Available online at www.sciencedirect.com

ScienceDirect

IFAC-PapersOnLine 49-15 (2016) 205–211

# Smooth Variable Structure Filter VSLAM

ALLAM Ahmed[1], NEMRA Abdelkrim[2]
Control Unit
Bp17, Ecole Militaire Polytechnique, Bordj El-Bahri
Algiers, Algeria
ahmedallam900@gmail.com[1], karim_nemra@yahoo.fr[2]

HAMERLAIN Mustapha[3]
Division Robotique
CDTA, Baba Hasan
Algiers, Algeria

***Abstract*—Simultaneous localization and mapping (SLAM) is vital for autonomous robot navigation. The robot must build a map of its environment while tracking its own motion through that map. There are many ways to approach the problem, mostly based on the sequential probabilistic approach, based around extended Kalman filter (EKF) or the Rao-Blackwellized particle filter. In order to improve the SLAM solution and to overcome some of the EKF and PF limitations, especially when the process and observation models contain uncertain parameters, we propose to use a robust approach to solve the SLAM problem based on variable structure theory. The new alternative called Smooth Variable Structure Filter SVSF is a predictor corrector estimator based on sliding mode control and estimation concepts. It has been demonstrated that the (SVSF) is stable and very robust face modeling uncertainties and noises. Visual SVSF-SLAM is implemented, validated and compared with EKF-SLAM filter. The comparison confirms the efficient and the robustness of localization and mapping using SVSF-SLAM.***



***Keywords-robot mobile; navigation; filtering; SLAM; EKF; SVSF***

## I. Introduction

The ability of a mobile robot to localize itself is critical to its autonomous operation and navigation. A robot that navigates using maps must be able to accurately localize itself. Consequently, there has been considerable effort on the problem of mobile robot localization and mapping. This problem is known as simultaneous localization and mapping SLAM and there is a vast amount of literature on this topic [1], [2], [3].

SLAM is a technique used by mobile robots to build a map of an unknown environment while simultaneously tracking its own motion. This presents a chicken-and-egg situation: an accurate map is necessary for localization, and accurate localization is necessary to build a map. The interdependency between the estimates of the robot location and the map of the environment makes SLAM an interesting and difficult research problem. There are many ways to approach the problem, mostly based on the sequential probabilistic approach, based around extended Kalman filter (EKF) [4] or the Rao-Blackwellized particle filter [5].

The use of the classical solution to SLAM based on the Extended Kalman Filter EKF-SLAM, displays several shortcomings such as quadratic complexity and sensitivity to incorrect feature tracking, problems due to the employment of linearized models of nonlinear motion and observation models and so inherits many caveats[2]. Nonlinearity and errors modeling can be a significant problem for EKF-SLAM and leads to inevitable and sometimes dramatic, inconsistency in solutions, further the assumption of Gaussian additive noise is often violated, which affects the vehicle and map state estimation and can lead to estimation process divergence. Convergence and consistency can only be guaranteed in the linear case with Gaussian additive noise

While EKF-SLAM and FastSLAM are the two most popular solution methods, newer alternatives, which offer much potential, have been proposed, including the use of the unscented Kalman filter (UKF) proposed by Julier and Uhlmann in SLAM [6]. Unlike the EKF, the UKF uses a set of chosen samples to represent the state distribution. The UKF-SLAM avoids the calculation of the Jacobian and Hessian matrices but also obtain higher approximation accuracy with the unscented transformation (UT). However, for high-dimensional systems, the computation load is still heavy; thus, the filter converges slowly. The cubature Kalman filter (CKF) based on a cubature transform which is more accurate and its complexity computation is lower than the one of the UKF was used in SLAM [7]. The State Dependent Riccati Equations (SDRE) nonlinear filtering formulation was also used with SLAM which avoids the linearization step [8].

However, the above filters are all based on the framework of the Kalman filter (KF); it can only achieve a good performance under the assumption that process, observation and noise model are accurately known. But in practice, the prior noise statistic is usually unknown totally, and the process and observation models may be not well known or contain modeling uncertainties, thus, the state estimation will have large errors, or, even, the filters will be possible to diverge.

To overcome some of these limitations, we propose to use the SVSF filter to solve the SLAM problem. Introduced on 2007, the smoothing variable structure filter (SVSF) is relatively a new filter [9]. It is a predictor correct estimator based on sliding mode control and estimation concepts. The SVSF is used for state and parameter estimation of dynamic system whether linear or nonlinear [10] and tracking applications [11]. Furthermore, it has been demonstrated that the SVSF is extremely robust and stable to modeling uncertainties and noise [9], [12].

In this work, we describe our efforts to investigate an alternative to the EKF based data fusion. This alternative uses the SVSF as a fusion filter for SLAM problem. We propose for the first time a smoothing variable structure filter (SVSF) implementation of SLAM algorithm using both robot wheel encoders and stereo vision system. Our motivations behind solving the SLAM problem using the SVSF is to overcome


Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2016.07.733

some EKF limitations and offer more robustness and stability and less-cost computation to the robot pose and map estimation. We first give an overview of the motion model of our robot and the observation model used and also a brief of the EKF-SLAM algorithm is presented. We give a summary of the SVSF estimation process, and then we explain how the SVSF is used to solve the SLAM problem with the derivation of some of the equations. We then showcase some simulation results from our implementation applied to theoretical datasets. Finally, we draw some conclusions, some final remarks and perspectives.

## II. Robot Motion and Stereo Vision Sensor Models

### A. Process Model

We proceed to model the movement of the robot and the noise associated with it. We assume that the robot is operating in planar environments Fig. 1, whose kinematic state, or pose, is summarized by three variables (1):

$$R = (x_r \quad y_r \quad \theta)^T. \tag{1}$$

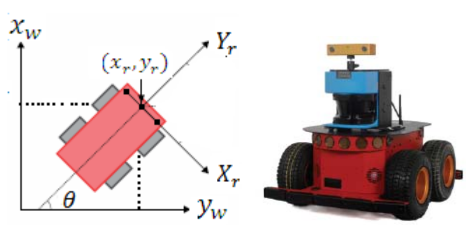


Figure 1. Robot configuration (Pionner 3-AT).

Where T is the sample period, we assume that the robot is controlled by two velocities: a translational velocity $v_k$ and a rotational velocity $w_k$ at sample $k$. We note the control input at sample $k$ by $u_k = (v_k \ \omega_k)^T$. With this control input and the location of the robot at the previous time step we can estimate the robot's current location according to

$$\begin{bmatrix} x_r(k+1) \\ y_r(k+1) \\ \theta(k+1) \end{bmatrix} \begin{bmatrix} x_r(k) + v_k cos\theta(k)T \\ y_r(k) + v_k sin\theta(k)T \\ \theta(k) + \omega_k T \end{bmatrix} \tag{2}$$

$$R_{k+1} = f(R_k, u_k). \tag{3}$$

The model (2) states the kinematic for an ideal, noise-free robot. In reality, robot motion is subject to noise, slip or lift. The actual velocities differ from the measured once by odometer sensor. We will model this difference by a random variable, let:

$$\tilde{u}_k = u_k + n\_w_k = \begin{bmatrix} v_k \\ \omega_k \end{bmatrix} + \begin{bmatrix} \varepsilon_v \\ \varepsilon_w \end{bmatrix}. \tag{4}$$

The measured velocity equals the true velocity plus some small, additive noise $n\_w_k = (\varepsilon_v \ \ \varepsilon_w)^T$[6].

### B. Observation Model

In order to execute SLAM the robot needs to be able to choose and track appropriate reference or landmarks in the environment to localize itself. These landmarks must be stable and invariant.

In this work, we opt for a point landmark approach, where the map is a collection of 3D landmark locations. So, how to obtain relative measurement of the landmarks from the images acquired from the stereo vision sensor?

Figure 2. Observation system geometry.

To validate in simulation the SVSF-SLAM algorithm, we will not use real data, instead of this, theoretical datasets (a set of 3D points) previously generated will be used. When the robot is moving, it detects landmarks (3D points) that are included in the vision sensor field Fig. 3.

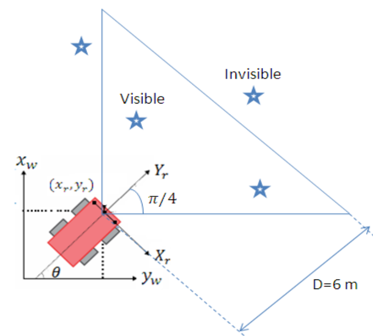


Figure 3. Sensor vision field.

As said previously, the stereo vision sensor provides relative measurement $Z = (L_x^r \quad L_y^r \quad L_z^r)^T$ of the landmarks with respect to the robot frame, this measurement (observation) will be noted $Z$.

The model representing the robot frame coordinates of an individual landmark according to its global frame coordinates $L^g = \left(L_x^g \quad L_y^g \quad L_z^g\right)^T$ and the robot configuration $R = (x_r \ y_r \ 0)^T$ is called the direct model observation and will be noted:

$$Z = h(R, L^g). \tag{5}$$

$$Z = M_{GR} \begin{pmatrix} L_x^g - x_r \\ L_y^g - y_r \\ L_z^g - 0 \end{pmatrix} \tag{6}$$

We denote $M_{GR}$the global to robot projection matrix:

$$M_{GR} = \begin{bmatrix} \cos(\theta) & \sin(\theta) & 0 \\ -\sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{7}$$

In reality the observation (measurement) is subject to noise. The real observation model is given in the following model:

$$\tilde{Z} = h(R, L^g) + n\_v_k. \tag{8}$$

Where $n\_v_k = (\varepsilon_x \ \varepsilon_y \ \varepsilon_z)^T$ is the measurement noise.

### C. Inverse Observation Model

Regarding the dynamic nature of the SLAM algorithm, new observed landmark must be initialized prior to be added to the