Online estimation of covariance parameters using extended Kalman filtering and application to robot localization

Gianluigi Pillonetto*   Gorkem Erinc†   and Stefano Carpin†

*Department of Information Engineering, University of Padova, Italy `giapi@dei.unipd.it`

†School of Engineering, University of California, Merced, USA {`gerinc,scarpin`}`@ucmerced.edu`

Abstract

This paper presents a novel method for the online estimation of variance parameters regulating the dynamics of a nonlinear dynamic system. The approach exploits and extends classical iterated Kalman filtering equations by propagating an approximation of the marginal posterior of the unknown variances over time. In addition to the theoretical foundations, this manuscript offers also a variety of numerical results. In particular, experiments with data collected both in simulation and with a real robot platform show how the proposed approach efficiently solves a robot localization problem.

Keywords: estimation; localization; non-linear systems.

1 Introduction

Kalman filter (KF) based algorithms are used to solve a variety of estimation problems spanning from robotics to biomedicine and economy. Numerous reasons account for its popularity, and the reader is referred to [1, 2] for detailed expositions of its properties. Central to its application is the assumption that transition and measurement noises are Gaussian. When the system is linear these hypotheses have favorable computational implications since they account for the conciseness of the parametric representation of the different variables involved in the estimation process. When the system is nonlinear the simplest generalization is the extended or iterated Kalman filter [3]. In either case, a drawback is the necessity to determine beforehand the parameters characterizing the noise statistics. This may be a time consuming process that may be rather complicated if the model is not simple. However, there are various situations when this is not the case. Moreover, certain classes of systems exhibit dynamics governed by time-varying noise distributions, so that one should update the system description while it is evolving, rather than relying on stationary assumptions or off-line measures.

These problems may be mitigated by the approach presented in this manuscript. The main contribution of this paper is an algorithm to estimate online the covariance parameters characterizing both the measurement and evolution noise. In particular, we interpret these parameters as random variables and propose an algorithm that propagates an approximation of their marginal posterior over time. When identifying dynamical systems in an off-line manner, this approach is known to be more robust than treating the covariance parameters as part of the state vector and using the standard KF to estimate them. Our novel contribution is in showing that this estimation can be efficiently computed online. In fact, integrating out the state vector from the joint posterior, i.e. the joint probability density of the state vector and the parameters conditional on the output data, may remove biases in variance estimation [4, 5]. Integration of the state vectors in an off-line context is also performed in stochastic grey box models identification and is known to enhance the performance of the estimation process, see [6]. Other methods have shown that covariance estimation can be performed thorough simulations accounting for different error sources characterizing the physical process being modeled [7]. Our method differs because it computes estimates online without making assumptions about the underlying process, e.g. it does not need to account for different error sources. Moreover, we demonstrate that this approximation is robust with respect to initial guesses of the parameter values.

To illustrate its strengths, the estimator is applied to a robot localization problem. In our experimental setup we will focus on mobile robots moving in planar environments, where the aim is to determine the position in the plane and the *yaw* orientation of the robot. In other words we look at algorithms that estimate the triple (x, y, θ), indicated as *pose* in the following. The recent book by Thrun et al. [8] offers an up to date description of the most commonly used probabilistic techniques adopted to solve this problem. Most of the published literature in the field resorts to some implementation of the Bayes filter, e.g. the extended Kalman filter (EKF)[1] for position tracking [9] and particle filters for global localization problems [10, 11, 12, 13]. Extensions to the so called multi-robot localization problem can also be found in [14, 15, 16, 17, 18, 19].

In the experimental section present in this paper a mobile platform moves in an environment conditioned with landmarks located at known locations, and it is equipped with a sensor identifying the landmarks and estimating their distance. We use an iterated Kalman filter to reconstruct robot's path and, differently from all the works cited above, we assume that the covariance parameters of the sensors or of the state transitions are unknown and must be estimated online. We show that the proposed approach enables the use of Kalman based techniques even in cases where one has no information at all concerning the statistics of the involved Gaussian distributions. The problem is studied both in simulation and with a real mobile platform. Results illustrate the performance of the proposed technique, also pointing out situations where the unknown covariance parameters cannot be identified.

The paper is organized as follows. Section 2 defines the problem and presents the theory supporting our approach. The experimental setup is presented in Section 3, and results are illustrated in Section

---

[1]Extended Kalmand Filter is *de facto* almost always preferred to the basic Kalman filter due to the nonlinearities in robot motion and sensing.

4. Finally conclusions are given in Section 5.

2 Problem definition and theoretical background

We are interested in estimating the n dimensional state $\mathbf{x} \in \mathbb{R}^n$ of a control system evolving according to the discrete-time state-space model:

$$\mathbf{x}(t) = g(\mathbf{x}(t-1), \mathbf{u}(t)) + \nu(t), \quad t = 0, 1, 2, \ldots \tag{1}$$

where $\nu(t)$ is a zero-mean n-dimensional Gaussian random vector with covariance matrix $Q(t)$, and $u(t) \in \mathbb{R}^k$ is a vector of inputs. We assume the availability of a sensor providing observations for $\mathbf{x}$, i.e.

$$\mathbf{z}(t) = h(\mathbf{x}(t)) + \zeta(t) \tag{2}$$

where $\mathbf{z}(t) \in \mathbb{R}^m$ and $\zeta(t)$ is a zero-mean m-dimensional Gaussian random vector with covariance matrix $R(t)$. Finally, we assume an initial estimate for $\mathbf{x}(0)$ is available as a Gaussian vector independent of transition and measurement noise, with known mean and covariance matrix. Under these assumptions, in a Bayesian framework the evolution in time of $\mathbf{x}$ can be estimated using the Kalman filter (KF). Due to possible nonlinearities in the functions g and h, a linearization around the current estimate for $\mathbf{x}$ is usually performed leading to the extended Kalman filter (EKF) [20].

The situation addressed in this paper is complicated because we consider that the matrices $R(t)$ and $Q(t)$ are not known, but rather depend on an unknown parameter vector $\theta \in \mathbb{R}^p$. Hence, these matrices shall rather be written as $R(t, \theta)$ and $Q(t, \theta)$. The problem we aim to solve is the online estimation of both $\mathbf{x}(t)$ and θ starting from the initial estimate for $\mathbf{x}(0)$, and the given sequences $\mathbf{u}(t)$, and $\mathbf{z}(t)$. In the spirit of the KF framework, the estimation process for both $\mathbf{x}$ and θ will produce a posterior distribution where both these quantities are modeled as Gaussian random vectors. In particular we aim solve the problem under the following assumption.

Assumption 1 *Given θ the random vectors $\{\nu(t)\}$ and $\{\zeta(t)\}$ are all mutually independent with zero mean and covariance matrix $Q(t, \theta)$ and $R(t, \theta)$.*

In the sequel we will indicate with $\hat{\mathbf{x}}(t|t)$ the estimate of $\mathbf{x}$ obtained taking into account all data available up to time t. $\hat{\mathbf{x}}(t|t-1)$ will instead indicate the estimate for $\mathbf{x}$ taking into account all inputs up to time t and all sensor readings up to time $t-1$, that is after the prediction step but before the update step. Similarly, $\hat{\Sigma}_{\mathbf{X}}(t|t)$ indicates the covariance matrix of the estimate for $\mathbf{x}$ taking into account all data up to time t, while $\hat{\Sigma}_{\mathbf{X}}(t|t-1)$ is the covariance matrix after the prediction but before the update. Estimates for θ will also be formulated as a Gaussian distribution, and the symbols $\hat{\theta}(t)$ and $\hat{\Sigma}_{\theta}(t)$ indicate the mean and the covariance matrix taking into account all data up to time[2] t. The estimation process involves two stages that are now described in detail.

---

[2]Due to the computations performed to estimate θ, there is no need to distinguish between prediction and update for θ.

2.1 Prediction

The first step is the prediction step of the EKF. During this initial stage both $\hat{\mathbf{x}}(t-1|t-1)$ and $\hat{\Sigma}_{\mathbf{X}}(t-1|t-1)$ are not considered as dependent on θ. Let us define the $n \times n$ Jacobian matrix around $\hat{\mathbf{x}}(t-1|t-1), \mathbf{u}(t)$, i.e.

$$G_t = \frac{\partial g(\mathbf{x}(t-1), \mathbf{u}(t))}{\partial \mathbf{x}(t-1)}|_{\hat{\mathbf{x}}(t-1|t-1), \mathbf{u}(t)}$$

Therefore the state evolution function g appearing in Eq. 1 can be linearly approximated as

$$g(\mathbf{x}(t-1), \mathbf{u}(t)) \approx g(\hat{\mathbf{x}}(t-1|t-1), \mathbf{u}(t)) + G_t [\mathbf{x}(t-1) - \hat{\mathbf{x}}(t-1|t-1)] + \nu(t).$$

The estimate for $\mathbf{x}$ at time t before considering the observation $\mathbf{z}(t)$ can be approximated as a Gaussian vector whose mean and covariance matrix are:

$$\hat{\mathbf{x}}(t|t-1) = g(\hat{\mathbf{x}}(t-1|t-1), \mathbf{u}(t))$$
$$\hat{\Sigma}_{\mathbf{X}}(t|t-1) = G_t \hat{\Sigma}_{\mathbf{X}}(t-1|t-1)G_t^T + Q(t, \theta) \tag{3}$$

The reader should observe that whilst we assumed $\hat{\Sigma}_{\mathbf{X}}(t-1|t-1)$ does not depend on θ, after the prediction step $\hat{\Sigma}_{\mathbf{X}}(t|t-1)$ depends on θ through $Q(t, \theta)$. In order to make this dependency explicit in the following we will therefore write $\hat{\Sigma}_{\mathbf{X}}(t|t-1)(\theta)$.

2.2 Update based on measurement

When the sensor reading $\mathbf{z}(t)$ is incorporated, a better estimate for $\hat{\mathbf{x}}$ is achievable, i.e. it is possible to compute $\hat{\mathbf{x}}(t|t)$ by taking into account the innovation. However, in this stage we also aim to provide a better estimate for θ, i.e. $\hat{\theta}(t)$ and $\hat{\Sigma}_{\theta}(t)$ starting from $\hat{\theta}(t-1)$ and $\hat{\Sigma}_{\theta}(t-1)$. The update stage relies on the following proposition[3].

Proposition 2 *Assume $\mathbf{x}(t)$ is Gaussian with mean $\hat{\mathbf{x}}(t|t-1)$ and covariance $\hat{\Sigma}_{\mathbf{X}}(t|t-1)(\theta)$. Then, for known θ and $\mathbf{z}(t)$, the maximum a posteriori estimate of $\mathbf{x}(t)$ is*

$$\hat{\mathbf{x}}(t|t) = \arg\min_{\mathbf{X}(t)} l(\mathbf{x}(t); \mathbf{z}(t), \theta)$$

where

$$l(\mathbf{x}(t); \mathbf{z}(t), \theta) = \frac{1}{2} \ln \det(2\pi R(t, \theta)) + \frac{1}{2} \ln \det(2\pi \hat{\Sigma}_{\mathbf{X}}(t|t-1)(\theta))$$
$$+ \frac{1}{2} \|\mathbf{z}(t) - h(\mathbf{x}(t)), R(t, \theta)^{-1}\|^2 + \frac{1}{2} \|\mathbf{x}(t) - \hat{\mathbf{x}}(t|t-1), \hat{\Sigma}_{\mathbf{X}}(t|t-1)(\theta)^{-1}\|^2. \tag{4}$$

The proof of proposition 2 is immediate once one writes down the expression for the joint density of $\mathbf{x}$ and $\mathbf{z}$ conditioned on θ and considers assumption 1.

---

[3]If W is a $r \times r$ symmetric positive definite matrix and $v \in \mathbf{R}^r$ we define $\|v, W\|^2 = v^T W v$.

Remark: the reader will recognize that Eq. 4 defines a function l that is related to the logarithm of a multidimensional Gaussian distribution. This function is explicitly introduced to simplify the description of the minimization algorithm described in the following.

In order to optimize the objective function $l(.)$ (Eq. 4), a Gauss-Newton method can be exploited (see e.g. [21]). It is now pivotal to recall that the iterated Kalman filter (IKF) update is exactly a Gauss-Newton method, i.e. once assigned the same initial point, the sequence of iterates generated by the Gauss-Newton method and the sequence of iterates produced by IKF are identical (see [22]). Recall also that IKF corresponds to the EKF when just a single iteration is performed. Thus, the minimizer of $l(.)$ can be achieved by inductively defining the sequences $\mathbf{x}^i$ and $\Sigma_{\mathbf{X}}^i$ as follows:

$$\mathbf{x}^0 = \hat{\mathbf{x}}(t|t-1) \tag{5}$$
$$\Sigma_{\mathbf{X}}^0 = \hat{\Sigma}_{\mathbf{X}}(t|t-1)(\theta)$$
$$\mathbf{x}^{i+1} = \hat{\mathbf{x}}(t|t-1) + K^i \left[\mathbf{z}_i - h(\mathbf{x}^i) - H_i \left(\hat{\mathbf{x}}(t|t-1) - \mathbf{x}^i \right) \right]$$
$$\Sigma_{\mathbf{X}}^{i+1} = (I - K_i H_i)\hat{\Sigma}_{\mathbf{X}}(t|t-1)(\theta)$$

where

$$H_i = \frac{\partial h(\mathbf{x})}{\partial \mathbf{x}(t)}(\mathbf{x}^i) \tag{6}$$
$$K_i = \hat{\Sigma}_{\mathbf{X}}(t|t-1)(\theta)H_i^T \left[H_i \hat{\Sigma}_{\mathbf{X}}(t|t-1)(\theta)H_i^T + R(t,\theta) \right]^{-1}$$

After computing a sufficient number of iterations to reach convergence, $\mathbf{x}^i$ and $\Sigma_{\mathbf{X}}^i$ provide the updated estimate $\hat{\mathbf{x}}(t|t)$ and the covariance matrix of the error $\hat{\Sigma}_{\mathbf{X}}(t|t)$. However, both these quantities now depend on θ through $R(t,\theta)$ and $Q(t,\theta)$. Therefore, similarly to the notation used for $\hat{\Sigma}_{\mathbf{X}}$, we will write $\hat{\mathbf{x}}(t|t)(\theta)$ to outline its dependency of $\mathbf{x}$ from θ. The core question at this point is how the estimate of the unknown vector θ can be updated.

Let $\pi(\mathbf{x}(t), \mathbf{z}(t)|\theta)$ denote the joint density for $\mathbf{x}(t)$ and $\mathbf{z}(t)$ conditioned on θ. Recalling Eq. 4, one has

$$\pi(\mathbf{x}(t), \mathbf{z}(t)|\theta) = e^{-l(\mathbf{X}(t);\mathbf{Z}(t),\theta)}$$

Let instead $\pi(\mathbf{z}(t)|\theta)$ be the marginal likelihood of $\mathbf{z}(t)$, which corresponds to the joint density of $\mathbf{z}(t)$ and $\mathbf{x}(t)$, where $\mathbf{x}(t)$ is integrated out, i.e.

$$\pi(\mathbf{z}(t)|\theta) = \int \pi(\mathbf{x}(t), \mathbf{z}(t)|\theta)d\mathbf{x}(t). \tag{7}$$

This integral is useful since it often allows to remove biases in parameter estimation [23]. Let $\pi_{t-1}(\theta)$ denote the current "prior" for θ, i.e. a Gaussian with mean $\hat{\theta}(t-1)$ and covariance $\hat{\Sigma}_\theta(t-1)$. Then, our target estimate for θ is

$$\hat{\theta} = \text{argmin}_\theta - \ln\left[\pi(\mathbf{z}(t)|\theta)\pi_{t-1}(\theta)\right]$$

under the constraint that the resulting matrices $Q(t,\theta)$ and $R(t,\theta)$ are valid covariance matrices, i.e. symmetric positive definite matrices.

Due to the possible non linearity of the measurement function h, the evaluation of $\pi(\mathbf{z}(t)|\theta)$ for a given θ requires to solve the integral in Eq. 7 which in general is analytically intractable. However, we now illustrate that the computations performed by the IKF provide an approximation for this integral. To start with, let us first consider the affine approximation of $l(\mathbf{x}(t); \mathbf{z}(t), \theta)$ for $\mathbf{x}(t)$ near $\mathbf{y} \in \mathbb{R}^n$. This is defined by

$$\begin{aligned}
\tilde{l}(\mathbf{x}(t), \mathbf{y}; \mathbf{z}, \theta) = {} & \frac{1}{2}\ln\det(2\pi R(t,\theta)) + \frac{1}{2}\ln\det(2\pi\hat{\Sigma}_{\mathbf{x}}(t|t-1)(\theta)) \\
& + \frac{1}{2}\|\mathbf{z}(t) - h(\mathbf{y}) - H_{\mathbf{y}}(\mathbf{x}(t) - \mathbf{y}), R(t,\theta)^{-1}\|^2 \\
& + \frac{1}{2}\|\mathbf{x}(t) - \hat{\mathbf{x}}(t|t-1), \hat{\Sigma}_{\mathbf{x}}(k|k-1)(\theta)^{-1}\|^2
\end{aligned}$$

with

$$H_{\mathbf{y}} = \frac{\partial h(\mathbf{x})}{\partial \mathbf{x}(t)}(\mathbf{y}).$$

The Hessian of $\tilde{l}$ with respect to $\mathbf{x}(t)$ is easily obtained and reads as follows

$$\frac{\partial^2 \tilde{l}(\mathbf{x}(t), \mathbf{y}; \mathbf{z}(t), \theta)}{\partial \mathbf{x}^2(t)} = H_{\mathbf{y}}^T R(t,\theta)^{-1} H_{\mathbf{y}} + \hat{\Sigma}_{\mathbf{x}}(t|t-1)(\theta)^{-1}. \tag{8}$$

Let us now consider again Eq. 7. The information matrix approximation for the marginal likelihood $\pi(\mathbf{z}(t)|\theta)$, denoted by $\tilde{\pi}(\mathbf{z}(t)|\theta)$, is given by (see [24] and [25])

$$\tilde{\pi}(\mathbf{z}(t)|\theta) = \det\left\{\frac{1}{2\pi}\frac{\partial^2 \tilde{l}(\mathbf{x}(t), \hat{\mathbf{x}}(t|t)(\theta); \mathbf{z}_k, \theta)}{\partial \mathbf{x}^2(t)}\right\}^{-1/2} e^{-l(\hat{\mathbf{x}}(t|t)(\theta); \mathbf{z}(t), \theta)} \tag{9}$$

Therefore, in view of Eq. 8 and Eq. 9, one can notice that such an approximation requires $\hat{\mathbf{x}}(t|t)(\theta)$ and the Jacobian of h computed at $\hat{\mathbf{x}}(t|t)(\theta)$ which represent quantities returned by IKF. Stated differently, for every θ value IKF can be used to evaluate an objective function whose optimization provides the estimate of θ. Thus, we are now in a position to define the estimate of θ, obtained after seeing all data up to instant t, as the solution to the problem

$$\hat{\theta}(t) = \text{argmin}_\theta - \ln\left[\tilde{\pi}(\mathbf{z}(t)|\theta)\pi_{t-1}(\theta)\right] \tag{10}$$

while $\hat{\Sigma}_\theta(t)$ is given by

$$\hat{\Sigma}_\theta(t) = -\left[\frac{\partial^2 \ln(\tilde{\pi}(\mathbf{z}(t)|\hat{\theta}(t))\pi_{t-1}(\hat{\theta}(t)))}{\partial \theta^2}\right]^{-1} \tag{11}$$

and can be calculated numerically.

This completes the update for parameter θ. Finally, the overall measurement update is completed by setting the estimate of $\mathbf{x}(t)$ to $\hat{\mathbf{x}}(t|t)(\hat{\theta}(t))$ with associated covariance matrix of the error given by

$\hat{\Sigma}_{\mathbf{x}}(t|t)(\hat{\theta}(t))$.

The covariance matrix of Eq. 11 provides an indication of the information on θ carried by output data collected up to instant k. Notice however that, due to possible modeling errors or particular noise realizations, the error covariance matrix could become small (as it may happen for state estimation and discussed e.g. in Chapters 8 and 9 of [20]). Thus, a drawback is that subsequent output data could have no effect on the estimate of θ. To prevent divergence, a simple approach consists in fixing a lower bound on the elements of matrix $\hat{\Sigma}_{\theta}(t|t)$. This approach could also have advantages if one assumes time-variance of the parameters and one wants to increase adaptivity of the estimates.

3   Experimental setup

The estimation algorithm has been experimentally tested using the two different dynamical models introduced later on in this section[4]. In the various situations we always compare the performance of three estimators. The first one, called EST in the following, is based on the new approach proposed in this paper. It will be used to estimate the state of the dynamical system and to identify online the unknown covariance parameters, either for measurement noise or for transition noise. The second estimator, indicated as IKF1 in the following, is an iterated Kalman filter that always uses the correct noise autocovariances and performs 10 iterations during its update step. In the context of these experiments, IKF1 can be considered as the ideal estimator for the state of the system. The third estimator, indicated as IKF2 in the following, is also an iterated Kalman filter performing 10 iterations, but it is fed with inaccurate information about the various covariances, as indicated in the different experiments described later on. IKF2 is introduced to outline that poorly chosen values for these parameters may lead to poor performance. On the contrary, our experiments demonstrate that even when boostrapping our proposed EST with arbitrarily wrong parameters, its performance eventually converges to the ideally calibrated iterated Kalman filter. The same symbols are used both when describing simulation results and results obtained with a mobile robotics platform. We conclude this introduction mentioning that both IKF1 and IKF2 use the linearized model during their iterations.

3.1   A realistic dynamical model for robot localization

We first consider a differential drive robot moving on a flat surface, as described in [8] (chapter 5). Assuming that its initial pose is known, the goal of the estimation process is to track its pose while the robot moves around. As usual in this kind of applications, the pose of the robot is a three dimensional vector $\mathbf{x} = (x, y, \theta)$, where x and y are the coordinates of the mid point on the axle between the wheels, and θ is the robot heading measured with respect to a fixed reference frame. Two inputs are used to drive the robot. These are $\mathbf{u} = (u_v, u_w)$, and correspond to the translational and rotational speeds,

---

[4]Matlab© code implementing the algorithm described in this paper and data sets used to produce the results presented in this section are available for download from http://robotics.ucmerced.edu.

respectively. When $u_w(t) \neq 0$, the equations describing the next state of the robot are the following (when $u_w(t) = 0$ the robot moves forward without turning and the relationships are simplified in the obvious way).

$$\begin{cases} x(t) = x(t - \Delta t) + \left[-\frac{u_v(t)}{u_w(t)} \sin \theta(t - \Delta t) + \frac{u_v(t)}{u_w(t)} \sin(\theta(t - \Delta t) + u_w(t)\Delta t) \right] \\ y(t) = y(t - \Delta t) + \left[\frac{u_v(t)}{u_w(t)} \cos \theta(t - \Delta t) - \frac{u_v(t)}{u_w(t)} \cos(\theta(t - \Delta t) + u_w(t)\Delta t) \right] \\ \theta(t) = \theta(t - \Delta t) + u_w(t)\Delta t \end{cases} \tag{12}$$

Consistently with the model set in Eq. 1, the evolution noise corrupting $\mathbf{x}(t)$ is assumed to be Gaussian, i.e. $\nu(t) \in \mathcal{N}(0, Q(t))$. $Q(t)$ can be either known, or depending on unknown parameters and possibly the input vector $\mathbf{u}$, as assumed in [8]. A detailed discussion about the form of Q and its estimation is postponed to a later subsection. Assuming C landmarks are located in the environment at known positions, we hypothesize the robot is equipped with a single sensor returning only measurements to landmarks closer than a certain threshold T. Therefore the measurement vector at time t is a vector $\mathbf{z}(t) \in \mathbb{R}^M$ with $M \leq C$. We furthermore assume that the *landmark correspondence problem* does not occur, i.e. for each element in $\mathbf{z}(t)$ it is known to which landmark it corresponds to[5]. Finally, we hypothesize that the individual readings of $\mathbf{z}(t)$ are mutually independent and affected by Gaussian noise with zero mean and variance ζ, i.e. the measurement noise covariance matrix at time t is

$$R(t) = \zeta I_M \tag{13}$$

where I_M is the $M \times M$ identity matrix. In the sequel, the value of ζ will be either assumed known or estimated online by the algorithm. We conclude this section pointing out that it is straightforward to extend this formulation assuming different values on the main diagonal of $R(t)$, for example to model the case where distance measurements from different landmarks imply different precision. This case will not be discussed here but can be accommodated with a careful implementation of the the algorithm, while the underlying theory does not change at all.

3.2 A simplified dynamical model for robot localization

The second dynamical model considers a simplified situation[6] where the robot translates only on the plane, and so its state vector is $\mathbf{x} = (x, y)$, where x and y are its coordinates in the plane. Two inputs are used to drive the robot, $\mathbf{u} = (u_1, u_2)$, and the state transition equations are as follow:

$$\begin{cases} x(t) = x(t - \Delta t) + u_1(t)\Delta t \\ y(t) = y(t - \Delta t) + u_2(t)\Delta t \end{cases} \tag{14}$$

Consistently with the model set in Eq. 1, the evolution noise corrupting $\mathbf{x}(t)$ is assumed to be Gaussian, i.e. $\nu(t) \in \mathcal{N}(0, Q)$. Q is assumed to be constant over time and diagonal. The robot is equipped with a sensor identical to the one considered in the previous model.

---

[5]The case of uknown correspondences can be dealt with as well via maximum likelihood, but it is not considered in this manuscript. The reader is referred to [8], chapter 7, for details.

[6]Although simplified, the model can be appropriate to model holonomic platforms.

4 Experimental results

In this section we first illustrate simulation results for both models and a variety of situations, and we then discuss an implementation on a mobile platform.

4.1 Simulated data

4.1.1 Estimating the measurement noise covariance

We start considering the evolution model given in Eq. 12 and assume that the transition noise covariance $Q(t)$ is known to all estimators. Instead, ζ (and then $R(t)$) is assumed to be unknown to EST; IKF2 uses a value 5 times bigger; and IKF1 is always provided with correct values for R. Figures 1 and 2 display prototypical results for one of the tests. Figure 1 shows the estimated value for ζ produced by the proposed estimator EST as a function of time. The true ζ value is 10^{-1}, and the plot shows that even though EST starts with an initial estimate of $\zeta = 10^2$, it quickly decreases its value and it eventually converges to the right number. Figure 2 instead shows the position and orientation error for the three different estimators through the mission. It is evident that there is no appreciable difference in performance between EST and IKF1, while IKF2, suffering from an poor choice for the measurement noise covariance, shows sporadic burst errors. We conclude remarking that the performance of EST is basically independent of the initial choice for ζ. In fact, even initializing EST with a ζ estimate that is 10^8 times larger than the correct value the algorithm shows the same performance. The figures display a single set of results, but these are consistent with the observed behavior over a large number of trials, as shown in Table 1. Similar findings are also evidenced when using the evolution model given by Eq. 14.

Average estimated ζ	Variance estimated ζ	EST average error	IKF1 average error	IKF2 average error
0.101	0.00789	0.0428	0.0424	0.0528

Table 1: Results obtained averaging 500 random trials with the same conditions used to produce Fig. 1 and 2. The unknown measurement noise covariance is $\zeta = 0.1$. Displayed errors are the average position errors throughout the 500 trials and are expressed in meters.

Finally, Table 2 outlines the robustness of the proposed estimator EST. Three different experiments are performed with EST starting with estimates for ζ off by a factor between 10^2 and 10^6 (second column). The third column shows that the overall error is insensitive to the initial mismatch, whereas the fourth column displays the estimated value for ζ. The true value for ζ, unknown to the estimator, is shown for comparison in the last column.
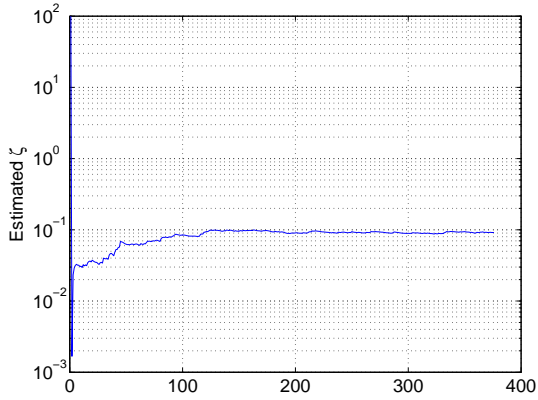
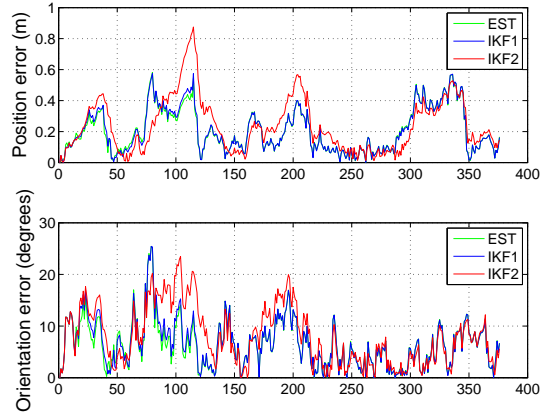Figure 1: Trend of the estimated ζ value produced by EST. The true value for ζ is 10^{-1}.

Figure 2: Trends for position and orientation error as a function of time.

Table 2: Measurement noise covariance estimation

Experiment	Bias	Error	Estimated ζ	True ζ
	100	0.015852	0.045365	0.050000
1	10000	0.015851	0.045354	0.050000
	1000000	0.015850	0.045354	0.050000
	100	0.010416	0.042807	0.050000
2	10000	0.010419	0.042813	0.050000
	1000000	0.010416	0.042800	0.050000
	100	0.061317	0.092201	0.100000
3	10000	0.061320	0.092205	0.100000
	1000000	0.061317	0.092225	0.100000

4.1.2 Estimating the evolution noise covariance

The situation may be more intricate when the evolution noise covariance needs to be estimated. These complications are not a limit of the proposed technique, but may rather outline an impossibility to determine the unknown covariance parameters. This kind of non identifiability issues are in particular related to the structure of the model in Eq. 12. In particular, recalling Eq. 3, the covariance matrix after the prediction step is

$$\hat{\Sigma}_{\mathbf{X}}(t|t-1) = G_t \hat{\Sigma}_{\mathbf{X}}(t-1|t-1) G_t^T + Q(t, \theta)$$

where the expression for G_t depends on the state transition equation. Using realistic values to fully define the model in Eq. 12, we have assessed that the first term in Eq. 3 always dominates the second one. As a consequence, it is not possible to reliably recover the unknown parameters θ. In fact, the estimator tends to underestimate Q since the uncertainty described by $G_t \hat{\Sigma}_{\mathbf{X}}(t-1|t-1) G_t^T$ is already

large enough to perform the state update in a satisfactory manner. This is shown in Fig. 3 where it is evident that underestimation of θ does not imply that the estimator returns erroneous estimates for the state $\mathbf{x}$. Instead, similarly to what was observed in the previous case it eventually shows a performance similar to IKF1.
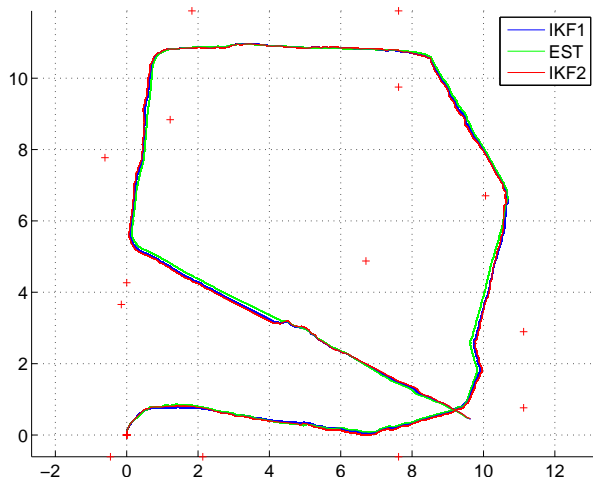

Figure 3: Estimated path by the three estimators while using the model given by Eq. 12. Red crosses show the landmarks locations, and distances are measured in meters.

The non identifiability scenario for θ estimation changes when the model given in Eq. 14 is considered. In this case $G_t = I_2$, so that Eq. 3 becomes

$$\hat{\Sigma}_{\mathbf{X}}(t|t-1) = \hat{\Sigma}_{\mathbf{X}}(t-1|t-1) + Q(t,\theta).$$

and $\hat{\Sigma}_{\mathbf{X}}(t-1|t-1)$ in general does not dominate $Q(t,\theta)$ any more. In particular, we consider two different cases for $Q(t,\theta)$, namely

$$Q_1(t,\theta) = \begin{bmatrix} \theta & 0 \\ 0 & \theta \end{bmatrix} \quad Q_2(t,\theta) = \begin{bmatrix} \theta_1 & 0 \\ 0 & \theta_2 \end{bmatrix}$$

Figures 4 and 5 show an example of results obtained while estimating Q_1 with $\theta = 10^{-2}$. Similarly to the previous situation, the estimated value for the unknown covariance parameter converges to the unknown value (Figure 4). Figure 5 instead shows the trend of the position error as a function of time (the reader should notice that in this case the state has dimension two, so no orientation error is defined). IKF2 in this case is run with a value for θ three times larger than the correct one. One can see that a moderately wrong choice for θ can lead to large errors. On the contrary, even though EST is initialized with $\theta = 10$ (i.e. off by a factor of 1000), its performance eventually matches IKF1, i.e. the estimator equipped with the correct θ.

Finally, we consider the case where Q_2 is estimated, with $\theta_1 = 10^{-2}$ and $\theta_2 = 10^{-3}$. Results are displayed in figures 6 and 7. Once again the estimation process eventually computes values for the unknown θ
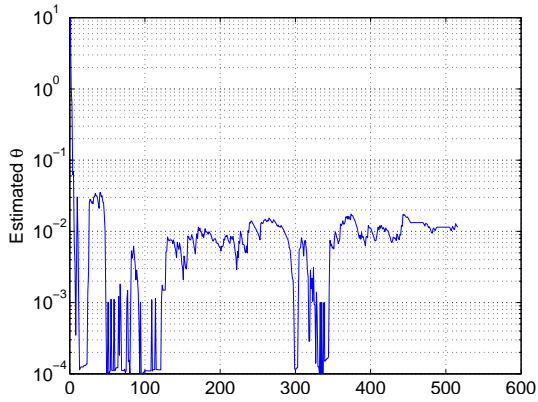
Figure 4: Trend of the estimated θ value produced by EST. The true value for θ is 10^{-2}.
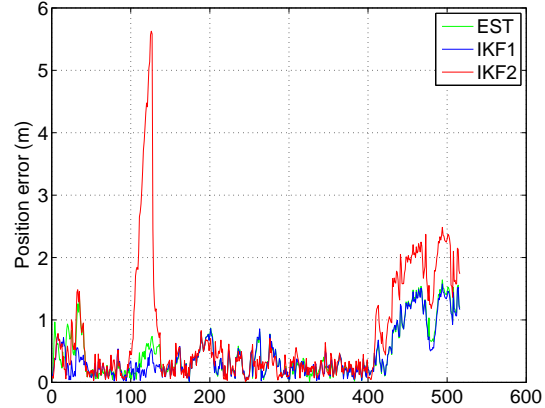

Figure 5: Trends for position error as a function of time.

components close to the correct ones. Figure 7 also reveals that, after an initial transitory phase where EST incurs in sporadic estimation errors, it eventually matches IKF1 performance. Initial errors are easily accountable to the initial largely wrong values chosen to initialize the covariance parameters being estimated. Once EST settles around the right values (time $t = 100$), its performance is identical to IKF1.
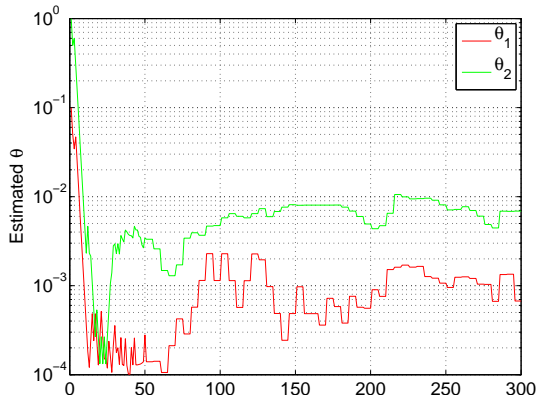

Figure 6: Trend of the estimated θ values produced by EST. The true values are $\theta_1 = 10^{-2}$ and $\theta_2 = 10^{-3}$
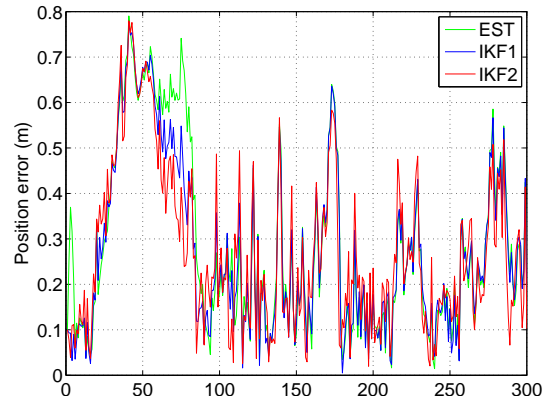

Figure 7: Trends for position error as a function of time.

4.2 Real data: estimating the measurement noise covariance using a mobile platform

Finally, we have tested the proposed algorithm with data collected from a mobile robot moving in an indoor environment. The robot moves according to the model expressed by Eq. 12. No ground truth is available for the path followed by the robot, so in this section we will compare the performance of EST, where the covariance of the transition noise is assumed known while the measurement noise statistics need to be estimated, against IKF1 which is still provided with the correct autocovariance matrices.

The robotic platform used is an ActiveMedia P3AT, as shown in figure 8. The distance sensor was

Figure 8: The robotic platform used to validate the proposed algorithm. On the left one marker used for localization is visible.

implemented using a camera with a resolution of 320×240 pixels mounted atop the robot. Landmarks have been implemented using distinctive, colored markers with a known size and hanging at known locations a fixed height (see Figure 8). In order to increase the number of landmarks visible by the robot during the experiment, the camera was mounted on a servo so that it can be directed towards a landmark even if it is not in front of the robot. Knowing the real height of the markers (H) and the focal length in pixels f of the camera, the distance from a perceived marker can be estimated by the following formula

$$d = \frac{H \cdot f}{h}$$

where h is the height in pixels of the marker in image plane. This simplistic approach ignores issues related to distortion and similar problems, but this aspect is not relevant here because the goal of the experiment is not to accurately measure distances using a camera. After having collected 200 measures, two parameters p_0 and p_1 were experimentally determined so that by correcting the returned distance as follows

$$z = (1 - p_1)d + p_0$$

the error of the distance returned by this sensor is appropriately modeled by a Gaussian distribution with 0 mean and covariance $\zeta = 0.538m^2$ (see [26] for details). This value should be interpreted as *indicative* and not as ground truth because inevitable changes in the light conditions of the environment will lead to a difference in the actual value of ζ.

During the data collection session the robot and the tiltable camera were remotely operated by a human so that sufficient landmarks could be observed. Figure 9 shows that the proposed algorithm eventually estimates a value of ζ converging towards the known true value, notwithstanding that EST started with an initial estimate for ζ off by a factor of 1000. Figure 10 instead displays the paths estimated by EST and IKF1[7]. Since during the real robot run it was not possible to collect data concerning the real robot

---

[7]The reader will notice that that landmarks are located as in Fig. 3. This is no coincidence, as the simulated environment was designed to replicate our lab testbed.

pose, the figure outlines that the estimates provided by EST and IKF1 are close throughout the estimate task.
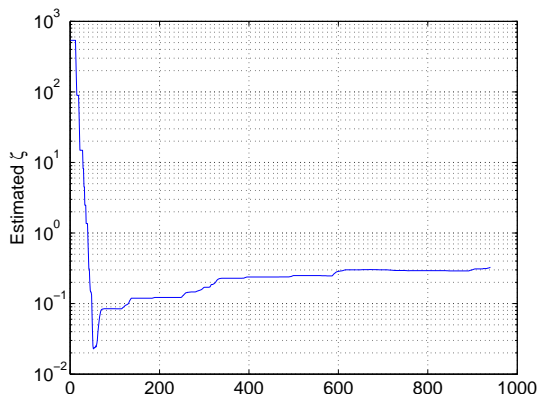

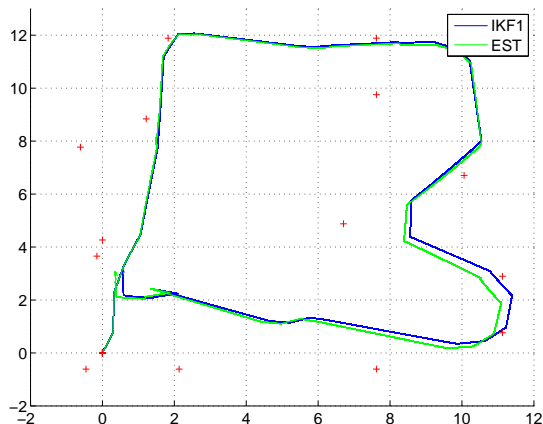Figure 9: Trend of the estimated θ values produced by EST.

Figure 10: Paths estimated by EST and IKF1.

5 Conclusions

We have introduced an algorithm able to estimate in an online manner the covariance parameters of the measurement and transition noise regulating the dynamics a nonlinear dynamic system. The algorithm relies upon the propagation of the marginal posterior of the unknown parameters over time by extending the equations underlying the iterated Kalman filter. As outlined in [27], the approach can also be related to Tikhonov regularization, and the reader is referred to the details provided therein for a more elaborate discussion. The effectiveness of the new numerical scheme has been assessed using both real and simulated data regarding a robot localization problem.

Acknowledgments

REFERENCES

[1] P. Maybeck, *Stochastic models, estimation and control.* Academic Press, 1979.

[2] R. Stengel, *Optimal control and estimation.* Dover, 1994.

[3] B. D. O. Anderson and J. B. Moore, *Optimal Filtering.* Englewood Cliffs, N.J., USA: Prentice-Hall, 1979.

[4] M. Davidian and D. Giltinan, *Nonlinear Models for Repeated Measurement Data.* Chapman & Hall, 1995.

[5] S. Searle, G. Casella, and C. McCulloch, *Variance Components*. New York: Wiley, 1979.

[6] N. Kristensen, H. Madsen, and S. Jørgensen, "Parameter estimation in stochastic grey-box models," *Automatica*, vol. 40, no. 225-237, 2004.

[7] L.-L. Fu, I. Fukumori, and R. Miller, "Fitting dynamic models to the geostat sea level observations in the tropical pacific ocean: Part II: A linear, wind-driven model," *Journal of Physical Oceanography*, vol. 23, pp. 2162–2181, 1993.

[8] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. MIT Press, 2006.

[9] J. Leonard and H. Durrant-Whyte, "Mobile robot localization by tracking geometric beacons," *IEEE Transaction on Robotics and Automation*, vol. 7, no. 3, pp. 376–382, 1991.

[10] R. Simmons and S. Koenig, "Probabilistic robot navigation in partially observable environments," in *Proceedings of IJCAI*, Los Angeles, pp. 1080–1087, 1995.

[11] S. Thrun, D. Fox, and W. Burgard, "A probabilistic approach to concurrent mapping and localization for mobile robots," *Machine Learning*, vol. 31, pp. 29–53, 1998. also appeared in Autonomous Robots 5(2):253–271 (joint issue).

[12] S. Thrun, D. Fox, W. Burgard, and F. Dellaert, "Robust MonteCarlo localization for mobile robots," *Artificial Intelligence*, vol. 128, no. 1-2, pp. 99–141, 2001.

[13] C. Kwok, D. Fox, and M. Meila, "Real-time particle filters," *Proceedings of the IEEE*, vol. 92, no. 3, pp. 469–484, 2004.

[14] I. Rekleitis, G. Dudek, and E. Milios, "Accurate mapping of an unknown world and online landmark positioning," in *Vision Interface*, pp. 455–461, 1998.

[15] I. Rekleitis, G. Dudek, and E. Milios, "Multi-robot cooperative localization: a study of trade-offs between efficiency and accuracy," in *Proceedings of the IEEE/RJS International Conference on Intelligent Robots and Systems*, Lausanne, pp. 2690–2695, 2002.

[16] A. Howard, M. Mataric, and G. Sukhatme, "Localization for mobile robot teams using maximum likelihood estimation," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Lausanne, pp. 434–439, 2002.

[17] S. Roumeliotis and G. Bekey, "Distributed multirobot localization," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 781–795, 2002.

[18] A. Mourikis and S. Roumeliotis, "Performance analysis of multirobot cooperative localization," *IEEE Transactions on robotics*, vol. 22, no. 4, pp. 666–681, 2006.

[19] A. Mourikis and S. Roumeliotis, "Optimal sensor scheduling for resource-constrained localization of mobile robot formations," *IEEE Transactions on Robotics*, vol. 22, no. 5, pp. 917–931, 2006.

[20] A. H. Jazwinski, *Stochastic Processes and Filtering Theory*. Academic Press, New York, 1970.

[21] D. Luenberger, *Linear and nonlinear programming*. Kluwer, 2003.

[22] B. Bell and F. Cathey, "The iterated Kalman filter update as a Gauss-Newton method," *IEEE Trans. on Automatic Control*, vol. 38, no. 2, pp. 294–297, 1993.

[23] B. Bell, "The marginal likelihood for parameters in a discrete gauss-markov process," *IEEE Trans. on Signal Processing*, vol. 48, no. 3, pp. 870–873, 2000.

[24] B. Bell, "Approximating the marginal likelihood estimate for models with random parameters," *Appl. Math. Comput.*, vol. 119, pp. 57–75, 2001.

[25] B. Bell and G. Pillonetto, "Estimating parameters and stochastic functions of one variable using nonlinear measurements models," *Inverse Problems*, vol. 20, no. 3, pp. 627–646, 2004.

[26] G. Erinc, G. Pillonetto, and S. Carpin, "Online estimation of variance parameters: experimental results with applications to localization," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Nice, pp. 1890–1895, 2008.

[27] G. Pillonetto and S. Carpin, "Multirobot localization with unknown variance parameters using iterated kalman filter," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, San Diego, pp. 1709–1714, 2007.