Bayesian State Estimation of a Flexible Industrial Robot

Rickard Karlsson, Mikael Norrlöf,

Division of Automatic Control
Department of Electrical Engineering
Linköpings universitet, SE-581 83 Linköping, Sweden
WWW: <http://www.control.isy.liu.se>
E-mail: rickard@isy.liu.se, mino@isy.liu.se,

26th January 2005

Report no.: [LiTH-ISY-R-2677](#)

Technical reports from the Control & Communication group in Linköping are available at <http://www.control.isy.liu.se/publications>.

Avdelning, Institution Division, Department Division of Automatic Control Department of Electrical Engineering		Datum Date 2005-01-26	
Språk Language ☐ Svenska/Swedish ☒ Engelska/English ☐ _____		Rapporttyp Report category ☐ Licentiatavhandling ☐ Examensarbete ☐ C-uppsats ☐ D-uppsats ☒ Övrig rapport ☐ _____		ISBN _____ ISRN _____ Serietitel och serienummer ISSN Title of series, numbering 1400-3902	
URL för elektronisk version http://www.control.isy.liu.se		LiTH-ISY-R-2677			
Titel Bayesian State Estimation of a Flexible Industrial Robot Title					
Författare Rickard Karlsson, Mikael Norrlöf, Author					
Sammanfattning Abstract <p>A sensor fusion technique for state estimation of an industrial robot is presented. By measuring the acceleration at the end-effector, the accuracy of the arm angular position, velocity, and acceleration estimates can be improved. The problem is formulated in a Bayesian estimation framework and two solutions are proposed; one using the extended Kalman filter and one using the particle filter. In an extensive simulation study on a realistic flexible industrial robot, the performance is shown to be close to the fundamental Cramér-Rao lower bound. A significant improvement in position accuracy is achieved using the sensor fusion technique and the method is also proven to be robust to parameter variations in the model.</p>					
Nyckelord Keywords Industrial robot, positioning, estimation, particle filter, extended Kalman filter, Cramér-Rao lower bound					

Bayesian State Estimation of a Flexible Industrial Robot

Rickard Karlsson and Mikael Norrlöf

Abstract—A sensor fusion technique for state estimation of an industrial robot is presented. By measuring the acceleration at the end-effector, the accuracy of the arm angular position, velocity, and acceleration estimates can be improved. The problem is formulated in a Bayesian estimation framework and two solutions are proposed; one using the extended Kalman filter and one using the particle filter. In an extensive simulation study on a realistic flexible industrial robot, the performance is shown to be close to the fundamental Cramér-Rao lower bound. A significant improvement in position accuracy is achieved using the sensor fusion technique and the method is also proven to be robust to parameter variations in the model.

Index Terms—Industrial robot, positioning, estimation, particle filter, extended Kalman filter, Cramér-Rao lower bound

I. INTRODUCTION

MODERN industrial robot control is usually based only upon measurements from the motor angles of the manipulator. However, the ultimate goal is to move the tool according to a predefined path. In [9] a method for improving the absolute accuracy of a standard industrial manipulator is described. The improved accuracy is achieved through identification of unknown or uncertain parameters in the robot system, and applying the *iterative learning control* (ILC) method, [2], [24], using additional sensors to measure the actual tool position. The aim of this paper is to evaluate the Bayesian estimation techniques for sensor fusion and to estimate the tool position from indirect measurements such as the acceleration at the end-effector. It is assumed that a high accuracy at the tool position is needed, for instance laser cutting, and that low cost sensors such as accelerometers are used to improve positioning. The methods presented are applied to a realistic flexible robot model and configuration of the system with the accelerometer is depicted in Fig. 1.

Traditionally, many nonlinear estimation problems are solved using the *extended Kalman filter* (EKF) [1], [14], [10]. In [11] an EKF is used to improve the trajectory tracking for a rigid 2 *degree-of-freedom* (DOF) robot. Bayesian techniques have been applied in mobile robot applications, see e.g., [21], [13], [6, Ch. 19], but to the best of the authors knowledge, general Bayesian techniques have not yet been applied to position estimation in industrial robotics. The robot dynamics and measurements are highly nonlinear and the measurement noise is not always Gaussian. Hence, linearized models may not always be a good approach. The *particle filter* (PF),

Submitted to IEEE Transactions on Control Systems Technology (regular paper). R. Karlsson (corresponding author) and M. Norrlöf are with the Division of Automatic Control, Dept. of Electrical Engineering, Linköping University, SE-58183 Linköping, Sweden (e-mail: rickard@isy.liu.se, mino@isy.liu.se). Fax +46 13 282622, phone +46 13 281000.

Fig. 1. The ABB IRB1400 robot with an accelerometer mounted at the end-effector. The base coordinate system, (x_0, y_0, z_0) , and the coordinate system for the accelerometer, (x_a, y_a, z_a) , are also shown.

[6], provides a general solution to many problems where linearizations and Gaussian approximations are intractable or would yield too low performance. The PF method is also motivated since it provides the possibility to design control laws and perform diagnosis in a much more advanced way. This paper extends the idea introduced in [18]. A performance evaluation in a simulation environment for both the EKF and the PF is presented and it is extensively analyzed using the *Cramér-Rao lower bound* (CRLB) [4], [19]. The sensitivity to model errors is also considered and different levels of model simplification in the measurement equation are discussed and evaluated.

The paper is organized as follows. In Section II the theory of Bayesian estimation is introduced and the equations for the EKF are given. The particle filter algorithm is explained, and the concept of CRLB is presented. The simulation model, the estimation model, and the sensor model, are introduced in Section III. The results from the simulation experiments are covered in Section IV and the EKF and PF methods are compared for nominal model parameters but also in a sensitivity analysis with respect to model uncertainty. In Section V several motivations for the Bayesian state estimation problem are presented. Finally, Section VI contains a summary and conclusive remarks.

II. BAYESIAN ESTIMATION

Consider the discrete state-space model

$$x_{t+1} = f(x_t, u_t, w_t), \quad (1a)$$

$$y_t = h(x_t) + e_t, \quad (1b)$$

with state variables $x_t \in \mathbb{R}^n$, input signal u_t and measurements $\mathbb{Y}_t = \{y_i\}_{i=1}^t$, with known *probability density functions* (pdfs) for the process noise, $p_w(w)$, and measurement noise $p_e(e)$. The nonlinear prediction density $p(x_{t+1}|\mathbb{Y}_t)$ and filtering density $p(x_t|\mathbb{Y}_t)$ for the Bayesian inference, [12], is given by

$$p(x_{t+1}|\mathbb{Y}_t) = \int_{\mathbb{R}^n} p(x_{t+1}|x_t)p(x_t|\mathbb{Y}_t)dx_t, \quad (2a)$$

$$p(x_t|\mathbb{Y}_t) = \frac{p(y_t|x_t)p(x_t|\mathbb{Y}_{t-1})}{p(y_t|\mathbb{Y}_{t-1})}. \quad (2b)$$

For the important special case of linear-Gaussian dynamics and linear-Gaussian observations the Kalman filter, [15], will give the solution. For nonlinear and non-Gaussian systems, the pdf can not in general be expressed with a finite number of parameters. Instead approximative methods must be used. Usually this is done in two ways; either by approximating the system or by approximating the posterior pdf. See for instance, [30], [3]. Here two different approaches of solving the Bayesian equations are considered; extended Kalman filter, and particle filter. The EKF will solve the problem using a linearization of the system and assuming Gaussian noise. The PF on the other hand will approximately solve the Bayesian equations by stochastic integration. Hence, no linearizations errors occur. The PF can also handle non-Gaussian noise models where the pdfs are known only up to a normalization constant. Also hard constraints on the state variables can be incorporated in the estimation without any problems.

A. The Extended Kalman Filter (EKF)

For the special case of linear dynamics, linear measurements and additive Gaussian noise the Bayesian recursions in Section II have an analytical solution, the Kalman filter. For many nonlinear problems the noise assumptions and the nonlinearity are such that a linearized solution will be a good approximation. This is the idea behind the EKF, [1], [14], [10], where the model is linearized around the previous estimate. Here the time update and measurement update for the EKF is presented,

$$\begin{cases} \hat{x}_{t+1|t} = f(\hat{x}_{t|t}, u_t, 0), \\ P_{t+1|t} = F_t P_{t|t} F_t^T + G_t Q_t G_t^T, \end{cases} \quad (3a)$$

$$\begin{cases} \hat{x}_{t|t} = \hat{x}_{t|t-1} + K_t(y_t - h(\hat{x}_{t|t-1})), \\ P_{t|t} = P_{t|t-1} - K_t H_t P_{t|t-1}, \\ K_t = P_{t|t-1} H_t^T (H_t P_{t|t-1} H_t^T + R_t)^{-1}, \end{cases} \quad (3b)$$

where the linearized matrices are given as

$$F_t = \nabla_x f(x_t, u_t, 0)|_{x_t=\hat{x}_{t|t}}, \quad (4a)$$

$$G_t = \nabla_w f(x_t, u_t, w_t)|_{x_t=\hat{x}_{t|t}}, \quad (4b)$$

$$H_t = \nabla_x h(x_t)|_{x_t=\hat{x}_{t|t-1}}. \quad (4c)$$

The noise covariances are given as

$$Q_t = \text{Cov}(w_t), \quad R_t = \text{Cov}(e_t). \quad (5)$$

B. The Particle Filter (PF)

In this section the presentation of the particle filter theory is according to [4], [6], [7], [16], [28]. The PF provides an approximate solution to the discrete time Bayesian estimation problem formulated in (2) by updating an approximate description of the posterior filtering density. Let x_t denote the state of the observed system and $\mathbb{Y}_t = \{y^{(i)}\}_{i=1}^t$ be the set of observed measurements until present time. The PF approximates the density $p(x_t|\mathbb{Y}_t)$ by a large set of N samples (particles), $\{x_t^{(i)}\}_{i=1}^N$, where each particle has an assigned relative weight, $\gamma_t^{(i)}$, chosen such that all weights sum to unity. The location and weight of each particle reflect the value of the density in the region of the state space. The PF updates the particle location and the corresponding weights recursively with each new observed measurement. For the common special case of additive measurement noise the unnormalized weights are given by

$$\gamma_t^{(i)} = p_e(y_t - h(x_t^{(i)})), \quad i = 1, \dots, N. \quad (6)$$

Using the samples (particles) and the corresponding weights the Bayesian equations can be approximately solved. To avoid divergence a resampling step is introduced. This is referred to as the *Sampling Importance Resampling* (SIR), [7], and is summarized in Algorithm 1.

Alg. 1 Sampling Importance Resampling (SIR)

- 1: Generate N samples $\{x_0^{(i)}\}_{i=1}^N$ from $p(x_0)$.
 - 2: Compute $\gamma_t^{(i)} = p_e(y_t - h(x_t^{(i)}))$ and normalize, i.e., $\bar{\gamma}_t^{(i)} = \gamma_t^{(i)} / \sum_{j=1}^N \gamma_t^{(j)}$, $i = 1, \dots, N$.
 - 3: Generate a new set $\{x_t^{(i)}\}_{i=1}^N$ by resampling with replacement N times from $\{x_t^{(i)}\}_{i=1}^N$, with probability $\bar{\gamma}_t^{(i)} = \text{Pr}\{x_t^{(i*)} = x_t^{(i)}\}$.
 - 4: $x_{t+1}^{(i)} = f(x_t^{(i*)}, u_t, w_t^{(i)})$, $i = 1, \dots, N$ using different noise realizations, $w_t^{(i)}$.
 - 5: Increase t and continue to step 2.
-

The estimate for each time, t , is often chosen as the minimum mean square estimate, i.e.,

$$\hat{x}_t = \mathbb{E}(x_t) = \int_{\mathbb{R}^n} x_t p(x_t|\mathbb{Y}_t) dx_t \approx \sum_{i=1}^N \gamma_t^{(i)} x_t^{(i)}, \quad (7)$$

but other choices, such as the ML-estimate, might be of interest. The PF approximates the posterior pdf, $p(x_t|\mathbb{Y}_t)$, by a finite number of particles. There exist theoretical limits [6], that the approximated pdf converges to the true as the number of particles tends to infinity.

C. Cramér-Rao Lower Bound

When different estimators are used it is fundamental to know the best possible achievable performance. As mentioned previously, the PF will approach the true pdf asymptotically, but for any implementation, due to finite number of particles, only an approximation is given. For other estimators, such as the EKF, it is important to know how much the linearization or model structure used, will affect the performance. The

Cramér-Rao lower bound (CRLB) is such a characteristic for the second order moment [19]. Here only state-space models with additive Gaussian noise are considered. The theoretical posterior CRLB for a general dynamic system was derived in [32], [31], [4], [6]. Here a continuous-time system is considered. By first linearizing and then discretizing the system, the fundamental limit can in practice be calculated as the stationary solution, $\bar{P} = \bar{P}(x_t^{\text{TRUE}})$, of the Riccati recursions in the EKF, where the linearizations are around the true state trajectory, x_t^{TRUE} . Note that for the industrial robot application a high sample rate and a small process noise make the approximation fairly accurate. The predicted value of the stationary covariance for each time t , i.e., for each point in the state-space, x_t^{TRUE} , is denoted $\bar{P}_p$ and given as the solution to

$$\bar{P}_p = \bar{F}(\bar{P}_p - (\bar{P}_p \bar{H}^T (\bar{H} \bar{P}_p \bar{H}^T + R)^{-1} \bar{H} \bar{P}_p) \bar{F}^T + \bar{G} Q \bar{G}^T). \quad (8)$$

where the linearized matrices are evaluated around the true trajectory, x_t^{TRUE} . The CRLB limit can now be calculated as

$$\bar{P} = \bar{P}_p - \bar{K} \bar{H} \bar{P}_p, \quad (9)$$

for each point along the state-trajectory.

III. MODELS

In this section a continuous-time 3 DOF robot model is discussed. The model is simplified and transformed into discrete time where it can be used by the EKF and PF. The measurements are in both cases angle measurements from the motor, with or without acceleration information from the arm.

A. Robot Model

A common assumption of the dynamics of the robot is that the transmission can be approximated by two or three masses connected by springs and dampers. The coefficients in the resulting model can be estimated from an identification experiment. See for instance [20]. Here it will be assumed that the transmission can be described by a two mass system and that the manipulator is rigid, schematically shown in Fig. 2. The equation describing the torque balance for the motor

Fig. 2. A two mass model of the dynamics between two joints in the industrial robot; spring (k), damper (d), friction f_m , gear ratio (r_g), moments of inertia (M_m , $M_a(q_a)$), torque (τ) and angles (q_m , q_a).

becomes

$$M_m \ddot{q}_m = -f_m \dot{q}_m - r_g k (r_g q_m - q_a) - r_g d (r_g \dot{q}_m - \dot{q}_a) + \tau, \quad (10)$$

where M_m is the motor inertia matrix, q_m the motor angle, q_a the arm angle, r_g the gear ratio, f_m , k , and d are the motor

friction, spring constant and damping respectively. Input to the system is the motor torque, τ . The corresponding relation for the arm becomes a nonlinear equation

$$M_a(q_a) \ddot{q}_a + C(q_a, \dot{q}_a) \dot{q}_a + g(q_a) = k(r_g q_m - q_a) + d(r_g \dot{q}_m - \dot{q}_a). \quad (11)$$

A more detailed model of the robot should include nonlinear friction such as Coulomb friction. An important extension would also be to model the nonlinear spring characteristics in the gear-boxes. In general the gear-box is less stiff for torques close to zero and more stiff when high torques are applied.

An industrial robot has, in general, 6 DOF. However, here only joint 1-3 (not the wrist joints) are used in the simulation study. The continuous-time robot model is implemented in MATLAB Simulink [23], as presented in Fig. 3. The manipulator block uses the Simulink block from Robotics Toolbox [5] to simulate the nonlinear dynamics.

Fig. 3. The MATLAB Simulink model of the robot and PID controller.

The Denavit-Hartenberg (DH) parameters for the robot are given in Table I. The robot is stabilized using a PID-controller,

$$F_{\text{PID}}(s) = K_p + \frac{K_I}{s} + \frac{K_D s}{s^2 + 1}, \quad (12)$$

where s denotes the Laplace operator. The parameters are described in Table II. For specific values of the dynamics, see [33].

B. Estimation Model

The estimation model has to reflect the dynamics in the true system. A straight forward choice of estimation model

TABLE I
DH PARAMETERS FOR THE ROBOT USED IN THE SIMULATION.

Joint/Link i	α_i [rad]	a_i [m]	θ_i [rad]	d_i [m]
1	$-\pi/2$	0.41	0	0.78
2	0	1.075	$-\pi/2$	0
3	$\pi/2$	1.056	$\pi/2$	0

TABLE II
PID PARAMETERS USED IN THE SIMULATION.

Parameter	Joint 1	Joint 2	Joint 3
K_p	40	50	40
K_I	40	40	40
K_D	2	2	2
b	150	150	150

is the state space equivalent of (11) and (10), this gives a nonlinear dynamic model with 12 states (motor and arm angular positions, velocities). Since the goal is to find an estimate of the arm angles the following state variables are used

$$x_t = (q_{a,t} \quad \dot{q}_{a,t} \quad \ddot{q}_{a,t})^T, \quad (13)$$

where $q_{a,t} = (q_{a,t}^1 \quad q_{a,t}^2 \quad q_{a,t}^3)^T$ contains the arm angles from the first three joints in Fig. 1 and $\dot{q}_{a,t}$ is the angular velocity and $\ddot{q}_{a,t}$ is the angular acceleration at time t . This yields the following state space model in discrete time

$$x_{t+1} = F_t x_t + G_{u,t} u_t + G_{w,t} w_t, \quad (14a)$$

$$y_t = h(x_t) + e_t, \quad (14b)$$

where

$$F_t = \begin{pmatrix} \mathcal{I} & T\mathcal{I} & T^2/2\mathcal{I} \\ \mathcal{O} & \mathcal{I} & T\mathcal{I} \\ \mathcal{O} & \mathcal{O} & \mathcal{I} \end{pmatrix}, \quad (15a)$$

$$G_{w,t} = \begin{pmatrix} \frac{T^2}{2}\mathcal{I} \\ T\mathcal{I} \\ \mathcal{I} \end{pmatrix}, \quad G_{u,t} = \begin{pmatrix} \frac{T^3}{6}\mathcal{I} \\ \frac{T^2}{2}\mathcal{I} \\ T\mathcal{I} \end{pmatrix}. \quad (15b)$$

This model is linear and the number of states is reduced to 9, compared to the 12 states in the nonlinear state space model. In Section II-C a linearized model of the nonlinear dynamics is presented for computing the Cramér-Rao lower bound. The input, u_t , is the arm jerk reference, i.e., the differentiated arm angular acceleration reference. The process noise, w_t and measurement noise e_t are considered Gaussian with zero mean and covariances, Q_t and R_t respectively. The sample time is denoted T and $\mathcal{I}$ and $\mathcal{O}$ are three by three unity and null matrices. The observation relation, (14b), is described in full detail in the next section.

C. Sensor Model

The observation relation is given by

$$h(x_t) = \begin{pmatrix} q_{m,t} \\ \ddot{\rho}_t \end{pmatrix}, \quad (16)$$

where $q_{m,t}$ is the measured motor angle and $\ddot{\rho}_t$ is the Cartesian acceleration vector in the accelerometer frame, Fig. 1. With the simplified discrete time model described in Section III-A, the motor angle $q_{m,t}$ is computed as

$$q_m = r_g^{-1} \left(q_a + k^{-1} (M_a(q_a) \ddot{q}_a + g(q_a) + C(q_a, \dot{q}_a) \dot{q}_a - d(r_g \dot{q}_m - \dot{q}_a)) \right). \quad (17)$$

The approach is similar to the one suggested in [8], which uses the relation given by (11) in a case when the system is scalar and linear. The results presented here are more general, since a multi-variable nonlinear system is considered.

The kinematics [29] of the robot is described by a nonlinear mapping $\rho_t = \mathcal{T}(q_{a,t})$, and its Jacobian is defined as

$$J(q_a) = \nabla_{q_a} \mathcal{T}(q_a). \quad (18)$$

The following equation relates the Cartesian acceleration with the state variables

$$\ddot{\rho}_t = J(q_{a,t}) \ddot{q}_{a,t} + \left(\sum_{i=1}^3 \frac{\partial J(q_{a,t})}{\partial q_{a,t}^{(i)}} \dot{q}_{a,t}^{(i)} \right) \dot{q}_{a,t} + n_g(q_{a,t}), \quad (19)$$

where $q_{a,t}^{(i)}$ is the i th element of $q_{a,t}$ and $n_g(q_{a,t})$ is the gravity vector measured by the accelerometer.

Remark: If the nonlinear dynamics (11) and (10), are used, see Section III-A, the relation in (17) becomes linear since $q_{m,t}$ is a state variable. However, the relation in (19) becomes more complex since $\ddot{q}_{a,t}$ is no longer a state, but has to be computed using (11).

D. CRLB Analysis of the Robot

In Section II-C the posterior CRLB was defined for a general nonlinear system with additive Gaussian noise. In this section the focus is on the CRLB expression for the industrial robot presented in Section III-A. Equation (11) gives

$$\kappa(q_a, \dot{q}_a) \triangleq \ddot{q}_a = -M_a(q_a)^{-1} (k(r_g q_m - q_a) - d(r_g \dot{q}_m - \dot{q}_a) - g(q_a) - C(q_a, \dot{q}_a) \dot{q}_a). \quad (20)$$

Here, the motor angular velocity, $\dot{q}_m$, is considered as an input signal, hence not part of the state-vector, $x(t) = (q_a \quad \dot{q}_a \quad \ddot{q}_a)^T$. The system can be written in state space form as

$$\dot{x} = \frac{d}{dt} \begin{pmatrix} q_a \\ \dot{q}_a \\ \ddot{q}_a \end{pmatrix} = f^c(x(t)) = \begin{pmatrix} \dot{q}_a \\ \ddot{q}_a \\ \Lambda(q_a, \dot{q}_a, \ddot{q}_a) \end{pmatrix}, \quad (21a)$$

$$\Lambda(q_a, \dot{q}_a, \ddot{q}_a) = \frac{d}{dt} \kappa(q_a, \dot{q}_a). \quad (21b)$$

The differentiation of κ is performed symbolically, using the MATLAB symbolic toolbox. According to Section II-C the CRLB is defined as the stationary Riccati solution of the EKF around the true trajectory, x_t^{TRUE} . This is formulated for a discrete-time system. Hence, the continuous-time robot model from (21) must first be discretized. This can be done by first linearizing the system and then discretize it, [10]. The system is too complicated to apply a symbolic gradient directly in the linearization. However, a numerical differentiation can be done around the true trajectory, yielding

$$A^c = \nabla_x f^c(x)|_{x=x_t^{\text{TRUE}}} = \begin{pmatrix} \mathcal{O} & \mathcal{I} & \mathcal{O} \\ \mathcal{O} & \mathcal{O} & \mathcal{I} \\ \frac{\partial \Lambda(q, \dot{q}, \ddot{q})}{\partial q} & \frac{\partial \Lambda(q, \dot{q}, \ddot{q})}{\partial \dot{q}} & \frac{\partial \Lambda(q, \dot{q}, \ddot{q})}{\partial \ddot{q}} \end{pmatrix}, \quad (22)$$

The desired discrete time system matrix is now given as

$$\bar{F} = e^{A^c T}, \quad (23)$$

where T is the sample time. In Section IV the CRLB is compared to the estimation result in Monte Carlo simulations, both with and without accelerations measurements, using the above system and the CRLB expressions presented in Section II-C.

IV. SIMULATION RESULTS

The model is implemented and simulated using the Robotics Toolbox [5] in MATLAB Simulink as is described in Section III-A. The simulation study is based mainly around the EKF approach, since it is a fast method well suited for large Monte Carlo simulations. The impact on several different parts in the modeling are studied together with a sensitivity analysis. Also performance simulations around nominal parameters are compared to the CRLB. The PF is much slower, hence a smaller Monte Carlo study is performed. The Monte Carlo simulations use the following covariance matrices for the process and measurement noise

$$Q = 4 \cdot 10^{-6} \mathcal{I}, \quad R = \begin{pmatrix} 10^{-6} \cdot \mathcal{I} & \mathcal{O} \\ \mathcal{O} & 10^{-4} \cdot \mathcal{I} \end{pmatrix}. \quad (24)$$

The system is simulated around the nominal trajectory and produces different independent noise realizations for the measurement noise in each simulation. The same covariances are used in the CRLB evaluation. The continuous-time Simulink model of the robot is sampled in 1 kHz. The data is then decimated to 100 Hz before any estimation method is applied. The arm reference path is presented in Fig. 4. The robot illustration is done using Robotics Toolbox, [5], and the path is created using the Path Generation Toolbox (PGT), [27].

Fig. 4. Illustration of the path in Cartesian space used in the simulation. The configuration of the robot is shown for the initial position of the path.

A. Modeling and Sensitivity Analysis

The model of the angular measurement relation (17) consists of different terms, the angular term, the gravitational term, the inertia term, the Coriolis term and the damping term. Since the system does not work properly without any angular measurements this is considered mandatory. The damping part can only be analyzed if $\dot{q}_m$ is available. $\dot{q}_m$ is not part of the state-vector, but could in principle be estimated if included or directly estimated by differentiating the signal q_m . In Fig. 5 the first four terms of (17) are shown for the data in the simulation. From Fig. 5 the importance of the different terms can be concluded. The term containing q_a is fundamental, the gravitational term is also important since it gives a bias to the

Fig. 5. Terms in the measurement equation, (17), (dash-dotted) represents joint 1, (dashed) joint 2, and (dotted) joint 3. (a) $r_g^{-1} q_a$, (b) $r_g^{-1} k^{-1} M_a(q_a) \ddot{q}_a$, (c) $r_g^{-1} k^{-1} g(q_a)$, and (d) $r_g^{-1} k^{-1} C(q_a, \dot{q}_a) \dot{q}_a$.

estimate. The inertia term also contributes together with the Coriolis term. The damping term has been neglected since it has a much lower amplitude than the other terms.

The different models are evaluated using the EKF and the error is measured as $\|q_{a,t} - \hat{q}_{a,t}\|_2$. As seen in Fig. 6, the incorporation of inertia and Coriolis significantly improve performance. This is also shown in Table III, where the

Fig. 6. The angular error, $\|q_{a,t} - \hat{q}_{a,t}\|_2$, for different models according to Table III using a noiseless realization.

average error for the noise free case is presented.

To illustrate the estimation idea, the estimate of q_a is presented for the EKF and directly from motor measurements only. The estimate without any model, $\hat{q}_a$ is based on

$$\hat{q}_a = r_g q_m - k^{-1} g(q_a), \quad (25)$$

where the gravitational part is compensated statically, using the true value. In Fig. 7 the estimation and the direct transformation from motor measurements only is illustrated based

TABLE III

AVERAGE ANGULAR ESTIMATION ERROR FOR DIFFERENT MODELS
AROUND THE NOMINAL TRAJECTORY, NOISE FREE SYSTEM.

Model	Model of measurement eq.	Avg. error [rad]
I.	$r_g^{-1}(q_a + k^{-1}g(q_a))$	$7.66 \cdot 10^{-5}$
II.	$r_g^{-1}(q_a + k^{-1}g(q_a) + k^{-1}M_a(q_a)\dot{q}_a)$	$1.31 \cdot 10^{-5}$
III.	$r_g^{-1}(q_a + k^{-1}g(q_a) + k^{-1}M_a(q_a)\dot{q}_a + k^{-1}C(q_a, \dot{q}_a)\dot{q}_a)$	$1.02 \cdot 10^{-6}$
IV.	$r_g^{-1}(q_a + k^{-1}g(q_a) + k^{-1}M_a(q_a)\dot{q}_a + k^{-1}C(q_a, \dot{q}_a)\dot{q}_a - k^{-1}d(r_g\dot{q}_m - \dot{q}_a))$	$7.53 \cdot 10^{-7}$

on 500 Monte Carlo simulations. The evaluation uses the *root mean square error* (RMSE) defined as

$$\text{RMSE}(t) = \left(\frac{1}{N_{\text{MC}}} \sum_{j=1}^{N_{\text{MC}}} \|x_t^{\text{TRUE}} - \hat{x}_t^{(j)}\|_2^2 \right)^{1/2}, \quad (26)$$

where N_{MC} is the number of Monte Carlo simulations, x_t^{TRUE} is the true state vector and $\hat{x}_t^{(j)}$ is the estimated state vector in Monte Carlo simulation j . As seen in Fig. 7, the EKF gives a significant improvement in arm position even if the accelerometer is not used. This is because the states are available in the EKF and a detailed model can be included. The filter also reduces the effect of measurement disturbances.

Fig. 7. The arm position estimate, $\hat{q}_a$ from motor-side measurements only (dotted) with static gravity compensation and the estimate $\hat{q}_a$ from EKF without using the accelerometer (solid).

It is also interesting to investigate the sensitivity of the estimator for important parameters, such as the damping coefficient d and the spring coefficient k . In Fig. 8 the total RMSE for different parameter variations is presented using 10 Monte Carlo simulations for each specific parameter.

B. Estimation Performance

EKF. In Fig. 9 the RMSE from 500 Monte Carlo simulations are compared with the CRLB limit, both with and without acceleration measurements. The CRLB is computed

Fig. 8. Total RMSE angular position sensitivity analysis for the spring k variation (upper) and the damper d variation (lower), $N_{\text{MC}} = 10$.

as the square root of the trace for the covariance matrix part corresponding to the angular states. As seen the RMSE is close the fundamental limit. The discrepancy is due to model errors, i.e., neglected damping term and the fact that the estimator uses a simplified system matrix consisting of integrators only. Also note that the accelerometer measurements reduce the estimation uncertainty. The results in Fig. 9 is of course for the chosen trajectory, but the acceleration values are not that large, so greater differences will occur for larger accelerations. The total RMSE, ignoring the initial transient is given in

Fig. 9. Angular position RMSE from 500 Monte Carlo simulations using the EKF without and with accelerometer sensor are compared to the CRLB limit, i.e., the square root of the trace of the angular position from the CRLB covariance.

Table IV for both angular position, velocity and acceleration. The improvement is substantial in angular position, but for control also the improvement in angular velocity and acceleration is important. On a 1.5 GHz PC running MATLAB the EKF performs in real-time on the 100 Hz data rate.

PF. The particle filter is rather slow compared to the

TABLE IV

TOTAL RMSE FOR ARM-SIDE ANGULAR POSITION (q_a), ANGULAR VELOCITY ($\dot{q}_a$) AND ANGULAR ACCELERATION ($\ddot{q}_a$), WITH AND WITHOUT THE ACCELEROMETER, USING 500 MONTE CARLO SIMULATIONS.

	Accelerometer	No accelerometer
RMSE q_a	$1.25 \cdot 10^{-5}$	$2.18 \cdot 10^{-5}$
RMSE $\dot{q}_a$	$7.57 \cdot 10^{-5}$	$4.08 \cdot 10^{-4}$
RMSE $\ddot{q}_a$	$1.23 \cdot 10^{-3}$	$3.91 \cdot 10^{-3}$

EKF for this model structure. Hence, the given MATLAB implementation of the system is not well suited for large Monte Carlo simulations. Instead a small Monte Carlo study over a much shorter time period than for the EKF case is considered. The PF and the EKF are compared, and a small improvement in performance is noted. The result is given in Fig. 10. Even though the PF is slow, it gives more insight in

Fig. 10. EKF and PF angular position RMSE with external accelerometer signal from 20 Monte Carlo simulations.

the selection of simulation parameters than the EKF, where the filter performance is more dependent on the ratio between the process and measurement noise. Since the data rate is rather high the linearization problem is not severe, so the EKF performs sufficiently well. To improve performance a more complicated system dynamics must be implemented in the filter.

V. MOTIVATIONS

In this section motivations for the Bayesian methods presented are given and future work are indicated. With a highly accurate tool position estimate, the control of the robot can be improved. However, to incorporate the estimates in a closed loop real-time system may not be possible due to the computational complexity in the estimation methods. This is not a problem in many practical applications. Consider for instance ILC, which is an off-line method. ILC has over the years become a standard method for achieving high accuracy in industrial robot control [2], [26], [22]. It utilizes a repetitive system dynamics to compensate for errors. Mathematically an

ILC control law can be written as

$$u_{t,k+1} = Q(u_{t,k} + \mathcal{L}\epsilon_{t,k}), \quad (27)$$

where $u_{t,k}$ is the ILC input in the k th iteration and $\epsilon_{t,k}$ is the error. The error is defined as $\epsilon_{t,k} = r_t - y_{t,k}$ where r is the reference and $y_{t,k}$ the measured output of the system. Q and $\mathcal{L}$ are design parameters for the control law, often chosen as linear filters [24]. In industrial robot systems the measured output does not correspond to the actual controlled output. An ILC experiment on the ABB IRB1400 in [25, Chapter 9] using only motor angle measurements, i.e., no accelerometer, shows that although the error on the motor-side is reduced the path on the arm-side does not follow the programmed path. This is illustrated in Fig. 11.

Fig. 11. Results from an ILC experiment on the ABB IRB1400 robot where ILC is applied using only motor angle measurements. Programmed path (left), iteration 0 (middle), and iteration 10 (right). The path on the arm-side does not follow the programmed trajectory.

The idea in this paper is to use an accelerometer at the end-effector to get measurements that reflects the actual tool motion, see Fig. 1. From these measurements and a model of the robot the estimated position error, $\hat{\epsilon}_{t,k}$, is used in the ILC update equation according to

$$u_{t,k+1} = Q(u_{t,k} + \mathcal{L}\hat{\epsilon}_{t,k}). \quad (28)$$

Using the EKF, $\hat{\epsilon}_{t,k}$ represent the mean error, with an estimate of the covariance from the EKF. Hence, this can be used in the improvement process and an idea in this direction is presented in [26]. The covariance could be used to change the gain of the learning operator $\mathcal{L}$ in order to reduce the effect of random disturbances. In [8] a 1 DOF lab-process is controlled using (28) but the estimation is simplified due to the inherent linearity of the system. Using the PF, the pdf $p(x_t|\mathbb{Y}_t)$ is given by samples, $x_t^{(i)}, i = 1, \dots, N$ as described in Section II-B. Hence, the error consists of samples $\epsilon_t^{(i)}$, so the ILC improvement can be done in a more sophisticated way. The mean estimate or the maximum likelihood (ML) estimate, or a combination thereof are logical choices.

The extra knowledge of the pdf from the PF can also be used in for instance diagnosis. A general model-based statistical decision rule based on a criterion, $g(x_t)$, can be formulated. The probability can be calculated, for instance $Pr\{g(x_t) > 0\} > 1 - \beta$, where β determines the confidence level. Here, the samples $x_t^{(i)}, i = 1, 2, \dots, N$ from the distribution $p(x_t|\mathbb{Y}_t)$ can be used to calculate the probability as

$$Pr\{g(x_t) > 0\} \approx \frac{\#g(x_t^{(i)}) > 0}{N}, \quad (29)$$

by simply counting the number of samples fulfilling the criteria. Since these samples are directly available in the PF

method it is well suited for hypothesis testing. Similar for the EKF, an off-line Monte Carlo integration technique can be used, but this introduces extra computations since the samples are not intrinsic in the algorithm. For details see for instance [17], where this idea was tested.

VI. CONCLUSIONS

A sensor fusion approach to find estimates of the tool position by combining a 3-axes accelerometer and the measurements from the motor angles of an industrial robot is presented. The estimation is formulated as a Bayesian problem and two solutions are proposed; extended Kalman filter and particle filter respectively. The algorithms were tested on data from a realistic robot model. For the linear dynamical model used in the estimation, sufficiently accurate estimates are produced. The performance both with and without accelerometer measurements are close to the fundamental Cramér-Rao lower bound limit. Estimation performance with the accelerometer is better, considering both the Cramér-Rao lower bound and the actual result from the Monte Carlo simulations. A comparison of a filter-based and a non-filter based approach to find estimates of the arm angle shows a significant improvement using the model-based filter. The velocity estimates are also proven to be much more accurate when the filter uses information from the accelerometer. This is important for control design in order to give a well damped response at the robot arm. Under the assumption that the robot can be statically calibrated the estimation algorithms are shown to be very robust to variations in the spring constant and the damping coefficient. Since the intended use of the estimates is to improve position control using an off-line method, like ILC, there are no real-time issues using the computational demanding particle filter algorithm, however the extended Kalman filter runs in real-time in MATLAB. The estimation methods presented in this paper are general and can be extended to higher degrees of freedom robots and additional sensors, such as accelerometers, gyros, or camera systems, can be included. The main effect is a more complex measurement equation.

ACKNOWLEDGMENT

This work was supported by the VINNOVA's Center of Excellence ISIS (Information Systems for Industrial Control and Supervision) at Linköping University, Sweden, particularly the partner ABB.

REFERENCES

- [1] B.D.O. Anderson and J. B. Moore. *Optimal Filtering*. Prentice Hall, Englewood Cliffs, NJ, 1979.
- [2] S. Arimoto, S. Kawamura, and F. Miyazaki. Bettering operation of robots by learning. *Journal of Robotic Systems*, 1(2):123–140, 1984.
- [3] M. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Trans. Signal Processing*, February 2002.
- [4] N. Bergman. *Recursive Bayesian Estimation: Navigation and Tracking Applications*. Linköping Studies in Science and Technology. Dissertations No. 579, Linköping University, Linköping, Sweden, 1999.
- [5] P. I. Corke. A robotics toolbox for MATLAB. *IEEE Robot. Automat. Mag.*, 3(1):24–32, March 1996.
- [6] A. Doucet, N. de Freitas, and N. Gordon, editors. *Sequential Monte Carlo Methods in Practice*. Springer Verlag, 2001.
- [7] N. J. Gordon, D. J. Salmond, and A.F.M. Smith. A novel approach to nonlinear/non-Gaussian Bayesian state estimation. In *IEEE Proceedings on Radar and Signal Processing*, volume 140, pages 107–113, 1993.
- [8] S. Gunnarsson and M. Norrlöf. Iterative learning control of a flexible robot arm using accelerometers. In *IEEE Conference on Control Applications*, Taipei, Taiwan, September 2004.
- [9] S. Gunnarsson, M. Norrlöf, G. Hovland, U. Carlsson, T. Brogårdh, T. Svensson, and S. Moberg. Pathcorrection for an industrial robot. European Patent Application No. EP1274546, April 2001.
- [10] F. Gustafsson. *Adaptive Filtering and Change Detection*. John Wiley & Sons Ltd, 2000.
- [11] R. Jassemi-Zargani and D. Neculescu. Extended Kalman filter-based sensor fusion for operational space control of a robot arm. *IEEE Trans. Instrum. Meas.*, 51(6):1279 – 1282, December 2002.
- [12] A. H. Jazwinski. *Stochastic Processes and Filtering Theory*, volume 64 of *Mathematics in Science and Engineering*. Academic Press, 1970.
- [13] P. Jensfelt. *Approaches to Mobile Robot Localization in Indoor Environments*. PhD thesis, Royal Institute of Technology, 2001.
- [14] T. Kailath, A.H. Sayed, and B. Hassibi. *Linear Estimation*. Information and System Sciences. Prentice Hall, Upper Saddle River, New Jersey, 2000.
- [15] R. E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the AMSE-Journal of Basic Engineering*, 82:35–45, 1960.
- [16] R. Karlsson. *Simulation Based Methods for Target Tracking*. Linköping Studies in Science and Technology. Licentiate Thesis No. 930, Linköping University, Linköping, Sweden, February 2002.
- [17] R. Karlsson, J. Jansson, and F. Gustafsson. Model-based statistical tracking and decision making for collision avoidance application. In *Proceedings of American Control Conference*, Boston, MA, USA, June 2004.
- [18] R. Karlsson and M. Norrlöf. Bayesian position estimation of an industrial robot using multiple sensors. In *Proceedings of the IEEE Conference on Control Applications*, Taipei, Taiwan, September 2004.
- [19] S. Kay. *Fundamentals of Statistical Signal Processing*. Prentice Hall, 1993.
- [20] K. Kozłowski. *Modelling and Identification in Robotics*. Springer-Verlag, 1998.
- [21] C. Kwok, D. Fox, and M. Meila. Real-time particle filters. *Proceedings of the IEEE*, 92(3):469–484, March 2004.
- [22] A. D. Luca, G. Paesano, and G. Ulivi. A frequency-domain approach to learning control: Implementation for a robot manipulator. *IEEE Trans. Ind. Electron.*, 39(1), February 1992.
- [23] MathWorks, The MathWorks, Mail 3 Apple Hill Dr. Natick, MA 01760-2098, USA. *SIMULINK: Model-Based and System-Based Design*, 2003.
- [24] K. L. Moore. *Iterative Learning Control for Deterministic Systems*. Advances in Industrial Control. Springer-Verlag, London, 1993.
- [25] M. Norrlöf. *Iterative Learning Control. Analysis, design and experiments*. PhD thesis, Linköping University, Linköping, Sweden, 2000. Linköping Studies in Science and Technology. Dissertations No. 653.
- [26] M. Norrlöf. An adaptive iterative learning control algorithm with experiments on an industrial robot. *IEEE Trans. Robot. Automat.*, 18(2):245–251, April 2002.
- [27] M. Nyström and M. Norrlöf. PGT - A path generation toolbox for Matlab (v0.1). Technical Report LiTH-ISY-R-2542, Department of Electrical Engineering, SE-581 83 Linköping, Sweden, September 2003.
- [28] B. Ristic, S. Arulampalam, and N. Gordon. *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. Artech House, 2004.
- [29] L. Sciacivico and B. Siciliano. *Modelling and Control of Robot Manipulators*. Springer, 2000.
- [30] H. W. Sorenson. *Recursive Estimation for Nonlinear Dynamic Systems*. In *Bayesian Analysis of Time Series and Dynamic Models* (Ed. J. C. Spall), Dekker, 1988.
- [31] P. Tichavsky, P. Muravchik, and A. Nehorai. Posterior Cramér-Rao bounds for discrete-time nonlinear filtering. *IEEE Trans. Signal Processing*, 46(5):1386–1396, 1998.
- [32] H. L. Van Trees. *Detection, Estimation and Modulation Theory*. Wiley, New York, 1968.
- [33] E. Wernholt and M. Östring. Modeling and control of a bending backwards industrial robot. Technical Report LiTH-ISY-R-2522, Department of Electrical Engineering, SE-581 83 Linköping, Sweden, May 2003.