Online estimation of variance parameters:
experimental results with applications to localization

Gorkem Erinc, Gianluigi Pillonetto, Stefano Carpin

Abstract— This paper presents an experimental validation of an online estimation algorithm we recently investigated theoretically. One of the peculiar characteristics of the approach we propose is the ability to perform an online estimation of the variance parameters that regulate the dynamics of the nonlinear dynamical model used. The approach exploits and extends classical iterated Kalman filtering equations by propagating an approximation of the marginal posterior of the unknown variances over time. The method has been previously used to model and solve a localization task for multiple robots equipped only with a sensor returning mutual distances. In this paper we present a first experimental validation of the algorithm that complements and confirms our initial promising theoretical findings. Our current implementation relies on a sensor returning distance estimates based on a simple image processing algorithm. Such sensor is inherently and intentionally noisy, and in this study we show that our technique is capable of appropriately estimating the variance describing the noise affecting this sensor. We conclude proving experimentally that the procedure we present ensures a performance comparable to similar algorithms that require significantly more a priori information.

I. INTRODUCTION

Tomorrow's society will see the deployment of more and more networked robots interacting among themselves and with low cost devices dispersed in the environment. The area of multi-robot systems steadily grew in the last fifteen years, and multi-robot solutions appear to be the winning choice for many different tasks [1]. More recently, a vigorous evolution in sensor networks have opened the doors to more intriguing scenarios where robots no longer just interact with other robots, but exchange information also with sensors scattered in the environment. These sensors may have been a priori deployed, or even positioned by the robot themselves while operating. The interaction between research in multi-robot systems and sensor networks appears to be one of the most exciting ones for the near future. These considerations are the basis for the research generating the results presented in this manuscript. In particular, we aim to develop multi-robot systems capable of completing complex tasks while operating in unstructured environments not offering features appropriate to solve the localization task. In this case, it is appealing to envision that robots will instead deploy a suitable supporting sensor network while operating, or that robots themselves will constitute the localization infrastructure. The reader should note that in this line of research we do not assume the availability of sensorial payload suitable to solve the full pose estimation based on SLAM algorithms producing two dimensional or three dimensional maps. Eventually, our goal is to use distance sensors based on time of flight like those being anticipated in [2]. When these sensors will be available, they will be deployed in the environment at run time, and treated as consumable components. Therefore, it would be not viable to invest significant time before hand to derive an accurate statistical model describing their performance. It is rather much more appealing to infer this statistical description online, while the robot uses these sensors. Starting from this scenario, we have recently developed an estimation algorithm compatible with these assumptions [3]. Its main strength is in the ability to estimate online the parameters characterizing the variance of the transition and measurement noise. We have already shown that such method can be used to solve a multi-robot localization problem where robots are only equipped with sensors returning mutual distances. We assumed that at every point in time some of the robots move whilst the others stand still in order to serve as fixed landmarks for the moving ones. A thorough theoretical investigation has shown that the method is capable of reliably estimating the covariance matrices of transition or measurement noise (but not both at the same time). Results based on extensive simulations also suggested that the algorithm behaves favorably when compared to the extended Kalman filter (EKF) and the iterative Kalman filter (IKF), even though it requires much less a priori information. In this paper we provide the first experimental validation of the proposed technique. Accordingly to our future plan involving deployable landmarks, we here exclusively focus on estimating the covariance of the measurement noise. In particular, we have implemented a vision based sensor that returns distance to visual landmarks based on a simple image processing algorithm. Our aim is to prove that the technique we have theoretically investigated is viable also from a practical point of view when implemented on real world systems. As detailed in the subsequent sections, it turns out that this is the case. The remaining of the paper is organized as follows. Related literature is presented in section II. The estimation problem and algorithm is shortly summarized in section III. Our original general purpose formulation is therein reworked in order to take into account the specific scenario considered. The experimental setup is described in section IV. Finally, conclusions and future work are proposed in section V.

Gorkem Erinc and Stefano Carpin are with the School of Engineering, University of California, Merced, USA. Gianluigi Pillonetto is with the Department of Information Engineering, University of Padova, Italy.

II. Related work

Localization is one of the most widely studied topics in mobile robotics. Since in our experimental setup we focus on mobile robots moving in planar environments, we restrict our discussion to techniques aimed 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. [4] offering an up to date description of the most commonly used probabilistic techniques adopted to solve this problem reveals that most of the published literature in the field resorts to some implementation of the Bayes filter.

In multi-robot localization, relevant for our experimental setup, the aim is to localize n robots moving in a shared environment and equipped with some sensors to detect not only the environment but also their mutual positions. While this problem can be solved by running n independent instances of the formerly described methods, or a single filter estimating the $3n$ components of the composed state, research has been done towards the development of *cooperative localization*. The term was introduced by Rekleitis et al. that first considered two robots using visual sensors to reduce odometry errors [5], and later on considered more than two robots and different sensors [6]. Howard et al. also considered the cooperative localization problem, using a maximum likelihood estimator instead [7]. Roumeliotis and Bekey [8] proposed to split the overall EKF[1] into n smaller communicating filters allocated on the n robots in the team. More recently, Mourikis and Roumeliotis [9], [10] investigated the intrinsic limitations of these approaches. In particular they relate the localization accuracy to available resources, like the amount of exchanged information, sensing frequency, and so on. When the environment offers no features, two strategies can be used in order to install a suitable infrastructure. When more robots are available, some of them could serve as artificial detectable features. This idea was pioneered by Kurazume, Hirose et al. [11], and by Grabowski and Khosla [12]. In these papers some robots in the team stay stationary and serve as landmarks, while the remaining ones move and use these artificial landmarks to localize themselves. From time to time, roles are swapped, and then the whole team can move. The above idea, however requires more than one robot and implies a suboptimal use of resources, because some of the robots need just to stay stationary. To overcome these limitations Kleiner et al. [13] recently illustrated a system where robots disperse RFID tags in the environment that are later on used as detectable landmarks. Similar studies were performed by Batalin and Sukhatme, who presented various papers where single or multiple robots interact with sensor networks positioned during their operation [14]. Other examples of interactions between robots and sensor networks were reported in [15], where multiple robots move towards areas with low sensor density in order to increase sensor localization accuracy.

III. Notation, problem definition, and the estimation algorithm

A. Notation and model

We here summarize the estimation algorithm we have formerly investigated theoretically in [3]. The reader is referred to details therein for a more complete and general description, as well as a formulation not restricted to robot localization. Let us first define the proper notation for setting up a Kalman filter based framework. Let p be the dimension of a parameter space, n the dimension of the state space, m the dimension of the measurement space, and q the dimension of the control space. $\chi \in \mathbf{R}^p$ is a parameter vector, $\mathbf{x}_0 \in \mathbf{R}^n$ is the initial value of state vector, and $z_k \in \mathbf{R}^m$ is the measurement vector. Finally, let $h_k : \mathbf{R}^n \to \mathbf{R}^m$ be the expected value of z_k given x_k, and $g_k : \mathbf{R}^n \times \mathbf{R}^q \to \mathbf{R}^n$ the expected value of x_k given x_{k-1} and input u_k. Let $R_k : \mathbf{R}^p \to \mathbf{R}^{m \times m}$ be the autocovariance of z_k given x_k and $Q_k \in \mathbf{R}^{n \times n}$ be the autocovariance of x_k given x_{k-1} and u_k. We assume that the autocovariance matrices R_k depend on a vector $\chi \in \mathbf{R}^p$ whose components may be unknown[2]. The strength of the algorithm we propose relies in its ability to determine these dependences online, while estimating the state $\mathbf{x}$. Let $\mathcal{N}(\mu, \Sigma)$ indicate a Gaussian random vector with mean μ and covariance matrix Σ. Finally, if $W \in \mathbf{R}^{s \times s}$ is a symmetric positive definite matrix, and $v \in \mathbf{R}^s$, let $v^T W v = \|v, W\|^2$. Assume there are N robots in the system. We here suppose that each robot is a differential drive system, i.e. the state of a single robot is $\mathbf{x}^i = [x^i \ y^i \ \theta^i]^T$ where x^i and y^i are the coordinates of a fixed point, for example the middle point in the wheels axle, and θ^i is the robot heading. Moreover, robot i can receive input $u^i = (v^i, \omega^i)$, where v^i is the translational velocity and ω^i is the rotational velocity. Based on these assumptions, it follows that $n = 3N$ and $q = 2N$. At time k the system is in state $\mathbf{x}_k$, receives input u_{k+1}, and transitions into state

$$\mathbf{x}_k = g_k(\mathbf{x}_{k-1}, u_k) + \nu_k.$$

Being more specific we write $g_k = [(g_k^1)^T \ \ldots (g_k^N)T]^T$, where each g_k^i indicates how the state of robot i evolves when control u_k^i is applied. The transition function g_i for robot i can be written as follows (see e.g. [4] for an analytical derivation of the following relationships):

$$\begin{cases} x_k^i = x_{k-1}^i + (-\frac{v_k^i}{\omega_k^i} \sin \theta_{k-1}^i + \frac{v_k^i}{\omega_k^i} \sin(\theta_{k-1}^i + \omega_k^i \Delta t)) \\ y_k^i = y_{k-1}^i + (\frac{v_k^i}{\omega_k^i} \cos \theta_{k-1}^i - \frac{v_k^i}{\omega_k^i} \cos(\theta_{k-1}^i + \omega_k^i \Delta t)) \\ \theta_k^i = \theta_{k-1}^i + \omega_k^i \Delta t \end{cases}$$

An obviously simplified relationship holds when $\omega_k^i = 0$, i.e. when the robot moves forward without turning. The transition noise is Gaussian, i.e. $\nu_k \in \mathcal{N}(0, Q_k)$ where Q_k is a block diagonal matrix with N blocks whose i-th block is defined as follows:

---

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

[2] in general it is also possible to assume that matrices Q_k depend on χ, but this is not relevant for the scope of the current paper. The reader is referred to our formerly cited publication for this extension.

$$Q_k^i = V_k^i M_k^i (V_k^i)^T + L_k^i$$

where

$$M_k^i = \begin{pmatrix} \alpha_1(v_k^i)^2 + \alpha_2(\omega_k^i)^2 & 0 \\ 0 & \alpha_3(v_k^i)^2 + \alpha_4(\omega_k^i)^2 \end{pmatrix}$$

$$V_k^i = \frac{\partial g_k^i(\mathbf{x}_{k-1}^i, u_k^i)}{\partial u_k^i}(\mathbf{x}_{k-1}^i, u_k^i)$$

$$L_k = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \alpha_5 v_k^2 + \alpha_6 \omega_k^2 \end{pmatrix}$$

The various α_js are constants determined off line and are the same for all robots. For the sensor model we assume that each robot is equipped only with a device returning the distance to other robots. Indicating with z_k the reading returned at time k, let us indicate with $z_k^{i,j}$ the distance between robot i and robot j. z_k has dimension $m = N(N-1)$ because it contains the mutual distances between all robots in the system. The entry in z_k returning the distance between robot i and robot j is

$$z_k^{i,j} = \sqrt{\left(x_k^i - x_k^j\right)^2 + \left(y_k^i - y_k^j\right)^2} + \zeta_k^{i,j} \tag{1}$$

where $\zeta_k^{i,j}$ are the components of an m dimensional random vector $\zeta_k \in \mathcal{N}(0, R_k(\chi))$, i.e. matrix R_k may depend on the unknown parameters in χ.

B. Problem definition

Our problem consists of estimating on-line both $\mathbf{x}_k$ for each robot, and the (possibly) unknown components of χ starting from output data $\{z_k\}$. Equation 1 outlines that we assume that robots are equipped only with sensors returning mutual distances. Notice also that by not assuming that robots are equipped with proprioceptive sensors a more challenging scenario is set for the algorithm we are proposing. If additional sensors are available, such information can be easily integrated yielding a simpler problem where more experimental evidence is provided. In our original formulation we assumed that each robot in the system could behave in two mutually exclusive ways: *landmark* or *moving*. When a robot acts as landmark, its translational and rotational velocities are 0, and it experiences no transition noise, i.e. $\nu_k = 0$. Otherwise, when a node acts as *moving*, the covariance of the random vector ν_k is given by the matrix Q_k, defined above.

It is assumed known whether a node is moving or a landmark. As far as R_k is concerned, no particular structure is stated here. Notice that if both robot i and robot j act as landmarks, whose positions are assumed known, then matrices R_k do not depend on unknown components of χ, and the measurements of their distance become irrelevant and can be removed from the model. In addition, depending on the particular problem under study, not all the measurements $\{z_k\}$ listed above could be available, as also specified later on.

C. Estimation algorithm

Building upon the Kalman filter framework, the estimation algorithm is based on two standard steps: prediction and correction based on measurement. The reader will note that the latter introduces significant novelties.

1) Prediction: In the following we use the notation $\hat{\chi}_{k-1}$ to indicate the estimate of χ at instant $(k-1)$ while $\hat{\Sigma}_{\chi,k-1}$ denotes the covariance matrix of the associated error. The estimate of $\mathbf{x}_{k-1}$ obtained exploiting measurements collected up to instant $(k-1)$ is instead denoted as $\hat{\mathbf{x}}_{k-1|k-1}$ with error covariance given by $\hat{\Sigma}_{k-1|k-1}$. It is worth stressing that both $\hat{\mathbf{x}}_{k-1|k-1}$ and $\hat{\Sigma}_{k-1|k-1}$ are quantities which are not interpreted as dependent on χ in this phase of the algorithm. Linearized dynamics of the system around the current estimate are used to perform the time-update. Let

$$G_k = \frac{\partial g_k(\mathbf{x}_{k-1}, u_k)}{\partial \mathbf{x}_{k-1}}(\hat{\mathbf{x}}_{k-1|k-1}, u_k)$$

then

$$g(\mathbf{x}_{k-1}, u_k) \approx g_k(\hat{\mathbf{x}}_{k-1|k-1}, u_k) + \tag{2}$$
$$+ G_k \cdot (\mathbf{x}_{k-1} - \hat{\mathbf{x}}_{k-1|k-1}) + \nu_k$$

From equation 2, it follows that $\mathbf{x}_k$ can be approximated as a Gaussian with mean and covariance given by the following

$$\hat{\mathbf{x}}_{k|k-1} = g_k(\hat{\mathbf{x}}_{k-1|k-1}, u_k)$$
$$\hat{\Sigma}_{k|k-1} = G_k \hat{\Sigma}_{k-1|k-1} G_k^T + Q_k$$

2) Correction based on measurement: We now need a better estimate for $\mathbf{x}_k$ and χ, as well as the corresponding covariance matrices of the error affecting these estimates, starting from z_k. It is easy to show that assuming that $\mathbf{x}_k$ is Gaussian with mean $\hat{\mathbf{x}}_{k|k-1}$ and covariance $\hat{\Sigma}_{k|k-1}(\chi)$, then for known χ and z_k, the maximum a posteriori estimate of $\mathbf{x}_k$ minimizes the following objective function:

$$l(\mathbf{x}_k, z_k|\chi) = \frac{1}{2}\log\det(2\pi R_k(\chi))$$
$$+ \frac{1}{2}\log\det(2\pi\hat{\Sigma}_{k|k-1}(\chi)) + \frac{1}{2}\|z_k - h(\mathbf{x}_k), R_k(\chi)^{-1}\|^2$$
$$+ \frac{1}{2}\|\mathbf{x}_k - \hat{\mathbf{x}}_{k|k-1}, \hat{\Sigma}_{k|k-1}(\chi)^{-1}\|^2$$

where $l(\mathbf{x}_k, z_k|\theta)$ is the minus log of the joint density of $\mathbf{x}_k$ and z_k conditioned on χ. Recalling that the iterated Kalman filter (IKF) update is a Gauss-Newton method, one can use IKF to minimize the $l(\cdot)$ objective function. Thus, the minimizer of $l(.)$ can be achieved by defining inductively the sequences $^i\mathbf{x}$ and $^i\Sigma$ as follows. Let $^0\mathbf{x} := \hat{\mathbf{x}}_{k|k-1}$ and $^0\Sigma := \hat{\Sigma}_{k|k-1}(\chi)$, then

$$^{i+1}\mathbf{x} = \hat{\mathbf{x}}_{k|k-1} + K_i\left(z_k - h(^i\mathbf{x}) - H_i\left(\hat{\mathbf{x}}_{k|k-1} - {}^i\mathbf{x}\right)\right)$$
$$^{i+1}\Sigma = (I - K_iH_i)\hat{\Sigma}_{k|k-1}(\chi)$$

where

$$H_i = \frac{\partial h(\mathbf{x})}{\partial \mathbf{x}}(^i\mathbf{x})$$

$$K_i = \hat{\Sigma}_{k|k-1}(\chi)H_i^T\left(H_i\hat{\Sigma}_{k|k-1}(\chi)H_i^T + R_k(\chi)\right)^{-1}$$

After computing a sufficient number of iterations to reach convergence, values of ${}^i\mathbf{x}$ and ${}^i\Sigma$ provide the updated estimate $\hat{\mathbf{x}}_{k|k}(\chi)$ and the covariance matrix of the error $\hat{\Sigma}_{k|k}(\chi)$. However, both these quantities depend on χ. Thus, the question now arises as how to estimate the unknown components of χ. Let $\pi(\mathbf{x}_k, z_k|\chi)$ denote the joint density for $\mathbf{x}_k$ and z_k conditioned on χ, i.e.

$$\pi(\mathbf{x}_k, z_k|\chi) = \exp\left(-l(\mathbf{x}_k, z_k|\chi)\right)$$

Let instead $\pi(z_k|\chi)$ be the marginal likelihood of χ, i.e.

$$\pi(z_k|\chi) = \int \pi(\mathbf{x}_k, z_k|\chi)d\mathbf{x}_k \qquad (3)$$

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

$$\hat{\chi} = \mathrm{argmin}_\chi - \log\left[\pi(z_k|\chi)\pi_{k-1}(\chi)\right]$$

subject to nonnegative constraints for the components of χ. The problem now is that, due to the nonlinear nature of function h, evaluation of $\pi(z_k|\chi)$ for a given χ requires solution of an integral in equation 3 which in general is analytically intractable. It is shown now how computations performed by IKF provide an approximation for such an integral. To this aim, consider the affine approximation of $l(\mathbf{x}_k, z_k|\chi)$ for $\mathbf{x}_k$ near y which is defined by

$$\tilde{l}(\mathbf{x}_k, y, z|\chi) = \frac{1}{2}\log\det(2\pi R_k(\chi))$$
$$+ \frac{1}{2}\log\det(2\pi\hat{\Sigma}_{k|k-1}(\chi))$$
$$+ \frac{1}{2}\|z_k - h(y) - h'(y)(\mathbf{x}_k - y), R_k(\chi)^{-1}\|^2$$
$$+ \frac{1}{2}\|\mathbf{x}_k - \hat{\mathbf{x}}_{k|k-1}, \hat{\Sigma}_{k|k-1}(\chi)^{-1}\|^2$$

The determinant of the Hessian of $\tilde{l}$ with respect to $\mathbf{x}_k$ is easily obtained and reads as follows

$$\det\left[\partial^2_{\mathbf{x}_k}\tilde{l}(\mathbf{x}_k, y, z_k|\chi)\right] = \qquad (4)$$
$$\det\left[(h'(y))^T R_k^{-1}(\chi)h'(y) + \hat{\Sigma}_{k|k-1}^{-1}(\chi)\right]$$

The information matrix (expected Hessian) approximation for the marginal likelihood $\pi(z_k|\chi)$ is denoted by $\tilde{\pi}(z_k|\chi)$ and given by

$$\det\left\{\partial^2_{\mathbf{x}_k}\tilde{l}(\mathbf{x}_k, \hat{\mathbf{x}}_{k|k}(\chi), z_k|\theta)/(2\pi)\right\}^{-1/2} \qquad (5)$$
$$\exp\left\{-l(\hat{\mathbf{x}}_{k|k}(\chi), z_k|\chi)\right\}$$

In view of equations 4 and 5, one can notice that such an approximation requires $\hat{\mathbf{x}}_{k|k}(\chi)$ and $h'(\hat{\mathbf{x}}_{k|k}(\chi))$ which represent quantities returned by IKF. Said in other words, for every χ value, IKF can be used to evaluate an objective function whose optimization provides the estimate of χ. Thus, we are now in a position to define the estimate of

χ, obtained after seeing data up to instant k, as the solution to the problem

$$\hat{\chi}_k = \mathrm{argmin}_\chi - \log\left[\tilde{\pi}(z_k|\chi)\pi_{k-1}(\chi)\right] \qquad (6)$$

while $\hat{\Sigma}_{\chi,k}$ is given by

$$\hat{\Sigma}_{\chi,k} = \left(-\partial^2_\chi \log\left[(\tilde{\pi}(z_k|\hat{\chi}_k)\pi_{k-1}(\hat{\chi}_k)]\right)\right)^{-1} \qquad (7)$$

and can be calculated numerically. This completes the update for parameter χ. Finally, the overall measurement update is completed by setting estimate of $\mathbf{x}_k$ to $\hat{\mathbf{x}}_{k|k}(\hat{\chi}_k)$ with associated covariance matrix of the error given by $\hat{\Sigma}_{k|k}(\hat{\chi}_k)$.

IV. EXPERIMENTAL RESULTS

In order to evaluate numerically the potential of the proposed technique, we have deployed a system aimed to estimate online the dependency between matrix R_k and vector χ, i.e. a system capable of estimating online the variance of the noise affecting the sensor returning mutual distances between robots. To keep the experimental setup easy, we consider a simplified scenario where just one robot is moving and all the others are acting as landmarks. In this way we can isolate and study a single problem, namely the online estimation without the risk of taking into account additional sources of uncertainty. The moving robot used is the P3AT mobile platform. Due to temporary space constraints, and to the impossibility to deploy a large team of robots, landmark robots are replaced by stationary artifacts, as clarified in the following subsection.

A. Visual guided distance measurement

Distance computation is provided by a single camera mounted on the robot. The camera has a resolution of 320×240 pixels, and is mounted on a Phidgets HS322 servo, so that it can rotate 180 degrees (see figure 1). In order

Fig. 1. The robot used for the experimental validation. Two iRobot Create platforms acting as static landmarks can be seen. Visual landmarks can be observed on the poles mounted on the stationary robots.

to extract distance estimation, carefully engineered markers are used. Each marker is about 18 centimeters high and

consists of five stripes where each stripe is either red, green or blue as shown in figure 1. While the robot navigates, captured images are processed to identify the region of interest by means of a probabilistic color detection technique applied in HSV color space where parameters of the color probability density functions are tuned experimentally. A template matching algorithm is then performed on marker candidates to eliminate false positive readings. Additionally, in order to increase the robustness in marker detection a confidence measure based on the color distributions along the marker is introduced. Only the marker candidates providing a confidence measure above a preset threshold is accepted as valid markers. Once the marker is detected, the color code is decoded for the ID of the maker and its relative distance from the camera is calculated from the size of the marker in image space. The distance calculation ignoring the minor effects of distortion is realized by the following equation.

$$d = \frac{H \cdot f}{h} \tag{8}$$

where d is the distance between the camera and the marker, H is the height of the real marker, f is the focal length in pixel units and h is the height of marker in image plane.

As experimentally studied this method entails a growing estimation error while the distance between the marker and camera gets bigger. This problem is inherently caused by the increasing quantization error due to the mapping of real world into the finite number of pixels in image plane. Experimental data collected by using this straightforward model under different lighting conditions and distances showed a linear trend in the estimation error. Therefore the returned distance is not the one produced by equation 8 but rather

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

where the coefficients $p1$ and p_0 have been obtained through a linear interpolation of the error profile. The resulting error between the estimation and the ground truth data shows a Gaussian distribution as presented in figure 2. We are fully aware that this distance estimation method is not accurate and can be greatly improved, but this is not the scope of this research. Our goal is instead to verify whether the error dynamics exhibited by this sensor can be determined online by the estimation algorithm we propose.

B. Comparisons of different localization algorithms

As we assume that only one robot acts as moving robot, and all the others act as landmarks, we are in fact dealing with a well studied instance of the localization problem. Therefore two widely used approaches, namely EKF and IKF, provide solid reference points to measure the value of our algorithm. The main difference to be considered while interpreting the results we provide, is in the different amount of a priori information required by these algorithms. In particular EKF, IKF and the algorithm we propose require full knowledge of the matrices Q_k characterizing the transition noise ν_k. Throughout the experimental trials we will illustrate, the same Q_k was used for all estimation
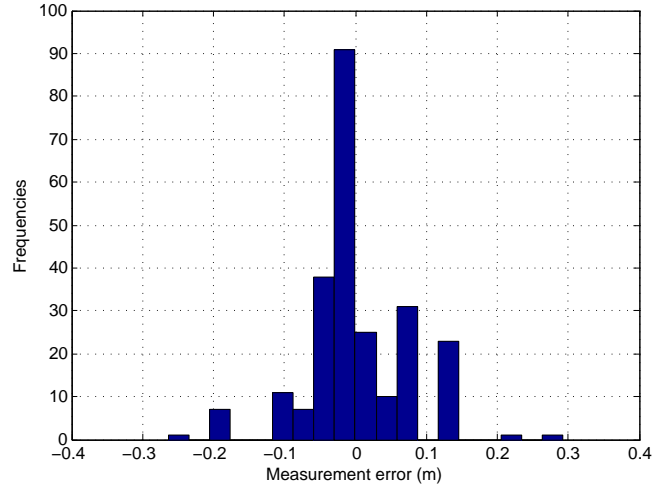

Fig. 2. Distribution of the distance measurement errors for 200 independent measures taken for the distances in the range of 0.25m to 4.5m under different conditions

algorithms. However, while IKF and EKF require also full knowledge of the matrix R_k characterizing measurement noise, our approach starts without knowing R_k and recovers the true value online by performing the noise covariance estimation at each cycle for the first 10 time steps and then every 10 cycle while also estimating the state at the same time. Each estimation step takes 0.56s in average on a AMD Sempron 1.8 GHz with 512MB RAM running SuSE 10.1. In this implementation 8 update iterations are carried out in IKF and in the IKF step of the proposed method. All these algorithms have been implemented in C++ and process the same data collected by the robot while moving among the landmarks. Figures 3 and 4 contrast the three estimation algorithms during the prototypical test run. Figure 3 shows the trajectory estimated by the three approaches, while figure 4 illustrates the estimated robot heading. The above figures clearly show that the estimator we are proposing produces results comparable to IKF and EKF but requiring much less information. Finally, we illustrate how during the localization process the variance of the noise characterizing the sensor being used is estimated online. As we have only one sensor, the online estimation process reduces to estimate the covariance of the error described in figure 2. We determined offline that such error distribution can be described by a Gaussian with 0 mean and variance 0.00538. This specific value was provided to IKF and EKF but not to our estimator. Figure 5 shows the trend of the estimated value during the run. The algorithm is intentionally started with 1,000,000 times of the true value and after few iterations it converges down and settles towards a value of 0.00553. Similar trends were observed throughout the trials we performed.

V. CONCLUSIONS

In this paper we have presented an algorithm that performs the online estimation of unknown variance parameters characterizing measurement noise. The methodology we have presented has broader applicability and can be used also to
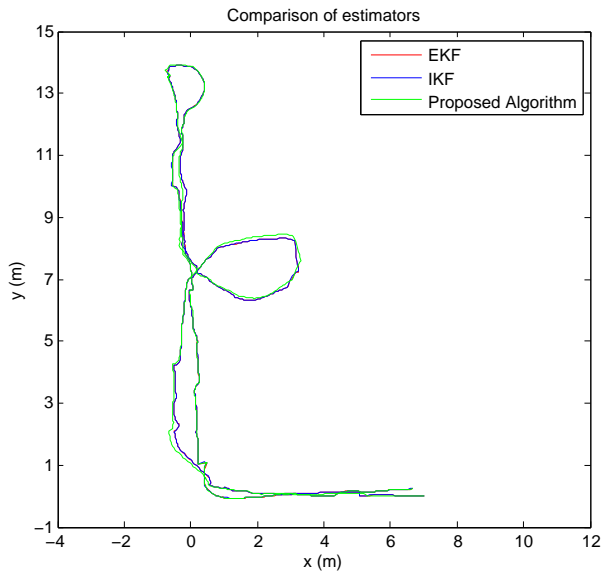
Fig. 3. Comparison between the three trajectories estimated by EKF, IKF and the algorithm described in the paper.
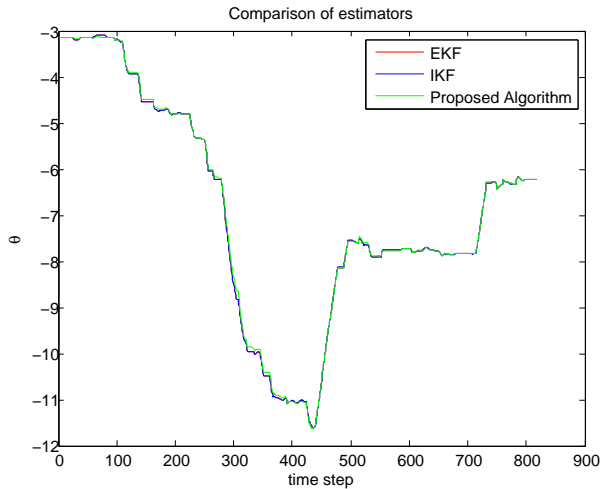

Fig. 4. Comparison of the estimated robot heading (θ) by the three algorithms versus the iteration number

estimate the transition noise covariance in a similar fashion, although this aspect has not been addressed in this paper. We have experimentally demonstrated the validity of the method we proposed by using an error prone distance sensor based on a simple image processing algorithm. The algorithm produces estimates comparable to the well known IKF and EKF algorithms, but requires much less a priori information. The long term goal we are working to, is the development of rescue robots capable of performing localization tasks by just using forthcoming sensors that return distance measures based on time of flight. Our vision is to deploy multi-robot systems that install a suitable localization infrastructure while operating, thus relieving the necessity to solve the whole SLAM problem. Moreover, such infrastructure could also be used to support the operation of human first responders entering the disaster scenario after robots deployed these
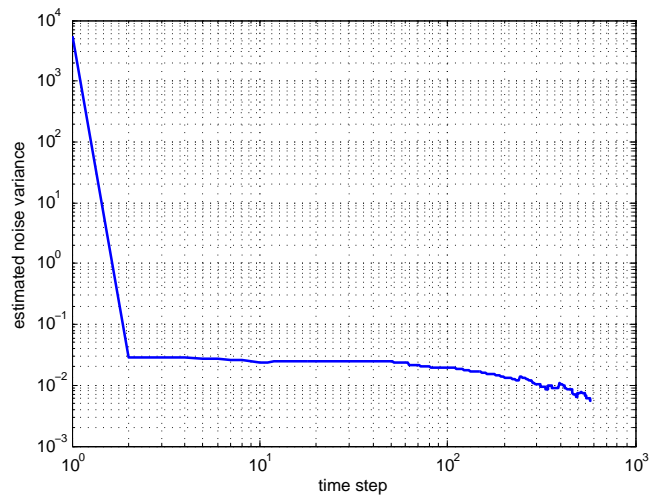

Fig. 5. Trend of the covariance estimation

sensors.

REFERENCES

[1] T. Arai, E. Pagello, and L. Parker, "Guest editorial advances in multirobot systems," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 655–661, 2002.
[2] S. Lanzisera, D. Lin, and K. Pister, "RF time of flight ranging for wireless sensor network localization," in *Workshop on Intelligent Solutions in Embedded Systems*, 2006.
[3] 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*, 2007, pp. 1709–1714.
[4] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. MIT Press, 2006.
[5] I. Rekleitis, G. Dudek, and E. Milios, "Accurate mapping of an unknown world and online landmark positioning," in *Vision Interface*, 1998, pp. 455–461.
[6] ——, "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*, 2002, pp. 2690–2695.
[7] 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*, 2002, pp. 434–439.
[8] S. Roumeliotis and G. Bekey, "Distributed multirobot localization," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 781–795, 2002.
[9] A. Mourikis and S. Roumeliotis, "Performance analysis of multirobot cooperative localization," *IEEE Transactions on robotics*, vol. 22, no. 4, pp. 666–681, 2006.
[10] ——, "Optimal sensor scheduling for resource-constrained localization of mobile robot formations," *IEEE Transactions on Robotics*, vol. 22, no. 5, pp. 917–931, 2006.
[11] R. Kurazume and S. Hirose, "An experimental study of cooperative positioning system," *Autonomous Robots*, vol. 8, no. 1, pp. 43–52, 2000.
[12] R. Grabowski and P. Kohsla, "Localization techniques for a team of small robots," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2001, pp. 1067–1072.

[13] A. Kleiner, J. Prediger, and B. Nebel, "Rfid technology-based exploration and slam for search and rescue," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Tobots and Systems*, 2006.

[14] M. Batalin and G. Sukhatme, "Efficient exploration without localization," in *Proceedings of the IEEE International Conference on Robotics and Automation*, 2003, pp. 2714–2719.

[15] T. C. H. Sit, Z. Liu, M. A. Jr., and W. K. GuanSeah, "Multi-robot mobility enhanced hop-count based localization in ad hoc networks," *Robotics and Autonomous Systems*, To appear.