

POSITION ESTIMATION AND MODELING OF A FLEXIBLE INDUSTRIAL ROBOT

R. Karlsson ^{*,1} M. Norrlöf ^{*,1}

** Dept. of Electrical Engineering, Linköping University*

Abstract: A sensor fusion technique is presented and it is shown to achieve good estimates of the position for a 3 degrees-of-freedom industrial robot model. By using an accelerometer the estimate of the tool position accuracy can be improved. The computation of the position is formulated as a Bayesian estimation problem and two solutions are proposed. One using the extended Kalman filter and one using the particle filter. Since the aim is to use the positions estimates to improve trajectory tracking with an iterative learning control method, no computational constraints arise. In an extensive simulation study the performance is compared to the Cramér-Rao lower bound. A significant improvement in position accuracy is achieved using the sensor fusion technique. *Copyright*©2005 IFAC.

Keywords: Industrial robots, estimation, extended Kalman filters, estimation algorithms

1. INTRODUCTION

Modern industrial robot control is usually based only upon measurements from the motor angles of the manipulator. The ultimate goal however is to make the tool move according to some predefined path. In (Gunnarsson *et al.*, 2001) 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, using additional sensors, and applying the *iterative learning control* (ILC) method,(Arimoto *et al.*, 1984; Moore, 1993). In (Gunnarsson *et al.*, 2001) no specific method to estimate the position is presented. The aim of this paper is therefore to evaluate the Bayesian estimation techniques for sensor fusion. The methods presented are applied to a realistic flexible robot model and the configuration of the system with the accelerometer is depicted in Fig. 1.

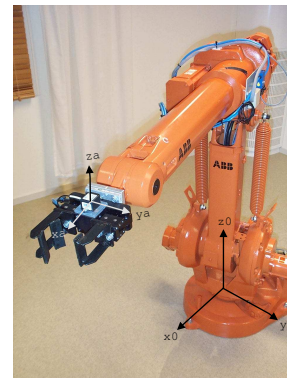


Fig. 1. An ABB IRB1400 robot with a 3-DOF accelerometer mounted at the tool.

During the 1980's there were several contributions directed towards control of joint flexible industrial robots using continuous time observers. One example is (Nicosia *et al.*, 1988) where an observer for a class of nonlinear systems is presented. The disturbances are not utilized in the estimation. In a Bayesian framework estimation problems are traditionally solved using linearized filters, mainly *extended Kalman filters* (EKF's) (Anderson and Moore, 1979). In (Jassemi-

¹ Supported by the ViNNNOVA Center of Excellence Information Systems for Industrial Control and Supervision.

Zargani and Neculescu, 2002) an EKF is used to improve the trajectory tracking for a rigid 2 *degree-of-freedom* (DOF) robot. 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),(Doucet *et al.*, 2001), 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 design control laws and perform diagnosis in a much more advanced way. This paper extends the idea introduced in (Karlsson and Norrlöf, 2004), where experimental data was used in an EKF together with tool acceleration measurements. Here, a performance evaluation in a simulation environment is presented for both the EKF and the particle filter. Performance is also analyzed using the *Cramér-Rao lower bound* (CRLB).

2. MOTIVATIONS

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 some practical applications. Consider for instance *iterative learning control* (ILC), which is an off-line method. ILC has over the years become a standard method for achieving high accuracy in industrial robot control (Arimoto *et al.*, 1984; Norrlöf, 2002; Norrlöf and Gunnarsson, 2002). It utilizes a repetitive system dynamics to compensate for errors. Mathematically an ILC control law can be written as

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

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. \mathcal{Q} and \mathcal{L} are design parameters for the control law. In industrial robot systems the measured output does not correspond to the actual controlled output. An ILC experiment on the ABB IRB1400 in (Norrlöf, 2000, 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. 2.

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

$$u_{t,k+1} = \mathcal{Q}(u_{t,k} + \mathcal{L}\hat{\epsilon}_{t,k}). \quad (2)$$

Using the EKF, $\hat{\epsilon}_{t,k}$ represent the mean error, with an estimate of the covariance from the EKF. Hence, this

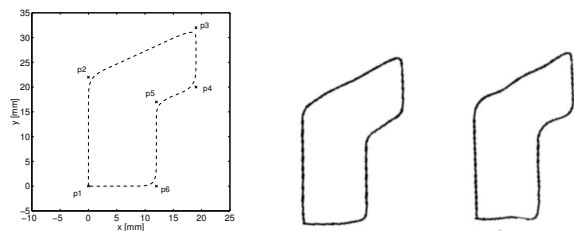


Fig. 2. Results from an ILC experiment on the ABB IRB1400 robot where ILC is applied using only motor angle measurements. Programmed path (left), iteration 0 (middle), and iteration 10 (right).

can be used in the improvement process and an idea in this direction is presented in (Norrlöf, 2002). 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 (Gunnarsson and Norrlöf, 2004) a 1 DOF lab-process is controlled using (2) but the estimation is simplified compared to the approach suggested in this paper due to the inherent linearity of the system. For the particle filter the whole *probability density function* (pdf) is available. Hence, the ILC improvement can be done in a more sophisticated way. The mean estimate or the *maximum likelihood* (ML) estimate, or some combination thereof are logical choices.

3. MODELS

In this section a continuous-time flexible 3 DOF robot model is presented. The model is simplified and transformed into discrete time where it can be used by the EKF and particle filter.

3.1 Robot Model

A general estimation problem consists of a nonlinear state equation and a nonlinear measurement relation (11) where the process noise, w_t , and measurement noise, e_t , are non-Gaussian. Often additive noises are assumed. 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. Here it will be assumed that the transmission can be described by a two mass system and that the manipulator is rigid and the parameters are from a mid size robot (larger and more flexible than the IRB1400).

The equation describing the torque balance for the motor becomes

$$M_m \ddot{q}_m = -f_m \dot{q}_m - r_g k (r_g q_m - q_a) - r_g d (r_g \dot{q}_m - \dot{q}_a) + u, \quad (3)$$

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 u . The corresponding relation for the arm becomes a nonlinear equation

$$M_a(q_a)\ddot{q}_a + C(q_a, \dot{q}_a)\dot{q}_a + g(q_a) = k(r_g q_m - q_a) + d(r_g \dot{q}_m - \dot{q}_a). \quad (4)$$

The goal is to estimate the arm position, q_a , by measuring the motor angle, q_m , and the Cartesian tool acceleration. The approach is similar to the one suggested in (Gunnarsson and Norrlöf, 2004), but the results presented here are more general, since a multi variable nonlinear system is considered.

An industrial robot has, in general, 6 DOF. However, here we will use only joint 1-3 (not the wrist joints). The following states are used

$$x_t = (q_{a,t} \ \dot{q}_{a,t} \ \ddot{q}_{a,t})^T, \quad (5)$$

where $q_{a,t} = (q_{a,t}^1 \ q_{a,t}^2 \ q_{a,t}^3)^T$ is the arm angle information from the first 3 axes in Fig. 1 and $\dot{q}_{a,t}$ is the angular velocity and $\ddot{q}_{a,t}$ is the angular acceleration at time t .

3.2 Estimation Model

We use the following simplified state space model in discrete time

$$x_{t+1} = f(x_t, u_t, w_t) = F_t x_t + G_{u,t} u_t + G_{w,t} w_t, \quad (6a)$$

$$y_t = h(x_t) + e_t, \quad (6b)$$

where

$$F_t = \begin{pmatrix} I & TI & T^2/2I \\ \mathcal{O} & I & TI \\ \mathcal{O} & \mathcal{O} & I \end{pmatrix}, G_{w,t} = \begin{pmatrix} T^3 I \\ \frac{6}{T} I \\ \frac{2}{T} I \end{pmatrix}, G_{u,t} = \begin{pmatrix} \frac{T^3}{2} I \\ \frac{6}{T} I \\ \frac{2}{T} I \end{pmatrix}$$

and T is the sampling time. I and \mathcal{O} are three by three unity and null matrices, and u_t is the derivative of the reference acceleration. The probability densities for the process noise, w_t , and measurement noise, e_t , are assumed to be Gaussian. The observation relation is described in Section 3.3.

3.3 Sensor Model

The observation relation is given by

$$h(x_t) = \begin{pmatrix} q_{m,t} \\ \ddot{\rho}_t \end{pmatrix}, \quad (7)$$

where $q_{m,t}$ is the measured motor angle and where $\ddot{\rho}_t$ is the Cartesian acceleration vector in the accelerometer frame, Fig. 1. The motor angle $q_{m,t}$ is computed from the arm angle using

$$q_m = r_g^{-1} \left(q_a + k^{-1} (M_a(q_a)\ddot{q}_a + g(q_a) + C(q_a, \dot{q}_a)\dot{q}_a + d(r_g \dot{q}_m - \dot{q}_a)) \right). \quad (8)$$

The kinematics (Sciavicco and Siciliano, 2000) of the robot is described by a nonlinear mapping $\rho_t = \mathcal{T}(q_{a,t})$, and its Jacobian is defined as

$$J(q_a) = \frac{\partial \mathcal{T}(q_a)}{\partial q_a}. \quad (9)$$

The following equation relates the Cartesian acceleration with the state variables

$$\ddot{\rho}_t = J(q_{a,t})\ddot{q}_{a,t} + \left(\sum_{i=1}^3 \frac{\partial J(q_{a,t})}{\partial q_{a,t}^{(i)}} \dot{q}_{a,t}^{(i)} \right) \dot{q}_{a,t}, \quad (10)$$

where $q_{a,t}^{(i)}$ is the i th element of $q_{a,t}$.

For the PF the nonlinear measurement relation can be applied directly. However, for the EKF it has to be linearized. In the model we do this symbolically, using a symbolic language and then auto-generate a linearized measurement function.

4. BAYESIAN ESTIMATION

Consider the discrete state-space model

$$x_{t+1} = f(x_t, u_t, w_t), \quad (11a)$$

$$y_t = h(x_t, e_t), \quad (11b)$$

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, (Jazwinski, 1970), is given by

$$p(x_{t+1}|\mathbb{Y}_t) = \int_{\mathbb{R}^n} p(x_{t+1}|x_t)p(x_t|\mathbb{Y}_t)dx_t, \quad (12a)$$

$$p(x_t|\mathbb{Y}_t) = \frac{p(y_t|x_t)p(x_t|\mathbb{Y}_{t-1})}{p(y_t|\mathbb{Y}_{t-1})}. \quad (12b)$$

These equations are in general not analytically solvable. However, for the important special case of linear-Gaussian dynamics and linear-Gaussian observations the Kalman filter, (Kalman, 1960), will give the solution. For a general nonlinear or non-Gaussian system, approximate methods must be applied. Here we will consider two different approaches of solving the Bayesian equations, extended Kalman filter (EKF), and particle filter (PF).

4.1 The Extended Kalman Filter (EKF)

For the special case of linear dynamics, linear measurements and additive Gaussian noise the Bayesian recursions in Section 4 have an analytical solution, the Kalman filter. For many nonlinear problems the noise assumptions are such that a linearized solution will be a good approximation. This is the idea behind the EKF, (Anderson and Moore, 1979), where the model

is linearized around the previous estimate. Here we only briefly present the time- and measurement update for the EKF,

$$\begin{cases} \hat{x}_{t+1|t} = f(\hat{x}_{t|t}, u_t), \\ P_{t+1|t} = F_t P_{t|t} F_t^T + G_t Q_t G_t^T, \end{cases} \quad (13a)$$

$$\begin{cases} \hat{x}_{t|t} = \hat{x}_{t|t-1} + K_t (y_t - h(\hat{x}_{t|t-1})), \\ P_{t|t} = P_{t|t-1} - K_t H_t P_{t|t-1}, \\ K_t = P_{t|t-1} H_t^T (H_t P_{t|t-1} H_t^T + R_t)^{-1}, \end{cases} \quad (13b)$$

where we use the linearized matrices

$$F_t = \nabla_x f(x_t, u_t)|_{x_t=\hat{x}_{t|t-1}}, \quad H_t = \nabla_x h(x_t)|_{x_t=\hat{x}_{t|t-1}}.$$

The noise covariances are given as

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

4.2 The Particle Filter (PF)

In this section the presentation of the particle filter theory is according to (Bergman, 1999; Doucet *et al.*, 2001; Gordon *et al.*, 1993; Gustafsson *et al.*, 2002). The particle filter provides an approximate solution to the discrete time Bayesian estimation problem formulated in (12) 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 particle filter 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 particle filter updates the particle location and the corresponding weights recursively with each new observed measurement. For the common special case of additive measurement noise, i.e.,

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

the unnormalized weights are given by

$$\gamma_t^{(i)} = p_e(y_t - h(x_t^{(i)})), \quad i = 1, \dots, N. \quad (15)$$

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), (Gordon *et al.*, 1993), and is summarized in Algorithm 1. As the estimate for each time we chose the minimum mean square estimate.

4.3 Cramér-Rao Lower Bound (CRLB)

When different estimators are used it is fundamental to know the best possible achievable performance. The Cramér-Rao lower bound is such a characteristic for the second order moment. Here we will only

Algorithm 1 Sampling Importance Resampling

- 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 $\bar{\gamma}_t^{(j)} = Pr\{x_t^{(i*)} = x_t^{(j)}\}$.
 - 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 iterate to step 2.
-

consider state-space models with additive noise and present the CRLB. The theoretical posterior CRLB for a general dynamic system was derived in (Bergman, 1999; Doucet *et al.*, 2001). Hence, we can for the simulation study evaluate these expressions around the true trajectory and calculate the covariance by iterating the Riccati equation (covariance expressions in (13)) for each time step until a stationary value is found. This corresponds to a lower bound on the covariance. Taking the square root of the trace for relevant states, this limit can be compared with the RMSE from Monte Carlo simulations.

The true system should be used in the CRLB evaluation, and therefore the discrete system equation must be calculated. This can be achieved by studying the continuous-time system from Section 3.1. The system can be written in state space form as

$$\dot{x} = \frac{d}{dt} \begin{pmatrix} q_a \\ \dot{q}_a \\ \ddot{q}_a \end{pmatrix} = \begin{pmatrix} \dot{q}_a \\ \ddot{q}_a \\ \alpha(q_a, \dot{q}_a, \ddot{q}_a) \end{pmatrix}, \quad (16)$$

where $\alpha(q_a, \dot{q}_a, \ddot{q}_a)$ is given by the time derivative of \ddot{q}_a given by (4). The system is too complicated for a symbolic differentiation. However, a numerical differentiation can be done around the true trajectory. The desired discrete time system matrix is now given by first linearization and then discretization, (Gustafsson, 2000)

$$A^c = \nabla_x f(x) = \begin{pmatrix} \mathcal{O} & I & \mathcal{O} \\ \mathcal{O} & \mathcal{O} & I \\ \frac{\partial \alpha(q, \dot{q}, \ddot{q})}{\partial q} & \frac{\partial \alpha(q, \dot{q}, \ddot{q})}{\partial \dot{q}} & \frac{\partial \alpha(q, \dot{q}, \ddot{q})}{\partial \ddot{q}} \end{pmatrix},$$

$$F = e^{A^c T}.$$

In Section 5 the CRLB is compared to the RMSE from Monte Carlo simulations, both with and without accelerations measurements.

5. SIMULATION RESULT

The model is implemented and simulated using the Robotics Toolbox (Corke, 1996) in Simulink and it is the same model as in (Svensson, 2004). The robot is stabilized using a PID-controller.

In Fig. 3 the first four terms of (8) are shown for the data in the simulation. The damping term is not shown since it is considered negligible compared to the other terms. From Fig. 3 the importance of the different

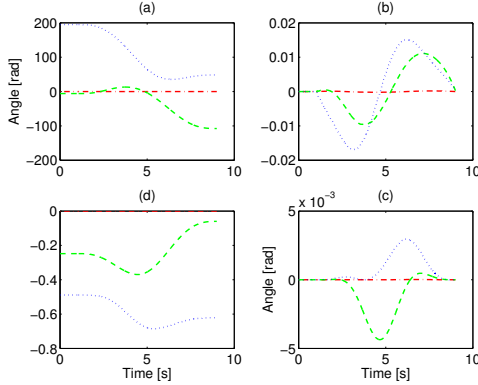


Fig. 3. Terms in the measurement equation, (8). (a) $r_g^{-1}q_a$, (b) $r_g^{-1}k^{-1}M_a(q_a)\ddot{q}_a$, (c) $r_g^{-1}k^{-1}g(q_a)$, and (d) $r_g^{-1}k^{-1}C(q_a, \dot{q}_a)\dot{q}_a$, (dash-dotted) axis 1, (dashed) axis 2, and (dotted) axis 3.

terms can be concluded. The term containing q_a is fundamental, the gravitational term is also important since it gives a bias to the estimate. The inertia term also contributes together with the Coriolis term. The 5th term, the damping, has been neglected.

The simulation study is based mainly around the EKF approach, since it is a fast method well suited for large Monte Carlo simulations. The particle filter is much slower so we will only include a smaller Monte Carlo study as well as some concluding remarks. The Monte Carlo simulations use the following Covariance matrices for the process and measurement noise

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

We simulate the system around the nominal trajectory and produce 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.

EKF. In Fig. 4 the RMSE from Monte Carlo simulations are compared with the CRLB limit, both with and without acceleration measurements. As seen the RMSE is close to 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. We also note that the accelerometer measurements reduces the estimation uncertainty. The results in Fig. 4 is of course for the chosen trajectory. The acceleration values are not that large, so greater differences occur for larger accelerations. On a 1.5 GHz PC running MATLAB the EKF performs in real-time on the 100 Hz data rate.

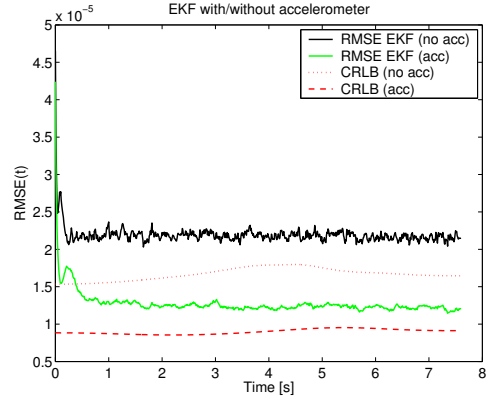


Fig. 4. EKF 500 MC-simulations without/with accelerometer.

PF. The particle filter is rather slow compared to the EKF for this model structure. The MATLAB implementation of the system is not well suited for large Monte Carlo simulations. Instead we focus on a small study over a much shorter time period than for the EKF case. We compare the particle filter and the EKF and show a small performance improvement. The result is given in Fig. 5. Even though the particle fil-

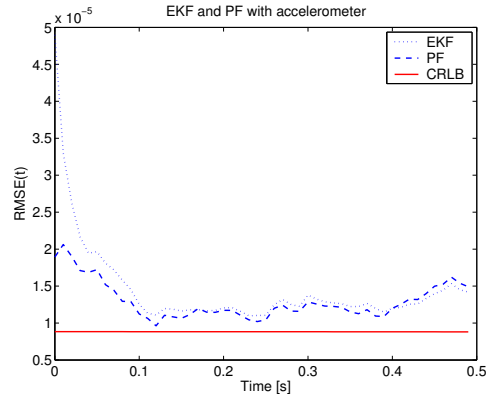


Fig. 5. EKF and PF position RMSE with external accelerometer signal from 20 Monte Carlo simulations.

ter is slow, it gives more insight in 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 another more complicated system dynamics must be implemented in the filter.

6. CONCLUSIONS

A multiple sensor fusion approach to find estimates of the tool position by combining a 3-axis accelerometer and the measurements from the motor angles of an industrial robot is presented. We formulate the position estimation as a Bayesian problem and propose two solutions, EKF and PF 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 CRLB limit. Estimation performance with the accelerometer is better, considering both the CRLB and the actual result from the MC-simulations. 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 EKF runs in real-time in MATLAB. It is left for future work to evaluate the robustness of the estimation algorithm and also to do the experiment on the robot using a highly accurate position measurement system for evaluation.

REFERENCES

- Anderson, B.D.O. and J. B. Moore (1979). *Optimal Filtering*. Prentice Hall. Englewood Cliffs, NJ.
- Arimoto, S., S. Kawamura and F. Miyazaki (1984). Bettering operation of robots by learning. *Journal of Robotic Systems* **1**(2), 123–140.
- Bergman, N. (1999). Recursive Bayesian Estimation: Navigation and Tracking Applications. Linköping Studies in Science and Technology. Dissertations No. 579. Linköping University, Linköping, Sweden.
- Corke, P. I. (1996). A robotics toolbox for MATLAB. *IEEE Robot. Automat. Mag.* **3**(1), 24–32.
- Doucet, A., de Freitas, N. and Gordon, N., Eds. (2001). *Sequential Monte Carlo Methods in Practice*. Springer Verlag.
- Gordon, N. J., D. J. Salmond and A.F.M. Smith (1993). A novel approach to nonlinear/non-Gaussian Bayesian state estimation. In: *IEE Proceedings on Radar and Signal Processing*. Vol. 140. pp. 107–113.
- Gunnarsson, S. and M. Norrlöf (2004). Iterative learning control of a flexible robot arm using accelerometers. In: *IEEE Conference on Control Applications*. Taipei, Taiwan.
- Gunnarsson, S., M. Norrlöf, G. Hovland, U. Carlsson, T. Brogårdh, T. Svensson and S. Moberg (2001). Pathcorrection for an industrial robot. European Patent Application No. EP1274546.
- Gustafsson, F. (2000). *Adaptive Filtering and Change Detection*. John Wiley & Sons Ltd.
- Gustafsson, F., F. Gunnarsson, N. Bergman, U. Forsell, J. Jansson, R. Karlsson and P-J Nordlund (2002). Particle filters for positioning, navigation and tracking. *IEEE Trans. Signal Processing* **50**, 425–437.
- Jassemi-Zargani, R. and D. Neacsulescu (2002). Extended Kalman filter-based sensor fusion for operational space control of a robot arm. *IEEE Trans. Instrum. Meas.* **51**(6), 1279 – 1282.
- Jazwinski, A. H. (1970). *Stochastic Processes and Filtering Theory*. Vol. 64 of *Mathematics in Science and Engineering*. Academic Press.
- Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Transactions of the AMSE—Journal of Basic Engineering* **82**, 35–45.
- Karlsson, R. and M. Norrlöf (2004). Bayesian position estimation of an industrial robot using multiple sensors. In: *Proceedings of the IEEE Conference on Control Applications*. Taipei, Taiwan.
- Moore, K. L. (1993). *Iterative Learning Control for Deterministic Systems*. Advances in Industrial Control. Springer-Verlag. London.
- Nicosia, S., P. Tomei and A. Tornambe (1988). A nonlinear observer for elastic robots. **4**(1), 45–52.
- Norrlöf, M. (2000). Iterative Learning Control – Analysis, design and experiments. PhD thesis. Linköping University, Linköping, Sweden. Linköping Studies in Science and Technology. Dissertations No. 653.
- Norrlöf, M. (2002). An adaptive iterative learning control algorithm with experiments on an industrial robot. *IEEE Trans. Robot. Automat.* **18**(2), 245–251.
- Norrlöf, M. and S. Gunnarsson (2002). Experimental comparison of some classical iterative learning control algorithms. *IEEE Trans. Robot. Automat.* **18**(4), 636–641.
- Sciavicco, L. and B. Siciliano (2000). *Modelling and Control of Robot Manipulators*. Springer.
- Svensson, C. (2004). Multivariable control of industrial robot. Masters Thesis LiTH-ISY-EX-3506-2004. Department of Electrical Engineering. Linköping University. In Swedish.