BACHELOR

Keeping controllers secret

An attackers perspective

Erens, Ruud M.

Award date:
2022

Department of Mechanical Engineering

De Rondom 70, 5612 AP Eindhoven P.O.
Box 513, 5600 MB Eindhoven
The Netherlands
www.tue.nl

Author
R.M. Erens (1440780)

Responsible Lecturer
Prof. Dr. Ir. M.S.T. Chong

Date
June 24, 2022

Bachelor Final Project

Keeping controllers secret - An attackers perspective

R.M. Erens (1440780)
r.m.erens@student.tue.nl

EINDHOVEN UNIVERSITY OF TECHNOLOGY

Table of Contents

Title
Bachelor Final Project

1 Introduction

A cyber-physical system like an automotive transportation system is controlled over communication networks with digital computers. The separate machines communicate with a central hub which in return gives commands to the machines, makes sure the machines can not damage each other and the machines work in the most efficient way. This interconnection via networks makes these system, even though very efficient and with great capabilities, vulnerable to attacks which can cause significant damage to the society and companies. Examples of this are the attacks on the Ukrainian power grid [6] and the attack on the Iranian uranium enrichment facility [4].

The world is highly interconnected today, and many important systems are controlled with interconnected controllers. Attackers aim to afflict damage through this controller systems vulnerability and cause adverse societal disruption. The focus will be on breaching the confidentiality of controller states via sensor data observation and manipulation. This confidentiality break of the controller states is one step in a bigger attack scheme that is often taken for granted or assumed to be done already [9].

For this project, it is assumed that the attacker has full knowledge of the plant and controller and it's noises. The attacker also has access to the sensory data and can manipulate this data. The attacker does not know what the initial state of the system is nor what the signal is that the controller sends to the plant. These signals are often better protected since they are of greater importance and can immediately damage the plant. Further assumptions are properties of the plant like stabilizabilty and detectability and that the closed loop system is stable. Finally, the discrete system should have reached steady-state before the attack begins.

2 Preliminary knowledge

Basic properties of linear time invariant systems in continuous an discrete time are recalled based on the material presented[1]. Within the state space representation of control systems, knowledge of certain properties is of importance in this project. In the state space representation, the plant and controller are represented in matrices and state variables. This can be both in continuous and discrete time with Equation 2.1 and Equation 2.2 as examples for time invariant plants.

$$\dot{x}(t) = Ax(t) + Bu(t), \quad y(t) = Cx(t) + Du(t) \tag{2.1}$$

$$x(k+1) = Ax(k) + Bu(k), \quad y(k) = Cx(k) + Du(k) \tag{2.2}$$

These equations have $x \in \mathbb{R}^n$ as column matrix with the state variable, $u \in \mathbb{R}^m$ as column input matrix and $y \in \mathbb{R}^l$ as column output matrix. The continuous time system is stable if the eigenvalues of A have a negative real part and the discrete time system is stable if the eigenvalues of A all have a magnitude smaller than 1.
The important properties of state space models are controllability, observability, stabilizabilty and detectability. Controllability and observability are dual aspects of the same problem. Controllability means that from every state the initial state can be reached, observability means that from every state with inputs and outputs, the initial state can be determined. Both have a matrix with which the property can be tested, the matrices are given in Equation 2.3.

$$\mathcal{C} = \begin{bmatrix} B & AB & A^2B & \dots & A^{n-1}B \end{bmatrix}, \quad \mathcal{O} = \begin{bmatrix} C \\ CA \\ CA^2 \\ \dots \\ CA^{n-1} \end{bmatrix} \tag{2.3}$$

Here is n the size of the state vecto x. The image of these matrices are the controllable and observable subspaces and if these images have the size of the states, there are only controllable and observable states. Thus if the rank of these matrices are equal to n, the system is controllable (if rank of controllability matrix is n) and observable (if rank of observability matrix is n).

Stabiliazability and detectability are weaker notions of controllability and observability respectively. Every system has an uncontrollable equivalent and an unobservable equivalent. If in these equivalents, the uncontrollable states or unobservable state respectively are stable, the system is stabilizable and detectable respectively. The following tests are used to check whether a system is stabilizable and detactable. The system is stabilizable if and only if every eigenvector of A^T corresponding to an eigenvalue with a positive or zero real part (continuos time) or with a magnitude larger than or equal to one (discrete time) is not in the kernel of B^T. The LTI system is detectable if and only if every eigenvector of A corresponding to an eigenvalue with a positive or zero real part (continuos time) or with a magnitude larger than or equal to one (discrete time) is not in the kernel of C.

3 Problem formulation

The goal is to find a way for an attacker to estimate the controllers internal state as a reaction on input values for a linear time invariant system. This will be done according to [9], where this is already done for discrete time. This discrete time case is used to verify the technique and then a continuous time case will be solved.

3.1 Discrete time

A standard system will be used as format with the plant in Equation 3.1 and the controller in Equation 3.2. The plant state x is in $\mathbb{R}^{n_x}$, the plant input $u \in \mathbb{R}^{n_u}$ and the plant output $y \in \mathbb{R}^{n_y}$, the controller state $x_c \in \mathbb{R}^{n_c}$. The plant further consists of the system matrix $A \in \mathbb{R}^{n_x \times n_x}$, the input matrix $B \in \mathbb{R}^{n_x \times n_u}$ and the output matrix $C \in \mathbb{R}^{n_y \times n_x}$. The controller further consists of the controller system matrix $A_c \in \mathbb{R}^{n_c \times n_c}$, the input matrix of the controller $B_c \in \mathbb{R}^{n_c \times n_y}$, the output matrix of the controller $C_c \in \mathbb{R}^{n_u \times n_c}$ and the feedthrough matrix from the measurements to the actuator signal $D_c \in \mathbb{R}^{n_u \times n_y}$. The closed loop system is visualized in Figure 3.1 where a(k) is the possibility for the attacker to adjust the sensory values and $\omega(k)$ and $\nu(k)$ are noises with mean 0 and their respective covariance matrices Σ_ω and Σ_ν.

$$x(k+1) = Ax(k) + Bu(k) + \omega(k), \quad y(k) = Cx(k) + \nu(k) \tag{3.1}$$

$$x_c(k+1) = A_c x_c(k) + B_c y(k), \quad u(k) = C_c x_c(k) + D_c y(k) \tag{3.2}$$

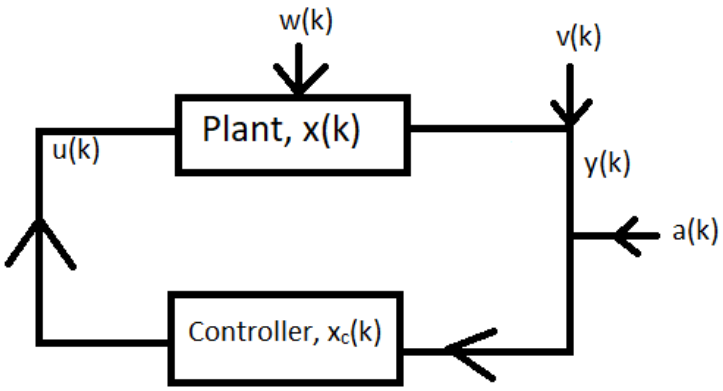

Figure 3.1: Closed loop system visualization

The system can be written down in a compact way by introducing $z(k) = \begin{bmatrix} x(k) \\ x_c(k) \end{bmatrix}$. Combining the plant and controller in one equation would lead to a system described in Equation 3.3 with $A'_z = \begin{bmatrix} A + BD_cC & BC_c \\ B_cC & A_c \end{bmatrix}$, $\eta'(k) = \begin{bmatrix} \omega(k) + BD_c\nu(k) \\ B_c\nu(k) \end{bmatrix}$ and $C_z = \begin{bmatrix} C & 0 \end{bmatrix}$.

$$z(k+1) = A'_z z(k) + \eta'(k), \quad y(k) = C_z z(k) + \nu(k) \tag{3.3}$$

The system with uncorrelated noises would be described by Equation 3.4. In this form, A_z is made up of plant matrices and controller matrices in the following way, $A_z = \begin{bmatrix} A & BC_c \\ 0 & A_c \end{bmatrix}$ and the noises $\eta(k) = \begin{bmatrix} \omega(k) \\ 0 \end{bmatrix}$ and $\nu(k)$ are uncorrelated. $S = \begin{bmatrix} BD_c\Sigma_\nu & B_c\Sigma_\nu \end{bmatrix}$ and $R = \Sigma_\nu$ are noise covariance matrices and $C_z = \begin{bmatrix} C & 0 \end{bmatrix}$ is the input matrix.

$$z(k+1) = A_z z(k) + \eta(k) + SR^{-1}y(k), \quad y(k) = C_z z(k) + \nu(k) \tag{3.4}$$

Making an estimation $\hat{z}(k)$ of z(k) for all k would than suffice in which the only the lower half of the error $e(k) = z(k) - \hat{z}(k)$ has to converge to zero and the covariance matrix of the error would have to converge Σ_∞ of the form $\begin{bmatrix} P & 0 \\ 0 & 0 \end{bmatrix}$ where P is a semi-positive definite matrix [9].
The goal is to show by simulation that this is true for the system used according to the results in [9].

3.2 Continuous time

In continuous time, we will design an observer in a situation with deterministic noise and in a situation with stochastic noise.

$$\dot{x}(t) = A_B x(t) + B_B u(t) + d(t), \quad y(t) = C_B x(t) + n(t) \tag{3.5}$$

$$\dot{x_c}(t) = A_{B,c} x_c(t) + B_{B,c} y(t), \quad u(t) = C_{B,c} x_c(t) + D_{B,c} y(t) \tag{3.6}$$

A standard system will be used as format with the plant in Equation 3.5 and the controller in Equation 3.6. The plant state $x \in \mathbb{R}^{n_x}$, the plant input $u \in \mathbb{R}^{n_u}$ and the plant output $y \in \mathbb{R}^{n_y}$, the controller state $x_c \in \mathbb{R}^{n_c}$. The plant further consists of the system matrix $A_B \in \mathbb{R}^{n_x \times n_x}$, the input matrix $B_B \in \mathbb{R}^{n_x \times n_u}$ and the output matrix $C_B \in \mathbb{R}^{n_y \times n_x}$. The controller further consists of the controller system matrix $A_{B,c} \in \mathbb{R}^{n_c \times n_c}$, the input matrix of the controller $B_{B,c} \in \mathbb{R}^{n_c \times n_y}$, the output matrix of the controller $C_{B,c} \in \mathbb{R}^{n_u \times n_c}$ and the feedthrough matrix from the measurements to the actuator signal $D_{B,c} \in \mathbb{R}^{n_u \times n_y}$. The closed loop system is visualized in Figure 3.1 where a(k) is the possibility for the attacker to adjust the sensory values and $d(t)$ and $n(t)$ are the unknown process noise and sensor noise in the deterministic case and are Gaussian random variables with mean zero and covariance matrices Σ_d and Σ_n in the stochastic case.
The plant matrices in discrete time and continuous time are related to each other and the sample time T_s in the following ways[3]:

- $A_{discrete} = e^{A_{continuous} * T_s}$
- $B_{discrete} = A_{continuous}^{-1} * e^{A_{continuous} * T_s} - I_3 * B_{continuous}$

Equation 3.5 and Equation 3.6 are written in closed-loop form in Equation 3.7 with $z(t) = \begin{bmatrix} x(t) \\ x_c(t) \end{bmatrix}$. In this form, $A_{B,z}$ is made up of plant matrices and controller matrices in the following way, $A_{B,z} = \begin{bmatrix} A_B & B_B C_{B,c} \\ 0 & A_{B,c} \end{bmatrix}$ and the noises $d(t)$ with $\bar{B} = \begin{bmatrix} I_3 \\ 0_3 \end{bmatrix}$ and $n(t)$ are uncorrelated. $B_z = \begin{bmatrix} B_B D_{B,c} \\ B_{B,c} \end{bmatrix}$ is the throughput matrix and $C_{B,z} = \begin{bmatrix} C_B & 0 \end{bmatrix}$ is the input matrix.

$$\dot{z}(t) = A_{B,z} z(t) + \bar{B} d(t) + B_z y(t), \quad y(t) = C_{B,z} z(t) + n(t) \tag{3.7}$$

Deterministic noise

This system can be estimated beginning by taking the noise to be deterministic and small, thus the noise statistics are unknown to the attacker. Using a Luenberger estimator [7], the goal is to find an estimate which will not converge to the real value of the state but also should not diverge from the values of the state. The noise is not taken into account in the estimation which means that the error will not be precisely 0.

Stochastic noise

A better converging way of estimation is done with stochastic noise for which the attacker knows the noise statistics. A Kalman filter[**Class2004MEFilters**] can be used to estimate the state variables of the system. The state estimation error covariance matrix, M, should converge to a stationary solution.

4 Problem setup

4.1 Discrete time

The problem setup in [9] is recalled. Using a Kalman filter, the state can be approximated in which the error for the controller state variables should converge tho zero and the covariance matrix of the error should converge to the form of $\begin{bmatrix} P & 0 \\ 0 & 0 \end{bmatrix}$. Equation 4.1 is used to make an estimation of the state $z(k)$, in this equation

$$\hat{z}(k+1) = A_z\hat{z}(k) + SR^{-1}y(k) + L_z(k)(y - C_z\hat{z}(k)) \tag{4.1}$$

$$e_z(k+1) = z(k) - \hat{z}(k) = (A_z - L_z(k)C_z)e_z(k) + \eta(k) + L_z\nu(k) \tag{4.2}$$

$$L_z(k) = A_z\Sigma_z(k)C_z^T(C_z\Sigma_z(k)C_z^T + R) \tag{4.3}$$

$$\Sigma_z(k+1) = A_z\Sigma_z(k)A_z^T + Q - (A_z\Sigma_z(k)C_z^T)(C_z\Sigma_z(k)C_z^T + R)^{-1}(A_z\Sigma_z(k)C_z^T)^T \tag{4.4}$$

$Q = \begin{bmatrix} \Sigma_w & 0 \\ 0 & 0 \end{bmatrix}$ represents the covariance matrix of $\eta(k)$ in these equations and the initial value of $\Sigma_z(k)$, $\Sigma_z(0) = \Sigma_0$ is the solution to equation Equation 4.5.

$$\Sigma_0 = A_z'\Sigma_0(A_z')^T + Q' \tag{4.5}$$

In Equation 4.5, $A_z' = \begin{bmatrix} A + BD_cC & BC_c \\ B_cC & A_c \end{bmatrix}$ and $Q' = \begin{bmatrix} \Sigma_\omega + BD_c\Sigma_\nu D_c^T B^T & BD_c\Sigma_\nu B_c^T \\ B_c\Sigma_\nu D_c^T B^T & B_c\Sigma_\nu B_c^T \end{bmatrix}$. $\Sigma_z(k)$ should converge to the the form $\begin{bmatrix} P & 0 \\ 0 & 0 \end{bmatrix}$ where P is a positive semi-definite matrix which is the solution to Equation 4.6 with A and C the matrices in Equation 3.1.

$$P = APA^T + \Sigma_\omega - APC^T(CPC^T + \Sigma_\nu)^{-1}CPA^T \tag{4.6}$$

The paper makes the following assumption about the standard system in Equation 3.1 and Equation 3.2 to ensure that the estimation will be correct:

Assumption 1 *(A,B) is stabilizable and (C,A) is detectable.*

Assumption 2 *(A,$\Sigma_\omega^{\frac{1}{2}}$) has no uncontrollable modes on the unit circle.*

Assumption 3 *The controller (A_c,B_c,C_c,D_c) is minimal.*

Assumption 4 *The closed loop system (plant and controller combined) is stable, thus the spectral radius of A_z' in Equation 3.3 is smaller than 1.*

Assumption 5 *The closed loop system has reached steady state before the estimation starts.*

Assumption 6 *The attacker has access to the matrices of the plant and controller, knows the noise statistics and has access to the sensor measurements up to current time, thus the attacker has knowledge of (A,B,C,A_c,B_c,C_c,D_c) and (Σ_ω,Σ_ν) and the measurements y(k) for $k \geq 0$. The attacker does not know the control signal u(k) or the initial state of the system z(0).*

Assumption 7 *The attacker uses measurements$y(k)$ up to current time to estimate the controllers state in the next time step, $z(k+1)$.*

4.2 Continuous time

4.2.1 Deterministic noise

Using a Luenberger estimator, a state estimation can be found as presented in Equation 4.7.

$$\dot{\hat{z}} = A_z\hat{z} + By + L(y - C\hat{z}), \quad y = Cz + n(t) \tag{4.7}$$

Here $A_z\hat{z} + By$ makes a prediction based on a copy of the dynamics of the system and $L(y - C\hat{z})$ does a correction based on the output error. In this method, L is the observer gain matrix determining the correction based on the error.

L can be designed accordingly using the error system in Equation 4.8. As stated in [7] L can be designed by placing the poles of $(A_z - LC_z)$ arbitrarily if (A_z, C_z) is observable, however the error system has to be asymptotically stable thus the poles are placed with strictly negative real parts.

$$\dot{e} = \dot{z} - \dot{\hat{z}} = (A_z - LC_z)e - Ln(t) + B_z d(t) \tag{4.8}$$

The following assumptions are made of the system which should hold to be able to use [7, Page 7]:

Assumption 8 *(A_z,C_z) is observable.*

Assumption 9 *The attacker has access to the matrices of the plant and controller and has access to the sensor measurements up to current time, thus the attacker has knowledge of (A,B,C,A_c,B_c,C_c,D_c) and the measurements y(k) for $t \geq 0$. The attacker does not know the control signal u(t) or the initial state of the system z(0).*

Assumption 10 *The noise terms $d(t)$ and $n(t)$ in Equation 3.7 are small not influencing the states too greatly.*

Assumption 11 *The attacker uses measurements $y(t)$ up to current time to estimate the controllers state in Equation 3.6 in the future with $\dot{z}(t)$.*

4.2.2 Stochastic noise

Using a Kalman filter, the estimation $\hat{z}(t)$ of the state $z(t)$ can be made as in Equation 4.9 [2]. The noise terms $d(t)$ and $n(t)$ are normal random variables with mean 0 and covariance matrices Σ_d and Σ_n respectively.

$$\dot{\hat{z}}(t) = A_z\hat{z}(t) + K(t)(y(t) - C_z\hat{z}(t) + SR^{-1}y(t) \tag{4.9}$$

$$K(t) = MC_z^T\Sigma_n^{-1} \tag{4.10}$$

$$\dot{M} = A_zM + MA_z^T + \bar{B}\Sigma_n\bar{B}^T - MC_z^T\Sigma_n^{-1}C_zM \tag{4.11}$$

The state estimation error covariance M should converge to a stationary solution which satisfies the Algebraic Riccati equation in Equation 4.12 [2, Theorem 1]. To get a perfect estimation of the controller state, M converge to a form of $\begin{bmatrix} P & 0 \\ 0 & 0 \end{bmatrix}$ with P a positive definite matrix. M should have a begin as descirbed by the initial errors in [2].

$$A_zM + MA_z^T = -\bar{B}\Sigma_n\bar{B}^T + MC_z^T\Sigma_n^{-1}C_zM \tag{4.12}$$

This can be proven in simulation when the following assumptions hold:

Assumption 12 *(A_z,C_z) is observable.*

Assumption 13 *The noises $d(t)$ and $n(t)$ are uncorrelated.*

Assumption 14 *The attacker has access to the matrices of the plant and controller, knows the noise statistics and has access to the sensor measurements up to current time, thus the attacker has knowledge of (A,B,C,A_c,B_c,C_c,D_c) and (Σ_ω,Σ_ν) and the measurements y(k) for $k \geq 0$. The attacker does not know the control signal u(t) or the initial state of the system z(0), which is a Gaussian variable.*

Assumption 15 *The attacker uses measurements $y(t)$ up to current time to estimate the controllers state in Equation 3.6 in the future with $\dot{z}(t)$.*

5 Case study

5.1 Modelling

To simulate attacks, a three tank system is used which is introduced in [9]. This system is used to validate the attack scheme which is also based on [9]. The plant is given in Equation 5.1 with $n_x = 3$, $n_y = 2$, $n_u = 2$. In this plant, $\omega(k)$ is a Gaussian random variable with mean 0 and covariance matrix a 3x3 identity matrix and $\nu(k)$ a Gaussian random variable with mean 0 and covariance matrix a 2x2 identity matrix divided by 10.

$$\dot{x}(t) = \begin{bmatrix} -2 & 2 & 0 \\ 2 & -4 & 2 \\ 0 & 2 & -3 \end{bmatrix} x(t) + \begin{bmatrix} 0.5 & 0 \\ 0 & 0 \\ 0 & 0.5 \end{bmatrix} u(t) + \omega(t), \quad y(t) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} x(t) + \nu(t) \tag{5.1}$$

5.1.1 discrete time

This system can be discretized with a sample time of 0.5 seconds resulting in the plant in Equation 5.2, this is done according to [3].

$$x(k+1) = \begin{bmatrix} 0.5244 & 0.3114 & 0.1331 \\ 0.3114 & 0.3462 & 0.2448 \\ 0.1331 & 0.2448 & 0.3355 \end{bmatrix} x(k) + \begin{bmatrix} 0.1755 & 0.0156 \\ 0.0566 & 0.0488 \\ 0.0156 & 0.1433 \end{bmatrix} u(k) + \omega(k), \quad y(k) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} x(k) + \nu(k) \tag{5.2}$$

The controller used in this system is given by Equation 3.2 with $n_c = 3$, where $A_c = A - BK_i - LC$, $B_c = L$, $C_c = -K_i$ and $D_c = 0$. Pole placement is used to find controller and observer gains, K_i (which can differ per situation, K_s for a stable controller and K_u for an unstable controller) and L respectively. The poles of $A - LC$ are placed at 0.1, 0.2 and 0.3 for the discrete time case and for a stable controller the poles of $A - BK$ are placed at 0.4, 0.5 and 0.6, this leads to observer gain matrix $L = \begin{bmatrix} 0.5629 & 0.2603 \\ 0.4117 & 0.3810 \\ 0.3595 & 0.1944 \end{bmatrix}$ and controller gain matrix $K_S = \begin{bmatrix} -1.2433 & 3.1261 & -0.4009 \\ -0.6624 & 3.3487 & -2.7885 \end{bmatrix}$. An unstable controller is designed in such a way that $\rho(A - BK_U) < 1$ and $A - BK_U - LC$ has an eigenvalue at 1.5. The resulting controller gain is $K_U = \begin{bmatrix} 0.5530 & 1.9589 & 1.2225 \\ 1.8414 & 27.0785 & -12.9349 \end{bmatrix}$. These matrices combine into the stable controller in Equation 5.3 with $A_c = \begin{bmatrix} 0.7529 & -0.8522 & -0.0135 \\ 0.4141 & -0.4060 & 0.0227 \\ 0.2473 & -0.6432 & 0.5469 \end{bmatrix}$, $B_c = \begin{bmatrix} 0.5629 & 0.2603 \\ 0.4117 & 0.3810 \\ 0.3595 & 0.1944 \end{bmatrix}$ and $C_c = -\begin{bmatrix} -1.2433 & 3.1261 & -0.4009 \\ -0.6624 & 3.3487 & -2.7885 \end{bmatrix}$

$$x_c(k+1) = A_c x_c(k) + B_c y(t), u(k) = C_c x_c(k) \tag{5.3}$$

5.1.2 Continuous time

The plant as in Equation 5.1 is considered in continuous time with a controller made for the continuous time system. In continuous time the poles are placed at the continuous time equivalent of the discrete time poles calculated with Equation 5.4 [10]. In this equation, z represents the pole in discrete time and s represents the pole in continuous time. T is the sample time of the discrete time system.

$$z = e^{sT} \tag{5.4}$$

This leads to a stable controller gain matrix $K_S = \begin{bmatrix} -3.6416 & 7.0743 & -2.4180 \\ -2.6463 & 7.0743 & -5.8773 \end{bmatrix}$ and an observer gain matrix $L = \begin{bmatrix} 2.2486 & 0 \\ -0.3732 & 2 \\ 2 & 1.6052 \end{bmatrix}$. Combining the matrices a stable controller is formed as in Equation 5.5 with $A_c = \begin{bmatrix} -0.1792 & -3.7857 & 1.2090 \\ 2.0000 & -3.6269 & 0 \\ 1.3231 & -3.6907 & -1.6665 \end{bmatrix}$, $B_c = \begin{bmatrix} 2.2486 & 0 \\ -0.3732 & 2 \\ 2 & 1.6052 \end{bmatrix}$ and $C_c = -\begin{bmatrix} -3.6416 & 7.0743 & -2.4180 \\ -2.6463 & 7.0743 & -5.8773 \end{bmatrix}$.

$$\dot{x_c}(t) = A_c x_c(t) + B_c y(t), u(t) = C_c x_c(t) \tag{5.5}$$

5.2 simulations

5.2.1 Discrete time

The three tank system can be described by Equation 3.4 when combining the plant and controller state variables, which has uncorrelated noises [9]. In these equation are both the plant states and the controller states represented by $z(k) = \begin{bmatrix} x(k) \\ x_c(k) \end{bmatrix}$. A_z is made up of plant matrices and controller matrices in the following way, $A_z = \begin{bmatrix} A & BK \\ 0 & A - BK - LC \end{bmatrix} = \begin{bmatrix} 0.5244 & 0.3114 & 0.1331 & 0.2285 & -0.6007 & 0.1137 \\ 0.3114 & 0.3462 & 0.2448 & 0.1027 & -0.3404 & 0.1588 \\ 0.1331 & 0.2448 & 0.3355 & 0.1143 & -0.5285 & 0.4058 \\ 0 & 0 & 0 & 0.7529 & -0.8522 & -0.0135 \\ 0 & 0 & 0 & 0.4141 & -0.4060 & 0.0227 \\ 0 & 0 & 0 & 0.2473 & -0.6432 & 0.5469 \end{bmatrix}$ and the noises $\eta(k) = \begin{bmatrix} \omega(k) \\ 0 \end{bmatrix}$ and $\nu(k)$ are uncorrelated. $S = \begin{bmatrix} 0 \\ L\Sigma_\nu \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0.0563 & 0.0260 \\ 0.0412 & 0.0381 \\ 0.0360 & 0.0194 \end{bmatrix}$ and $R = \Sigma_\nu = \begin{bmatrix} 0.1 & 0 \\ 0 & 0.1 \end{bmatrix}$ are noise covariance matrices and $C_z = \begin{bmatrix} C & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$ is the input matrix. This system can be simulated leading to the state variables plotted in Figure 5.1 in which z_1, z_2 and z_3 represent the state variables for the plant and z_4, z_5 and z_6 represent the state variables for the controller. The initial state is a Gaussian random variable with mean 0 and covariance matrix Σ_0 which is the solution to Equation 4.5 as in [9].

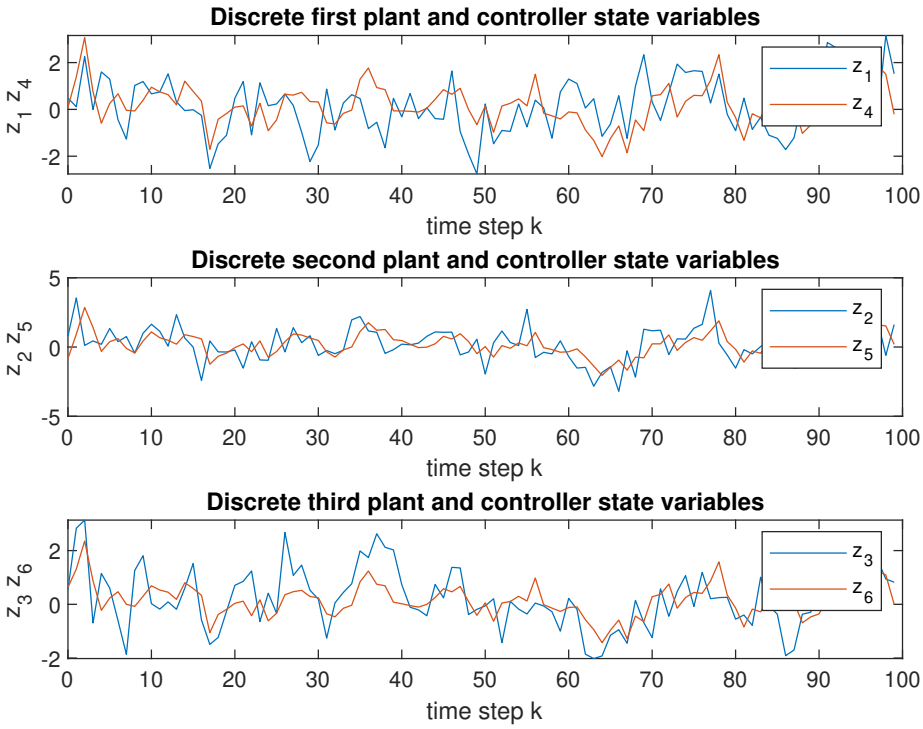

Figure 5.1: State variables discrete time simulation three tank system

Firstly, all assumptions which the system should comply to are met. These assumptions are stated in chapter 3 and are all true for this system, which means that the attack strategy applied in [9] should be applicable.
assumption one is met since the rank of both the controllability matrix and observability matrix are both equal to three, which is also the size of the plant state column matrix. There are no uncontrollable eigenmodes of (A,$\Sigma_\omega^{\frac{1}{2}}$

thus assumption 2 is automatically met. The controller is minimal since the number of controller states necessary to control a plant are minimally the same as those of the plant, which are both three in this case. The spectral radius of A_z' is equal to 0.5388 thus assumption 4 is met as well and the last three assumptions are met in way of modelling. The simulations of both the actual z values and estimation values have been completed for both a stable and unstable controller and the error of the first controller state variable is plotted in Figure 5.5. The initial estimation state is taken to be zero, as in [9] to get resembling confirmation of the right technique of estimation. To show that the covariance matrix Σ_z converges to the form of $\begin{bmatrix} P & 0 \\ 0 & 0 \end{bmatrix}$ as the theorem and its proof states, the norm of the matrix is plotted over time in Figure 5.6 and Figure 5.7 for both the stable and unstable controller respectively. P is the unique solution of the algebraic Riccati equation in Equation 4.6 and the norm of $\begin{bmatrix} P & 0 \\ 0 & 0 \end{bmatrix}$ is 1.5644 which the plot of the covariance norm of z of the stable controller converges to. In these figures it can be observed that the covariance matrix with a stable controller actually converges to the correct value and for the unstable controller this does not happen, which means that the estimation only works for the stable controller.

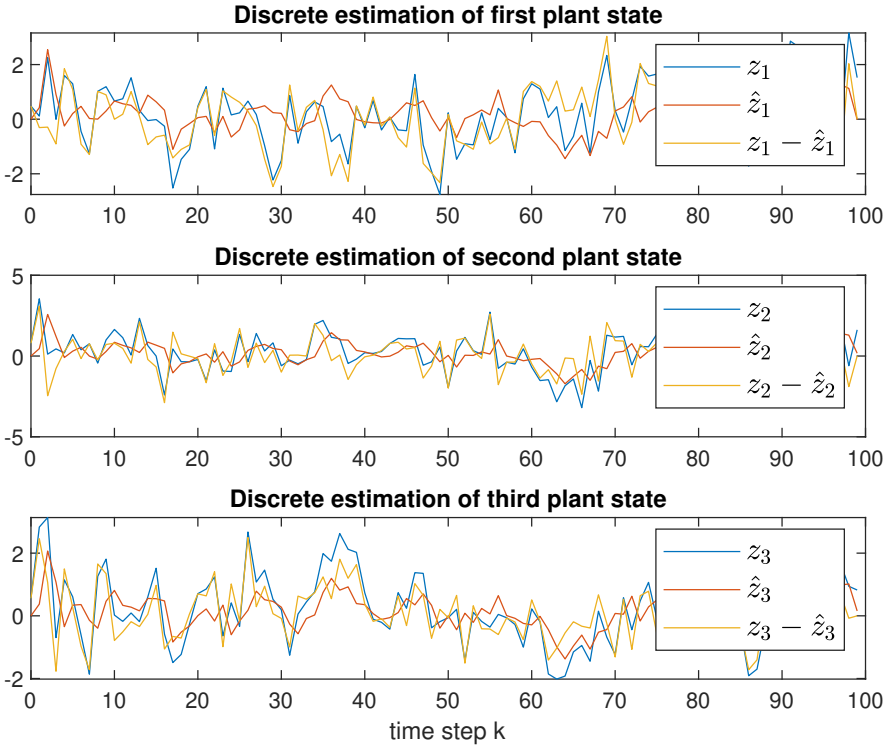

Figure 5.2: Plant states, estimations and errors for a stable controller in discrete time

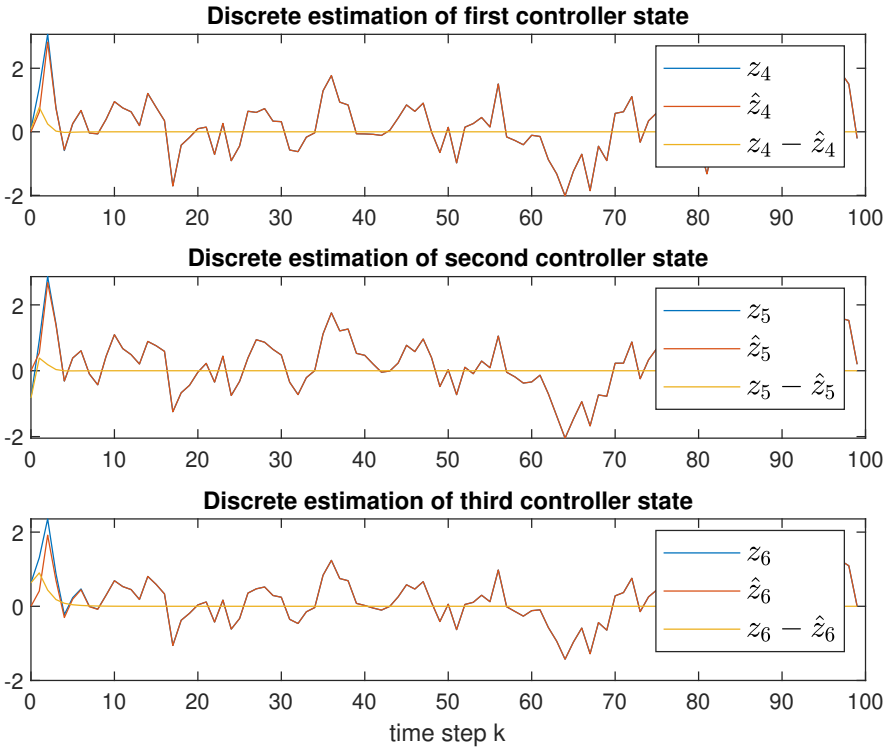

Figure 5.3: Controller states, estimations and errors for a stable controller in discrete time

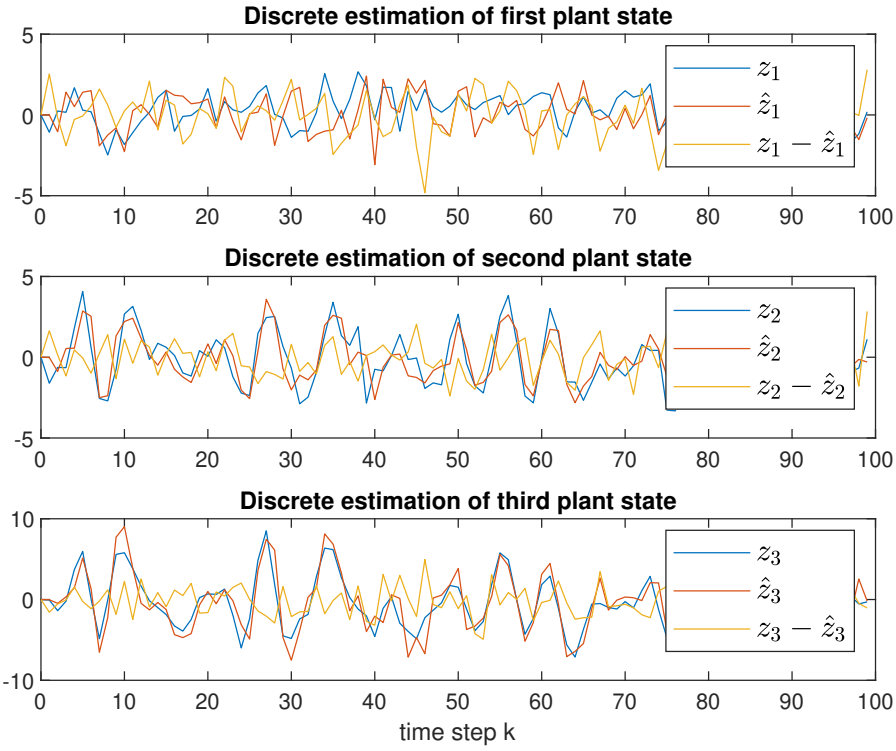

Figure 5.4: Plant states, estimations and errors for an unstable controller in discrete time

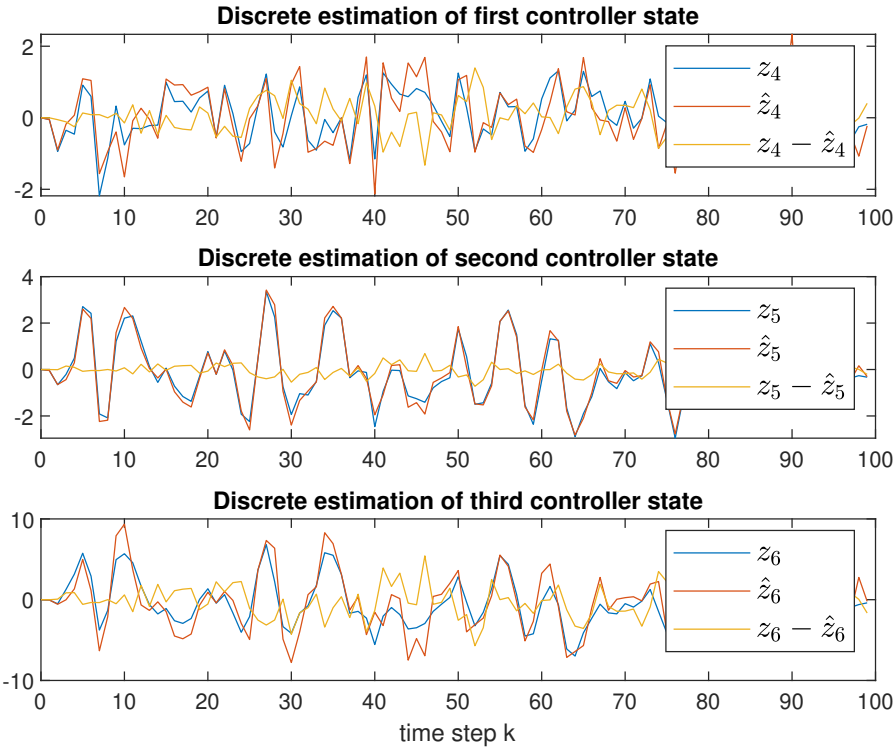

Figure 5.5: Controller states, estimations and errors for an unstable controller in discrete time

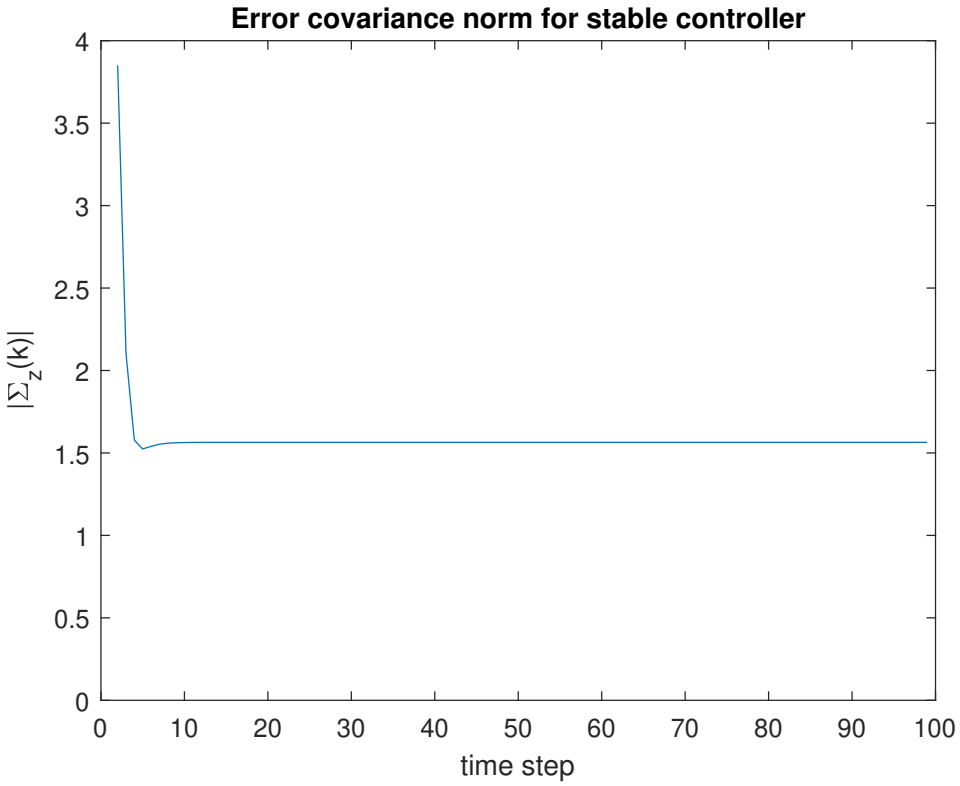

Figure 5.6: Norm of the covariance matrix for a stable controller

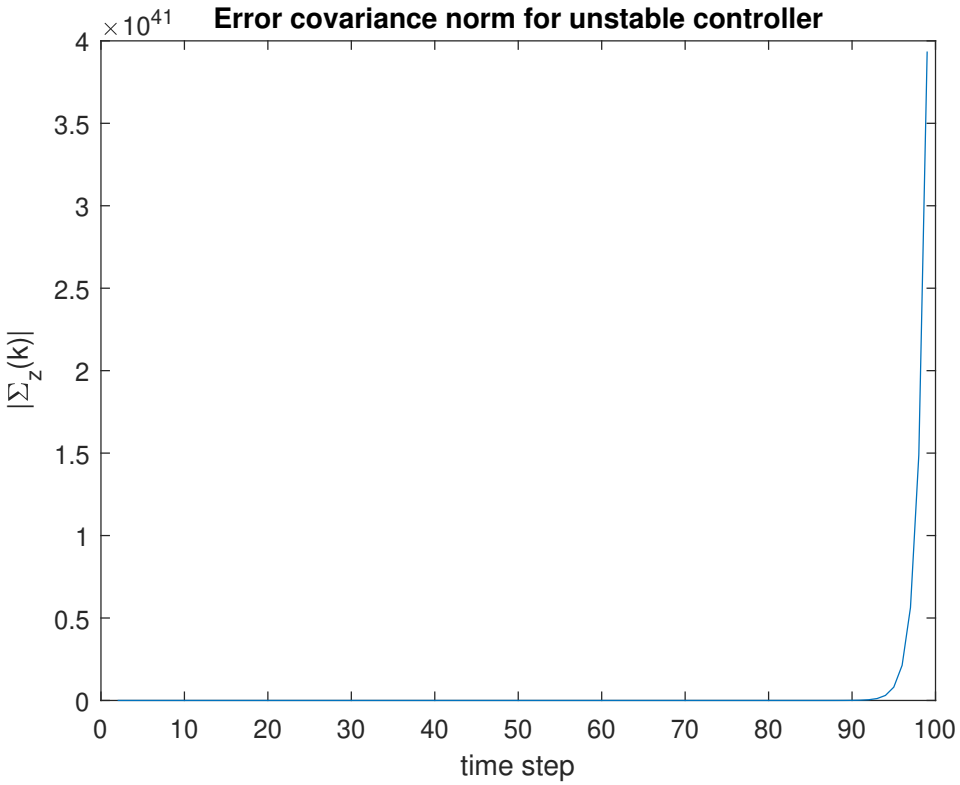

Figure 5.7: Norm of the covariance matrix for an unstable controller

5.2.2 Continuous time

Deterministic case

Using a Luenberger estimator, a state estimator can be found as presented in Equation 4.7. The assumptions for this case (assampion 8-11) are all true for the system, with the rank of the observability matrix equalling 6, the amount of states. The attack only uses known matrices to the attacker and measurement up to current time. Also the plots show limited influence of noise on the states. The noises are now taken to be Gaussian random variables with $\Sigma_d = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ and $\Sigma_n = \begin{bmatrix} 0.1 & 0 \\ 0 & 0.1 \end{bmatrix}$ which is a small noise in continuous time. The poles of $(A_z - LC_z)$ are now placed on -2, -4, -6, -8, -10 and -12, they were first chosen arbitrarily and than optimized a small bit based on how fast the estimation would move to be similar to the real states of the controller. The results of both the plant state estimation and controller state estimation are in Figure 5.8 and Figure 5.9. The estimation clearly approaches the real values of the states, which shows in simulation that it works for this system.

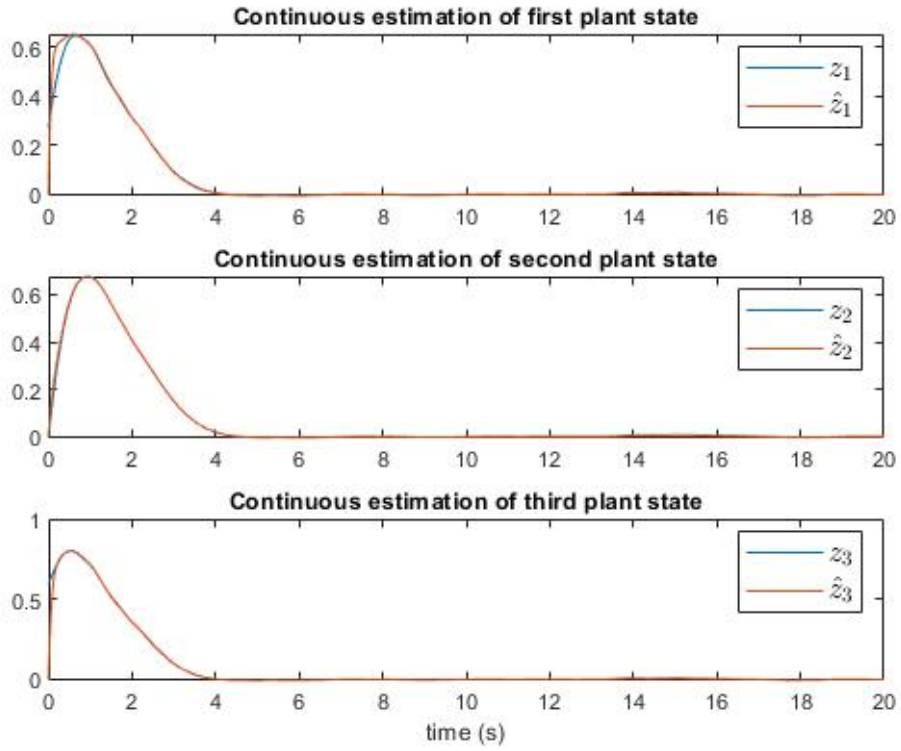

Figure 5.8: Plant states, estimations and errors for a stable controller with Luenberger observer

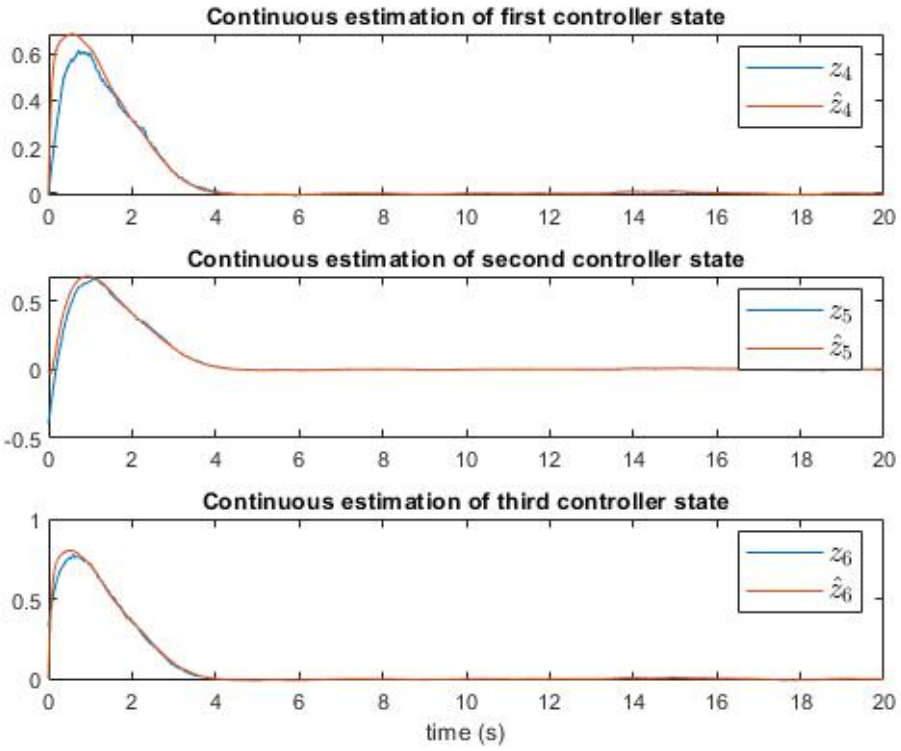

Figure 5.9: Controller states, estimations and errors for a stable controller with Luenberger observer

Stochastic noise

For the stochastic noise in continuous time, a maximum error of the sensors of 1% is used [5]. This would mean that the standard deviation is 0.01 and thus n would be a Gaussian random variable with covariance matrix $\Sigma_n = 0.01^2 I_2$ and as in [9] d is taken to have a variance ten times bigger, thus d is a Gaussian random variable with mean zero and covariance matrix $\Sigma_d = 10 * 0.01^2 I_3$. All the assumptions to use the Kalman filter are met with the system, the observability is already shown in the deterministc noise case and the noises are uncorrelated. Equation 4.9 - Equation 4.11 are used to find a estimation which converges to the real state values, but it does it slowly. This is shown in Figure 5.10 and Figure 5.11. The error covariance matrix M does indeed converge to the stationary solution to Equation 4.12. This can be tested by plotting the norm of the covariance matrix against time and see whether it converges to the norm of the unique solution of Equation 4.12, which is 0.4425 and the plot shows that the norm of M converges to that value. However, the begin values of M should be taken a look at since it should not be a zero matrix but have a begin situation with which the estimation is stable as described in [2], but this did not work out yet and should be worked on in the future.

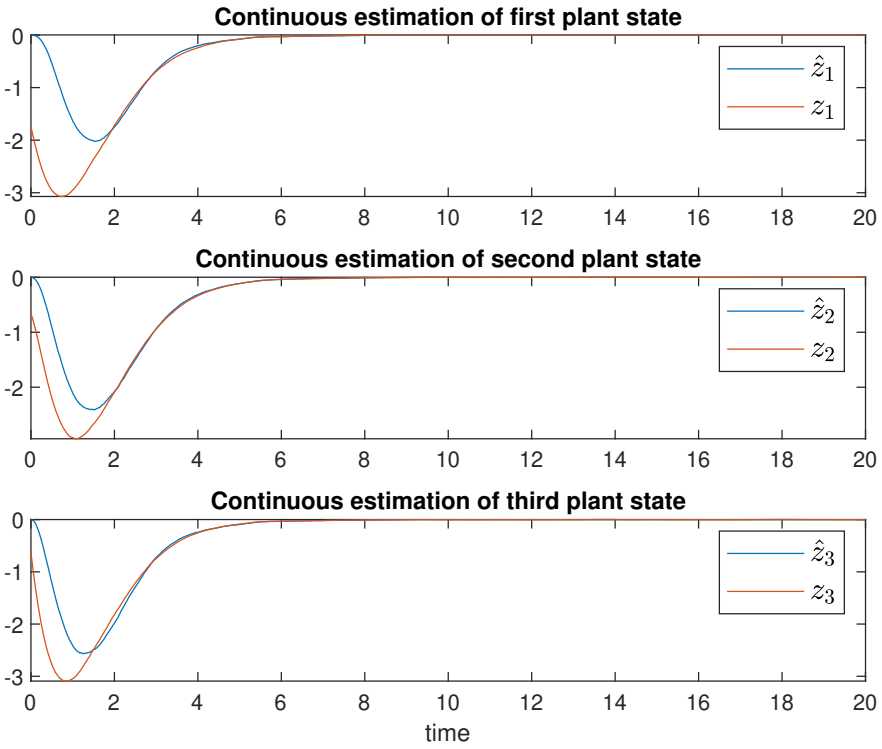

Figure 5.10: Plant states and estimations for a stable controller with a Kalman filter in continuous time

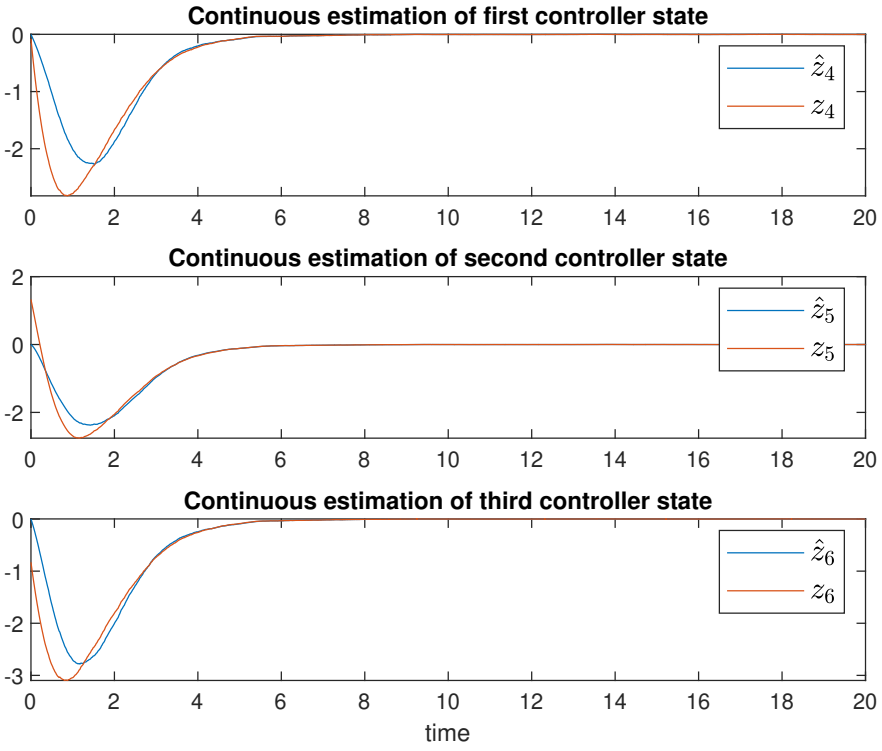

Figure 5.11: Controller states and estimations for a stable controller with a Kalman filter in continuous time

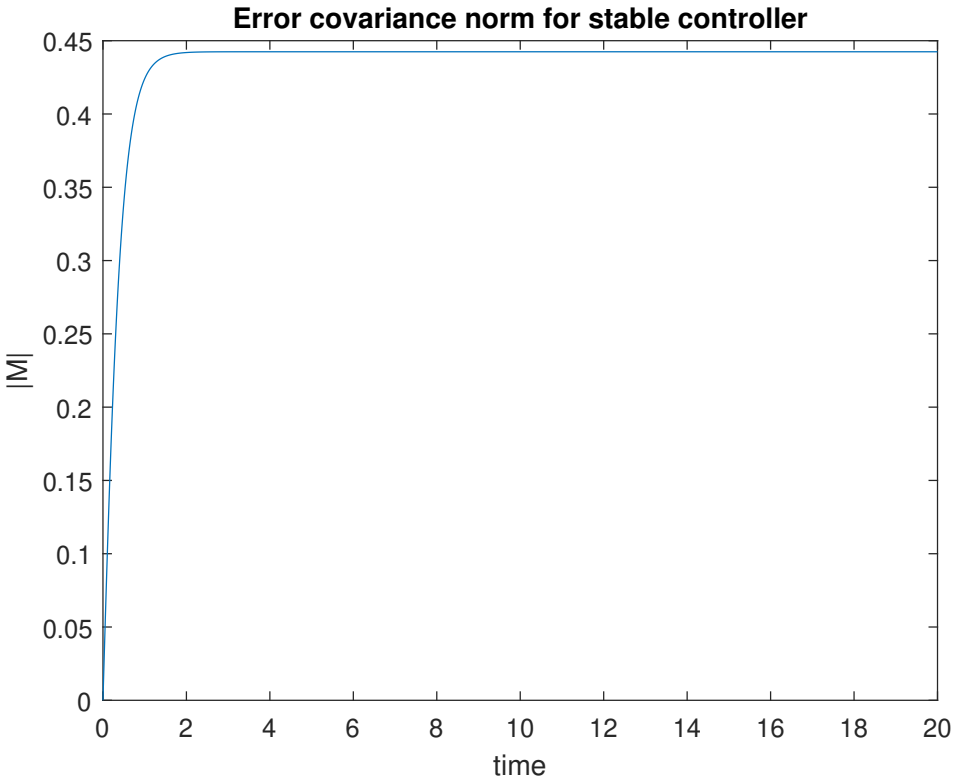

Figure 5.12: Norm of the covariance matrix M

6 Defence Mechanisms

The goal of the defence mechanisms is to make it impossible for the attacker to estimate the controller state without taking away much of the functionality of the controller. The simple way of doing this is by looking at the assumptions which have to hold for the estimation to work and make sure they do not hold for the case. [9] already mentions the techniques of using an unstable controller and injecting noise into the controller side which would basically mean that the measurement values of the attacker would not be usable. Other ways to defend the system would be to defend the sensor measurement signal $y(k)$ or $y(t)$ better against attacks in the same way the attacker can not use the control signal $u(k)$ or $u(t)$ or make the controller unobservable in the discrete time case or the closed loop system unobservable in the continuous time case by using non-minimal control.

6.1 Signal defence

The control signal is stated to be better protected in [9] since manipulation of this signal could lead to immediate physical problems. Since in this case the goal is to prevent attackers to find an estimation of the controller state, a similar protection on the measurement data could do the job. The estimation could lead to knowledge about the control signal when all other system parameters in the form of system and controller matrices are known, which could lead to similar physical problems when the measurement data is manipulated. A possible way of protection is by using subchannels with more sets of data, so producing both valuable data and invaluable data and splitting them into separate subchannels and using multiresolution coding to protect the most valuable data [8].

6.2 Unobservability by non minimal control

Non minimal control could make the system unobservable, a minimal realization of a system is always controllable and observable [1, Theorem 17.1], so using non minimal realizations could have the benefit of being unobservable and thus making it impossible to estimate the state of the controller.

7 Conclusion and future work

Concluding, the estimator for the discrete time systems work very well in simulation. The error covariance matrix converges to the form it should converge to and the error thus converges to zero for the controller states. In continuous time the case with deterministic noise behaves as expected in simulation, where the error approaches zero. The noise factors are not taken into account in the estimation which means that the estimation error does not actually converge to 0 but only approaches 0. In the stochastic noise case the error does actually converge to 0 in simulation, but the only problem is that the initial condition of the error covariance matrix is not yet implemented correctly. This means that the estimation works over time but could possibly work much quicker if the initial condition of the error covariance matrix is implemented correctly.

For future work the completion of the continuous time case is the starting point, making sure that the stochastic noise case in continuous time works completely correctly and than start looking at the newly imposed defence mechanisms. This would mean looking at how to implement the defence of measurement signals and whether it is possible to design a non-minimal controller which ensures the confidentiallity of the controller states. The system has to become unobservable and also the observable decomposition should not give value to the attacker.

Bibliography

[1] Joao P. Hespanha. *Linear systems theory*. 1st ed. Vol. 1. Princeton: Princeton University Press, 2009. ISBN: 9780691140216.

[2] Tony Kelman. *ME 233 Advance Control II Lecture 13 Continuous Time Kalman Filters*. University of California at Berkely, 2004.

[3] Sarah Koskie. *Discretization of Continuous-Time Systems*. 2005.

[4] David Kushner. "The real story of stuxnet". In: *IEEE Spectrum* 50.3 (2013), pp. 48–53. ISSN: 00189235. DOI: 10.1109/MSPEC.2013.6471059.

[5] Konstantinos Loizou and Eftichios Koutroulis. "Water level sensing : State of the art review and performance evaluation of a low-cost measurement system". In: *Measurement* 89 (2016), pp. 204–214. ISSN: 0263-2241. DOI: 10.1016/j.measurement.2016.04.019. URL: http://dx.doi.org/10.1016/j.measurement.2016.04.019.

[6] Michael Assante; Tim Conway; Robert Lee. "Analysis of the Cyber Attack on the Ukrainian Power Grid". In: *SANS Industrial Control Systems Security Blog* (2016), pp. 1–26. URL: https://ics.sans.org/media/E-ISAC_SANS_Ukraine_DUC_5.pdf.

[7] Richard M Murray. *Introductory Control Theory: Lecture 1-2 Observability and State Estimation*. California Institue of Technology, 2007.

[8] Ganesh R. Naik. *Signal Processing: New Research*. 1st ed. New York: New York : Nova Science Publishers, Inc. 2013.

[9] David Umsonst and Henrik Sandberg. "On the confidentiality of controller states under sensor attacks". In: *Automatica* 123 (Jan. 2021). ISSN: 00051098. DOI: 10.1016/j.automatica.2020.109329.

[10] Won Young Yang. *Signals and Systems with MATLAB*. Berlin: Springer-Verlag, 2009. DOI: 10.1007/978-3-540-92954-3.