, x_k)$, where $\alpha \in \mathbb{R}^q$ is unknown while h is not, a similar methodology can be applied.

In particular, one possibility is to apply the Extended Kalman Filter to the problem of finding the joint state of the system

$$\begin{aligned} \begin{bmatrix} x_{k+1} \\ \alpha_{k+1} \end{bmatrix} &= \begin{bmatrix} f(x_k, u_k) \\ 0 \end{bmatrix} \\ y_k &= h(\alpha_k, x_k). \end{aligned}$$

And, a more direct method to use in this case of a parameterized output equation is the so-called Grizzle-Moraal Newton observer [20].

It should be noted that these methods both rely on a parametrization of the environment as $y_k = h(\alpha, x_k)$. And, as stated already stated, the choice of representation of the environment becomes key when solving particular instantiations of the SLAM problem. This choice can be related to different ways in which the nonlinear output map is represented, e.g. through a class of basis functions such as wavelets, sigmoids, or Gaussian kernel functions.

V. CONCLUSIONS

In this paper we reformulated the SLAM problem in robotics as a problem involving the simultaneous estimation of both the state of a controlled dynamic system and the output mapping itself. In this manner, a natural representation is obtained that explicitly captures the way the environment maps the robot state onto sensor readings. We show how we can use this formulation to solve the SLAM problem in the linear case, together with sufficient conditions for this solution to exist uniquely. Potentially fruitful directions for further research are outlined with regards to the general, nonlinear problem.

REFERENCES

[1] M. G. Dissanayake, P. Newman, S. Clark, H. Durrant-Whyte, and M. Corba, "A solution to the simultaneous localization and map building (SLAM) problem," *IEEE Transactions on Robotics and Automation*, vol. 17, pp. 229–241, June 2001.

[2] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping (SLAM): Part I the essential algorithms," *IEEE Robotics & Automation Magazine*, vol. 13, no. 2, pp. 99–108, 2006.

[3] T. Bailey and H. Durrant-Whyte, "Simultaneous localization and mapping (SLAM): Part II state of the art," *IEEE Robotics & Automation Magazine*, vol. 13, no. 3, pp. 108–117, 2006.

[4] S. Julier and J. Uhlmann, "A counter example to the theory of simultaneous localization and map building," in *Proc. of the IEEE International Conference on Robotics and Automation (ICRA'85)*, vol. 4, pp. 4238–4243, 2001.

[5] J. A. Castellanos, J. Neira, and J. D. Tardós, "Limits to the consistency of EKF-based SLAM," in *5th IFAC/EURON Symposium on Intelligent Autonomous Vehicles*, July 2004.

[6] T. Bailey, J. Nieto, J. Guivant, M. Stevens, and E. Nebot, "Consistency of the EKF-SLAM algorithm," in *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'06)*, 2006.

[7] J. Andrade-Cetto, *Environment Learning for Indoor Mobile Robots*. PhD thesis, Universitat Politécnica de Catalunya, Barcelona, 2003.

[8] R. Sim and N. Roy, "Global a-optimal robot exploration in slam," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, (Barcelona, Spain), 2005.

[9] J. E. Guivant and E. M. Nebot, "Optimization of the simultaneous localization and map-building algorithm for real-time implementation," *IEEE Transactions on Robotics and Automation*, vol. 17, pp. 242–257, June 2001.

[10] J. J. Leonard and H. J. S. Feder, "Decoupled stochastic mapping," *IEEE Journal of Oceanic Engineering*, pp. 561–571, Oct. 2001.

[11] M. Bosse, P. Newman, J. Leonard, M. Soika, W. Feiten, and S. Teller, "An Atlas framework for scalable mapping," in *Proc. of the IEEE International Conference on Robotics and Automation (ICRA'03)*, (Taipei, Taiwan), 2003.

[12] U. Frese, *An $O(logn)$ Algorithm for Simulateneous Localization and Mapping of Mobile Robots in Indoor Environments*. PhD thesis, University of Erlangen-Nrnberg, 2004.

[13] R. Eustice, M. Walter, and J. Leonard, "Sparse extended information filters: Insights into sparsification," in *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'05)*, pp. 641–648, Aug. 2005.

[14] J. Folkesson and H. Christensen, "Graphical SLAM - a self-correcting map," in *Proc. of the IEEE International Conference on Robotics and Automation (ICRA'04)*, (New Orleans, LA, USA), 2004.

[15] J. Folkesson, P. Jensfelt, and H. Christensen, "Vision SLAM in the measurement subspace," in *Proc. of the IEEE International Conference on Robotics and Automation (ICRA'05)*, (Barcelona, Spain), pp. 30–35, Apr. 2005.

[16] R. Smith, M. Self, and P. Cheeseman, *Autonomous robot vehicles*, ch. Estimating uncertain spatial relationships in robotics, pp. 167–193. New York, NY, USA: Springer-Verlag New York, Inc, 1990. ISBN:0-387-97240-4.

[17] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit, "FastSLAM: A factored solution to the simultaneous localization and mapping problem," in *Proc. of the National Conference on Artificial Intelligence (AAAI-02)*, (Edmonton, Canada), 2002.

[18] A. Elfes, "Using occupancy grids for mobile robot perception and navigation," *Computer*, vol. 22, no. 6, pp. 46–57, 1989.

[19] R. A. Brooks, "Intelligence without representation," *Artifical Intelligence*, vol. 47, pp. 139–159, 1991.

[20] P. Moraal and J. Grizzle, "Asymptotic observers for detectable and poorly observable systems," in *Proc. of the 1995 IEEE Conference on Decision and Control*, (New Orleans), Dec. 1995.