

# **A Simulation Study on the Arm Estimation of a Joint Flexible 2 DOF Robot Arm**

Patrik Axelsson

Division of Automatic Control

E-mail: [axelsson@isy.liu.se](mailto:axelsson@isy.liu.se)

8th December 2009

Report no.: LiTH-ISY-R-2926

Address:

Department of Electrical Engineering

Linköpings universitet

SE-581 83 Linköping, Sweden

WWW: <http://www.control.isy.liu.se>

AUTOMATIC CONTROL  
REGLERTEKNIK  
LINKÖPINGS UNIVERSITET



## **Abstract**

The main task for an industrial robot is to move the tool into specific positions. It is therefore necessary to have an accurate knowledge about the tool position. This report describes a simulation study where an accelerometer attached to the robot tool is used. The acceleration and measured motor angles are used with an Extended Kalman Filter (EKF) to estimate the tool position. The work has been focused on a robot with two degrees of freedom. Simulations have been performed with different kind of errors and on different paths. The EKF uses covariance matrices of the process noise and measurement noise which are unknown. An optimization problem has therefore been proposed and solved to get covariance matrices that give good estimations.

**Keywords:** Extended Kalman Filter, Industrial manipulator, Accelerometer

# A Simulation Study on the Arm Estimation of a Joint Flexible 2 DOF Robot Arm

Patrik Axelsson

2009-12-08

## Abstract

The main task for an industrial robot is to move the tool into specific positions. It is therefore necessary to have an accurate knowledge about the tool position. This report describes a simulation study where an accelerometer attached to the robot tool is used. The acceleration and measured motor angles are used with an Extended Kalman Filter (EKF) to estimate the tool position. The work has been focused on a robot with two degrees of freedom. Simulations have been performed with different kind of errors and on different paths. The EKF uses covariance matrices of the process noise and measurement noise which are unknown. An optimization problem has therefore been proposed and solved to get covariance matrices that give good estimations.

## 1 Introduction

The problem is to estimate the tool position for a flexible manipulator. The manipulator is a resonant system with uncertainties in the model parameters. There are also high demands on the accuracy of the estimation. In the past when the robots were more rigid than today it was enough to measure the motor angles and use kinematic models. Nowadays this is not enough due to flexibilities in the structure. Earlier work, see [3] and [4], using experimental data have shown that the estimation is good for frequencies from 3 to 30 Hz but not so good for lower frequencies. The aim is therefore to improve the estimation in the low frequency range. The work is limited to a 2 DOF manipulator and uses only simulated data.

The report describes the robot model and the accelerometer model in Section 2 and the observer in Section 3. The simulation setup is described in Section 4 and the result is presented in Section 5. Conclusions and future work is given in Sections 6 and 7.

## 2 Mathematical models

This section presents the mathematical models for the robot and the accelerometer. The equations for the observer are presented in Section 3.

## 2.1 Robot model

The robot model is a joint flexible two-axes model from [5], see Figure 2.1, that includes both nonlinear stiffness and friction in the joints. Let

$$q = \begin{pmatrix} q_a \\ q_m \end{pmatrix}, \quad u = \begin{pmatrix} 0 \\ u_m \end{pmatrix}, \quad (2.1)$$

where  $q_a$  and  $q_m$  are the arm and motor angles and  $u_m$  is the motor torque. A dynamic equation can then be derived as

$$M(q)\ddot{q} + C(q, \dot{q}) + G(q) + \tau_s(q) + D(\dot{q}) + \kappa(\dot{q}) = u. \quad (2.2)$$

Here  $M(q)$  is the inertia matrix,  $C(q, \dot{q})$  is the Coriolis- and centrifugal terms,  $G(q)$  is the gravitational torque,  $\tau_s(q)$  is the nonlinear stiffness torque,  $D(\dot{q})$  is the damping torque and  $\kappa(\dot{q})$  is the nonlinear friction torque. A state space model can be derived from (2.2) as

$$\begin{aligned} \dot{x} &= \begin{pmatrix} x_3 \\ x_4 \\ M^{-1}(x_1)(u - C(x_1, x_3) - G(x_1) - D(x_3, x_4) - \tau_s(x_1, x_2) - \kappa(x_4)) \end{pmatrix} \\ &= f(x, u), \end{aligned} \quad (2.3)$$

where

$$x = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} q_a \\ q_m \\ \dot{q}_a \\ \dot{q}_m \end{pmatrix}. \quad (2.4)$$

The equations are expressed on the arm side. This means that  $q_m = \frac{\tilde{q}_m}{\eta}$ ,  $u_m = \tilde{u}_m \eta$  where  $\eta$  is the gear ratio and  $\tilde{q}_m$  and  $\tilde{u}_m$  are the motor angle and motor torque expressed on the motor side of the gearbox.

See [3] and [5] for more details.

## 2.2 Accelerometer model

The accelerometer measures the acceleration in a frame  $\{s\}$  fixed to the sensor. The measured acceleration is

$$\ddot{\rho}_s^M = \ddot{\rho}_s + R_s^w(q_a)G_w + \delta_s + e_s, \quad (2.5)$$

where  $\ddot{\rho}_s$  is the acceleration due to the motion,  $G_w = (0 \ 0 \ -g)^T$  models the gravitation in the world frame  $\{w\}$ ,  $\delta_s$  models the drift and  $e_s$  is the measurement noise, both expressed in  $\{s\}$ .  $R_s^w(q_a)$  is a rotation matrix that represents the transformation from frame  $\{w\}$  to frame  $\{s\}$ .

Both the simulation model and the observer must have a mathematical expression for  $\ddot{\rho}_s$ . First a vector from the origin in frame  $\{w\}$  to the origin in frame  $\{s\}$ , see Figure 2.2, is defined as

$$\rho_w = \begin{pmatrix} x(q_a) \\ z(q_a) \end{pmatrix} = \Gamma(q_a), \quad (2.6)$$

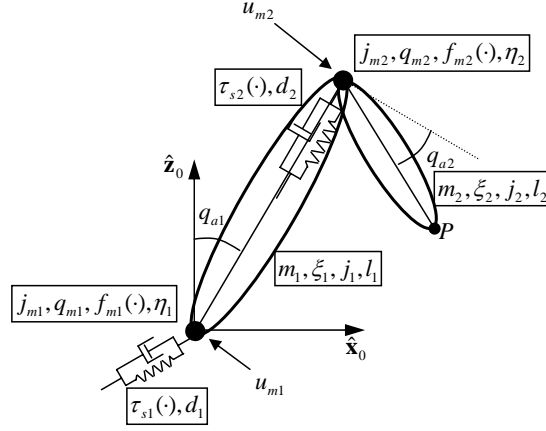


Figure 2.1: Serial robot with two degrees of freedom. The kinematic and the dynamic equations are derived in [5].

where  $\Gamma$  is a system of nonlinear equations for the robot kinematics. The derivative of (2.6) with respect to time gives

$$\dot{\rho}_w = \dot{\Gamma}(q_a) = J(q_a)\dot{q}_a, \quad (2.7a)$$

$$\ddot{\rho}_w = \ddot{\Gamma}(q_a) = J(q_a)\ddot{q}_a + \dot{J}(q_a)\dot{q}_a, \quad (2.7b)$$

where  $J(q_a) = \frac{\partial \Gamma}{\partial q_a}$  is the jacobian matrix. The vector  $\ddot{\rho}_w$  is then transformed from frame  $\{w\}$  to frame  $\{s\}$  according to

$$\ddot{\rho}_s = R_s^w(q_a)\ddot{\rho}_w. \quad (2.8)$$

It is now possible to calculate the acceleration caused by the motion of the robot in different positions. Insert (2.8) in (2.5) and the final expression for the measured acceleration can be expressed as

$$\ddot{\rho}_s^M = R_s^w(q_a)(\ddot{\rho}_w + G_w) + \delta_s + e_s. \quad (2.9)$$

More about the accelerometer modeling can be found in [3].

### 3 Observer

An Extended Kalman Filter (EKF), see [1], is used as the observer. The EKF is a suboptimal filter that linearizes the system equation and the measurement equation around the estimated states at every time step before a Kalman filter is applied.

#### 3.1 System and Measurement Equations

The continuous-time model  $f(x, u)$  in (2.3) is first discretized before the EKF is used. By using a first order Taylor approximation, we get

$$x_{k+1} = F(x_k, u_k) + v_k, \quad (3.1a)$$

$$F(x_k, u_k) = x_k + T_s \dot{x}_k = x_k + T_s f(x_k, u_k), \quad (3.1b)$$

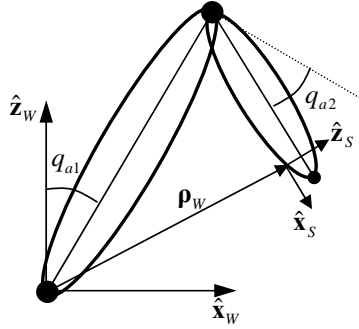


Figure 2.2: The vector  $\rho_w$  is a vector from the origin of frame  $\{w\}$  to the origin of frame  $\{s\}$ .

where  $T_s$  is the sample time and  $v_k$  is process noise. This approximation can introduce some errors but is used for its simplicity. Measurements are available of the motor angles and the acceleration of the sensor in frame  $\{s\}$ . This can be expressed as

$$z_k = h(x_k, u_k) + w_k = \begin{pmatrix} x_{2k} \\ R_s^w(x_{1k})(\ddot{\rho}_w + G_w) \end{pmatrix} + w_k, \quad (3.2)$$

where  $w_k$  is measurement noise. The drift is included in the accelerometer model (2.8) but not in the measurement equation (3.2). This means that the EKF can not handle the drift and make a good estimation. Therefore, let

$$b_{k+1} = b_k + v_k, \quad (3.3)$$

be a dynamic model for the drift where  $v_k$  is noise. This model says that the drift is a random walk that only changes with the noise. Augmenting the state space vector (2.4) with  $b_k$  gives

$$x = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ b \end{pmatrix} = \begin{pmatrix} q_a \\ q_m \\ \dot{q}_a \\ \dot{q}_m \\ b \end{pmatrix}. \quad (3.4)$$

A new model and measurement equation are now obtained as

$$x_{k+1} = \begin{pmatrix} F(x_k, u_k) \\ x_{5k} \end{pmatrix} + v_k, \quad (3.5)$$

$$z_k = h(x_k, u_k) + w_k = \begin{pmatrix} x_{2k} \\ R_s^w(x_1)(\ddot{\rho}_w + G_w) + x_{5k} \end{pmatrix} + w_k. \quad (3.6)$$

Here the new state for the drift is included in the measurement equation and the observer should then be able to handle the drift. It remains to Section 5 to see whether this new model description is better or not, but according to [2] this is a common way to handle bias in sensors. The EKF based on (3.5) and (3.6) is called AEKF in the rest of this report.

## 3.2 EKF algorithm

The EKF is a recursive algorithm based on a measurement update and a time update.

### Measurement update

$$K_k = P_{k|k-1} H_k^T (H_k P_{k|k-1} H_k^T + R_k)^{-1} \quad (3.7a)$$

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k (z_k - h(\hat{x}_{k|k-1}, u_k)) \quad (3.7b)$$

$$P_{k|k} = P_{k|k-1} - K_k H_k P_{k|k-1} \quad (3.7c)$$

### Time update

$$\hat{x}_{k+1|k} = F(\hat{x}_{k|k}, u_k) \quad (3.8a)$$

$$P_{k+1|k} = A_k P_{k|k} A_k^T + Q_k \quad (3.8b)$$

### Initial values

$$\hat{x}_{1|0} = x_0 = (q_{m1}(0) \quad q_{m2}(0) \quad q_{m1}(0) \quad q_{m2}(0) \quad 0 \quad 0 \quad 0 \quad 0)^T \quad (3.9a)$$

$$P_{1|0} = P_0 \quad (3.9b)$$

Here is  $P$  the covariance matrix for the estimation error and  $K$  is the filter gain. The matrices  $A_k$  and  $H_k$  are obtained from a linearization of (3.1a) and (3.2) around the estimated states according to

$$x_{k+1} = F(\hat{x}_{k|k}, u_k) + A_k (x_k - \hat{x}_{k|k}) + v_k, \quad (3.10a)$$

$$z_k = h(\hat{x}_{k|k-1}, u_k) + H_k (x_k - \hat{x}_{k|k-1}) + w_k, \quad (3.10b)$$

where

$$A_k = \left. \frac{\partial F(x, u_k)}{\partial x} \right|_{x=\hat{x}_{k|k}}, \quad (3.11a)$$

$$H_k = \left. \frac{\partial h(x, u_k)}{\partial x} \right|_{x=\hat{x}_{k|k-1}}. \quad (3.11b)$$

For the AEKF (3.5) and (3.6) are linearized instead. In (3.7a) and (3.8b)  $R_k$  and  $Q_k$  denote the measurement and process noise at time  $k$ . These matrices are assumed to be constant over time in the rest of this report.

## 3.3 Tuning of covariance matrices

In order to use the EKF one must choose good estimates of the covariance matrices  $Q$  and  $R$ . This has been done in an automatic way in [3]. The same approach has been used in this work with some changes. Firstly, the optimization algorithm has been changed from Complex-RF to an Active Set method (*fmincon* in MATLAB). Secondly, the path error is calculated in a different way. Here the path error is calculated as

$$e_k = \min_i \sqrt{|p_{x,k} - \hat{p}_{x,i}|^2 + |p_{z,k} - \hat{p}_{z,i}|^2}, \quad (3.12)$$

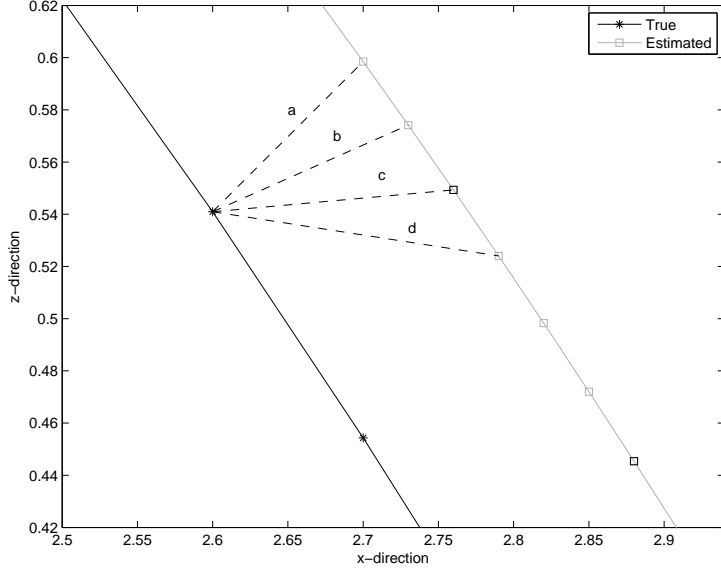


Figure 3.1: Geometric interpretation of the path error. The stars and black squares indicate the original data points and the gray squares are interpolated data points.

where  $p_{x,k}$ ,  $\hat{p}_{x,i}$ ,  $p_{z,k}$  and  $\hat{p}_{z,i}$  are the true and estimated position for the tool in x- and z-direction at time  $k$  and time  $i$ , respectively. The path error is the shortest distance between the true path and the estimated path for every data point in the true path. Figure 3.1 shows how (3.12) should be interpreted, and it is easy to see that the shortest distance is  $b$ . The stars and the black squares in Figure 3.1 are the original data points in the true and estimated paths, respectively. Equation (3.12) eliminates the time dependence and only takes the geometrical properties into consideration. A cubic spline interpolation is performed to obtain the data points indicated with gray squares.

The problem is now to minimize the L2-norm of the path error (3.12) with respect to the covariance matrices and it can be formulated as

$$\text{Minimize } f_{obj}(\hat{p}_x, \hat{p}_z) = \sqrt{\sum_{k=1}^N |e_k|^2}$$

$$\text{subject to } \lambda_j > 0 \quad j = 1, \dots, 5$$

$$\tilde{Q}_\lambda = \begin{pmatrix} \lambda_1 I_{2 \times 2} & 0 & 0 & 0 \\ 0 & \lambda_2 I_{2 \times 2} & 0 & 0 \\ 0 & 0 & \lambda_3 I_{2 \times 2} & 0 \\ 0 & 0 & 0 & \lambda_4 I_{2 \times 2} \end{pmatrix} \tilde{Q}$$

$$\tilde{R}_\lambda = \begin{pmatrix} \lambda_5 I_{2 \times 2} & 0 \\ 0 & I_{2 \times 2} \end{pmatrix} \tilde{R}$$

$$(\hat{p}_x, \hat{p}_z) = \text{EKF}(\tilde{Q}_\lambda, \tilde{R}_\lambda)$$



where  $\lambda_j$  are the optimization parameters.  $\tilde{Q}$  and  $\tilde{R}$  are diagonal matrices with the elements taken from the covariances of the process noise  $v$  and the measurement noise  $w$ . The noise is estimated as

$$\hat{v}_k = x_{k+1} - F(x_k, u_k), \quad (3.13)$$

$$\hat{w}_k = z_k - h(x_k, u_k). \quad (3.14)$$

It is possible to estimate the noise like this since this work is performed on simulated data where the true states are available. The process noise includes both discretization errors and measurement noise that has been filtered through the controller. The EKF does not use the cross-correlation between the states and the cross-correlation between the measurements since only the diagonal elements are used.

The matrices  $Q$  and  $R$  in the optimization problem are restricted to the diagonal elements to simplify the optimization.

## 4 Simulation Setup

The simulations were performed on four different paths as can be seen in Figure 4.1. The path starts at the star and goes clockwise. The circle indicates the tool position in the zero-position of the robot and the thick line indicates which part of the path that has been magnified in later figures. The reference signals for these four paths were created using a standard ABB controller. Figure 4.2 shows the location of the four paths relative to the zero-position of the robot. Four simulations according to Table 4.1 were performed on all four paths to cover model errors, drift, and calibration errors. How realistic these errors are can be questioned. The calibration errors in the x- and z-direction may be a bit large, it would be possible to place the sensor more accurate. The error for the orientation is more difficult to say something about. An error of  $2^\circ$  is chosen because it is small enough so that the sensor can look like being straight. The model errors can be seen as the worst case and are chosen based on suggestions from the authors of [5].

The true path for Sim2 is different compared to the other simulations due to model errors in the controller, compare Figures 5.2 to 5.4. Some of the simulations were then used to optimize the covariance matrices used in the EKF and the AEKF, which can be seen in Table 4.2. Different combinations of the simulations and the matrices were then used to estimate the tool position for all four paths.

## 5 Result

In this work the largest path errors for the estimations are less than 3 mm. In Figure 5.1 the whole path is shown. As can be seen it is impossible to distinguish the true path and the estimates. Figures 5.2 to 5.5 show therefore a fraction of the path. Figure 5.2 shows that the EKF for Cov1, Cov2 and Cov3 are very similar to the true paths which is not odd since Sim1 is without errors. Figures 5.3 to 5.5 where calibration errors, drift and model errors are present show instead that the estimations differ from the true paths. The estimations differ more from the true path in Figure 5.3 than Figures 5.4 and 5.5 due to

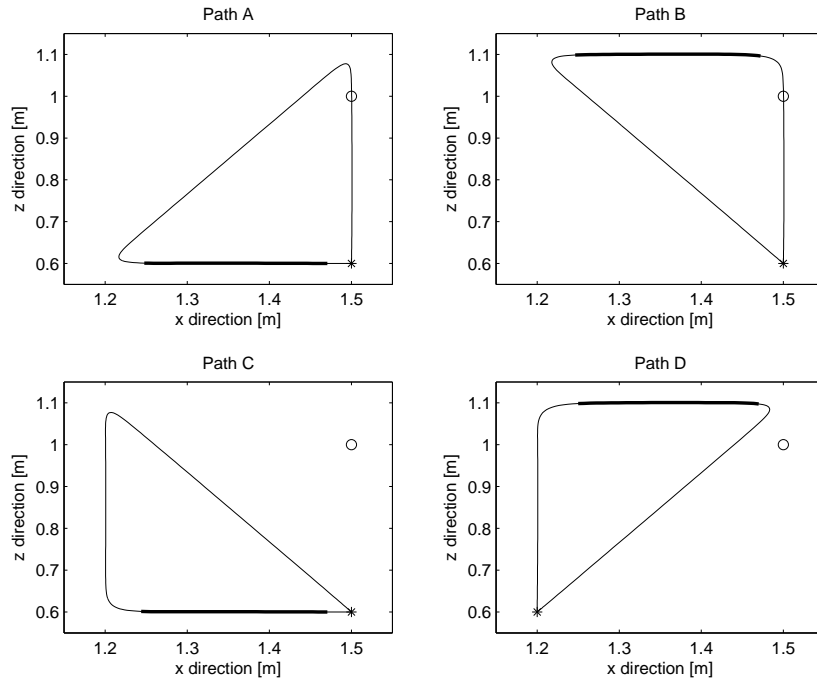


Figure 4.1: References for the four paths that has been simulated. The path starts at the star and the robot moves clockwise. The circle indicates the tool position for the zero-position. The thicker segment of the path shows which part that is magnified during the evaluation.

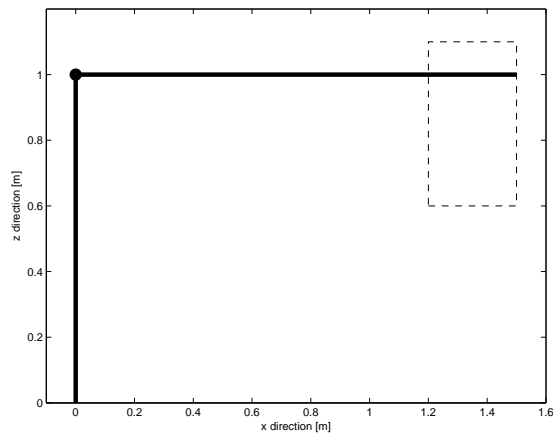


Figure 4.2: The location of the four paths relative the zero-position of the robot. The dashed rectangle indicates the location of the four paths.

Name	Description
Sim1	Without calibration errors, drift and model errors.
Sim2	With calibration errors (4 mm in x-direction, -5 mm in z-direction and 2° in orientation), drift (0.1 m/s <sup>2</sup> in both directions) and model errors (20% in stiffness parameters and 50% in friction parameters).
Sim3	With calibration errors (4 mm in x-direction, -5 mm in z-direction and 2° in orientation), drift (0.1 m/s <sup>2</sup> in both directions) and without model errors.
Sim4	With calibration errors (4 mm in x-direction, -5 mm in z-direction and 2° in orientation), drift (0.2 m/s <sup>2</sup> in both directions) and without model errors.

Table 4.1: Four different simulation scenarios used to evaluate the observers.

Name	Description
Cov1	Covariance matrices for the EKF tuned on path A for Sim1
Cov2	Covariance matrices for the EKF tuned on path A for Sim2
Cov3	Covariance matrices for the EKF tuned on path A for Sim3
Cov4	Covariance matrices for the AEKF tuned on path A for Sim2
Cov5	Covariance matrices for the AEKF tuned on path A for Sim3

Table 4.2: Five different covariance matrices used to evaluate the observers.

model errors. Model errors is thus a big problem which in practice is inevitable. The drift in Figure 5.5 is twice the drift in Figure 5.4 but the estimation is still very similar. It could imply that the estimation is robust for changes in the drift but a more detailed study must be performed to conclude it and is therefore left for future work.

It can also be seen that the EKF produces the same type of estimation error independent of which covariance matrices that are used. It is only the magnitude that changes. This can be seen better in Figure 5.6 where the path error is shown as a function of time for the AEKF on Sim2. It is easy to see that the path errors oscillate in the same way but with different magnitudes. The path error for the other simulations gives the same behavior both for the EKF and the AEKF and is therefore not included here.

Figure 5.7 shows how the bias parameter  $b$  in (3.4), (3.5) and (3.6) changes over time for Sim3. The parameters oscillate more for the covariance matrices optimized on Sim3 (Cov5) compared to the one optimized on Sim2 (Cov4). But the parameters converge to the same value in both cases. Since the drift is constant and known in the simulations, 0.1 m/s<sup>2</sup> or 0.2 m/s<sup>2</sup>, the bias parameters should converge to that value but that is not the case here. The bias parameters for Sim2 and Sim4 show the same behavior and is therefore not included here.

Tables 5.1 to 5.4 show maximal and mean of the path error for the EKF and Tables 5.5 to 5.8 show the same for the AEKF. The smallest maximal error is indicated with bold numbers and the smallest mean error is indicated with italic numbers. We can see that the error is larger for path B, C and D. This could imply that the covariance matrices are dependent of the states. But it would require a more thorough study with motions in all possible areas of the

workspace to conclude this and is therefore left for future work. We can also see that the AEKF does not reduce the estimation error.

Another important conclusion, not seen in these tables, is that the estimation changes with the optimization of the covariance matrices. Table 5.1 show that Cov1 gives minimum values for Sim1, Cov2 gives minimum values for Sim2, and Cov3 gives minimum values for Sim3. This would be an obvious result since Cov1 is optimized for Sim1, Cov2 is optimized for Sim2, and Cov3 is optimized for Sim3 but this is not always correct. Cov3 gives sometimes the best estimation on Sim1 although Cov1 is optimized for Sim1 which would give a minimum. This concludes that the optimization gives a local minimum.

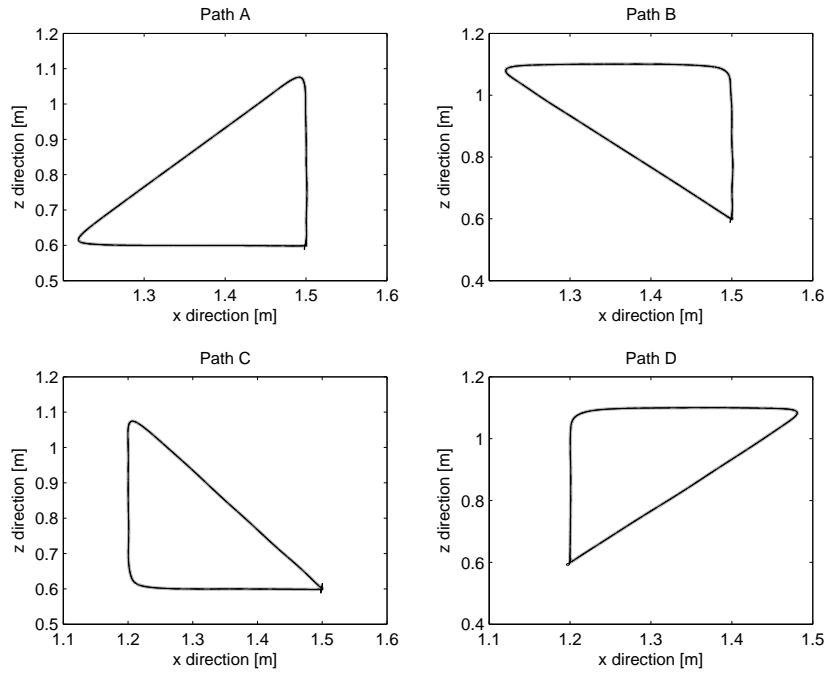


Figure 5.1: Estimation of path A, B, C and D on Sim1 with Cov1 (-), Cov2 (--) and Cov3 (-.). The grey line is the true path.

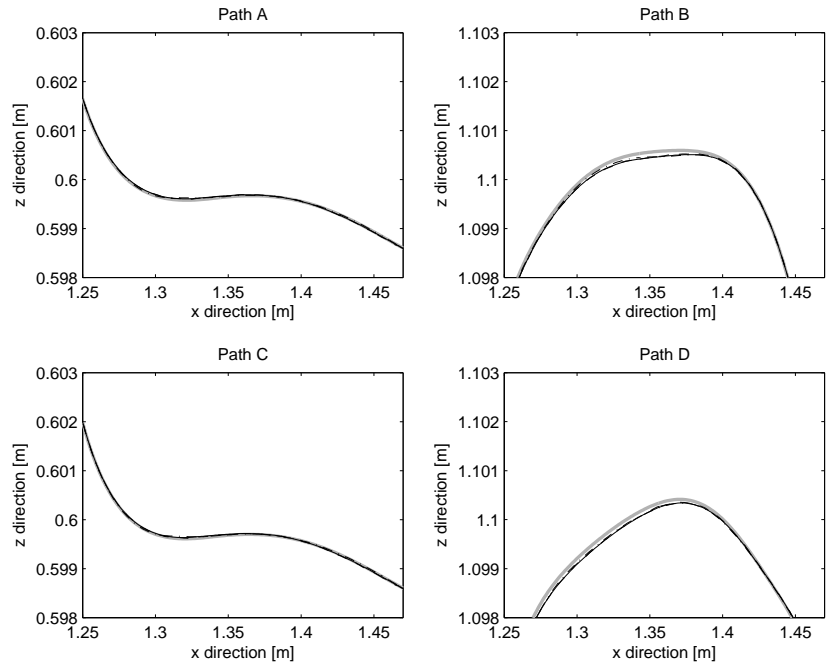
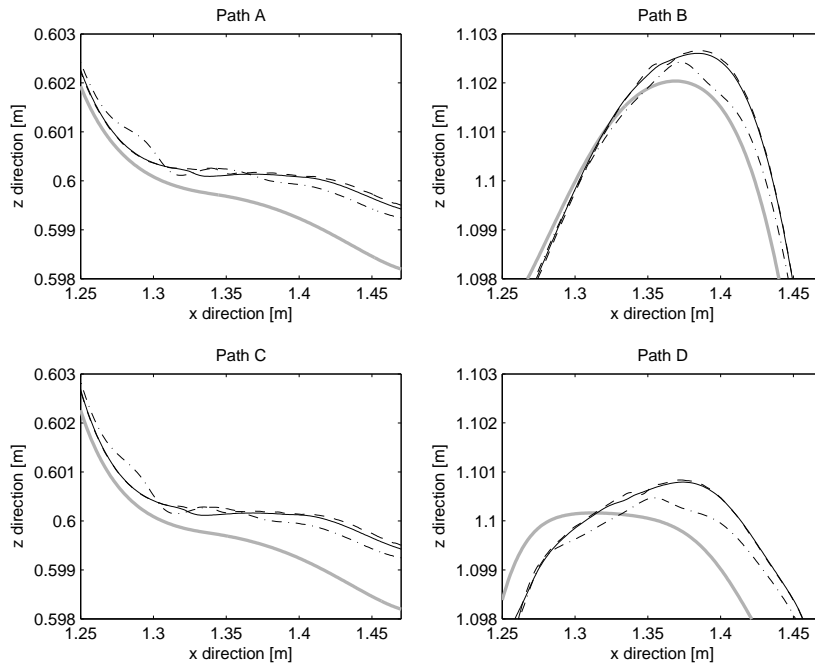
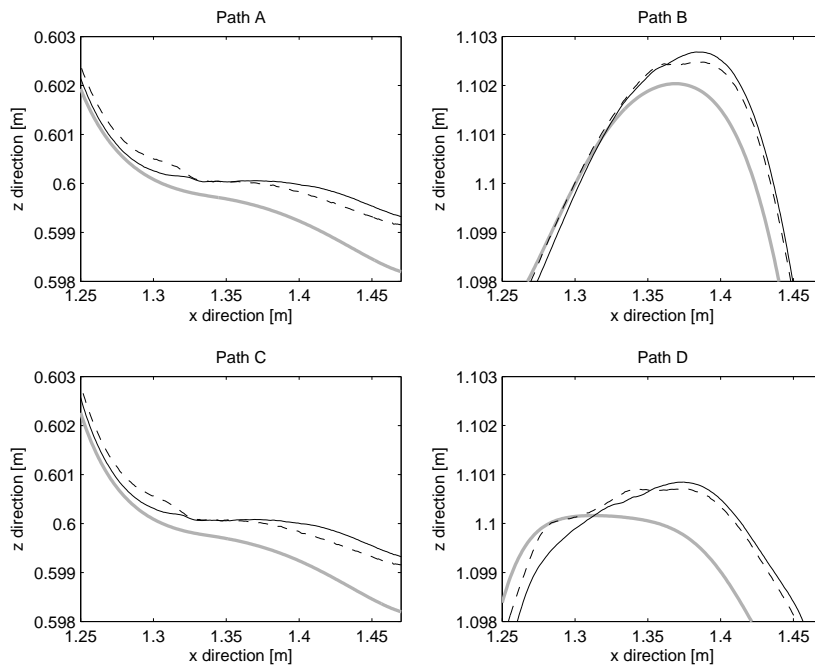


Figure 5.2: Magnified estimation of path A, B, C and D on Sim1 with Cov1 (-), Cov2 (--) and Cov3 (-.). The grey line is the true path.

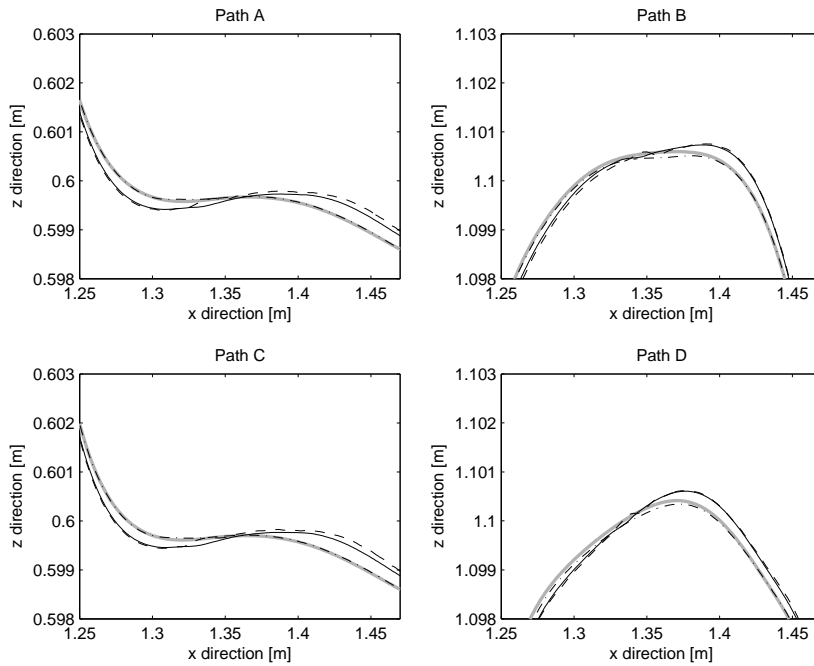


(a) EKF. Cov1 (-), Cov2 (--), and Cov3 (-.).

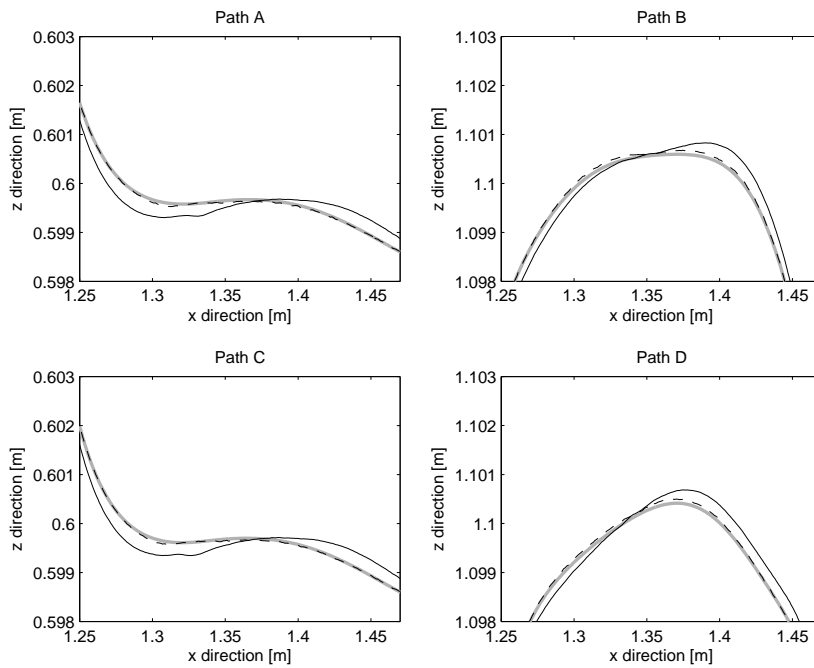


(b) AEKF. Cov4 (-) and Cov5 (--).

Figure 5.3: Magnified estimation of path A, B, C and D on Sim2 for EKF (a) and AEKF (b). The grey line is the true path.



(a) EKF. Cov1 (-), Cov2 (--) and Cov3 (-).



(b) AEKF. Cov4 (-) and Cov5 (--).

Figure 5.4: Magnified estimation of path A, B, C and D on Sim3 for EKF (a) and AEKF (b). The grey line is the true path.

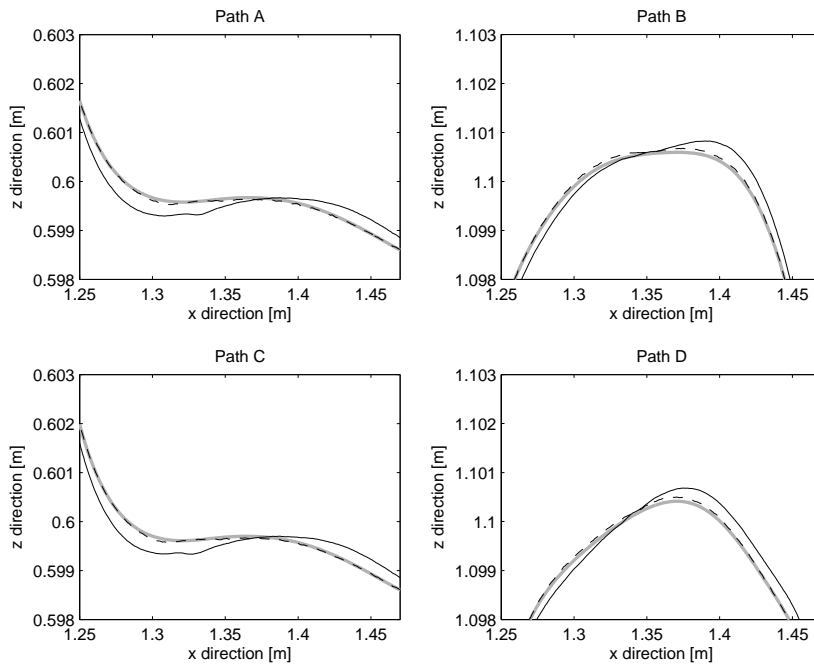


Figure 5.5: Magnified estimation of path A, B, C and D on Sim4 with Cov4 (-) and Cov5 (--). The grey line is the true path.

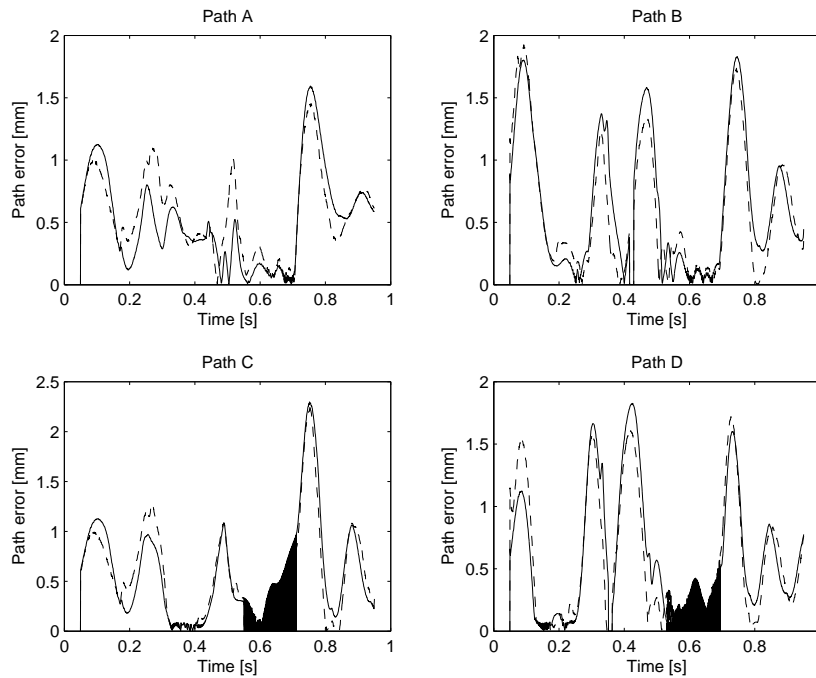


Figure 5.6: Path error for the estimation of Sim2 for AEKF with Cov4 (-) and Cov5 (--).



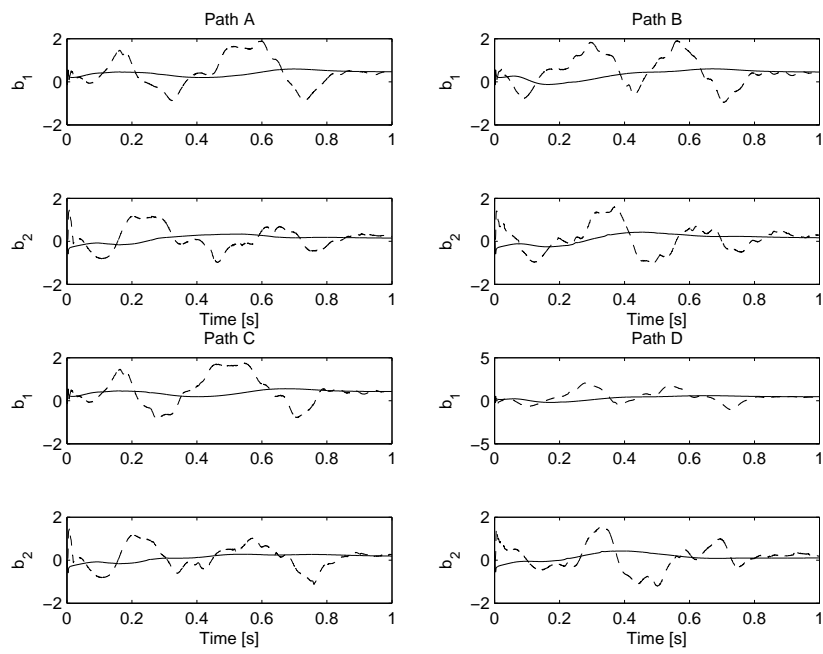


Figure 5.7: Bias parameters for the estimation of Sim3 with Cov4 (-) and Cov5 (--).

Table 5.1: Max and mean error in mm for the EKF on path A.

Path A	EKF					
	Cov1		Cov2		Cov3	
	Max	Mean	Max	Mean	Max	Mean
Sim1	<b>0.078</b>	<i>0.025</i>	0.080	0.025	0.080	0.026
Sim2	1.681	0.550	<b>1.577</b>	<i>0.543</i>	1.910	0.661
Sim3	0.400	0.113	0.903	0.172	<b>0.079</b>	<i>0.027</i>

Table 5.2: Max and mean error in mm for the EKF on path B.

Path B	EKF					
	Cov1		Cov2		Cov3	
	Max	Mean	Max	Mean	Max	Mean
Sim1	0.124	<i>0.035</i>	0.126	0.035	<b>0.112</b>	0.035
Sim2	<b>1.908</b>	<i>0.654</i>	1.966	0.657	2.137	0.687
Sim3	0.419	0.082	0.842	0.120	<b>0.111</b>	<i>0.035</i>

Table 5.3: Max and mean error in mm for the EKF on path C.

Path C	EKF					
	Cov1		Cov2		Cov3	
	Max	Mean	Max	Mean	Max	Mean
Sim1	0.085	<i>0.027</i>	<b>0.085</b>	0.028	0.105	0.030
Sim2	2.340	<i>0.605</i>	<b>2.272</b>	0.607	2.360	0.637
Sim3	0.329	0.081	0.457	0.117	<b>0.103</b>	<i>0.030</i>

Table 5.4: Max and mean error in mm for the EKF on path D.

Path D	EKF					
	Cov1		Cov2		Cov3	
	Max	Mean	Max	Mean	Max	Mean
Sim1	0.152	0.035	0.169	<i>0.034</i>	<b>0.134</b>	0.035
Sim2	1.812	0.611	<b>1.805</b>	0.619	2.219	<i>0.590</i>
Sim3	0.394	0.093	0.907	0.124	<b>0.142</b>	<i>0.035</i>

Table 5.5: Max and mean error in mm for the AEKF on path A.

Path A	AEKF			
	Cov4		Cov5	
	Max	Mean	Max	Mean
Sim2	1.594	<i>0.520</i>	<b>1.453</b>	0.568
Sim3	0.729	0.157	<b>0.076</b>	<i>0.020</i>
Sim4	0.728	0.155	<b>0.076</b>	<i>0.020</i>

Table 5.6: Max and mean error in mm for the AEKF on path B.

Path B	AEKF			
	Cov4		Cov5	
	Max	Mean	Max	Mean
Sim2	<b>1.831</b>	0.643	1.925	<i>0.587</i>
Sim3	0.624	0.112	<b>0.089</b>	<i>0.028</i>
Sim4	0.629	0.111	<b>0.089</b>	<i>0.028</i>

Table 5.7: Max and mean error in mm for the AEKF on path C.

Path C	AEKF			
	Cov4		Cov5	
	Max	Mean	Max	Mean
Sim2	2.297	<i>0.578</i>	<b>2.240</b>	0.582
Sim3	0.412	0.101	<b>0.100</b>	<i>0.024</i>
Sim4	0.419	0.099	<b>0.100</b>	<i>0.024</i>

Table 5.8: Max and mean error in mm for the AEKF on path D.

Path D	AEKF			
	Cov4		Cov5	
	Max	Mean	Max	Mean
Sim2	1.827	0.591	<b>1.720</b>	<i>0.560</i>
Sim3	0.558	0.119	<b>0.094</b>	<i>0.020</i>
Sim4	0.565	0.119	<b>0.094</b>	<i>0.020</i>

## 6 Conclusions

The aim was to improve the estimation in the low frequency range. A first look at the result would say that the aim is fulfilled. This is an incorrect conclusion since the offset in the low frequency range was not present in this simulation study at all. There are anyway a few things to point out. The optimization of the covariance matrices is a hard work. The optimization problem in Section 3.3 works decent but a global solution is not guaranteed. Moreover, the estimation seems robust for changes in the drift, this requires however a more detailed study. The estimation seems also robust for the paths. A more detailed study is once more necessary to conclude this. Another important observation is that model errors affect the estimation negative. This is a big problem which in practice is inevitable.

## 7 Future Work

This work has introduced more questions than answers and most of the questions have been put to future work. It can be summerized as:

- Validate the result on experimental data.
- Choice of covariance matrices. Is there a better way to solve the optimization problem or is it possible to optimize the matrices in a better way than in Section 3.3.
- Are the covariance matrices depending of the states? That is, must the EKF use different matrices depending in which area of the workspace the robot operates. Moreover, the covariance matrices can change when the velocity, corner zones etc. changes for the path.
- Does Euler forward introduce errors during the discretization? Investigate how different methods to discretize the system affect the result.
- Investigate if the estimation changes with higher values on the calibration error, drift, and model errors.


Other types of observers that can be tested are:

- Unscented Kalman Filter (UKF)
- Particle Filter (PF)
- Other type of nonlinear observer

## References

- [1] Fredrik Gustafsson. *Adaptive Filtering and Change Detection*. Wiley, Chichester, England, 1 edition, 2000.
- [2] Fredrik Gustafsson, Lennart Ljung, and Mille Millnert. *Signalbehandling*. Studentlitteratur, Lund, Sweden, 2 edition, 2001.

- [3] Robert Henriksson. Observatör för skattning av verktygspositionen hos en industrirobot. Master's thesis, Linköping University, 2009. ISRN LITH-isy-ex--09/4271--se.
- [4] Robert Henriksson, Mikael Norrlöf, Stig Moberg, Erik Wernholt, and Thomas B. Schön. Experimental comparison of observers for tool position estimation of industrial robots. In *Proceedings of 48th IEEE Conference on Decision and Control*, Shanghai, China, December 2009.
- [5] Stig Moberg, Jonas Öhr, and Svante Gunnarsson. A benchmark problem for robust control of a multivariable nonlinear flexible manipulator. Technical report, Department of Electrical Engineering, Linköping University, 2008. URL: <http://www.robustcontrol.org>.

	<b>Avdelning, Institution</b> Division, Department  Division of Automatic Control Department of Electrical Engineering	<b>Datum</b> Date  2009-12-08
	<b>Språk</b> Language  <input type="checkbox"/> Svenska/Swedish <input checked="" type="checkbox"/> Engelska/English  <input type="checkbox"/> _____	<b>Rapporttyp</b> Report category  <input type="checkbox"/> Licentiatavhandling <input type="checkbox"/> Examensarbete <input type="checkbox"/> C-uppsats <input type="checkbox"/> D-uppsats <input checked="" type="checkbox"/> Övrig rapport <input type="checkbox"/> _____
<b>URL för elektronisk version</b>  <a href="http://www.control.isy.liu.se">http://www.control.isy.liu.se</a>		LiTH-ISY-R-2926
<b>Titel</b> A Simulation Study on the Arm Estimation of a Joint Flexible 2 DOF Robot Arm Title		
<b>Författare</b> Patrik Axelsson Author		
<b>Sammanfattning</b> Abstract  <p>The main task for an industrial robot is to move the tool into specific positions. It is therefore necessary to have an accurate knowledge about the tool position. This report describes a simulation study where an accelerometer attached to the robot tool is used. The acceleration and measured motor angles are used with an Extended Kalman Filter (EKF) to estimate the tool position. The work has been focused on a robot with two degrees of freedom. Simulations have been performed with different kind of errors and on different paths. The EKF uses covariance matrices of the process noise and measurement noise which are unknown. An optimization problem has therefore been proposed and solved to get covariance matrices that give good estimations.</p>		
<b>Nyckelord</b> Keywords            Extended Kalman Filter, Industrial manipulator, Accelerometer		