Two-loop static state feedback for the MIMO gantry crane

Simons, W.F.J.

Award date:
1993

7085

EINDHOVEN UNIVERSITY OF TECHNOLOGY
DEPARTMENT OF ELECTRICAL ENGINEERING
Measurement and Control section

TWO-LOOP STATIC STATE FEEDBACK

FOR THE MIMO GANTRY CRANE.

by: W.F.J. Simons

Master of Science Thesis
carried out from March 1993 to December 1993
commissioned by prof. dr. ir. A.C.P.M. Backx
under supervision of dr. ir. A.A.H. Damen and dr. S. Weiland
date: december 1993

Summary.

Simons, W.F.J.; Two-loop state control of the MIMO gantry crane.

M.Sc. Thesis, Measurement and Control Section ER, Electrical Engineering, Eindhoven University of Technology, The Netherlands, December, 1993.

Using a nonlinear simulation model of the gantry crane, the two-loop scheme proposed by Zhu [Zhu88] has been tested. An initial static state controller, has been applied to make the system stable and to reduce its nonlinear behaviour. Then a Minimum Polynomial (MP) model [Fal93] of the system (process in closed loop) has been estimated. Based on this model of the system, a new second loop static state controller operating parallel to the first state controller has been designed using a simulation method proposed by Boyd [Boy91]. The new first loop controller is then formed by addition of the initial and second loop controller. In the next iteration the procedure described has been repeated. The performance is measured with the tracking error, which is the difference between the reference track and the actual position of the load. The robust performance is optimized and it is possible to design a controller that satisfies constraints in time domain. Vidyasagar [Vid85] showed that if the system is strongly stabilizable the first loop controller puts no restrictions on the second loop controller. Because the gantry crane has only stable zero dynamics (minimum phase) it is allowed to use the two-loop structure.
The procedure of system identification and controller design can be applied recursively which makes the controller adaptive to changing dynamics. If the estimated linear time invariant model of the system in each iteration is accurate enough, the resulting controller based on the model of the system can probably perform better and tend to the best controller possible under the given constraints.
Beside promising simulation results of the gantry crane obtained by this method, a comprehensive derivation of the nonlinear model, zero dynamics, linearized model and a method for measuring the nonlinearity of a system are included.

Contents.

1. Introduction.

Most modern techniques for the design of Linear Time Invariant (LTI) controllers need a model of the process to be controlled. But sometimes this model might be very hard to get, e.g., when the process is unstable, or when it has some nonlinearity and/or time variations. In these cases open loop identification with a LTI model set may be very difficult. Another example is an industrial process. Most of the times it is too expensive to run such a process in open loop for identification.

Zhu [Zhu88] proposed to do the identification in closed loop. Now the system (process in closed loop) is stabilized by the first loop controller which can also reduce the nonlinear behaviour and or time variance. With the model of the system a new controller can be designed using a simulation method proposed by Boyd [Boy91], optimizing robust performance, and satisfying constraints in time domain. The complete scheme can be applied recursively which makes it adaptive to changing dynamics.

The method is applied to a nonlinear simulation model of a gantry crane. This is a Multiple Input Multiple Output (MIMO) system, consisting of a trolley moving along a rail, and a cable of varying length attached to the trolley and the load. It is often used in harbour areas to move the cargo from e.g. freighter to truck. The performance is measured with the tracking error, the difference between the reference signal and the actual position of the load. Some properties of the nonlinear simulation model are derived and used as a-priori information during identification and controller design. If the system with the new controller is more linear (we formalize this concept later), we can estimate a more accurate model of the system. This enables us to design a controller with less robustness giving a higher performance e.g. a smaller tracking error. If the estimated model is more accurate each iteration, the resulting controller will tend to the controller providing the smallest tracking error possible under the given constraints. The simulations show promising results of this technique.

2. Two-loop state control.

The Two-Loop Control (TLC) scheme proposed by Zhu [Zhu87] is briefly discussed. We show how it works, when it can be used and what the advantages are. Some results from Vos [Vos92] and Belt [Bel93] for the SISO case are explained. Finally, the two-loop structure is applied to nonlinear MIMO systems with static state feedback and a design rule is derived.

2.1. Basic ideas of two-loop control.

For optimal controller design, a model of the process under consideration needs to be available. We consider here only Linear Time Invariant (LTI) models for the design of LTI controllers. But, obtaining such models can be very difficult when the process is e.g. unstable, has some nonlinearities and/or time-variance or when open loop identification experiments are not allowed or too expensive.

In [Zhu87] is proposed to do the identification on the closed loop system, see figure 2.1. Now the process input u (which has to be sufficiently rich for identification) is correlated with the output disturbance d through the feedback loop which makes identification of the process *P* very difficult. But for the system $\hat{S}$ (the process in closed loop) the input v is independent of the disturbance d, so identification is much easier. A properly chosen first loop controller K stabilizes the unstable process and hopefully reduces the nonlinearities and/or time variances. It can also reduce the influence of disturbance at the output and the effective order of the system. With the model of the system we can design a second loop controller ΔK which operates parallel to the first loop controller and has the same structure, see again figure 2.1 which also explains the name two-loop control.

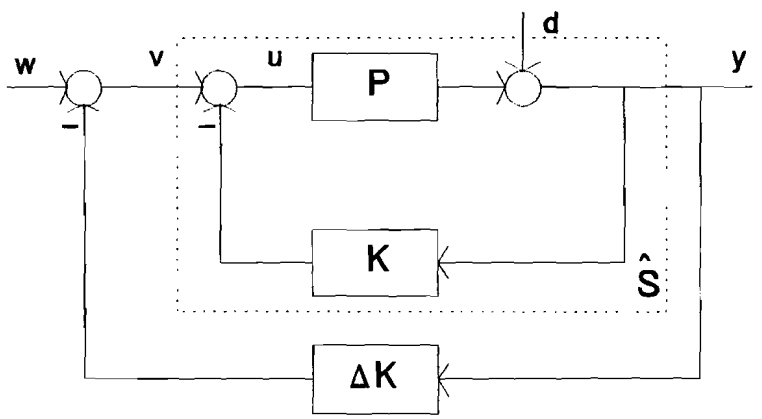


figure 2.1 Principle of two-loop control.

The new first loop controller K_n can directly be implemented in the system as $K_n = K + \Delta K$ possibly resulting in a more linear system with less time variance. After implementation of K_n we can start all over again and again, each time designing a controller with better performance e.g. a smaller tracking error. The scheme is also adaptive to changing dynamics. Of course the dynamics may not change faster than the update of the controller, which depends on the length of the impulse response of the system, and the calculation time necessary for identification and controller design. For identification we need a dataset which roughly has to be 5 to 10 times the length of the impulse response of the system. This means that we can only adapt to changing dynamics after 5 to 10 times its time constant.

By improving the static state feedback controller K, the estimated model of the system can be more accurate. Then we can design a controller with less robustness, giving a higher performance.

Under certain conditions the variance of the estimated model of the system is proportional to the square of the sensitivity S. This relationship is given by Zhu [Zhu88], see formula (2.1).

$$var[\hat{S}(e^{i\omega})] \approx \frac{n}{N}\frac{\Phi_{d^*d^*}}{\Phi_v} = \frac{n}{N}\frac{\Phi_d}{\Phi_v}|S|^2 = \frac{n}{N}\frac{\Phi_d}{\Phi_v}|(I + PK)^{-1}|^2 \qquad (2.1)$$

where: $\hat{S} \approx P(I + PK)^{-1}$ is the estimated model of the system (process in closed loop)
Φ_d and Φ_v are the power spectral densities of d and u
n is the model order
N is the number of samples
S is the real sensitivity given by (2.2):

$$S = (I + PK)^{-1} \qquad (2.2)$$

$\Phi_d|S|^2$ can be interpreted as a new output disturbance Φ_{d^*}, acting on the output y of the model of the system $\hat{S}$ in figure 2.2. A small sensitivity will result in a good model of the system. If we are controlling the tracking behaviour we shall soon see (section 2.2) that we also want a small sensitivity. So identification and control become mutually supporting.

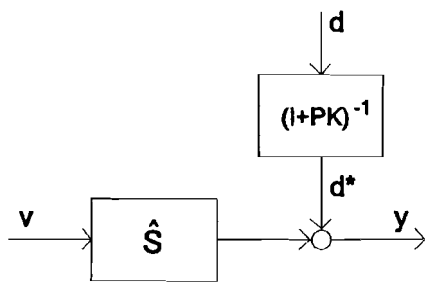


figure 2.2 Coloured noise on estimated system.

There is still one question. Puts the first loop controller constraints on the effect of the second loop controller? Vidyasagar [Vid85] showed that the first loop controller puts no constraints on the effect of the second loop controller if the process is strongly stabilizable which means that the number of all real poles between any pair of real zeros in the Right Half Plane (RHP) is even (including zero's at infinity). It is unknown if this is possible if the system is not strongly stabilizable.
The main purpose of the initial loop controller is stabilizing the process. Because we do not have an estimated model of the process at that time, we propose to design a "loose" initial loop controller which stabilizes the process enough to estimate a linear model, but is far from optimal.

2.2. Modification of TLC for tracking.

If we are interested in tracking behaviour we usually have the schemes of figure 2.3 [Vos92], [Bel93], which are identical when the controller K is stable. The first configuration is advantageous for tracking and the second configuration for identification of the system $\hat{S}$, see [Vos92].

For the tracking yields:

$$e = y - r = (I + PK)^{-1} = Sr \qquad (2.3)$$

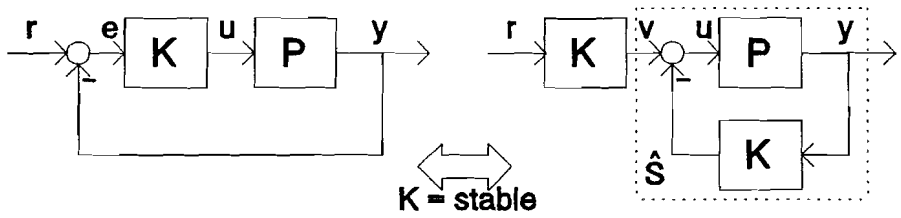


figure 2.3 Configuration for track following.

For good tracking behaviour we want a small sensitivity S given in (2.2), especially for those frequencies that we are interested in. If we wish for example a small tracking error in steady state, the sensitivity S should be small for the lower frequencies.

Till yet the controller K is a LTI output feedback controller. Belt [Bel93] used a special structure of K, a Finite Impulse Response (FIR), so that it won't blow up each iteration. From now on, we will use it as a static state feedback controller. K and ΔK are of the same structure, a static matrix.

2.3. Static state feedback of nonlinear systems.

In this section we focus on the nonlinearity of a process. So we assume that there is no noise involved and that the disturbance is due to the nonlinearity of the system. In practice we are usually not skilled enough to apply nonlinear system control techniques. In system control it is a well known experience that under feedback most nonlinear systems behave more linear in the neighbourhood of a working point x_0. This also holds when we feedback the states of the system. In the two-loop scheme we control the system (process under feedback). If the system is more linear each iteration (identification and controller design), we can estimate a more accurate LTI model. Based on that model we can design a controller with higher performance (e.g. a smaller tracking error) with less robustness.

Define a dynamical nonlinear state space model with n states, p inputs and q outputs in a limited class as:

$$\begin{aligned} \dot{x} &= f(x) + g(x)u \\ y &= h(x) \end{aligned} \tag{2.4}$$

with:

$$\begin{aligned} u &= (u_1,\ldots,u_p)^T \in U \subset \mathbb{R}^p \\ y &= (y_1,\ldots,y_q)^T \end{aligned} \tag{2.5}$$

We assume that all the states are observable and controllable. We exclude those systems which are only controllable with a nonlinear controller. Applying a static state feedback controller results in the scheme of figure 2.4.

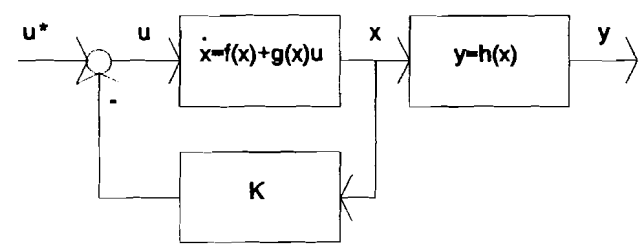


figure 2.4 Static state feedback of a nonlinear system.

A state controller for a linear system is a *static* matrix which can change all system dynamics! Hence it is superfluous to design a *dynamical* state controller based on a linear model of the system. Because the states of the process under consideration (gantry crane) are measured, it is a logical choice to apply state feedback.
For the nonlinear system under feedback yields:

$$\begin{aligned} \dot{x} &= f(x) + g(x)(u^* - Kx) \\ &= f(x) + g(x)u^* - g(x)Kx \\ &= f(x) - g(x)Kx + g(x)u^* \\ &= f_n(x) + g_n(x)u^* \end{aligned} \tag{2.6}$$

$$\begin{aligned} y &= h(x) \\ &= h_n(x) \end{aligned}$$

$$\begin{aligned} f_n(x) &= f(x) - g(x)Kx \\ g_n(x) &= g(x) \\ h_n(x) &= h(x) \end{aligned} \tag{2.7}$$

So under static state feedback the system dynamics change as in (2.7). Now we want that the system under feedback behaves more linear around a working point x_0 than the system without feedback. From (2.7) it is obvious that we can not change the nonlinear dynamics caused by g(x) and h(x) by only a state feedback. When an output feedback is applied, also g(x) and h(x) can change. We will not discuss that here. If f(x) is linear and g(x) is nonlinear, the dynamics under feedback $f_n(x) = f(x) - g(x)Kx$ will not be linear anymore if $K \neq 0$. However, there is a class of systems which do behave more linear under feedback. These systems will be defined here.

First define a linear approximation of the nonlinear system in (2.4) around the working point x_0 as:

$$\begin{aligned} \dot{\tilde{x}} &= A\tilde{x} + B\tilde{u} \\ \tilde{y} &= C\tilde{x} \end{aligned} \tag{2.8}$$

where $\tilde{x}$ is the state vector of the linearized model which can be interpreted as the deviation from the linearization point x_0. $\tilde{u}$ and $\tilde{y}$ are respectively the input and output of the linearized model which can be interpreted as the deviation from u_0 and y_0. A,B and C are obtained using the classical Taylor expansion and are given in (2.9). We assume that f(x), g(x) and h(x) are all smooth enough and that the high order terms tend to zero.

$$\begin{aligned} A &= \frac{\partial f}{\partial x}(x_0) \\ B &= g(x_0) \\ C &= \frac{\partial h}{\partial x}(x_0) \end{aligned} \tag{2.9}$$

where the Jacobian matrix is defined as:

$$\frac{\partial q(x)}{\partial x} = \begin{bmatrix} \frac{\partial q_1}{\partial x_1} & \cdots & \frac{\partial q_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial q_n}{\partial x_1} & \cdots & \frac{\partial q_n}{\partial x_n} \end{bmatrix} \tag{2.10}$$

We must be careful with the linear approximation when the system is operating in a large range. Then it would be better to split up the operating range in smaller parts, estimating a linear approximation in each part.

Given a nonlinear system in (2.4) and any linear approximation in (2.8), the nonlinearity index α of the nonlinear system is defined by (2.11), given a set of input signals U:

$$\alpha(y,\tilde{y}) \equiv \sup_{u \in U} \frac{|y-y_0-\tilde{y}|_2}{|\tilde{y}|_2} \tag{2.11}$$

where α is the supremum of the power (l_2-norm) of the difference between nonlinear system dynamics (without offset y_0) and the linear system dynamics, normed by the power in the linear dynamics. If the power in the linear dynamics is zero $\| \tilde{y} \|_2 = 0$, the nonlinear system has no linear term and $\alpha = \infty$. If the "nonlinear system" is linear $\alpha = 0$.

With this definition we can measure the nonlinearity of a dynamical system, given an input set U (see example appendix 1). For example we can choose U={u=ampl·d(k) | amplϵ[-10,-1,1,10]}, U={u=cos(2πf) | fϵ[1,2,...]}, or U={u | $\| u \|_2 < 1$}. Only when α is small we can estimate an accurate LTI model.
In this section we focused on the nonlinear aspect, and the effect of linearization by static state feedback. As said, there is a class of systems which behave more linear under feedback. For these systems, it holds that $\alpha_{\{fn,gn,hn\}} < \alpha_{\{f,g,h\}}$ (see also appendix 1). But we will see that this is not true for the gantry crane (see section 5.3, table 5.2).

2.4. Two-loop static state feedback for tracking control.

Assume a nonlinear system described by (2.4) with a stabilizing static state feedback, p inputs $u = [u_1,...,u_p]^T$, q outputs $y = [y_1,...,y_q]^T$ and n states $x = [x_1,...,x_n]^T$. Instead of feeding back the states x only, we feedback the states x and the tracking error e = y - r, where r is a q dimensional reference $r = [r_1,...,r_q]^T$, see figure 2.5.

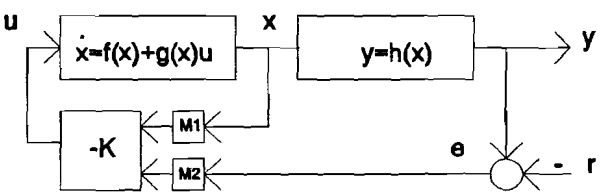


figure 2.5 State transformation providing reference tracking.

The controller generates the new input signal u applied to the process based on the current states and the current tracking error, so that when the states are in steady state, the output y is equal to the reference r. M_1 is a nxn matrix with only n-q entries on the diagonal non zero and equal one, selecting the states which are fed back. M_2 is a nxp matrix of columns with only one element non zero and equal one, leading the multivariable error signals $e = [e_1 \ ... \ e_q]^T$ to the correct input of the controller K. Now we can split up the controller in a static state feedback part K_1 and a static output feedback part K_2 which is a square matrix, see figure 2.6 where K_1 and K_2 are given by:

$$\begin{aligned} K_1 &= KM_1 \\ K_2 &= KM_2 \end{aligned} \tag{2.12}$$

and u is given by:

$$u = [K_1 K_2] \begin{bmatrix} x \\ e \end{bmatrix} = K_1 x + K_2 e \tag{2.13}$$

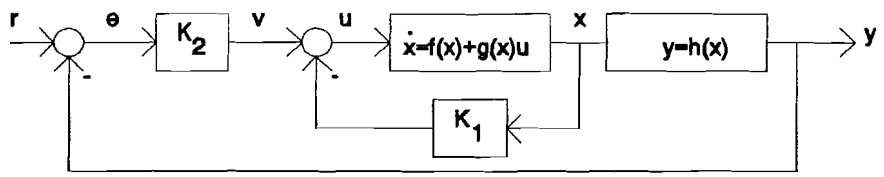


figure 2.6 Partitioning controller in state and output feedback.

Assume that the states x and the output y are measured. Then a LTI model $\hat{S}$ of the system transfer from reference r to states x and a LTI model $\hat{H}$ for the transfer from states x to output y can be estimated. An important feature is that we can recover the signal v, because K_2 is a square (invertible) static matrix. v is given by $v = K_2 e$ and the second loop controller output can be added to the signal v. This means that no additional measurement of the signal v is necessary while the two-loop structure can still be used. For the gantry crane we assume that h(x) is known. Measurement of the output y is then superfluous. We will detail this in section 5.1. When the second loop controller is placed parallel to the first loop controller, we obtain the ultimate two-loop static state tracking control scheme, shown in figure 2.7.

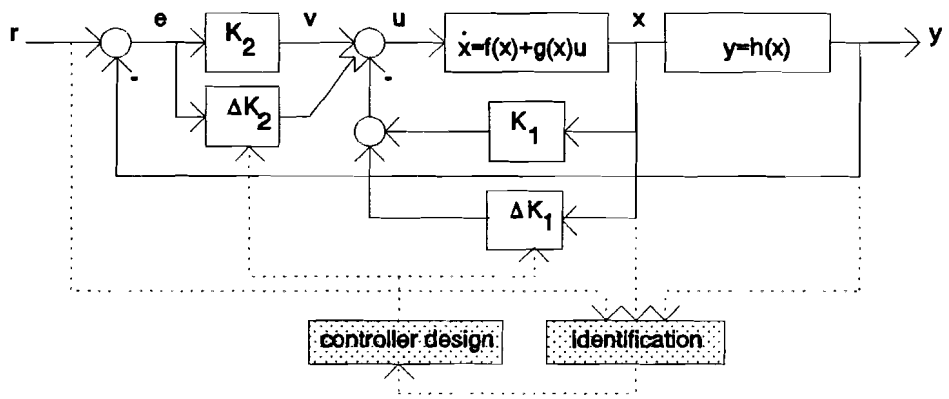


figure 2.7 Ultimate two-loop static state control scheme for tracking.

We summarize the various steps:

1) Design a stabilizing first loop *static* state controller, K which stabilizes the system, but is far from optimal.
2) Estimate the transfer from r to x with a LTI model $\hat{S}$ and the transfer from x to y with a LTI model $\hat{H}$.
3) Design the second loop static state controller ΔK based on the estimated model of the system $\hat{S}$ and $\hat{H}$ for optimal reference control taking some robustness into account.
4) Add ΔK and K to form the new first loop controller K + ΔK
5) Execute steps 2 to 4 again and if the estimated model is more accurate, we can put more constraints on the optimality and less on the robustness.

3. The gantry crane.

The gantry crane is used to test the two-loop control scheme with static state feedback. A dynamical nonlinear state space model is derived, implemented in SIMULAB and used to generate data for experimental controller design for the laboratory process. A linearized model is used to design a first loop controller. Some characteristics of the plant will be explained. Finally, sensor and actuator noise, together with some nonlinearities in the actuators, are modelled too.

3.1. A nonlinear model of the gantry crane.

The gantry crane is used to transport a load through the air from one place to the other. It exists of a movable trolley on a rail and a cable of varying length, see figure 3.1.

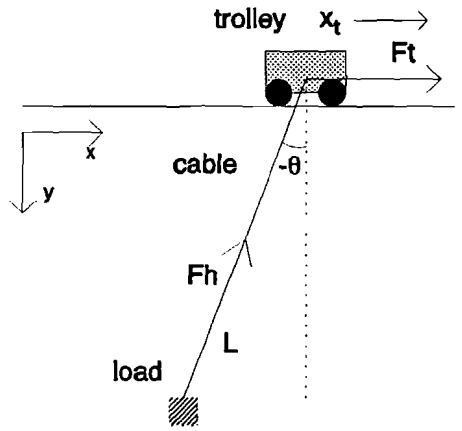


figure 3.1 Simplified gantry crane.

The six measured states (n=6) of the system are: the position of the trolley x_t bounded by the length of the rail ($-1 \leq x_t \leq 1$ [m]), the angle θ, bounded by the sensor ($-0.5 \leq \theta \leq 0.5$ [rad]), the length of the cable L, bounded by the height of the rail ($0.1 \leq L \leq 2$ [m]) and all their derivatives $\dot{x}_t$, $\dot{\theta}$, and $\dot{L}$. The derivatives are actually calculated but using a high sample frequency we may assume that we measure them. The two inputs (p=2) of the system are the forces on the trolley F_t and the rope F_h both bounded by the maximum motor torque ($-100 \leq F_t, F_h \leq 100$ [N]). The two outputs (q=2) are the position of the load given by x_l and y_l, which must follow a predefined track given by x_{ref} and y_{ref}.

The nonlinear model that we use is based on the model used in [Sch93]. A comprehensive derivation is given in Appendix 2. We repeat here the state space description:

$$\begin{aligned} \dot{x} &= f(x) + g(x)u \\ y &= h(x) \end{aligned} \tag{3.1}$$

with input u, state x, output y and reference track r:

$$x = \begin{bmatrix} x_t & \dot{x}_t & \theta & \dot{\theta} & L & \dot{L} \end{bmatrix}^T \quad u = \begin{bmatrix} F_t & F_h \end{bmatrix}^T \quad y = \begin{bmatrix} x_l & y_l \end{bmatrix}^T \quad r = \begin{bmatrix} x_{ref} & y_{ref} \end{bmatrix}^T \tag{3.2}$$

and f(x), g(x) and h(x) are given in (3.3) where:

mass of the trolley	$m_t = 3.5$ kg;
mass of the load	$m_L = 1.5$ kg;
gravitation acceleration	$g = 9.81$ m/s^2;
linear friction in trolley	$\alpha_t = 0.1$ Ns/m;
linear friction in angle	$\alpha_\theta = 0.01$ Ns/rad;
linear friction in cable	$\alpha_L = 10$ Ns/m;

$$f(x)=\begin{pmatrix} \dot{x}_t \\ -\frac{\alpha_t}{m_t}\dot{x}_t \\ \dot{\theta} \\ -\frac{g}{L}\sin(\theta) + \frac{\alpha_t}{m_t L}\dot{x}_t\cos(\theta) - (2\frac{\dot{L}}{L} + \frac{\alpha_\theta}{m_L})\dot{\theta} \\ \dot{L} \\ g\cos(\theta) + L\dot{\theta}^2 + \frac{\alpha_t}{m_t}\dot{x}_t\sin(\theta) - \frac{\alpha_L}{m_L}\dot{L} \end{pmatrix} \quad g(x)=\begin{pmatrix} 0 & 0 \\ \frac{1}{m_t} & \frac{1}{m_t}\sin(\theta) \\ 0 & 0 \\ -\frac{1}{m_t L}\cos(\theta) & -\frac{1}{m_t L}\sin(\theta)\cos(\theta) \\ 0 & 0 \\ -\frac{1}{m_t}\sin(\theta) & -\frac{1}{m_L}-\frac{1}{m_t}\sin^2(\theta) \end{pmatrix} \tag{3.3}$$

$$h(x)=\begin{pmatrix} x_t + L\sin(\theta) \\ L\cos(\theta) \end{pmatrix}$$

The mass of the load is taken constant, but it is possible to define it as an additional input in a more complex approach.
This model is an approximation of the true plant. The friction terms are linear, so no attention is paid at the dead zone and saturation behaviour. In the real case it is not possible to put a force on the load in the positive y-direction which is possible in the model. But this will only occur when $F_h < -m_L g$, which can be constrained during controller design. Further we assume that the cable does not bow. But still this model has great value, because it gives us an indication of the structure of system and how the real nonlinear crane reacts on control actions. Further $L \neq 0$ must hold else we divide by zero. A additional remark can be made when θ is small. When we tolerate an error of 1% in $\sin(\theta)$ then we may replace it by θ if $\theta < 0.39$ rad. In the same way we may replace $\cos(\theta)$ by $1-\theta$. This does not mean that the output has a tolerance of 1%. We can derive the propagation of faults from the states x to the output y. Therefore we look at the change of the output when we change one state slightly and keep the other states constant. Then follows from h(x) by partial differentiating with respect to the changing state:

$$\frac{\partial x_l}{\partial x_t} = 1 \quad \frac{\partial x_l}{\partial \theta} = L\cos(\theta) \quad \frac{\partial x_l}{\partial L} = \sin(\theta)$$
$$\frac{\partial y_l}{\partial x_t} = 0 \quad \frac{\partial y_l}{\partial \theta} = -L\sin(\theta) \quad \frac{\partial y_l}{\partial L} = \cos(\theta) \tag{3.4}$$

This means that for example an error in the position of the trolley x_t, will always result in an error of the same size in the x-position of the load x_l. But an error in the angle θ will result in an error in the x-position of the load multiplied with $L\cos(\theta) \approx L$ if $\theta \approx 0$.

To investigate if the nonlinear model of the system is strongly stabilizable we introduce the zero dynamics of a nonlinear system which can be interpreted as the 'zeros' of a linear system [Isi80]. They tell us for which input signals and initial conditions the output can be kept at an equilibrium for all times. The states of the system are than constrained to evolve so that $y(t) = y_0 \ \forall t$.
In a more mathematical way we define the zero dynamics as:

$$zero\ dynamics := \{u_{zd}(t) \mid y(t) = y_0 \ \ \forall\ t\} \tag{3.5}$$

When there exist no unstable zero dynamics it holds that:

$$\lim_{t\to\infty} u_{zd}(t) = 0 \tag{3.6}$$

which means that the system is "minimum phase" in terms of linear system theory. In that case a linear system has no zeros in the Right Half Plane (RHP). This guarantees that the linear system is strongly stabilizable, and the two-loop structure can be used. We expect that also for nonlinear systems with stable zero dynamics the two-loop structure can be used. In appendix 4 the zero

dynamics of the gantry crane are derived. The resulting zero dynamics are stable because $\lim_{t\to\infty} u_{zd}(t) = 0$, and the system is "minimum phase". Hence we expect that the two-loop structure can be used.

To investigate if the nonlinear system is controllable we first explain this concept for a linear system. A system is completely state controllable if for any initial state, the state of the system can be brought to any final state in finite time by appropriate control [Boo92]. A linear system is controllable if and only if $\Delta = [B\ AB\ A^2B\ \dots\ A^{n-1}B]$ is non singular (has full rank = n). This holds for both continuous and discrete time, where the continuous LTI state space model is defined by (3.15) and the discrete LTI state space model by (3.7):

$$\begin{aligned} \tilde{x}(k+1) &= A\tilde{x}(k) + B\tilde{u}(k) \\ \tilde{y}(k) &= C\tilde{x}(k) \end{aligned} \tag{3.7}$$

where $\tilde{u}$ the p dimensional input, $\tilde{y}$ the q dimensional output, and $\tilde{x}$ is a n-dimensional vector $[\tilde{x}_1 \dots \tilde{x}_n]^T$. This can easily be understood by writing out the states for each time in the discrete time case:

$$\begin{aligned} &\tilde{x}(1) = A\tilde{x}_0 + B\tilde{u}(0) \\ &\tilde{x}(2) = A\tilde{x}(1) + B\tilde{u}(1) = A^2\tilde{x}_0 + B\tilde{u}(1) + B\tilde{u}(2) \\ &\dots \\ &\tilde{x}(k) = A^k\tilde{x}_0 + \underbrace{[B\ AB\ A^2B\ \dots\ A^{k-1}B]}_{\Delta} \begin{bmatrix} \tilde{u}(k-1) \\ \tilde{u}(k-2) \\ \vdots \\ \tilde{u}(0) \end{bmatrix} \end{aligned} \tag{3.8}$$

If we have n controlling moments, we have n degrees of freedom for control of $\tilde{x}(k)$ so we can bring $\tilde{x}(k)$ to any desired state using appropriate u under the strict condition that the matrix Δ has full rank. We can interpret this as if we are in some point x_0 in an n-dimensional linear vector space $\tilde{X}$ and we want to another point x_e in that space. For each control action we apply, we walk in a direction of that space. But if we have not enough freedom in direction, we can never reach the point x_e. The basis of control actions *must* have the same dimension as the basis of the vector space to reach every x_e we want. This idea can be generalized for nonlinear systems. Than we have to deal with nonlinear vector spaces, so the derivation is slightly more difficult [Nij92].
Consider the nonlinear system with p inputs and q outputs:

$$\begin{aligned} \dot{x} &= f(x) + g(x)u \\ y &= h(x) \end{aligned} \tag{3.9}$$

on a n-dimensional state space X. This system is controllable if the span of all repeated Lie brackets of $f, g_1, \dots, g_p$ is n-dimensional at every point of X.
The Lie bracket is defined as a mapping from $R^n \to R^n$:

$$[f,g](x) = \frac{\partial g}{\partial x} f(x) - \frac{\partial f}{\partial x} g(x) \tag{3.10}$$

In appendix 5 we derived that the gantry crane is controllable. It will be clear that this nonlinear approach is only possible if we have an exact description of the system. But this will never be true because we always have a limited model set. Further the approach is very time consuming. I will confine myself to remarking the possibility of exact linearizing state feedback [Nij92]. This would lead to a linear system without neglecting higher order terms, and all linear controller design methods are valid again.

To stabilize the system we apply an initial loop controller obtained by LQR controller design. Therefore we need a linear model of the gantry crane.

3.2. A linear model of the gantry crane.

Using the classical Taylor expansion (see section 2.2) we can linearize the model. For a comprehensive derivation see Appendix 3. As linearization point $x_0 = [0\ 0\ 0\ 0\ 1\ 0]^T$ we take the middle of the xy-plane where the load can be moved in, and defined by (3.11):

$$\begin{aligned} x_t &= x_{t_0} + \Delta x_t = \Delta x_t \\ \dot{x}_t &= \dot{x}_{t_0} + \Delta \dot{x}_t = \Delta \dot{x}_t \\ \theta &= \theta_0 + \Delta\theta = \Delta\theta \\ \dot{\theta} &= \dot{\theta}_0 + \Delta\dot{\theta} = \Delta\dot{\theta} \\ L &= L_0 + \Delta L = 1 + \Delta L \\ \dot{L} &= \dot{L}_0 + \Delta\dot{L} = \Delta\dot{L} \end{aligned} \tag{3.11}$$

Redefine the state vector as the deviation from the linearization point x_0 and the input vector:

$$\begin{gathered} \tilde{x} = [\Delta x_t\ \Delta \dot{x}_t\ \Delta\theta\ \Delta\dot{\theta}\ \Delta L\ \Delta\dot{L}]^T \\ \tilde{u} = [\Delta F_t\ \Delta F_h]^T \quad \tilde{y} = [\Delta x_l\ \Delta y_l]^T \quad \tilde{r} = [\Delta x_{ref}\ \Delta y_{ref}]^T \end{gathered} \tag{3.12}$$

where

$$\begin{aligned} F_t &= F_{t_0} + \Delta F_t = m_L \cdot g + \Delta F_t \\ F_h &= F_{h_0} + \Delta F_h = \Delta F_h \\ x_l &= x_{l_0} + \Delta x_l = \Delta x_l \\ y_l &= y_{l_0} + \Delta y_l = 1 + \Delta y_l \\ x_{ref} &= x_{ref_0} + \Delta x_{ref} = \Delta x_{ref} \\ y_{ref} &= y_{ref_0} + \Delta y_{ref} = 1 + \Delta y_{ref} \end{aligned} \tag{3.13}$$

The linear system is now given by (3.14):

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & -\frac{\alpha_t}{m_t} & \frac{m_L \cdot g}{m_t} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & \frac{\alpha_t}{m_t \cdot L_0} & -\frac{g}{L_0}\left(1+\frac{m_L}{m_t}\right) & -\frac{\alpha_\theta}{m_L} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & -\frac{\alpha_L}{m_L} \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 \\ \frac{1}{m_t} & 0 \\ 0 & 0 \\ -\frac{1}{m_t \cdot L_0} & 0 \\ 0 & 0 \\ 0 & -\frac{1}{m_L} \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 & L_0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \tag{3.14}$$

Note that the model is decoupled. The states ΔL and $\Delta\dot{L}$ are independent from the other states. Looking at the zero and poles:

zeros: -1.4627e+3

poles: 0
-0.0196
-0.0075 + 3.7416i
-0.0075 - 3.7416i
0
-6.7000

where the first four poles are from Δx_t $\Delta \dot{x}_t$ $\Delta\theta$ $\Delta\dot{\theta}$ (fourth order system) and the last two poles from ΔL $\Delta \dot{L}$ (second order system). The A matrix has not full rank (n=6 and rank(A)=4) because two eigenvalues are zero (integration). The complex conjugated pole pair with the small real part is due to the badly damped pendulum frequency. The linear system has no zeros in the RHP (minimum fase). There are no unstable poles, but because of the small negative real part of the poles and the poles at zero, the system is not asymptotically stable. Further the system is fully controllable and observable [boo_87] because Δ = [B AB A^2B ... $A^{n-1}B$] and Γ = [C CA CA^2 ... CA^{n-1}] have full rank (n=6). It is not our purpose to study this linear model thoroughly. It is only one way to find an initial first loop controller necessary to get rid of the poles in zero so that the system is stable and a linear model can be estimated.

3.3. LQR first loop controller design in continuous time.

Given the following linear model:

$$\begin{aligned} \dot{\tilde{x}} &= A\tilde{x} + B\tilde{u} \\ \tilde{y} &= C\tilde{x} \end{aligned} \tag{3.15}$$

with: dim[A] = nxn;
dim[B] = nxp;
dim[C] = qxn;

and with characteristic polynomial: $a(s) = \det(sI-A) = s^n + a_1 s^{n-1} + \ldots + a_n$

We can change this polynomial using static state feedback: $\tilde{u} = \tilde{u}^* - K\tilde{x}$, see figure 3.2.

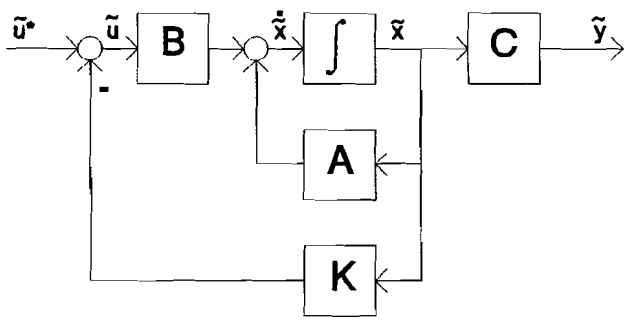


figure 3.2 LQG or state feedback control.

Now the new state space model is defined by:

$$\begin{aligned} \dot{\tilde{x}} &= (A-BK)\tilde{x} + B\tilde{u}^* \\ \tilde{y} &= C\tilde{x} \end{aligned} \tag{3.16}$$

with characteristic polynomial: $a(s) = \det(sI-A+BK) = s^n + \alpha_1 s^{n-1} + \ldots + \alpha_n$

Since {A,B} is controllable, the poles of this characteristic polynomial may be put in any predefined position by the choosing elements in K properly [Dam90].

Now let the system be at rest in $\tilde{y}_0$. Due to some (impulsive) disturbance, the output at some time t_0 is replaced by $\tilde{y}(t_0) \neq \tilde{y}_0$. The regulator problem is to apply a control signal u such that the output $\tilde{y}$ is returned to its equilibrium $\tilde{y}_0$. We can translate the problem to the states. We want to apply a control input u which has to return the disturbed state vector $\tilde{x}(t_0)$ to the equilibrium state vector $\tilde{x}_0$. Therefore we can choose a suitable K so that $\tilde{x}(t_0)$ reaches $\tilde{x}_0$ as fast as we wish. The rate of decay depends on how negative the real parts of the eigenvalues of A-BK are. The more negative these are, the larger the values of K will be and therefore the higher the required input

signal energy. Clearly we are dealing with an optimization problem. We have to deal with two constraints. First we want $\bar{y}(t) = \bar{y}_0$ and second there is a limit in the energy of $\bar{u}(t)$. Therefore we have to define a performance so that we can calculate the best solution. We take:

$$J = \int_{t_0}^{\infty} \left[\bar{x}^T(t) Q \bar{x}(t) + \bar{u}^T(t) R \bar{u}(t)\right] dt \tag{3.17}$$

subject to $Q \geq 0$, $R \geq 0$.
This provides that $\lim_{t \to \infty} x(t) = 0$ and we obtain an optimal *stable* linear state feedback controller K. Using Langrange multipliers we can write the optimization problem in a Algebraic Riccati Equation (ARE):

$$0 = PA + A^T P - PBR^{-1}B^T P + Q \tag{3.18}$$

and K is given by:

$$K = R^{-1}BP \tag{3.19}$$

We can solve the ARE using the function lqr.m in MATLAB. Now we have to choose a proper Q and R matrix. The Q matrix is already defined, because we want to control the output signal. Essentially we want to minimize $\bar{y}$ so:

$$J = \int_{t_0}^{\infty} \left[\bar{y}^T(t) \bar{y}(t) + \bar{u}^T(t) R \bar{u}(t)\right] dt \tag{3.20}$$

Remember that in the linear system all variables are defined as the deviation from the linearization point.
Since $\bar{y} = C\bar{x}$ we can write:

$$J = \int_{t_0}^{\infty} \tfrac{1}{2} \left[\bar{x}^T(t) C^T C \bar{x}(t) + \bar{u}^T(t) R \bar{u}(t)\right] dt \tag{3.21}$$

From (3.21) and (3.17) follows:

$$Q = C^T C = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{3.22}$$

Now we must choose R in such way that the effort in the actuators F_t and F_h is only small obtaining a "loose" controller as suggested in section 2.1. After some trial and error the next Q and R are chosen:

$$Q = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 10 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad R = \begin{bmatrix} 0.1 & 0 \\ 0 & 0.1 \end{bmatrix} \tag{3.23}$$

Q is modified in such way that there is a stronger feedback in the $\Delta\theta$, else $\Delta\theta$ was badly damped which followed from validation of the controller on the linear model. The actuator effort (only low power dissipation for optaining a "loose" controller) was also checked during the validation. This controller also aligned the length of the impulse responses of the output $\bar{y}$ and hence the impulse response of the states $\bar{x}$. This will prevent numerical problems when we are sampling the states of the real crane. When the impulse responses of the states are not of the same length, sampling at a high rate will result in no difference in the samples of the 'slow' impulse responses,

while a low sample rate will cause aliasing effects in the 'quick' impulse responses. In fact we move the real parts of the poles to each other in a suitable region so that we obtain a suitable bandwidth of the system. The resulting initial state controller K equals:

$$K = \begin{bmatrix} 3.1623 & 5.8638 & -2.9401 & -0.5707 & 0 & 0 \\ 0 & 0 & 0 & 0 & -3.1623 & -0.4636 \end{bmatrix} \tag{3.24}$$

Note that the controller has six structural zeros which are caused by the decoupling of the linearized system.
The new poles are:

-0.3686 + 3.7651i
-0.3686 - 3.7651i
-0.5682 + 0.5444i
-0.5682 - 0.5444i
-6.6592
-0.3166

From the structure of the decoupled linear system it also follows that Δx_1 is obtained from a fourth order differential equation and Δy_1 from a second order differential equation.

3.4. The gantry crane with linear static state feedback.

Applying the static state feedback controller (3.24) to the nonlinear system, we have to correct for the constant gravity term. The resulting system is implemented in SIMULAB (see appendix 5). Figure 3.3 shows the deterministic simulation model where no noise is involved.

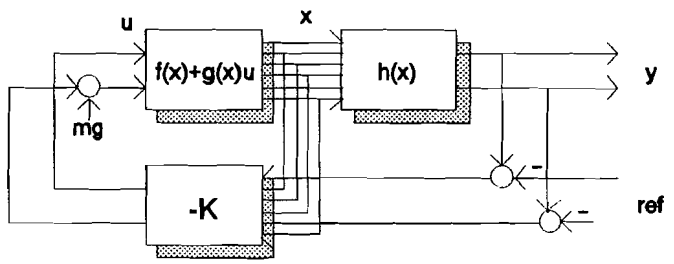


figure 3.3 Deterministic simulation model.

There is some danger in applying this linear controller to the nonlinear system because we do not know if the nonlinear system will be stabilized by this controller, even if we design a "loose" one. If we look at the linearization of L we choose $L = L_0 + \Delta L$ where $L_0 = 1$. With $-0.9 \leq \Delta L \leq 1$ we can not speak of a small deviation from the linearization point anymore and hence the linearized model can be very different from the nonlinear model of the gantry crane. Therefore we verified that the nonlinear system under state feedback with the initial state controller K(0) in (3.22) is stable for an arbitrary set of input signals U_{st} (no plots here). But for the design of the second loop controller we must take some robustness into account.
See that instead of the states x_t and L, the error $(x_t - x_{ref})$ and respectively (y_1-y_{ref}) are fed back. This is possible because in steady state θ equals zero and hence:

$$h(x_e) = \begin{cases} x_t + \cos(\theta) \\ L\sin(\theta) \end{cases} = \begin{cases} x_t \\ L \end{cases} \quad \wedge\ \theta = 0 \tag{3.25}$$

where x_e is the state vector in a steady state. In fact we apply a state transformation so that the states tend to x_e. See also section 5.1.

To gather more knowledge of the closed loop system behaviour, the response of the nonlinear system on a reference impulse in respectively the x- and y-direction is shown in figure 3.4.

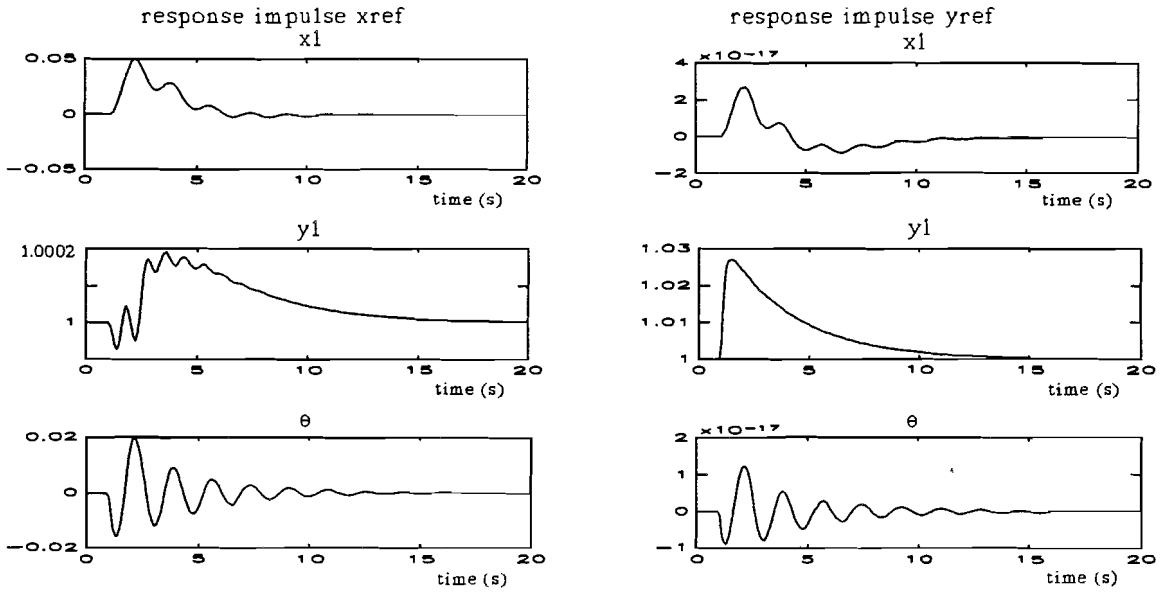


figure 3.4 Closed loop impulse respones.

The behaviour shows indeed a minimum phase characteristic in terms of linear system theory, because the "first" response is not opposite to the reference signal. The coupling between y_{ref} and x_l and between y_{ref} and θ is due to numerical errors, so the numerical accuracy is about 10^{-16}. For an impulse on x_{ref} the response of y_l is more than two decades smaller than the response of x_l. This would suggest that the coupling could be neglected, but as we will see soon, this is not allowed when we use a white uniformly distributed reference signal. Because the positive y-direction of the load is downwards, the response of y_l for an impulse on x_{ref} is indeed minimum phase in terms of linear system theory. From the badly damped response of θ for an impulse on x_{ref}, we detect the pendulum frequency f_p caused by the swinging of the pendulum. If we neglect the mass of the cable and assume that the cable does not bow, f_p equals (mathematical pendulum):

$$f_p = \frac{1}{2\pi}\sqrt{\frac{g}{L}} \approx 0.5\ Hz \quad for\ L = 1 \tag{3.23}$$

where $g = 9.81\ m/s^2$ is the gravitational acceleration. This means that the pendulum time T_p equals $T_p \approx 2$ seconds. This pendulum time is visible in the response of x_l and θ for an impulse on x_{ref}. In the response of y_l for an impulse on x_{ref} we detect the doubled frequency of the pendulum, due to the fact that when the pendulum makes only one period in the x-direction, it will make two periods in the y-direction. This is a tipical nonlinear effect, and we need an additional complex pole pair in a linear system to describe the doubled frequency.

The closed loop transfer from the state space model with linear state feedback K can be calculated by $H_{cl}(j\omega) = C(sI - A + BK)^{-1}B$. But for the nonlinear system this is not possible. To get an indication of the "bandwidth" of the system, we use a Fourier transformation based on input and output data sampled with $f_s = 10$ Hz. In fact we estimate a high order LTI approximation for the spectra of the input and output data. An indication for the "bandwidth" of a nonlinear system is than gathered from the plot of the ratio of the output and input spectra for each frequency.

$$F\begin{pmatrix} y \\ u \end{pmatrix} = \begin{pmatrix} \hat{y}(\omega) \\ \hat{u}(\omega) \end{pmatrix} \quad plot: \quad \frac{\hat{y}(\omega)}{\hat{u}(\omega)} \tag{3.24}$$

The input is chosen a uniformly distributed white noise:

x_{ref} = uniform distributed white noise ϵ [-3,3], mean(x_{ref}) = 0

y_{ref} = uniform distributed white noise ϵ [-3,5], mean(y_{ref}) = 1

This does not guarantee that a nonlinear system is exited enough. Therefore the results are only indicating the bandwidth of the system.

The spectra obtained with the Fourier transformations based on the data of the inputs and outputs from the nonlinear system, are given in figure 3.5.

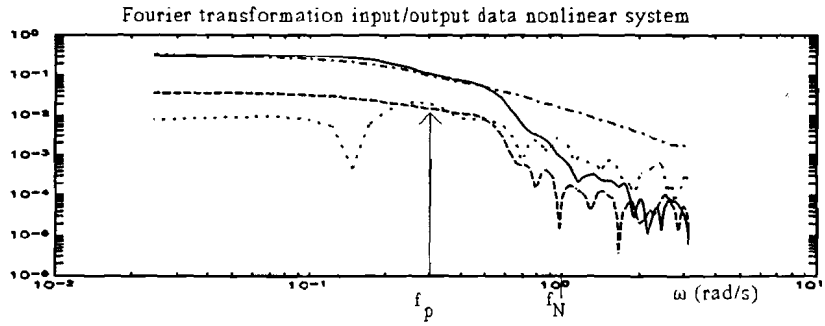


figure 3.5 Closed loop spectra from r to y of the nonlinear system.

where :

— = spectrum from x_{ref} to x_l

--- = spectrum from y_{ref} to x_l

... = spectrum from x_{ref} to y_l

.-. = spectrum from y_{ref} to y_l

and where $\omega = \pi$ rad/s equals $\frac{1}{2}f_s = 5$ Hz, half the sampling rate.

For the Fourier transformation of the data from x_{ref} to y_l and from y_{ref} to x_l we detect that there is coupling in the nonlinear model. This was not visible in the impulse responses. But now the input was a uniform distributed white noise, and θ is not negligible anymore, and also coupling between y_{ref} and x_l can physically be explained. For frequencies above 1 rad/s the damping is higher than 10^{-2}, which can be interpreted as the "bandwidth" of the nonlinear gantry crane. Again the pendulum frequency is visible in the Fourier transformation of the data from x_{ref} to y_l. In the spectrum is a large amount of power concentrated in the frequencies around $\omega = 0.3$ rad/s equal to $f \approx 0.5$ Hz corresponding to the pendulum frequency $f_p \approx 0.5$ Hz for L = 1.

A nice interpretation can be given to the spectra from x_{ref} to θ and from y_{ref} to θ, shown in figure 3.6.

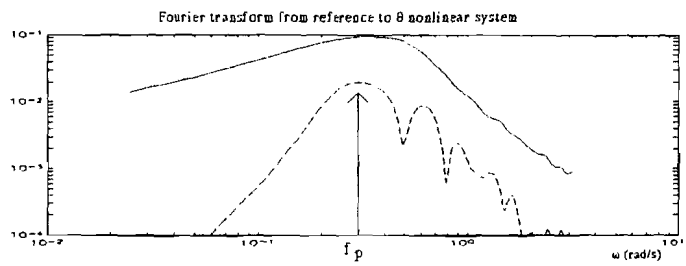


figure 3.6 Closed loop transfer from r to θ nonlinear model.

where:

— = spectrum from x_{ref} to θ
--- = spectrum from y_{ref} to θ

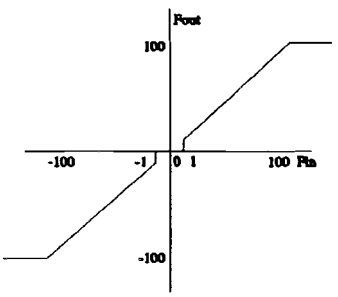
This spectrum has the most power at the frequencies around the pendulum frequency.

For identification figure 3.4 is extended with sensor noise η and actuator noise σ (see appendix 6) both modelled as Gaussian zero mean white noise with:

η_{xt}	ϵ [-0.1, 0.1]	η_L	ϵ [-0.1, 0.1]
η_{xtdot}	ϵ [-0.2, 0.2]	η_{Ldot}	ϵ [-0.2, 0.2]
η_{theta}	ϵ [-0.05, 0.05]	σ_{Ft}	ϵ [-1, 1]
$\eta_{thetadot}$	ϵ [-0.1, 0.1]	σ_{Fh}	ϵ [-1, 1]

This means a sensor noise of 10% and an actuator noise of 1% which can be seen as a worst case model. Motor nonlinearities e.g. saturation and dead zones (caused by cogwheels) are also modelled with (for both forces F_t and F_h):

$$F_{out} = \begin{cases} 100 & F_{in} > 100 \\ F_{in} & 1 \le F_{in} \le 100 \\ 0 & -1 < F_{in} < 1 \\ F_{in} & -100 \le F_{in} \le -1 \\ -100 & F_{in} < -100 \end{cases}$$



The resulting model is shown in figure 3.7. Because we measure the states of the system, the sensor noise is added to the states. Using the function h(x), given in (3.25), with high reliability, we can calculate the virtually measured output y^*. y is the real output of the gantry crane.

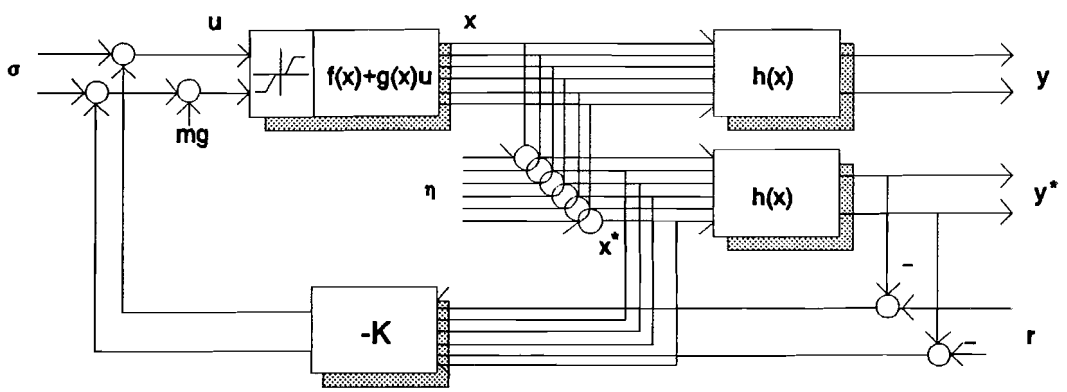


figure 3.7 Stochastic model of the system in closed loop.

This scheme is used for identification in the next section.

4. Identification of the system.

System identification deals with the problem of estimating mathematical models of dynamical systems based on observed data from the system. In the next sections a linear time invariant Minimum Polynomial output error model of the gantry crane is estimated based on input/output data simulated with the nonlinear model. The identification method and experimental conditions will be explained and the resulting model is validated. The estimated model can be used to compare with the results of other identification methods.

4.1. Minimum Polynomial Estimation.

Estimation of a MIMO LTI model based on observed input/output data is most times a delicate problem. Due to a limited model set and all kinds of disturbances (actuator noise, sensor noise, nonlinearities) the model will never describe the system perfectly. Finding the best model in a space spanned by the number of parameters results in a nonlinear optimization problem. Decent methods for finding a minimum get often stuck in local minima of the cost function. This is a function which measures the performance of the model with e.g. the two norm on the error signal. A good initial model can help to find the global minimum. For the gantry crane we used Minimal Polynomial Estimation (MPI) [Fal93] for obtaining a synthetic, dynamical, linear, time invariant, lumped, parametric, causal, and discrete time model, see [Boo87], [Söd89]. Define the model set:

$$A(z^{-1})y(k) = \frac{B(z^{-1})}{F(z^{-1})}u(k) + \frac{C(z^{-1})}{D(z^{-1})}d(k) \tag{4.1}$$

where:

$$\begin{aligned} A(z^{-1}) &= (1 + a_1 z^{-1} + \ldots + a_{na} z^{-na}) I_q \\ B_{ij}(z^{-1}) &= B_{ij,0} + B_{ij,1} z^{-1} + \ldots + B_{ij,nb_{ij}} z^{-nb_{ij}} \\ C(z^{-1}) &= C_0 + C_{i,1} z^{-1} + \ldots + C_{i,nc_i} z^{-nc_i} \\ D(z^{-1}) &= 1 + d_1 z^{-1} + \ldots + d_{nd} z^{-nd} \\ F(z^{-1}) &= 1 + f_1 z^{-1} + \ldots + f_{nf} z^{-nf} \end{aligned} \tag{4.2}$$

where z^{-1} is the delay operator defined by $z^{-1}q(k) = q(k-1)$. The polynomials $A(z^{-1})$, $D(z^{-1})$ and $F(z^{-1})$ are scalar polynomials while $B(z^{-1})$ and $C(z^{-1})$ are matrix polynomials of size (q x p) and (q x q) respectively. The structure of the model is represented by [na, nb_{ij}, nc_i, nd, nf, nk_{ij}], where nk is the number of delays from input j to output i. Assume that $C(z^{-1})$ is a diagonal matrix polynomial for reducing the complexity. This in fact eliminates the true MIMO character of the noise model. u(k) are the p process inputs at time kT_s where $T_s = 1/f_s$ is the sample time and d(k) the q dimensional disturbance vector at time kT_s. The q process outputs at time kT_s are denoted by y(k). We define the parameter vector $\underline{\theta}$ containing all parameters of $A(z^{-1})$,$B(z^{-1})$,$C(z^{-1})$,$D(z^{-1})$ and $F(z^{-1})$ which must not be confused with θ which is the measured angle of the gantry crane! Based on the data Z^N = [y(0), u(0); y(1), u(1); ... ; y(N), u(N)] with N input/output data samples we search for the best $\underline{\theta}$ obtained from a cost function $V(\underline{\theta},Z)$:

$$\hat{\underline{\theta}} = \underset{\hat{\underline{\theta}} \in \underline{\theta}}{arg\ min}\ V(\underline{\theta},Z^N) \tag{4.3}$$

A natural choice for obtaining a model for controller design is the least squares approach:

$$V(\underline{\theta},Z^N) = \frac{1}{N}\sum_{k=1}^{N} e^T(k)e(k) \tag{4.4}$$

where e(k) is defined as:

$$e(k) = y(k) - \hat{y}(k) \tag{4.5}$$

where y(k) is the real process output and $\hat{y}(k)$ the estimated output. After estimating the parameters in $\underline{\theta}$ the obtained model is the best model in least squares sense!
Methods for numerical minimization of the function $V(\underline{\theta},Z^N)$ update the estimate of the minimizing argument iteratively according to:

$$\hat{\theta}^{i+1} = \hat{\theta}^i + \beta \cdot f^i \tag{4.6}$$

where f^i is the search direction based on information about $V(\underline{\theta},Z^N)$ gathered at previous iterations, and ß is a constant, determined so that an appropriate decrease in the value of $V(\underline{\theta},Z^N)$ is obtained. In this particular case it is possible to give an analytical expression for f^i in the Gauss-Newton direction [Fal93]. This iterative search procedure leads to a model which corresponds with a local minimum in the cost function. Nothing guarantees that this local minimum is also the global minimum. Therefore the initial model is of crucial importance. To obtain an initial model in the neighbourhood of the global minimum we use the method proposed by Zhu [Zhu_91]. This method is based on asymptotic properties of the estimated model. Define the models:

$$y(k) = \sum_{l=1}^{\infty} \left(g_l u(k-l) + h_l d(k-l)\right) \tag{4.7}$$

g_l and h_l are the parameters of the impulse response of the model and the disturbance respectively. With the unity delay operator z^{-1} the model in (4.7) can be written as:

$$y(k) = G(z^{-1})u(k) + H(z^{-1})d(k) \tag{4.8}$$

where: $G(z^{-1})=A^{-1}(z^{-1})F^{-1}(z^{-1})B(z^{-1})$
$H(z^{-1})=A^{-1}(z^{-1})D^{-1}(z^{-1})C(z^{-1})$
With this model we associate the transfer functions:

$$\begin{aligned} G(e^{j\omega}) &= \sum_{k=1}^{\infty} g_k e^{-jk\omega} \qquad -\pi \le \omega \le \pi \\ H(e^{j\omega}) &= \sum_{k=1}^{\infty} h_k e^{-jk\omega} \qquad -\pi \le \omega \le \pi \end{aligned} \tag{4.9}$$

Under certain conditions Ljung [Lju85] proved that:

$$\begin{aligned} &\hat{G}_N^n(\hat{\underline{\theta}},e^{j\omega}) \to G_t(e^{j\omega}) \\ &\text{if } n \to \infty \text{ and } \frac{n}{N^2} \to 0 \end{aligned} \tag{4.10}$$

where $\hat{G}_N^n(\hat{\underline{\theta}},e^{j\omega})$ is the n^{th} order estimated model transfer based on N samples input/output data, with parameter vector $\hat{\underline{\theta}}$ and $G_t(e^{j\omega})$ is the true process transfer. This means that if we take the order of the estimated process big enough, and the number of data is still much bigger than the order, the estimated model equals the true model.
When we take $F(z^{-1})=1$, we obtain an equation error (ee) model with $G(z^{-1})=A^{-1}(z^{-1})B(z^{-1})$. The least square cost function $V(\underline{\theta},Z)$ is now convex in the parameters of $\underline{\theta}$. This means that the global minimum will be found. The disadvantage is that an ee model is concentrated on the estimation of the high frequencies. In the ultimate case the one step ahead prediction $\hat{y}(k+1)$ is simply the previous output y(k).
When we take $A(z^{-1})=1$, we obtain an output error (oe) model with $G(z^{-1})=F^{-1}(z^{-1})B(z^{-1})$. The least square cost function $V(\underline{\theta},Z)$ is now nonlinear in the parameters of $\underline{\theta}$. Nothing guarantees that the global minimum is found. But this model has a large horizon, and therefore necessary for controller design.

The initial output error model is now obtained in four steps:

1). Estimate a high n_h order ee model $\hat{G}_N^{n_h}(\hat{\underline{\theta}}_h, e^{j\omega})$ based on a large data set N.

2). Filter u with $A_h(z^{-1})$ yielding u^*, i.e. $u^* = A_h(z^{-1})u$

3). Simulate new data y^* from u^* and $\hat{G}_N^{n_h}(\hat{\underline{\theta}}_h, e^{j\omega})$, i.e. $y^* = \hat{G}_N^{n_h}(\hat{\underline{\theta}}_h, e^{j\omega})u$

4). Estimate a low n_l order oe model $\hat{G}_N^{n_l}(\hat{\underline{\theta}}_l, e^{j\omega})$ on the new data set.

Zhu [Zhu91] showed that that if input u and disturbance d are uncorrelated stationary mean zero white noise sequences, there holds that, for step 4 [Rei93]:

$$V(\hat{\underline{\theta}}, Z_N) = \frac{1}{2}\pi \int_{-\pi}^{\pi} |G_N^{n_h}(e^{j\omega}) - G_N^{n_l}(e^{j\omega})|^2 \frac{1}{var\left(G_N^{n_h}(e^{j\omega})\right)} d\omega \qquad (4.11)$$

We can interpret this as a weighted cost function. At frequencies where the variance in the high order model $G_N^{n_h}$ is high, the weighting is small and the low order model $G_N^{n_l}$ may deviate from $G_N^{n_h}$ because $G_N^{n_h}$ is unreliable. At frequencies where $G_N^{n_h}$ is reliable (small variance) the weighting is large and $G_N^{n_l}$ is kept close to $G_N^{n_h}$. $G_N^{n_l}(\hat{\underline{\theta}})$ can be used as initial model for estimation of an output error model on the real data set.

Essentially we first estimated a high order ee model, which is easy because it is convex in the parameters. With this LTI model we simulated new data without noise and of finite order. Using this new data we estimated the transfer of the ee model with a output error model, weighted with the reliability. Because this time no noise or disturbances are involved, the oe model can be found easy. This oe model is used as an initial estimate on the real data, using nonlinear optimization techniques to improve the model, obtaining the model for controller design. Because the initial estimate was already in the neigbourhood of the global minimum of $V(\underline{\theta},Z)$, the resulting model after optimization is hopefully in the global minimum of $V(\underline{\theta},Z)$.

4.2. Experimental conditions.

In general terms the experimental condition is a description of how the identification is carried out. This includes the choice of the sampling interval, input signal and pre-filtering of the data prior to estimation of the parameters [Lju87].

4.2.1. Sample frequency.

With the initial feedback we aligned the length of the impulse responses of the load in horizontal and vertical direction so that they both are about 10 seconds. The sample frequency must be high enough to guarantee no aliasing and according to Shannon's theorem we take $f_s \geq 2 \cdot f_N = 6$ Hz where $f_N = 3$ Hz (see figure 3.5) is the Nyquist frequency, the highest frequency in the system. But there is also an upper limit on f_s. If f_s is very high the measured signals change very slightly between two samples. This will cause numerical problems and all poles will cluster around the point one [Lju87]. Further the model fit may be concentrated to the high frequency band. We choose $f_s = 10$ Hz.

4.2.2. Input signals.

The most appropriate choice for identification of linear systems is a white noise input signal, because white noise has a flat frequency spectrum and therefore is sufficiently rich. A zero mean uniformly distributed white noise input signal stays between two values (zero mean):

$$u(k)_{white\ noise} \in [-a, a]$$

and each value occurs with equal probability.

For nonlinear systems we do not know if the white noise input signal is a good choice for identification. But if there is no a-priori knowledge, this seems to be the best choice.
Another commonly used input signal for identification is a Pseudo Random Binary Noise Squence (PRBNS) [Bac92b]. The frequency spectrum of this signal is essentially a sinc function, but the lowest frequency band is flat. And if the frequency band of the process is in this band we can compare the signal with a white noise, which is for linear systems sufficiently rich. This signal can jump between two possible values (zero mean):

$u(k)_{PRBS} \in [-a; a]$

and each value occurs with equal probability.
In general we are dealing with systems which behave only linearly around a working point. This causes troubles as we can see in the following examples:

1). $y(k) = u(k) * u(k-1) * \ldots * u(k-n)$
A PRBNS input signal will keep the output on either two values $y(k) \in [-a^{n+1}; a^{n+1}]$, and identification is impossible when we did not choose a very specific modelset. With a uniformly distributed white noise as input the output is in the range $y(k) \in [(-a)^{n+1}, a^{n+1}]$, and identification seems possible. But when n is large the output y(k) will be very small because u(k) will probably be close to zero at one time instant of the last n samples and hence y(k) will be close to zero too.

2). $y(k) = u(k)^n$
A PRBNS input signal will keep the output on either two values $y(k) \in [(-a)^n; a^n]$, and identification is impossible. With a uniformly distributed white noise as input the output is in the range $y(k) \in [(-a)^n\ a^n]$, and identification seems possible.

3). $y(k) = \sin(u(k))$
A PRBNS input signal will keep the output on either two values $y(k) \in [\sin(-a); \sin(a)]$, and identification is impossible. With a uniformly distributed white noise as input the output is in the range $y(k) \in [\sin(-a), \sin(a)]$ and identification seems possible.

For these examples the PRBNS input signal is not rich enough. Identification based on that data can lead to a limited model. The model will be perfectly able to simulate the response on a PRBNS input signal with the same amplitude but no other input signal. All the examples are static nonlinear functions. For dynamical nonlinear systems this effect is compensated due to the memory in the system when the frequency of the PRBNS signal is high enough. Although the external input takes only two values, the internal states still vary over a range.
For identification of the gantry crane a white input signal is used:

x_{ref} = uniformly distributed white noise $\in [-3, 3]$, mean = 0, $\text{std}(x_{ref})=2.2511$
y_{ref} = uniformly distributed white noise $\in [-4, 6]$, mean = 1, $\text{std}(y_{ref})=2.8944$

The amplitude (and so the mean power) of the input signal is very important, because a too powerful input signal can drive the states of the crane into saturation and a linear model is hard to fit. We also want to identify the crane in its working area. Therefore we choose an input signal so that the outputs x_l and y_l vary in the range from -0.9 to 0.9 metre. Because this is the maximum output possible, the signal to noise ratio on the output is also as good as possible.

An important rule of thumb [Bac92b] is that the experiment for model estimation last 5 to 10 times the largest time constant of the system. An indication of the largest time constant is determined with the pole of the linearized system in section 3.2. $p(z)_{max} = 6.6592$. $\text{Re}\{p(z)_{max}\} = e^{-T_s/\tau_{max}} = 6.6592$ hence $\tau_{max} \approx 19$ seconds. This means that the identification experiments must last between 100 and 200 seconds (1000 and 2000 samples with $T_s = 0.1$).

In figure 4.1 the input reference signals x_{ref} and y_{ref} in time and frequency domain are given.

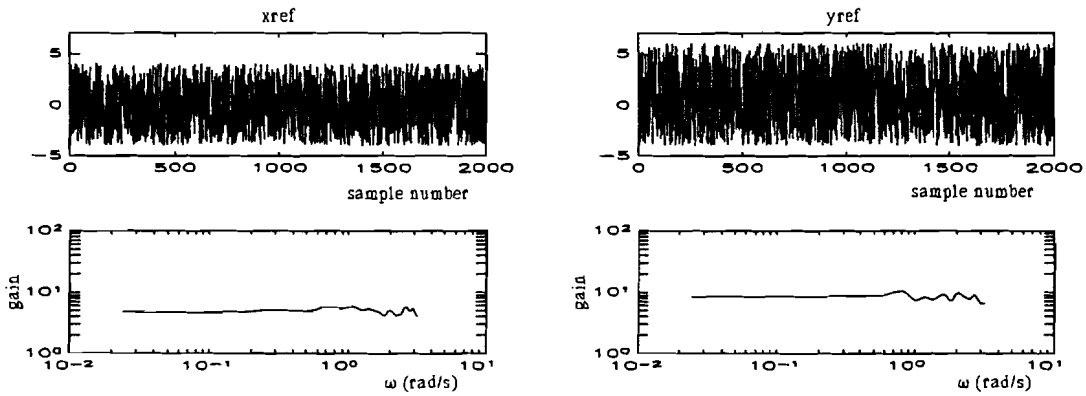


figure 4.1 Reference signals used for identification.

4.2.3. Order of the estimation.

Because the nonlinear model has six states it is trivial to estimate a sixth order model. But the decoupling in the linearized model would suggest to estimate a 4th order model for x_l and 2nd order for y_l. This a-priori knowledge is used during the identification and several Minimal Polynomial (MP) models are estimated. First, we estimated a 4th order SISO oe model for the transfer from x_{ref} to x_l and a 2nd order SISO oe model for the transfer from y_{ref} to y_l, on data of the deterministic nonlinear simulation model given in figure 3.3. The estimated SISO model structure is:

$$x_l = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + b_3 z^{-3} + b_4 z^{-4}}{1 + f_1 z^{-1} + f_2 z^{-2} + f_3 z^{-3} + f_4 z^{-4}} x_{ref} \qquad y_l = \frac{\bar{b}_0 + \bar{b}_1 z^{-1} + \bar{b}_2 z^{-2}}{1 + \bar{f}_1 z^{-1} + \bar{f}_2 z^{-2}} y_{ref} \tag{4.12}$$

These models can be brought in one state space model of the same structure as (3.14) by:

$$A = \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} \quad B = \begin{bmatrix} B_1 & 0 \\ 0 & B_2 \end{bmatrix} \quad C = \begin{bmatrix} C_1 & 0 \\ 0 & C_2 \end{bmatrix} \tag{4.13}$$

where $\{A_1,B_1,C_1\}$ forms the state space model of the fourth order SISO and $\{A_2,B_2,C_2\}$ forms the second order SISO state space model. Then we estimate a MIMO model with structure:

$$\begin{bmatrix} x_l \\ y_l \end{bmatrix} = \frac{1}{1 + f_1 z^{-1} + \ldots + f_6 z^{-6}} \begin{bmatrix} b_{11,0} + b_{11,1} z^{-1} + \ldots + b_{11,4} z^{-4} & b_{12,0} + b_{12,1} z^{-1} + \ldots + b_{12,6} z^{-6} \\ b_{21,0} + b_{21,1} z^{-1} + \ldots + b_{21,6} z^{-6} & b_{22,0} + b_{22,1} z^{-1} + B_{22,2} z^{-2} \end{bmatrix} \begin{bmatrix} x_{ref} \\ y_{ref} \end{bmatrix} \tag{4.14}$$

on the deterministic and stochastic simulation model in figure 3.3 and 3.7 respectively.

4.2.4. Primary signal processing.

First the data set is splitted in an estimation (sample 1:1500) and validation set (sample 1500:2000). The estimation set is pre-processed to obtain a better model [Bac92a]. First offset correction and then scaling correction is applied. The numerical values of the inputs and outputs (all the same dimension for the gantry crane) may differ significantly. The signals with the largest numerical values will automatically get highest weight in the quadratic criterion. This problem can be overcome by correcting the signals for offset and by scaling them afterwards. The next step is the correction of the delay times. This is done by shifting input and output signals relative to each other. A process with p inputs and q outputs has maximal p*q delay times, while we can maximal compensate p+q-1 (one for the reference). Therefore the compensation has to be such that the total summed delay left is minimal.

When real data is used it may be necessary to detrend the measured signals and correct them for peaks (peak shaving) prior to the manipulations above. After estimation the model is corrected in the reverse order and in the opposite direction. The resulting performance of model is measured during validation on the validation set.

4.2.5. Performance of the model.

After identification we need to assure that the estimated model of the system is performing well. As performance index we use the L_2-norm of the error divided by the L_2-norm of the output:

$$\gamma_x = \frac{\sqrt{\sum_{k=1}^{N} |e_x(k)|^2}}{\sqrt{\sum_{k=1}^{N} |\tilde{x}_l(k)|^2}} \qquad \gamma_y = \frac{\sqrt{\sum_{k=1}^{N} |e_y(k)|^2}}{\sqrt{\sum_{k=1}^{N} |\tilde{y}_l(k)|^2}} \qquad \gamma = \sqrt{\gamma_x^2 + \gamma_y^2} \tag{4.15}$$

where $e_x = x_l - \hat{x}_l$ and x_l the real x-position of the load and $\hat{x}_l$ the estimated x-position of the load and $\tilde{x}_l$ the offset corrected real x-position of the load. The same yields for the y-position of the load. Of course there is a lower bound on this performance because we estimate a LTI model of finite order and based on N observed input/output data samples perturbated with white noise.
Another criterion is the whiteness of the resulting error. If the disturbance acting on the system is white and the system is in the model set then the error should resemble these disturbances and therefore be white too. In our case however the disturbances are also caused by the nonlinearity of the crane and hence are not white at all. Therefore this criterion is not used.

4.3. Identification results.

First we used the deterministic model in figure 3.3 to get input/output data, and estimated a fourth order SISO model for the x_l and a second order SISO model for y_l as defined in (4.12) connecting them to a sixth order state space model as defined in (4.13). Then the MIMO model with structure defined in (4.14) was estimated on the same data set. The performance is given in the second and third column in tabel 4.1.

tabel 4.1. Performance of the estimated model over the validation set.

norm	SISO determ.	MIMO determ.	MIMO stoch.
γ_x	0.3998	0.3333	0.3349
γ_y	0.0697	0.0685	0.0770
γ	0.4058	0.3403	0.3437

The deterministic model is given in figure 3.3 and the stochastic model in figure 3.7.

The MIMO model overall performance is circa 15% better, due to the fact it has six states of freedom and it can estimate cross transfers (especially in the x-position of the load). Hence the SISO model is rejected. But in fact only the SISO model with the same structure can be compared with the linearized model in (3.14). A transformation from the continuous time to the discrete time of the linearized model is then necessary. This is done with a first order hold filter and a sample frequency of f_s = 10 Hz resulting in a discrete model with the following zeros and poles in the complex z-plane:

zeros:	-9.2631	poles:	0.8963 + 0.3544i
	-0.0961		0.8963 - 0.3544i
	-0.9502		0.9434 + 0.0514i
	-0.7931		0.9434 - 0.0514i
			0.5138
			0.9688

Compare this with the poles and zeros of the 2 x SISO estimated model:

zeros:	3.9218	poles:	0.8024 + 0.3396i
	1.8887		0.8024 - 0.3396i
	1.0191 + 0.4988i		0.9224
	1.0191 - 0.4988i		0.7852
	0		0.6580
	0		0.9679

Note that these time discrete systems are non minimum phase, because of the zeros out of the unit circle. Remember that the first four poles are due to x_t and θ and the last two poles are due to L. The complex pole pair close to 1 in the linearized system is caused by the badly damped pendulum frequency. Note that this pole pair is not estimated in the 2 x SISO model. The difference between the linearized model and the 2 x SISO estimated model is not due to noise, because the data was noisefree. The linearization point and the nonlinearity of the simulation model are causes of the difference.

Column four in tabel 4.1 shows the resulting model performance of the estimated MIMO model of the structure given in (4.14) estimated on the input/output data generated with the stochastic model of figure 3.7. The difference in performance with the deterministic model is very small which means that the most model errors are due to the nonlinear behaviour of the crane. Figure 4.2 shows the resulting model validation. The maximum value of the error $\|e\|_\infty = \sup_{k \in N} |y(k) - \hat{y}(k)| \approx$ 0.4 m, is relatively large. This is due to the fact that the pendulum frequency is not estimated. In the lower left plot, the error in the x-position of the load, we detect the pendulum frequency f_p = 0.5 Hz. This means that the model is uncertain around the frequencies $f_p \approx 0.5$ Hz.

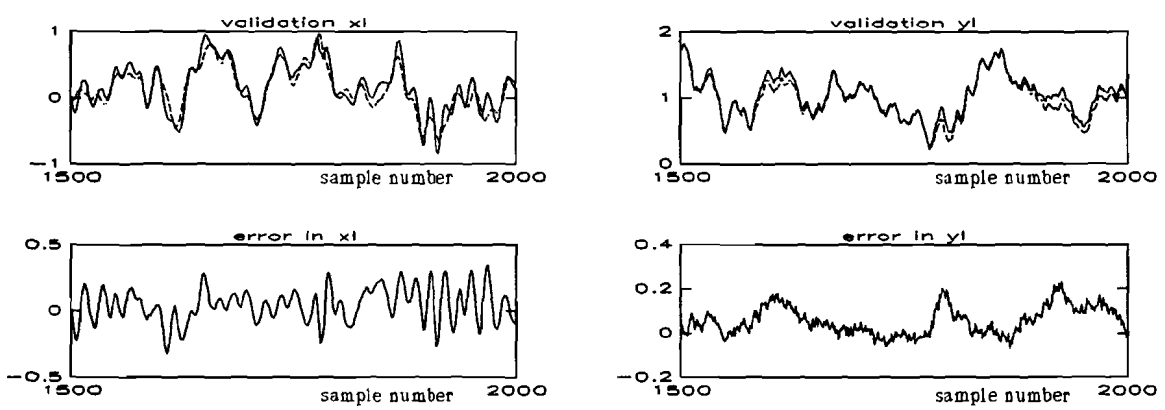


figure 4.2 Output signals MIMO model estimation on noisy data.

where:

— = real data

--- = estimated model

The estimated MIMO models on the data generated with the deterministic and stochastic simulation model in figure 3.3 and 3.7 repectively, with the performance given in column 3 and 4 in table 4.1 had the following minimum polynomial poles:

MIMO determ.	MIMO stochastic
0.8115 + 0.2974i	0.8222 + 0.2893i
0.8115 - 0.2974i	0.8222 - 0.2893i
0.9813	0.9792
0.9421	0.9546
0.2492	0.1699
0.8434	0.8113

Note that the complex pole pair due to the pendulum is still not estimated. The resulting third and fourth pole have only real parts.

Figure 4.3 shows the validation of an impulse response. The pendulum frequency $f_p \approx 0.5$ Hz is not estimated, which is clearly shown in the response of x_l on an impulse on x_{ref}.
As mentioned earlier, the effect of the cross transfer is very small for an impulse on a reference (even zero for an impulse on y_{ref}). But for a uniformly distributed white noise, the coupling is greater. The identified model is estimated on the response of a uniformly distributed white noise sequence, and therefore there exist a cross transfer. But this estimated cross transfer gives poor results for an impulse response, as shown in the lower left and upper right plot in figure 4.3. This is probably due the the nonlinear behaviour of the coupling.

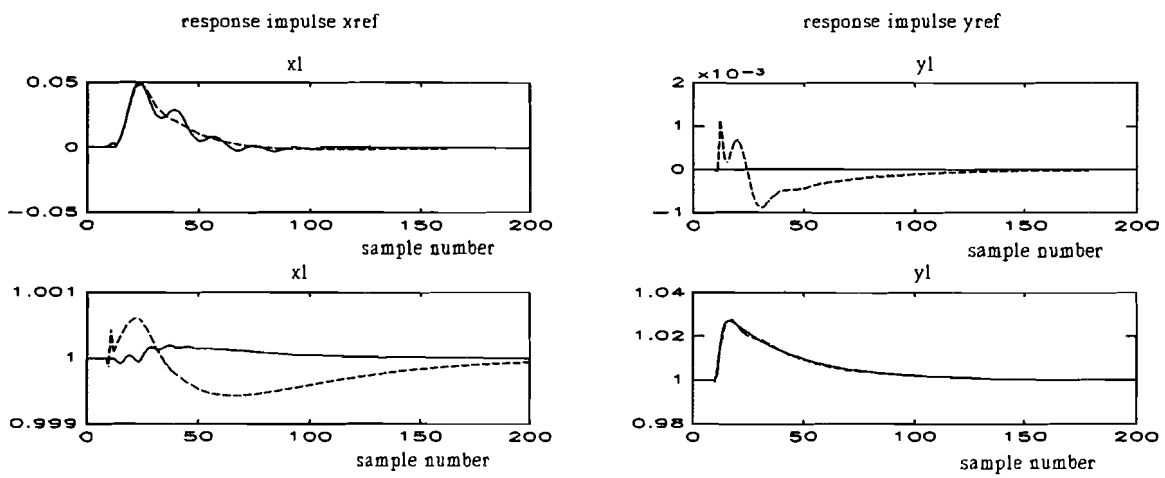
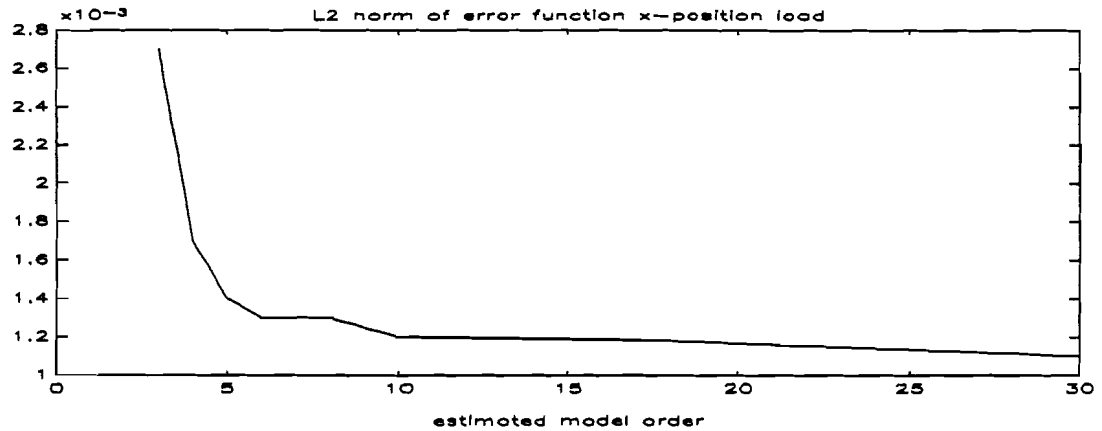


figure 4.3 Validation impulse noiseless data.

Note that also for the white noise te estimated cross transfer is almost two decades smaller than the transfer from x_{ref} to x_l and from y_{ref} to y_l. This means that the coupling is still very small.

Using oe models of higher order had no spectacular effects on the performance of the model (see figure 4.5). The pendulum frequency is estimated only by oe models of order 15 and higher. This means that for low order oe models the performance is better when the model does not include the dynamics of the pendulum.



figuur 4.5 Error versus model order.

Looking at the loss function the choice of a sixth order model is (as expected) a proper choice.

Additional remark:

The identification method proposed by Backx [Bac89] for obtaining a MPSSM model is also used for identification of the gantry crane. However for a "loose" controller the impulse response is about 100 samples. The initial estimate is based on a Markov Parameter model, which is in fact the impulse response. This means that we need at least 400 parameters (2 inputs x 2 outputs x 100 samples) for obtaining an initial estimate. This led to memory problems during program execution.

5. Controller design.

The design of the second loop static state controller for the gantry crane is based upon the identified model of the measured states, a-priori knowledge of h(x) and a given set of design specifications which include e.g. actuator saturation. Using a simulation method [Boy91], the second loop controller is calculated. The two-loop scheme is used iteratively on the nonlinear simulation model of the gantry crane. Results of the performance of the designed controller acting on the deterministic nonlinear simulation model of the gantry crane are given, given in figure 3.3.

5.1. Second loop static state controller design.

Given the basic plant including shaping filters and the controller arranged as in figure 5.1.

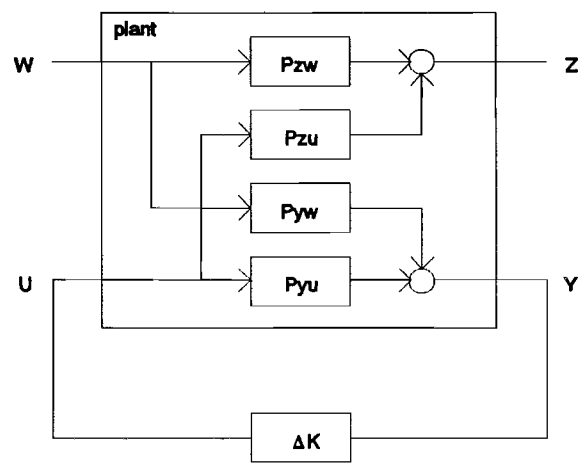


figure 5.1 Basic plant.

We can distinguish:

input U = the actuator signal vector consisting of inputs which can be manipulated by the controller, e.g. forces, torques, flows, etc. It is exactly the output signal of the controller.

input W = the exogenous input vector consisting of all other inputs, e.g. reference signals, sensor noise, actuator noise, disturbances etc.

output Y = the sensor signal vector consisting of output signals that are accessible to the controller, e.g. the measured position, measured temperature, etc. It is exactly the input signal of the controller.

output Z = the regulated output signal vector consisting of all plant outputs to be minimized, e.g. tracking error, temperature diviation, etc., and all constrained signals, e.g. input voltage, valve position, etc.

When Pzw, Pzu, Pyw, Pyu and K are all estimated LTI transfer functions we can write:

$$\begin{bmatrix} Z \\ Y \end{bmatrix} = \begin{bmatrix} P_{zw} & P_{zu} \\ P_{yw} & P_{yu} \end{bmatrix} \cdot \begin{bmatrix} W \\ U \end{bmatrix} = P \cdot \begin{bmatrix} W \\ U \end{bmatrix} \tag{5.1}$$

$$U = \Delta K \cdot Y$$

The closed loop transfer from exogenous inputs W to the regulated outputs Z is then given by:

$$H_{zw} = P_{zw} + P_{zu}\Delta K(I - P_{yu}\Delta K)^{-1}P_{yw} \tag{5.2}$$

For the gantry crane we assume that:

1). h(x) is known and equal to:

$$h(x) = \begin{cases} x_t + L\sin(\theta) \\ L\cos(\theta) \end{cases} \tag{5.3}$$

and with a linearization around $[x_t\ \theta\ L]^T = [0\ 0\ 1]^T$ (see section 3.2) we get:

$$\hat{H} = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \tag{5.4}$$

In steady state yields when $\theta = 0$, see (3.25):

$$y_e = \begin{bmatrix} x_t \\ L \end{bmatrix} \tag{5.5}$$

Instead of x_t we feed back $e_x = x_t - y_{ref}$ and instead of L we feedback $e_y = y_t - y_{ref}$, see figure 5.2. Now the controller K generates a input signal u based on the current states and the current tracking error, so that in steady state the output y equals the reference r.

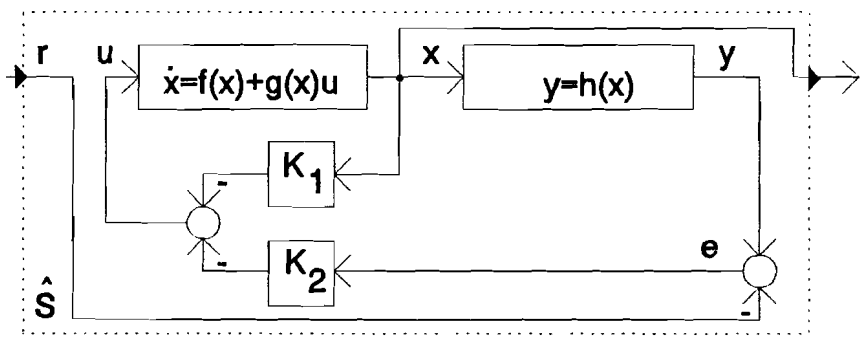


figure 5.2 Tracking control with first loop static state feedback.

where K_1 and K_2 are defined by:

$$\begin{aligned} K_1 &= KM_1 \\ K_2 &= KM_2 \end{aligned} \tag{5.6}$$

and where M_1 and M_2 are given by:

$$M_1 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad M_2 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \tag{5.7}$$

Remember from section 2.4 that in fact K_1 is a static state feedback controller and K_2 is a static output feedback controller.

2). the states x_t, θ and L are measured. Now we estimate the system $\hat{S}$ between reference r and the states x_t, θ and L. The other three states are then calculated as the derivatives of the estimated states. This is possible because we knów that the other states are just the derivatives. The sample frequency must be high enough so that the estimated $\dot{x}_t$, $\dot{\theta}$, $\dot{L}$ agree with the real states. The advantage above estimating all the states is that during estimation of the transfer from the reference

r to the states x, the global minimum is much easier to obtain. Only half the number of parameters have to be estimated which provides that the cost has less local minima. The resulting estimated system $\hat{S}$ describes the transfer from reference r to all the states of x, see figure 5.2.

When we want to add the second loop controllers ΔK_1 and ΔK_2, there arises a problem because the addition point (output of the first loop controller, denoted with ap in figure 5.3) is intern in the estimated system $\hat{S}$. The only possibility left, is to add the output of the second loop controller $\hat{u}$ to the input of the system, see figure 5.3. But without any precaution this would not lead to a parallel construction of K and ΔK. Therefore we multiply the output of the second loop controller $\hat{u}$ with K_2^{-1}. Adding the resulting signal u^* to the input of the system $\hat{S}$, we virtually add the output of the second loop controller $\hat{u}$ to the output of the first loop controller u as shown in figure 5.3. This is only possible because K_2 is a static square invertible matrix.

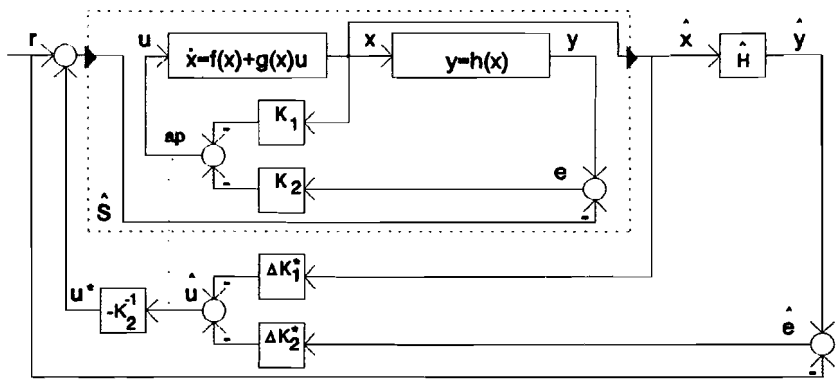


figure 5.3 Twoloop static state feedback tracking control.

Note the parallel structure of the scheme in figure 5.3.
For ΔK_1^* and ΔK_2^* yields:

$$\Delta K_1^* = \Delta K * M_1$$
$$\Delta K_2^* = \Delta K * M_2 \qquad (5.8)$$

When we bring this scheme in the general structure of figure 5.1 we obtain the scheme of figure 5.4.

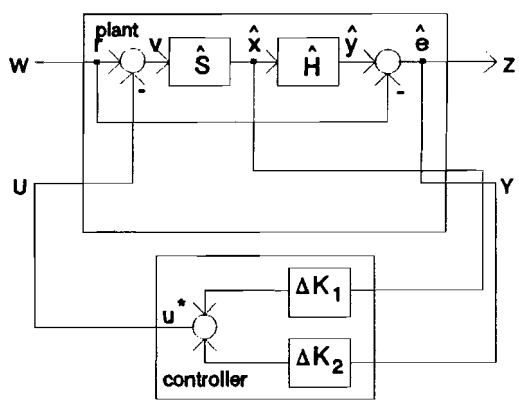


figure 5.4 Twoloop structure in basic plant.

where the second loop controller transfer from Y to U is given by:

$$\begin{aligned} \Delta K_1 &= -K_2^{-1}\Delta K_1^* = -(KM_2)^{-1}\Delta KM_1 \\ \Delta K_2 &= -K_2^{-1}\Delta K_2^* = -(KM_2)^{-1}\Delta KM_2 \\ u^* &= \Delta K_1 x + \Delta K_2 e \end{aligned} \tag{5.9}$$

Now yields:

$$\begin{bmatrix} e \\ x \\ e \end{bmatrix} = \begin{bmatrix} \hat{H}\hat{S}-I & -\hat{H}\hat{S} \\ \hat{S} & -\hat{S} \\ \hat{H}\hat{S}-I & -\hat{H}\hat{S} \end{bmatrix} \cdot \begin{bmatrix} r \\ u \end{bmatrix} \tag{5.10}$$

and for the closed loop transfer H_{zw} from the reference r to the error e, which is in fact the sensitivity S, yields:

$$H_{zw} = (\hat{H}\hat{S} - I + \hat{S}\Delta K_1)(I + \hat{S}\Delta K_1 + \hat{H}\hat{S}\Delta K_2)^{-1} \tag{5.11}$$

Because ΔK_1 is different from ΔK_2, the H_∞-norm of the closed loop transfer $\| H_{zw} \|_\infty$ is uniform descending in the parameters of ΔK, where $\| . \|_\infty$ is defined as the maximum singular value which is equal to the maximum gain in the SISO case. This can easily be seen when all transfers and matrices are SISO. Then the optimal controller is reached when $\Delta K_1 \rightarrow \infty$, $\Delta K_2 \rightarrow \infty$ and $\Delta K_2/\Delta K_1 \rightarrow \infty$, making the sensitivity S (the transfer from r to e) infinitely small. This implies that for $\Delta K_1 \rightarrow \infty$, $\Delta K_2 \rightarrow \infty$ and $\Delta K_2/\Delta K_1 \rightarrow \infty$, we have perfect tracking or deadbeat control. The error in that case would be zero. But this will result in an infinitely large input power demand and no robustness as we will see in the next section. We expect that this also yields for MIMO systems.

5.2. Robust stability and robust performance.

The ultimate objective of control system design is that the controller works "well" on the real plant. For example the controller must not make the system unstable if the dynamics of the estimated model are not exactly equal to those of the real system. To simplify the scheme we assume that $\hat{H}$ is exactly equal to h(x). Now we can describe the real system with a nominal estimated model and a perturbation of it, see the scheme in figure 5.7.

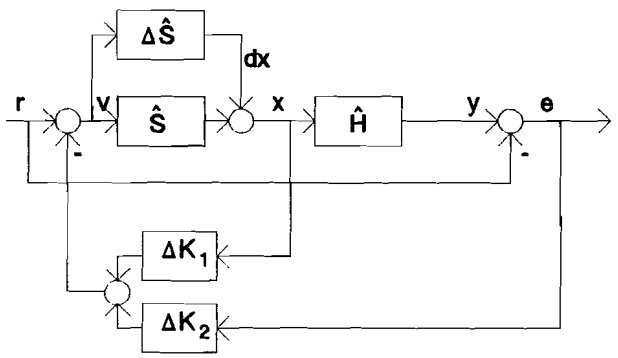


figure 5.7 Esimated model of the system with uncertainty ΔS.

Without giving a detailed description of robust controller design, assume the system given in figure 5.7 and let $\hat{S}$ and $\Delta\hat{S}$ be stable.
The sensitivity S (transfer from reference r to tracking error e, see figure 5.7) is given by:

$$S = (\hat{H}\hat{S} - I + \hat{S}\Delta K_1)(I + \hat{S}\Delta K_1 + \hat{H}\hat{S}\Delta K_2)^{-1} \tag{5.12}$$

and has to be small for good signal tracking. If we want a good steady state tracking, the sensitivity S has to be small especially in the lower frequencies. Therefore we define the weighted sensitivity W_1S, where W_1 is a low pass filter, putting penalty on the low frequencies of S.

The control sensitivity R (transfer from dx to estimated system input v, see figure 5.7) is given by:

$$R = -(\Delta K_1 + \Delta K_2\hat{H})(I + \hat{S}\Delta K_1 + \hat{H}\hat{S}\Delta K_2)^{-1} \tag{5.13}$$

Define the family of all additive perturbed systems Σ:

$$\Sigma: \{\hat{S}_\Delta = \hat{S} + \Delta\hat{S} \mid |\Delta\hat{S}| \leq |W_2| \ \forall\ \omega\} \tag{5.14}$$

Now the system is robustly stable iff $\hat{S}$ is stabilized by ΔK, and $\| R\Delta S \|_\infty < 1$ known as the baby version small gain theorem, where $\| . \|_\infty$ is defined as the maximum singular value, R the control sensitivity and $\Delta\hat{S}$ the perturbation on the model of the system $\hat{S}$. From the triangle inequality follows that $\| R \|_\infty \cdot \| \Delta\hat{S} \|_\infty < 1$ and if $| \Delta\hat{S} | < | W_2 |\ \forall\ \omega$, robust stability (guaranteed stability for all plants in Σ) is guaranteed as $\| W_2^{-1}R \|_\infty < 1$.

When all transfers are SISO, then it follows that when $\Delta K_1 \to \infty$, $\Delta K_2 \to \infty$ and $\Delta K_2/\Delta K_1 \to \infty$, R tends to $\hat{S}^{-1}$. Since $\hat{S}$ is chosen proper, representing a physical system $\hat{S}^{-1}(j\omega) \to \infty$ as $\omega \to \infty$, it follows that $\| W_2^{-1}R \|_\infty > 1$, and hence the controller is not robust. Taking smaller ΔK_1 and ΔK_2 (just scalars in the SISO case), and increasing robustness will result in a decrease in performance. This means that there always exist a tradeoff between performance and robustness.
A second limit for ΔK_1 and ΔK_2 is caused by the actuators (or other signals). When $\Delta K_1 \to \infty$ or $\Delta K_2 \to \infty$, the input signal of the system $v \to \infty$ (see figure 5.7), which also means that the input power of the actuators for the process u must tend to infinity, and that is not allowed. We expect that the previous also yields for MIMO systems.

Now the regulator problem is to design the controller which minimizes $\| W_1S \|_\infty$ under the given constraints:
1). The controller must be robust and hence stabilizing all the systems in Σ and therefore stabilize the nominal plant and $\| W_2^{-1}R \| < 1$.
2). Keeps actuators (or other signals) out of saturation.

The implementation in MATLAB is based upon the function constr.m in the optimization library. This function can solve inequality constrained optimization problems, and uses a Sequential Quadratic Programming (SQP) method (see users guide of the optimization toolbox). The solution is found iteratively according to (4.6). The search direction at the i^{th} iteration: f^i, is found numerically using the information of previous iterations. In each iteration the optimization criterion J and the constraints are evaluated, which implies a high computational effort. The advantage of the design in time domain is that we can put restriction on time signals. This in opposite to H_∞-design, where it is not possible to design a controller based on constraints in time domain.
When the optimization criterion J, the constraints and the change in the parameters of the controller ΔK are smaller than a predefined tolerance the optimization stops and the second loop controller is designed. The new first loop controller is given by $K + \Delta K$.

For the gantry crane we implemented the scheme of figure 5.8.

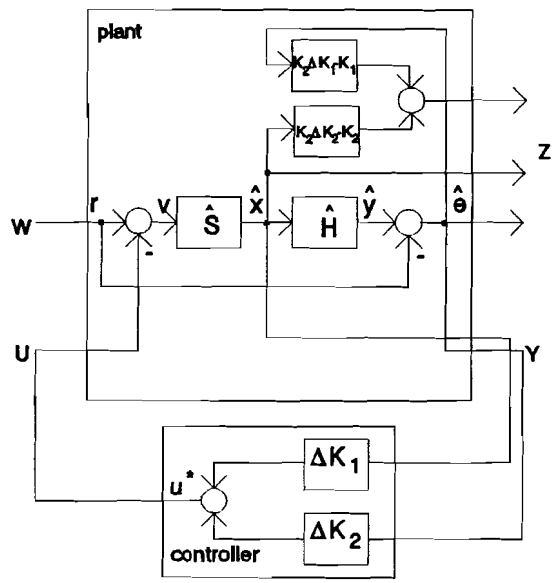


figure 5.8 Implementation of the gantry crane.

where:

$$W = \begin{bmatrix} x_{ref} \\ y_{ref} \end{bmatrix} \quad U = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \quad Z = \begin{bmatrix} e_x \\ e_y \\ x_t \\ \theta \\ L \\ \hat{F}_t \\ \hat{F}_h \end{bmatrix} \quad Y = \begin{bmatrix} x_t \\ \dot{x}_t \\ \theta \\ \dot{\theta} \\ L \\ \dot{L} \\ e_x \\ e_y \end{bmatrix} \tag{5.15}$$

$u = [F_t\ F_h]^T$ is not explicit known because they are internal signals in the estimated model of the system $\hat{S}$. But we can easily recover those signals. If we look at the scheme in figure 5.3 then yields:

$$u = -K_1 x - K_2 e - \Delta K_1^* \hat{x} - \Delta K_2^* \hat{e} \tag{5.16}$$

This is used in the scheme in figure 5.8, where:

$$\begin{aligned} \hat{u} &= -K_1 \hat{x} - K_2 \hat{e} - \Delta K_1^* \hat{x} - \Delta K_2^* \hat{e} \\ &= -K_1 \hat{x} - K_2 \hat{e} + K_2 \Delta K_1 \hat{x} + K_2 \Delta K_2 \hat{e} \\ &= (K_2 \Delta K_1 - K_1)\hat{x} + (K_2 \Delta K_2 - K_2)\hat{e} \\ &= \begin{bmatrix} \hat{F}_t \\ \hat{F}_h \end{bmatrix} \end{aligned} \tag{5.17}$$

Now it is possible to calculate the closed loop transfer with the estimated model and the a-priori information about h(x). To obtain a better controller not $\hat{H}$ is used during controller design, but the real h(x). Further we restricted us to a controller with the structure of (3.24), with only six entries non zero. This results in only half the calculation time for controller design, and since the

cross transfer is only small (see figure 4.3), we hope that this constraint on the controller structure, has a neglectable influence on the performance of the controller.

We split Z in two parts:

1). The first 2 variables: $[e_x\ e_y]^T = e$.
The tracking error is minimized. The minimization criterion is given by:

$$J = \|W_1 e\|_2^2 \tag{5.18}$$

where W_1 a MIMO digital filter. By choosing the filter W_1 low pass, we put more penalty at the low frequencies than at the high frequencies in the tracking error and hence at the sensitivity S. The resulting controller will give a smaller tracking error for low frequencies in the reference track (and worse performance for the higher frequencies) than without weighting filter W_1. For W_1 we used for example a first order Butterworth filter with a cut-off frequency at 2, 3 or 4 Hz.

2). The last 5 variables: x_t, θ, L, F_t, F_h.
All these variables are constrained:

-1	$\leq$	x_t	$\leq$ 1	[m]
-0.5	$\leq$	θ	$\leq$ 0.5	[rad]
0.1	$\leq$	L	$\leq$ 2	[m]
-50	$\leq$	F_t	$\leq$ 50	[N]
-14,7	$\leq$	F_h	$\leq$ 50	[N]

The designed controller is to "strong" when these values are exceeded. F_h is bounded with a lower value of $m_L g = 14.7$ [N]. This results in a controller wich does not put a force in the positive y-direction. The other constraints are due to the construction of the gantry crane, explained in section 2.1. When we take the minimization criterion given in (5.18), we expect that for $|k_{ij}| \rightarrow \infty\ \forall_{ij}$, $J = \|W_1 e\|_\infty \rightarrow 0$, if W_1 is chosen proper. The solution for the optimization problem will be limited by the constraints. Taking the constraints convex, [Boy91], will probably provides a minimum number of local minima in the cost function.

Till yet, no explicit precaution is taken to obtain a robust controller, but in fact the constraints on x_t, θ and L make the system robust. They keep the signals in the state x small and hence the resulting transfer from d_x to the states x, which we call the complementary sensitivity T, which addresses multiplicative robustness, see Doyle [Doy90]. The constraints on F_t and F_h will keep the input of the system v, indirectly small and hence R will be small. That is the reason that in the approach of Belt [Bel93] only the performance is optimized together with bounds on the input. Robustness is implicit in the constraints. But this is not enough in the case of the nonlinear gantry crane, because the resulting controller, obtained with the previous described method, made the nonlinear system unstable. Therefore we put additional penalty on the complementary sensitivity T. This is done by weighting the state $\dot{\theta}$. The transfer from d_x to $\dot{\theta}$ is kept small which implies that the complementary sensitivity T has to be small, which is necessary for robust controller design, see again Doyle [Doy90]. It is better to use the constraint: $\|W_2^{-1}R\|_\infty < 1$, which is a convex constraint. This can be checked in the folowing way [Boy91]. Let $\|W_{2,1}^{-1}R_1\|_\infty < 1$ and $\|W_{2,2}^{-1}R_2\|_\infty < 1$ then $(\|W_{2,1}^{-1}R_1\|_\infty + \|W_{2,2}^{-1}R_2\|_\infty)/2 < 1$. The problem for obtaining W_2 is not discussed here. Therefore we need more information about the accuracy of the estimated system. In [Boo92] and [Zhu91] methods for identification for robust control are proposed.

To obtain a robust controller for the gantry crane we weighted $\dot{\theta}$ in the cost function J. In this way large values of $\dot{\theta}$ are penalized (and the control sensitivity is kept small). This means that

especially those input signal, which give rise to large outputs of $\dot{\theta}$, are penalized in the cost funtion, and the entries in the controller ΔK are kept small. This is wanted because we know that the "dynamics of the pendulum" (e.g. the pendulum frequency which was not estimated in section 4) are uncertain in the estimated model of the system. The resulting optimization criterion which is used for controller design in each iteration is:

$$J = \|e_x\|_2^2 + \|e_y\|_2^2 + \|\dot{\theta}\|_2^2 \tag{5.19}$$

5.3. Identification results for controller design.

In section 2.4 we summarized the various steps for controller design. The gantry crane is stabilized by the initial static state controller obtained in section 3.3. Now we are going to optimize that controller (with the same structure) for a predefined reference signal. Remember that we chose the same controller structure as in (3.24) because the cross transfer was very small). It is important that the reference signal is sufficiently rich. Firstly, the estimated model must represent the behaviour of the system which is only possible if the data contains enough information (input rich enough). Secondly, we desire a controller which performs well for a class of reference signals U_R. The chosen reference signal must be rich enough to represent U_R. The reference signal used, is given in figure 5.9.

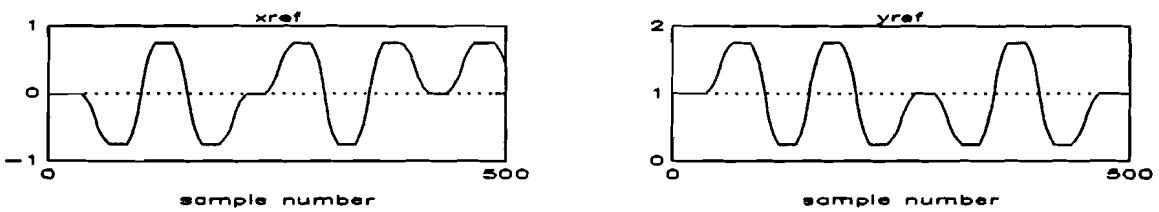


figure 5.9 Reference track for optimization.

To prevent actuator saturation we used a smooth reference track, which is obtained by filtering a random sequence consisting of three values [-0.75, 0, 0.75] with a FIR filter as given in (5.20), and hence the first and second derivative of the reference signal are continuous.

$$y(k) = \frac{1}{21} \sum_{l=-10}^{l=10} u(k+l) \tag{5.20}$$

The next step in the "two-loop procedure" is to identify the system $\hat{S}$, the transfer from reference r to the states x. The minimum polynomial structure used is:

$$\begin{bmatrix} x_t \\ \theta \\ L \end{bmatrix} = \frac{1}{1 + f_1 z^{-1} + \ldots + f_6 z^{-6}} \begin{bmatrix} b_{11,0} + b_{11,1}z^{-1} + \ldots + b_{11,4}z^{-4} & b_{12,0} + b_{12,1}z^{-1} + \ldots + b_{12,6}z^{-6} \\ b_{21,0} + b_{21,1}z^{-1} + \ldots + b_{21,6}z^{-6} & b_{22,0} + b_{22,1}z^{-1} + \ldots + b_{22,6}z^{-6} \\ b_{31,0} + b_{31,1}z^{-1} + \ldots + b_{31,6}z^{-6} & b_{32,0} + b_{32,1}z^{-1} + b_{32,2}z^{-2} \end{bmatrix} \begin{bmatrix} x_{ref} \\ y_{ref} \end{bmatrix} \tag{5.21}$$

The other three states $\dot{x}_t$, $\dot{\theta}$, $\dot{L}$ are calculated as the derivatives of the estimated states x_t, θ, and L. The data is processed before estimation in the same way as in section 4.2.

Figure 5.10 shows the validation of the estimated system on the output data of the states x. The sample frequency is f_s = 10 Hz.

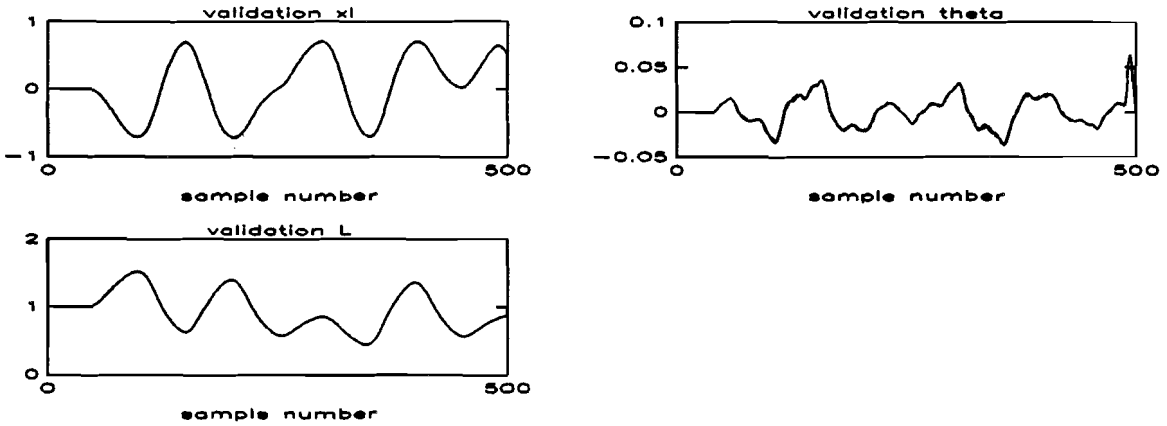


figure 5.10 Validation of the estimated model of the system.

where: — real output
--- estimated output

It is obvious that the system (the nonlinear model with initial loop controller given in (3.24)) behaves almost linear for the reference signal, because we can estimate a very accurate LTI model. The performance of the estimated models each iteration is given in table 5.1. K(0) is the initial controller given in (3.24), and K(1) to K(4) are the successive designed controllers.
The performance is measured by:

$$\gamma_x = \frac{\sqrt{\sum_{k=1}^{N} |e_{x_r}(k)|^2}}{\sqrt{\sum_{k=1}^{N} |\tilde{x}_r(k)|^2}} \qquad \gamma_\theta = \frac{\sqrt{\sum_{k=1}^{N} |e_\theta(k)|^2}}{\sqrt{\sum_{k=1}^{N} |\tilde{\theta}(k)|^2}} \qquad \gamma_L = \frac{\sqrt{\sum_{k=1}^{N} |e_L(k)|^2}}{\sqrt{\sum_{k=1}^{N} |\tilde{L}(k)|^2}} \qquad \gamma = \sqrt{\gamma_x^2 + \gamma_\theta^2 + \gamma_L^2} \qquad (5.22)$$

where $e_q = q - \hat{q}$, the difference between the real signal q and the estimated signal $\hat{q}$, and $\tilde{q}$ the real signal correted for offsets.

Table 5.1 Resulting performance of the estimated model each iteration.

controller	γ_x	γ_θ	γ_L	γ
K(0)	0.0078	0.0793	0.0027	0.0797
K(1)	0.0218	0.1649	0.0057	0.1665
K(2)	0.0216	0.1429	0.0054	0.1446
K(3)	0.0218	0.1178	0.0045	0.1199
K(4)	0.0222	0.0735	0.0042	0.0768

Note that the angle θ in the third column of table 5.1 is estimated worst, probably due to its nonlinear behaviour. For each iteration the estimated model has comparable accuracy, as indicated in the last column of table 5.1, which implies that the real system can be estimated by a LTI model each iteration. This is also found with the nonlinearity index α see table 5.2. The set of inputs signals U exist of only the input reference signal used for controller design. In each

iteration α is small enough to estimate a LTI model of the system. However, α is increasing each iteration, which means that the gantry crane does not belong to the class of systems which behave more linear under stronger feedback. This can be explained by the fact that when a stronger feedback is applied, the angle θ takes higher values and for example $\sin(\theta) \neq \theta$ and $\cos(\theta) \neq (1-\theta)$ anymore and the nonlinear model in (3.3) will behave more nonlinear.
To estimate a LTI model of the system however, θ is small enough, because the entries in table 5.1 (each γ) have all less accuracy.

table 5.2 Nonlinearity index each iteration, $U=\{r\}$.

controller	α
K(0)	0.4636e-4
K(1)	3.8361e-4
K(2)	8.8205e-4
K(3)	18.9930e-4
K(4)	40.0149e-4

5.4. Controller design results.
Figure 5.11 shows the outputs of the systems with the iteratively calculated controllers. Iteration 5 results in controller K(5) which is almost equal to K(4) and hence we may conclude that the obtained controller tends to the best possible controller under the given constraints. For the x-position of the load x_l we obtain an overshoot when the controller becomes "tighter". If this is undesired we must put additional constraints on the output of x_l during optimization.

For this reference signal x_t is the limiting factor, see the response of x_t in figure 5.11. The state x_t would even slightly exceed the maximum value of 1 metre when it was not clamped in the nonlinear model. This means that even if the maximum value is not exceeded for the estimated model during controller design, it is possible that on the real crane the maximum value is rather exceeded, due to unmodelled nonlinearities. The constraints on θ and L are not reached, see the response of θ and L in figure 5.11. Without constraints, and without taking some robustness into account, the optimization will result in a deadbeat controller for the estimated model. Of course this controller will fail (making the system unstable) when the real plant is only slightly different from the estimated model. So performance and robustness are mutually exchangeable.
Also the forces F_t and F_h are not limiting the optimization procedure, see the response of F_t and F_h in figure 5.11, and possibly a less smooth input reference signal can be used for optimization.

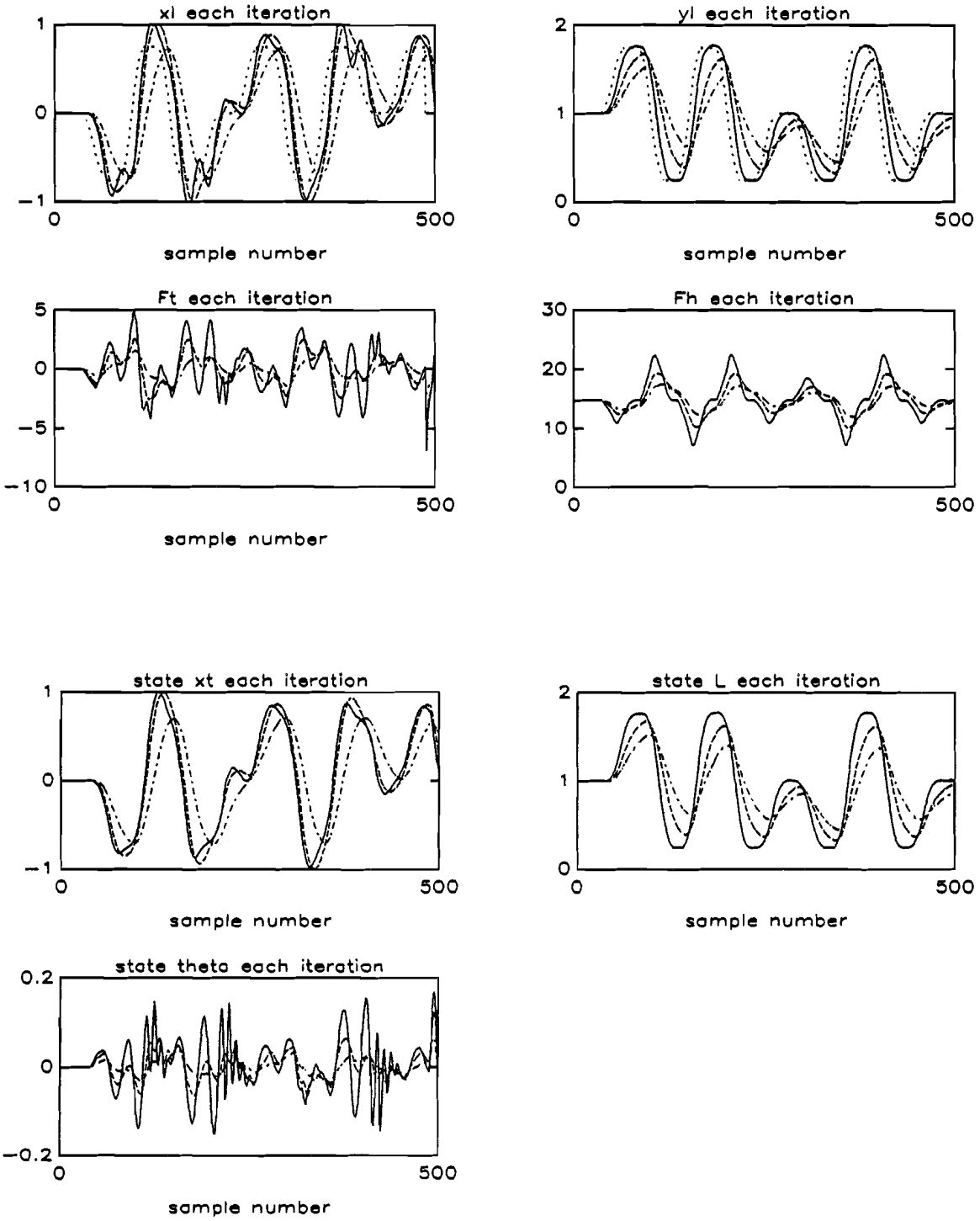


figure 5.11 Resulting system output each iteration.

where:

:	reference signal	--	iteration 2 = controller K(2)
.-	initial controller K(0)	—	iteration 4 = controller K(4)

The performance of the controller obtained in each iteration is given in table 5.3. The performance is measured by:

$$\gamma_x = \frac{\sqrt{\sum_{k=1}^{N} |e_x(k)|^2}}{\sqrt{\sum_{k=1}^{N} |\tilde{x}_l(k)|^2}} \qquad \gamma_y = \frac{\sqrt{\sum_{k=1}^{N} |e_y(k)|^2}}{\sqrt{\sum_{k=1}^{N} |\tilde{y}_l(k)|^2}} \qquad \gamma = \sqrt{\gamma_x^2 + \gamma_y^2} \tag{5.23}$$

where $e_x = x_{ref} - x_l$ and $e_y = y_{ref} - y_l$, equal to the tracking error and $\tilde{x}_l$, $\tilde{y}_l$ the x- and y-position of the load corrected for offset.

Table 5.3 Resulting performance of the controllers each iteration.

controller	γ_x	γ_y	γ
K_0	0.9777	0.8937	1.3246
K_1	0.7752	0.7783	1.0985
K_2	0.6609	0.7316	0.9859
K_3	0.5969	0.6127	0.8554
K_4	0.5549	0.4045	0.6872

Note that each iteration the controller performs better.
The resulting controller is given by:

$$K(5) = \begin{bmatrix} 10.5705 & 5.6694 & -7.2582 & -2.0235 & 0 & 0 \\ 0 & 0 & 0 & 0 & -10.5336 & 3.8505 \end{bmatrix} \tag{5.24}$$

Figure 5.12 shows the resulting impulse response. Especially the response of the y-position of the load is very fast. The x-position of the load is clearly hard to control, due to the pendulum frequency (also visible in the resulting impulse response of x_l) making the impulse response of x_l badly damped.

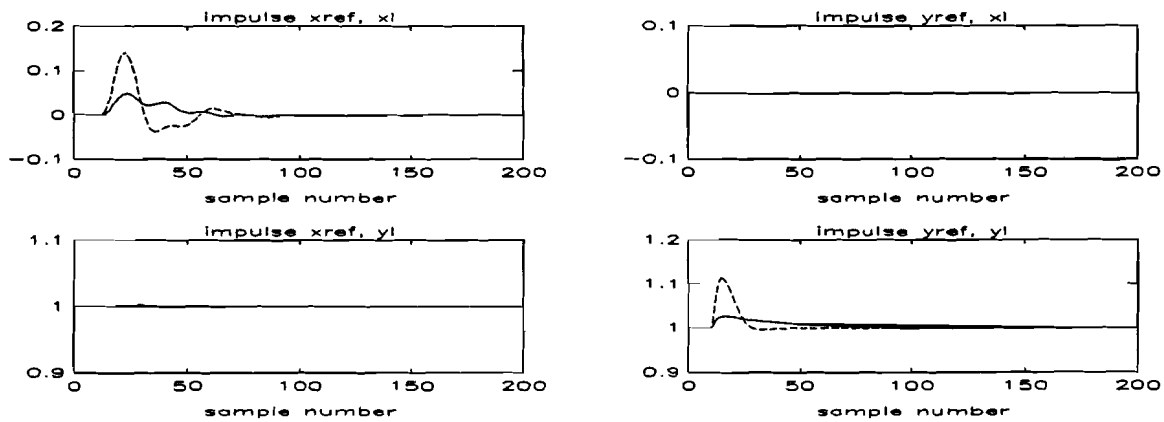


figure 5.12 Resulting impulse response of obtained controller.

where:

— results with controller K(0)
--- results with controller K(5)

Finally the step responses of the initial and the obtained controller are given in figure 5.13. Both references x_{ref} and y_{ref} are fed with a filtered (see formule 5.20) step with an amplitude of 0.75[m] at the same time. Again it is obvious hat x_l is much harder to control than y_l.

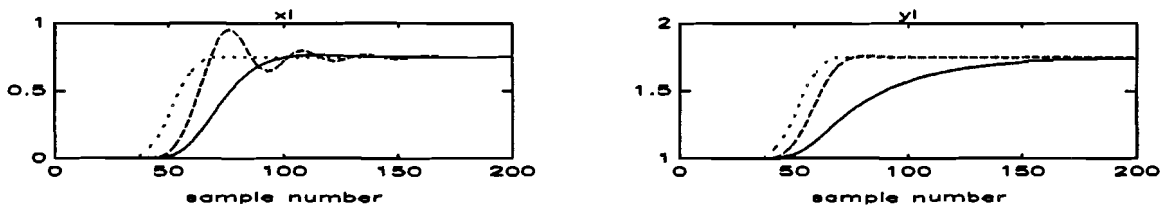


figure 5.13 Resulting step responses of obtained controller.

where:

: reference signal
— results with controller K(0)
--- results with controller K(4)

Penalizing $\dot{\theta}$ in the optimization criterion J to obtain a robust controller makes the performance in the x-position of the load poor. High values in x_t and $\dot{\theta}$ are necessary for fast control of x_l but these are now penalized in the optimization criterion. No penalty on one of the states will lead to a controller which is not stabilizing the nonlinear system. Better results can be obtained when we implement the robustness by the constraint that $\| W_2^{-1}R \|_\infty < 1$. Hence W_2 is of crucial importance. Therefore a description of the perturbation at the estimated model of the system for each frequency in each iteration is necessary. Then it is possible to gain performance at the frequencies where the model is more accurate by choosing the correct weighting filter W_1, and increasing the robustness of the controller for those frequencies where the model is uncertain, defined by the filter W_2. Probably W_2 has large values around the pendulum frequency and for the higher frequencies, where the model is uncertain. Increasing the maximum value of x_t will also lead to a better controller.
The most practical solution is to alter the construction of the crane. In practical situations the load is attached to two cables. This increases the damping of the pendulum frequency.

6. Conclusions and recommendations.

The two-loop control scheme performed well for static state feedback control of the MIMO gantry crane, which is not only unstable in open loop, but also nonlinear. The model of the gantry crane is investigated thoroughly, and one result was that the nonlinear simulation model was minimum phase in terms of linear system theory. Therefore we expect that the two-loop structure is allowed. The estimated MIMO minimum polynomial model of the system (process in closed loop) was each iteration accurate enough to use for controller design, also indicated with the nonlinearity index. Each iteration, a controller with better performance is obtained, and without restrictions a deadbeat controller for the estimated model of the system will be designed. But this leads to no robustness and the controller will make the nonlinear system unstable, even if the estimated model of the system deviates only slightly of the nonlinear simulation model. Therefore robustness is brought into the optimization procedure by penalizing the states, e.g. the derivative of the angle θ. Constraints in time domain e.g. actuator saturation are also used in the optimization procedure. These constraints were satisfied on the linear model of the system, but not always on the nonlinear simulation model, due to model errors. Therefore we must include some redundancy, and take the constraints during controller design slightly more restrictive than actually necessary. This depends on the accuracy of the model. After some iterations a controller is obtained, which performs much better than the initial controller, found by lqr design. Especially the y-position of the load is controlled much more accurate. The x-position of the load is hard to control, due to the badly damped swinging of the pendulum.

To obtain better results, the constraint for obtaining a robust controller: $\| W_2^{-1}R \|_\infty < 1$ must be implemented in the controller design procedure. Then we need more information about the accuracy of the estimated model (given by W_2) for each frequency and in each iteration. A method for identification for robust control is given in [Boo92] and [Zhu91], but not applied in this study. If we have proper filters W_1 and W_2 we can optimize the always existing trade off between performance and robustness.

For the controller design we used a simulation method. But it is also possible to obtain the second loop controller by other design methods. For example we can design each iteration an H_∞ optimal controller. The disadvantage of this method is that one can not put constraints on signals in time domain. Also the main problem, obtaining proper weighting filters, will stay. For all these methods we search for a balance between robustness and performance which are mutually exchangeable.

We may conclude that the scheme gives promising results for the gantry crane, and the question arises if it also performs good for other nonlinear systems. By using the scheme iteratively it is possible to obtain a better controller each time, and the controller will tend to the optimal controller, giving the best performance under the given constraints as actuator satuaration, robustness and controller structure. Necessary conditions for convergence are that the system is strongly stabilizable and the estimated model of the system is accurate enough each iteration.

References.

[Bac89] Backx, A.C.P.M. and A.A.H. Damen
Identification of industrial MIMO processes for fixed controllers, Part I: General theory and practice.
Journal A, vol. 30 (1989), no. 1, pp. 3-12

[Bac92a] Backx, A.C.P.M. and A.A.H. Damen
Identification for the Control of MIMO Industrial Processes.
IEEE Transactions on Automatic Control, vol. 37 (1992), no. 7, pp 980-986

[Bac92b] Backx, A.C.P.M. and A.J.W. van den Boom
Toegepaste Systeem Analyse: college diktaat.
Eindhoven, University of Technology, Faculty of Electrical Engineering, Measurement and Control Group, 1992

[Bel93] Belt, H.J.W.
Two-loop adaptive control with open-loop identification.
M.Sc. Thesis, Eindhoven University of Technology, Faculty of Electrical Engineering, Measurement and Control Group, 1993

[Boo87] Boom, van den, A.J.W. and A.A.H. Damen
Stochastische Systeemtheory: college diktaat.
Eindhoven, University of Technology, Faculty of Electrical Engineering, Measurement and Control Group, 1992

[Boo92] Boom, van den, T.J.J. and A.A.H. Damen, M. Klompstra
Identification for Robust Control Using an H-infinity Norm.
Eindhoven, University of Technology, Faculty of Electrical Engineering, Measurement and Control Group, EUT Report 92-E-261, 1992

[Dam90] Damen, A.A.H. and H.M. van de Ven
Moderne Regeltechniek: college diktaat.
Eindhoven, University of Technology, Faculty of Electrical Engineering, Measurement and Control Group, 1990

[Doy90] Doyle, J. and B. Francis, A. Tannenbaum
Feedback Control Theory
Macmillan Publishing Co., 1990

[Fal93] Falkus, H.M.
Minimum Polynomial identification: user guide.
Eindhoven, University of Technology, Faculty of Electrical Engineering, Measurement and Control Group, 1992

[Isi80] Isidori, A. and C.H. Moog
"On the nonlinear equivalent of the notion of transmission zeros", in modelling and adaptive control.
Lect. Notes Contr. Inf. Sci., vol. 105 (1988), pp. 146-158

[Lju85] Ljung, L and Z, Yuan
Asymptotic Properties of Black-Box Identification of Transfer Functions.
IEEE Transactions on Automatic Control, vol. 30 (1985), no. 6, pp. 514-529

[Lju87] Ljung, L
System Identification: Theory for the user.
Prentice Hall, Englewood Cliffs, New Jersey, 1987

[Nij92] Nijmeijer, H. and A.J. van der Schaft
Nonlinear Dynamical Control Systems.
Springer-Verlag, New York, 1990

[Sch93] Schreppers, M.A.M
Control of a Gantry Crane Process by a Fuzzy Logic Controller.
M.Sc. Thesis, Eindhoven University of Technology, Faculty of Electrical Engineering, Measurement and Control Group, 1993

[Söd89] Söderström, T. and P. Stoica
System Identification.
Prentice Hall, Englewood Cliffs, New Jersey, 1989

[Vid85] Vidyasagar, H.
Control System Synthesis: A Factorization approach.
M.I.T. press, 1985

[Vos92] Vos, F.M.
Adaptive control with open-loop identification.
M.Sc. Thesis, Eindhoven University of Technology, Faculty of Electrical Engineering, Measurement and Control Group, 1992

[Zhu88] Zhu, Y.C. and A.A.H. Damen, P. Eykhoff
A new scheme for identification and control.
Eindhoven, University of Technology, Faculty of Electrical Engineering, Measurement and Control Group, EUT Report 88-E-213, 1988

[Zhu91] Zhu, Y.C. and A.C.P.M. Backx, P. Eykhoff
Multivariable Process Identification for Robust Control.
Eindhoven, University of Technology, Faculty of Electrical Engineering, Measurement and Control Group, EUT Report 91-E-249, 1991

List of symbols

v	system input
w	external input
p	number of inputs (p=2 for the gantry crane)
q	number of outputs (q=2 for the gantry crane)
d	output disturbance $[d_1,\ldots,d_q]$
u	process input $[u_1,\ldots u_p]^T$
y	process output $[y_1,\ldots,y_q]^T$
r	reference input $[r_1,\ldots,r_q]^T$
e	tracking error signal $[e_1,\ldots e_q]^T$
x	state vector $[x_1,\ldots,x_n]^T$
P	process dynamics
$\hat{P}$	estimated process dynamics
K(0)	initial loop controller
K	first loop controller
ΔK	second loop controller
K_1	$K_1 = KM_1$ part of K
K_2	$K_2 = KM_2$ part of K
ΔK_1	$\Delta K_1 = \Delta KM_1$ part of ΔK
ΔK_2	$\Delta K_2 = \Delta KM_2$ part of ΔK
M_1	interconnection matrix 1
M_2	interconnection matrix 2
$\hat{S}$	estimated system!
S	sensitivity!
n	model order (n=6 for the gantry crane)
N	number of samples
Φ_d	power spectral density of d
Φ_u	power spectral density of u
A,B,C	linear state space model matrices
$\tilde{u},\tilde{x},\tilde{y}$	linearized input, state, output
f(x),g(x),h(x)	nonlinear state space model functions
u,x,y	nonlinear input, state, output
U	set of input signals $u \in U \subset \mathbf{R}^q$
$\mathbf{R}$	real numbers
$\alpha(y,\tilde{y})$	$\sup_{u\in U} \frac{\|y-\tilde{y}\|_2}{\|\tilde{y}\|_2}$ nonlinearity index
x_t	position of the trolley
$\dot{x}_t$	velocity of the trolley
θ	angle of the cable
$\dot{\theta}$	angle speed of the cable
L	length of the cable
$\dot{L}$	velocity of the cable
F_t	force on the trolley
F_h	force on the cable
x_l	x-position of the load
y_l	y-position of the load

x_{ref}	x-reference for the load
y_{ref}	y-reference for the load
Δx_t	linearized position of the trolley
$\Delta \dot{x}_t$	linearized velocity of the trolley
$\Delta\theta$	linearized angle of the cable
$\Delta\dot{\theta}$	linearized angle speed of the cable
ΔL	linearized length of the cable
$\Delta \dot{L}$	linearized velocity of the cable
ΔF_t	linearized force on the trolley
ΔF_h	linearized force on the cable
Δx_l	linearized x-position of the load
Δy_l	linearized y-position of the load
Δx_{ref}	linearized x-reference for the load
Δy_{ref}	linearized y-reference for the load
m_t	mass of the trolley
m_l	mass of the load
g	gravitation acceleration g=9.81 m/s^2
α_t	linear friction in the trolley
α_θ	linear friction in the angle
α_L	linear friction in the cable
Δ	controllability matrix
Γ	observability matrix
P	solution Ricatti equation
Q	weighting matrix states for LQR design
R	weighting matrix inputs for LQR design
J	cost function for controller design
f_p	(mathematical) pendulum frequency
T_p	pendulum time
f_s	sample frequency
T_s	sample time
u(k)	input signal at time kT_s
$H(j\omega)$	transfer function
$F\{.\}$	Fourier transformation on data .
z^{-1}	delay operator
A,B,C,D,F	Minimum Polynomial polynomial matrices
I_q	Identity matrix of size q
$\underline{\theta}$	parameter vector
$\underline{\hat{\theta}}$	estimated parameter vector
Z^N	input/output data (N samples)
$V(\underline{\theta},Z^N)$	cost function for identification
$\underline{\hat{\theta}}^i$	ith iteration during identification
ß	norm of search direction
f^i	search direction
g_l	impulse response of the model
h_l	impulse response of the disturbance
$G(z^{-1})$	transfer operator of the model
$H(z^{-1})$	transfer operator of the disturbance
$G(j\omega)$	transfer function of the model
$H(j\omega)$	transfer function of the disturbance

$\hat{G}_N^n(\hat{\theta},e^{j\omega})$	n^{th} order, based on N data samples $G(j\omega)$
$u(k)_{\text{white noise}}$	uniformly distributed white noise sequence
$u(k)_{\text{PRBNS}}$	Pseudo Random Binary Noise Sequence
std(q)	standard deviation from q $= \sqrt{\frac{1}{N}\sum_{k=1}^{N}(q(k)-mean(q))^2}$
mean(q)	mean of q $= \frac{1}{N}\sum_{k=1}^{N} q(k)$
γ	performance indicator
W	exogenous input signal vector
U	actuator input signal vector
Z	regulated output signal vector
Y	measured output signal vector
P_{zw}	open loop transfer from W to Z
P_{zu}	open loop transfer from U to Z
P_{yw}	open loop transfer from W to Y
P_{yu}	open loop transfer from U to Y
H_{zw}	closed loop transfer from W to Z
$\hat{H}$	estimated transfer for h(x)
y_e	output in steady state
W_1	filter for performance
W_2	filter for robustness
$\Delta\hat{S}$	perturbation on estimated model of the system $\hat{S}$
$\hat{S}_\Delta$	perturbed system
dx	additional disturbance input on the states x
$\hat{x}$	estimated states
$\hat{u}$	estimated process input vector
$\hat{F}_l$, $\hat{F}_h$	estimated forces
$\hat{y}$	estimated output
$\hat{e}$	estimated error
ΔK^*	second loop controller without correction K_2^{-1}
u^*	output of second loop controller with correction
LTI	Linear Time Invariant
SISO	Single Input Single Output
MIMO	Multiple Input Multiple Output
LQR	Linear Quadratic Regulater

Appendix 1. An example of a linearizing state feedback.

See the following example where we are interested in the position x.

$$\ddot{x} = x^2 + u \tag{A1.1}$$

We look at this nonlinear unstable system around a working point $x_0 = 1$ and for a predefined input set $U := \{ u = 1 + \text{ampl}\cdot d(t) \mid \text{ampl} \in [\pm 1, \pm 10 \pm 100 \pm 1000] \}$ where d(t) is the unity impulse.
Using $\underline{x}$ as state vector we become in state space form:

$$f(x) = \begin{bmatrix} \dot{x} \\ x^2 \end{bmatrix}; \quad g(x) = \begin{bmatrix} 0 \\ 1 \end{bmatrix}; \quad h(x) = [x]; \quad x = \begin{bmatrix} x \\ \dot{x} \end{bmatrix} \tag{A1.2}$$

where f(x), g(x) and h(x) defined by the nonlinear state space model:

$$\begin{aligned} \dot{x} &= f(x) + g(x)u \\ y &= h(x) \end{aligned} \tag{A1.3}$$

See that g(x) and h(x) are linear functions. The linearized system is derived using a Taylor expansion around $x_0 = 1$ (see section 2.3) taking:

$$A = \begin{bmatrix} 0 & 1 \\ 2 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad C = [1 \;\; 0] \tag{A1.4}$$

where A,B and C are defined by the linear state space model:

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx \end{aligned} \tag{A1.5}$$

The system under feedback is given in figure 2.4:
For the system under feedback yields:

$$f_a(x) = \begin{bmatrix} \dot{x} \\ x^2 - c_1 x - c_2 \dot{x} \end{bmatrix} \quad g(x) = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad h(x) = [x] \tag{A1.6}$$

For the linearized system under feedback (using again the Taylor expansion arround $x_0 = 1$) yields:

$$A = \begin{bmatrix} 0 & 1 \\ 2-c_1 & c_2 \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad C = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \tag{A1.7}$$

Because the open system is unstable we apply the first loop static state feedback. This system can be seen as the initial system. We took $c_1 = 10$ and $c_2 = 10$ which results in a stable system. The response from the system on the input signals is given in figure A1.1.

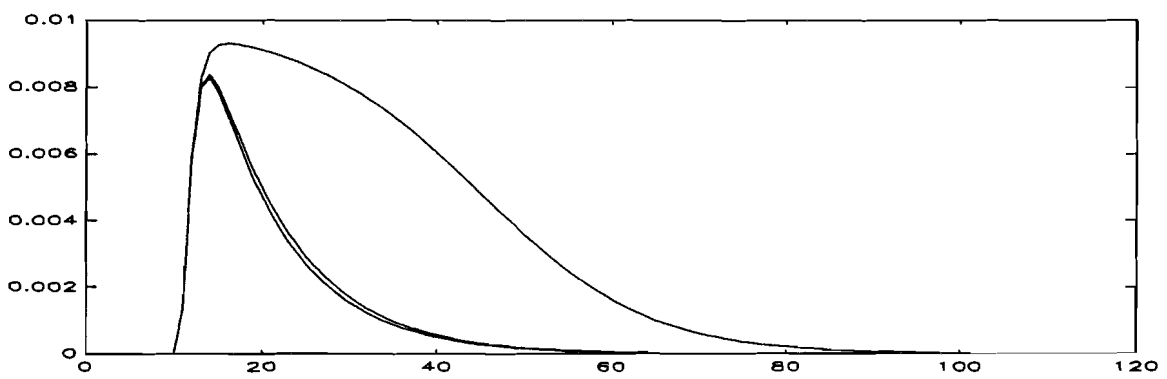


figure A1.1 Feedback $c_1 = 10$, $c_2 = 10$, scaled impulse responses.

If we apply a second loop static state feedback with $\Delta c_1 = 90$ and $\Delta c_2 = 0$ the resulting $c_1 = 100$ and $c_2 = 10$. Again this results in a stable system. The response on the input signals is given in figure A1.2.
In the second case the behaviour of the system is visual more linear arround $x_0 = 1$. This can be measured with the nonlinearity index. For each input the index is:

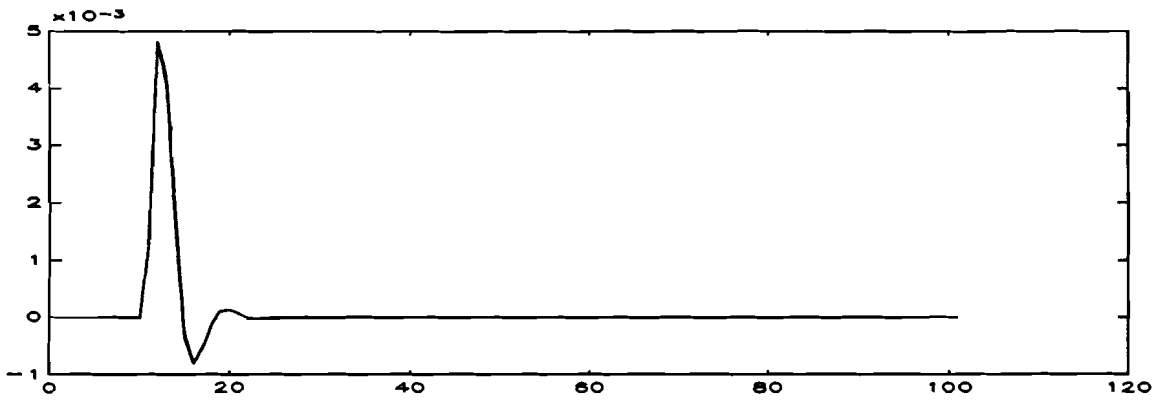


figure A1.2 Feedback $c_1=100$, $c_2=10$, scaled impulse responses.

Table A1.1 nonlinearity index for set U of system in (A1.6).

input	$c_1=10$, $c_2=10$	$c_1=100$, $c_2=10$
u=1-1000·d(t)	0.1541	0.0033
u=1-100·d(t)	0.0293	5.8581e-4
u=1-10·d(t)	0.0304	4.0494e-5
u=1-1·d(t)	0.0314	3.9326e-4
u=1+1·d(t)	0.0315	3.9310e-4
u=1+10·d(t)	0.0312	3.9376e-4
u=1+100·d(t)	0.0177	2.7493e-4
u=1+1000·d(t)	1.7043	4.4630e-4

Taking the supremum follows that $\alpha_1 = 1.7043$ and $\alpha_2 = 0.0033$ which means that in the second case the system behaves more linear for that input set. Note that for all inputs the system with $c_1=10$ and $c_2=100$ is more linear than the system with $c_1=10$ and $c_2=10$. For ampl ϵ [-100, -10, -1, 1, 10, 100, 1000] the system with $c_1=10$ and $c_2=100$ behaves very linear indicated with the small α in column 3 of table A1.1.

Appendix 2. Comprehensive derivation of the nonlinear equations.

The main idea for modelling the gantry crane is that we obtain a pole coördinate system moving along with the trolley (Galileï transformation). This causes an always present relative force F_r in the opposite direction in the polar system [Roe87]. We may now calculate the force balans in both the (innert) subsystems apart, so for each subsystem must yield:

$$m\ddot{\underline{r}} = \sum_i \bar{F}_i \qquad \text{(A2.1)}$$

where $\underline{r} = (x,y)$ for the cartesian system and $\underline{r} = (r\cos(\theta), r\sin(\theta))$ in polar system.

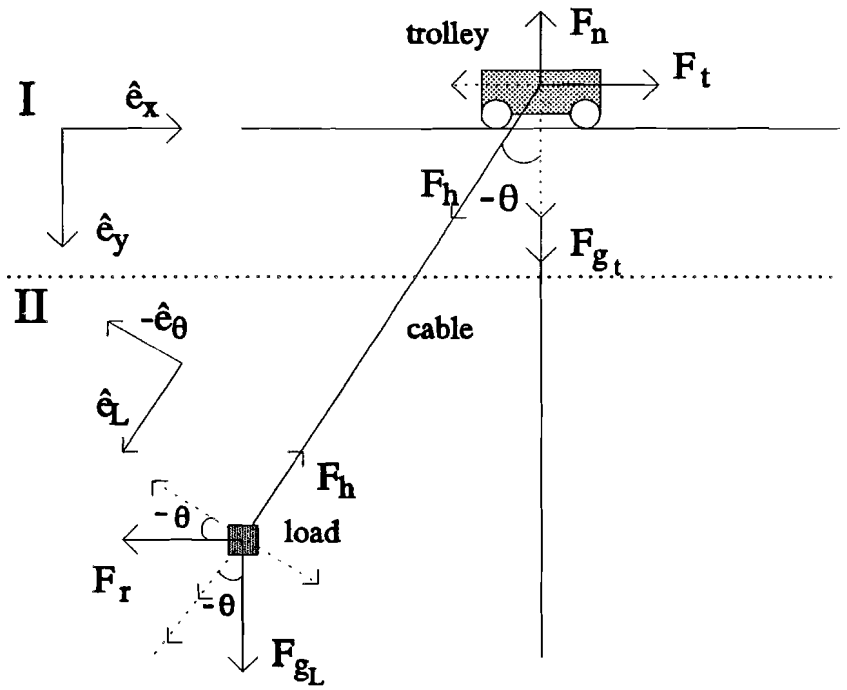


figure A2.1 Forces balance of the gantry crane.

We see that when the trolley is moving to the right, the angle between cable and the equilibrium position of the cable becomes negative. We assume that $-\frac{1}{2}\pi < \theta < \frac{1}{2}\pi$. The forces $\begin{bmatrix} F_t \\ F_h \end{bmatrix}$ can be controlled. The position of the load given by $y = \begin{bmatrix} xl \\ yl \end{bmatrix}$ must be controlled in such way that it follows as good as possible a predefined track $ref = \begin{bmatrix} xref \\ yref \end{bmatrix}$. Let us look at:

I. Force balans of the trolley.

$$\begin{aligned} \hat{e}_x:&\ m_t\ddot{x}_{t_x} = F_t - F_h\sin(-\theta) \\ \hat{e}_y:&\ m_t\ddot{y}_{t_y} = F_{g_t} + F_h\cos(-\theta) - F_n \end{aligned} \qquad \text{(A2.2)}$$

Of course the resulting acceleration of the trolley in the y-direction will be zero $\ddot{y}_t = 0$ if the crane is designed properly! The resulting acceleration in the x_t-direction equals $\ddot{x}$. Further yields: : $\cos(-\theta) = \cos(\theta)$ and $\sin(-\theta) = -\sin(\theta)$ for $-\frac{1}{2}\pi < \theta < \frac{1}{2}\pi$. Rewriting (A2.2) gives:

$$\begin{aligned} \hat{e}_x:&\ m_t\ddot{x}_t = F_t + F_h\sin(\theta) \\ \hat{e}_y:&\ F_n = F_{g_t} + F_h\cos(\theta) \end{aligned} \qquad \text{(A2.3)}$$

II. Force balans of the load.

$$\begin{aligned} \hat{e}_L: \quad & m_L\ddot{r}_L = F_{g_L}\cos(-\theta) + F_r\sin(-\theta) - F_k \\ \hat{e}_\theta: \quad & m_L\ddot{r}_\theta = -F_r\cos(-\theta) + F_{g_L}\sin(-\theta) \end{aligned} \tag{A2.4}$$

In the polar system we can decompose $\underline{r}$ in a radial direction $\hat{e}_L$ and a orthogonal direction $\hat{e}_\theta$. In the polar system yields: $\underline{r} = L\cos(\theta)\hat{e}_x + L\sin(\theta)\hat{e}_y$ but also $\underline{r} = L\hat{e}_L$. Now the acceleration $\underline{a}$ and velocity $\underline{v}$ of a mass point at place $\underline{r}$ are related as:

$$\begin{aligned} \underline{v} &= \dot{\underline{r}} = \frac{d\underline{r}}{dt} \\ \underline{a} &= \dot{\underline{v}} = \ddot{\underline{r}} = \frac{d^2\underline{r}}{dt^2} \end{aligned} \tag{A2.5}$$

with the flux defined as:

$$\dot{q} = \frac{dq}{dt} = \lim_{\Delta t \to 0}\frac{\Delta q}{\Delta t} \tag{A2.6}$$

If we take as place vector $\underline{r} = L\hat{e}_L$ we derive for the velocity:

$$\underline{v} = \frac{d\underline{r}}{dt} = \frac{dL\hat{e}_L}{dt} = \dot{L}\hat{e}_L + L\dot{\hat{e}}_L = \dot{L}\hat{e}_L + L\dot{\theta}\hat{e}_\theta \tag{A2.7}$$

Here we used: $\dot{\hat{e}}_L = \dot{\theta}\hat{e}_\theta$ which can be derived in the folowing way. In a timestep Δt the place vector $\underline{r}$ will change not only in length but also turn in direction over $-\Delta\theta$. But also the axis $\hat{e}_L$ and $\hat{e}_\theta$ are turned over the angle $-\Delta\theta$ in the timestep Δt. Because we take the axis normalized on 1 each timestep, must yield: $\| \hat{e}_L(t) \| = \| \hat{e}_L(t+\Delta t) \| = 1$ and $\| \hat{e}_L(t) \| = \| \hat{e}_L(t+\Delta t) \| = 1$. The change in $\hat{e}_L$ equals: $\| \Delta\hat{e}_L \| = \| 2\sin(-\frac{1}{2}\Delta\theta) \| = 2\sin(\frac{1}{2}\Delta\theta)$ and also the change in $\hat{e}_\theta$ equals: $\| \Delta\hat{e}_\theta \| = \| 2\sin(-\frac{1}{2}\Delta\theta) \| = 2\sin(\frac{1}{2}\Delta\theta)$, as follows from figure A2.2.

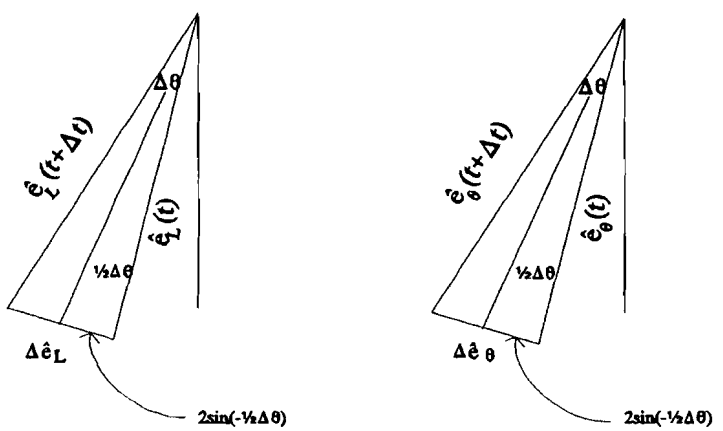


figure A2.2 Change in the axis vectors.

$\Delta\hat{e}_L$ makes an angle of $\frac{1}{2}\pi + \frac{1}{2}\Delta\theta$ with $\hat{e}_L$. So if $\Delta\theta \to 0$, $\Delta\hat{e}_L$ has the same direction as $\hat{e}_\theta$. In the same way, $\Delta\hat{e}_\theta$ makes an angle of $\frac{1}{2}\pi + \frac{1}{2}\Delta\theta$ with $\hat{e}_\theta$. So if $\Delta\theta \to 0$, $\Delta\hat{e}_\theta$ has opposite direction of $\hat{e}_L$. Now follows:

$$\begin{aligned} \dot{e}_L &= \frac{d\hat{e}_L}{dt} = \lim_{\Delta t\to 0}\frac{\Delta\hat{e}_L}{\Delta t} = \hat{e}_\theta \lim_{\Delta t\to 0}\frac{|\Delta\hat{e}_L|}{\Delta t} = \hat{e}_\theta \lim_{\Delta t\to 0}\frac{|\Delta\hat{e}_L|}{\Delta\theta}\frac{\Delta\theta}{\Delta t} \\ &= \hat{e}_\theta \lim_{\Delta\theta\to 0}\frac{2\sin(\frac{1}{2}\Delta\theta)}{\Delta\theta}\lim_{\Delta t\to 0}\frac{\Delta\theta}{\Delta t} = \hat{e}_\theta \cdot 1 \cdot \dot{\theta} = \dot{\theta}\hat{e}_\theta \end{aligned} \tag{A2.8}$$

which was used in (A2.7). Also yields:

$$\dot{\hat{e}}_\theta = \lim_{\Delta t \to 0} \frac{\Delta \hat{e}_\theta}{\Delta t} = -\hat{e}_L \lim_{\Delta\theta \to 0} \frac{|\Delta \hat{e}_\theta|}{\Delta\theta} \lim_{\Delta t \to 0} \frac{\Delta\theta}{\Delta t} = -\hat{e}_L \cdot 1 \cdot \dot{\theta} = -\dot{\theta}\hat{e}_L \tag{A2.9}$$

For the resulting acceleration of the load we can now derive:

$$\begin{aligned} a = \ddot{y} &= \frac{d(\dot{L}\hat{e}_L + L\dot{\theta}\hat{e}_\theta)}{dt} = \ddot{L}\hat{e}_L + \dot{L}\dot{\hat{e}}_L + \dot{L}\dot{\theta}\hat{e}_\theta + L\ddot{\theta}\hat{e}_\theta + L\dot{\theta}\dot{\hat{e}}_\theta \\ &= \ddot{L}\hat{e}_L + \dot{L}\dot{\theta}\hat{e}_\theta + \dot{L}\dot{\theta}\hat{e}_\theta + L\ddot{\theta}\hat{e}_\theta - L\dot{\theta}^2\hat{e}_L \\ &= (\ddot{L} - L\dot{\theta}^2)\hat{e}_L + (L\ddot{\theta} + 2\dot{L}\dot{\theta})\hat{e}_\theta \end{aligned} \tag{A2.10}$$

substituting this result in (A2.4) gives:

$$\begin{aligned} \hat{e}_L: \quad & m_L(\ddot{L} - L\dot{\theta}^2) = F_{g_L}\cos(\theta) - F_r\sin(\theta) - F_h \\ \hat{e}_\theta: \quad & m_L(L\ddot{\theta} + 2\dot{L}\dot{\theta}) = -F_{g_L}\sin(\theta) - F_r\cos(\theta) \end{aligned} \tag{A2.11}$$

Expressing the forces with Newton's first law:

$$\begin{aligned} F_{g_L} &= m_L g \\ Fr &= m_L \ddot{x}_t \end{aligned} \tag{A2.12}$$

The system can now be described by three important coupled nonlinear differential equations as given in (A2.13). As proposed in [Sch93] we add three linear friction terms α_t, α_θ and α_L. This means that nonlinearities due to hystereses and saturation are not modeled. Also other nonlinearities are not modeled. For example, if the mass of the cable can not be ignored, the cable will bow when the trolley moves. Further this model is derived for a cable that can put a force on the load in positive $\hat{e}_L$ direction (downwards) which is not possible in the real case. During simulations this can be overcome by taking $F_h > -m_L g$.

$$\begin{aligned} & m_t \cdot \ddot{x}_t = F_t + F_h \cdot \sin(\theta) - \alpha_t \cdot \dot{x}_t \\ & m_L \cdot L \cdot \ddot{\theta} = -m_L \cdot g \cdot \sin(\theta) - m_L \cdot \ddot{x}_t \cdot \cos(\theta) - 2 \cdot m_L \cdot \dot{L} \cdot \dot{\theta} - \alpha_\theta \cdot L \cdot \dot{\theta} \\ & m_L \cdot \ddot{L} = m_L \cdot g\cos(\theta) - Fh + m_L \cdot L \cdot \dot{\theta}^2 - m_L \cdot \ddot{x}_t \cdot \sin(\theta) - \alpha_L \cdot \dot{L} \end{aligned} \tag{A2.13}$$

We can distinguish the states of the system as $\underline{x} = [x_t\ \dot{x}_t\ \theta\ \dot{\theta}\ L\ \dot{L}]^T$.
The position of the load can be expressed using these states with:

$$\begin{aligned} x_l &= x_t + L\sin(\theta) \\ y_l &= L\cos(\theta) \end{aligned} \tag{A2.14}$$

The velocity of the load is given by (only for completeness):

$$\begin{aligned} \dot{x}_l &= \dot{x}_t + \dot{L}\sin(\theta) + L\dot{\theta}\cos(\theta) \\ \dot{y}_l &= \dot{L}\cos(\theta) - L\dot{\theta}\sin(\theta) \end{aligned} \tag{A2.15}$$

Using the nonlinear state space model:

$$\begin{aligned} \dot{x} &= f(x) + g(x)u \\ y &= h(x) \end{aligned} \tag{A2.16}$$

we can rewrite (A2.13) in:

$$\begin{aligned} \ddot{x}_t &= \frac{F_t}{m_t} + \frac{F_h}{m_t}\sin(\theta) - \frac{\alpha_t}{m_t}\dot{x}_t \\ \ddot{\theta} &= -\frac{g}{L}\sin(\theta) - \frac{1}{m_t L}(F_t + F_h\sin(\theta) - \alpha_t\dot{x}_t)\cos(\theta) - (2\frac{\dot{L}}{L} + \frac{\alpha_\theta}{m_L})\dot{\theta} \\ \ddot{L} &= g\cos(\theta) - \frac{F_h}{m_L} + L\dot{\theta}^2 - \frac{1}{m_t}(F_t + F_h\sin(\theta) - \alpha_t\dot{x}_t)\sin(\theta) - \frac{\alpha_L}{m_L}\dot{L} \end{aligned} \tag{A2.17}$$

and again rewrite in:

$$\begin{aligned}
\ddot{x}_t &= \frac{F_t}{m_t} + \frac{F_h}{m_t}\sin(\theta) - \frac{\alpha_t}{m_t}\dot{x}_t \\
\ddot{\theta} &= -\frac{g}{L}\sin(\theta) - \frac{F_t}{m_t L}\cos(\theta) - \frac{F_h}{m_t L}\sin(\theta)\cos(\theta) + \frac{\alpha_t}{m_t L}\dot{x}_t\cos(\theta) - (2\frac{\dot{L}}{L} + \frac{\alpha_\theta}{m_L})\dot{\theta} \\
\ddot{L} &= g\cos(\theta) - \frac{F_h}{m_L} + L\dot{\theta}^2 - \frac{F_t}{m_t}\sin(\theta) - \frac{F_h}{m_t}\sin^2(\theta) + \frac{\alpha_t}{m_t}\dot{x}_t\sin(\theta) - \frac{\alpha_L}{m_L}\dot{L}
\end{aligned} \tag{A2.18}$$

Now the non linear state space model becomes:

$$\dot{x} = \begin{bmatrix} \dot{x}_t \\ \ddot{x}_t \\ \dot{\theta} \\ \ddot{\theta} \\ \dot{L} \\ \ddot{L} \end{bmatrix} = \begin{bmatrix} \dot{x}_t \\ -\frac{\alpha_t}{m_t}\dot{x}_t \\ \dot{\theta} \\ -\frac{g}{L}\sin(\theta) + \frac{\alpha_t}{m_t L}\dot{x}_t\cos(\theta) - (2\frac{\dot{L}}{L} + \frac{\alpha_\theta}{m_L})\dot{\theta} \\ \dot{L} \\ g\cos(\theta) + L\dot{\theta}^2 + \frac{\alpha_t}{m_t}\dot{x}_t\sin(\theta) - \frac{\alpha_L}{m_L}\dot{L} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ \frac{1}{m_t} & \frac{1}{m_t}\sin(\theta) \\ 0 & 0 \\ -\frac{1}{m_t L}\cos(\theta) & -\frac{1}{m_t L}\sin(\theta)\cos(\theta) \\ 0 & 0 \\ -\frac{1}{m_t}\sin(\theta) & -\frac{1}{m_L} - \frac{1}{m_t}\sin^2(\theta) \end{bmatrix} \begin{bmatrix} F_t \\ F_h \end{bmatrix} \tag{A2.19}$$

$$y = \begin{bmatrix} x_t + L\sin(\theta) \\ L\cos(\theta) \end{bmatrix} \tag{A2.20}$$

The equations hold only when $L \neq 0$, else we divide by zero.

Appendix 3. Comprehensive derivation of the linearized model.

Define the working point $\underline{x}_0$:

$$\begin{aligned} x_t &= x_{t_0} + \Delta x_t = \Delta x_t \\ \dot{x}_t &= \dot{x}_{t_0} + \Delta \dot{x}_t = \Delta \dot{x}_t \\ \theta &= \theta_0 + \Delta\theta = \Delta\theta \\ \dot{\theta} &= \dot{\theta}_0 + \Delta\dot{\theta} = \Delta\dot{\theta} \\ L &= L_0 + \Delta L = 1 + \Delta L \\ \dot{L} &= \dot{L}_0 + \Delta\dot{L} = \Delta\dot{L} \end{aligned} \tag{A3.1}$$

We can substitute this in the nonlinear coupled equations yielding:

$$\begin{aligned} & m_t \cdot \Delta\ddot{x}_t = F_t + F_h \cdot \sin(\Delta\theta) - \alpha_t \cdot \Delta\dot{x}_t \\ & m_t \cdot (L_0 + \Delta L) \cdot \Delta\ddot{\theta} = -m_t \cdot g \cdot \sin(\Delta\theta) - m_L \cdot \Delta\ddot{x}_t \cdot \cos(\Delta\theta) - 2 m_L \cdot (L_0 + \Delta L) \cdot \Delta\dot{\theta} - \alpha \cdot (L_0 + \Delta L) \cdot \Delta\dot{\theta} \\ & m_L \cdot (L_0 + \Delta L) = m_L \cdot g \cdot \cos(\Delta\theta) - F_h - m_L \cdot (L_0 + \Delta L) \cdot \Delta\dot{\theta}^2 - m_L \cdot \Delta\ddot{x}_t \cdot \sin(\Delta\theta) - \alpha_L \cdot (L_0 + \Delta L) \end{aligned} \tag{A3.2}$$

Using:

$$\begin{aligned} & \cos(x) = 1 \quad \text{if } x \approx 0 \\ & \sin(x) = x \end{aligned} \tag{A3.3}$$

$$\begin{aligned} & \frac{\partial(c + \Delta x)}{\partial t} = \Delta\dot{x} \\ & \Delta x \cdot \Delta y \approx 0 \\ & (\Delta x)^2 \approx 0 \end{aligned} \tag{A3.4}$$

follows:

$$\begin{aligned} & m_t \cdot \Delta\ddot{x}_t = F_t + F_h \cdot \Delta\theta - \alpha_t \cdot \Delta\dot{x}_t \\ & m_L \cdot \Delta\ddot{\theta} = -\frac{m_L}{L_0} \cdot g \cdot \Delta\theta - \frac{m_L}{L_0} \cdot \Delta\ddot{x}_t - \alpha_\theta \cdot \Delta\dot{\theta} \\ & m_L \cdot \Delta\ddot{L} = m_L \cdot g - F_h - \alpha_L - \alpha \cdot \Delta\dot{L} \end{aligned} \tag{A3.5}$$

Substitude of the first equation in the other gives the final decoupled linear equations:

$$\begin{aligned} & m_t \cdot \Delta\ddot{x}_t = \Delta F_t + \Delta F_h \cdot \Delta\theta - \alpha_t \cdot \Delta\dot{x}_t \\ & m_L \cdot \Delta\ddot{\theta} = -\frac{m_L}{L_0} \cdot g \cdot \Delta\theta - \frac{m_L}{L_0} \cdot \left(\frac{\Delta F_t}{m_t} + \frac{m_L \cdot g}{m_t} \cdot \theta - \frac{\alpha_t}{m_t} \cdot \Delta\dot{x}_t\right) - \alpha_\theta \cdot \Delta\dot{\theta} \\ & m_L \cdot \Delta\ddot{L} = -\Delta F_h - \alpha_L \cdot \Delta\dot{L} \end{aligned} \tag{A3.6}$$

where we used:

$$\begin{aligned} F_t &= F_{t_0} + \Delta F_t = \Delta F_t \\ F_h &= F_{h_0} + \Delta F_h = m_L \cdot g + \Delta F_h \end{aligned} \tag{A3.7}$$

If we use the state space notation, then must yield:

$$\begin{aligned} \dot{x} &= A \cdot x + B \cdot u \\ y &= C \cdot x \end{aligned} \tag{A3.8}$$

where A,B,C are matrices and x,y and u vectors. In the situation of the gantry crane we have:

$$x = (\Delta x_t \ \Delta\dot{x}_t \ \Delta\theta \ \Delta\dot{\theta} \ \Delta L \ \Delta\dot{L})^T \qquad u = (\Delta F_t \ \Delta F_h)^T \tag{A3.9}$$

where x is the deviation from the linearization point. Now we can calculate A and B:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & -\frac{\alpha_t}{m_t} & \frac{m_L \cdot g}{m_t} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & \frac{\alpha_t}{m_t \cdot L_0} & -\frac{g}{L_0}\left(1 + \frac{m_L}{m_t}\right) & -\frac{\alpha_\theta}{m_L} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & -\frac{\alpha_L}{m_L} \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 \\ \frac{1}{m_t} & 0 \\ 0 & 0 \\ -\frac{1}{m_t \cdot L_0} & 0 \\ 0 & 0 \\ 0 & -\frac{1}{m_L} \end{bmatrix} \tag{A3.10}$$

In the same way we calculate C. Define the linearization point of the load as:
which results in:

$$\begin{aligned} x_l &= x_{l_0} + \Delta x_l; \quad x_{l_0} = 0; \quad \rightarrow xl = \Delta x_l \\ y_l &= y_{l_0} + \Delta y_l; \quad y_{l_0} = 1; \quad \rightarrow yl = y_{l_0} + \Delta y_l \end{aligned} \tag{A3.11}$$

$$\begin{aligned} \Delta x_l &= \Delta x_t + (L_0 + \Delta L)\cdot\sin(\Delta\theta) \\ y_{l_0} + \Delta y_l &= (L_0 + \Delta L)\cdot\cos(\Delta\theta) \end{aligned} \tag{A3.12}$$

Neglecting the second order terms and knowing that $L_0 = y_{l0}$:

$$\begin{aligned} \Delta x_l &= \Delta x_t + L_0\cdot\theta \\ \Delta y_l &= \Delta L \end{aligned} \tag{A3.13}$$

Now follows C from the definition of the state space model.

$$C = \begin{bmatrix} 1 & 0 & L_0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \tag{A3.14}$$

Appendix 4. Zero dynamics of the gantry crane.

For the output must yields:

$$\begin{aligned} x_l &= x_t + L\sin(\theta) = 0 \\ y_l &= L\cos(\theta) = 1 \end{aligned} \tag{A4.1}$$

Now follows:

$$\begin{aligned} \sin(\theta) &= \frac{-x_t}{L} \\ \cos(\theta) &= \frac{1}{L} \end{aligned} \tag{A4.2}$$

which yields:

$$\begin{aligned} &\sin^2(\theta) + \cos^2(\theta) = 1 \\ &\frac{x_t^2}{L^2} + \frac{1}{L^2} = 1 \\ &L = \sqrt{1+x_t^2} \quad \textit{on physical ground } L > 0 \end{aligned} \tag{A4.3}$$

We also know that the derivative of the output is zero:

$$\begin{aligned} \dot{x}_l &= \dot{x}_t + \dot{L}\sin(\theta) + L\dot{\theta}\cos(\theta) = 0 \\ \dot{y}_l &= \dot{L}\cos(\theta) - L\dot{\theta}\sin(\theta) = 0 \end{aligned} \tag{A4.4}$$

$$\dot{x}_t - \frac{\dot{L}x_t}{L} + \dot{\theta} = 0 \tag{A4.5}$$

$$\frac{\dot{L}}{L} = -\dot{\theta}x_t$$

$$\dot{\theta} = \frac{-\dot{x}_t}{1+x_t^2} \tag{A4.6}$$

After doing some mathematical work follows:

$$\begin{aligned} \sin(\theta) &= \frac{-x_t}{\sqrt{1+x_t^2}} \\ \cos(\theta) &= \frac{1}{\sqrt{1+x_t^2}} \\ L &= \sqrt{1+x_t^2} \\ \dot{\theta} &= \frac{-\dot{x}_t}{1+x_t^2} \\ \dot{L} &= \frac{\dot{x}_t x_t}{\sqrt{1+x_t^2}} \\ \ddot{\theta} &= \frac{2x_t\dot{x}_t^2 - (1+x_t^2)\ddot{x}_t}{(1+x_t^2)^2} \\ \ddot{L} &= \frac{\ddot{x}_t(x_t+x_t^3)+\dot{x}_t^2}{(1+x_t^2)^{3/2}} \end{aligned} \tag{A4.7}$$

The equation of motion for the load are:

$$\begin{cases} m_t\cdot\ddot{x}_t = u_t + u_h\cdot\sin(\theta) - \alpha_t\cdot\dot{x}_t \\ m_L\cdot L\cdot\ddot{\theta} = -m_L\cdot g\cdot\sin(\theta) - m_L\cdot\ddot{x}_t\cdot\cos(\theta) - 2\cdot m_L\cdot\dot{L}\cdot\dot{\theta} - \alpha_\theta\cdot L\cdot\dot{\theta} \\ m_L\cdot\ddot{L} = m_L\cdot g\cdot\cos(\theta) - u_h + m_L\cdot L\cdot\dot{\theta}^2 - m_L\cdot\ddot{x}_t\cdot\sin(\theta) - \alpha_L\cdot\dot{L} \end{cases} \tag{A4.8}$$

where:

$$\begin{aligned} u_t &= F_t + [c_{11}\ c_{12}\ c_{13}\ c_{14}\ c_{15}\ c_{16}]\ (x - [0\ 0\ 0\ 0\ 1\ 0]^T) \\ u_h &= F_h + [c_{21}\ c_{22}\ c_{23}\ c_{24}\ c_{25}\ c_{26}]\ (x - [0\ 0\ 0\ 0\ 1\ 0]^T) + m_L g \\ \text{and: } C &= \begin{bmatrix} c_{11} & c_{12} & c_{13} & c_{14} & c_{15} & c_{16} \\ c_{21} & c_{22} & c_{23} & c_{24} & c_{25} & c_{26} \end{bmatrix} \end{aligned} \tag{A4.9}$$

Substitution of the derived equations (A4.7) in (A4.8) gives:

$$\begin{cases} \ddot{x}_t = \dfrac{u_t}{m_t} - \dfrac{u_h x_t}{m_t\sqrt{1+x_t^2}} - \dfrac{\alpha_t \dot{x}_t}{m_t} \\ \dfrac{2x_t\dot{x}_t^2-(1+x_t^2)\ddot{x}_t}{(1+x_t^2)^2} = \dfrac{gx_t}{1+x_t^2} - \dfrac{\ddot{x}_t}{1+x_t^2} + \dfrac{2x_t\dot{x}_t^2}{(1+x_t^2)^2} + \dfrac{\alpha_\theta \dot{x}_t}{m_L(1+x_t^2)} \\ \dfrac{(x_t+x_t^3)\ddot{x}_t+\dot{x}^2}{(1+x_t^2)^{3/2}} = \dfrac{g}{\sqrt{1+x_t^2}} + \dfrac{\dot{x}_t^2}{(1+x_t^2)^{3/2}} + \dfrac{x_t\ddot{x}_t}{\sqrt{1+x_t^2}} - \dfrac{u_h}{m_L} - \dfrac{\alpha_L x_t \dot{x}_t}{m_L\sqrt{1+x_t^2}} \end{cases} \tag{A4.10}$$

rewriting to:

$$\begin{cases} \ddot{x}_t = \dfrac{u_t}{m_t} - \dfrac{u_h x_t}{m_t\sqrt{1+x_t^2}} - \dfrac{\alpha_t \dot{x}_t}{m_t} \\ gx_t = -\dfrac{\alpha_\theta \dot{x}_t}{m_L} \\ 0 = \dfrac{g}{\sqrt{1+x_t^2}} - \dfrac{u_h}{m_L} + \dfrac{\alpha_L x_t \dot{x}_t}{m_L\sqrt{1+x_t^2}} \end{cases} \tag{A4.11}$$

and to:

$$\begin{cases} u_t = m_t\ddot{x}_t + \dfrac{m_L g x_t - \alpha_L x_t^2 \dot{x}_t}{1+x_t^2} + \alpha_t \dot{x}_t \\ \dot{x}_t = -\dfrac{m_L g x_t}{\alpha_\theta} \\ u_h = \dfrac{m_L g - \alpha_L x_t \dot{x}_t}{\sqrt{1+x_t^2}} \end{cases} \tag{A4.12}$$

From the second equation in (A4.12) follows that x must behave like the solution of a first order differential problem with a stable pole in the LHP.
The inputs are given by:

$$u_h = \frac{\alpha_\theta m_L g + \alpha_L m_L g x_t^2}{\alpha_\theta\sqrt{1 + x_t^2}} \tag{A3.13}$$

$$u_t = m_t\left(\frac{m_L g}{\alpha_\theta}\right)^2 x_t - \frac{\alpha_L m_L g x_t^3}{\alpha_\theta\sqrt{1 + x_t^2}} \tag{A4.14}$$

and around the equilibrium point:

$$u_{zd_h} = \frac{\alpha_L m_L g x_t^2}{\alpha_\theta\sqrt{1 + x_t^2}} \tag{A3.15}$$

$$u_{zd_t} = m_t\left(\frac{m_L g}{\alpha_\theta}\right)^2 x_t - \frac{\alpha_L m_L g x_t^3}{\alpha_\theta\sqrt{1 + x_t^2}} \tag{A4.16}$$

Now follows that $\lim_{t\to\infty} u_{zd}(t) = 0$ since $\lim_{t\to\infty} x(t) = 0$. Therefore the zero dynamics are said to be stable. This can be compared with zeros in the RHP in linear system theory. For linear systems this implies that the system is strongly stabilizable. We expect that this also yields for nonlinear systems, and hence we may use the two-loop scheme for the gantry crane.

Appendix 5. Controllability of the gantry crane.

For controllability must yield:

$$rank\ ([g1,g2,[g1,f],[g2,f],[[g1,f],f],[[g2,f],f]]) = n \tag{A5.1}$$

where the nonlinear vectors are given by:

$$g_1 = \begin{bmatrix} 0 \\ \frac{1}{m_t} \\ 0 \\ -\frac{1}{m_t L}\cos(\theta) \\ 0 \\ -\frac{1}{m_t}\sin(\theta) \end{bmatrix} \quad g_2 = \begin{bmatrix} 0 \\ \frac{1}{m_t}\sin(\theta) \\ 0 \\ -\frac{1}{m_t L}\sin(\theta)\cos(\theta) \\ 0 \\ -\frac{1}{m_L} - \frac{1}{m_t}\sin^2(\theta) \end{bmatrix} \tag{A5.2}$$

$$[g_1,f] = \frac{\partial f}{\partial x} g_1(x) - \frac{\partial g_1}{\partial x} f(x) =$$

$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & -\frac{\alpha_t}{m_t} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & \frac{\alpha_t}{m_t L}\cos(\theta) & -\frac{g}{L}\cos(\theta) - \frac{\alpha_t}{m_t L}\dot{x}_t\sin(\theta) & -(2\frac{\dot{L}}{L} + \frac{\alpha_\theta}{m_L}) & \frac{g}{L^2}\sin(\theta) - \frac{\alpha_t}{m_t L^2}\dot{x}_t\cos(\theta) + \frac{2\dot{L}}{L^2}\dot{\theta} & -\frac{2}{L}\dot{\theta} \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & \frac{\alpha_t}{m_t}\sin(\theta) & -g\sin(\theta) + \frac{\alpha_t}{m_t}\dot{x}_t\cos(\theta) & 2L\dot{\theta} & \dot{\theta}^2 & -\frac{\alpha_L}{m_L} \end{bmatrix} * \begin{bmatrix} 0 \\ \frac{1}{m_t} \\ 0 \\ -\frac{1}{m_t L}\cos(\theta) \\ 0 \\ -\frac{1}{m_t}\sin(\theta) \end{bmatrix} \tag{A5.3}$$

$$- \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{m_t L}\sin(\theta) & 0 & \frac{1}{m_t L^2}\cos(\theta) & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{m_t}\cos(\theta) & 0 & 0 & 0 \end{bmatrix} * \begin{bmatrix} \dot{x}_t \\ -\frac{\alpha_t}{m_t}\dot{x}_t \\ \dot{\theta} \\ -\frac{g}{L}\sin(\theta) + \frac{\alpha_t}{m_t L}\dot{x}_t\cos(\theta) - (2\frac{\dot{L}}{L} + \frac{\alpha_\theta}{m_L})\dot{\theta} \\ \dot{L} \\ g\cos(\theta) + L\dot{\theta}^2 + \frac{\alpha_t}{m_t}\dot{x}_t\sin(\theta) - \frac{\alpha_L}{m_L}\dot{L} \end{bmatrix}$$

$$[g1,f] = \begin{bmatrix} \frac{1}{m_t} \\ -\frac{\alpha_t}{m_t^2} \\ -\frac{1}{m_t L}\cos(\theta) \\ \frac{\alpha_t}{m_t^2 L}\cos(\theta) + \frac{1}{m_t L}(2\frac{\dot{L}}{L} + \frac{\alpha_\theta}{m_L})\cos(\theta) + \frac{2}{m_t L}\dot{\theta}\sin(\theta) \\ -\frac{1}{m_t}\sin(\theta) \\ \frac{\alpha_t}{m_t^2}\sin(\theta) - \frac{2}{m_t}\dot{\theta}\cos(\theta) + \frac{\alpha_L}{m_t m_L}\sin(\theta) \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{1}{m_t L}\dot{\theta}\sin(\theta) + \frac{\dot{L}}{m_t L^2}\cos(\theta) \\ 0 \\ -\frac{1}{m_t}\dot{\theta}\cos(\theta) \end{bmatrix} \tag{A5.4}$$

$$[g1\,f] = \begin{bmatrix} \frac{1}{m_t} \\ -\frac{\alpha_t}{m_t^2} \\ -\frac{1}{m_t L}\cos(\theta) \\ \frac{\alpha_t}{m_t^2 L}\cos(\theta)+\frac{1}{m_t L}\left(\frac{\dot{L}}{L}+\frac{\alpha_\theta}{m_L}\right)\cos(\theta)+\frac{1}{m_t L}\dot{\theta}\sin(\theta) \\ -\frac{1}{m_t}\sin(\theta) \\ \frac{\alpha_t}{m_t^2}\sin(\theta)-\frac{1}{m_t}\dot{\theta}\cos(\theta)+\frac{\alpha_L}{m_t m_L}\sin(\theta) \end{bmatrix} \tag{A5.5}$$

$$[g_2 f] = \frac{\partial f}{\partial x} g_2(x) - \frac{\partial g_2}{\partial x} f(x) =$$

$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & -\frac{\alpha_t}{m_t} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & \frac{\alpha_t}{m_t L}\cos(\theta) & -\frac{g}{L}\cos(\theta)-\frac{\alpha_t}{m_t L}\dot{x}_t\sin(\theta) & -\left(2\frac{\dot{L}}{L}+\frac{\alpha_\theta}{m_L}\right) & \frac{g}{L^2}\sin(\theta)-\frac{\alpha_t}{m_t L^2}\dot{x}_t\cos(\theta)+\frac{2\dot{L}}{L^2}\dot{\theta} & -\frac{2}{L}\dot{\theta} \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & \frac{\alpha_t}{m_t}\sin(\theta) & -g\sin(\theta)+\frac{\alpha_t}{m_t}\dot{x}_t\cos(\theta) & 2L\dot{\theta} & \dot{\theta}^2 & -\frac{\alpha_L}{m_L} \end{bmatrix} * \begin{bmatrix} 0 \\ \frac{1}{m_t}\sin(\theta) \\ 0 \\ -\frac{1}{m_t L}\sin(\theta)\cos(\theta) \\ 0 \\ -\frac{1}{m_L}-\frac{1}{m_t}\sin^2(\theta) \end{bmatrix}$$

$$- \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{m_t}\cos(\theta) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{m_t L}\cos(2\theta) & 0 & \frac{1}{m_t L^2}\frac{1}{2}\sin(2\theta) & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{m_t}\sin(2\theta) & 0 & 0 & 0 \end{bmatrix} * \begin{bmatrix} \dot{x}_t \\ -\frac{\alpha_t}{m_t}\dot{x}_t \\ \dot{\theta} \\ -\frac{g}{L}\sin(\theta)+\frac{\alpha_t}{m_t L}\dot{x}_t\cos(\theta)-\left(2\frac{\dot{L}}{L}+\frac{\alpha_\theta}{m_L}\right)\dot{\theta} \\ \dot{L} \\ g\cos(\theta)+L\dot{\theta}^2+\frac{\alpha_t}{m_t}\dot{x}_t\sin(\theta)-\frac{\alpha_L}{m_L}\dot{L} \end{bmatrix}$$

$$[g_2 f] = \begin{bmatrix} \frac{1}{m_t}\sin(\theta) \\ -\frac{\alpha_t}{m_t^2}\sin(\theta) \\ -\frac{1}{m_t L}\frac{1}{2}\sin(2\theta) \\ -\frac{1}{m_t L}\left(2\frac{\dot{L}}{L}+\frac{\alpha_\theta}{m_L}+\frac{\alpha_t}{m_t}\right)\frac{1}{2}\sin(2\theta)+\frac{2}{L}\dot{\theta}\left(\frac{1}{m_L}+\frac{1}{m_t}\sin^2(\theta)\right) \\ -\left(\frac{1}{m_L}+\frac{1}{m_t}\sin^2(\theta)\right) \\ \frac{\alpha_t}{m_t^2}\sin^2(\theta)-\frac{\dot{\theta}}{m_t}\sin(2\theta)+\frac{\alpha_L}{m_L}\left(\frac{1}{m_L}+\frac{1}{m_t}\sin^2(\theta)\right) \end{bmatrix} - \begin{bmatrix} 0 \\ \frac{1}{m_t}\dot{\theta}\cos(\theta) \\ 0 \\ -\frac{\dot{\theta}}{m_t L}\cos(2\theta)+\frac{\dot{L}}{m_t L^2}\frac{1}{2}\sin(2\theta) \\ 0 \\ -\frac{\dot{\theta}}{m_t}\sin(2\theta) \end{bmatrix} \tag{A5.7}$$

$$[g_2 f] = \begin{bmatrix} \frac{1}{m_t}\sin(\theta) \\ -\frac{1}{m_t}\left(\frac{\alpha_t}{m_t}\sin(\theta)+\dot{\theta}\cos(\theta)\right) \\ -\frac{1}{m_t L}\frac{1}{2}\sin(2\theta) \\ \frac{1}{m_t L}\left(\frac{\dot{L}}{L}+\frac{\alpha_\theta}{m_L}+\frac{\alpha_t}{m_t}\right)\frac{1}{2}\sin(\theta)+\frac{1}{L}\dot{\theta}\left(\frac{1}{m_t}+\frac{2}{m_L}\right) \\ -\left(\frac{1}{m_L}+\frac{1}{m_t}\sin^2(\theta)\right) \\ \frac{\alpha_t}{m_t^2}\sin^2(\theta)+\frac{\alpha_L}{m_L}\left(\frac{1}{m_L}+\frac{1}{m_t}\sin^2(\theta)\right) \end{bmatrix} \tag{A5.8}$$

$$[[g1,f]f] = \frac{\partial[g_1 f](x)}{\partial x}[g_1 f](x) - \frac{\partial[g_1 f](x)}{\partial x}f(x) =$$

$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & -\frac{\alpha_t}{m_t} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & \frac{\alpha_t}{m_t L}\cos(\theta) & -\frac{g}{L}\cos(\theta)-\frac{\alpha_t}{m_t L}\dot{x}_t\sin(\theta) & -(2\frac{\dot{L}}{L}+\frac{\alpha_\theta}{m_L}) & \frac{g}{L^2}\sin(\theta)-\frac{\alpha_t}{m_t L^2}\dot{x}_t\cos(\theta)+\frac{2\dot{L}}{L^2}\dot{\theta} & -\frac{2}{L}\dot{\theta} \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & \frac{\alpha_t}{m_t}\sin(\theta) & -g\sin(\theta)+\frac{\alpha_t}{m_t}\dot{x}_t\cos(\theta) & 2L\dot{\theta} & \dot{\theta}^2 & -\frac{\alpha_L}{m_L} \end{bmatrix} * \begin{bmatrix} \frac{1}{m_t} \\ -\frac{\alpha_t}{m_t^2} \\ -\frac{1}{m_t L}\cos(\theta) \\ \frac{1}{m_t L}\left(\frac{\dot{L}}{L}+\frac{\alpha_\theta}{m_L}+\frac{\alpha_t}{m_t}\right)\cos(\theta)+\frac{1}{m_t L}\dot{\theta}\sin(\theta) \\ -\frac{1}{m_t}\sin(\theta) \\ \frac{1}{m_t}\left(\frac{\alpha_t}{m_t}+\frac{\alpha_L}{m_L}\right)\sin(\theta)-\frac{1}{m_t}\dot{\theta}\cos(\theta) \end{bmatrix}$$

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{m_t L}\sin(\theta) & 0 & \frac{1}{m_t L^2}\cos(\theta) & 0 \\ 0 & 0 & -\frac{1}{m_t L}\left(\frac{\dot{L}}{L}+\frac{\alpha_\theta}{m_L}+\frac{\alpha_t}{m_t}\right)\sin(\theta)+\frac{1}{m_t L}\cos(\theta) & \frac{1}{m_t L}\sin(\theta) & -\frac{1}{m_t L^2}\left(\frac{2\dot{L}}{L}+\frac{\alpha_\theta}{m_L}+\frac{\alpha_t}{m_t}\right)\cos(\theta)-\frac{1}{m_t L^2}\dot{\theta}\sin(\theta) & \frac{1}{m_t L^2}\cos(\theta) \\ 0 & 0 & -\frac{1}{m_t}\cos(\theta) & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{m_t}\left(\frac{\alpha_t}{m_t}+\frac{\alpha_L}{m_L}\right)\cos(\theta)+\frac{1}{m_t}\dot{\theta}\sin(\theta) & -\frac{1}{m_t}\cos(\theta) & 0 & 0 \end{bmatrix} * \tag{A5.9}$$

$$\begin{bmatrix} \dot{x}_t \\ -\frac{\alpha_t}{m_t}\dot{x}_t \\ \dot{\theta} \\ -\frac{g}{L}\sin(\theta)+\frac{\alpha_t}{m_t L}\dot{x}_t\cos(\theta)-(2\frac{\dot{L}}{L}+\frac{\alpha_\theta}{m_L})\dot{\theta} \\ \dot{L} \\ g\cos(\theta)+L\dot{\theta}^2+\frac{\alpha_t}{m_t}\dot{x}_t\sin(\theta)-\frac{\alpha_L}{m_L}\dot{L} \end{bmatrix}$$

$$\begin{bmatrix} -\frac{\alpha_t}{m_t^2} \\ \frac{\alpha_t^2}{m_t^3} \\ \frac{1}{m_t L}\left(\frac{\alpha_\theta}{m_L}+\frac{\alpha_t}{m_t}\right)\cos(\theta) \\ \frac{\alpha_t^2}{m_t^3 L}\cos(\theta)-\frac{\alpha_\theta}{m_t m_L L}\left(\frac{\alpha_\theta}{m_L}+\frac{\alpha_t}{m_t}\right)-\frac{\dot\theta}{m_t L}\left(\frac{\alpha_t}{m_t}+2\frac{\alpha_L}{m_L}\right)\sin(\theta) \\ \frac{1}{m_t}\left(\frac{\alpha_t}{m_t}+\frac{\alpha_L}{m_L}\right)\sin(\theta) \\ -\frac{\alpha_t^2}{m_t^3}\sin(\theta)+\frac{\dot\theta}{m_t}\left(\frac{\alpha_\theta}{m_L}+\frac{\alpha_t}{m_t}\right)\cos(\theta)-\frac{1}{m_t}\left(\frac{\alpha_t\alpha_L}{m_t m_L}+\frac{\alpha_L^2}{m_L^2}\right)\sin(\theta) \end{bmatrix} \tag{A5.10}$$

$$[[g\,f]f] = \frac{\partial f(x)}{\partial x}[g,f](x) - \frac{\partial [g,f](x)}{\partial x}f(x) =$$

$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & -\frac{\alpha_t}{m_t} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & \frac{\alpha_t}{m_t L}\cos(\theta) & -\frac{g}{L}\cos(\theta)-\frac{\alpha_t}{m_t L}\dot x_t\sin(\theta) & -(2\frac{\dot L}{L}+\frac{\alpha_\theta}{m_L}) & \frac{g}{L^2}\sin(\theta)-\frac{\alpha_t}{m_t L^2}\dot x_t\cos(\theta)+\frac{2\dot L}{L^2}\dot\theta & -\frac{2}{L}\dot\theta \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & \frac{\alpha_t}{m_t}\sin(\theta) & -g\sin(\theta)+\frac{\alpha_t}{m_t}\dot x_t\cos(\theta) & 2L\dot\theta & \dot\theta^2 & -\frac{\alpha_L}{m_L} \end{bmatrix} * \begin{bmatrix} \frac{1}{m_t}\sin(\theta) \\ -\frac{1}{m_t}\left(\frac{\alpha_t}{m_t}\sin(\theta)+\dot\theta\cos(\theta)\right) \\ -\frac{1}{m_t L}\frac{1}{2}\sin(2\theta) \\ \frac{1}{m_t L}\left(\frac{\dot L}{L}+\frac{\alpha_\theta}{m_L}+\frac{\alpha_t}{m_t}\right)\frac{1}{2}\sin(2\theta)+\frac{\dot\theta}{L}\left(\frac{1}{m_t}+\frac{2}{m_L}\right) \\ -1\left(\frac{1}{m_L}+\frac{1}{m_t}\sin^2(\theta)\right) \\ \frac{\alpha_t}{m_t^2}\sin^2(\theta)+\frac{\alpha_L}{m_L}\left(\frac{1}{m_L}+\frac{1}{m_t}\sin^2(\theta)\right) \end{bmatrix}$$

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{m_t}\cos(\theta) & 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{m_t}\left(\frac{\alpha_t}{m_t}\cos(\theta)-\dot\theta\sin(\theta)\right) & -\frac{1}{m_t}\cos(\theta) & 0 & 0 \\ 0 & 0 & -\frac{1}{m_t L}\cos(2\theta) & 0 & \frac{1}{m_t L^2}\frac{1}{2}\sin(2\theta) & 0 \\ 0 & 0 & \frac{1}{m_t L}\left(\frac{\dot L}{L}+\frac{\alpha_\theta}{m_L}+\frac{\alpha_t}{m_t}\right)\cos(2\theta)-\frac{1}{L}\left(\frac{1}{m_t}+\frac{2}{m_L}\right) & -\frac{1}{m_t L}\left(2\frac{\dot L}{L}+\frac{\alpha_\theta}{m_L}+\frac{\alpha_t}{m_t}\right)\frac{1}{2}\sin(2\theta)-\frac{\dot\theta}{L^2}\left(\frac{1}{m_t}+\frac{2}{m_L}\right) & \frac{1}{m_t L^2}\frac{1}{2}\sin(2\theta) \\ 0 & 0 & -\frac{1}{m_t}\sin(2\theta) & 0 & 0 & 0 \\ 0 & 0 & \frac{\alpha_t}{m_t^2}\sin(2\theta) & 0 & 0 & 0 \end{bmatrix} * \tag{A5.11}$$

$$\begin{bmatrix} \dot x_t \\ -\frac{\alpha_t}{m_t}\dot x_t \\ \dot\theta \\ -\frac{g}{L}\sin(\theta)+\frac{\alpha_t}{m_t L}\dot x_t\cos(\theta)-(2\frac{\dot L}{L}+\frac{\alpha_\theta}{m_L})\dot\theta \\ \dot L \\ g\cos(\theta)+L\dot\theta^2+\frac{\alpha_t}{m_t}\dot x_t\sin(\theta)-\frac{\alpha_L}{m_L}\dot L \end{bmatrix}$$

$$[[g_2 f]f] = \begin{bmatrix} -\frac{\alpha_t}{m_t^2}\sin\theta \\ -\frac{\alpha_t^2}{m_t^3}\sin(\theta)+\frac{\dot\theta^2}{m_t}\sin(\theta)+\frac{g}{m_t L}\frac{1}{2}\sin(2\theta)-\frac{\alpha_t \dot x_t}{m_t^2 L}\cos^2(\theta)+\frac{\dot\theta}{m_t}\left(2\frac{\dot L}{L}+\frac{\alpha_\theta}{m_L}\right)\cos(\theta) \\ \frac{1}{m_t L}\left(\frac{\alpha_\theta}{m_L}+\frac{\alpha_t}{m_t}\right)\frac{1}{2}\sin(2\theta)+\frac{\dot\theta}{L}\left(\frac{1}{m_t}+\frac{2}{m_L}\right)+\frac{\dot\theta}{m_t L}\cos(2\theta) \\ \left(\frac{\alpha_t^2}{m_t^3 L}+\frac{\alpha_\theta^2}{m_t m_L^2 L}+\frac{\alpha_\theta \alpha_t}{m_t^2 m_L L}\right)-\frac{\dot\theta}{m_t}\left(\frac{\alpha_t}{m_t m_L}+\frac{\alpha_\theta \alpha_t}{m_t^3}\right)\sin(2\theta) \\ \frac{\alpha_t}{m_t m_L}-\frac{\dot\theta}{m_t}\sin(2\theta) \\ \frac{\dot\theta}{m_t}\left(\frac{\alpha_\theta}{m_L}+\frac{\alpha_t}{m_t}\right)\sin(\theta)+\frac{g}{m_t L}\frac{1}{2}\sin(2\theta)\sin(\theta)-\frac{\alpha_t \dot x_t}{m_t^2 L}\frac{1}{2}\sin(2\theta)\cos(\theta)-\left(\frac{\alpha_L \alpha_t}{m_L m_t^2}+\frac{\alpha_L^2}{m_L^2 m_t}\right)\sin^2(\theta) \end{bmatrix} \tag{A5.12}$$

Now we have to check that the rank of the repeated Lee-bracket is 6, for example for:

$$\begin{array}{c} x_t \in [-1,1] \\ \dot x_t \in [-1,1] \\ \theta \in [-0.5,0.5] \\ \dot\theta \in [-0.5,0.5] \\ L \in [0.1,2] \\ \dot L \in [-1,1] \end{array} \tag{A5.13}$$

This is checked for some points.

Appendix 6. Implementation nonlinear model in simulink.

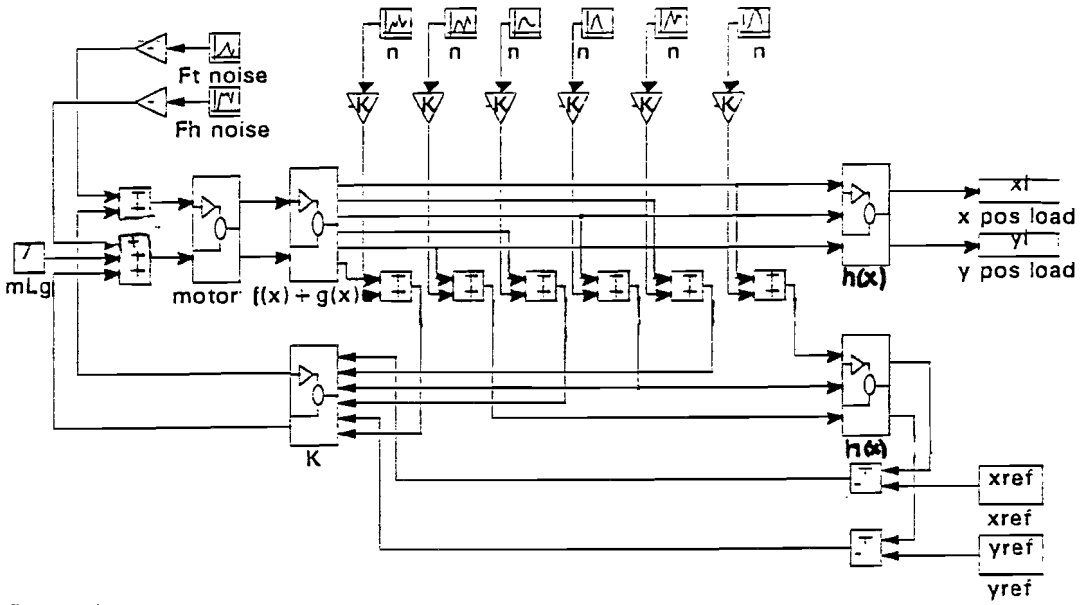


figure A6.1 Total system with sensor and actuator noise and motor nonlinearies.

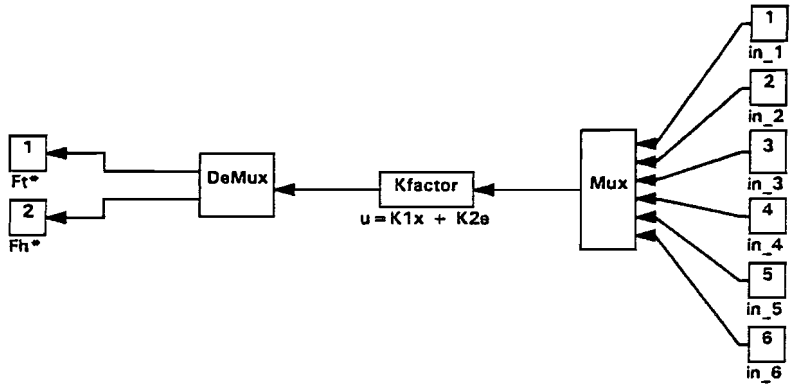


figure A6.2 Block: K = static state controller.

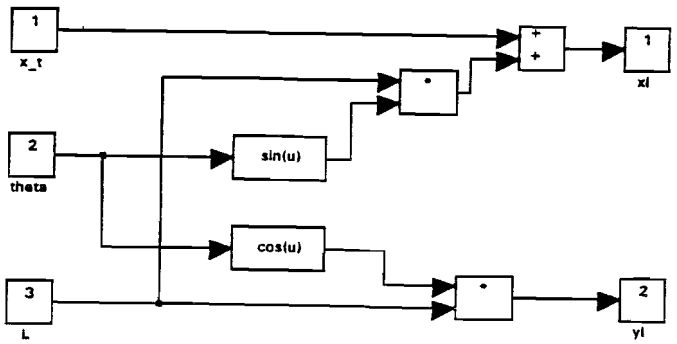


figure A6.3 Block: h(x) = transformation from states to output y.

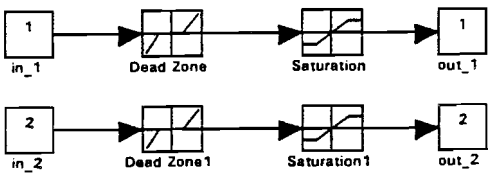


figure A6.4 Block: motor = motor nonlinearities.

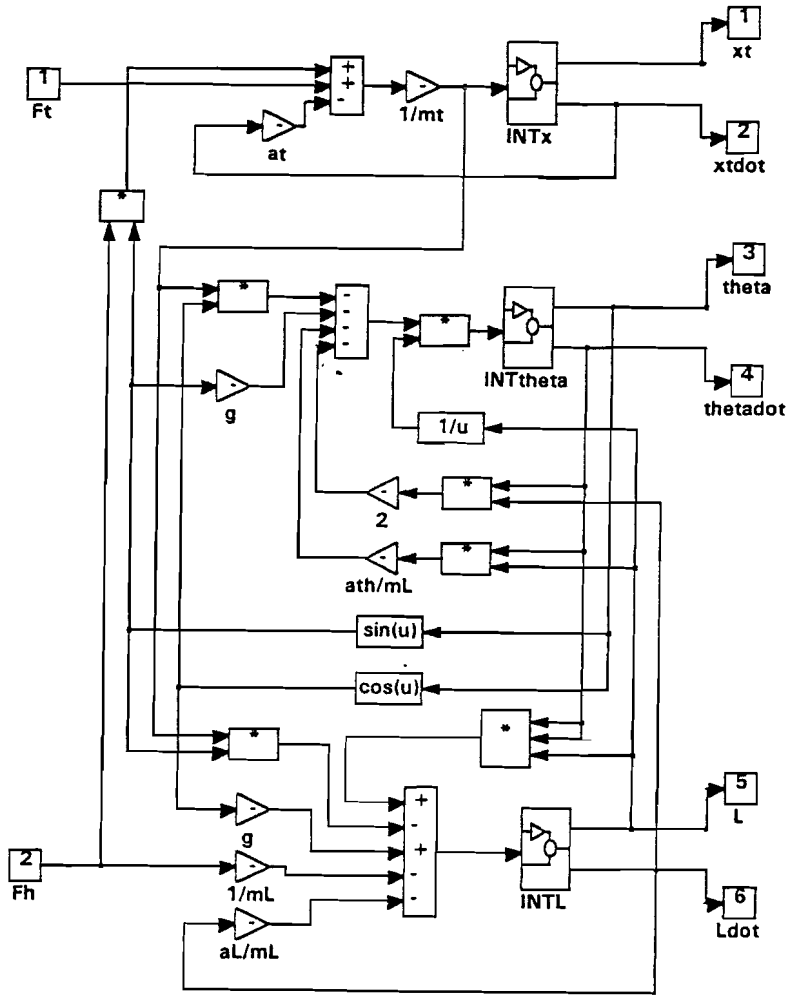


figure A6.5 Block: f(x)+g(x)u = nonlinear dynamics.

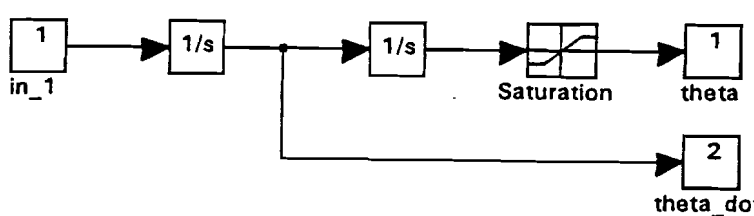


figure A6.6 Block: INT... = integration part with saturation.