Modern Control Theory

Ad Damen

course: (5N050)
Measurement and Control Group
Department of Electrical Engineering
Eindhoven University of Technology
P.O.Box 513
5600 MB Eindhoven

Draft version of October 11, 2002

Contents

Chapter 1

Analysis of nonlinear systems

1.1 Introduction

The analysis and design of control systems in the basic courses was confined to linear, SISO (single input single output) systems either in continuous or in discrete time. For such systems the superposition principle holds. For nonlinear systems this is not the case by definition and unfortunately (or should we say fortunately ?) most physical systems show nonlinear dynamics. Linearisation about a working point can help here, provided the signals are bounded to a range where the linearisation leads to a sufficiently accurate description. We will discuss this later in sections 1.3 and 1.4 when we study particular systems that show only smooth nonlinearities, i.e. the involved nonlinear static functions and their derivatives are continuous. This smoothness is not present in very frequent and familiar effects like saturation, dead zone, static friction and backlash. These nonlinearities, inherent to the process to be controlled, are certainly not wanted but one also deliberately introduces discontinuous nonlinearities in controllers like switches and relays or nonlinearities are undesired side effects like the limited resolution of AD- and DA-converters. One should not think that the use of relays and switches is outdated as these controllers are very cheap and robust and still ubiquitary in e.g. controlled heat systems in the household environment like central heating, airconditioning, irons, refrigerators, freezers, washing machines, dish washers, hairfohns, boilers etc. However, the drawback of these controllers is that they always lead to small oscillations around the wished equilibrium as we will see and indicate as 'limit cycle'. For all these discontinuous nonlinearities linearisation is not the proper way for analysis because, whatever small the signals are, a linear description right in the discontinuities is fundamentally impossible. What seems to remain is to analyse the system in the time domain in piecewise continuous subdomains bounded by the constraints of the discontinuities and trespassing the constraints with the use of boundary values for the states. Done by hand, this rapidly appears to become a meticulous job but computer simulations offer a solution, though to be treated with care and suspicion. We will discuss the simulation later in section 1.4. First we introduce the method of the "describing function" as it represents a basically simple tool leading to the insight why or why not such a system will oscillate and what the approximate frequency and amplitude for a possible (stable) oscillation will be.

1.2 Describing functions

1.2.1 Theory

The method under this name is also indicated by "equivalent linearisation method" or "harmonic balance" and aims at analysing the system behaviour in the frequency domain. This might seem a strange thing to do as nonlinear system behaviour is hard to describe in the frequency domain because higher harmonics appear depending on the type and amplitude of the input signal. It is precisely by ignoring these higher harmonics for sinusoidal inputs that we sort of "linearise" the system whereby we keep the amplitude of the input sine wave as a parameter to analyse stability properties. The principle can very simply be elucidated by Fig. 1.1.

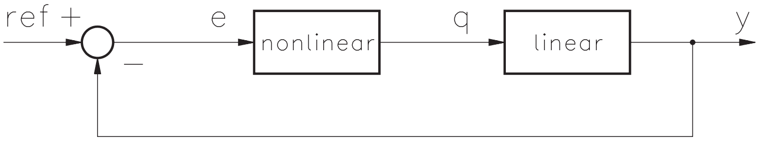


Figure 1.1: Block scheme describing function method.

It is supposed that the total process can be represented by a cascade of a nonlinear and a linear part. The nonlinear part should be such that a sinewave on the input e leads to a periodic signal q of the same period where the form depends on the amplitude of the incoming sinewave but not on its frequency. Examples of such nonlinearities can be found in Fig. 1.2 which depicts the library of nonlinear blocks in Simulink.

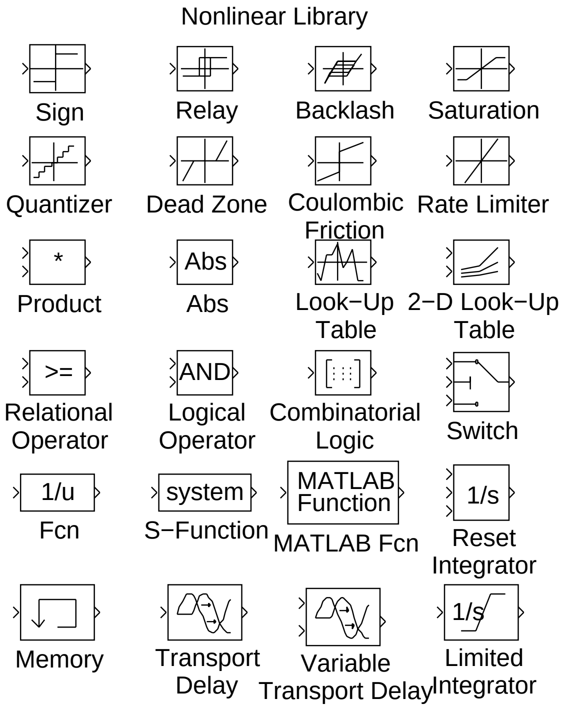


Figure 1.2: Nonlinear blocks in Simulink.

It is easy to conclude for the nonlinear functions in the first two rows that excitation by a sine wave will yield a periodic signal of the same period where the form is only dependent on the amplitude of the incoming wave and not influenced by the frequency, so:

$$\begin{cases} e = \hat{e}\sin(\omega t) \\ q = \hat{q}(\hat{e})\sin(\omega t + \phi(\hat{e})) + \text{higher harmonics} \end{cases} \tag{1.1}$$

The same holds for the nonlinear functions in the third row defined by the look-up table or an explicit function of the input temporal value. The other blocks can be used to compose more complicated nonlinearities as we will show in this section. In that case the fulfilment of the mentioned assumption should be watched. This crucial assumption, put in other words, is that the nonlinear block only effects a possible delay (represented by a phase shift ϕ) and adds higher harmonics to an incoming sinusoidal signal depending on the amplitude but **not on the frequency** of this sine.

Next the linear block is supposed to be stable and sufficiently low pass so that all higher harmonics are effectively filtered out. This "linearisation" allows us to analyse the system behaviour for each frequency by considering the "describing function" of the nonlinear block. The describing function is defined as:

$$f(\hat{e}) = \frac{\hat{q}(\hat{e})e^{j\phi(\hat{e})}}{\hat{e}} = \frac{a_1(\hat{e}) + jb_1(\hat{e})}{\hat{e}} \tag{1.2}$$

where $\hat{e}$ is the amplitude of the input sinusoidal signal and a_1 and b_1 are the Fourier coefficients of the fundamental frequency ω, so:

$$q(t) = \Sigma_{k=0}^{\infty}(a_k \sin(k\omega t) + b_k \cos(k\omega t)) \tag{1.3}$$

$$a_k = \frac{1}{\pi}\int_{\gamma}^{2\pi+\gamma} q(\omega t)\sin(k\omega t)d\omega t \tag{1.4}$$

$$b_k = \frac{1}{\pi}\int_{\gamma}^{2\pi+\gamma} q(\omega t)\cos(k\omega t)d\omega t \tag{1.5}$$

So the describing function is just the ratio of the outgoing to the incoming fundamental (harmonic) in complex plane. It is, so to say, the frequency independent, complex amplification by the nonlinearity of only the fundamental frequency. Note that the amplitude of the outgoing fundamental $\hat{q}$ is a function of the amplitude $\hat{e}$ of the incoming sinewave. By neglecting the higher harmonics, the closed loop transfer function for the system of Fig.1.1 is then approximated by:

$$M(j\omega) = \frac{f(\hat{e})H(j\omega)}{1 + f(\hat{e})H(j\omega)} \tag{1.6}$$

where $H(j\omega)$ is the stable, low pass transfer function of the linear block. Again the characteristic equation:

$$1 + f(\hat{e})H(j\omega) = 0 \tag{1.7}$$

indicates stability. In the complex plane we can draw the polar plots of $f(\hat{e})H(j\omega)$ and study the situation with respect to the point -1. The function $f(\hat{e})H(j\omega)$ depends on both the frequency and the amplitude $\hat{e}$. So we get a set of curves each defined by a certain value of the amplitude $\hat{e}$ as shown in Fig.1.3.

In case a particular curve passes through the point -1 we may compute the frequency and the amplitude of the oscillation from $f(\hat{e})H(j\omega) = -1$. The computation and the

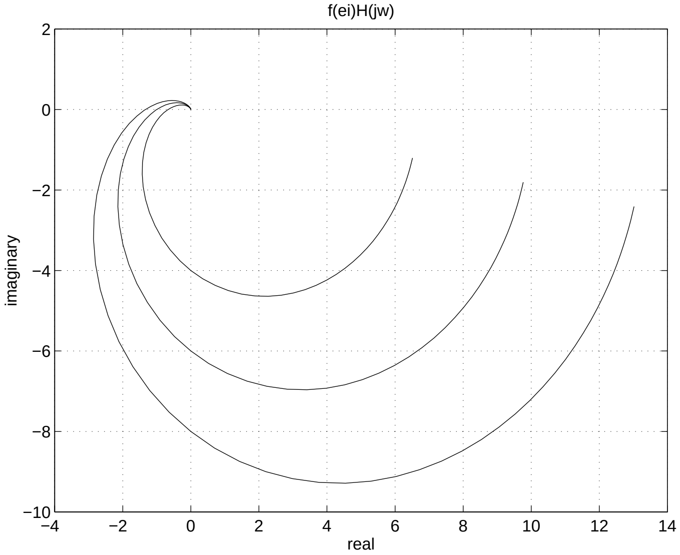


Figure 1.3: Nyquist plots for various $\hat{e}_i$.

drawing of all these curves is quite cumbersome. A simpler method is to consider that, if $f(\hat{e}) \neq 0$, the stability is also governed by:

$$\frac{1}{f(\hat{e})} + H(j\omega) = 0 \qquad \Rightarrow \qquad H(j\omega) = \frac{-1}{f(\hat{e})} \tag{1.8}$$

We then actually replace the point -1 by $-1/f(\hat{e})$ and we just have to draw two curves i.e. $H(j\omega)$ and $-1/f(\hat{e})$. The part of the curve $-1/f(\hat{e})$ encompassed by the $H(j\omega)$ Nyquist curve indicates the amplitudes $\hat{e}$ for which the system is unstable and vice versa. The intersection of both curves defines the possible oscillation on which the system behaviour will stay. Examples will clarify this simple analysis.

1.2.2 Example0: Saturation

[1]

Let us analyse the effects of a nonlinearity in the form of a frequently occurring saturation. Fig. 1.4 illustrates how a sinusoidal input with sufficiently large amplitude will be distorted leading to higher harmonics.

For the describing function we need only to compute a_1 and b_1. From symmetry we can immediately conclude that all coefficients b_i are zero so that there is no phase shift and the describing function will be real while:

$$\begin{cases} \forall \hat{e} \geq e_s: & a_1 = \frac{1}{\pi}\int_0^{2\pi} q(\omega t)\sin(\omega t)d\omega t = \frac{2\hat{e}K}{\pi}\{\arcsin\left(\frac{e_s}{\hat{e}}\right) + \frac{e_s}{\hat{e}}\sqrt{1-\left(\frac{e_s}{\hat{e}}\right)^2}\} \\ \forall \hat{e} \leq e_s: & a_1 = K\hat{e} \end{cases} \tag{1.9}$$

with $K = \tan(\alpha)$. So the describing function is given by $f(\hat{e}) = a_1/\hat{e}$ which is real. In Fig. 1.5 $f(\hat{e})$ has been plotted for $K = 1$ and $e_s = .5$.

Note that, for $\hat{e} > .5$, $f(\hat{e})$ is always less than K. Finally in Fig. 1.6 we have displayed both $-1/f(\hat{e})$ and $H(j\omega)$ which was chosen to be $100/((j\omega+1)(j\omega+2)(j\omega+3))$.

[1] ”Examplei” refers to a Simulink file examplei.m representing example i. The Simulink files can be obtained from ftp.nt01.er.ele.tue.nl/ModReg

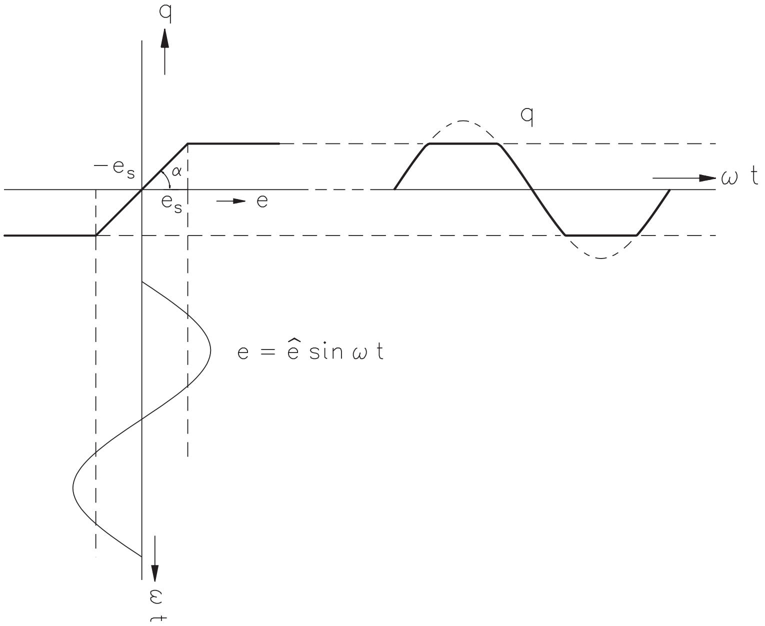


Figure 1.4: Distortion by saturation.

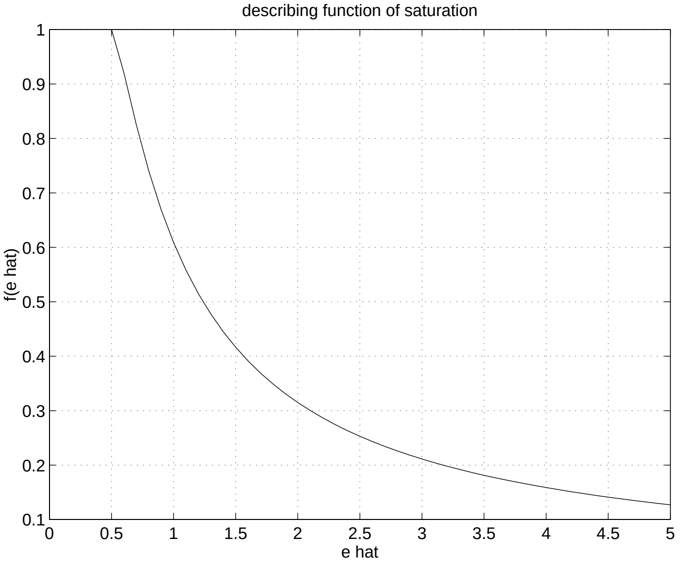


Figure 1.5: Describing function of saturation.

Because $f(\hat{e})$ is a real positive function $-1/f(\hat{e})$ follows the negative real axis from the point $-1/K$. This point $-1/K$ simply represents the usual point -1 before saturation occurs as we extracted the "gain" K from the loop transfer. So we observe that, also in the linear situation for small amplitudes $\hat{e}$ before saturation takes place, the closed loop system is unstable because $-1/K$ is contained in the Nyquist loop. Consequently, the system will show unstable behaviour starting on any small initial value or disturbance. This instability will soon lead to increasing $\hat{e}$ and saturation so that we proceed on the curve $-1/f(\hat{e})$ in the polar plane from $-1/K$ to the left until at a sufficiently large $\hat{e}$ the crossing with $H(j\omega)$ is met. As soon as an increasing $\hat{e}$ effects a trespassing of this point the system becomes stable because the loop transfer $H(j\omega)$ no longer encompasses the

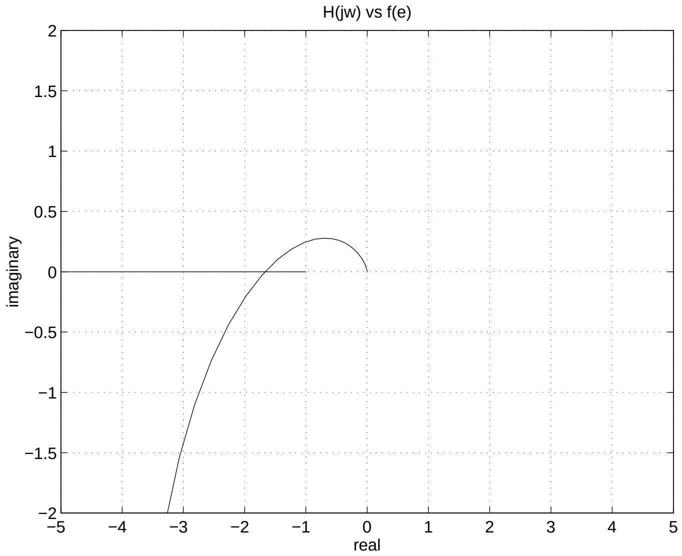


Figure 1.6: $H(j\omega)$ versus $-1/f(\hat{e})$.

point $-1/f(\hat{e})$. Oscillation will then starve but immediately be restarted as soon as the loop transfer $H(j\omega)$ is trespassed from the outer side again due to the decrease of $\hat{e}$. So we must conclude that the oscillation is stabilised in amplitude by this effect precisely on the crossing point as the increasing $\hat{e}$ causes a trespassing from the inside to the outside. If we happened to have a very large initial value, the $\hat{e}$ would certainly lead to saturation but the system would be stable for that amplitude. The $\hat{e}$ would decrease so that the crossing with $H(j\omega)$ from the outside would start the oscillation again and as before the ultimate effect is again that the system will remain in the crossing point. The oscillation frequency can easily be obtained from

$$\mathrm{Im}(H(j\omega_0)) = 0 \quad \Rightarrow \quad \omega_0 = \sqrt{11} \quad \Rightarrow \quad f_0 = \frac{\sqrt{11}}{2\pi} = .5278Hz \tag{1.10}$$

For this frequency $H(j\omega_0) = -5/3 = -1/f(\hat{e}) = -\hat{e}/a_1$ from which we can compute $\hat{e}$. By defining $p = e_s/\hat{e}$ we find:

$$p = \sin(\frac{3\pi}{10} - p\sqrt{1-p^2}) \tag{1.11}$$

We try to solve this iteratively by the Picard-algorithm (see also section 1.3.4):

$$p(k+1) = \sin(.3\pi - p(k)\sqrt{1-p(k)^2}) \tag{1.12}$$

Starting with $p(0) = 1$ and executing 100 iterations (easily done in Matlab) leads to $p = .49186$ so that $\hat{e} = 1.0165$.

In Fig. 1.7 a representation in Simulink is shown.

A state space representation is used in order to be able to easily define initial values of the system for starting the oscillation. The state equations are given by:

$$\dot{x} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -6 & -11 & -6 \end{pmatrix} x + \begin{pmatrix} 0 \\ 0 \\ 100 \end{pmatrix} q \tag{1.13}$$

$$y = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix} x \tag{1.14}$$

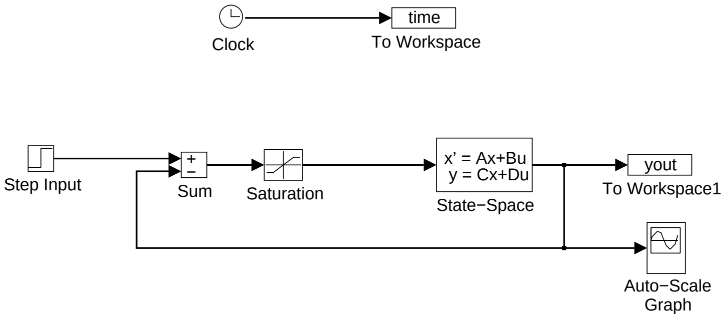


Figure 1.7: Configuration of example0.

Fig. 1.8 shows the output of the system as a reaction on the initial state $(1,1,1)^T$ while the input step is zero. We note that the values $f = .5278$ Hz and $\hat{e} = 1.0165$ are well predicted.

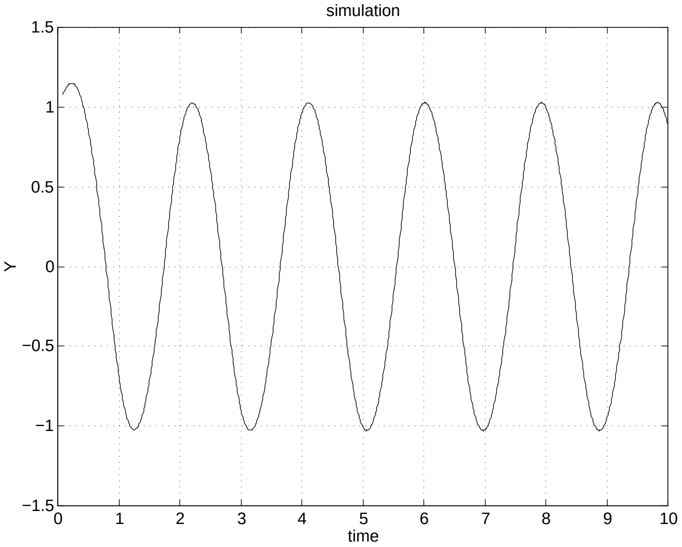


Figure 1.8: Output of example0 for $x(0) = (1,1,1)^T$.

Also the actual reference signal can cause this oscillation when the initial value of the state is zero. A non-sinusoidal reference signal would also contribute to the saturation effect and the superposition property does certainly not hold! Consequently the reference signal itself would partly define the position on the $-1/f(\hat{e})$ curve which completely disrupts our analysis which is based upon single frequency, sinusoidal signals. So we can only allow for very low frequent reference signals that are approximately constant compared to the oscillation frequency. The system will then show oscillation around the reference signal as plotted in Fig. 1.9 for a step input, valued 2, while now the initial states are zero.

Note that the oscillation still has the same amplitude and frequency but the average value is not centered around the step value (=2). The average value does also not correspond to the final value for the linear situation being 200/106 (check this for yourself). As a matter of fact the analysis with describing functions becomes much more complicated now. Any final error e_f (in first "linear" estimate here 2-200/106) effectively "lifts" the

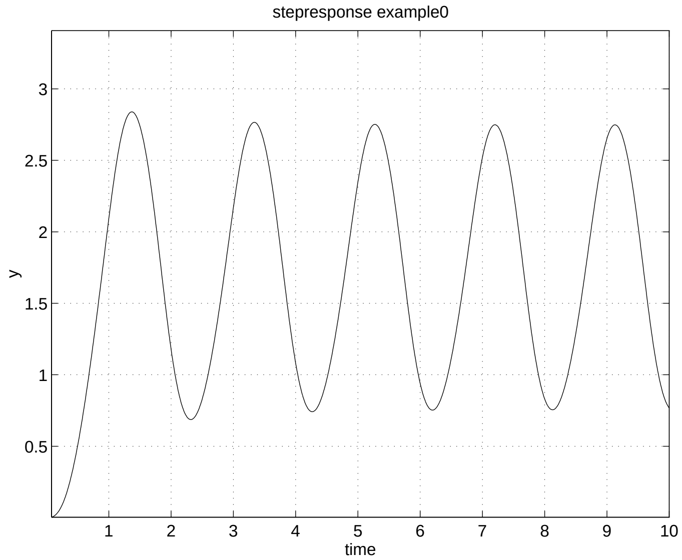


Figure 1.9: Output of example0 for step input of value 2.

incoming sine wave (see Fig. 1.10) and thus disrupts the symmetry of the outgoing curve so that we obtain not only a different a_1 but also a b_1 and b_0 unequal to zero.

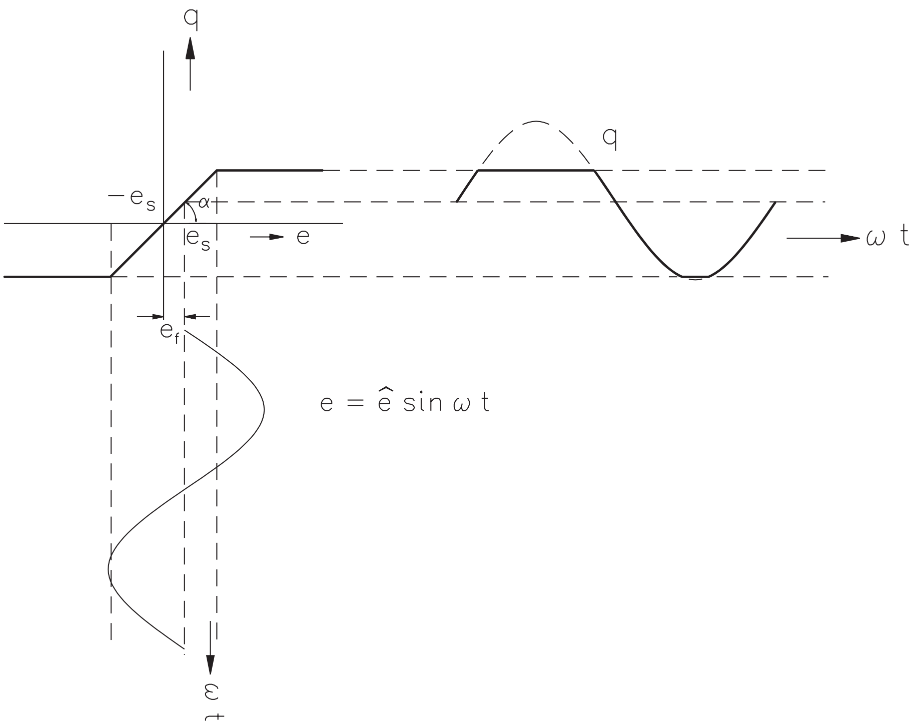


Figure 1.10: Saturation distortion for lifted input.

In turn the $b_0 \neq e_f$ changes the DC-gain that we used to compute the final error e_f so that this should be balanced again (iteratively ?). All this together really complicates the analysis with describing functions and, although it is possible in principle, we prefer to be satisfied with the simulation here. It is clear that superposition does not hold and a more accurate study can better be done by simple simulation.

1.2.3 Example1: Relay with hysteresis

The nonlinearity considered in the previous example does not show any memory, so that there was no phase shift between the input sine and the output fundamental, at least for the reference signal being zero. Consequently the $-1/f(\hat{e})$ curve follows the negative real axis for increasing $\hat{e}$. A more interesting picture occurs when there is some memory in the nonlinear block. This is the case in the next example where we study the effect of relays with hysteresis as illustrated in Fig. 1.11.

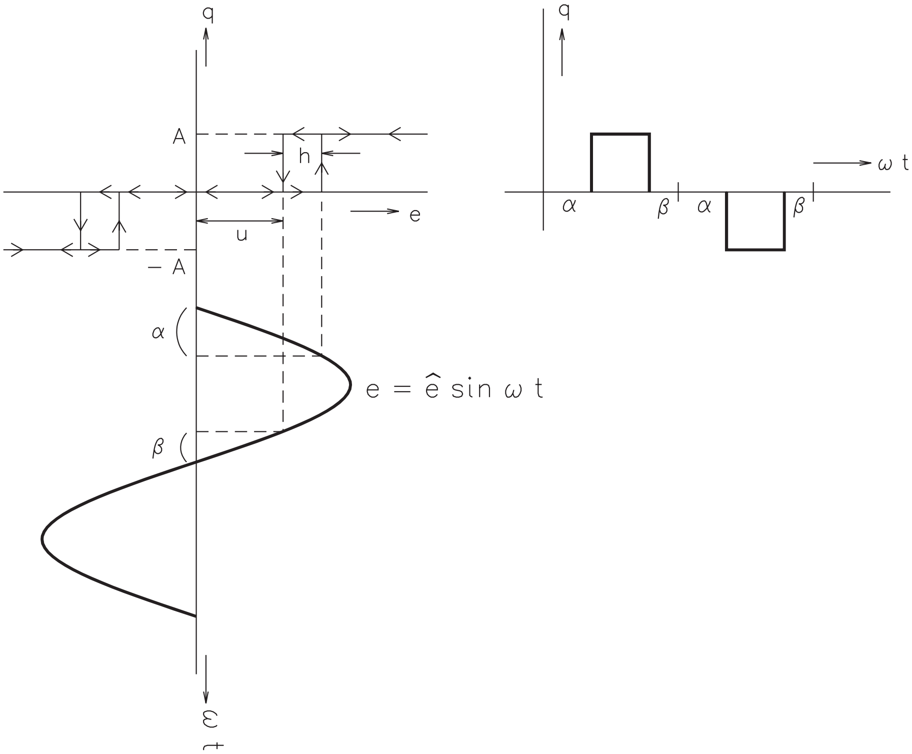


Figure 1.11: Distortion by hysteresis.

It can easily be deduced from this figure that (the fundamental of) the output is no longer "symmetric" with respect to the input: $\alpha \neq \beta$. Ergo, there is a phase shift in the fundamental and the describing function will be complex:

$$\begin{cases} \forall \hat{e} \geq u+h : \\ a_1 = \frac{2A}{\pi}\{\sqrt{1-(\frac{u}{\hat{e}})^2}\} + \sqrt{1-(\frac{u+h}{\hat{e}})^2} \\ b_1 == \frac{2Ah}{\pi \hat{e}} \\ f(\hat{e}) = \frac{a_1+jb_1}{\hat{e}} \\ -\frac{1}{f(\hat{e})} = \frac{-\hat{e}(a_1-jb_1)}{a_1^2+b_1^2} \end{cases} \tag{1.15}$$

In Fig. 1.12 the $-1/f(\hat{e})$ curve has been displayed for $A = 1$, $u = .4$ and $h = .2$ together with the transfer-function of the linear block which has been chosen as

$$H(j\omega) = \frac{K}{j\omega(1 + j\omega\tau)} \tag{1.16}$$

for $K = 100$ and $\tau = .1$.

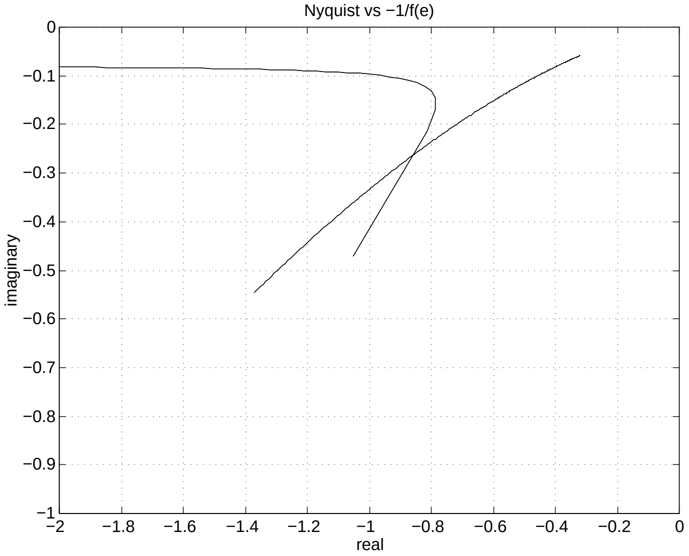


Figure 1.12: Nyquist plane example1.

In this case the amplitude and the frequency of the oscillation are respectively .63 and 5.2 Hz. (you may compute this yourself as an exercise in Matlab). In Fig. 1.13 the simulation of this example1 in Simulink is shown.

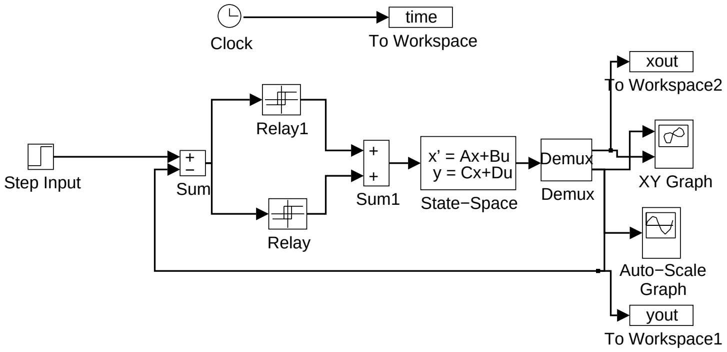


Figure 1.13: Blockscheme example1.

Again we used the state space representation in order to easily introduce initial values:

$$\begin{cases} \dot{x} = \begin{pmatrix} 0 & 1 \\ 0 & -10 \end{pmatrix} x + \begin{pmatrix} 0 \\ 1000 \end{pmatrix} q \\ y = \begin{pmatrix} 1 & 0 \end{pmatrix} x \\ x(0) = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \end{cases} \tag{1.17}$$

Fig. 1.14 presents the measured output. Indeed the oscillation corresponds to the expectations of amplitude and frequency after the transient effects.

Note that the initial value of $x_1(0) = 1$ exceeds the value $u + h = .6$ so that indeed the relay is activated. If it were less than .6 the oscillation would not start (see also section 1.4). So we have a conditional oscillation. Such a conditional oscillation is even more

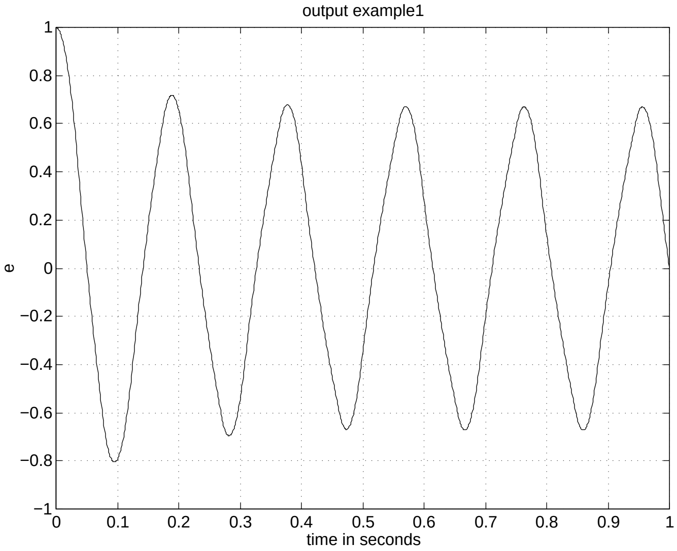


Figure 1.14: Output of example1.

striking if we deal with two intersection points between $H(j\omega)$ and $-1/f(\hat{e})$. This is the case for $H(s) = 200/(s^3 + 2s^2 + 101s)$ and shown in Fig. 1.15.

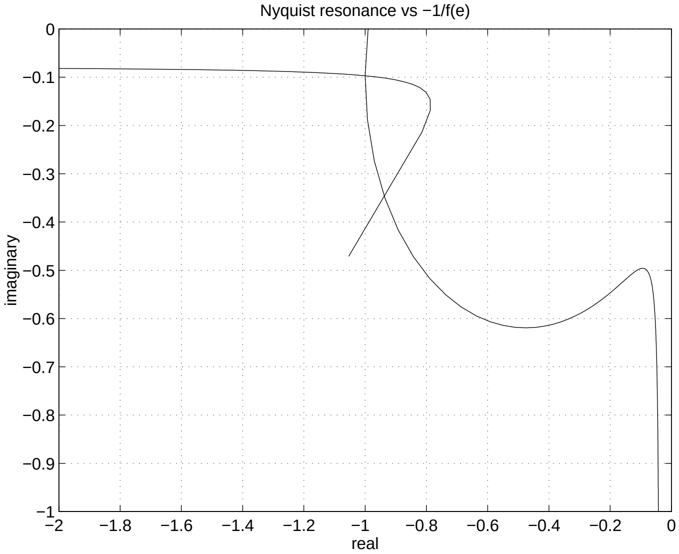


Figure 1.15: Two intersections.

Only the point where increasing $\hat{e}$ leads to an exit of the field encompassed by the Nyquist curve $H(j\omega)$ is the ultimate equilibrium point. The crossing where an increase of $\hat{e}$ causes an entrance of the encompassed field just indicates that only if disturbances or initial values cause a passing of this point the system becomes unstable and follows the $-1/f(\hat{e})$ curve till it reaches the other, "correct" equilibrium point. The system has been simulated according to the same scheme as sketched in Fig. 1.13 but now the state space description is (exampl1A.m):

$$\begin{cases} \dot{x} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -101 & -2 \end{pmatrix} x + \begin{pmatrix} 0 \\ 0 \\ 200 \end{pmatrix} q \\ y = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix} x \end{cases} \tag{1.18}$$

When we feed this system with an initial value $x(0) = (1, 0, 0)^T$ no oscillation will result, as Fig. 1.16 shows, despite of the fact that $y(0)$ is certainly greater than $u+h = .6$.

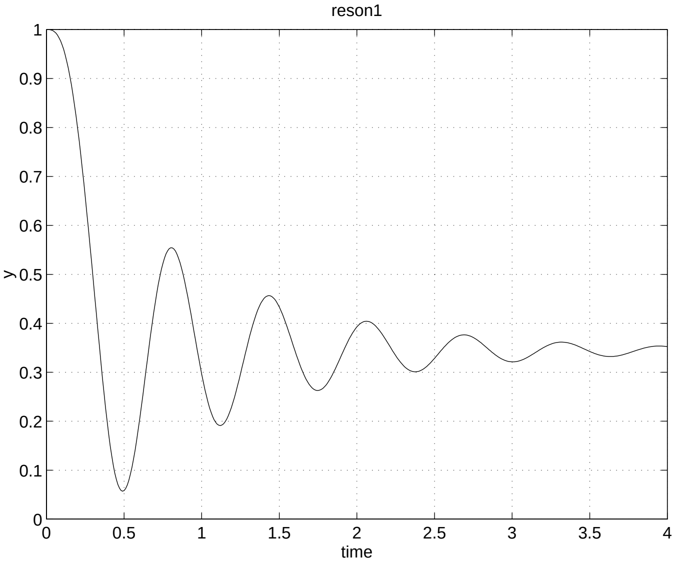


Figure 1.16: Insufficient excitation.

It does not even help increasing the value of $y(0)$, the system will remain stable. This changes dramatically if we excite the system properly i.e. by a proper initial state such that the $\hat{e}$ is sufficiently large so that we indeed arrive on the curve $-1/f(\hat{e})$ inside the Nyquist curve. Then the system starts to oscillate on the second intersection point. This is shown in Fig. 1.17 where initial value $x(0) = (0, 10, -10)^T$ was given.

Such a 'stable' oscillation where the system runs into is generally called a limit cycle and we will discuss this more accurately in section 1.4 about the state space descriptions. Here we only provided a simple method which gives some insight but which is still an approximation. The approximation will only be appropriate if the higher harmonics are sufficiently filtered out by the linear block. This is clearly not the case in the following example.

1.2.4 Example2: Heating system

A simple model of a (central) heating system is outlined in electrical components in Fig. 1.18.

The capacitance represents the heat capacity of a body (iron, room etc). The parallel resistance represents the heat losses from this body to the surroundings with a fixed temperature being the earth potential. Depending on the temperature of the body (voltage) a heat source (current source) of value A is switched on and off to control the heat flow.

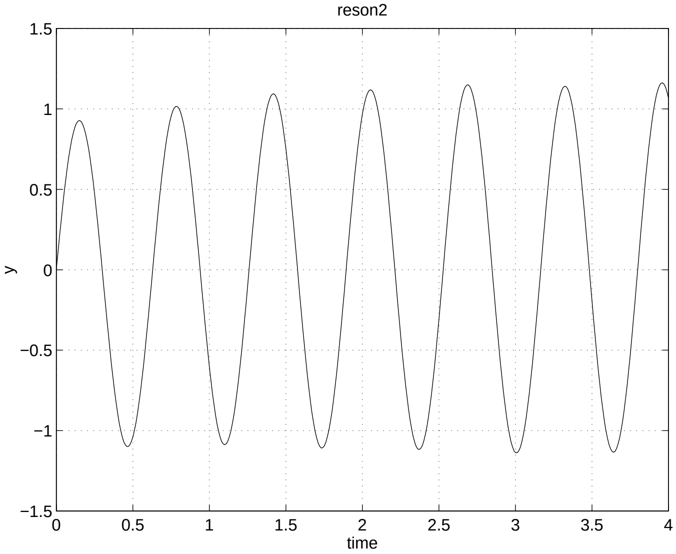


Figure 1.17: Proper excitation.

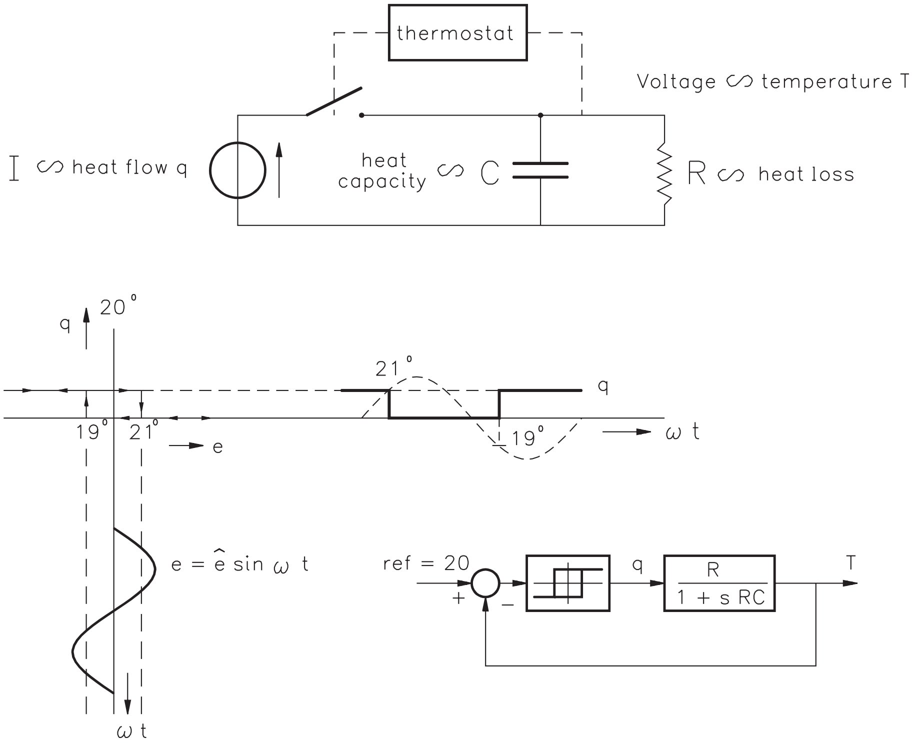


Figure 1.18: Heating system.

The relay switches at values $\pm p$. The $-1/f(\hat{e})$ derivation yields:

$$\forall \hat{e} \geq p: \qquad a_1 = \frac{2A}{\pi}\sqrt{1-\left(\frac{p}{\hat{e}}\right)^2} \qquad b_1 = -\frac{2Ap}{\pi\hat{e}} \tag{1.19}$$

$$\Rightarrow \qquad -\frac{1}{f(\hat{e})} = -\frac{\pi\hat{e}}{2A}\{\sqrt{1-\left(\frac{p}{\hat{e}}\right)^2} + j\left(\frac{p}{\hat{e}}\right)\} \tag{1.20}$$

so that $-1/f(\hat{e})$ lives in the third quadrant as shown in Fig. 1.19 for $A = 1$ and $p = 1^0$.

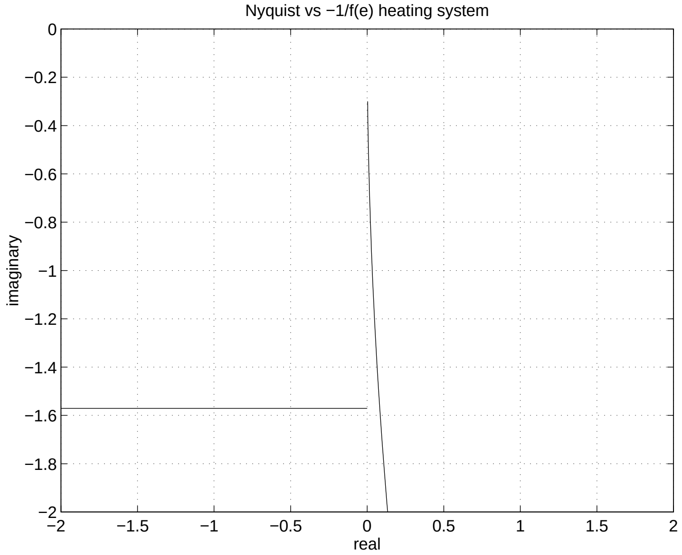


Figure 1.19: Nyquist plot example2.

This Fig. 1.19 also shows the Nyquistplot of a first order linear block that lives completely in the fourth quadrant so that no intersection can be found. Nevertheless such a system certainly shows a limit cycle, as we all know and what the simulation (outlined in Fig. 1.20 in Simulink) indeed produces in Fig. 1.21.

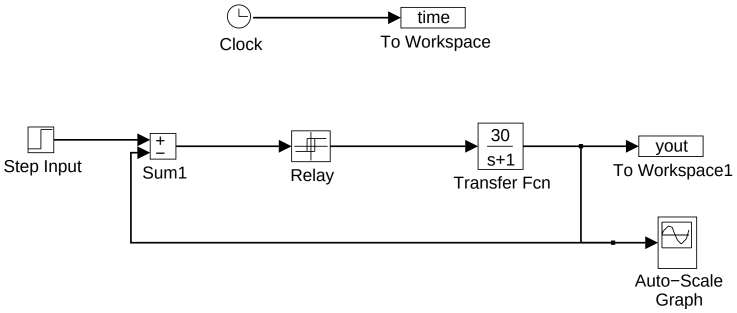


Figure 1.20: Blockscheme example2 heating system.

There we see that the oscillation wave is far from a single sine wave so that the condition for the describing function method, that higher harmonics are sufficiently filtered out, does not hold. If there is more filtering, e.g. in the case of a multi compartmental model (i.e. more "heat"capacities linked with "heat loosing" resistances) this higher order roll-off necessarily causes extra phase shift so that there will be an intersection point. Another problem with the describing function method is the cumbersome computation of the fundamental in the output of the nonlinear block. We can do this analytically with a lot of effort or we can simply let the necessary integrals be computed by e.g. Matlab or Simulink. If we have to choose for the latter, it makes not so much sense any longer because some extra programming and we simulate the whole process and can simply observe what happens and analyse the behaviour in state space domain. We will do so in section 1.4. A typical example for such a situation is a continuous, linear process which is controlled by a digital controller as in the following subsection.

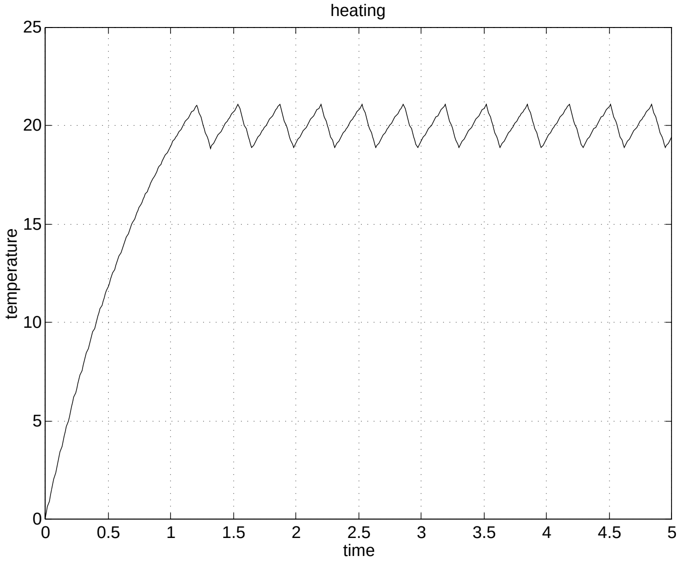


Figure 1.21: Output example2.

1.2.5 Example3: Quantisation error and sampling

The inevitable quantisation in the analog-digital and the digital-analog converters introduces a nonlinear effect and the closed loop system often shows a limit cycle as can easily be demonstrated in the next example3 in Fig. 1.22.

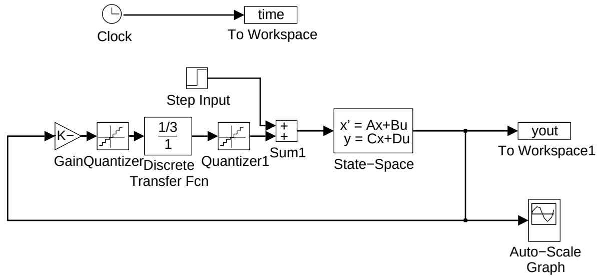


Figure 1.22: Digital controller for a continuous process.

For simplicity the linear process is taken to be just an integrator $3/s$ and the .5 step input represents a disturbance offset causing a ramp trend at the output in open loop. In continuous time a simple negative constant feedback would be sufficient to stabilise the output on a constant value 1.5 as there is an integrator in the feedback loop. For a digital controller this is not the case due to the limited resolution which is set to 1: i.e. the least significant bit corresponds to 1. Also the sampling period, taken to be 1 here, plays an important role. This choice is defensible as the -6dB point for the process lies at approximately 1 Hz ($|3/s| = 3/2\pi f \approx 1/2$). As this is a rather low sampling rate we have to be sure that at least in z-space, ignoring the quantisation errors, the closed loop system is stable. For a constant feedback with gain K the $s \rightarrow z$-transform can be given by:

$$\frac{1-e^{-sT}}{s}\frac{3K}{s} \xrightarrow{s\rightarrow z} (1-z^{-1})\frac{3KTz}{(z-1)^2} = \frac{3KT}{z-1} \tag{1.21}$$

so that the pole of the closed loop system can be obtained from:

$$1 + \frac{3KT}{z-1} = 0 \rightarrow \tag{1.22}$$

$$z - 1 + 3KT = 0 \rightarrow \tag{1.23}$$

$$z = 1 - 3K \tag{1.24}$$

The most stable system is designed by putting the pole in the origin (dead beat control!) so that only a delay remains. This is effected by choosing K=1/3 and it would imply that after one sample the system is at rest in 1.5. Fig. 1.23 indicates that this is not true due to the quantisation effect.

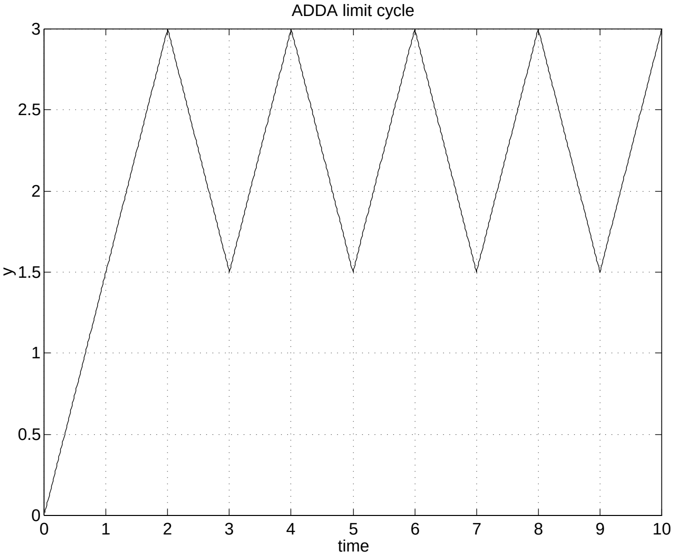


Figure 1.23: Output example3.

A limit cycle occurs that can be explained as follows:

Let y be the continuous output of the integrator and z the sampled and quantisised output both being zero at time zero. Let u be the input of the integrator inclusive the disturbing offset. Then we can write:

$$\begin{array}{lllll} y = 3\int_0^1 .5dt & \rightarrow & y(1) = 1.5 & z(1) = 1 & u(1) = .5 \\ & \rightarrow & y(2) = 3 & z(2) = 3 & u(2) = -.5 \end{array} \tag{1.25}$$

$$\begin{array}{llll} y(3) = 3 - 3\int_0^1 .5dt = 1.5 & \rightarrow & z(3) = 1 & u(3) = .5 \\ y(4) = 1.5 + 3\int_0^1 .5dt = 3 & \rightarrow & z(4) = 3 & u(4) = -.5 \end{array} \tag{1.26}$$

$$\vdots \tag{1.27}$$

So we conclude that the quantisation causes a limit cycle with an amplitude and offset of several quantisation levels.

1.3 Stability analysis by linearisation.

1.3.1 Theory for continuous time systems.

In cases where we deal with smooth nonlinearities and remain in the neighbourhood of a reference trajectory or reference point we may approximate the dynamics of the variations about the reference trajectory by a set of linear equations. We consider a non-linear system to be given in state space form:

$$\dot{x} = f(x, u) \tag{1.28}$$

where f is a continuous, nonlinear function in x and u with continuous derivatives in the neighbourhood of the reference trajectory which is defined by:

$$\dot{x}^0 = f(x^0, u^0) \tag{1.29}$$

For such a reference trajectory the initial value $x^0(0)$ and the driving input $u^0(t)$ are fixed. If these are chosen such that $\dot{x}^0 = 0$ for all t then the trajectory degenerates into a reference point x^0.

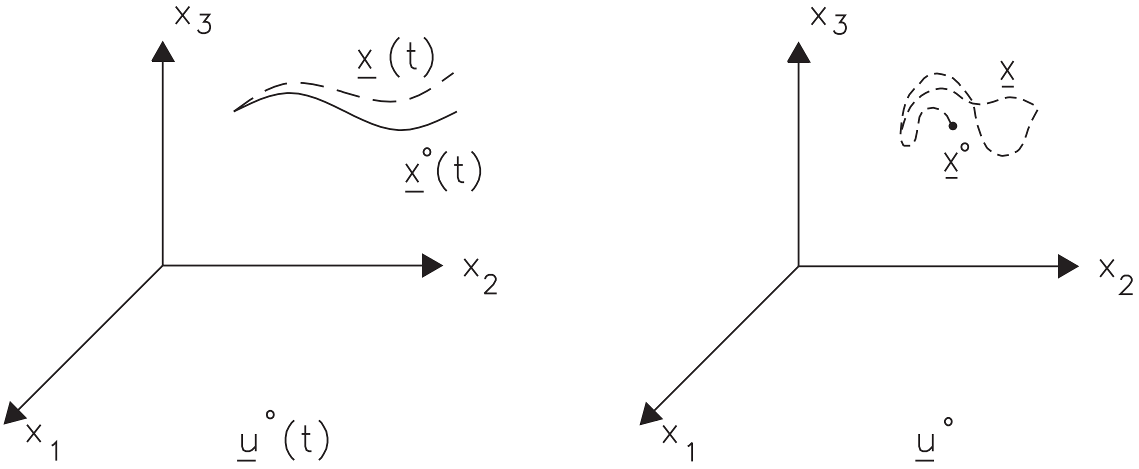


Figure 1.24: Reference and neighbouring trajectories.

Along the dashed real trajectory in Fig. 1.24 we find x caused by u in relation to the full reference trajectory x^0 caused by u^0 according to:

$$x = x^0 + \delta x \tag{1.30}$$

$$u = u^0 + \delta u \tag{1.31}$$

It is for the small excursions δx and δu that we would like to have an approximate linear description. Therefore we suppose that the δ-deviations are small so that we may neglect the higher order terms (H.O.T) in a Taylor series expansion. Consequently:

$$\frac{d}{dt}(x^0 + \delta x) = \dot{x}^0 + \delta\dot{x} = f(x^0 + \delta x, u^0 + \delta u) = \tag{1.32}$$

$$f(x^0, u^0) + \left(\frac{\partial f}{\partial x^T}\right)^0 \delta x + \left(\frac{\partial f}{\partial u^T}\right)^0 \delta u + H.O.T. \tag{1.33}$$

so that the δ-variations about the reference trajectory can be described by:

$$\delta\dot{x} \approx A\delta x + B\delta u \tag{1.34}$$

where the state matrix A and input matrix B are given by the coefficients of the linear terms, i.c. the so-called Jacobi matrices:

$$A = \left(\frac{\partial f}{\partial x^T}\right)^0 = \begin{pmatrix} \left(\frac{\partial f_1}{\partial x_1}\right)^0 & \cdots & \left(\frac{\partial f_1}{\partial x_n}\right)^0 \\ \vdots & \ddots & \vdots \\ \left(\frac{\partial f_n}{\partial x_1}\right)^0 & \cdots & \left(\frac{\partial f_n}{\partial x_n}\right)^0 \end{pmatrix} \tag{1.35}$$

$$B = \left(\frac{\partial f}{\partial u^T}\right)^0 = \begin{pmatrix} \left(\frac{\partial f_1}{\partial u_1}\right)^0 & \cdots & \left(\frac{\partial f_1}{\partial u_m}\right)^0 \\ \vdots & \ddots & \vdots \\ \left(\frac{\partial f_n}{\partial u_1}\right)^0 & \cdots & \left(\frac{\partial f_n}{\partial u_m}\right)^0 \end{pmatrix} \tag{1.36}$$

For general reference trajectories we are now left with vari-linear (i.e. time variant) equations governing the δ-variations. In most cases the reference trajectory will be a reference point or working point at which we would like to keep the system as shown in Fig. 1.24 right. This equilibrium point is then defined by:

$$\dot{x}^0 = 0 = f(x^0, u^0) \tag{1.37}$$

where also u^0 is in general a constant reference input. In deviation from linear systems this last equation can lead to many more solutions than one. Now finally the stability of the linearised system is of course determined by the A-matrix and in particular the eigenvalues of A define the poles of the linearised system. Nevertheless one should always keep in mind that these poles only determine the dynamics for small excursions about the reference point and substantial deviating behaviour can be found if the higher order terms of the Taylor expansion can no longer be neglected. The linearised equations surely define the local dynamics only! Let us elucidate this with a simple example.

1.3.2 Example4: Pendulum

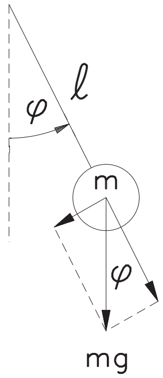


Figure 1.25: Pendulum

For the pendulum of Fig. 1.25 we assume that it can swing without friction with a weightless beam of length l and all mass m concentrated at the end point. The inertial moment equals the moment due to the gravity so that we get:

$$ml^2\ddot{\varphi} = -mgl\sin(\varphi) \tag{1.38}$$

We observe that the mass is irrelevant for the dynamics as we can divide both sides by it. Furthermore this is an autonomous system as there is no input u involved. For a state space description we can define x_1 to be the angle φ and x_2 its derivative yielding:

$$\dot{x} = \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} x_2 \\ -\frac{g}{l}\sin(x_1) \end{pmatrix} = f(x) \tag{1.39}$$

By putting this time-derivative of x equal to zero we obtain two equilibrium points:

$$x_{down} = \begin{pmatrix} x_{1d} \\ x_{2d} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad x_{up} = \begin{pmatrix} x_{1u} \\ x_{2u} \end{pmatrix} = \begin{pmatrix} \pi \\ 0 \end{pmatrix} \tag{1.40}$$

corresponding to respectively the hanging and inverted position. For the local stability of these equilibrium points we have to study the Jacobi- matrices:

$$A = \frac{\partial f}{\partial x^T} = \begin{pmatrix} 0 & 1 \\ -\frac{g}{l}\cos(x_1) & 0 \end{pmatrix} \tag{1.41}$$

where x_1 is either x_{1d} or x_{1u}. The poles for the respective equilibria can be obtained from:

$$|\lambda I - A| = \det(\lambda I - A) = 0 \tag{1.42}$$

For x_{down} we get:

$$x_1 = 0 \qquad \Rightarrow \qquad \lambda_{1,2} = \pm j\sqrt{\frac{g}{l}} \tag{1.43}$$

which we recognise immediately as the familiar undamped oscillation with the pendulum natural frequency:

$$\frac{1}{2\pi}\sqrt{\frac{g}{l}} \tag{1.44}$$

It is evident that for larger l the frequency will decrease. For the upper position we get:

$$x_1 = \pi \qquad \Rightarrow \qquad \lambda_{1,2} = \pm\sqrt{\frac{g}{l}} \tag{1.45}$$

which is surely unstable as we deal with a pole in the right half s-plane. This agrees with our daily experience and also the fact that for smaller l the system is more unstable as the pole shifts further from the origin causing faster, unstable exponential falling. It is easier to balance a broom-stick than a match. A tyrannosaurus could stand upright despite of a very slow nervous system while insects need six legs to do so. You, yourself, had to learn walking at the moment that this was most difficult given your length.

1.3.3 Theory for discrete time systems.

For discrete time systems the whole procedure develops similarly so that we write immediately:

$$x(k+1) = f(x(k), u(k)) \Rightarrow \tag{1.46}$$

$$x^0(k+1) + \delta x(k+1) = f(x^0(k) + \delta x(k), u^0 + \delta u(k) = \tag{1.47}$$

$$f(x^0(k), u^0(k)) + \left(\frac{\partial f}{\partial x^T}\right)^0 \delta x(k) + \left(\frac{\partial f}{\partial u^T}\right)^0 \delta u(k) \Rightarrow \tag{1.48}$$

$$\delta x(k+1) = A\delta x(k) + B\delta u(k) \tag{1.49}$$

Again A and B are the proper Jacobi matrices where the independent variable t for continuous time is now replaced by the integer k indicating the sample number. For a working point in stead of a time depending trajectory we get constant A and B matrices. For local stability we have to check whether the eigenvalues of matrix A live in the unit disc of z-domain, i.e. $\lambda_i < 1$. Just mind that for such an equilibrium point in discrete time we get from $x(k+1) = x(k) = x^0$ in equation 1.46 the condition:

$$x^0 = f(x^0, u^0) \tag{1.50}$$

1.3.4 Example5: Chain Wheel

While repairing your bike you might have experienced (as I did) that , although you want to keep your hands as clean as possible, the chain system plays games with you and again and again the chain rolls off the chain wheel as together they represent an unstable system. If you have dismounted a pedal the other pedal can supply a torque which might be sufficient to stop the rolling off in a stable equilibrium point as indicated in Fig. 1.26.

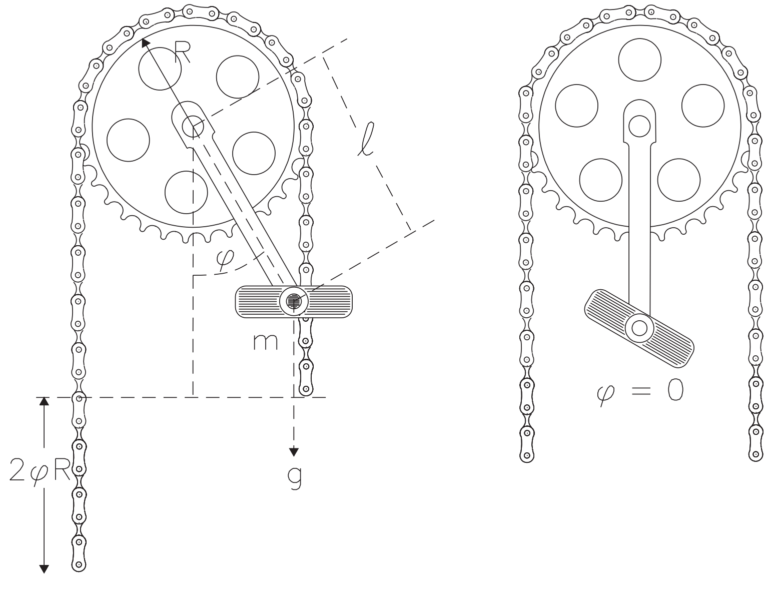


Figure 1.26: Chain wheel configuration.

If the angle $\varphi = 0$ corresponds to the situation that the pedal is at its lowest position and the chain hangs evenly over the wheel. For any other φ a stable equilibrium might occur when the torques are compensating:

$$2R^2\rho g\varphi = mgl\sin(\varphi) \tag{1.51}$$

where ρ represents the specific mass per length of the chain. We can rewrite this as:

$$x = \sin(\omega x) \qquad \omega = \frac{ml}{2R^2\rho} \qquad x = \frac{\phi}{\omega} \tag{1.52}$$

We can try to solve this equation in x by simply transforming it into a so-called Picard algorithm by the following iterative procedure:

$$x(k+1) = \sin(\omega x(k)) = f(x(k)) \tag{1.53}$$

so that we are again confronted with an autonomous system. Note that this nonlinear state space description does not represent the dynamics of the chain wheel but a procedure to compute the equilibrium points of the chain wheel system. It is not sure that this procedure will arrive at the desired equilibrium point but this is just the thing we want to study. We can get a good insight into the problem by studying the Picard diagram as shown in Fig. 1.27.

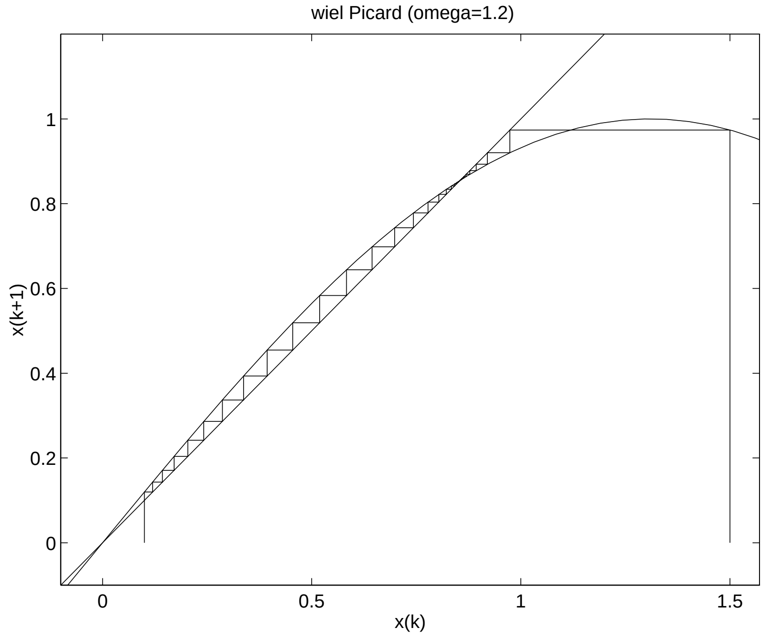


Figure 1.27: Picard chain wheel $\omega = 1.2$.

An initial guess $x(0) = .1$ will lead to $x(1)$ according to the given function $f(x(k))$. This $x(1)$ then functions as a next guess which puts it on the horizontal axis by the reflection against the line $x(k+1) = x(k)$. Next this procedure repeats until a stop is given. For the given value of $\omega = 1.2$ we observe that the procedure converges to the equilibrium point i.e. the intersection of the function f with the line $x(k+1) = x(k)$. The same can be observed for the negative equilibrium point which shows a complete symmetry. However, the equilibrium $x = 0$ will never be found as the procedure will always walk away. Locally this can be explained by linearisation which yields a state matrix A as follows:

$$A = \frac{\partial f}{\partial x} = \frac{\partial \sin(\omega x)}{\partial x} = \omega\cos(\omega x) \tag{1.54}$$

Because the state is one dimensional the A-matrix is a scalar and directly equals the pole of the local linear system related to the computing procedure. For $x = 0$ and $\omega = 1.2$ we get a pole of 1.2 which is really outside the unit disc so that the system is locally unstable about the origin. Note that A or the pole equals the derivative of the function f in the equilibrium point and this is greater than one which is represented by the line $x(k+1) = x(k)$ in Fig. 1.27. Along the same lines we see that the other equilibrium points are indeed locally stable for $\omega = 1.2$. The derivative in the equilibrium point can be linked immediately to the pole positions in z-domain according to Fig. 1.28.

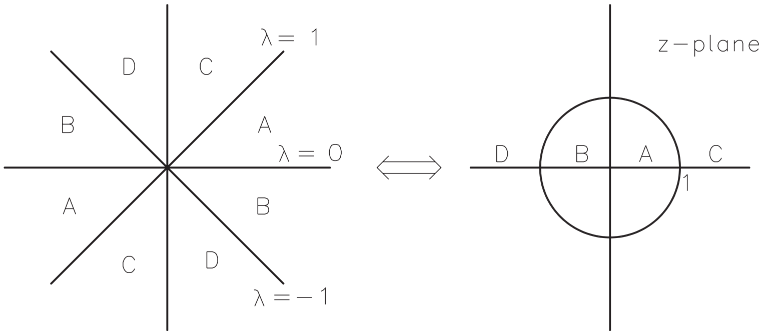


Figure 1.28: Relation derivatives and poles in z-plane.

Region A with pole $0 < \lambda < 1$ yields a stable locally convergent procedure where the final equilibrium is approached from one side as seen in Fig. 1.27. Region B idem for $-1 < \lambda < 0$ but now the equilibrium will be approached alternatingly from both sides as one can observe in Fig. 1.29 for $\omega = \pi/2 + .4$.

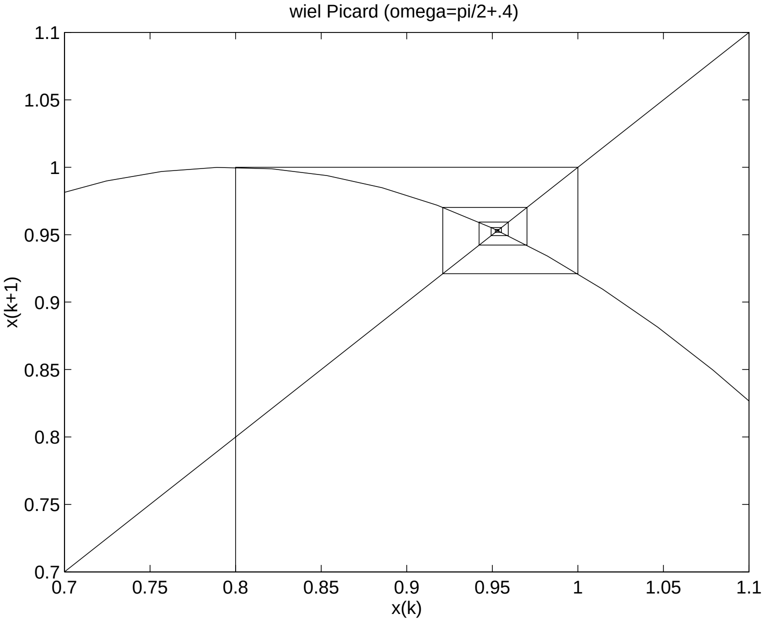


Figure 1.29: Picard chain wheel $\omega = \pi/2 + .4$.

Locally unstable behaviour is linked to regions C and D where in C the output will drift away in one direction while in D there is again the alternating direction effect. An example of C was the equilibrium point $x = 0$. Situation D is illustrated in Fig. 1.30 where we have chosen $\omega = 2.5$.

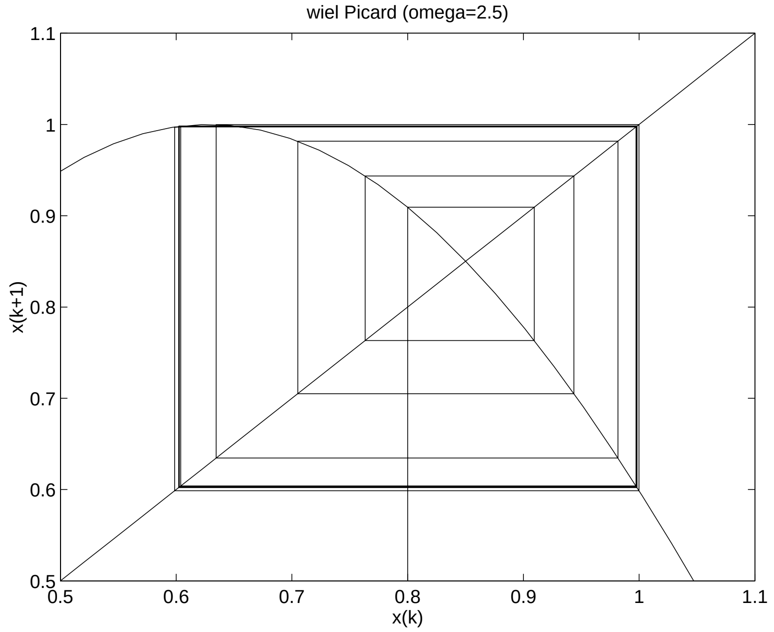


Figure 1.30: Picard chain wheel $\omega = 2.5$.

Although the system is locally unstable it finally shows a "stable" oscillation. This is due to the fact that the local instability forces the state x to fly from the equilibrium point but the curved function $f(x(k))$ prohibits it to go further away than 1. As a result the system starts oscillating. It is said that the original final equilibrium value has bifurcated into two values between which the system is now oscillating. If we still increase the value of ω new bifurcations will occur and the system will alternatingly take all these values. Very soon with increasing ω the number of bifurcations is infinite and the situation is denoted by *chaos*. It means that the system is unstable but the amplitude is bounded. The state will wander about the equilibrium point in a chaotic way so that this point is indicated as a strange attractor. Fig. 1.31 illustrates this situation where $\omega = \pi$.

If you watch the signal x as a function of time as shown in Fig. 1.32 it looks like a noise signal. Nevertheless in principle we can completely describe the signal deterministicly as we know the driving function $f(x(k))$. However, we need an infinite accuracy to actually doing this! As the system is locally unstable a small difference in initial value will soon result in completely different signals even though the amplitude remains finite. This is a typical property of chaos which causes the signal never to come back to exactly the same value as before. The simulation with the computer is therefore insufficient as there we have to be content with a finite accuracy. Now one could remark that it is a pity this computing scheme, to find the equilibrium, does not succeed here but who cares? The same effects occur, though, with similar functions that describe real processes like population growth in the following famous example borrowed from May. You may take it as e.g. a model for viral disease development as was verified in practice.

1.3.5 Example6: Population growth

Let $x(k)$ represent the scaled number of individuals in a population for generation k. The scaling is such that if $x(k) = 1$ there is no next generation because of limited food supply and epidemics. Let the number of individuals in the next generation $x(k+1)$ be proportional to $x(k)$ due to relative number of births per individual but let it also contain

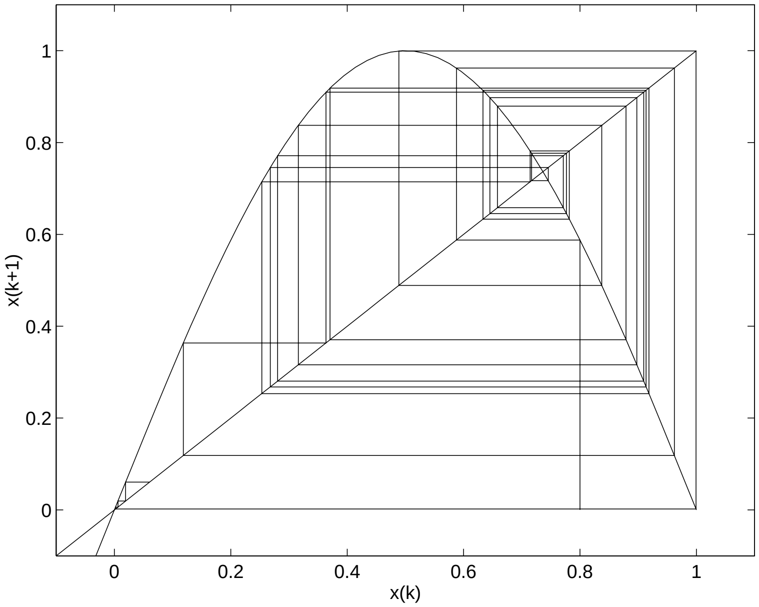


Figure 1.31: Picard chain wheel $\omega = \pi$.

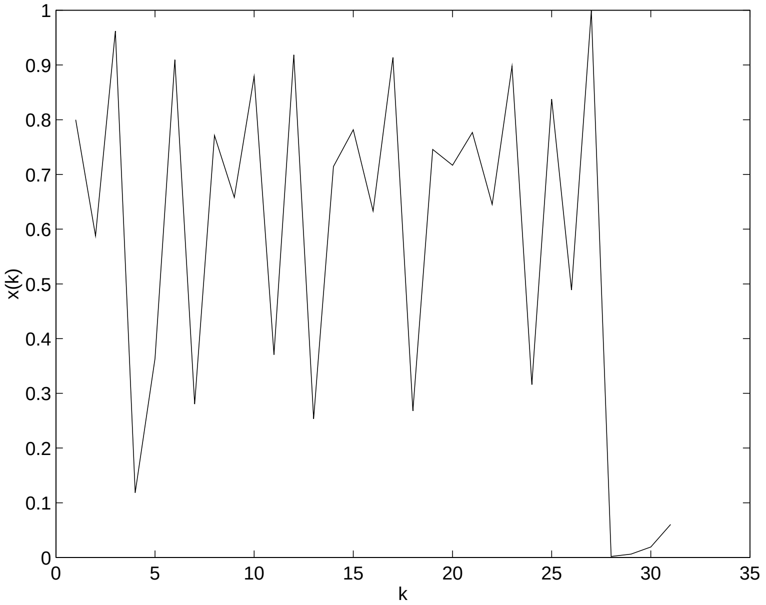


Figure 1.32: $x(k)$ chain wheel for $\omega = \pi$.

a term which limits unbounded increase. This term is caused by limited availability of food in the habitat and increased danger of epidemic illnesses and is simply incorporated here by a negative square term. So we consider:

$$x(k+1) = r(x(k) - x(k)^2) \qquad r > 0 \tag{1.55}$$

In Figs. 1.33 to 1.37 one may observe exactly the same effects as seen with the chain wheel example simply because the function $f(x(k))$ has a similar shape. We recognise respectively the stable equilibrium which bifurcates for increasing r (kind of birth rate) more and more until chaos rules. In this chaos sometimes the population almost dies out to recover again in initially exponential grow (local instability) until the curve $f(x(k)) =$

$r(x(k) - x(k)^2)$ bounds the increase again. Only if $r > 4$ the top of this curve becomes greater than one for $x(k) = .5$ so that $x(k+1)$ becomes greater than one which implies that the population has died out. So one can indeed value birth control! If this kind of control could independently be applied according to:

$$x(k+1) = r(x(k) - x(k)^2) + u(k) \tag{1.56}$$

then a control to keep the population at a constant level say c is evident by just proclaiming the birth control:

$$u(k) = r(x(k) - x(k)^2) + c \tag{1.57}$$

This is a typical example of linearisation by feed back which we will shortly discuss in section 3.1.

1.3.6 Exercise with Mandelbrot set

An attractive exercise to test your ability to apply the linearisation as a test for local stability is offered by the Mandelbrot set from fractals theory which is well known nowadays. This set is generated by the "seed" equation:

$$z(k+1) = z(k)^2 - a = f(z(k), a) \qquad z = x + iy, a\epsilon\mathbb{C} \tag{1.58}$$

and the Mandelbrot set is defined by:

$$M = \{a : a\epsilon\mathbb{C}, z(0) = 0 \cap \lim_{k\to\infty} z(k) \neq \infty\} \tag{1.59}$$

In words the Mandelbrot set is the set of complex valued a's such that all sequences started in 0 and defined by the seed equation remain finite. The set is represented in black in Fig. 1.38.

As an exercise one might compute the equilibrium points of $z(k+1) = F(z(k))$ with F equal to the seed f but also to $f(f)$, $f(f(f))$, $f(f(f)))$ etc. and study for which a's these equilibria are stable (i.e.$|dF/dz| < 1$). These concatenated mappings then represent oscillations between 1,2,3 etc. points that prohibit the sequence from going unstable. One then gets a series of subgeometries sequentially building the Mandelbrot figure such as the circles, parts of the real axis, the 'cardoid'(almost) etc.

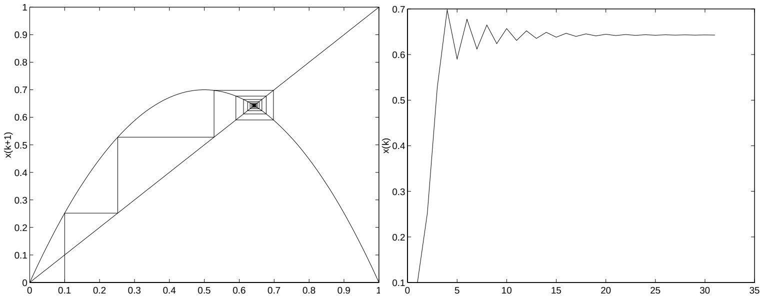


Figure 1.33: Population $r = 2.8$.

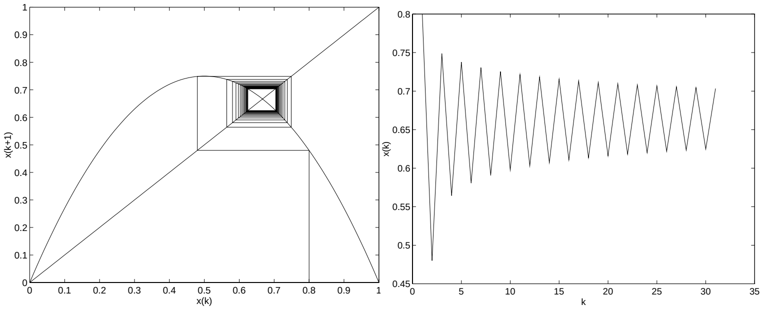


Figure 1.34: Population $r = 3$.

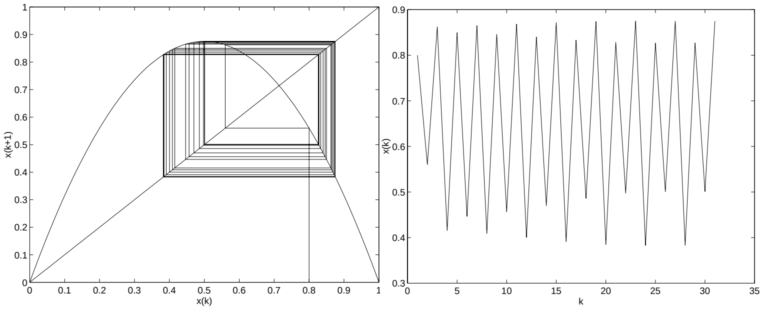


Figure 1.35: Population $r = 3.5$.

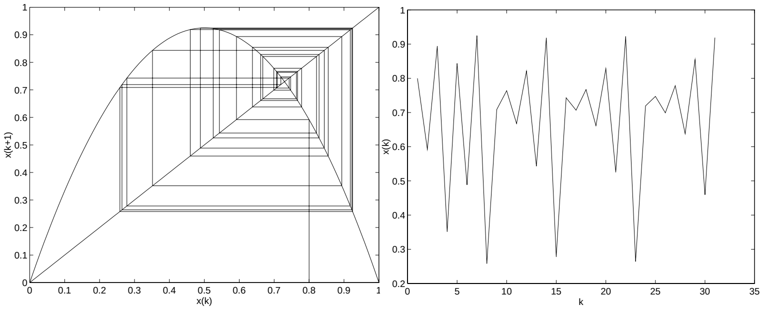


Figure 1.36: Population $r = 3.7$.

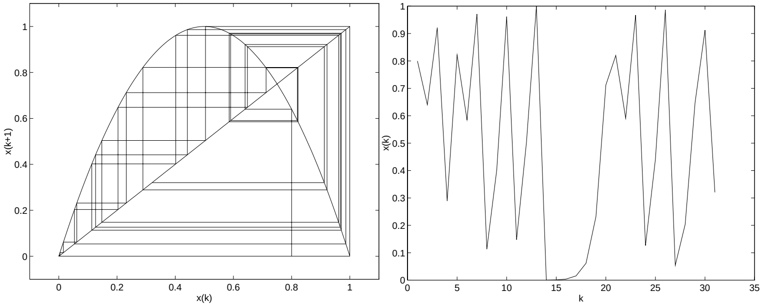


Figure 1.37: Population $r = 4$.

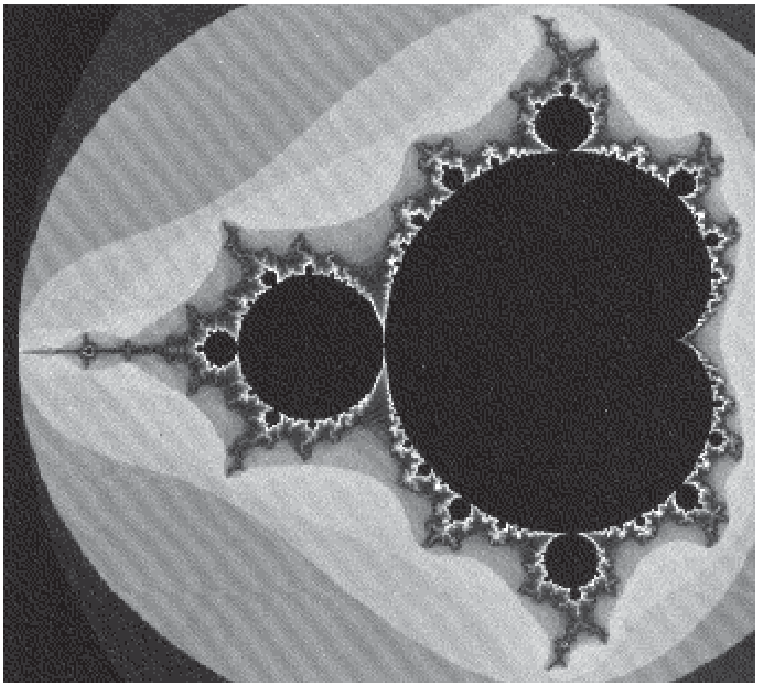


Figure 1.38: Mandelbrot set.

1.4 State space descriptions/ Phase portraits.

1.4.1 Theory

In section 1.3 we have seen how revealing a Picard diagram can be for analysing the development in time of difference equations. In continuous time a similar aid is offered by the state space. So first a number of states have to be defined equal to the order of the highest derivative present in the differential equation describing the dynamics. If there are more equations it simply is the total number of independent memory locations i.e. the number of possible energy storage positions. In the solution of the dynamical equations each state takes a value for every time moment. This is represented by a point in a space where the states are the coordinates. As time develops we observe a trajectory as shown before in Fig. 1.24. If we draw these trajectories for many different initial values we get an idea of the different areas in state space where the system behaves more or less similar, we observe separators that separate these areas and we see equilibrium points and limit cycles that can be stable, unstable or showing both behaviour: saddle points. If we deal with a second order system this space is a simple plane which is called the phase portrait of the system. Quite often one state represents a position or angle and the other state is nothing more than its derivative. By tradition the vertical axis is then linked with the derivative so that all trajectories show a direction which is clockwise: if the derivative is positive the first state will increase (first and second quadrant) and if the derivative is negative the first state will decrease (third and fourth quadrant). Very characteristic equilibrium points, separators and limit cycles can be distinguished which is best illustrated and studied with the help of examples.

1.4.2 Example4: Pendulum

The differential equations and state space representations have been derived in section 1.3:

$$\ddot{\varphi} + \frac{g}{l}\sin\varphi = 0 \tag{1.60}$$

$$x_1 = \varphi \qquad x_2 = \dot{x}_1 \tag{1.61}$$

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} x_2 \\ -\frac{g}{l}\sin(x_1) \end{pmatrix} \tag{1.62}$$

By means of a simulation in Simulink it is of course very easy to make the phaseportrait. Fig. 1.39 shows the blockscheme while Fig. 1.40 plots the phaseportrait for $l = g$.

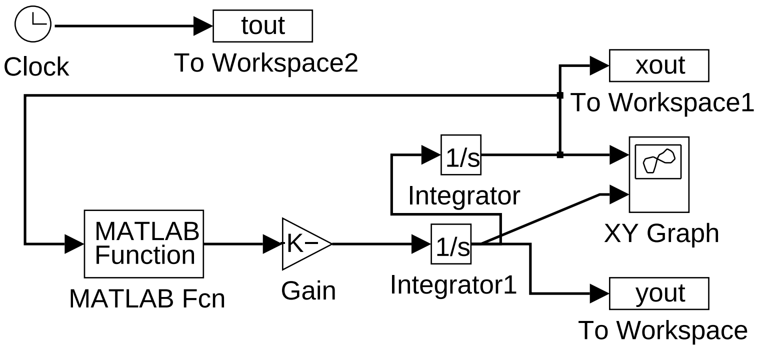


Figure 1.39: Block scheme of pendulum.

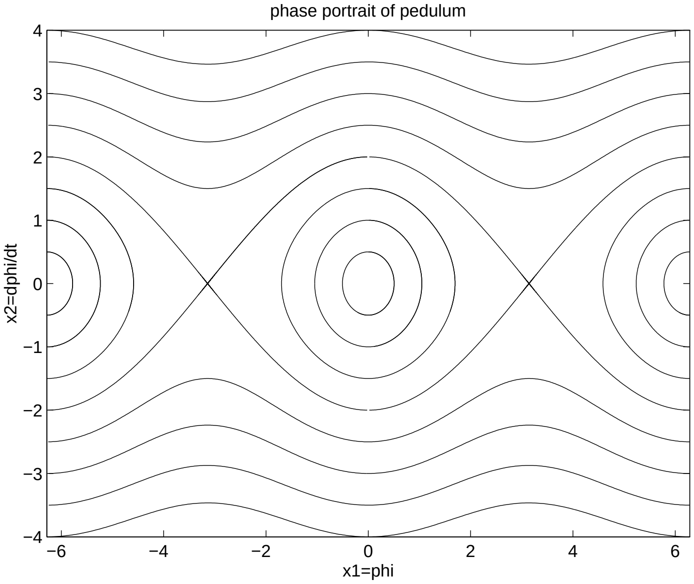


Figure 1.40: Phase portrait of pendulum for $l = g$.

If the initial condition is $\varphi = 1$ and $d\varphi/dt = 0$ the familiar undamped swinging occurs which can be seen as almost elliptic trajectories in the phase portrait. All initial values of $|\varphi| < \pi$ and $d\varphi/dt = 0$ will yield similar trajectories. For $\varphi = \pi$ we find ourselves in the singular point of the exactly inverted pendulum. From section 1.3 we know that in linearisation it has a stable and an unstable pole which makes it a saddle point in the phase portrait. A small perturbation will destabilise the pendulum and its angle will increase or decrease depending on the falling direction. As there is no damping, after a while the pendulum will be exactly inverted again except for the very small perturbation started with. So the falling will continue again in the same direction so that the angle will increase or decrease even more. The trajectory is typically a separator which separates the swinging behaviour from the pure rotation which we observe in the top and the bottom of the phase portrait. Finally we observe many stable equilibrium points or nodes in $\varphi = 2k\pi$, simply called centers or vortices, and also many saddle points in $\varphi = (2k+1)\pi$ for $k = \ldots -3, -2, -1, 0, 1, 2, 3, \ldots$. The trajectories change completely if we add some damping to the system (example5):

$$\ddot{\varphi} + \frac{d}{ml^2}\dot{\varphi} + \frac{g}{l}\sin\varphi = 0 \tag{1.63}$$

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} x_2 \\ -\frac{d}{ml^2}x_2 - \frac{g}{l}\sin(x_1) \end{pmatrix} \tag{1.64}$$

The blockscheme and the phase portraits are given in Figs. 1.41 and 1.42 respectively.

The undamped orbits around the stable equilibrium points have now been changed into spirals converging towards the stable points. This is easily explained as permanently the friction consumes energy from the combined potential and kinetic energy of the pendulum. Hence, also the rotational trajectories at the top and the bottom will finally end in a spiral trajectory at a distance $2k\pi$.

We have encountered several singular points or nodes defined by $dx/dt = f(x) = 0$. Its behaviour can easily be analysed as the system can be linearised for the close neighbourhood. The undamped pendulum about $\varphi = 0$ is characterised by:

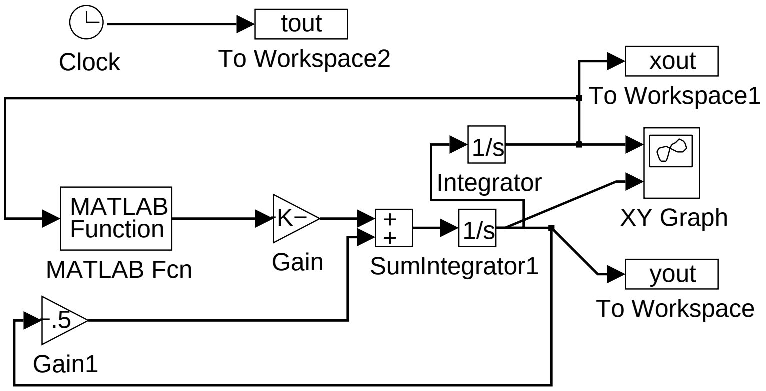


Figure 1.41: Block scheme of damped pendulum ($l = g$ and $d = ml^2/2$).

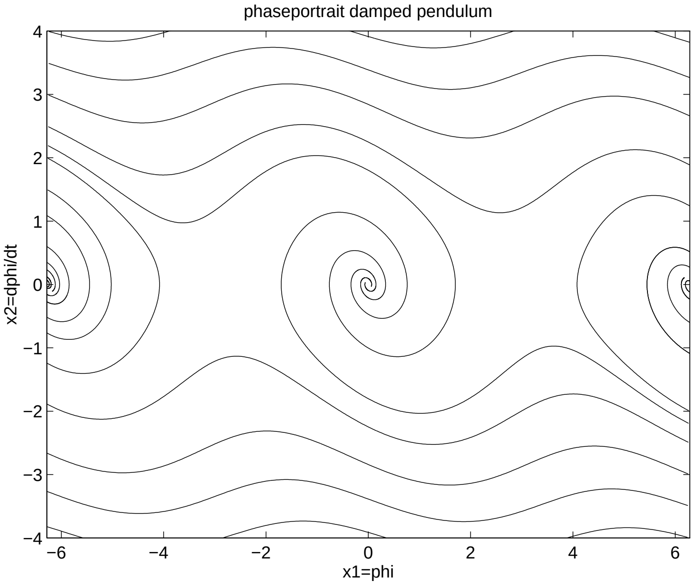


Figure 1.42: Phase portrait of damped pendulum ($l = g$ and $d = ml^2/2$).

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -\frac{g}{l} & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \tag{1.65}$$

as we saw in section 1.3. The solution is obviously given by:

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} R\sin(\omega_0 t + \psi) \\ R\omega_0 \cos(\omega_0 t + \psi) \end{pmatrix} \qquad \omega_0^2 = \frac{g}{l} \tag{1.66}$$

where amplitude R and phase ψ are determined by the initial values. By squaring the time t can be eliminated yielding:

$$\frac{x_1^2}{R^2} + \frac{x_2^2}{R^2\omega_0^2} = 1 \tag{1.67}$$

which is clearly the analytical description of an ellipse. Consequently two undamped poles on the imaginary axis induce ellipses in the phase portrait close to the singular point. If there is some damping involved as we introduced for the pendulum as well so that we obtain a complex adjoint pole pair in the left half s-plane we are similarly led to the following linearised state space description and solution:

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -\frac{g}{l} & -\frac{d}{ml^2} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \qquad \omega_0^2 = \frac{g}{l} \qquad \zeta\omega_0 = \frac{d}{2ml^2} > 0 \tag{1.68}$$

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} Re^{-\zeta\omega_0 t}\cos(\omega_1 t + \psi) \\ -R\zeta\omega_0 e^{-\zeta\omega_0 t}\cos(\omega_1 t + \psi) - R\omega_1 e^{-\zeta\omega_0 t}\sin(\omega_1 t + \psi) \end{pmatrix} \tag{1.69}$$

$$\omega_1 = \omega_0\sqrt{1-\zeta^2} \tag{1.70}$$

After elimination of time t we find:

$$(x_2 + \zeta\omega_0 x_1)^2 + \omega_1^2 x_1^2 = Ce^{\frac{2\zeta\omega_0}{\omega_1}} \arctan(\frac{x_2 + \zeta\omega_0 x_1}{\omega_1 x_1}) \tag{1.71}$$

These equations represent the spiral curves we observed in Fig. 1.42 close to the origin in linearisation. For positive relative damping ζ (and consequently positive damping d) the time direction is towards the spiral node or focus. For negative damping we obtain exactly the same curves but time direction is reversed. The trajectories are followed fleeing the node. From above we learn that even for simple examples computations become very complicated if to be done by hand. Some simulation tool, like Simulink, is practically indispensable.

1.4.3 More nodes and isoclines

One might wonder now what happens if we deal with real poles in a singular point. An example was encountered before in the saddle points of Fig. 1.40. These also occur, by the way, in Fig. 1.42 though these have not been exactly determined there. These saddle points represented both a stable and an unstable pole. Let us see what happens for a stable singular point with real poles given in a linearisation by:

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} -7 & -2 \\ +2 & -2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \tag{1.72}$$

The state matrix can be transformed to an eigenvalue decomposition:

$$\begin{pmatrix} -7 & -2 \\ +2 & -2 \end{pmatrix} = \begin{pmatrix} -1 & 2 \\ 2 & -1 \end{pmatrix} \begin{pmatrix} -3 & 0 \\ 0 & -6 \end{pmatrix} \begin{pmatrix} -1 & 2 \\ 2 & -1 \end{pmatrix}^{-1} \tag{1.73}$$

showing two stable poles in -3 and -6. The solution as a function of time is thus given by:

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} -1 & 2 \\ 2 & -1 \end{pmatrix} \begin{pmatrix} e^{-3t} & 0 \\ 0 & e^{-6t} \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} \tag{1.74}$$

Of course we could try to eliminate time but we would only arrive at a rather complicated formula not allowing extra insight. More revealing is to watch the derivative $dx2/dx1$ in the phase portrait plotted in Fig. 1.43.

For very large positive t the effect of the pole -6 is completely negligible with respect to the effect of pole -3 so that we may write:

$$\lim_{t\to\infty} \frac{x_2}{x_1} = -2 = \lim_{t\to\infty} \frac{dx_2}{dx_1} \tag{1.75}$$

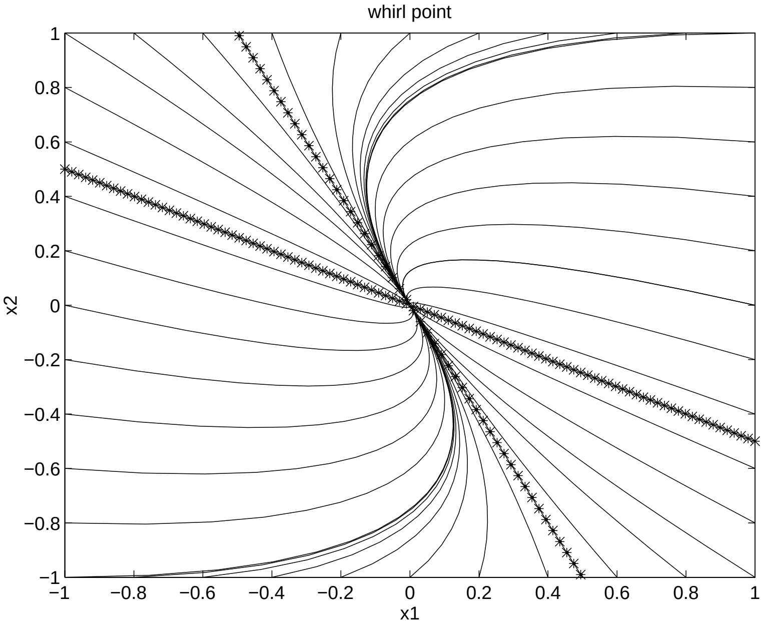


Figure 1.43: Singular point with real poles.

leading to

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} -1 \\ 2 \end{pmatrix} \alpha e^{-3t} \tag{1.76}$$

which exactly coincides with the line

$$x_2 = -2x_1 \tag{1.77}$$

which is defined thus by the eigenvector belonging to pole -3 and extra displayed in Fig. 1.43. For very large negative t (approaching $-\infty$) just the opposite occurs as exp(-6t) will be far dominant. Then the lines will turn towards the line defined by the other eigenvector given by:

$$x_2 = -\frac{1}{2}x_1 \tag{1.78}$$

and also displayed in Fig. 1.43. Since there is a smooth transition between these extremes we can indeed expect a whirl like pattern as exposed in Fig. 1.43. Such an equilibrium point is simply called a node and if the two eigenvalues coincide we have the extreme presence of the whirl-like behaviour which we then denote by whirl point or stable improper node. If the poles would have been unstable all that happens is that time arrows are just reversed. A degenerate node, or rather degenerate situation, occurs if at least one of the poles is zero. The solution then is given in terms of eigenvectors e_i and eigenfunctions:

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} e_1 & e_2 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & e^{pt} \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \alpha e_1 + \beta e^{pt} e_2 \tag{1.79}$$

which clearly shows that for a stable pole p the trajectory is simply along a line parallel to e_2 until it ends on a point on the line defined by e_1. For an unstable pole it simply starts there. So actually we have a complete line of equilibrium points given by the vector

e_1. Such a situation wil be dealt with for the relays with hysteresis example in section 1.4.5.

Finally the most symmetric node originates when we have two distinct equal poles not to be mixed up with double poles which lead to whirl points. Two distinct poles occur e.g. when the state matrix simply equals pI where p is the pole value and I the identity matrix. The trajectories are then given in parametric form where t is the parameter:

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} \alpha \\ \beta \end{pmatrix} e^{pt} \quad \Rightarrow \quad x_1 = \frac{\alpha}{\beta} x_2 \tag{1.80}$$

Consequently any line through the origin is a trajectory so that the name is a star point. It is a sink or a source depending on the sign of p. Finally Fig. 1.44 (next page) acts as a survey of all possible node characteristics and their corresponding pole positions in the linearisation.

It is clear for the star point that if this is the behaviour for all space (so a globally linear system) that everywhere on a trajectory the derivative $dx2/dx1$ will be the same. Such a line where the derivative is constant is called an isocline. For the star point these isoclines are straight lines. In general isoclines are curved lines and they are helpful as a tool to produce the phase portraits by hand. By far it is preferable to make the phase portrait by a simulation on a computer as this takes least time in general. Of course one can also try to eliminate time t from the parametric description of the solution but this can only be done with success for the very simple cases. So what is left are the isoclines. Once we have an analytic description of these isoclines we can draw them in the phase plane and in several points at about equal distances we can draw small lines with the proper derivative (compare Fig. 1.45). Then with many isoclines the trajectories will gradually evolve. How to find an isocline for derivative Ψ? This is very easy if we have the system equations in an explicit form:

$$\dot{x} = f(x) \tag{1.81}$$

because then we can divide the time derivatives:

$$\frac{dx_2}{dx_1} = \frac{dx_2/dt}{dx_1/dt} = \frac{\dot{x_2}}{\dot{x_1}} = \frac{f_2(x)}{f_1(x)} \qquad (\dot{x}_1 \neq 0) \tag{1.82}$$

For instance in case of a complex pole pair we get:

$$\Psi = \frac{dx_2}{dx_1} = -2\zeta\omega_0 - \omega_0^2 \frac{x_1}{x_2} \tag{1.83}$$

We obviously obtain straight lines:

$$x_2 = -\frac{\omega_0^2 x_1}{\Psi + 2\zeta\omega_0} \tag{1.84}$$

We might have expected this directly for $\zeta = 0$, as we obtained ellipses then, but it occurs to be true for the damped situation as well. For the real pendulum it is not more complicated:

$$\Psi = \frac{dx_2}{dx_1} = -\frac{d}{ml^2} - \frac{g}{l}\frac{\sin(x_1)}{x_2} \quad \Rightarrow \quad x_2 = -\frac{g}{l}\frac{\sin(x_1)}{\Psi + d/ml^2} \tag{1.85}$$

so that the isoclines are simple sinusoids. In Fig. 1.45 these isoclines have been drawn and on them small line pieces with the characteristic direction atan(Ψ).

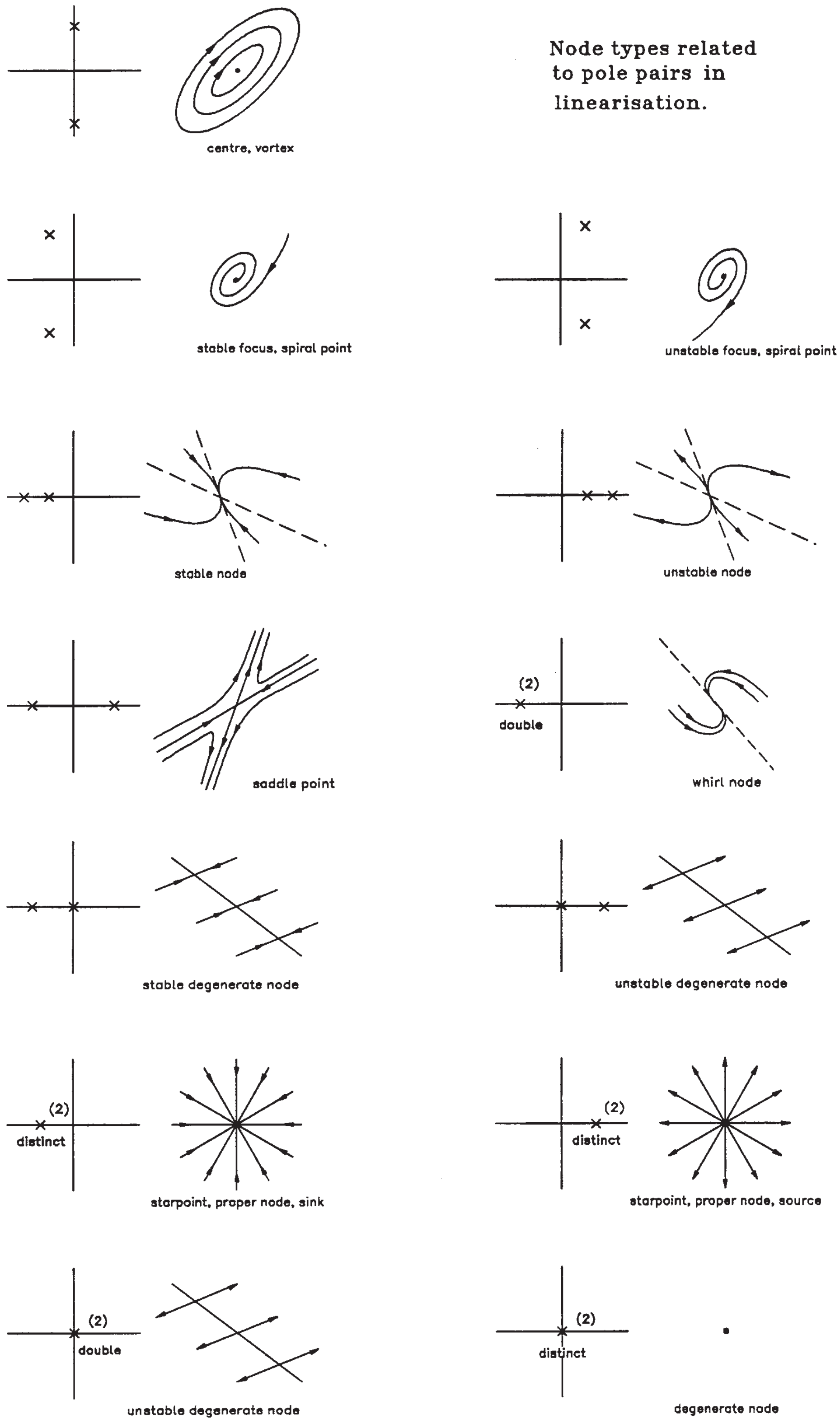


Figure 1.44: Node types related to pole pairs in linearisation

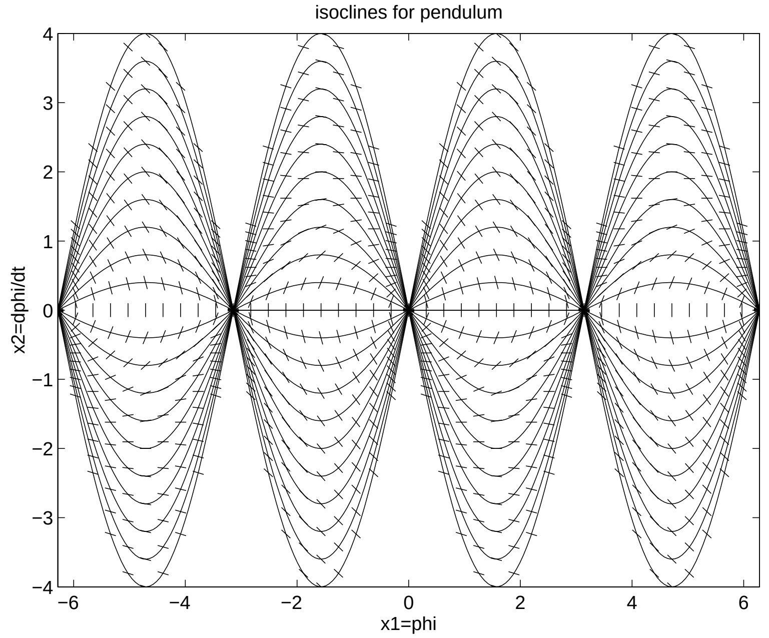


Figure 1.45: Isoclines of damped pendulum.

Combination of successive linepieces can be seen to produce the phase portrait of Fig. 1.42.

Sometimes by integration one can also find the trajectories itself e.g. for the undamped pendulum from the isocline:

$$\int x_2 dx_2 = \int -\frac{g}{l}\sin(x_1)dx_1 \qquad \Rightarrow \qquad x_2 = \pm\sqrt{\frac{2g\cos(x_1)}{l} + C} \tag{1.86}$$

where C, the integration constant, determines which trajectory is found. Of course each trajectory corresponds to a set of initial values exactly given by the trajectory itself. It is left to the reader to verify that for trajectories which stay close to the center points the trajectory converges to an elliptic orbit.

For the construction of the isoclines we have used the time derivatives and by division eliminated the time dependence via dt. There are methods to recover the time dependence from the phase portrait again to compute for instance the necessary time to reach one point on a trajectory from a given starting point on the same trajectory. From the last equation representing the trajectories for the undamped pendulum we can proceed as follows:

$$x_2 = \pm\sqrt{\frac{2g\cos(x_1)}{l} + C} = \frac{dx_1}{dt} \Rightarrow \tag{1.87}$$

$$\int \frac{1}{\pm\sqrt{\frac{2g\cos(x_1)}{l} + C}} dx_1 = \int dt = t \tag{1.88}$$

where t is the desired time span, C and the sign are determined by the trajectory under study while the first integral can be computed from the starting to the end point.

1.4.4 Limit cycles

A typical effect of nonlinear systems is the occurrence of oscillations that start spontaneously from many initial values and always converge to a same orbit in state space. This asymptotic orbit is called a limit cycle. The swinging of the undamped pendulum is no real limit cycles as, depending on the initial value, a specific orbit is taken immediately with indeed a special frequency characteristic for the pendulum properties but with an amplitude, purely defined by the initial value. There is no question of convergence. In daily life many limit cycles or relaxation oscillations can be found as the following citation from B. van der Pol and J. van der Mark (Phil.Mag.1928) illustrates:

Some instances of typical relaxation oscillations are: the aeolian harp, a pneumatic hammer, the scratching noise of a knife on a plate, the waving of a flag in the wind, the humming sometimes made by a water-tap, the sqeaking of a door,... the tetrode multivibrator, the periodic sparks obtained from a Wimshurst machine,... the intermittent discharge of a condenser through a neon tube, the periodic re-occurrence of epidemics and of economical crises, the periodic density of an even number of species of animals living together and the one species serving as food for the other, the sleeping of flowers, the periodic re-occurrence of showers behind a depression, the shivering from cold, menstruation, and, finally, the beating of the heart.

In honour of van der Pol we will start with his famous example:

$$\ddot{y} + (y^2 - 1)\dot{y} + y = 0 \tag{1.89}$$

By defining the states:

$$x_1 = y \qquad x_2 = \dot{x}_1 \tag{1.90}$$

the state equations are:

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} x_2 \\ x_2(1 - x_1^2) - x_1 \end{pmatrix} \tag{1.91}$$

This can readily be compared with a second order linear oscillator where the coefficient of y being $\omega_0^2 = 1$ but the coefficient of the damping term $\zeta\omega_0 = 1 - y^2$, so being output/state dependent. We see that the oscillator has a negative damping for $y^2 < 1$ causing the amplitude to increase. But as soon as the output/state y trespasses the threshold $y^2 = 1$ the damping is positive so that the increasing of the amplitude is turned into a decrease. Finally the amplitude stabilises but cannot lead to a constant damping so that the resultant oscillation won't be a pure sine wave. This effect is clearly shown in Fig. 1.47 being made from a demo in MATLAB Simulink under the name vdpol.m where the blockscheme is given in Fig. 1.46.

It can be seen that the oscillatory path is reached very quickly from starting points both inside and outside the limit cycle. In the origin resides a spiral node.

1.4.5 Separators

Separators are curves in the phase portrait that bound areas of similar behaviour. We have seen one so far for the undamped pendulum represented by the trajectories through the saddle points. Inside we got the swinging and outside the rotations.

Another separator is found in example1 of section 1.2.3 where it was found by means of describing functions that the system could either show a relaxation oscillation or activity could damp out depending on the initial values. In this case the trajectories can be

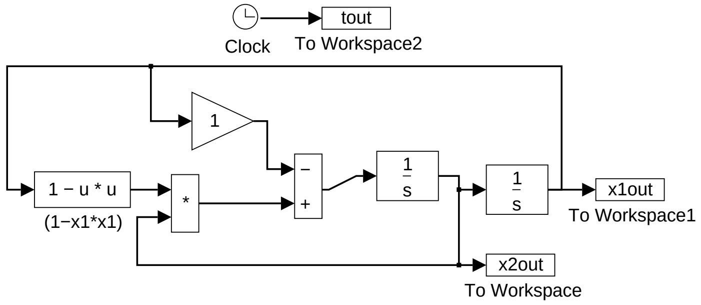


Figure 1.46: Blockscheme of Van der Pol Equation.

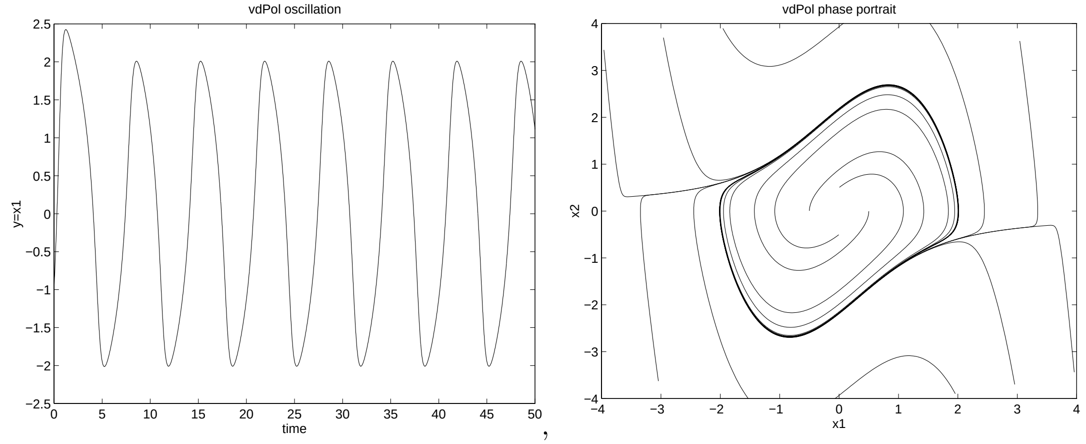


Figure 1.47: Time behaviour of Van der Pol Equation and various trajectories.

computed analytically. Given the transfer-function $H = K/(s + s^2\tau)$ it is obvious that the describing differential equation for zero reference is:

$$s(1+s\tau).-e = Kp \Rightarrow \tag{1.92}$$

$$\tau\ddot{e} + \dot{e} = -pK \tag{1.93}$$

where p indicates the output of the relays and thus the input of the linear system:

$$p\epsilon\{-A, 0, A\} \tag{1.94}$$

Now the following general ”trick” can be applied:

$$\ddot{e} = \frac{d\dot{e}}{dt} = \frac{d\dot{e}}{de}\frac{de}{dt} = \frac{d\dot{e}}{de}\dot{e} \quad \Rightarrow \quad \int de = \int \frac{\dot{e}d\dot{e}}{\ddot{e}} \tag{1.95}$$

so that for:

$$\ddot{e} = \frac{-pK - \dot{e}}{\tau} \tag{1.96}$$

results:

$$\int de = \int \frac{-\tau \dot{e} d\dot{e}}{pK + \dot{e}} \tag{1.97}$$

$$e = -\tau \dot{e} + p\tau K \ln(\frac{p}{|p|}(\dot{e} + pK)) + C \tag{1.98}$$

For $p = 0$ the logarithmic term disappears so that we have a linear relationship between e and de/dt. This holds for the area where all activity is damped out and where we deal with a degenerate node (one integration!). This area is represented in the center of Fig. 1.48 where the phase portrait is indicated by several trajectories.

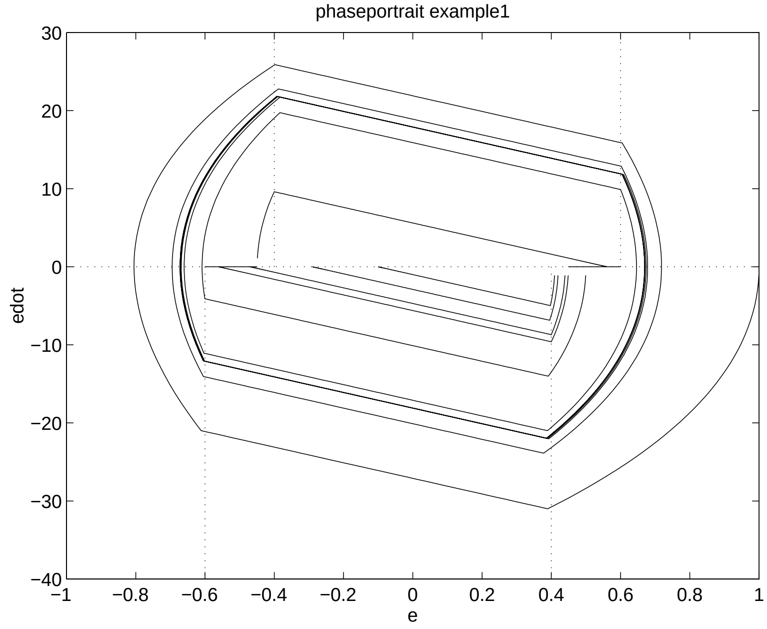


Figure 1.48: Phase portrait of the relay system example1

For clarity's sake just trajectories starting at several positive values for e and zero de/dt have been drawn. Of course there is symmetry with respect to the $e = 0$ axis. We clearly observe the straight trajectories ending on the e-axis between $-(u + h)$ and $(u + h) = .6$. If they end outside this interval the relay is activated and a logarithmic curve is followed until one of the lines $e = \pm u = \pm.4$ is reached. Then the relay falls off and we are back to a linear trajectory again until one of the lines $e = \pm(u + h) = \pm.6$ is encountered where again the relays switches on and a logarithmic curve is followed etc. The amplitude increases with each revolution until the clearly visible limit cycle is reached asymptotically. Starting at points outside the limit cycle will cause a trajectory that converges from the outside according to the same principles.

The separator is drawn explicitly. It is defined by the straight trajectories that end in $(\pm.6, 0)$, extended with the corresponding logarithmic trajectories that themselves start at about $(\pm.4493, 0)$ and completed with the small line-pieces on the horizontal axis linking the last points to $(\pm.6, 0)$. Inside this separator the trajectories stop on the horizontal axis between -.6 and +.6. Outside this separator the trajectories converge to the limit cycle. So the separator clearly indicates which initial values will lead to an oscillation or will simply die out. We have foreseen this in the describing function section but there we could only roughly indicate these effects.

Such an improved insight in what leads to limit cycles seems to have caused M. Schuler to heaving the following sigh (after a presentation by K. Magnus on this topic in Heidelberg 25-29 september 1956): "*Ich habe manche von meinen Schülern bei den Versuchen mit automatischen Flugzeugsteuerungen verloren. Jetzt, nach den Ausführungen von Herrn Magnus, wird mir mit einem Schlage klar, wie so etwas müglich war.*"[2]

Fortunately, we take better care of our students nowadays (in our group).

1.4.6 Chaos

One might wonder why there are not yet encountered any chaotic trajectories in the previous examples in continuous time phase portraits. This has a very simple reason: A chaotic trajectory will never turn back to a previous point and is limited in amplitude. For second order systems it is therefore impossible as, without crossings of trajectories, a bounded orbit will necessarily converge to limit cycles. For the discrete time systems we had not such a problem in the Picard diagrams so that even first order systems could provoke chaos. For continuous time systems we need at least a third order system where the extra order and thus extra ordinate helps to avoid crossings of trajectories. A famous example is from Lorenz (1963), a study of two-dimensional convection in a horizontal layer of fluid heated from below described by the following equations:

$$\begin{cases} \dot{x} = \sigma x + \sigma y \\ \dot{y} = rx - y - zx \\ \dot{z} = -bz + xy \end{cases} \tag{1.99}$$

in which x represents the velocity and y, z the temperature of the fluid at each instant, and r, σ, b are positive parameters determined by the heating of the layer of fluid, the physical properties of the fluid, and the height of the layer. It is left to the reader to simulate and analyse these dynamics for e.g. r=28, σ=10 and b=8/3. (Or simply type "Lorenz" in matlab!)

Of course many more interesting facts can be studied about chaos but we have to confine to control and much has to be treated still so that we will proceed with control in phase portraits in the next section. Nevertheless many processes really show chaotic behaviour though only recently discovered. Many of the examples cited from van der Pol are actually chaotic and this is a *positive* characterisation. Chaotic processes appear very robust in keeping up an oscillation: your heart beat should be irregular (i.e.chaotic); if it is very regular (limit cycle) you should really worry!

1.4.7 Nonlinear state space control

Since very much is known about the (state space) control of *linear* systems (still to be treated in the second half of this course) control engineers have tried to linearise systems first and apply the known linear control theory next. However, linearisation about a working point is only of limited value because of very restricted amplitudes. Quite often there is a possibility though to really linearise the system *by feedback* as illustrated in Fig. 1.49.

Feedback via a precompensator F which is nonlinear itself can linearise the total transfer between u^* and y. As an example we can take the pendulum where we suppose that the rod of the pendulum is mounted perpendicularly on the axis of a motor so that we can apply an input torque u. The equations of motion become:

[2] W.W.Solodownikow,"Grundlage der Selbststätiger Regelung", Band II

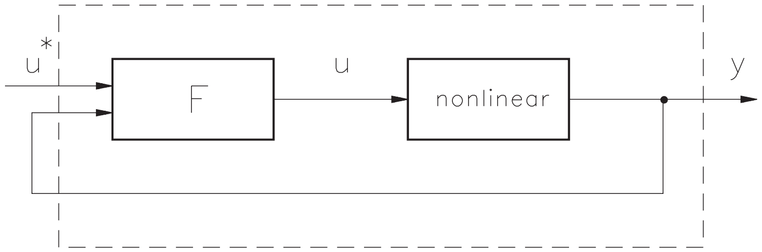


Figure 1.49: Linearisation by feedback.

$$\ddot{\varphi} + \frac{g}{l}\sin(\varphi) = u \qquad y = \varphi \tag{1.100}$$

It is trivial that by taking:

$$u = \frac{g}{l}\sin(y) + u^* \tag{1.101}$$

we will simply be left with a double integrator which is easy to control as it is a linear system.

Extensive discussion of this principle can be found in Isidori [1] or Nijmeijer/van der Schaft [5]. A severe condition is that the modeling errors, disturbances and measurement noise can really be neglected. To a very high degree this is e.g. true for robot systems. If not, the corresponding deteriorating signals are fed back as well and disrupt the whole effect which can easily be seen in the example for a measurement noise term ξ:

$$u = \frac{g}{l}\sin(y+\xi) \tag{1.102}$$

$$\ddot{y} = \frac{g}{l}(\sin(y+\xi) - \sin(y)) + u^* \tag{1.103}$$

Actually, this technique only compensates the nonlinearity. When one is confronted with such a nonlinearity which is inseparable from the process it seems all right. But sometimes the nonlinearity is deliberately brought into the transfer by the actuator as we have noted before. For instance a switching actuator is in general a very cheap way to put power into a system. A valve can easily be opened or closed and it is easier to switch on and off a pump than to control it linearly over a sufficiently broad range. We have seen the abundant class of examples in heating systems. Particularly these discontinuities are very hard to compensate in particular when the characteristics are not accurately known or change slightly in time. For such a system it may pay off to study it carefully in state space and design a suitable controller directly as for the next example7 illustrated in Fig. 1.50.

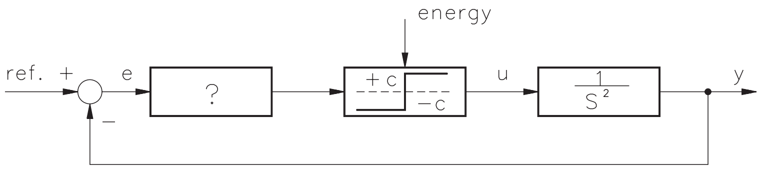


Figure 1.50: Double integration process with switching actuator.

The difference e between the reference signal ref and output y of a double integrator should be brought to zero as fast as possible where we can only switch between inputs $-c$

and $+c$. The trajectories belonging to $\pm c$ can easily be found. Because $e = ref - y$ and ref is a constant we may write:

$$\ddot{e} = -\ddot{y} = \mp c = \frac{d\dot{e}}{dt} = \frac{d\dot{e}}{de}\frac{de}{dt} = \frac{d\dot{e}}{de}\dot{e} \Rightarrow \tag{1.104}$$

$$\int \mp c de = \int \dot{e} d\dot{e} \qquad \Rightarrow \tag{1.105}$$

$$e = \mp\frac{\dot{e}^2}{2c} + C \tag{1.106}$$

which yields parabolic trajectories as displayed in Fig. 1.51. The capital C is an integration constant and is determined by the starting point.

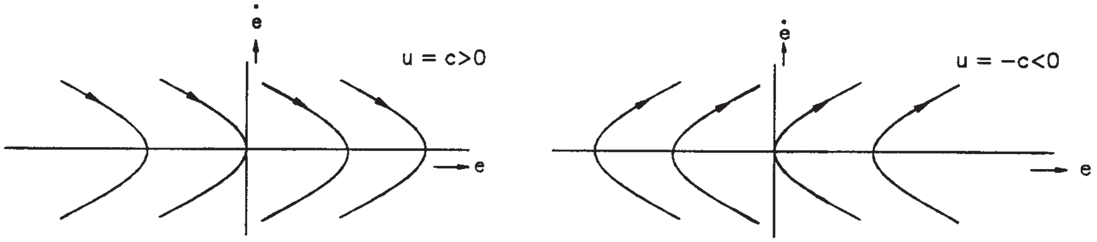


Figure 1.51: Trajectories for $u = c$ and $u = -c$.

The trajectories that pass trough the origin are very important as they can bring us to the desired position $e = 0$ $de/dt = 0$. In the second quadrant the curve $e = -\dot{e}^2/2c$ is the proper trajectory given the direction and $u = c$. In the fourth quadrant it is just the opposite so $e = +\dot{e}^2/2c$ for $u = -c$. These two half parabolas in the second and fourth quadrant divide the whole plane in two parts, see Fig. 1.52.

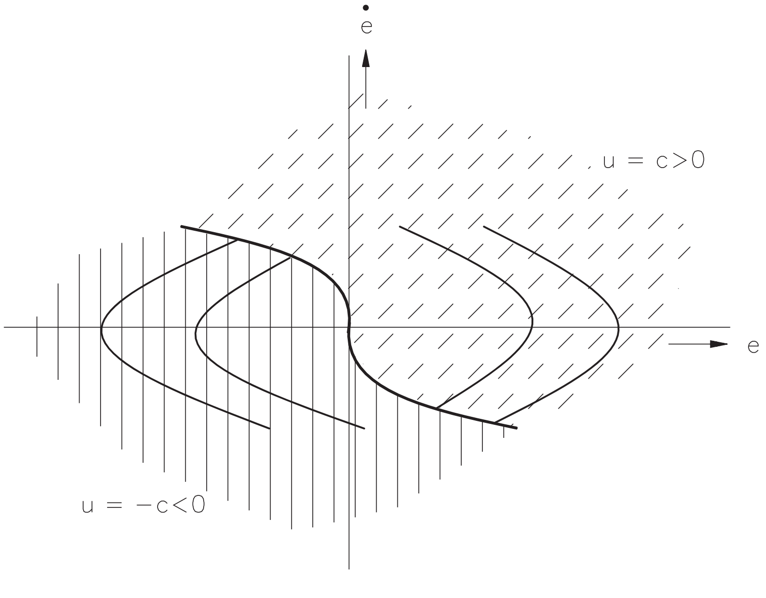


Figure 1.52: Phase portrait with proper switching strategy.

In the upper right half plane we observe that all left open parabolas (i.e. $e = -\dot{e}^2/2c+C$ with $C > 0$) will end on the bounding trajectory. In the lower left half plane the right open parabolas ($e = +\dot{e}^2/2c + C$ with $C < 0$) will end on the bounding trajectory. One could think of all these trajectories as being trains with a infinitely dense railway network and trains running at every moment. Then it is obvious that the fastest way to get in the origin from a certain starting point is to follow the fastest train to get to the bounding trajectory, change train immediately and travel in the bounding trajectory until it reaches the origin and jump out over there. In formulas it takes the form:

$$\begin{cases} e + \frac{1}{2c}\dot{e}|\dot{e}| > 0 & \Rightarrow u = c \\ e + \frac{1}{2c}\dot{e}|\dot{e}| < 0 & \Rightarrow u = -c \end{cases} \tag{1.107}$$

while the bounding trajectories or switching curve is given by:

$$e + \frac{1}{2c}\dot{e}|\dot{e}| = 0 \tag{1.108}$$

The example7 is worked out in Simulink in Fig. 1.53.

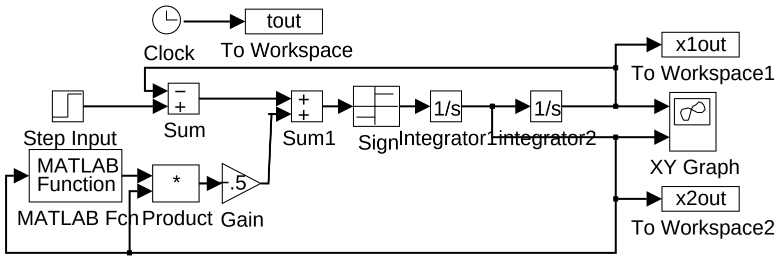


Figure 1.53: Bang-bang control of example7.

In Fig. 1.54 the controlled output and its derivative are displayed as a time function and in a phase portrait respectively.

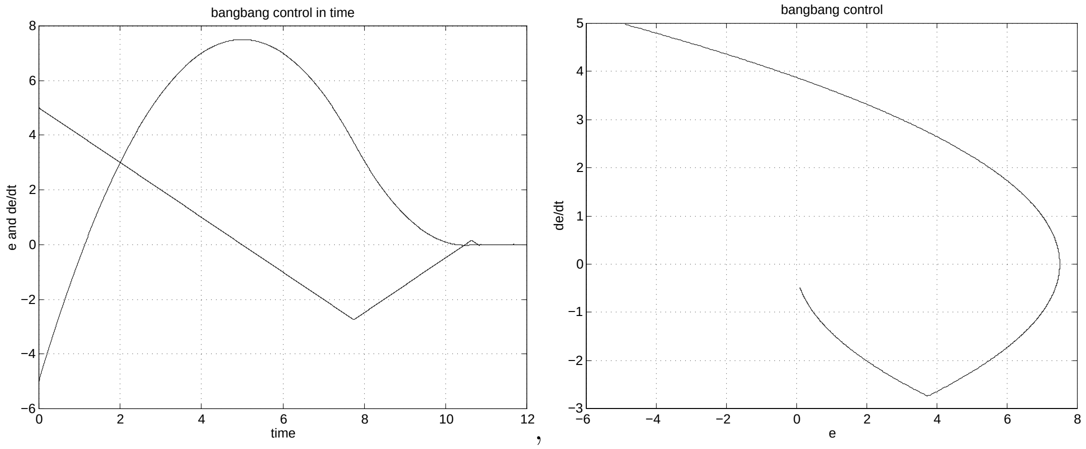


Figure 1.54: Outputs y and $\dot{y}$ and phase portrait of example7.

Since the output is the double integral of a constant input ($\pm c$) also the time functions are parabolas. The control takes care that the last trajectory is such that at the final moment both e and $\dot{e}$ are zero. In two switch actions the end point is reached. Appealing names for this type of control are: predictor control (for predicting the proper switching times), maximum effort or race-brake or (more common) bang-bang control. This kind of minimum time control performs very well even in cases where, during the control action, the reference signal changes stepwise. The reader is invited to simulate this himself and to analyse what happens if the switching times occur too early or too late particularly in the end point.

Chapter 2

Stability of nonlinear systems, Lyapunov.

2.1 Introduction

Until now stability has not been well defined. For *linear* systems the accepted interpretation is that the eigenfunctions belonging to the various poles (eigenvalues of the state matrix A) die out for large time whereas metastability then allows for constant values (single integration pole) or oscillations with constant amplitude (single undamped pole pairs). We are then sure that all states remain bounded.

For *transfer functions* we speak of BIBO-stable systems which stands for bounded input bounded output relations. However, judging stability just on the output behaviour is insufficient. In principle it is still possible then that some unobservable states increase exponentially. So we will define stability on the behaviour of all *states*. As a matter of fact we did not yet define the outputs of systems and only considered the states as these together represent all dynamics of the system whether reachable from the inputs u and observable from the outputs y or not. The insufficiency of just considering the outputs will be discussed further in the second part of this course concerning linear systems, where we require that all states are bounded. Here we take a somewhat broader viewpoint as we deal with nonlinear systems.

For *nonlinear* systems it makes no sense to talk about poles except for local dynamics about a working point in a linearisation. So we need to define what we mean by stability and for which area in state space it may possibly hold. A generally accepted definition is given by Lyapunov in his dissertation of 1892 entitled: "*Probleme général de la stabilité du mouvement*".

2.2 Definition of stability.

As the title of Lyapunov's dissertation suggests he not only considers points in state space but whole trajectories. In words the idea is that a system is called stable if a trajectory for a certain input and initial value is only slightly changed under the influence of small deviations in inputs or initial values. In mathematical terms this translates into:

Given the reference trajectory x(t) as a solution of:

$$\dot{x} = f(x, u) \tag{2.1}$$

where $u(t)$ and $x(t0)$ are given inputs and initial values.

Let a neighbouring trajectory be the solution of:

$$\dot{x}' = f(x', u') \tag{2.2}$$

then the system is (BIBO) stable if for all $\varepsilon > 0$ there exists $\delta > 0$ and $\delta_u > 0$ such that

$$\forall t \geq t_0 : \begin{cases} \|u'(t) - u(t)\| \leq \delta_u \\ \|x'(t_0) - x(t_0)\| \leq \delta \end{cases} \tag{2.3}$$

it holds that

$$\forall t \geq t_0 : \|x'(t) - x(t)\| < \varepsilon \tag{2.4}$$

where t_0 is the initial time under consideration.
The used norms can be any, e.g.:

$$\|x\|_1 = \Sigma_i |x_i| \tag{2.5}$$

$$\|x\|_2 = \sqrt{\Sigma_i x_i^2} \tag{2.6}$$

$$\|x\|_\infty = \sup_i \|x_i\| \tag{2.7}$$

where x_i are the components of x. The familiar Euclidean norm $\|.\|_2$ is most widely used as in the illustration in state space in Fig. 2.1.

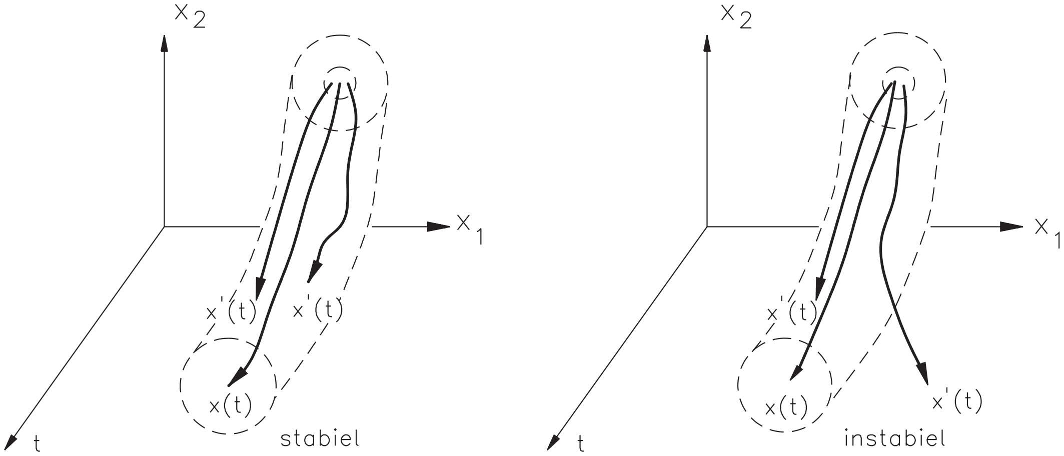


Figure 2.1: Visualisation of stable and unstable systems in statespace.

For stable systems the neighbouring trajectories should stay inside the tube of radius ε about the reference trajectory. One deviating trajectory is sufficient to declare the system unstable.

As an example we may look at the simple autonomous system consisting of a ball rolling under influence of gravity on a rail, curved in a vertical plane, slipless and without friction losses as drawn in Fig. 2.2.

Because there is no slip the rotation is uniquely coupled to the translation so that there is only one independent form of kinetic energy and the same for potential energy due to gravity. Hence, system dynamics can be described by a second order system with no damping terms or losses. From any starting position $-a < x(t0) < a$ and $\dot{x}(t0) = 0$

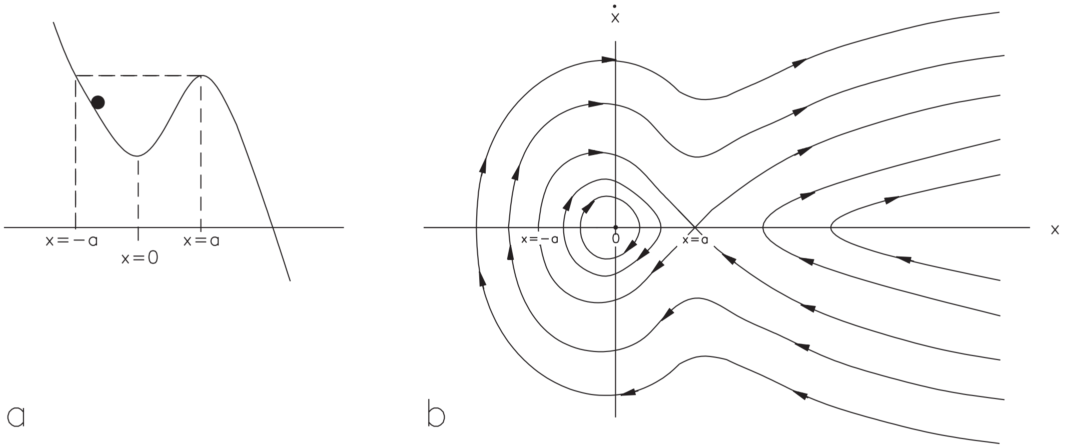


Figure 2.2: Ball on curved rail system.

the trajectories are orbits about the centre (0,0). The trajectory through the saddle point $(a, 0)$ functions as a separator. Outside this separator all trajectories go to infinity through the first quadrant i.e. the ball will fall down to the right of $x = a$ in the real world. One can readily see that the 'stable' orbits about $(0, 0)$ indeed remain in the direct neighbourhood for small deviations in initial values as long as it stays within the separator. So in that area the system is indeed stable about the trajectories. As a matter of fact the trajectories are lines of constant energy being the sum of potential and kinetic energy and all that happens is a continuous exchange between these two types of energy. So a small deviation in total energy won't force the trajectories to be much different.

Outside the separator the situation is quite different. If we take only a small deviation in the initial position and *not* in the initial speed and if both initial values of reference and neighbouring trajectory are beyond the separator the growth of the deviation will stay bounded so that in those cases the system might be called stable as well. One can see this happening if one considers two equal balls in free fall where there is no difference in initial speed. Strictly speaking we did not allow for all freedom then in the initial values and this is not according to the definition of Lyapunov stability. We should also allow for small deviations in initial speed and consequently there will be a term taking care for the effect of this speed difference in the position x that will integrate this constant speed difference leading to a ramp. So it follows that the system is not stable about the trajectories that go to infinity. This might seem strange at a first sight because the trajectory pattern is fixed. Indeed whether there is a small velocity difference or not in both cases neighbouring trajectories will be followed. However, there is no direct indication of time along the trajectories and it is just as a function of time that the positions on both neighbouring trajectories are widely divergent for initial speed differences. This is not the case by the way for the stable orbits about $(0, 0)$: the orbit revolution times are the same like it is for a pendulum of fixed length but different initial values.

In conclusion it can be stated that for stability of movement the reference trajectory itself is irrelevant. It may be unstable without prohibiting the movement itself to be stable. So we just have to study the dynamics of the small excursions $\Delta x(t)$ about the reference $x(t)$. In order not to complicate it too much we assume that the input $u(t)$ is kept the same. In formulas we can then state:

$$\Delta x(t) = x'(t) - x(t) \qquad \Rightarrow \tag{2.8}$$
$$\dot{x}' = \dot{x} + \Delta\dot{x} = f(x + \Delta x, u) \qquad \Rightarrow \tag{2.9}$$
$$\Delta\dot{x} = f(x + \Delta x, u) - f(x, u) = h(u(t), x(t), \Delta x) = g(t, \Delta x) \tag{2.10}$$

By doing so we have actually applied a coordinate transformation where on each moment the reference trajectory $x(t)$ is transformed to the origin of the Δx-system. According to Lyapunov the origin is now stable if:

$$\forall \epsilon > 0 \quad \exists \delta : \|\Delta x(t_0)| \leq \delta \Rightarrow \forall t > t_0 : \|\Delta x(t)\| < \varepsilon \tag{2.11}$$

If $\|\Delta x(t)\| \to 0$ for $t \to \infty$ we call the solution *asymptotically stable.* If stability holds for the whole space we speak of *global stability.* If it holds only in a restricted area it is called *local stability.* Local stability can only occur for nonlinear systems because for linear systems behaviour is independent of amplitude and thus always global:

$$\begin{cases} \dot{x} = Ax + Bu \\ \dot{x}' = Ax' + Bu \end{cases} \tag{2.12}$$
$$\Rightarrow \Delta\dot{x} = \dot{x}' - \dot{x} = A(x' - x) = A\Delta x \tag{2.13}$$

Also the reference trajectory is completely irrelevant when dealing with linear systems, in fact because of the superposition property. Of course for linear systems the Lyapunov stability is equivalent with "all poles of A in the left half s-plane". Derive yourself how the imaginary axis is allowed.

For discrete time systems the definitions are completely parallel:

$$x(k+1) = f(x(k), u(k)) \tag{2.14}$$
$$\Delta x(k) = x'(k) - x(k) \Rightarrow \tag{2.15}$$
$$x'(k+1) = x(k+1) + \Delta x(k+1) = f(x + \Delta x(k), u(k)) \Rightarrow \tag{2.16}$$
$$\Delta x(k+1) = f(x(k) + \Delta x(k), u(k)) - f(x(k), u(k)) = \tag{2.17}$$
$$= h(u(k), x(k), \Delta x(k) = g(k, \Delta x(k)) \tag{2.18}$$

The transformed origin is now stable if:

$$\forall \varepsilon > 0 \quad \exists \delta : \|x(k_0)\| \leq \delta \Rightarrow \forall k > k_0 : \|x(k\| < \varepsilon \tag{2.19}$$

For linear systems the concept of stability for the original movement and for the disturbed movement are congruent again. There is a global asymptotic stability *iff* (i.e. if and only if) all poles are within the closed unit disc in z-plane.

2.3 Stability analysis according to the second method of Lyapunov.

The stability analysis in the previous section is indicated by the *first method of Lyapunov.* It leaves us with the problem of analysing the (local or global) stability of the origin of the new coordinate system where the dynamics of $\Delta x(t)$ (or $\Delta x(k)$) have been given by the "forcing" functions h or g.

The *second method of Lyapunov* considers this last problem and provides means to establish the local stability inclusive an area for which this holds.

For reasons of convenience we will rename the variables without the prefix Δ again from here on and restrict ourselves to time independent "forcing" functions g. We then just transformed as a matter of fact equilibrium points corresponding to mostly constant u to the origin and are going to analyse the local or global stability of these points in the origin of our coordinate system. The dynamics are thus given by the autonomous equations:

for continuous time systems:

$$\dot{x} = g(x(t)) \tag{2.20}$$

for time discrete systems:

$$x(k+1) = g(x(k)) \tag{2.21}$$

We have seen that it makes no difference for linear system in which coordinate system we study the stability. For nonlinear systems this is certainly not true as we have seen e.g. in the case of the pendulum. Let us use this example to illustrate the coordinate transformation. The dynamics were found to be:

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = f(x) = \begin{pmatrix} x_2 \\ -\frac{g}{l}\sin(x_1) \end{pmatrix} \qquad \Rightarrow \tag{2.22}$$

$$\begin{pmatrix} \Delta\dot{x}_1 \\ \Delta\dot{x}_2 \end{pmatrix} = \begin{pmatrix} x_2 + \Delta x_2 - x_2 \\ -\frac{g}{l}\sin(x_1 + \Delta x_1) + \frac{g}{l}\sin(x_1) \end{pmatrix} = h(x, \Delta x) \tag{2.23}$$

If the reference trajectory is the equilibrium point $(x1, x2) = (\pi, 0)$, i.e. the stand up position, the transformed coordinate system is given by:

$$\begin{pmatrix} \Delta\dot{x}_1 \\ \Delta\dot{x}_2 \end{pmatrix} = \begin{pmatrix} \Delta x_2 \\ \frac{g}{l}\sin(\Delta x_1) \end{pmatrix} \tag{2.24}$$

Note that we have a positive sign in front of the sine leading to the locally unstable (linearised) system. We just moved the point $(\pi, 0)$ for x to the origin $(0, 0)$ for Δx.

The second method of Lyapunov gives a possibility to decide upon the local stability without resolving the differential (or difference) equations explicitly. It is inspired by the physical energy functions of Lagrange where in stable equilibrium the energy is minimal.

A good example is the pendulum where for the undamped dynamics the trajectories in the phase portrait (Fig. 1.40) also represent lines of constant total energy i.e. the sum of kinetic and potential energy. The minimum energy is then found in the stable equilibrium points $(2k\pi, 0)$. From these points the consecutive lines represent higher energies ever increasing from swinging movements to ever faster rotations.

If there is damping in the pendulum this damping will be responsible for the energy losses which causes the trajectories of the corresponding phase portrait (Fig. 1.42) always to intersect the iso-energy lines of the undamped situation inwards. By this we mean that the trajectories will always be driven to lower energy positions because mechanical energy is "consumed" by friction and transformed in heat. This can be observed by putting Fig. 1.42 on top of Fig. 1.40 as is done in Fig. 2.3.

So it seems as whether the trajectories follow a way that would be followed if there was a ball descending (very slowly, for neglect of inertial, centrifugal and Coriolis forces) the mountainous landscape of the undamped energy levels. It is then easy to imagine

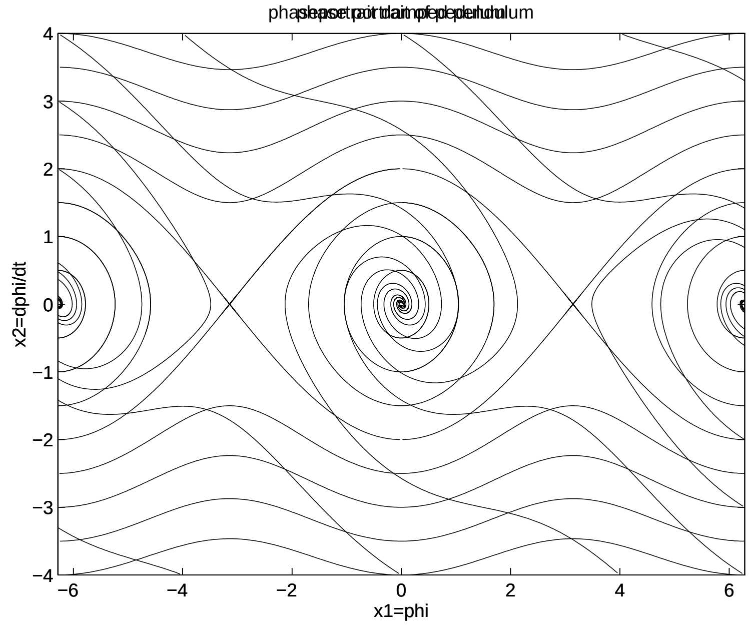


Figure 2.3: Lyapunov function for damped pendulum.

that the ball will finally end in one of the minima, i.e. the equilibrium points, which were already found to be locally stable in linearisation. On top of that we can observe that for what region the trajectories will finally end in the origin which defines the area wherein local stability for the origin can be guaranteed. This is in fact the principle that Lyapunov provides.

1

We were lucky in this case by having the energy function, say V consisting of a kinetic energy and a potential energy term:

$$V = \frac{1}{2}ml^2\omega^2 + mgl(1-\cos(\varphi)) = \tag{2.25}$$

$$\frac{1}{2}ml^2x_2^2 + mgl(1-\cos(x_1)) \tag{2.26}$$

This was so because we enjoyed all the insight in the physics of the example. In general this is not true and we may only use the dynamical equations of the system as mathematical facts. We thus have to come up with some proper *abstract/mathematical* energy function by imagination and trial and error. This is in fact the weak point of Lyapunov: the problem of establishing stability is replaced by the problem of finding a good energy function.

Lyapunov defines a proper energy function $V = V(x)$ as follows:

1. In an area Ω of the state space which includes the origin the function V is unique, continuous and has continuous first partial derivatives.

2. $V(0) = 0$.

3. For $x \neq 0$, $V(x) > 0$ and if $\|x\| \to \infty$ (if allowed still in Ω) then $V(x) \to \infty$

Condition 1 is clearly to facilitate computations. Condition 2 is just for scaling and guaranteeing that the minimum is over there in combination with condition 3 which says that the function is positive definite. If this is true then in a certain region about the

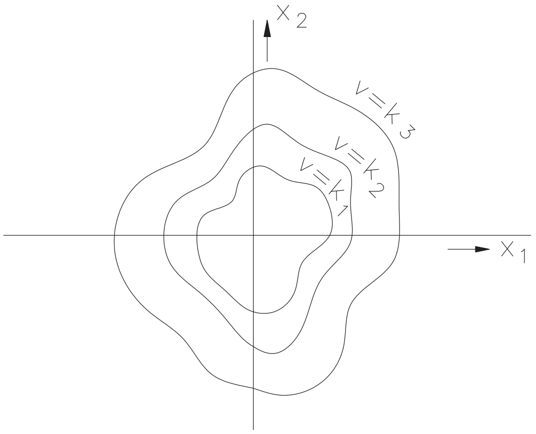


Figure 2.4: Equilevel or iso-energy lines of V.

origin (not necessarily full Ω as we will see!) there will be successively inclusive equilevel lines of increasing positive values k_i as indicated in Fig. 2.4 for $k1 < k2 < k3 \ldots$

In the example of the pendulum the kinetic plus potential energy is indeed fulfilling these conditions if we define Ω by $-2\pi < x_1 < 2\pi$ and $-\infty < x_2 < \infty$ where we precisely excluded the neighbouring minima for $(x1, x2) = (\pm 2\pi, 0)$. We could have chosen an Ω as big as the whole space exclusive all minima $(x1, x2) = (\pm 2k\pi, 0)$ for $k = 1, 2, 3, \ldots$. For the subarea which shows successively including equilevel lines we have to exclude all areas within the separator except the "valley" around $(0, 0)$. We now have:

$$k_i = \frac{1}{2} ml^2 x_2^2 + mgl(1 - \cos(x_1)) \Rightarrow \tag{2.27}$$

$$\frac{1}{2}\frac{l}{g} x_2^2 = \cos(x_1) + (\frac{k_i}{mgl} - 1) \tag{2.28}$$

which indeed corresponds to the explicit description of the trajectories we derived before in section 1.4.3. Now (local) stability can be proved if:

- **4**. $dV/dt \leq 0$

This fourth condition makes the function V a so called Lyapunov function for that subarea Ω_0 where it holds.

For the damped pendulum we see that as expected for the subarea $\Omega_0 : -\pi < x1 < \pi$ the derivative in time is indeed negative:

$$\dot{x}_1 = x_2 \qquad \dot{x}_2 = -\frac{g}{l}\sin(x_1) - \frac{d}{ml^2} x_2 \tag{2.29}$$

$$\dot{V} = ml^2 x_2 \dot{x}_2 + mgl \sin(x_1) \dot{x}_1 \qquad \Rightarrow \tag{2.30}$$

$$\dot{V} = -mglx_2 \sin(x_1) - x_2^2 d + mglx_2 \sin(x_1) \qquad \Rightarrow \tag{2.31}$$

$$\dot{V} = -dx_2^2 \leq 0 \tag{2.32}$$

The negative term is precisely the power lost in the friction and the system will converge to a point of minimum potential energy. This is not necessarily the point $(0, 0)$ as for bigger values of x_2 there will be first some rotations before sufficient energy is dissipated so that swinging sets in and ultimate convergence to $(x_1, x_2) = (2k\pi, 0)$ $k > 0$. So if we want to guarantee the domain for which the convergence is to $(0, 0)$ we have to prohibit that the trajectories leave Ω. This brings us to the last condition the Lyapunov function:

- **5** The domain of guaranteed convergence Ω_1 is bounded by $V(x) < c$.

The choice of domain Ω_1 for the pendulum is trivially given by the domain included in the separator around $(0, 0)$. Only incoming trajectories are then seen as is clear from Fig. 2.3. So Ω_1 is like a black hole: no trajectories are allowed at the border that point outwards. So the theorem says: *If V is a Lyapunov function on Ω_1 then all trajectories with initial conditions within Ω_1 will be stable. If $\dot{V} < 0$ then $x = 0$ is convergence point.* So Ω_1 is a guaranteed domain of convergence to $x = 0$. The actual domain of convergence is greater because all trajectories that enter Ω_1 will belong to the convergence domain. For the damped pendulum also all trajectories coming in from the top left (see Fig. 2.3) will satisfy this condition.

The proof of the theorem is based on Fig. 2.5.

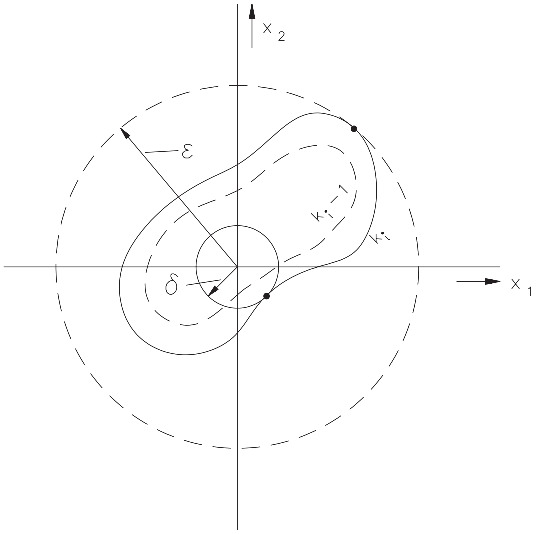


Figure 2.5: Proof of Lyapunov theorem.

Consider the ball ε: $\|x\| < \varepsilon$ where we have taken the 2-norm (Euclidean norm) in Fig. 2.5. Let the minimum value for V on this boundary be k_i. The isolevel line $V = k_i$ will be contained in the ball ε. We can now choose a ball $\delta : \|x(0)\| \leq \delta$ which is contained in $\{\forall x : V(x) \leq k_i\}$. So each trajectory starting in ball δ will stay in ball ε because if $\|x(0)\| < \delta$ then $\|V(x(0))\| \leq k_i$ and since $\dot{V} \leq 0$ it will hold that $V(x) \leq k_i$ for all $t \geq t_0$ so that $\|x\| \leq \varepsilon$. So it is stable. If $\dot{V} = 0$ at some point $x \neq 0$ trajectories may possibly stop at those points. So absolute convergence to $x = 0$, i.e. *asymptotic stability*, is only obtained for $\dot{V} < 0$. In that case all trajectories starting within Ω_1 will lead to lower V and necessarily have to end in $x = 0$. On the other hand if $\dot{V} > 0$ the domain will certainly be *unstable*. We could easily visualise this in two dimensions but of course it also holds for higher dimensions of x.

Examples of V are:

- $V = x_1^2 + x_2^2 + x_3^4 > 0$ in $\mathbb{R}^3$
- $V = x_1^2 + 2x_1x_2 + 3x_2^2 + x_3^2 > 0$ in $\mathbb{R}^3$
- $V = x_1^2 + x_2^2 + x_3^2 - x_3^3 > 0$ in $\Omega \subset \mathbb{R}^3$!

- $V = x_1^2[(x_1-3)^2+1] > 0$ in $\mathbb{R}^1$

The time derivative $\dot{V}$ depends on the process dynamics. Let for the last example hold that $\dot{x_1} = -x_1$. This system is certainly stable as it has a pole in -1. The last V-function then yields:

$$\dot{V} = \frac{dV}{dx_1}\frac{dx_1}{dt} = \frac{d(x_1^4 - 6x_1^3 + 10x_1^2)}{dx_1}\dot{x}_1 = (4x_1^3 - 18x_1^2 + 20x_1)(-x_1) \tag{2.33}$$

$$= -4x_1^2(x_1-2)(x_1-2.5) < 0 \quad \text{for} \quad (-\infty < x_1 < 2) \vee (2.5 < x_1 < \infty) \tag{2.34}$$

So we have:

$\Omega \,:\, -\infty < x_1 < \infty$

$\Omega_0 \,:\, (-\infty < x_1 < 2) \vee (2.5 < x_1 < \infty)$

$\Omega_1 \,:\, -.732 < x_1 < 2$ with $V(-.732) = V(2) = 8$

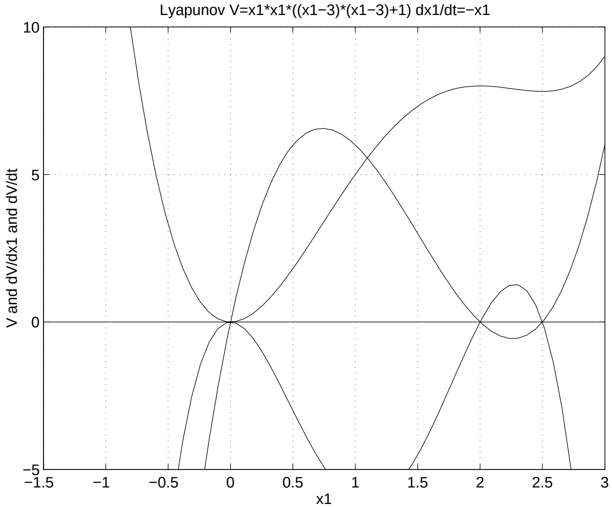


Figure 2.6: A Lyapunov function and its derivatives.

Fig. 2.6 illustrates why this is by showing V, $dV/dx1$ and dV/dt. Now it is easy to show that $V = x_1^2$ is a much better Lyapunov function for $\Omega_i = \mathbb{R}$, but it shows at least that a Lyapunov function is not unique. This is fortunate as the Lyapunov function has to be found by trial and error!

2.4 The positive definite function x^TPx.

A function which is often used and found valuable is simply: $V = x^TPx$ where matrix P is positive definite. Only the symmetric part of P is relevant as the skew-symmetric component does not contribute to V which can readily be seen from the following example:

$$\begin{cases} x^TPx = x^TP_{symmetric}x + x^TP_{skewsymmetric}x \qquad \text{e.g.} \\ \begin{pmatrix} x_1 & x_2 \end{pmatrix} \begin{pmatrix} 1 & 3 \\ 1 & 4 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \\ = \begin{pmatrix} x_1 & x_2 \end{pmatrix} \begin{pmatrix} 1 & 2 \\ 2 & 4 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} x_1 & x_2 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \qquad \text{or} \\ (x_1^2 + 4x_2^2 + (1+3)x_1x_2) = (x_1^2 + 4x_2^2 + (2+2)x_1x_2) + ((1-1)x_1x_2) \end{cases} \tag{2.35}$$

So it makes sense to discuss only *symmetric* matrices P.

For recapitulation from mathematics on positive (and nonnegative) definiteness the next intermezzo:

INTERMEZZO:

The following predicates on a symmetric matrix $P\epsilon\mathbb{R}^{nxn}$ are equivalent:

1. P is positive (nonnegative) definite [notation: p.d. or > 0 (nonn.d. or ≥ 0)]
2. $\forall x\epsilon\mathbb{R}^n, x \neq 0 : x^TPx > (\geq)0$
3. eigenvalues $\lambda_i(P)\epsilon\mathbb{R}$ and $\lambda_i > (\geq)0$ for $i = 1, 2, \ldots n$.
4. singular values $\sigma_i(P) > (\geq)0$, left and right singular vectors are the same.
5. $P = R^TR$ with R a nonsingular matrix (R may be singular).
6. all diagonal submatrices P_i have determinant $> (\geq)0$.

COMMENTS:

1. is simply the verbal definition
2. is the reason we use it as V
3. an easy way to establish it by e.g. MATLAB

   as a consequence we have: det(P) $= \Pi\lambda_i > (\geq)0$ and trace$(P) = \Sigma\lambda_i > (\geq)0$
4. idem, in fact $\sigma_i = \lambda_i$ and the singular vectors are eigenvectors
5. an easy way to construct a P.
6. theorem of Sylvester: a fast way to check it for small *symmetric* matrices P by hand (in exams!) e.g. for the above example: det(P_1) = det(1)> 0, det(P_2)= det(P) = 1*4 - 2*2 = 0 $\Rightarrow P \geq 0$ (P is nonnegative definite). (P actually has eigenvalues (and singular values) 5 and 0)

END INTERMEZZO

For $V = x^TPx$, $P > 0$, at least the first three conditions for a Lyapunov function are satisfied irrespective of the underlying process. The 4th (and 5th) condition depends on the dynamics of the process. If we apply it to a linear process the fifth condition is irrelevant because for linear systems the local and the global characteristics are the same. Let us see how the application to a linear process develops:

$$\dot{x} = Ax \qquad V = x^T Px \qquad \Rightarrow \tag{2.36}$$

$$\dot{V} = x^T P\dot{x} + \dot{x}^T Px = x^T (A^T P + PA)x \overset{\text{def}}{=} -x^T Qx \leq 0? \tag{2.37}$$

So the nonnegative definiteness of Q makes V a Lyapunov function where Q satisfies the so-called Lyapunov equation:

$$Q = -(A^T P + PA) \tag{2.38}$$

Note that Q is automatically symmetric! Straightforwardly one would thus choose a P, compute the corresponding Q and test it on its nonnegativity. It turns out that we then rarely find a Lyapunov function. The opposite way is very effective though: i.e. choose a nonnegative (or positive) definite matrix Q, compute the corresponding P (which is harder than the opposite!) and test P on its positive definiteness. In this last order we first guarantee that $\dot{V} \leq 0$ and then check whether V is p.d. everywhere. Let us take as an example the linearised equations of the damped pendulum about $(\varphi, \dot{\varphi}) = (0, 0)$ as introduced in section 1.4.2:

$$\begin{cases} A = \begin{pmatrix} 0 & 1 \\ -\frac{g}{l} & -\frac{d}{ml^2} \end{pmatrix} \qquad \Rightarrow \qquad Q = I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = \\ = \left(\begin{pmatrix} 0 & -\frac{g}{l} \\ 1 & -\frac{d}{ml^2} \end{pmatrix} \begin{pmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{pmatrix} + \begin{pmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{pmatrix} \begin{pmatrix} 0 & 1 \\ -\frac{g}{l} & -\frac{d}{ml^2} \end{pmatrix} \right) = \\ \begin{pmatrix} \frac{g}{l}(p_{12} + p_{21}) & \frac{g}{l}p_{22} - p_{11} + \frac{d}{ml^2}p_{12} \\ \frac{g}{l}p_{22} - p_{11} + \frac{d}{ml^2}p_{21} & 2\frac{d}{ml^2}p_{22} - p_{12} - p_{21} \end{pmatrix} \end{cases} \tag{2.39}$$

Each entry in Q leads to a linear equation in p_{ij} so that we get with Sylvester:

$$\begin{cases} P = \frac{1}{2} \begin{pmatrix} \frac{ml^2}{d}\left(1 + \frac{g}{l}\right) + \frac{d}{ml^2}\frac{l}{g} & \frac{l}{g} \\ \frac{l}{g} & \frac{ml^2}{d}\left(\frac{l}{g} + 1\right) \end{pmatrix} \\ \det(p_{11}) = \frac{ml^2}{d}\left(1 + \frac{g}{l}\right) + \frac{d}{ml^2}\frac{l}{g} > 0 \\ \det(P) = \frac{ml^2}{d}\left(\frac{l}{g} + 1\right)\left(\frac{ml^2}{d}\left(1 + \frac{g}{l}\right) + \frac{d}{ml^2}\frac{l}{g}\right) - \frac{l^2}{g^2} = \\ = \frac{m^2 l^4}{d^2}\left(\frac{l}{g} + \frac{g}{l} + 2\right) + \frac{l}{g} > 0 \qquad \Rightarrow \qquad P > 0 \end{cases} \tag{2.40}$$

So that indeed $V = x^T Qx$ is a Lyapunov function and $\dot{x} = Ax$ is an asymptotically stable system. For $g/l = 10$ and $d/ml^2 = .5$ both the equilevel lines (dotted) of the Lyapunov function $V = x^T Px$, when $Q = I$, and a representative trajectory are shown in Fig. 2.7.

It can be well observed that when the trajectory intersects with the equilevel lines the direction is inwards and thus downwards.

The opposite way, where we choose P first, will generally fail. For instance take $P = I$. The equilevel lines (full) are then circles in Fig. 2.7 though visually ellipses because of the scalings of the axes. We now remark that also outward (=upward) intersections with the trajectory occur. Consequently from this plot we already conclude that $\dot{V}$ is not everywhere less than zero or equivalently P will not be positive definite. In formula:

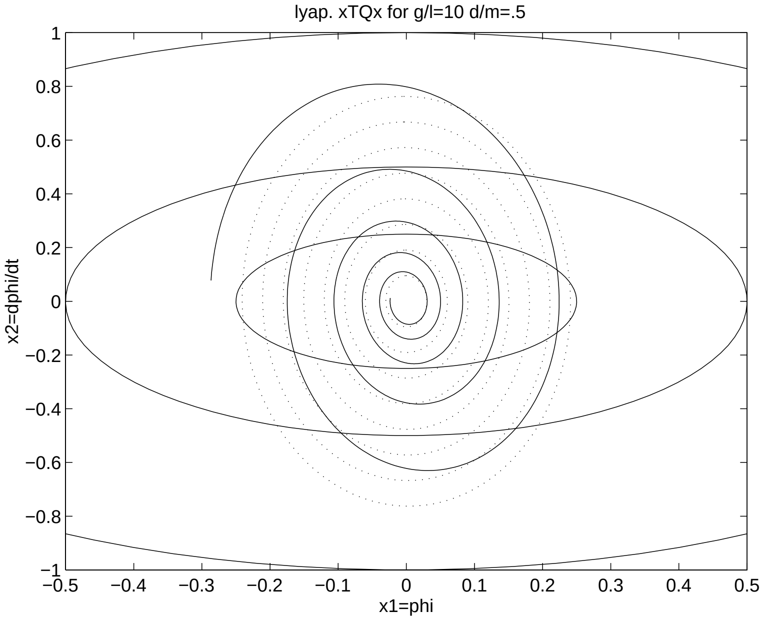


Figure 2.7: Lyapunov function of linearised damped pendulum.

$$Q = -(A^T + A) = \begin{pmatrix} 0 & \frac{g}{l} - 1 \\ \frac{g}{l} - 1 & \frac{2d}{ml^2} \end{pmatrix} \Rightarrow \tag{2.41}$$

$$q_{11} = 0 \qquad \det(Q) = -\left(\frac{g}{l} - 1\right)^2 \leq 0 \tag{2.42}$$

So for general values of g, l, d and m the matrix Q will certainly not be nonnegative definite unless the exceptional case $g/l = 1$. It happened that we have chosen this particular value in section 2.3.

So it seems better first to guarantee that $\dot{V} \leq 0$ by taking $Q > 0$ and then checking the corresponding V by solving P from the Lyapunov equation. It can even be proved that the answer obtained in that way is unambiguously:

$$\{A \textbf{ stable }\} \Leftrightarrow \{(Q = R^T R \textbf{ and } (A, R) \textbf{ observable }) \Rightarrow (P > 0)\}.$$

We take this theorem for granted and let us not be distracted by the formal proof.

For nonlinear systems there is generally no evident way of first establishing that $\dot{V} \leq 0$ so that we seem to be forced to start with choosing P positive definite yielding:

$$\dot{x} = g(x) \qquad V = x^T P x \qquad \Rightarrow \tag{2.43}$$

$$\dot{V} = \dot{x}^T P x + x^T P \dot{x} = g(x)^T P x + x^T P g(x) \leq 0? \tag{2.44}$$

Indeed a lot of trial and error can lead to a solution. As an example can serve:

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} x_2 \\ -x_1 - k_1 x_2 - k_2 x_2^3 \end{pmatrix} \quad k_1 > 0, k_2 > 0 \quad P = I \quad \Rightarrow \tag{2.45}$$

$$\dot{V} = \frac{d(x_1^2 + x_2^2)}{dt} = 2(x_1\dot{x}_1 + x_2\dot{x}_2) = -2(k_1 x_2^2 + k_2 x_2^4) \leq 0 \tag{2.46}$$

In this case we can indeed establish the global stability of the example system. However, this method often fails and a good alternative is offered by Krasovskii, that actually tries to accomplish for nonlinear systems what we exercised for linear systems. That is, first establish the negativity of $\dot{V}$ and then analyse positivity of a corresponding P-matrix.

2.5 Krasovskii's method.

The principle of this method is very simple: instead of taking the weighted squared states x, use the weighted squared derivatives of the states, i.e. $\dot{x} = g(x)$:

$$V = \dot{x}^T P \dot{x} = g(x)^T P g(x) \qquad P > 0 \tag{2.47}$$

For the computation of the derivative $\dot{V}$ we need the second derivative in time of the state vector which can be expressed as a function of the known first derivative by means of the Jacobi matrix J (Jacobian) that we have encountered before in section 1.3:

$$\ddot{x} = \frac{\partial g(x)}{\partial x^T}\dot{x} = J(x)\dot{x} \tag{2.48}$$

So that we get

$$\dot{V} = \ddot{x}^T P \dot{x} + \dot{x}^T P \ddot{x} = \dot{x}^T J^T P \dot{x} + \dot{x}^T P J \dot{x} = \tag{2.49}$$

$$= \dot{x}^T (J^T P + PJ)\dot{x} = -\dot{x}^T Q \dot{x} \leq 0 \quad \text{iff} \quad Q \geq 0 \tag{2.50}$$

So we have a similar situation as before for in the Lyapunov equation for linear systems where J takes the role of the state matrix A. Indeed in a linearisation J would be the state matrix in a fixed point. In the general case, we consider now, J needs not to be a constant matrix but may be a general function of the state x. Another essential difference is that in the linear case the (in)stability is always global so that P and cP with any $c > 0$ yields the same result. This certainly needs not to be true for nonlinear systems so that some trial and error with c can be useful. What is important is that again we can start with taking care that Q is positive definite thereby defining $\dot{V} < 0$ and next analysing whether P is positive definite (i.e. $V > 0$) again, which was shown to be a preferable order. For the last example we can apply Krasovskii:

$$\left\{ \begin{array}{l} \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} x_2 \\ -x_1 - k_1 x_2 - k_2 x_2^3 \end{pmatrix} \quad k_1 > 0, k_2 > 0 \quad V = \dot{x}^T P \dot{x} \\ \Rightarrow J = \begin{pmatrix} 0 & 1 \\ -1 & -(k_1 + 3k_2 x_2^2) \end{pmatrix} \quad \text{take} \quad Q = cI = \begin{pmatrix} -c & 0 \\ 0 & -c \end{pmatrix} = \\ \begin{pmatrix} -2p_{12} & p_{11} - p_{22} - p_{12}(k_1 + 2k_2 x_2^2) \\ p_{11} - p_{22} - p_{12}(k_1 + 3k_1 + 3k_2 x_2^2) & 2p_{12} - 2p_{22}(k_1 + 3k_2 x_2^2) \end{pmatrix} \end{array} \right. \tag{2.51}$$

for $c = 2$ we can easily solve P and test the positive definiteness:

$$P = \begin{pmatrix} \frac{2}{k_1+3k_2x_2^2} + (k_1 + 3k_2x_2^2) & 1 \\ 1 & \frac{2}{k_1+3k_2x_2^2} \end{pmatrix} \quad \Rightarrow \tag{2.52}$$

$$\begin{cases} p_{11} = \frac{2}{k_1+3k_2x_2^2} > 0 \\ \det(P) = \frac{4}{(k_1+3k_2x_2^2)^2} + 1 > 0 \end{cases} \tag{2.53}$$

$$\Rightarrow P > 0 \tag{2.54}$$

so that the system is indeed stable, even globally as no constraints on x were necessary. Finally: If a Lyapunov function cannot be found the question of stability is still open.

2.6 Second method of Lyapunov for discrete time systems.

For discrete time systems the second method applies with only small adaptations:

$$x(k+1) = g(x(k)) \tag{2.55}$$

1. $0\epsilon\Omega$: $V(x)$ is unique, continuous with continuous derivatives.
2. $V(0) = 0$
3. For $x \neq 0$: $V(x) > 0$; if $\|x\| \to \infty$ then $V(x) \to \infty$.
4. $\Delta V(x(k)) = V(x(k+1)) - V(x(k)) \leq 0$ for $x(k)\epsilon\Omega_0$.
5. The domain of guaranteed convergence Ω_1 is bounded by $V(x) < c$.

For linear systems the quadratic Lyapunov function is appropriate again:

$$V(x(k)) = x(k)^T P x(k) \qquad \Rightarrow \tag{2.56}$$

$$\Delta V(x(k)) = x(k+1)^T P x(k+1) - x(k)^T P x(k) = \tag{2.57}$$

$$= x(k)^T A^T P A x(k) - x(k)^T P x(k) = \tag{2.58}$$

$$= x(k)^T (A^T P A - P) x(k) \overset{\text{def}}{=} -x(k)^T Q x(k) < 0? \tag{2.59}$$

so that the Lyapunov equation for discrete time systems is given by:

$$Q = -(A^T P A - P) \tag{2.60}$$

For certain discrete time systems a *norm* on the state x can be a proper Lyapunov function sometimes, so: $V = \|x(k)\|$. If the system equation is a contraction i.e.

$$x(k+1) = g(x(k)) \qquad \text{while} \qquad \|g(x)\| < \|x\| \tag{2.61}$$

then it is easy to show that V is a Lyapunov function because:

$$V = \|x\| > 0 \tag{2.62}$$

$$\Delta V(x(k)) = V(g(x(k))) - V(x(k)) = \|g(x(k))\| - \|x(k)\| < 0 \tag{2.63}$$

Chapter 3

Introduction to ”optimal control”.

3.1 Introduction.

The previous chapters were mainly concerned with the *analysis* of nonlinear systems in state space both in continuous and in discrete time. In this chapter the *synthesis* of control systems for the analysed, nonlinear processes will be studied. The aim is to force the states to follow a prescribed trajectory or at least to let the state trajectory converge to a prescribed equilibrium point. At the same time the costs of the control effort itself, mainly resulting from system actuators, will be taken into account. As far as the sensors are concerned it is supposed that these can be installed at will in such a way as to measure all states with infinite accuracy and without costs. Problems, raised by insufficient or inaccurate sensors, will thus be neglected here. However, attention will be paid to these effects in the next chapters.

Control design is called ”optimal control” when a predefined criterion is optimised. By no means can you interpret it as being the ultimate, optimal control among all possible controllers. Optimality is just with respect to the criterion at hand and the real performance depends on the suitability of the chosen criterion. An example of such a criterion is for instance the often used, quadratic, integral cost function:

$$J = \int_{t_0}^{t_\infty} \{(x(t) - x_{ref}(t))^T Q (x(t) - x_{ref}(t)) + u(t)^T R u(t)\} dt \quad Q \geq 0 \quad R > 0 \tag{3.1}$$

where x and u are the finite dimensional state and control input of the dynamical system under study. The deviation of x from the reference trajectory x_{ref} is penalised quadratically with a *nonnegative* (symmetric) weighting matrix Q in order to reflect different weights attached to different state components or products. A zero eigenvalue of Q is allowed because some states or combination of states may not be penalised. At the same time the input u is also quadratically weighted with positive definite (symmetric) matrix R in order to keep all inputs within the range that is allowed by the particular actuator for each component. The integration is from t_0 to t_f but these need not be fixed. We may require e.g. that at t_f the state has reached a certain working point and make the moment of arrival t_f subordinate to this constraint. More about this will be presented in section 3.2.

The optimisation refers to minimising the cost as represented by the criterion J under the ”constraint” of the system dynamics:

$$\dot{x} = f(x, u, t) \tag{3.2}$$

So it is an optimisation in the context of the system dynamics: "dynamic optimisation".

If the dynamics of the system are neglected (i.e. by putatively taking $\dot{x} = 0$) the integral in the criterion is superfluous too so that the integrant remains and we just have to solve a static optimisation problem. This is (was) frequently done for chemical processes where control is restricted to sustain the equilibrium condition:

$$\dot{x} = 0 = f(x, u, t) \tag{3.3}$$

where $x_{ref}(t) = x_d$ is the desired equilibrium state for all t. From this equation the necessary control can be computed possibly in combination with a penalty function J which incorporates the costs on $u(t) = u_d$. Let this be illustrated by the example6 of the population growth as introduced insection 1.3.5. The dynamics are governed by:

$$x(k+1) = rx(k) - rx(k)^2 + u(k) \tag{3.4}$$

The total number of individuals x in a population of generation $k + 1$ is proportional to the population number of the previous generation which is reflected in the first term. The second term represents the loss of individuals due to starvation in a limited habitat, by civil war deaths caused by limited living space, pollution effects, epidemics etc. On top of that we have added a control input enforced by governmental laws concerning birth control, abortion, euthanasia, medical care... (In any democratic decision making process it makes sense to view u as a *state* feedback control policy.) Let us suppose that we deal with a parameter $r = 3$ so that without control ($u = 0$) the population number tends to oscillate (normed about ±.68) very close to the maximum number possible (normed ±.75) in the habitat as can be seen from Fig. 1.34. This is a very unpleasant situation as living close to the maximum implies that the second term is large which does not really contribute to the well-being of the individuals. So the government might take the plan to stabilise the population on the normed equilibrium of say $x_d = .25$ and ask the minister of welfare to design a controller to accomplish this but with the usually limited costs. So the civil servants at the ministry set up the following cost criterion:

$$J = (x - \frac{1}{4})^2 + \beta u^2 \tag{3.5}$$

where the first term represents the goal and the second term is to penalise the cost the minister is going to make by implementing his control. Furthermore the minister is tied by the constraint of (non-dynamic) equilibrium:

$$x = 3x - 3x^2 + u \tag{3.6}$$

that he derives by simply putting $x(k + 1) = x(k) = x$ in the dynamic difference equation. He then properly includes this constraint by a Lagrange multiplier in the general criterion to be minimised (not too bad for a minister):

$$H = (x - .25)^2 + \beta u^2 + \lambda(-x + 3x - 3x^2 + u) \tag{3.7}$$

Apparently, either the minister did ever study himself or one of his civil servants so that they could solve this as follows:

$$\begin{cases} \frac{\partial H}{\partial x} = 0 & \Rightarrow \quad 2x - .5 + 2\lambda - 6\lambda x = 0 \\ \frac{\partial H}{\partial u} = 0 & \Rightarrow \quad 2\beta u + \lambda = 0 \\ constraint: & 2x - 3x^2 + u = 0 \end{cases} \tag{3.8}$$

For each β a solution can be found for x, λ and u in principle. Action groups from the population (one of the cost components!) compel the minister to limit the obviously very impopular law to $u \geq -.25$. It is clear that the control action has to be negative and thus there is an lower bound here. This lower bound will lead most closely to desired equilibrium (can be checked) so the minister puts u equal to $-.25$ and lets the weighting β follow from that choice. Combined with above equations this leads to the solution:

$$u = -\frac{1}{4} \quad x = \frac{1}{6} \quad \lambda = \frac{1}{6} \quad \beta = \frac{1}{3} \tag{3.9}$$

Notice that the weighting factor β is adjusted a posteriori based upon strict bounds on u which will be common practice as we will see later. So the minister reports that he can do no more than this. Though the king complains that he then has even less citizens than planned (viz. 1/6 instead of 1/4) the government agrees and the law is set to work. To the astonishment of the government and the pleasure of the king it happens that the population stabilises in no time on $x = .5$! Based upon what we learned in section 1.3.4, the effect can easily be analysed by drawing the Picard diagram as shown in Fig. 3.1.

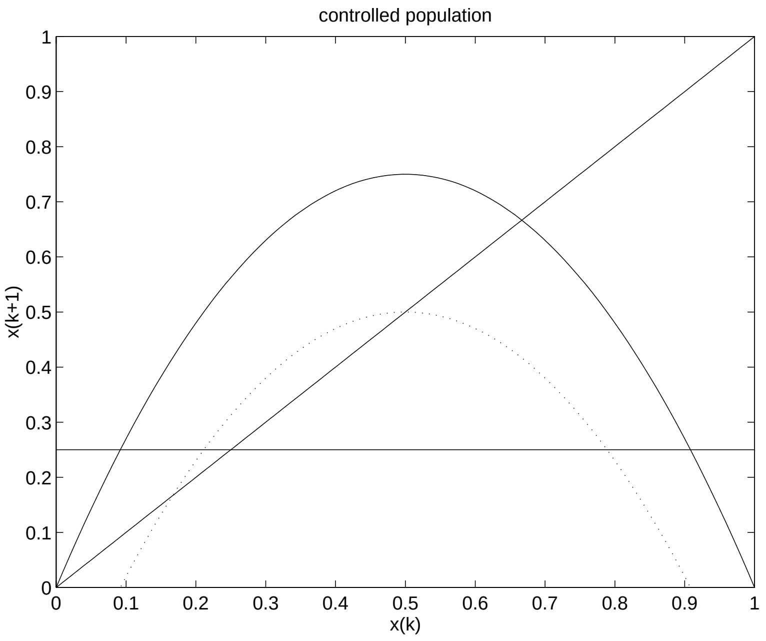


Figure 3.1: Controlled population curves.

The full line represents the population grow function without control so $u = 0$. After implementation of the law $u = -.25$ the grow curve is just shifted by $-.25$ and we observe two equilibrium points. The point $x = 1/6$ indeed is the solution of the minimisation procedure as it is closest to the desired $x_d = 1/4$ but it is certainly unstable as the derivative is greater than one. (If we speak about the derivative we again consider the dynamic situation which is the real thing going on at all times whatever static model the ministry might make of it.) The false solution $x = 1/2$ though happens to be stable and convergence is very fast as the derivative is zero. In the parliament from the opposition an old student from Eindhoven university (very rare because of previous malfunctioning measures by the government) who had studied section 1.3.5 proposed a linearisation by feedback to arrive at the system:

$$x(k+1) = \frac{1}{4} \qquad \Rightarrow \tag{3.10}$$

$$u(k) = -3x(k) + 3x(k)^2 + .25 \tag{3.11}$$

which yields the horizontal line in Fig. 3.1. By that dynamic control the population would be steady once and for all in one generation. The civil servants at the ministry immediately computed that for the average uncontrolled population of $.68 = x(1)$ this would cost a control of $u(1) = -.43$ which is far outside the limit. The ex-student defended his proposal by pointing out that this was just the cost in the first iteration and after the first generation the costs would turn into $u(k) = -.31$ for all future k. (Note that costs of control are measured in a norm like $\|u\|_2$. Here we indeed had the term $\beta u^2 = \beta \|u\|_2^2$.) Nevertheless the government abandoned the proposal in view of the forthcoming elections. The anger of the voters, raised by the initial, impopular measure $u(1) = -.43$, would let the opposition win and in the next term of office they would benefit from the effect i.e. $x(k) = .25$ so the people would accept the state control $u(k) = -.31$. The reader is invited to study what the minister would have caused if, with some more courage, he could have obtained a constant control of $u = -1/3$.

Of course the example is grossly oversimplified and we have to deal with substantial model errors and disturbances like migration and unpredictable effects in economy. Nevertheless the example illustrates that more study of the control of nonlinear dynamical systems is indispensable, despite of the costs, for any government. The techniques that have been used for the optimisation of the above *static* approximation will be extended for the *dynamical* case. These techniques cover the use of Lagrange multipliers and general methods to minimise nonlinear functions or solving nonlinear equations. The tools for this, like (conjugated) gradient methods, Newtonian algorithms and even linear programming (for linear criteria and constraints), are taught in several other courses and are supposed to be known here.

3.2 Theory.

In the previous section an example was given of a criterion J to be minimised which took the form of an integral. Often, such a criterion is extended with terms that penalise the states and inputs explicitly at the moments t_0 and t_f . Sometimes even one is only interested in the costs at these moments and not in the cumulative (or integrated) effects during the time span between t_0 and t_f. Hence we can categorise the criteria, named after their first users:

$$\begin{array}{ll} Lagrange: & J = \int_{t_0}^{t_f} F(x,u,t)dt \\ Bolza: & J = \int_{t_0}^{t_f} F(x,u,t)dt + \phi_0(x(t_0),u(t_0),t_0) + \phi_f(x(t_f),u(t_f),t_f) \\ Mayer: & J = \phi_0(x(t_0),u(t_0),t_0) + \phi_f(x(t_f),u(t_f),t_f) \end{array} \tag{3.12}$$

Fortunately, there is no substantial difference between the various criteria because we can transform the one into the other as follows. It is easy to redefine the functions at the times t_0 and t_f in one function:

$$\phi_0(x(t_0),u(t_0),t_0) + \phi_f(x(t_f),u(t_f),t_f) = \Phi(x,u,t)|_{t_0}^{t_f} \tag{3.13}$$

e.g. by:

$$\Phi(x,u,t) = \{\phi_0(x(t),u(t),t)(t_f - t) + \phi_f(x(t),u(t),t)(t - t_0)\}/(t_f - t_0) \tag{3.14}$$

On the other hand we may also redefine :

$$F(x,u,t) = \frac{d\Phi(x,u,t)}{dt} \tag{3.15}$$

so that we conclude:

$$\int_{t_0}^{t_f} F(x,u,t)dt = \int_{t_0}^{t_f} \frac{d\Phi(x,u,t)}{dt}dt = \int_{t_0}^{t_f} d\Phi(x,u,t) = \Phi(x,u,t)|_{t_0}^{t_f} \tag{3.16}$$

Ergo the Lagrange problem can be transformed to the Mayer problem and vice versa. If we just transform parts we can arrive at the Bolza problem. In the remainder of this chapter we will treat the Bolza problem as the most general and direct one (without transformations) and take the time boundary function Φ only dependent on $x(t)$ and t:

$$J = \int_{t_0}^{t_f} F(x,u,t)dt + \Phi(x,t)|_{t_0}^{t_f} \tag{3.17}$$

The optimal trajectory $x(t)$ and the necessary control $u(t)$ have to be found by minimising the criterion J but they should also satisfy a number of constraints. The most obvious constraint is of course the differential equation describing the dynamics of the process itself:

$$\dot{x} = f(x,u,t) \tag{3.18}$$

The (initial and final) states x may be fixed at the integration boundary time instances or not. So initial state $x(t_0)$ may be given as x_0 or not and final state $x(t_f)$ may be required to become a desired x_f or not. Next, the initial time t_0 may in principle be chosen dependent on an advantageous value of the state $x(t_0) = x_0$. This occurs for instance when we have to switch over a production process or a power plant. In this case it may have advantages to wait for the states to obtain an appropriate value for the start of the switching. In order not to complicate the formulas too much we will ignore such a possibility here. So t_0 will be fixed and most of the cases it will be defined as time 0. If this is not the case in a problem, you will be confronted with in future, the procedure, to handle it, is completely analogous to a free final time t_f which will be treated here.

If t_f is taken free, there should be given a constraint to fix t_f implicitly. The final time t_f is then depending on the final state $x(t_f)$ which is represented in a formalised implicit constraint:

$$g(x(t_f), t_f) = 0 \tag{3.19}$$

Such a final time t_f is then subordinated to the final, desired state x_f (or condition on this state). For e.g. ballistic systems or the coupling of a "Shuttle" with a "Saljuz" the exact time of the strike or the contact is not as important as the correctness of the hit and touch and the speeds at that moment. Consequently, in these applications, we require $x(t_f) = x_f$ so that we may put:

$$g(x(t_f), t_f) = x(t_f) - x_f = 0 \tag{3.20}$$

Summarising, the mathematical representation of the real world problem now is:
PROBLEM DEFINITION

$$\begin{array}{rll} \min_{x(t),u(t)} J: & J = \int_{t_0}^{t_f} F(x,u,t)dt + \Phi(x,t)|_{t_0}^{t_f} & \\ \text{under constraints}: & \dot{x} = f(x,u,t) & \\ \text{(optional)} & x(t_0) = x_0 & \\ \text{(optional)} & x(t_f) = x_f \quad \vee \quad g(x(t_f),t_f) = 0 & \end{array} \tag{3.21}$$

CALCULUS OF SOLUTION

Analogous to the static case we may add the constraints to the criterion function by means of Lagrange multipliers $\lambda(t)$ and η respectively where λ is a function of time as we have a time dependent constraint. This yields the generalised criterion J_g :

$$J_g = \int_{t_0}^{t_f} \{F + \lambda^T(f - \dot{x})\}dt + \Phi|_{t_0}^{t_f} + [\eta^T g]_{t_f} \tag{3.22}$$

For historical and analytical reasons the function:

$$H \overset{p.d.}{=} F + \lambda^T f \tag{3.23}$$

is called the *Hamiltonian* and it actually represents an energy function. For reasons of lack of time and space we will not go into details about H but will conform to the name convention in the literature. Furthermore, it will appear to be convenient to eliminate $\dot{x}$ and replace it by a term containing $\dot{\lambda}$ which can be done based on partial integration:

$$\int_{t_0}^{t_f} \frac{d(\lambda^T x)}{dt}dt = \lambda^T x|_{t_0}^{t_f} = \int_{t_0}^{t_f} (\dot{\lambda}^T x + \lambda^T \dot{x})dt \tag{3.24}$$

so that the generalised criterion becomes:

$$J_g = \int_{t_0}^{t_f} \{H + \dot{\lambda}^T x)\}dt + [\lambda^T x - \Phi]_{t_0} + [\Phi - \lambda^T x + \eta^T g]_{t_f} \tag{3.25}$$

According to variation calculus we define the optimal trajectory $\hat{x}(t)$, necessary input $\hat{u}(t)$ and final time $\hat{t}_f$ together with their small variations $\delta x(t), \delta u(t), \delta t_f$ by:

$$x(t) = \hat{x}(t) + \delta x(t) \qquad u(t) = \hat{u}(t) + \delta u(t) \qquad t_f = \hat{t}_f + \delta t_f \tag{3.26}$$

Next we develop a Taylor expansion about the optimal variables where we analyse just the first order terms. Two facts from mathematics have to be remembered therefor :

$$\text{if} \quad a \neq a(b) \qquad \text{then} \qquad \frac{\partial(a^T b)}{\partial b} = a \tag{3.27}$$

$$\int_{t_0}^{t_f+\delta t_f} Gdt = \int_{t_0}^{t_f} Gdt + [G]_{t_f}\delta t_f + H.O.T. \tag{3.28}$$

where H.O.T. stands for higher order terms. The full Taylor expansion now is:

$$J_g(x,u,t_f) = J_g(\hat{x},\hat{u},\hat{t}_f) + \int_{t_0}^{t_f} \{[\frac{\partial H}{\partial x} + \dot{\lambda}]^T \delta x + [\frac{\partial H}{\partial u}]^T \delta u\}dt + \tag{3.29}$$

$$+[\lambda - \frac{\partial \Phi}{\partial x}]_{t_0}^T \delta x(t_0) + [\frac{\partial \Phi}{\partial x} - \lambda + \frac{\partial g^T}{\partial x}\eta]_{t_f}^T (\delta x(t_f) + \dot{x}(t_f)\delta t_f) + \tag{3.30}$$

$$+[-\dot{\lambda}^T x]_{t_f}\delta t_f + [\frac{\partial \Phi}{\partial t} + \frac{\partial g^T}{\partial x}\eta]_{t_f}\delta t_f + [H + \dot{\lambda}^T x]_{t_f}\delta t_f + H.O.T. \tag{3.31}$$

This rather complicated expression can straightforwardly be obtained with the following comments. For the second term on the second line remember that the variations at moment t_f are given by:

$$[\delta J]_{t_f} = [\frac{\partial J}{\partial x(t_f)}]^T \delta x(t_f) + [\frac{\partial J}{\partial x(t_f)}]^T [\frac{dx}{dt}]_{t_f} \delta t_f + \frac{\partial J}{\partial t_f} \delta t_f + \tag{3.32}$$

$$+ [\frac{\partial J}{\partial \eta}]^T \delta \eta + [\frac{\partial J}{\partial \lambda(t_f)}]^T \delta \lambda(t_f) + [\frac{\partial J}{\partial \lambda(t_f)}]^T [\frac{d\lambda}{dt}]_{t_f} \delta t_f + H.O.T. \tag{3.33}$$

It is not necessary in the above equation to evaluate explicitly the variations $\delta\lambda(t)$ and specifically $\delta\lambda(t_f)$ nor $\delta\eta$ because this will only lead to the original, respective constraints. The variations in the last term, however, which relate to variations δt_f are relevant and it yields the first term on the third line of the Taylor expansion because:

$$\frac{\partial[\Phi - \lambda^T x + \eta^T g]}{\partial \lambda} = \frac{\partial[-\lambda^T x]}{\partial \lambda} \qquad \Rightarrow \tag{3.34}$$

$$\Rightarrow [\frac{\partial(-\lambda^T x)}{\partial \lambda}]^T_{t_f} \dot{\lambda}(t_f) \delta t_f = [-x^T \dot{\lambda}]_{t_f} \delta t_f = [-\dot{\lambda}^T x]_{t_f} \delta t_f \tag{3.35}$$

Finally the higher order terms (H.O.T.) vanish in the limit after we have taken the derivative with respect to all δx, δu and δt_f so that:

$$\lim_{(\delta x^T, \delta u^T, \delta t_f)^T \to 0} \frac{\partial J_g}{\partial(\delta x^T, \delta u^T, \delta t_f)^T} = 0 \tag{3.36}$$

constitutes the conditions that define $\hat{x}(t), \hat{u}(t), \hat{t}_f$. Surely the coefficients are nothing else than the coefficients of the first order terms in the Taylor expansion. Together with the constraints they yield the solution of the problem. Of course, it is possible that the extremum which is found this way, is not a minimum. It could also refer to a maximum or a saddle point. In order to discriminate minima we should also consider the second derivative which is highly complicated though. Fortunately, in practice the obtained solution is the proper one except for exceptional, anomalous cases.

SOLUTION

$$H \overset{p.d.}{=} F + \lambda^T f \tag{3.37}$$

Dynamics:

$$\begin{array}{llll} I & \text{constraint}: & \dot{x} = f(x,u,t) & process - dynamics \\ II & \delta x(t) \Rightarrow & \dot{\lambda} = -\frac{\partial H}{\partial x} & Euler - Lagrange \\ III & \delta u(t) \Rightarrow & \frac{\partial H}{\partial u} = 0 & Euler - Lagrange \end{array} \tag{3.38}$$

Conditions:

$$\begin{array}{llcl} IV & \text{constraint}: & x(t_0) = x_0 & initial - time \\ V & \text{constraint}: & [x(t_f) = x_f] \vee [g(x(t_f), t_f) = 0] & final - time \\ VI & t_0: & \delta x(t_0)^T [\lambda - \frac{\partial \Phi}{\partial x}]_{t_0} = 0 & transversality \\ VII & t_f: & [\delta x(t_f) + \dot{x}(t_f)\delta t_f]^T [\frac{\partial \Phi}{\partial x} - \lambda + \frac{\partial g^T}{\partial x} \eta]_{t_f} = 0 & transversality \\ VIII & \delta t_f \Rightarrow & [\frac{\partial \Phi}{\partial t} + H + \frac{\partial g^T}{\partial t} \eta]_{t_f} = 0 & transversality \end{array} \tag{3.39}$$

At a first sight the solution might look awkward but the interpretation is quite simple. The process dynamics together with the so called Euler-Lagrange equations define the complete *structure* of the process and the controller where λ is the state vector in the controller also called the co-state. Next the initial and final time conditions and the so called transversality conditions together provide just enough boundary conditions for x and λ to uniquely define the optimal trajectories as we will show. Let us first clarify the structure defined by equations I, II and III. Obviously I represents the process dynamics we started with. From II we obtain:

$$\dot{\lambda} = -\frac{\partial H}{\partial x} = -\frac{\partial F}{\partial x} - \frac{\partial f^T}{\partial x}\lambda \tag{3.40}$$

which is still a function of u. With the help of III:

$$0 = \frac{\partial H}{\partial u} = \frac{\partial F}{\partial u} + \frac{\partial f^T}{\partial u}\lambda \tag{3.41}$$

we may solve $u = s(x, \lambda, t)$. Substitution into 3.40 yields a first order differential equation for λ only dependent on x, λ and t, so formally we can write:

$$\dot{\lambda} = h(x, \lambda, t) \tag{3.42}$$

In a block scheme this works out as in Fig. 3.2.

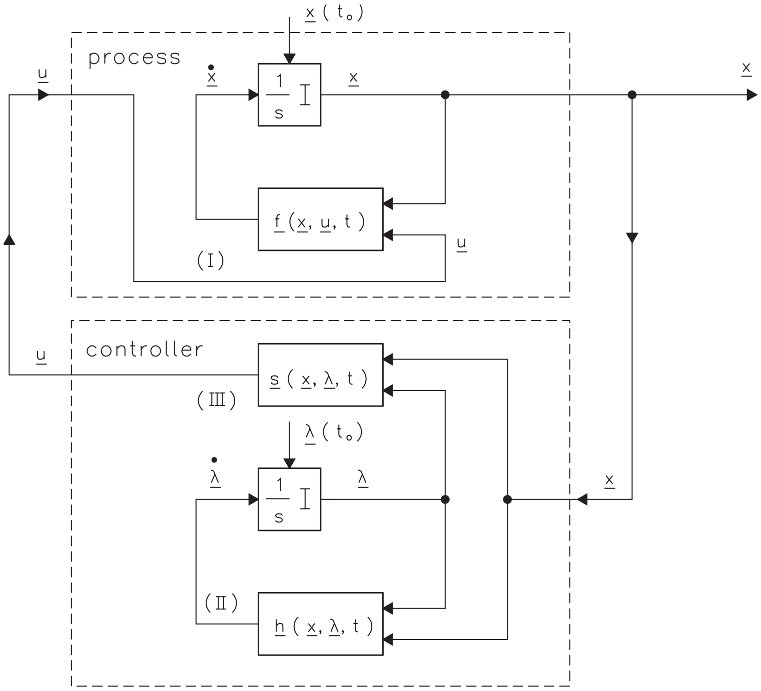


Figure 3.2: Structure optimal controller.

If we would know the initial states $x(t_0)$ and $\lambda(t_0)$ the optimal trajectories could easily be computed and the obtained controller be installed. The initial states have to be derived from conditions IV through VIII and this is the difficult part of the job. Generally condition IV is given. If not, condition VI comes into force where $\delta x(t_0) \neq 0$ in that case so that x and λ are related at t_0. Note that there is no explicit initial condition $\lambda(0)$. However, there are constraints at moment t_f which determine implicitly the final state $\lambda(t_f)$. E.g. the constraint V. If $x(t_f)$ is not given explicitly then condition VII comes into force where $\delta x(t_f) \neq 0$. If t_f is free, constraints V, VII and VIII have to be used in combination. Examples will be shown how to do so. Even if finally $\lambda(t_f)$ is available it is still troublesome to arrive at the proper trajectory because we then deal with a so called "two point boundary value problem" (TPBVP). We are given a set of first order differential equations where part of the state vector, i.e. x, is conditioned at initial time t_0 and the other part λ is fixed at the final time t_f. General methods are offered in other courses (e.g. numerical analysis) and we will confine to relatively simple cases where the solution is more or less obvious.

Time for examples.

3.3 Example inverted pendulum

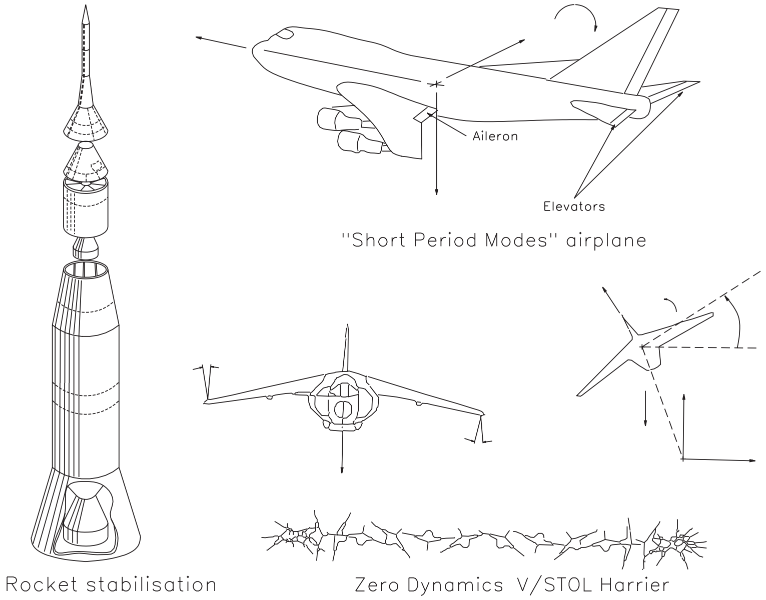


Figure 3.3: Examples of inverted pendulum dynamics.

In previous discussions about the inverted pendulum (example4) the process was autonomous i.e. no external input u was available. If we think of a pendulum mounted perpendicularly on the shaft of a servo motor we can control the torque of the shaft and thereby enforce an input u affecting the acceleration of the angle directly. One is tended to take this example as an artificial toy but it actually is a laboratory scale model of a process that frequently occurs in practice. As illustrated in Fig. 3.3 the same dynamics can be found in the stabilisation of a rocket by vectoring the thrust or in the tail-controlled missiles by fin deflection. For the missiles the gravity force has to be considered together with the "wind" forces and it is completely substituted by the wind in case of normal airplanes where we find the inverted pendulum dynamics again in the so-called "short period modes". Also part of the rolling behaviour of a V/STOL Harrier is exactly described by the dynamics of the inverted pendulum. In general, many processes which are in a meta-stable position share the local linearised dynamics with the inverted pendulum determined by two real poles mirrored with repect to the imaginary axis. So the study of the inverted pendulum is valuable. Consider again the familiar differential equations describing the pendulum dynamics that we extended with the introduced torque input u:

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} x_2 \\ -\frac{g}{l}\sin(x_1) - \frac{d}{ml^2}x_2 + bu \end{pmatrix} \qquad \begin{pmatrix} x_1 = \varphi \\ x_2 = \omega = \dot{\varphi} \end{pmatrix} \tag{3.43}$$

Let us suppose that we want to bring the pendulum in one second from the hanging to the upright position where another controller takes over to stabilise the pendulum in the upright position. Therefore it is not really necessary to exactly arrive at the upright position given by $(x_1, x_2) = (\pi, 0)$. We could try to come close, considering the cost of the control action. The criterion could then look like:

$$J = \frac{1}{2}\int_0^1 [(x_1 - \pi)^2 + \alpha x_2^2 + ru^2]dt + \frac{\gamma}{2}[(x_1 - \pi)^2 + x_2^2]_{t_f=1} \tag{3.44}$$

If we want to avoid saturation of the actuator, the servo motor in this case, we either have to bound the input u or to buy a very expensive high power and fast servomotor. The actuator cost is represented by the term ru^2. The term αx_2^2 weights the angle speed to protect the bearings against extreme centrifugal forces. Finally in the Φ-term the γ weights the relative importance of the final state so that the stabilising controller can take over. It is important to notice that another controller should take care for the stabilising task because the optimal controller obtained from above criterion will not be stabilising. The simple reason is that we just required an optimal behaviour in the time span between 0 and 1. Therefore, stability is not an issue and the controller will use precisely the effect of unstable dynamics to speed up the effect with minimum costs.

From the above process dynamics and criterion the Hamiltonian can easily be derived:

$$H = \frac{1}{2}[(x_1 - \pi)^2 + \alpha x_2^2 + ru^2] + \lambda_1 x_2 + \lambda_2[-\frac{g}{l}\sin(x_1) - \frac{d}{ml^2}x_2 + bu] \tag{3.45}$$

For the elimination of u we use equation III:

$$\frac{\partial H}{\partial u} = ru + \lambda_2 b = 0 \qquad \Rightarrow \qquad u = -\frac{b}{r}\lambda_2 \tag{3.46}$$

so that equation II yields:

$$\begin{pmatrix} \dot{\lambda}_1 \\ \dot{\lambda}_2 \end{pmatrix} = \begin{pmatrix} -(x_1 - \pi) \\ -\alpha x_2 \end{pmatrix} - \begin{pmatrix} 0 & -\frac{g}{l}\cos(x_1) \\ 1 & -\frac{d}{ml^2} \end{pmatrix} \begin{pmatrix} \lambda_1 \\ \lambda_2 \end{pmatrix} \Rightarrow \tag{3.47}$$

$$\begin{pmatrix} \dot{\lambda}_1 \\ \dot{\lambda}_2 \end{pmatrix} = \begin{pmatrix} -(x_1 - \pi) + \frac{g}{l}\cos(x_1)\lambda_2 \\ -\alpha x_2 - \lambda_1 + \frac{d}{ml^2}\lambda_2 \end{pmatrix} \tag{3.48}$$

that defines the dynamics of the controller. Remember that costate λ is in fact the state vector of the feedback controller that has the real state x as input. The output of the controller is u which was already found to be $-b\lambda_2/r$.

At time 0 the state $x(0) = x_0$ is given by $(0, 0)^T$. Hence, there is no freedom left in $x(0)$ so that $\delta x(0)$ is zero and condition VI is immediately satisfied. The final time t_f is fixed to value 1 so that $\delta t_f = 0$. At this final time there is no constraint V and thus $\delta x(1)$ is existing and not necessarily zero. Hence, the first factor of condition VII is not zero so that condition VII can only be guaranteed by putting the second factor equal to zero which yields:

$$\lambda(1) = [\frac{\partial \Phi}{\partial x}]_1 = \gamma \begin{pmatrix} x_1(1) - \pi \\ x_2(1) \end{pmatrix} \tag{3.49}$$

By the found structure of the controller i.e. its differential equations, by the initial values for state x and finally the final conditions for the costate λ in terms of the final values of the real states x, we thus have reduced the original problem to a TPBV-problem:

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{\lambda}_1 \\ \dot{\lambda}_2 \end{pmatrix} = \begin{pmatrix} x_2 \\ -\frac{g}{l}\sin(x_1) - \frac{d}{ml^2}x_2 - \frac{b^2}{r}\lambda_2 \\ -(x_1 - \pi) + \frac{g}{l}\cos(x_1)\lambda_2 \\ \alpha x_2 - \lambda_1 + \frac{d}{ml^2}\lambda_2 \end{pmatrix} \tag{3.50}$$

$$\begin{pmatrix} x_1(0) \\ x_2(0) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad \begin{pmatrix} \lambda_1(1) \\ \lambda_2(1) \end{pmatrix} = \begin{pmatrix} \gamma(x_1(1) - \pi) \\ \gamma x_2(1) \end{pmatrix} \tag{3.51}$$

We are not going to solve this TPBV-problem but it is easy to simulate it in Simulink and optimise $\lambda(0)$ until the boundary conditions are satisfied. In Fig. 3.4 a representation in Simulink is shown (example8.m). After some trial and error we obtained values $\lambda(0) = (-34, -6\pi)^T$ that rather accurately steered x from $x(0) = 0$ to $x(1) = (\pi, 0)^T$ when the coefficients of the differential equations are all 1 or .1 as on Fig. 3.4. The obtained trajectories are shown in Fig. 3.5. However, notice that, by doing so, we solved the problem to find a controller that fits the constraint $x(1) = (\pi, 0)^T$ rather than satisfying the derived contraint VII that relates $\lambda(1)$ to $x(1)$.

3.4 Coupling control.

The problem under study is to speed up the rotation of a motor shaft until it matches the rotation speed of another shaft so that at that moment both shafts can be coupled. Such a problem occurs in the clutch of a car but also in the switching of an electricity generator on the net. The dynamics are greatly simplified in order to focus on the control effects related to the chosen criteria. Let the dynamics be given by an input torque T that has only to overcome the inertial moment $J = 1(kgm^2)$, while the state is the rotational speed which is to be brought to $\omega_0 = 3(rad/s)$:

$$\begin{cases} u = T = J\dot{\omega} \\ J = 1 \\ \omega = x \end{cases} \qquad \Rightarrow \qquad \begin{cases} \dot{x} = u \\ x(0) = 0 \end{cases} \tag{3.52}$$

This example suits to illustrate and compare several commonly used criteria and constraints.

3.4.1 A) quadratic criterion

Let us first consider a simple straightforward quadratic criterion:

$$J = \frac{1}{2}\int_0^{t_f} [(x - 3)^2 + ru^2]dt \tag{3.53}$$

The coefficient $r > 0$ weights the control torque whose power is a direct cost and whose amplitude should be bounded to prevent saturation. The first term weights the deviation from the desired $x = 3 = \omega_0$ from the very moment $t_0 = 0$. This implies that we are not

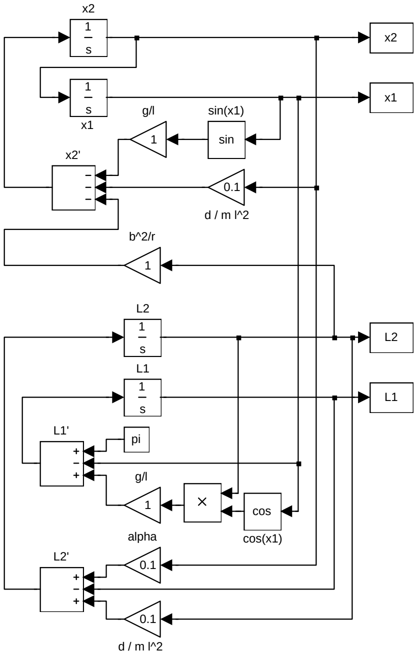


Figure 3.4: Controlled inverted pendulum in Simulink.

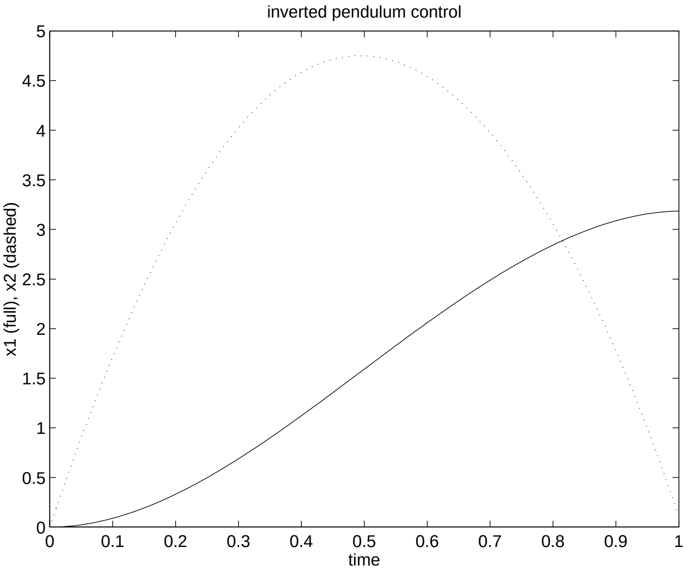


Figure 3.5: Controlled pendulum trajectories.

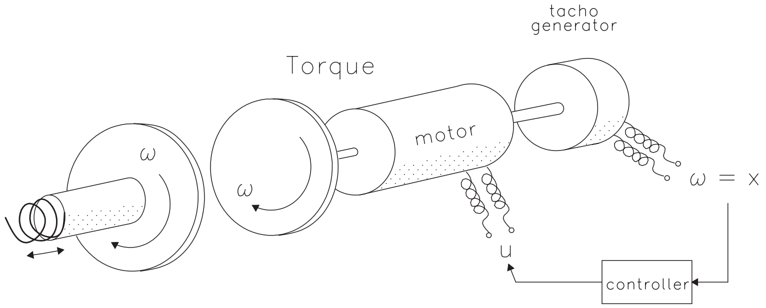


Figure 3.6: Coupling control situation.

satisfied with a control that at last at moment t_f causes $\omega = \omega_0 = 3$. We require from the initial moment on that $x(t) = \omega(t)$ speeds up as fast as possible to ω_0.

The coefficient 1/2 is optional and we could have multiplied the criterion by any other positive number. Of course the solution won't change by it because it just scales the J-axis and the independent variables (x and u) are unaffected by it. The chosen coefficient 1/2 compensates all 2's in forthcoming differentiations. The Hamiltonian and the Euler-Lagrange equations can easily be computed:

$$\begin{array}{lll} H = \frac{1}{2}(x-3)^2 + \frac{1}{2}ru^2 + \lambda u & \Rightarrow \quad \frac{\partial H}{\partial u} = ru - \lambda = 0 \quad \Rightarrow & u = -\frac{\lambda}{r} \\ \begin{cases} -\frac{\partial H}{\partial x} = \dot{\lambda} = -x + 3 \\ u = \dot{x} = -\frac{\lambda}{r} \end{cases} & \Rightarrow \quad \begin{pmatrix} \dot{x} \\ \dot{\lambda} \end{pmatrix} = \begin{pmatrix} 0 & -\frac{1}{r} \\ -1 & 0 \end{pmatrix} \begin{pmatrix} x \\ \lambda \end{pmatrix} + \begin{pmatrix} 0 \\ 3 \end{pmatrix} & \end{array} \tag{3.54}$$

The block diagram of the process and controller defined by these equations is represented in Fig. 3.7.

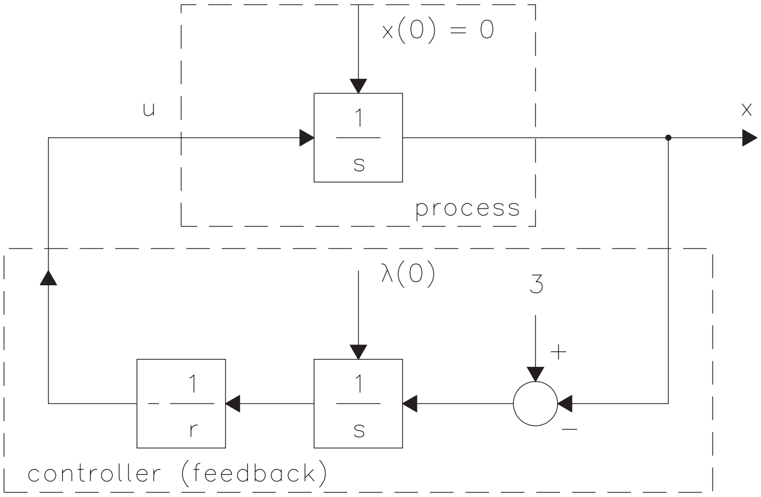


Figure 3.7: Closed loop block diagram.

The eigenvalues of the state matrix, describing the total system dynamics in above equation, determine the exponential modes and the corresponding eigenvectors indicate the relative components in the states. Consequently the trajectories can be expressed accordingly, together with the particular solution as:

$$\begin{pmatrix} x \\ \lambda \end{pmatrix} = \alpha \begin{pmatrix} -\sqrt{\frac{1}{r}} \\ 1 \end{pmatrix} e^{\sqrt{\frac{1}{r}}t} + \beta \begin{pmatrix} \sqrt{\frac{1}{r}} \\ 1 \end{pmatrix} e^{-\sqrt{\frac{1}{r}}t} + \begin{pmatrix} 3 \\ 0 \end{pmatrix} \tag{3.55}$$

Let us start with the simplest boundary conditions:

3.4.2 A1) $x(0) = 0$ and $x(t_f) = 3$ (exampla1.m)

These conditions (actually IV and V) provide two linear equations in α and β. The solution in α and β yields:

$$\begin{pmatrix} x \\ \lambda \end{pmatrix} = \frac{-3}{e^{\sqrt{\frac{1}{r}}t_f} - e^{-\sqrt{\frac{1}{r}}t_f}} \begin{pmatrix} e^{\sqrt{\frac{1}{r}}(t_f-t)} - e^{-\sqrt{\frac{1}{r}}(t_f-t)} \\ \sqrt{r}(e^{\sqrt{\frac{1}{r}}(t_f-t)} + e^{-\sqrt{\frac{1}{r}}(t_f-t)}) \end{pmatrix} + \begin{pmatrix} 3 \\ 0 \end{pmatrix} \tag{3.56}$$

From this we can compute $\lambda(0)$ and initialise the controller accordingly:

$$\lambda(0) = \frac{-3\sqrt{r}}{e^{\sqrt{\frac{1}{r}}t_f} - e^{-\sqrt{\frac{1}{r}}t_f}} \left(e^{\sqrt{\frac{1}{r}}t_f} + e^{-\sqrt{\frac{1}{r}}t_f} \right) = -3\sqrt{r}\coth(\sqrt{\frac{1}{r}}t_f) \tag{3.57}$$

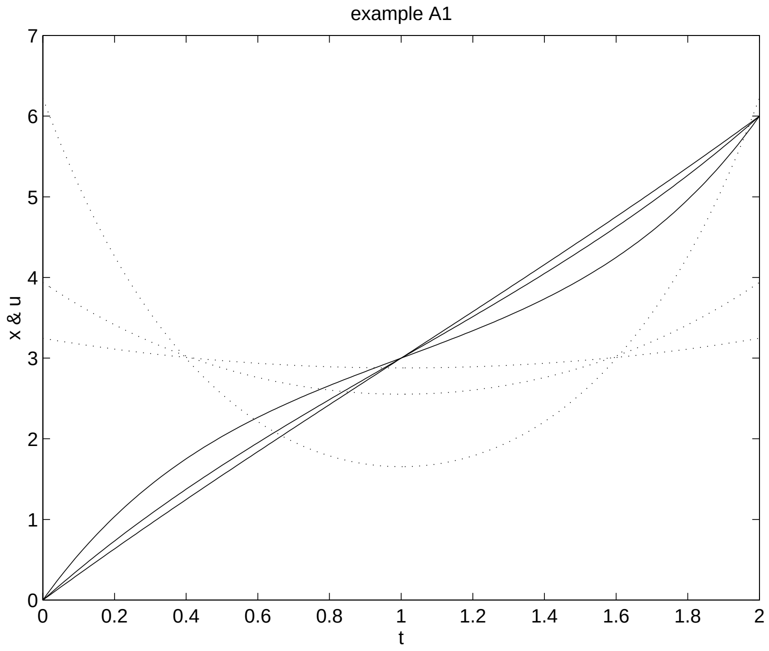


Figure 3.8: Various trajectories (–) and control signals (. . .) for $r = .25, 1, 4$.

Fig. 3.8 presents the trajectories and necessary control signals for various values of $r = .25, 1, 4$. If r increases the control action is more penalised and the system becomes "slower". Note that the closed-loop system indeed brings the state x from 0 to 3 in time t_f but it is *unstable*. So in order to keep the state at 3, another controller should take over at moment t_f.

One might also have noticed that even this simple first order system with straightforward boundary conditions causes a lot of complex computations. In the second half of this course we will show how these computations can be streamlined and simplified considerably for linear systems with quadratic constraints.

Another simplification occurs if we let t_f approach infinity:

3.4.3 A2) like A1 but $t_f = \infty$ (exampla2.m)

By letting t_f go to infinity we essentially force the closed loop system to be stable. In the limit we obtain from the above expression for $(x(t), \lambda(t))^T$:

$$\begin{pmatrix} x \\ \lambda \end{pmatrix} = \begin{pmatrix} 3 - 3e^{-\sqrt{\frac{1}{r}}t} \\ -3\sqrt{r}e^{-\sqrt{\frac{1}{r}}t} \end{pmatrix} \qquad \Rightarrow \qquad \lambda(0) = -3\sqrt{r} \tag{3.58}$$

Note that the unstable mode has disappeared but this is actually effected in the implementation by the proper initial value $\lambda(0)$. This value therefore is very critical! Any small deviation, which is inevitable numerically and in any practical implementation, will activate the unstable mode. So we need to implement the controller such that the unstable mode is *structurally* absent. This can be done by observing that we really have a first order system and that :

$$\forall t : \qquad \lambda = \sqrt{r}(x - 3) \qquad \Rightarrow \qquad u = -\frac{\lambda}{r} = \sqrt{\frac{1}{r}}(3 - x) \tag{3.59}$$

So a better implementation is provided by Fig. 3.9.

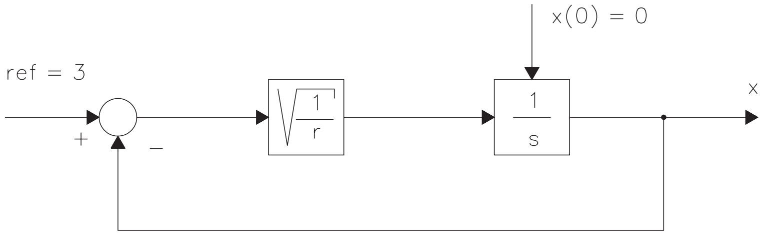


Figure 3.9: Stable implementation.

This is simply a constant state feedback. The closed loop system is robustly stable now and we do not need any initial value for a costate λ as this costate has essentially been eliminated. There is also no need to have another controller for stabilisation. But we have to pay for this comfortable solution by the fact that the final desired value 3 is only obtained for $t_f = \infty$ via an exponential trajectory as shown in Fig. 3.10.

Again it is obvious that higher penalties ($r = .25, 1, 4$) on u cause the corresponding controller to be "tame". Further discussions on this issue will be presented in the second half of this course.

For a further study of the effect of the various constraints we want to avoid the burden of extensive computations due to the quadratic criterion in x and focus on the pure effects of the various constraints. Therefore, we change the criterion in the following way:

3.4.4 B) $F \neq F(x)$: F is independent of x

For instance:

$$J = \frac{1}{2}\int_0^{t_f} [k + u^2]dt + \Phi \tag{3.60}$$

satisfies this curtailment. A variable weighting r for u^2 has been skipped because its effect has been illustrated before. We are not interested in the way that x moves from 0 to 3 but $k \geq 0$ represents the cost of the *time* which is necessary to reach the desired value for x. We also can not constrain x for $t > t_f$ so that we expect an unstable closed loop

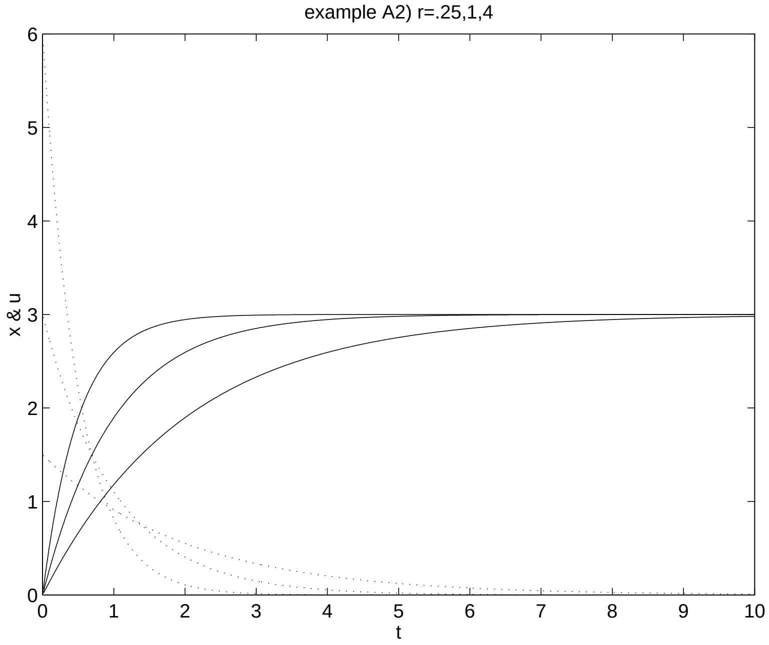


Figure 3.10: Optimal stable trajectories (—-) and control signals (. . .) for $r = .25, 1, 4$.

system again. In several sub-examples we will vary t_f, k and Φ but irrespective of these variations we can immediately derive the Hamiltonian, the Euler-Lagrange equations and thus the overall structure of all solutions:

$$H = \frac{k}{2} + \frac{u^2}{2} + \lambda u \qquad ; \qquad 0 = \frac{\partial H}{\partial u} = u + \lambda \qquad \Rightarrow \qquad u = -\lambda \quad (3.61)$$

$$\dot{\lambda} = -\frac{\partial H}{\partial x} = 0 \ \Rightarrow \ \lambda(t) = \lambda(0) = \lambda_0 \ ; \ \dot{x} = u = -\lambda_0 \ \Rightarrow \ x(t) = -\lambda_0 t + x(0) \quad (3.62)$$

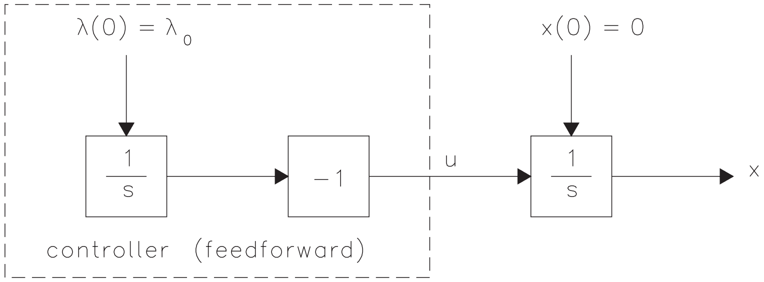


Figure 3.11: Structure problem B.

So, apparently, we deal with a feedforward control as illustrated in Fig. 3.11 and the different conditions will eventually only affect the constant λ_0. Let us first analyse this for the straightforward constraints:

3.4.5 B1) $\Phi = 0, t_f = 1, x(0) = 0, x(1) = 3$

Then conditions IV and V immediately imply:

$$3 = -\lambda_0 \quad ; \quad u = 3 \quad ; \quad x(t) = 3t \quad ; \quad J = (k + 9)/2 \qquad (3.63)$$

In fact the final value defines λ_0 which in turn determines u, irrespective of a possible weight r on u^2 in the criterion J. Suppose that the resulting u is too big and that the actuator is saturated so that we have to decrease the control effort u. We are then forced to relax our requirements. One possibility is to weaken the condition at $t_f = 1$ by letting $x(1)$ free but forcing it to be close to 3 by introducing a proper Φ:

3.4.6 B2) $\Phi(x(t_f), t_f) = \frac{1}{2}\alpha(x(t_f) - 3)^2, t_f = 1, x(0) = 0\ (\alpha > 0)$.

A constraint $\Phi(x(t_0), t_0)$ is irrelevant and has no effect because $x(0)$ is fixed. By manipulating $\alpha > 0$ we can tune the solution such that u is small enough and simultaneously $x(1)$ is closest to 3 as these two requirements are contradictory. Condition IV is trivial:

$$x(0) = 0 \quad \Rightarrow \quad x(t) = -\lambda_0 t \tag{3.64}$$

and VII yields:

$$t_f = 1: \ \frac{\partial \Phi}{\partial x} = \lambda \quad \Rightarrow \quad \alpha(x(1) - 3) = \lambda(1) = \lambda_0 \tag{3.65}$$

By substitution of $x(1) = -\lambda_0$ we obtain an equation in λ_0 with the solution:

$$-\lambda_0 = \frac{3\alpha}{1+\alpha} = u \quad \Rightarrow \quad x(t) = \frac{3\alpha t}{1+\alpha} \tag{3.66}$$

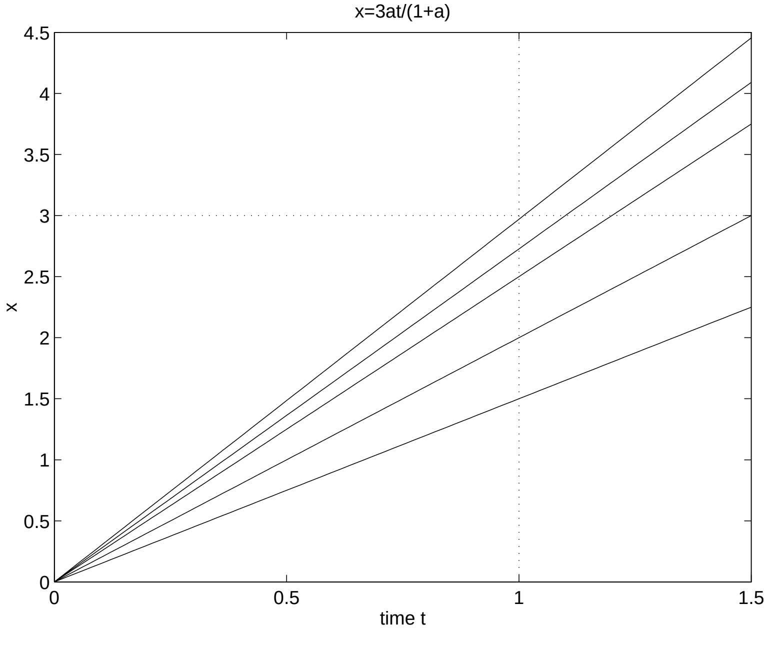


Figure 3.12: $x(t)$ for various α.

As illustrated in Fig. 3.12 the trajectories for different values of α are all less than $3t$ and x does not reach 3 at time t. But the effort in the control signal is also weaker and thus the total costs J are less as illustrated in Fig. 3.13.

Another way of decreasing the effort in u is to relax the time span to reach the final value 3:

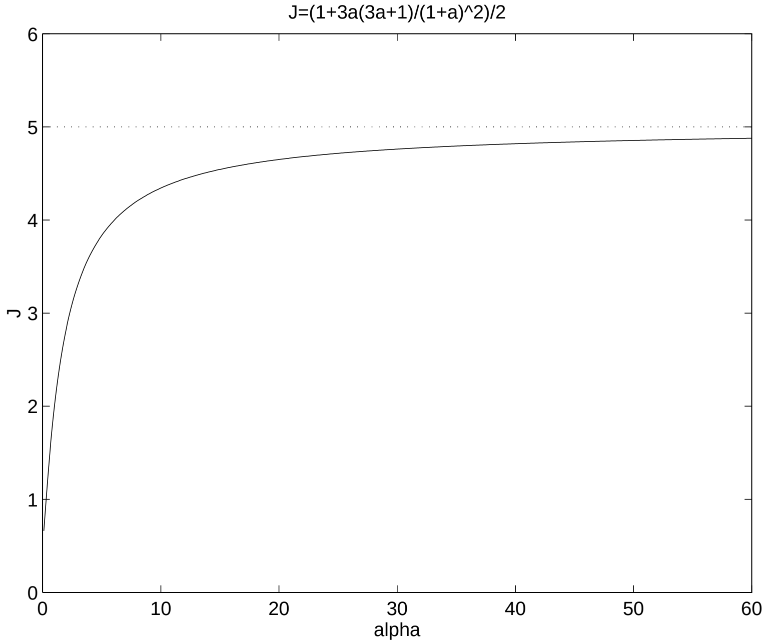


Figure 3.13: Total cost as a function of α.

3.4.7 B3) $\Phi = 0, x(0) = 0, x(t_f) = 3, t_f =$free.

So we have a final time constraint:

$$g(x(t_f), t_f) = x(t_f) - 3 = 0 \tag{3.67}$$

Condition IV is as before but now we have to deal with conditions VII and VIII:

$$\begin{cases} VII: \quad \delta x(t_f) = 0, \quad \dot{x}\delta t_f \neq 0 \quad \Rightarrow \quad -\lambda + \eta = 0 \quad \Rightarrow \quad \eta = \lambda_0 \\ VIII: \quad H(t_f) = 0 = \frac{1}{2}k + \frac{1}{2}\lambda_0^2 - \lambda_0^2 \quad \Rightarrow \quad \lambda_0 = \pm\sqrt{k} \end{cases} \tag{3.68}$$

Only the negative λ_0 in the last expression leads to a solution:

$$x(t) = \sqrt{k}t \qquad u = \sqrt{k} \qquad x(t_f) = 3 = \sqrt{k}t_f \quad \Rightarrow \quad t_f = 3/\sqrt{k} \tag{3.69}$$

Evidently the time costs determine t_f. If k increases the final time decreases at the cost of higher u. If $k > 9$ then even $t_f < 1$ but also $u > 3$. Notice that the time costs have had no influence so far. Till now we tried to overcome possible saturation of the actuator by increasing weights on u. For the simple example under study we could solve the problem explicitly and see which weights accomplish this. In more complicated problems prevention from saturation amounts to iteratively solving the problem until the amplitude of u is small enough. By doing so we also decrease the value of u in time periods where there is no saturation at all. So we cannot expect that this is an optimal way of avoiding saturation. A better solution is offered in the next section and illustrated by the same example as used till now.

3.5 Properties of Hamiltonian; Pontryagin.

The "Hamiltonian" was not discussed and only introduced by name. As H in fact denotes an energy function, the following remarks can be useful when solving for the various constraints:

1. In case the Hamiltonian function is continuous and differentiable in its arguments $(x,\, u,\, \lambda,\, t)$ we can write for any triple of differentiable trajectories $x,\, u,\, \lambda$:

   $$\frac{dH}{dt} = \frac{\partial H}{\partial t} + \frac{\partial H}{\partial x^T}\dot{x} + \frac{\partial H}{\partial u^T}\dot{u} + \frac{\partial H}{\partial \lambda^T}\dot{\lambda} \tag{3.70}$$

   By substitution of the trajectories defined by the Euler-Lagrange and process equations:

   $$\frac{\partial H}{\partial u} = 0 \quad ; \quad -\frac{\partial H}{\partial x} = \dot{\lambda} \quad ; \quad \frac{\partial H}{\partial \lambda} = f = \dot{x} \tag{3.71}$$

   so that we get for the optimal trajectories:

   $$\frac{dH}{dt} = \frac{\partial H}{\partial t} \tag{3.72}$$

   Hence, the time derivative of the Hamiltonian is given by the partial derivative which implies that, if there is no explicit time dependence on t, the Hamiltonian is *constant* in time or time-invariant. This possible explicit time dependence can only be introduced by functions F or f explicit in t. Consequently along the optimal solution the H can therefore be seen as a "conservation law" or a preserved quantity. In all presented examples H was constant!

2. Sometimes there are inevitable and very strict bounds on the control input u. We mentioned the saturation effect of e.g. pumps and servomotors. So in the best case of bidirectional steering : $u_{\min} \le u \le u_{\max}$. Another typical example is a valve that can not be more open than 100% and not more close than 0%. So evidently $0 \le u \le u\text{max}$. Drugs can only be administered so that $u \ge 0$. Pontryagin studied this problem and presented the solution in his famous "maximum principle". Introduction and proof of this method requires a lot of formal definitions and mathematical juggling. We will not go into those details and just present the result in the context we described the material so far. The Euler-Lagrange equation $\frac{\partial H}{\partial u} = 0$ actually represents the condition on u for which H is *minimised.* If u is bounded to belong to an admissible set U we still have to minimise H but simply putting the derivative with respect to u to zero would be in conflict with the constraints on u. We have to find the minimum of H at each time moment for the admissible set U on u. For this we only have to consider the *explicit* dependence of H on u comparably to the consideration of only the *partial* derivative in case of unbounded u. This can best be illustrated with the last example where the constraints are now given by:

3.5.1 B4) $\Phi = 0$, $x(0) = 0$, t_f =free, $x(t_f) = 3$, $U = \{u| - 1 \le u \le 1\}$

As before we have:

$$g(x(t_f), t_f) = x(t_f) - 3 = 0 \quad ; \quad \dot{\lambda} = 0 \quad \Rightarrow \quad \lambda(t) = \lambda_0 \tag{3.73}$$

The control u is now to minimise:

$$H = \frac{1}{2}k + \frac{1}{2}u^2 + \lambda u = \frac{1}{2}[k + (u + \lambda_0)^2 - \lambda_0^2] \tag{3.74}$$

Obviously the global minimum is obtained for $u = -\lambda_0$ but the question is whether this u is admissible. Fig. 3.14 illustrates this for $k = 1$, but where we do not know λ_0 beforehand. So H is only shown for two fancy values $\lambda_0 = -1$ and $\lambda_0 = -3$.

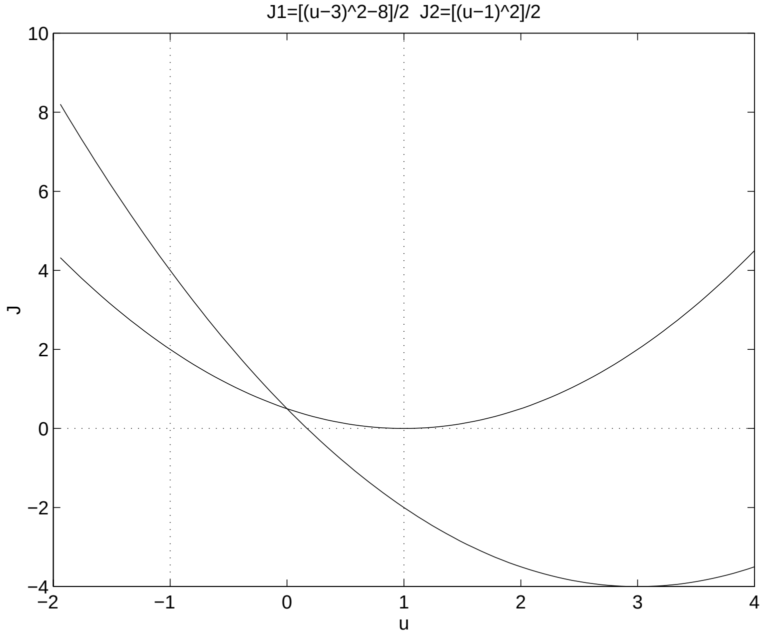


Figure 3.14: Hamiltonian and the set $U = \{u| - 1 \leq u \leq 1\}$.

Two possibilities are then evident:

a) Suppose $u = -\lambda_0 \epsilon U$ then the solution is as under B3) and we obtain $u = -\lambda_0 = \sqrt{k}$ and this is all right as long as $k \leq 1$.

b) If $k \geq 1$ then apparently u takes values at the bound so:

b1) $u = -1 \Rightarrow x(t) = -t$ and x will thus never reach the desired $x(t_f) = 3$. also from Fig. 3.14 we can see that this is not a feasible solution as it is not a boundary minimum.

b2) $u = 1 \Rightarrow x(t) = t \Rightarrow x(3) = 3 \Rightarrow t_f = 3$ which corresponds to a boundary minimum for $H = 1/2k + 1/2 + \lambda_0$ where λ_0 can be computed from conditions VII and VIII:

$$\begin{cases} VII: \quad \lambda_0 = \eta \\ VIII: H(t_f) = 0 = k/2 + 1/2 + \lambda_0 \quad \Rightarrow \quad \lambda_0 = -(1+k)/2 \end{cases} \tag{3.75}$$

So indeed for $k = 1$ both solutions a) and b2) coincide but λ_0 is actually unimportant here. Effectively u is kept on the bound during the full time span which was not the case in the previous adaptations B2) and B3). It is easy to see that the solution for u was either $+1$ or -1 for the whole time span because λ was already found to be constant. This is not necessarily true for more complicated examples so that during the interval of interest the u may switch or even take partly a continuous trajectory. As an example may serve the bang-bang control of section 1.4.7 that can be solved by Pontryagin's principle. This is left to the reader as an exercise. It is also remarkable that in both cases a) and b), the resultant optimal Hamiltonian is zero after computation of λ_0. From Fig. 3.14 the suggestion would be the adverse but here the λ_0 is taken fixed while actually this same λ_0 is depending on u and *after* the choice for u this takes such a value that H becomes zero. This illustrates that the minimisation of H with respect to u is in fact explicit: the implicit dependence of λ_0 on u needs not be taken into account.

3.6 Discrete time version.

The optimal control in discrete time systems remains to be discussed. Instead of a continuous time t we then have to deal with discrete time instances $t_k = kT$. We will study optimal control problems for k ranging from 0 to N. In order not to complicate a basically analogous derivation too much we will skip the possibility of a free final time t_f. We generally assume that $x(0) = x_0$ is fixed so that $\Phi(x(0), 0)$ is superfluous. We confine to the PROBLEM:

$$x(k+1) = f(x(k), u(k), k) \tag{3.76}$$

$$J = \Sigma_{k=0}^{N-1} F(x(k), u(k), k) + \Phi(x(N)) \tag{3.77}$$

$$H(k) = F(x(k), u(k), k) + \lambda(k+1)^T f(x(k), u(k), k) \tag{3.78}$$

Note that the summation is till $N-1$ while the constraint Φ acts on x evaluated at instant N. Furthermore λ is indexed by $k+1$ which appears to be a fortunate choice afterwards for obtaining symmetric solutions in x and λ. Other choices would simply cause time shifts in the solution of λ. The exact derivation will also not be given explicitly as it is similar to the continuous time situation. The SOLUTION is outlined by the following formulae:

I	constraint	$x(k+1) = f(x(k), u(k), k)$	$processdynamics$
II	$\delta x(k) \quad \Rightarrow$	$\lambda(k) = \frac{\partial H(k)}{\partial x(k)}$	$Euler - Lagrange$
III	$\delta u(k) \quad \Rightarrow$	$\frac{\partial H(k)}{\partial u(k)} = 0$	$Euler - Lagrange$
$IIIa$	if $\quad u \epsilon U$	$\min_{u \epsilon U} H(k)$	$Pontryagin$
IV	constraint :	$x(0) = x_0$	$initial \quad time \quad condition$
V	constraint :	$x(N) = x_N$	$final \quad time \quad condition$
VI	at $\quad k = 0 :$	$\delta x(0)^T \lambda(0) = 0$	$transversality \quad condition$
VII	at $\quad k = N :$	$\delta x(N)^T [\frac{\partial \Phi}{\partial x(N)} - \lambda(N)] = 0$	$transversality \quad condition$

Note that there is no minus sign in the Euler-Lagrange II implication. Again, explanation is best obtained by means of an example :

3.6.1 Example8: car acceleration (bike suits as well).

After stopping for a traffic light we want to leave the place as fast as possible but higher acceleration costs more petrol and thus more money and an increase of pollution. (For a cyclist the cost is immediate in a shortage of oxygen.) We propose a very simple model in the form of Newton's law: $F = mdv/dt$. F is the force of propulsion controlled by gas pedal combined with brakes, m is the total mass of car and driver while v is the forward speed. We neglected the friction because of excellent lubrication and optimal aerodynamic streamlining. In a discretised version we get :

$$x(k+1) = x(k) + u(k) \quad ; \quad x(0) = 0 \tag{3.79}$$

if appropriate scaling of mass and force is applied. The state x represents the speed v and the input u is linearly related to the force F. The motivation to speed up during ten sample periods within the possibilities of car and purse is translated into the following criterion and constraint:

$$J = \frac{1}{2}\Sigma_0^9 u(k)^2 - \beta x(10) \quad ; \quad \beta > 0 \quad ; \quad u \leq 1 \tag{3.80}$$

From the Hamiltonian we derive:

$$H(k) = \frac{1}{2}u(k)^2 + \lambda(k+1)[x(k) + u(k)] \tag{3.81}$$

$$\lambda(k) = \frac{\partial H(k)}{\partial x(k)} = \lambda(k+1) \quad \Rightarrow \quad \lambda(k) = \lambda_0 \tag{3.82}$$

Because there are bounds on u it might be that we have to apply Pontryagin's maximum principle. In order to find out we consider two possibilities:

a) Suppose the optimal solution is not contradicting the bounds on u so that we find the minimum of H by differentiation:

$$\begin{array}{rl} \frac{\partial H(k)}{\partial u(k)} = 0 = \lambda(k+1) + u(k) & \Rightarrow \quad u(k) = -\lambda(k+1) = -\lambda_0 \\ VII: \quad -\beta - \lambda_0 = 0 & \Rightarrow \quad \lambda_0 = -\beta \end{array} \tag{3.83}$$

which implies that $u(k) = \beta$ provided $\beta \leq 1$ so that we get:

$$x(k) = x(0) + \Sigma_{i=0}^{k-1}\beta = k\beta \quad \Rightarrow \quad x(10) = 10\beta \leq 10 \tag{3.84}$$

b) If $\beta > 1$ then the costs do not outweigh the desire for speed and the driver goes for maximum speed. This maximum gas implies $u = 1$ by limitations of the car as can be found from the Hamiltonian: The solution $u(k) = 1$ leads to a boundary minimum leading to the solution:

$$x(k) = k \quad \Rightarrow \quad x(10) = 10 \tag{3.85}$$

which is the maximum possible for the car under study. Of course λ_0 can be computed as well but is irrelevant for the solution. The same remarks can be made on λ_0 as in the continuous time example sub B).

Chapter 4

Introduction linear systems

This second part of the course ‘modern control’ is distinct from the first part, as it only considers **linear** systems. Nevertheless, it shares the crucial role of the state space representation. The **states** completely define the status of the system, as they represent its memory. Each component of the system, that can independently **collect and return** information in the form of energy, is described by a state like an electric capacitor and coil, a mechanic spring and mass (potential and kinetic energy), thermal and fluid capacitances, etc. Components, that can only **produce** within the context of the system description, are sources like batteries, motors, pumps and often take the form of inputs. Components, that can only **dissipate**, contribute to the system description, but do not lead to states, like electrical, thermal and fluid resistances and mechanical dampers. So the states represent components of the system, that can exchange information in the form of energy with the other components in both directions and, as such, they act as the memory locations of the system. A minimum realisation (of a state space representation) of the system is based on this minimum number of memory locations and, in an analogue simulation, these memory locations are represented by the integrators. If we can control all states, we can control the total behaviour of the system. In the first part of this course we have seen the crucial role of the states, but also experienced, that a full description of the states and its control soon degenerates into a very complicated and cumbersome task. This is mainly caused by the **nonlinear** characteristics. Due to the nonlinearity, general algorithms can hardly be developed, as each nonlinearity displays a very specific behaviour dependent on the values of the states. However, linear systems show the same characteristics for smaller or greater values of the states, as the effect is linearly related to the state. Consequently, the superposition theorem is applicable and this facilitated the growth of a general theory, to be treated in this second part of the course. Certainly, the restriction to linear systems is quite a drastic one and only holds in approximation for limited ranges of the variables under study, but the resultant insight and applicable algorithms are overwhelming.

We will start in chapter 5.1 with a short, historical development, that has led to the insight, that states are the proper variables to be fed back instead of the available outputs. Next, in chapter 5.4 we will show, that indeed state feedback is ruled by straightforward algorithms, if we deal with quadratic criteria. The control inputs u then simply are linear combinations of the states x, realised by :

$$u = -Lx \tag{4.1}$$

and illustrated in Fig. 4.1 . Once we have obtained this proper feedback, we are faced by the problem, that only the outputs y of the process are available for processing and not the states x. Nevertheless, we can copy the process in a simulation model, represented

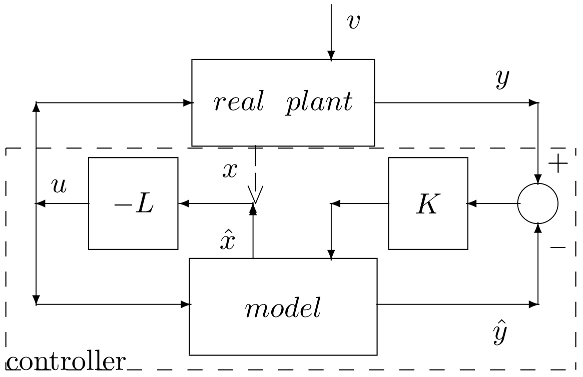


Figure 4.1: Total state control.

in a computer and feed it with the same numerical inputs u as the real process. This provides us with an estimated state $\hat{x}$, that incorporates only the effect of the inputs u though. We also have to deal with the effects of state disturbances v and unknown initial values of x. Only the outputs y bear information about this and it turns out, that by comparing the outputs y with the estimated outputs $\hat{y}$ and multiplying the difference by a matrix K, we can generate a kind of estimate of the state disturbance and the initial values and thus feed these into the model. This complete setup for obtaining $\hat{x}$ is called an **observer** and yields the final estimate $\hat{x}$, that can be fed back instead of the real, unavailable x. Such general (Luenberger) observers and in particular the Kalman-Bucy filters are treated in chapter 6. The symmetry in Fig. 4.1 reflects the dualism, that exists in the computation of L and K. They both follow from so-called Riccati equations. In chapter 7 the consequence of feeding back $\hat{x}$ instead of x will be treated as the "separation principle". Furthermore, it will be shown, how these resultant LQG-controllers can be improved in their tracking behaviour.

Until here, all theory has been developed for continuous time systems, but, because most controllers are nowadays implemented in digital systems, the complete theory will be reconsidered for discrete time systems in chapter 8. This chapter will end with the introduction of a model, the so-called innovations representation, that follows in a natural way from the observer theory and appears to be very useful in practical implementation.

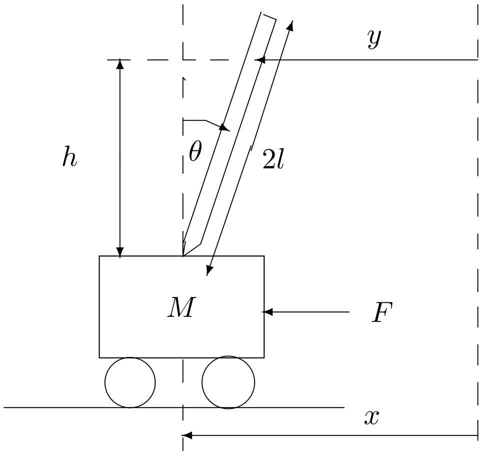
Chapter 5

The continuous, optimal control problem

5.1 Compensation in right half plane

In this section it will be shown, that state space analysis can reveal much more than straight transfer-functions. Certain states can become unobservable or uncontrollable, which makes them invisible in the transfer-function. Nevertheless, these states may have unpleasant effects and in particular unstable states can then ruin a seemingly sound transfer.

This can well be explained by means of an example : the familiar inverted pendulum, driven here by a horizontal force as depicted in Fig. 5.1. The carriage of mass M is driven horizontally by a force F, that acts as primary input. The position of the carriage is denoted by x. On the carriage an inverted pendulum of mass m, that is homogeneously distributed along its length $2l$, is to be balanced, i.e. its angle θ with the vertical is to be kept close to zero. At height h the position of the pendulum is measured optically yielding y, that functions as the output of the process. By means of Lagrange's method, based upon all potential and kinetic energies or by direct mechanical equations, the describing,



h

Figure 5.1: The inverted pendulum on a carriage

nonlinear equations can be derived. Linearisation about $(\theta, x, y, F) = 0$ leads to:

$$\begin{aligned} (M+m)\ddot{x} - ml\ddot{\theta} &= F \\ (ml^2 + \tfrac{1}{3}ml^2)\ddot{\theta} - ml\ddot{x} - mgl\theta &= 0 \end{aligned} \tag{5.1}$$

We can apply a subloop controller such, that a reference for x is being given and that the force F is controlled causing x to follow that reference accurately. E.g. in an x-y plotter, often used to implement the carriage/pendulum-plant, such a fast tracking loop has been installed. As a consequence, we just have to consider the second equation, where x functions as the input, so that the transfer from x to θ is given by:

$$\theta = \frac{3}{4l}\frac{s^2}{s^2 - \frac{3g}{4l}}x \tag{5.2}$$

If we detect the position y of the pendulum at height h by means of an optical sensor this output relates to the input x in a linearised form as:

$$y = x - h\tan(\theta) \approx x - h\theta \tag{5.3}$$

so that we finally get:

$$\frac{y}{x} = (1 - \frac{3h}{4l})\frac{s^2 - \frac{3g}{4l-3h}}{s^2 - \frac{3g}{4l}} \tag{5.4}$$

Note, that there are two zeros, whose positions depend on the height h of the sensor. For $h = 0$, the zeros coincide with the poles and indeed we then have $y = x$. For increasing h, the zeros move away from the origin along the real axis till infinity for $h = \frac{2}{3}2l$, for which value the gain is zero. This is precisely the point, around which the pendulum will rotate under the force F. (In reciprocal form, this is the reason, why you hit the billiard ball above the center, if you want it to roll without slip.) For still greater h, there is a phase shift of 180^0 (increasing x will result in a decreasing y), reflected in a zero-pair now lying on the imaginary axis from infinity. For maximum $h = 2l$ the zero-positions will end in $\pm j\sqrt{\frac{3g}{2l}}$, i.e. $\sqrt{2}$ times the distance of the poles to the origin.

For the demonstration of pole-zero compensation effects we will take $l = \frac{3}{4}g \approx 7.5m$, yielding poles at ± 1. Furthermore, $h = l$ will put the zeros at ± 2. In order not to complicate the analysis, we will ignore the left half plane zeros and poles, because these can be compensated without much of a problem, as we will show. Consequently, we are just left with a plant transfer:

$$P = \frac{s-2}{s-1} \tag{5.5}$$

that we would like to compensate by a compensator:

$$C = \frac{s-1}{s-2} \tag{5.6}$$

in a simple feedforward control, illustrated in the next block scheme.

According to conventional notation, the input x has been redefined as u, which is converted into a new input u^* by the prepositioned compensator C. Indeed, the mathematical transfer between u^* and y is now equal to 1, so that it seems, that y follows the input u^* exactly and promptly. Implementation would reveal quite a different outcome, as can be analysed by state space description. The following block-scheme, directly prompting states, can be drawn by considering, that $P = 1 - \frac{1}{s-1}$ and $C = 1 + \frac{1}{s-2}$.

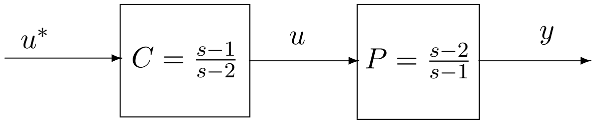


Figure 5.2: Blockscheme of controller and plant.

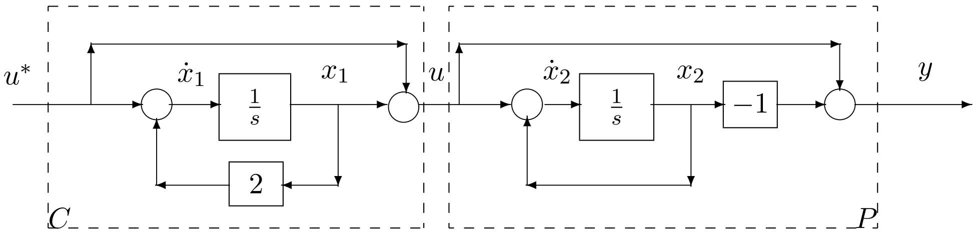


Figure 5.3: "Sequential" state space representation.

The corresponding state space equations are given by:

$$\begin{pmatrix} \dot{x_1} \\ \dot{x_2} \end{pmatrix} = \begin{pmatrix} 2 & 0 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} 1 \\ 1 \end{pmatrix} u* \tag{5.7}$$

$$y = \begin{pmatrix} 1 & -1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} 1 \end{pmatrix} u^* \tag{5.8}$$

If there is zero initial state and no state disturbance, indeed the unstable mode in x_2 viz. e^t is not excited by the input signal u^*, as the compensator, having a zero at 1, will filter out all signals, that can put energy in that special mode. This mode is not **reachable** from u^*. Nevertheless, each small disturbance or nonzero initial state will certainly cause the pendulum to fall down. The particular, unstable mode has become uncontrollable and thus not visible in the transfer function. This can be observed from the rank of the so-called controllability matrix as follows. If a general, state space description of dimension n is given by:

$$\dot{x} = Ax + Bu \tag{5.9}$$

$$y = Cx + Du \tag{5.10}$$

then the number of controllable (reachable) states is given by the rank of the *controllability matrix*:

$$\begin{pmatrix} B & AB & A^2B & \dots & A^{n-1}B \end{pmatrix} \tag{5.11}$$

(Exactly the same holds for discrete time systems!) Ergo for the system under study this yields:

$$rank\begin{pmatrix} B & AB \end{pmatrix} = rank\begin{pmatrix} 1 & 2 \\ 1 & 2 \end{pmatrix} = 1 \tag{5.12}$$

so that we conclude, that one state is uncontrollable.

On the other hand, the unstable mode in x_1 viz. e^{2t} will certainly be excited by the input u^*, nonzero initial state or state noise, but this particular mode will be filtered out by the process zero at 2, so that this mode will not be **detectable** in the output. This particular, unstable mode is not observable at the output y, as can be learned from the so-called observability matrix as follows. The number of observable (detectable) modes is equal to the rank of the *observability matrix*:

$$\begin{pmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{pmatrix} \tag{5.13}$$

(idem for time discrete systems) Ergo for the system under study this yields:

$$rank\begin{pmatrix} C \\ CA \end{pmatrix} = rank\begin{pmatrix} 1 & -1 \\ 1 & -1 \end{pmatrix} = 1 \tag{5.14}$$

so that we conclude that one state is unobservable.

Above analysis is even more evident, if we transform the "sequential" states as illustrated in Fig. 5.3 to "parallel" states as shown in next Fig. 5.4 by means of a Jordan (canonical) transform. This is accomplished by the eigenvalue decomposition of the system matrix A:

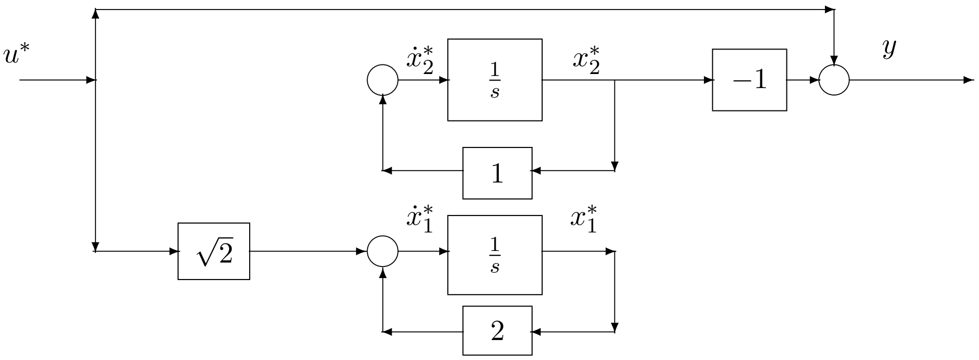


Figure 5.4: "Parallel" (Jordan canonical) states.

$$A = E\Lambda E^{-1} \tag{5.15}$$

where the diagonal matrix Λ contains the eigenvalues and E the corresponding eigenvectors as its columns. For the system under study:

$$\begin{pmatrix} 2 & 0 \\ 1 & 1 \end{pmatrix} = \begin{pmatrix} \frac{1}{2}\sqrt{2} & 0 \\ \frac{1}{2}\sqrt{2} & 1 \end{pmatrix} \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \sqrt{2} & 0 \\ -1 & 1 \end{pmatrix} \tag{5.16}$$

Generally, substitution of the decomposed matrix A into the original equations and premultiplying the first equation by E^{-1} results in the Jordan form in the new states $\underline{x}^* = E^{-1}\underline{x}$:

$$(E^{-1}\dot{\underline{x}}) = (E^{-1}E)\Lambda(E^{-1}\underline{x}) + (E^{-1}B)\underline{u}^* \quad \Rightarrow \quad \dot{\underline{x}}^* = \Lambda\underline{x}^* + B^*\underline{u}^* \tag{5.17}$$

$$\underline{y} = (CE)E^{-1}\underline{x} + D\underline{u}^* \quad \Rightarrow \quad \underline{y} = C^*\underline{x}^* + D\underline{u}^* \tag{5.18}$$

For the system under study the Jordan form then is:

$$\begin{pmatrix} \dot{x}_1^* \\ \dot{x}_2^* \end{pmatrix} = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x_1^* \\ x_2^* \end{pmatrix} + \begin{pmatrix} \sqrt{2} \\ 0 \end{pmatrix} u* \tag{5.19}$$

$$y = \begin{pmatrix} 0 & -1 \end{pmatrix} \begin{pmatrix} x_1^* \\ x_2^* \end{pmatrix} + \begin{pmatrix} 1 \end{pmatrix} u^* \tag{5.20}$$

Because of the diagonal structure of the system matrix Λ and the zero entries in B^* and C^*, we immediately observe, that x_2^* is not controllable and x_1^* is not observable. This is very evident when observing the block-scheme of this Jordan form in Fig. 5.4.

Indeed, x_1^* is excited by u^*, leading to $\frac{\sqrt{2}u^*}{s-2}$, which causes the unstable mode e^{2t}, that is not observed in y though. Nevertheless, the system will internally saturate or burn out. Also, the unstable state x_2^* is theoretically not excited by input u^*, but any nonzero, initial state $x_2^*(0)$ (or some noise) will lead to $\frac{x_2^*(0)}{s-1}$, so an unstable mode $x_2^*(0)e^t$.

Consequently, the main conclusion to be drawn from this example is, that the internal behaviour of a realisation (state space) may be more complicated than is indicated by the mathematical, external behaviour (transfer function). The internal behaviour is determined by the natural frequencies of the (nondriven) realisation, which in our case are $s = 1, 2$. However, because of cancellation, not all the corresponding modes of oscillation will appear in the overall transfer function. Or, to put it another way, since the transfer function is defined under zero initial conditions, it will not display all the modes of the actual realisation of the system. For a complete analysis, we shall need to have good ways of keeping track of all the modes, those explicitly displayed in the transfer function and also the "hidden" ones. It is possible to do this by careful bookkeeping with the transfer function calculations, but actually it was the state equation analysis as in the above example, that first clarified these and related questions. It directly shows, that pole zero cancellation in the right half plane, thus of unstable poles, is strictly forbidden, as it only makes these unstable modes or states either uncontrollable or unobservable, but they will still damage the nice theoretical transfer, as these signals grow uncontrollably without bounds.

If we apply pole-zero cancellation in the left half plane, the same effects will occur, but the corresponding signals will die out after some time, depending on the pole position i.e. its damping. For the inverted pendulum example we could thus apply pole-zero cancellation for the poles at -1 and -2, just causing uncontrollable and unobservable signals e^{-t} and e^{-2t}, that do not affect stability. Nevertheless, these effects can still be troublesome. As an example, remember the laboratory experiment with the water vessels (as at least most students, having followed the basic courses control at electrical engineering, can). This plant showed only three real, stable poles and no finite zeros. A PID-controller had to be designed, that was asked to annihilate the two slowest poles, as indicated in the next block scheme:

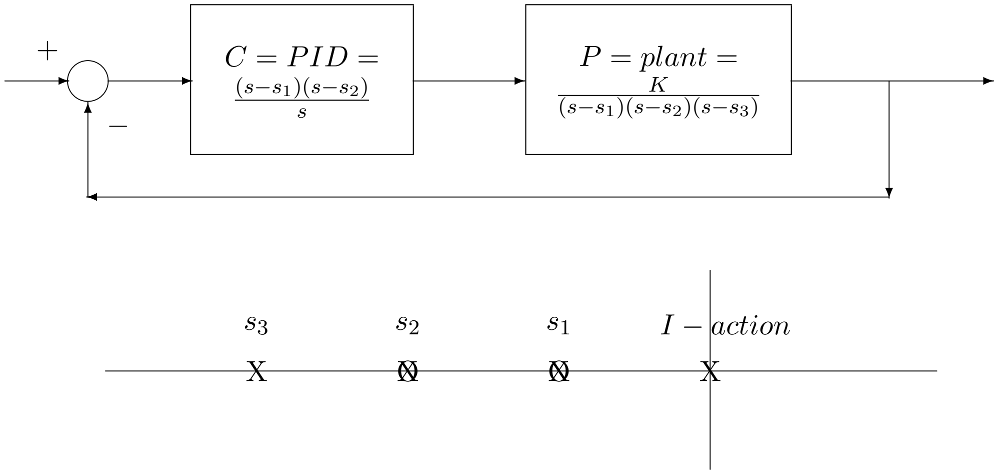

+
−
$C = PID = \frac{(s-s_1)(s-s_2)}{s}$
$P = plant = \frac{K}{(s-s_1)(s-s_2)(s-s_3)}$
s_3
s_2
s_1
$I-action$

Figure 5.5: Laboratory experiment with water vessels.

The loop transfer is then given by $\frac{K}{s(s-s_3)}$ and, by choosing the gain K properly, we can obtain a sufficiently damped and 'fast' pole pair for the closed loop system. Nevertheless, after implementation of the controller, one had to wait quite some time before the system was in steady state. This is precisely due to the compensated poles, which still give rise to slow transients $e^{s_1 t}$ and $e^{s_2 t}$, not apparent in the transfer function, but still reacting on nonzero initial states! The corresponding canonical states have become uncontrollable.

From now on we will assume, that all systems, to be observed and/or controlled, are both (completely) observable and (completely) controllable and that the dimension n of the state vector is minimal, i.e. a minimal realisation.

Now we have shown, that compensation is not the way to get rid of right half plane zeros and poles, the question arises how then to cope with them. In the next section it will be analysed how feedback can solve the problem. That is to say, the poles can be shifted to the left half plane by feedback, but the zeros will stay. Zeros in the right half plane, i.e. nonminimum phase zeros, cause ineffective response: initially the response is in the wrong direction (opposite sign). Furthermore, the rootloci end in zeros, so that these rootloci are drawn towards the unstable, right half plane. It further appears that nonminimum phase zeros put fundamental bounds on the obtainable bandwidth of closed loop systems. Consequently, what means are there to eliminate the nonminimum phase zeros? The answer has to be sought in the system itself of plant and sensors and actuators. For the inverted pendulum example the zeros were actually caused by the sensor. If we replace it by a direct angle sensor, so that $y = \theta$, the transfer function is simply given by equation 5.2. Unfortunately we then are confronted by two zeros in the origin. These were introduced by the input x, that appears in a second derivative in equation 5.1b., certainly because we deal with equations in forces and torques. So we have to turn back to forces or torques as primary inputs. We can accomplish this in two ways. First we can disassemble the tracking subloop for x (in e.g. the x-y plotter), thereby reinforcing equation 5.1a . If we now take mass M of the carriage much larger than the mass of the pendulum, the forces of the pendulum are negligible and we simply get $\ddot{x} = F/M$. Substitution in equation 5.1b simply yields equation 5.2 without the zeros and with some changed gain:

$$y = \theta = \frac{\frac{3}{4Ml}}{s^2 - \frac{3g}{4l}} F \tag{5.21}$$

Alternatively we can abolish the carriage altogether (or put $M = 0$) and apply a torque to the pendulum directly. A configuration could be to weld the foot of the pendulum perpendicularly onto the horizontal shaft of a DC-motor. By this actuator the term $ml\ddot{x}$ in equation 5.1b is simply replaced by the applied torque T again resulting in :

$$y = \theta = \frac{\frac{3}{4ml^2}}{s^2 - \frac{3g}{4l}} T \tag{5.22}$$

The lesson from this exercise is, that by proper choice and positioning of actuators and sensors, the occurrence and position of zeros can be corrected up to a high degree. If this is not sufficient, the plant itself should be changed. Transfer blocks *parallel* to the original transfer will completely change the zeros, but this requires a fundamental redesign of the plant itself. So it advocates the principle of negotiating about the building of plant in the very early stage of design in order to guarantee, that an efficient and optimal control is possible later on!

5.2 Analysis of stabilisation by output feedback.

We shall begin with an analysis of some feedback compensation schemes for modifying the transfer function of a given system and shall try to give the reader an idea of the situation in the late 1950's, that set the stage for state space methods.

In the previous section we discussed the stabilisation of an unstable system with unstable pole $s = 1$ by the use of a series compensator without feedback. However, we saw, that this was not satisfactory, because the cancellation in the transfer did not mean, that the unstable, natural frequency had disappeared from the overall realisation. This emphasised the distinction between internal and external descriptions. Let us depart from equations 5.21 or 5.22, compensate the stable pole and get a gain of 1 by proper choice of variables, so that we simply have:

$$y = \frac{1}{s-1}\,u \tag{5.23}$$

When we look more closely at the problem of pendulum stabilisation, it is clear that we need some feedback. For we may see, that a small nonzero value of x_2 (or θ) will give rise to an exponentially growing or exponentially decreasing value of x_2 (or θ) according to whether $x_2(0)$ (or $\theta(0)$) is positive or negative. One should know in what direction the pendulum is falling in order to be able to correct. However, without feedback, there is no way of knowing at the input whether x_2 is growing or falling, and therefore there is no way of introducing a control u to compensate for this. Consequently, the use of feedback is inevitable for most control problems. In the present problem, it is reasonably obvious that simple, proportional output feedback, as shown in the next figure, will be satisfactory.

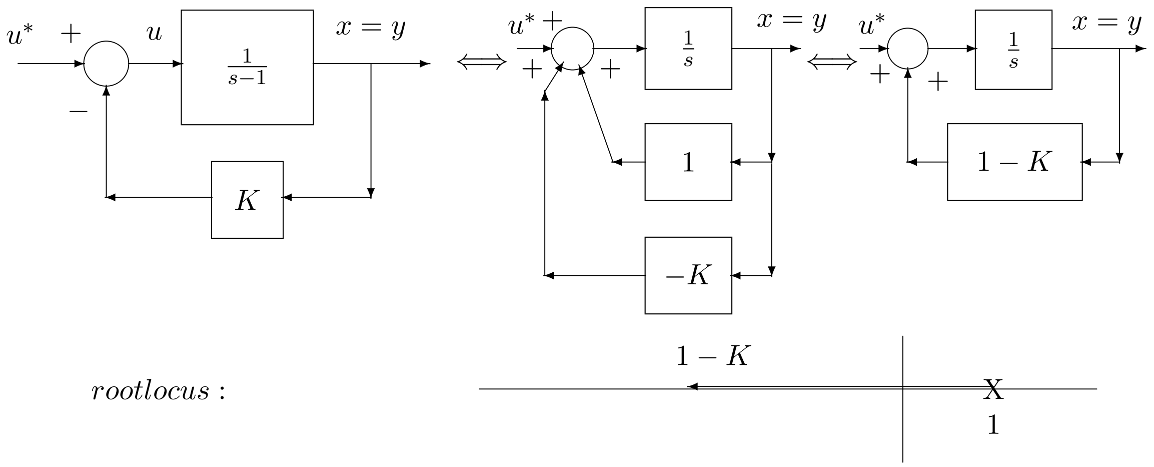


Figure 5.6: First order plant with proportional feedback.

The state equations are simply:

$$\dot{x} = x - Kx + u^* \qquad y = x \tag{5.24}$$

so that the closed loop transfer function becomes:

$$H(s) = \frac{\frac{1}{s-1}}{1 + \frac{K}{s-1}} = \frac{1}{s + K - 1} \tag{5.25}$$

Notice that there is no cancellation of the pole at $s = 1$ by a zero but that the pole is "pulled over" the instability border, the imaginary axis, by means of the feedback by $-K$

which "adds" to the "open loop" feedback, being simply 1. So this solution is excellent, because, by choosing K, we can position the closed loop pole wherever we want. This looks very promising but we must ask how the method works for more complicated processes. Thus, consider the process being extended by an integration, e.g. from the actuator, resulting in:

$$P(s) = \frac{1}{s-1}.\frac{1}{s} = \frac{1}{s^2 - s} \tag{5.26}$$

Ergo, the describing differential equation is given by:

$$\ddot{y} - \dot{y} = u \tag{5.27}$$

We immediately observe that this second order system has a negative damping term $-\dot{y}$ (negative friction, negative resistance, ...) which causes the instability. For stabilisation we should at least effect a positive damping of minimally $\dot{y}$ to compensate for the internally introduced energy. But the strategy up till now, i.e. proportional control, only feeds back the y itself, so:

$$u = -Ky + u^* \tag{5.28}$$

yielding the characteristic equation for the closed loop system:

$$s^2 - s + K = 0 \tag{5.29}$$

whose roots are at

$$s = \frac{1}{2} \pm \frac{1}{2}\sqrt{1-4K} \tag{5.30}$$

This is the rootlocus for the open loop system with a pole in the origin and at $s = 1$ with proportional feed back as shown in the next figure.

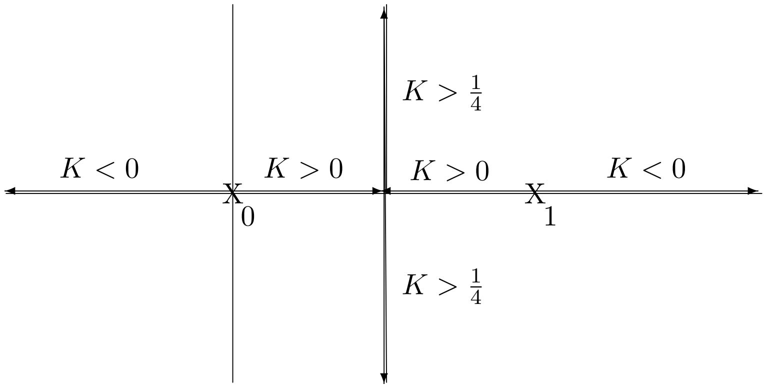


Figure 5.7: Second order plant with proportional feedback.

It is clear that no stable system can be obtained in this way. We obviously have to add a damping term by feedback, i.e. a term containing the derivative of the output $\dot{y}$ as proposed in the next Fig. 5.8.

We can also interprete this as needing a zero to pull the rootlocus over the imaginary axis. By the ideal PD-control we have actually fed back:

$$u = u^* - K_1 y - K_2 sy = u^* - K_2(s + \frac{K_1}{K_2})y = u^* - K_2(s + \delta)y \tag{5.31}$$

so the zero lies at $-\delta = -K_1/K_2$. For a fixed δ the rootlocus for K_2 is then given in Fig. 5.9.

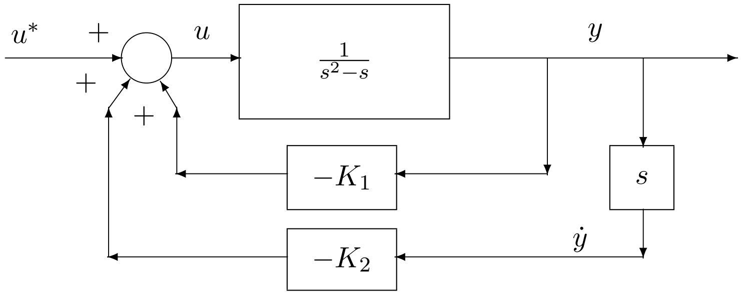


Figure 5.8: Second order plant with PD-control.

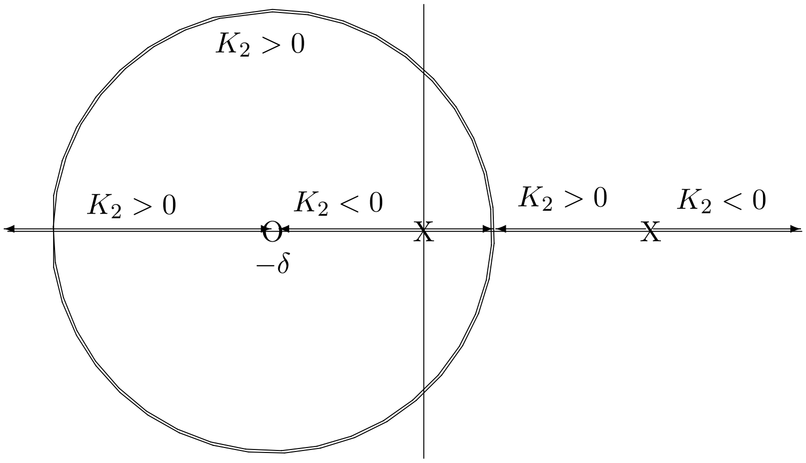


Figure 5.9: Rootlocus for K_2 for system Fig.5.8.

So by all means we can place the poles everywhere on the real axis and on all circles around δ. Consequently by proper choice of K_1 and K_2 we can put the poles at all places we like. This is also easy to observe from the characteristic polynomial for the closed loop system:

$$s^2 + (K_2 - 1)s + K_1 \tag{5.32}$$

which can turn into any arbitrary polynomial by choosing K_1 and K_2. Moreover, the same statement holds for any original second-order process of the form $P(s) = 1/(s^2 + a_1 s + a_2)$ and not only for $P(s) = 1/(s^2 - s)$. This is a nice result, but how general is it? Consider now a process with the transfer function:

$$P(s) = \frac{(s-1)(s-3)}{s(s-2)(s-4)} \tag{5.33}$$

With the applied PD-controller the new characteristic polynomial would be:

$$s(s-2)(s-4) + K_2(s-1)(s-3)(s+\delta) \qquad \delta = \frac{K_1}{K_2} \tag{5.34}$$

The rootlocus for gain K_2 for a particular choice of δ are as in the next plot 5.10.

We see that stabilisation, not to mention arbitrary pole location, cannot be achieved by whatever choice of δ and K_2 (i.e. K_1 and K_2). This is not unexpected, since we deal with a third order process and we are using only two parameters in the feedback. We might try to introduce more paramaters by also using the acceleration feedback (so

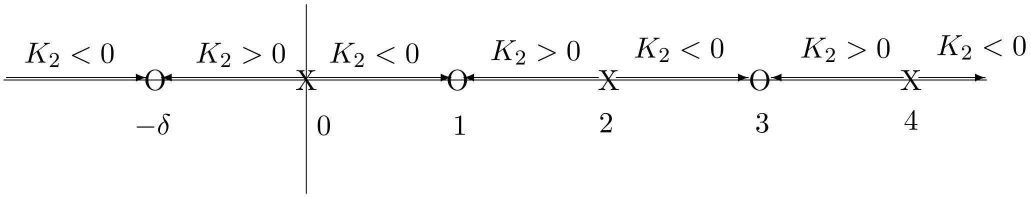


Figure 5.10: Rootlocus of ver unstable plant.

actually PDD-control), but such an extra differentiation of the output y is not feasible. Due to inevitable noise, real differentiation causes indefinitely high peaks. Besides, even if possible, it will not work: extra zeros at the left half plane cannot change the rootloci on the positive real axis. (Citing freely Kailath [2], we remark:)

"Historically, it was at this stage that it was not clear what exactly could or could not be done. It was at some such point of vague confusion that, motivated by Kalman's work on state-variable descriptions of linear systems, Rissanen argued that instead of feeding back y and its derivatives [or feeding back y through a linear compensation network] , the proper thing to feed back was the *state* x of a realisation of the process, since, after all, the state summarises all the "current information" about the process. Therefore, anything we can do with y or $\dot{y}$ etc., we must also be able to do with the states and, more important, anything we cannot do with the states probably cannot be done in any other general way."

In the second order example this is quite obvious if one studies the following Fig. 5.11. Here the PD-control of the second order plant has been displayed in a rearranged block-scheme so that it is obvious that actually a *state* feedback was implemented.

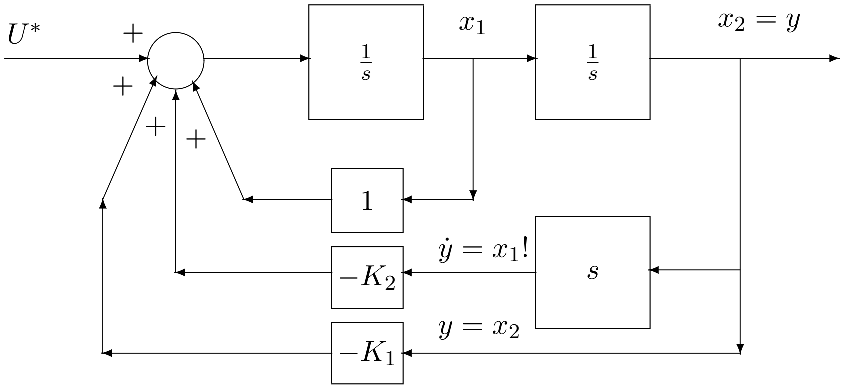


Figure 5.11: PD-control as disguised state control.

Rissanen showed that state feedback could be used to modify at will the modes of the process and, in particular, to make them all stable, provided only that the realisation used to define the states of the process is state controllable, i.e. the controllability matrix $[B\ AB\ A^2B\ \ldots\ A^{n-1}B]$ is of rank n.

This is a striking result and a good justification of the importance of the concept of state. However, as stressed by Rissanen, the usefulness of this result is dependent on our ability to obtain the states. Nevertheless, partly for clarity of discussion and also for historical and pedagogical reasons, we shall treat the two problems - of controlling and observing the states - separately for a while. For the moment we assume that, by some means, the states can be made available.

As far as the unresolved third order problem is concerned, we will return to this problem when all theory is presented so that we then can show how this problem can elegantly be solved.

5.3 State-variable feedback, pole location

Motivated by the discussion in the previous section, we shall consider the following problem. We are given a minimum realisation:

$$\begin{array}{rcrcr} \dot{x} & = & Ax & + & Bu \\ y & = & Cx & & \end{array} \tag{5.35}$$

with an input vector u of p inputs, an output vector y of q outputs and a state vector x of n states. All state space matrices have proper dimensions and $\{A, B\}$ is controllable. Let the characteristic polynomial be given by:

$$a(s) = \det(sI - A) = s^n + a_1 s^{n-1} + \ldots + a_n \tag{5.36}$$

We wish to modify the given system by the use of state-variable feedback so as to obtain a new system with specified eigenvalues or, equivalently, a specified characteristic polynomial, say

$$\alpha(s) = s^n + \alpha_1 s^{n-1} + \ldots + \alpha_n \tag{5.37}$$

Now state-variable feedback, or shortly state feedback, is obtained by the substitution

$$u = u^* - Lx \tag{5.38}$$

where u^* is the new external input (vector) and L is a constant matrix of proper dimensions (pxn). Fig. 5.12 represents this operation in block-scheme.

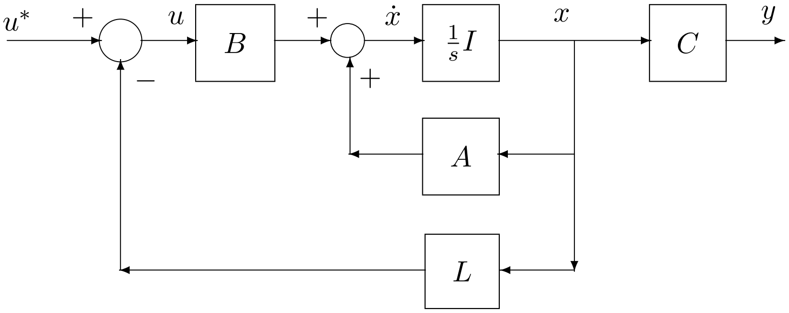


Figure 5.12: Realisation modified by state feedback

Note that in this block-scheme all lines represent vectors. The integration block operates upon all state variables simultaneously so that it contains an identity matrix I of dimension n. It will be clear from this blockscheme that, in open loop, the integrator block is only fed back by block A, while, in closed loop, block A is bypassed by the concatenation of blocks B and L and, not forgetting, the minus sign. Of course this is also reflected in the formulas by substituting 5.38 into 5.35 yielding:

$$\begin{array}{rcrcr} \dot{x} & = & (A - BL)x & + & Bu^* \\ y & = & Cx & & \end{array} \tag{5.39}$$

It is clear that the new characteristic polynomial is given by :

$$\alpha(s) = \det(sI - A + BL) \tag{5.40}$$

It appears that by proper choice of feedback matrix L we can accomplish any proposed monic polynomial $\alpha(s)$. (Monic simply means that the coefficient of the highest power is one.) Ergo, we can theoretically place the poles of the closed loop system wherever we want. The proof is given now for the case that there is only a single input so p=1. It will be clear that for more inputs it will then certainly be possible, as we have more freedom in L. For the single input process we may transform the state space representation by a proper similarity transformation $x^* = Tx$ into the controller canonical form (also called phase-variable form):

$$\begin{array}{ccccccccc} \dot{x}^* & = & T\dot{x} & = & TAT^{-1}Tx & + & TBu & = & A^*x^* & + & B^*u \\ y & = & CT^{-1}Tx & & & & & = & C^*x^* & & \end{array} \tag{5.41}$$

This canonical form shows the following structure:

$$A^* = TAT^{-1} = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 \\ \vdots & 0 & 1 & \cdots & \cdot \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ -a_n & -a_{n-1} & \cdot & \cdots & -a_1 \end{pmatrix} \tag{5.42}$$

while the input matrix B changes into:

$$B^* = TB = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix} \tag{5.43}$$

If we would feed back these canonical states by

$$u = -L^*x^* + u^* \tag{5.44}$$

it is readily observed that the dynamics of the closed loop system is determined by the new state matrix:

$$A^* - B^*L^* \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 \\ \vdots & 0 & 1 & \cdots & \cdot \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ -a_n - l_n^* & -a_{n-1} - l_{n-1}^* & \cdot & \cdots & -a_1 - l_1 \end{pmatrix} \tag{5.45}$$

when

$$L^* = \begin{pmatrix} l_n^* & l_{n-1}^* & l_{n-2}^* & \cdots & l_1^* \end{pmatrix} \tag{5.46}$$

By proper choice of L^* we can accomplish that

$$\alpha_i = a_i + l_i^* \tag{5.47}$$

so that, indeed, the closed loop system obeys the prescribed dynamics. It is then only a detail to conclude that the feedback rule for the original state variables is derived from equation 5.44 as:

$$u = -L^*x^* + u^* = -L^*Tx + u^* = -Lx + u^* \quad \Rightarrow \quad L = L^*T \tag{5.48}$$

So theoretically we find ourselves in the comfortable position of being in full control of the system poles. We are then tended to place the poles as far as possible from the origin in the left half plane in order to create very broad-banded and thus very fast systems. However, this would mean very big polynomial coefficients α_i which in turn implies large feedback coefficients l_i. This high feedback gains would finally cause a gigantic input signal u, certainly saturating the actuator(s). Consequently, practical implementation puts its constraints on the actual pole positions. This aspect (and later on the acquisition of state measurements) needs further study. In the next section we will deal with the problem of actuator saturation by explicitly defining a control criterion, where, apart from considering the state variables x, we also weight the cost of the input signal u.

5.4 Linear Quadratic Regulator

(The title of this section will become clear later on.)

Again we consider a state feedback, i.e. we suppose that all states are available and we don't bother so much about the actual outputs so that we depart from the familiar part of the state space equation:

$$\dot{x} \quad = \quad Ax \quad + \quad Bu \tag{5.49}$$

For zero input u the equilibrium state is $x = 0$, but we suppose that at initial time t_0 (mostly $t_0 = 0$) the state is given by $x(t_0) = x_0 \neq 0$. This initial value can be due to starting conditions or by previous disturbances. As a matter of fact, this is not relevant, but we do care to reducing the state value to zero as fast as possible.

The result of the previous section suggests that, if $\{A, B\}$ is controllable, we can obtain a finite energy input by putting $u = -Lx$, so that

$$\dot{x} = (A - BL)x \tag{5.50}$$

and by choosing L suitably large, we can make x decay to zero as fast as we wish (theoretically). The rate of decay depends on how negative the real parts of the eigenvalues of $A - BL$ are. The more negative these are, the larger the values of L will be and therefore the higher the required signal energy. These facts suggest that we should try to make a trade-off between the rate of decay of x and the amplitude of the input. In the *quadratic regulator problem* , this is done by choosing u to minimise:

$$J = \frac{1}{2}\int_{t_0}^{t_f} (x^T Q x + u^T R u) dt + \frac{1}{2} x^T(t_f) P_f x(t_f) \tag{5.51}$$

subject to $Q \geq 0$, $R > 0$, $P_f > 0$, $x(t_0) = x_0$, $\{A, B\}$ controllable, and certainly:

$$\dot{x} = Ax + Bu \tag{5.52}$$

By choice of R, Q, t_f and P_f we can give different weights to the cost of control and the cost of deviations from the desired state which is 0 for all t. The choice of these quantities is again more an art than a science and in most of the applications there is just an initial guess. After computation of the optimal control for the given criterion, the closed loop behaviour is studied and unwanted effects are corrected by adjusting the previous weightings for a second iteration and so on, until satisfactory results are obtained.

Our interest here lies in the fact that for $t_f = \infty$ the optimal control turns out (as we will show) to be a *linear* feedback control, i.e.:

$$u = -Lx \tag{5.53}$$

where L depends on the parameters $\{A, B, Q, R, \}$ but is *not* a function of t. This is of course exactly the kind of state feedback, studied in section 5.3 and clearly the optimum feedback gain must be associated with an optimum set of pole positions. This is called the *steady state* ($t_f = \infty$) quadratic regulator theory as a solution to the *linear quadratic control problem.* Evidently, "linear" refers to the ultimate feedback law $u = -Lx$ and "quadratic" describes the integrant in the criterion.

The solution of above problem can be straightforwardly derived from the Euler-Lagrange equations and proper final time condition as presented in the first half of this course. As we have:

$$\begin{gathered} \dot{x} = Ax + Bu = f(x, u) \\ F = \tfrac{1}{2}(x^T Q x + u^T R u) \\ \Phi = \tfrac{1}{2} x^T(t_f) P_f x(t_f) \end{gathered} \tag{5.54}$$

The Hamiltonian is clearly defined by:

$$H = F + \lambda^T f = \frac{1}{2}(x^T Q x + u^T R u) + \lambda^T (Ax + Bu) \tag{5.55}$$

The Euler-Lagrange/condition equations

$$\dot{\lambda} = \frac{-\partial H}{\partial x} \quad \lambda(t_f) = \frac{\partial \Phi}{\partial x}(t_f) \quad \frac{\partial H}{\partial u} = 0 \tag{5.56}$$

consequently yield:

$$\begin{aligned} \dot{\lambda} &= -A^T \lambda - Qx \\ \lambda(t_f) &= P_f x(t_f) \\ Ru + B^T \lambda &= 0 \end{aligned} \tag{5.57}$$

Because we penalised all inputs by taking R nonsingular, we may invert R and express u as a function of λ:

$$u = -R^{-1} B^T \lambda \tag{5.58}$$

Substituting this control u in the process equation:

$$\dot{x} = Ax + Bu \tag{5.59}$$

then finally results into the following set of 2n equations:

$$\begin{pmatrix} \dot{x} \\ \dot{\lambda} \end{pmatrix} = \begin{pmatrix} A & -BR^{-1}B^T \\ -Q & -A^T \end{pmatrix} \begin{pmatrix} x \\ \lambda \end{pmatrix} \overset{\text{def}}{=} S \begin{pmatrix} x \\ \lambda \end{pmatrix} \tag{5.60}$$

but with mixed, or two-point boundary conditions:

$$x(t_0) = x_0 \quad \lambda(t_f) = P_f x(t_f) \tag{5.61}$$

Figure 5.13 represents above equations in a block scheme.

From this figure it is clear that the controller can be implemented as soon as we have available $\lambda(t_0)$, but we only know that $\lambda(t_f) = P_f x(t_f)$. We can try to extent this last relation to all time t by defining:

$$\lambda(t) = P(t) x(t) \tag{5.62}$$

So let us substitute $\lambda = Px$ into equation 5.60:

$$\begin{aligned} \dot{x} &= Ax - BR^{-1}B^T P x \\ \dot{P}x + P\dot{x} &= -Qx - A^T P x \end{aligned} \tag{5.63}$$

By eliminating $\dot{x}$ we finally get:

$$(\dot{P} + PA + A^T P - PBR^{-1}B^T P + Q)x = 0 \tag{5.64}$$

Since equation 5.64 has to hold for all x(t), the coefficient matrix is zero for all t, that is known as the

$$Riccati \quad Equation(RE): \tag{5.65}$$

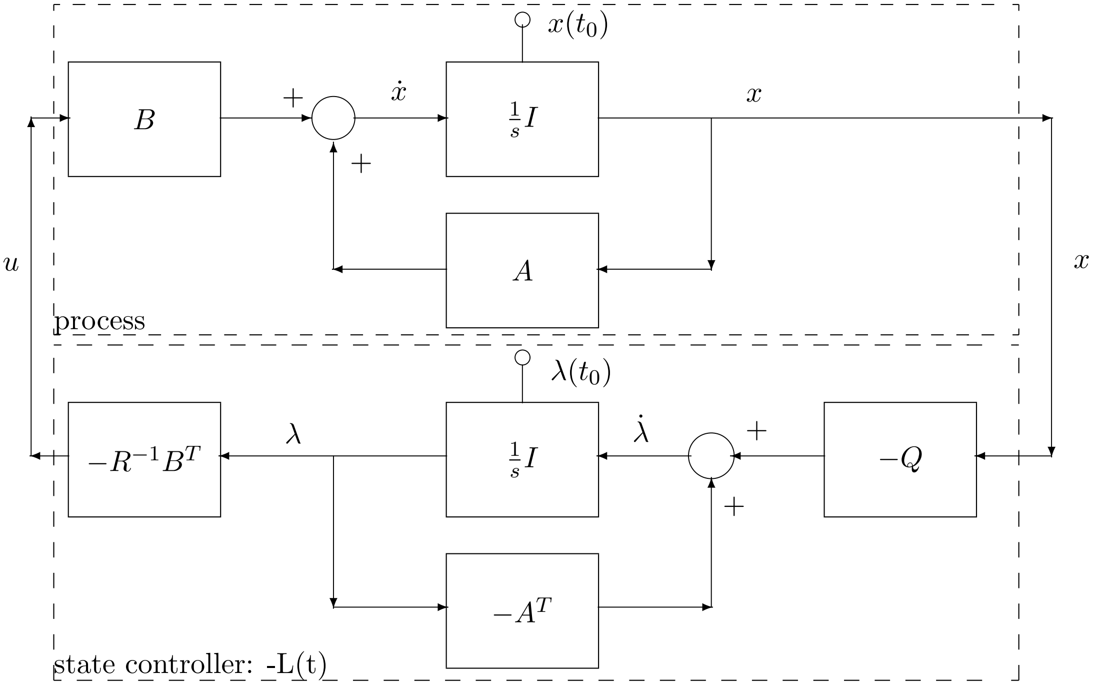


Figure 5.13: General state x and costate λ realisation for optimal linear quadratic problem.

$$\boxed{-\dot{P} = PA + A^T P - PBR^{-1}B^T P + Q}$$

with the terminal condition:

$$P(t_f) = P_f \tag{5.66}$$

Consequently we have to solve this quadratic, first order, matrix differential equation *backwards* in time. If we have done this off line and in advance, and we could do this on an analogue or digital computer, we have available $P(t)$ for all t and we can simply obtain the control input from:

$$Linear \;\; State \;\; Control \tag{5.67}$$

$$\boxed{u(t) = -R^{-1}B^T\lambda(t) = -R^{-1}B^T P(t)x(t) = -L(t)x(t)}$$

So, obviously, the complete control block in Fig. 5.13 has simply be replaced by the time dependent coefficient matrix $-L(t)$. Alternatively, we could conclude that we also have available $P(t_0)$ and thereby $\lambda(t_0) = P(t_0)x(t_0)$ so that we can simply run equations 5.60 or use directly block scheme 5.13. This last alternative will soon turn out to be ill conditioned though.

Furtheron, note that we have not made use of the time independence of matrices A, B, Q, and R. So, even if these matrices are time dependent, the solution is obtained along exactly the same lines.

By step-wise integration we can indeed solve the Riccati equation, but let us analyse more carefully its characteristics and in particular its behaviour in cases where t_f approaches infinity. Because both Q and R are symmetric, it is easy to derive from the given Riccati equation that $P(t)$, being symmetric and positive definite at time t_f as P_f, remains symmetric and positive definite for *all* time $t < t_f$. It appears that, if the process is controllable, the solution of the Riccati equation converges (quickly) to $\bar{P}$, which is the unique *positive definite* solution of the

Algebraic Riccati Equation (ARE) (5.68)

$$\boxed{\bar{P}A + A^T\bar{P} - \bar{P}BR^{-1}B^T\bar{P} + Q = 0}$$

obtained by putting $\dot{P} = 0$.

It will be clear that one always arrives at this solution from the (dynamic) Riccati Equation for $t_f \to \infty$, whatever the final P_f might have been. As a matter of fact, the P_f becomes completely irrelevant, because for $t_f \to \infty$ it will be clear that $x(t)$ can only contain stable modes so that $x(t_f) = 0$ for $t_f \to \infty$. Apparently, also $\lambda(t)$ contains only stable modes for the same reason as we have $\lambda(t_f) = P(t_f)x(t_f) \to 0$ for $t_f \to \infty$.

This brings us to another method for computing the optimal state controller. It can be proved that the state matrix S of equation 5.60 has its poles *symmetric* with respect to the imaginary axis:

$$\det(sI - S) = \left[\Pi_{i=1}^n (s - p_i)\right]\left[\Pi_{i=1}^n (s + p_i)\right] \tag{5.69}$$

where p_i are the poles in the left half plane. This implies that exactly half the number of poles refer to unstable modes that should not be excited in case that we have a criterion lasting until infinity, i.e. $t_f = \infty$. Since neither x nor λ can contain unstable modes because we want to minimise J and $\lambda(t_f) = 0$ for $t_f \to \infty$, apparently the initial values of $\lambda(t_0)$ have to be chosen such that these unstable modes are *not* excited. If so, only n stable modes determine the dynamical behaviour illustrated in Fig. 5.13 and effectively there are only n independent states corresponding to the stable poles:

$$\begin{aligned} x(t) &= \underbrace{\begin{pmatrix} e_1 & e_2 & \dots & e_n \end{pmatrix}}_{E} \begin{pmatrix} e^{p_1 t} \\ e^{p_2 t} \\ \vdots \\ e^{p_n t} \end{pmatrix} \\ \lambda(t) &= \underbrace{\begin{pmatrix} f_1 & f_2 & \dots & f_n \end{pmatrix}}_{F} \begin{pmatrix} e^{p_1 t} \\ e^{p_2 t} \\ \vdots \\ e^{p_n t} \end{pmatrix} \end{aligned} \tag{5.70}$$

where e_i and f_i are the column vectors of E and F. Consequently by putting $\lambda = \bar{P}x$ we get $F = \bar{P}E$ so that finally $\bar{P} = FE^{-1}$. Furthermore the vectors $\begin{pmatrix} e_i \\ f_i \end{pmatrix}$ are the eigenvectors corresponding to the stable eigenvalues p_i of the matrix S.

From this little excursion about the eigenvalues of matrix S it also becomes more evident that $\bar{P}$ should be the *positive definite* solution of the ARE while there are many others. Above we have selected the n stable poles from the total $2n$ poles. We could (wrongly) have chosen another set of n poles leading to a solution $\bar{P}$ of the ARE but not causing a stabilising control. In principle we can choose n poles out of $2n$ in $\left(\begin{array}{c} 2n \\ n \end{array}\right)$ways all leading to unstable closed loop systems except one which corresponds to the *positive definite* $\bar{P}$. Returning to the main line, we may now conclude that for $t_f \rightarrow \infty$, but in practice for sufficiently large t_f, we simply get:

$$Constant\ \ State\ \ Feedback \tag{5.71}$$

$$\boxed{u(t) = -R^{-1}B^T\lambda(t) = -R^{-1}B^T\bar{P}x(t) = -Lx(t)}$$

So, like in the pole placement algorithm, we have a *constant* state feedback gain:

$$L = R^{-1}B^T\bar{P} \tag{5.72}$$

This time, the resultant L is a compromise between our wish to bring the state to zero as fast as possible and the limitations of the actuator(s). The implementation is straightforward: the state control block in Fig. 5.13 is simply replaced by a constant matrix L. Certainly, theoretically we could use the realisation with the costate λ by giving the proper initial value $\lambda(t_0) = \bar{P}x(t_0)$, but this is very tricky, because any small numerical or implementation error will excite the unstable modes. And even in the theoretical case that it could be given the exact initial value, the inevitable noise would do the job. So the only save way is to feed back with the constant L because, as we already learned in the pole placement section, the unstable poles are simply not there and can thus never be excited by accident.

5.4.1 A simple, one dimensional, numerical example

If we take a SISO-process of the first order all matrices turn into scalars and we can do all computations analytically. In order to stress that we deal with scalars we will use small characters corresponding to the capitals for the matrices. Let the numerical values be given by:

a=4, b=1, q=9, r=1, p_f=4, t_f >0

Notice that we deal here with an unstable process as it has a pole at $a = 4$. Then the Riccati equation (RE) is defined by:

$$p^2 - 8p - 9 = (p-9)(p+1) = \dot{p} \tag{5.73}$$

which we can solve easily as:

$$\frac{dp}{(p-4)^2 - 25} = dt \tag{5.74}$$

by integration of:

$$\int \frac{d\left(\frac{p-4}{5}\right)}{1 - \left(\frac{p-4}{5}\right)^2} = \int -5dt \tag{5.75}$$

If $|\frac{p-4}{5}| < 1$ then $\text{atanh}\left(\frac{p-4}{5}\right) = c_0 - 5t$ so that we have:

$$\frac{p-4}{5} = \tanh(c_0 - 5t) \tag{5.76}$$

If the integration constant c_0 is expressed in a new constant c_1 according to:

$$e^{c_0} = c_1 e^{5t_f} \tag{5.77}$$

we can write the tanh-function explicitly yielding:

$$p = 4 + 5\frac{c_1 e^{-5(t-t_f)} - c_1^{-1} e^{5(t-t_f)}}{c_1 e^{-5(t-t_f)} + c_1^{-1} e^{5(t-t_f)}} \tag{5.78}$$

Now it is easy to see that, for $t = t_f$, $p(t_f) = p_f = 4$ so that $c_1 = 1$.

For $t << t_f$ it holds that $e^{5(t-t_f)} << e^{-5(t-t_f)}$ so that $p(t) \approx \bar{p} = 9$. Starting at t_f and going into the negative time direction, this steady state solution $\bar{p}$ is reached relatively fast. In Fig. 5.14 this is shown for $t_f = 2$.

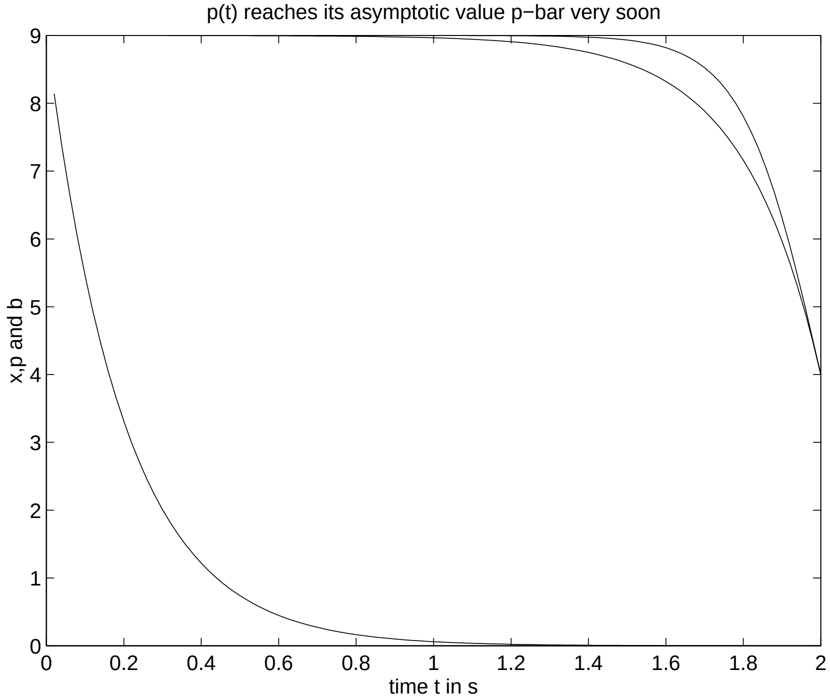


Figure 5.14: The behaviour of $P(t)$ (upper right curve) compared to the resultant $x(t)$ lower left curve. The function $b(t) = 4 + 5(1 - \exp\{-5(t_f - t)\})$ can be found as the less curved function upper right, which converges also to $\bar{P} = 9$ for $t \downarrow 0$.

For reasons of comparison we have also drawn the curve $b(t) = 4 + 5(1 - e^{-5(t-t_f)})$ i.e. when p(t) would converge (in negative time) from $p(t_f) = p_f = 4$ to $\bar{p} = 9$ with a time constant $1/5 = .2$. The actual convergence with the atanh-function, represented by the upper curve is faster indeed.We have chosen as a comparison for that particular time constant .2, because this appears to be the time constant of the closed loop system in steady state:

For $\bar{p} = 9$ we get as constant state feedback $l = r^{-1}b\bar{p} = 9$ so that the pole of the closed loop system is determined by $s - a + bl = s + 5 = 0$, ergo a pole of -5, implying the time constant .2. Consequently, as long as $p \approx \bar{p}$ from time $t = 0$ on, we obtain for the closed loop that $x(t) = x(0)e^{-5t}$ as represented in Fig. 5.14. Only at a time t close to $t_f = 2$ the actual p no longer equals the constant $\bar{p}$ but that hardly influences $x(t)$, which is effectively zero at that time. So we observe that if the final time $t_f >>$ twice the (largest) time constant of the closed loop system in steady state, we may very well approximate the solution of the Riccati equation (RE) $p(t)$ by the steady state solution of the algebraic Riccati equation (ARE) $\bar{p}$ and thus suppose $t_f \approx \infty$.

This algebraic Riccati equation (ARE) is straightforwardly given by:

$$(\bar{p} - 9)(\bar{p} + 1) = 0 \tag{5.79}$$

The positive definite solution $\bar{p} = 9$ is the proper one as we have seen. The corresponding controller $u(t) = -9x(t)$ shifts the original, unstable pole at $a = 4$ towards the stable pole at $a - bl = -5$. Had we chosen the wrong solution $\bar{p} = -1$, we would have obtained an unstable closed loop pole at $a - bl = 4 + 1 = 5$. Indeed, this corresponds to the proper pole at 5 mirrored with respect to the imaginary axis. These both poles can also be found directly as the eigenvalues of system matrix S:

$$S = \begin{pmatrix} a & -br^{-1}b \\ -q & -a \end{pmatrix} = \begin{pmatrix} 4 & -1 \\ -9 & -4 \end{pmatrix} \tag{5.80}$$

so that:

$$|sI - S| = (s - 4)(s + 4) - 9 = s^2 - 25 = 0 \quad \Rightarrow \quad s_{1,2} = \pm 5 \tag{5.81}$$

5.4.2 Angular velocity stabilisation problem

Again we will study a first order SISO-process so that the solution can be obtained via a tanh-function. We will not do it explicitly here but focus on two aspects:

1. How to transform a practical problem into the mathematical framework we have learned to solve.

2. What is the effect of the design parameters q, r, t_f and p_f?

The process under study consists of a DC-motor, the shaft of which has the angular velocity $x(t)$ and which is driven by the input voltage $u(t)$. The process is described by the scalar differential equation:

$$\dot{x}(t) = ax(t) + bu(t) \tag{5.82}$$

with $a = -.5s^{-1}$ and $b = 150rad/(Vs^2)$. We want to study the problem of stabilising the angular velocity $x(t)$ at the desired value ω_0. In the formulation of the general regulator problem, we have chosen the origin of the state space as the equilibrium point. Since in the present problem the desired equilibrium position is $x(t) = \omega_0$, we simply shift the origin. Let u_0 be the constant input voltage to which ω_0 corresponds as the steady state angular velocity. Then u_0 and ω_0 are related by :

$$0 = a\omega_0 + bu_0 \quad \Rightarrow \quad u_0 = -\frac{a}{b}\omega_0 \tag{5.83}$$

Introduce the new state variable x' and input variable u':

$$x'(t) = x(t) - \omega_0 \quad u'(t) = u(t) - u_0 \tag{5.84}$$

so that:

$$\dot{x}(t) = \dot{x}' \quad x(t) = x'(t) + \omega_0 \quad u(t) = u'(t) + u_0 \tag{5.85}$$

By substitution in eq. 5.82 and application of eq. 5.83, it is easy to see that the dashed system indeed obeys the original differential equation:

$$\dot{x}'(t) = ax'(t) + bu'(t) \tag{5.86}$$

but whereas originally the state had to be brought from an arbitrary initial $x(0) = \omega_1$ to a final $x(t_f) = \omega_0$ we now have initial $x'(0) = \omega_1 - \omega_0$ and final $x'(t_f) = 0$. Thus, without restricting the generality of the example, we will consider the problem of regulating the original system to zero state.

As the optimisation criterion we choose:

$$J = \frac{1}{2}\{\int_0^1 (qx^2 + ru^2)dt + p_1 x^2(1)\} \tag{5.87}$$

so $t_f = 1$ and of course the design variables $q,\ r,\ p_1 > 0$. Certainly, the solution won't be influenced by multiplying J by any positive constant. Apparently, we can choose one of the design variables being one as only the relative values count. We choose to take $q = 1$ so that we only have to study the (relative) influence of r and p_f. Their proper values must be obtained by trial and error as we will see.

The poles of the optimally controlled system are defined by:

$$\det \begin{pmatrix} s-a & b^2/r \\ q & s+a \end{pmatrix} = 0 \tag{5.88}$$

which leads to poles:

$$s_{1,2} = \pm\sqrt{a^2 + b^2\frac{q}{r}} \tag{5.89}$$

These poles are real and nicely mirrored with respect to the imaginary axis. The corresponding Riccati equation is given by:

$$-\dot{p} = 2ap + q - \frac{p^2 b^2}{r} \tag{5.90}$$

If $t_f \to \infty$ the algebraic equation holds, i.e. $\dot{p} = 0$, and this quadratic equation yields:

$$\bar{p} = a\frac{r}{b^2}\left(1 - \sqrt{1 + \frac{b^2}{a^2}\frac{q}{r}}\right) \tag{5.91}$$

Since a is negative, this solution defines the positive definite solution. The optimal steady state solution is thus given by:

$$u = -lx = \frac{b\bar{p}}{r}x = -\frac{ax}{b}\left(1 - \sqrt{1 + \frac{b^2}{a^2}\frac{q}{r}}\right) \tag{5.92}$$

so that the pole of the closed loop system is expressed as:

$$a - bl = a - \frac{b^2\bar{p}}{r} = a\sqrt{1 + \frac{b^2}{a^2}\frac{q}{r}} \tag{5.93}$$

which equals the stable pole found in equation 5.89.

Remarks:

- The closed loop pole (see eq. 5.93) is always *less* than the open loop pole a unless $q = 0$. In the latter case there is no feedback, because we apparently had no interest in weighting x but still the control action was costly: $r \neq 0$. Next we observe that the relative importance q/r, linked to either bringing x to zero or to avoiding costly control u, will determine how far the closed loop pole will be in the left half s-plane. The larger we choose q/r, the faster the closed loop response will be at the expense of a larger feedback control signal u. This effect can be observed in Fig. 5.15 where $q = 1$, $p_1 = 0$ and r=100, 1000 and 10000. The larger r, the more we penalise the control action and the slower the decrease of x is, but certainly the control effort is less.

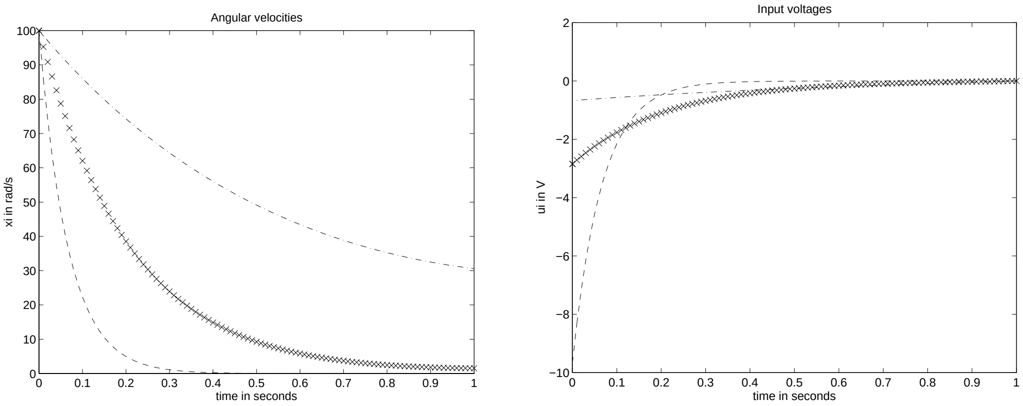


Figure 5.15: The behaviour of the angular velocities $x(t)$ and control inputs $u(t)$ for $p_f = 0$ and r=100 $(- - --)$, r=1000 (xxxx), r=10000 $(- \cdot - \cdot -)$.

- As we are dealing with a finite time $t_f = 1$ optimal control, the control strategy will be time dependent. The solution $p(t)$ of the RE will be time dependent but will soon reach its steady state value $\bar{p}$ from $t < t_f$ on. How soon and how big a $\bar{p}$ depends again on the ratio q/r. The larger r the more time it takes for p to reach its higher steady state $\bar{p}$. This is illustrated in Fig. 5.16.

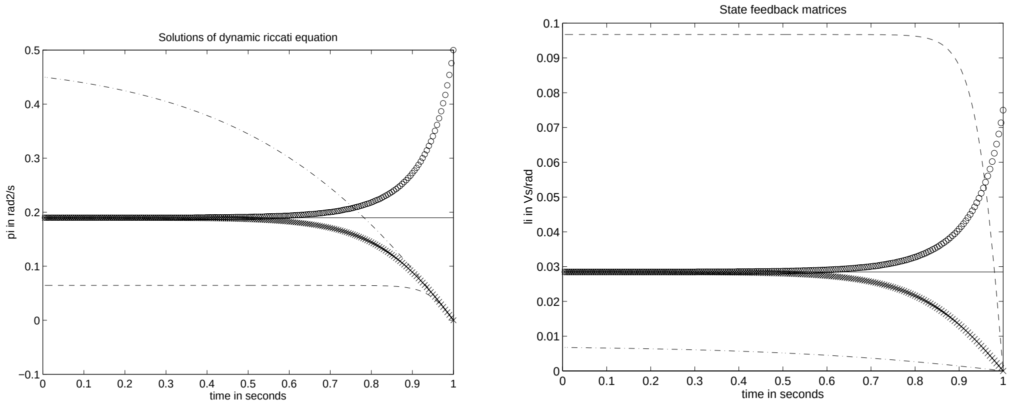


Figure 5.16: The behaviour of $P(t)$ and $L(t)$ for the angular velocity problem. For $p_f = 0$ we have r=100 $(- - --)$ and r=10000$(- \cdot - \cdot -)$. For $r = 1000$ we have various values of p_f: p_f=0 (xxxx), p_f=.19 (———) and p_f=.5 (oooo)

Only for r=10000 we observe that p has not reached steady state within the time span of one second.

.

It may look strange at first sight that $\bar{p}$ is larger for larger r, but for the actual feedback l we have to multiply $\bar{p}$ by b/r and this certainly leads to a smaller feedback for larger values of r as we can see in Fig. 5.16.

For the case that r=1000 we have three final weights p_f. From the previous figures we learned that till time $t \approx .5$ the steady state $\bar{p} = .19$ is practically reached. (For $p_f = \bar{p}$ certainly p(t)=$\bar{p}$ is constant in time.) After the time .5 we deal with really different p(t) but this has very limited effect on the actual control u and state x as Fig. 5.17 reveals.

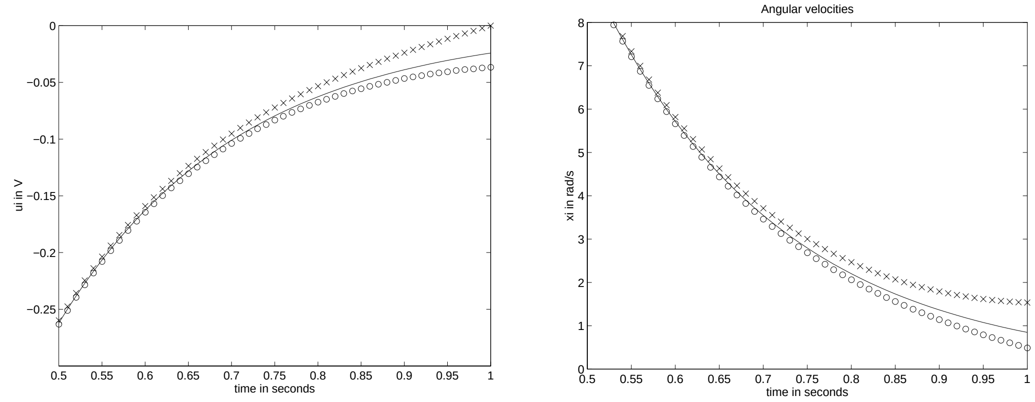


Figure 5.17: The behaviour of inputs $u(t)$ and states $x(t)$ for the angular velocity problem for $r = 1000$ and $p_f = 0$ (xxxx), $p_f = .19$ (——) and $p_f = .5$ (oooo)

5.4.3 Remarks on the cost function

Remark 1: Quite often there is hardly any insight on the weighting of the states by matrix Q because one is only familiar with the outputs y of the process. This is very easy to overcome. If the penalty on y is given by $y^T Q_y y$, we simply use the relation $y = Cx$ to arrive at $x^T C^T Q_y Cx$ so that we obtain: $Q = C^T Q_y C$. (If we deal with a biproper system: $y = Cx + Du$, we can substitute as well, but this will yield cross terms of u and x. We refer to textbooks for this more complicated problem.)

Remark 2: For the final time $t_f = \infty$, the minimum value of the cost function $Jmin$ equals a simple quadratic function of the initial $x(0)$, viz.:

$$J_{min} = \frac{1}{2} x^T(0) \bar{P} x(0) \tag{5.94}$$

The proof is very straightforward and can shortly be given as follows:

Because $J = \frac{1}{2}\int_0^\infty (x^T Qx + u^T Ru)dt$ and for the optimum $u = -R^{-1}B^T\bar{P}x$ we have:

$$J_{min} = \frac{1}{2}\int_0^\infty x^T(Q + \bar{P}BR^{-1}B^T\bar{P})x\ dt\ = \tag{5.95}$$

$$\frac{1}{2}\int_0^\infty x^T(-\bar{P}A - A^T\bar{P} + 2\bar{P}BR^{-1}B^T\bar{P})x\ dt \tag{5.96}$$

by means of the ARE in $\bar{P}$. Furthermore, the closed loop state equation $\dot{x} = (A - BR^{-1}B^T\bar{P})x$ can be used to obtain:

$$\frac{d(x^T\bar{P}x)}{dt} = x^T\bar{P}\dot{x} + \dot{x}^T\bar{P}x = \tag{5.97}$$

$$x^T(\bar{P}A - \bar{P}BR^{-1}B^T\bar{P})x + x^T(A^T\bar{P} - \bar{P}BR^{-1}B^T\bar{P})x = \tag{5.98}$$

$$\tag{5.99}$$

$$x^T(\bar{P}A + A^T\bar{P} - 2\bar{P}BR^{-1}B^T\bar{P})x \tag{5.100}$$

Recognising this as the (negative) integrant of equation 5.95 we thus obtain:

$$J_{min} = \frac{1}{2}\int_0^\infty \frac{d(-x^T\bar{P}x)}{dt}dt = \frac{1}{2}\int_{t=0}^{t=\infty} d(-x^T\bar{P}x) = -\frac{1}{2}x^T\bar{P}x|_0^\infty = \frac{1}{2}x^T(0)\bar{P}x(0) \tag{5.101}$$

QED.

Note that the minimum cost function is a function of the initial state but the actual control gain $L(t)$ is *not*. Whatever the initial condition is, the controller itself is independent of $x(0)$. It is precisely this property which makes this controller suitable for the reduction of (white) state noise effects as will be shown in the next subsection.

5.4.4 Stochastic regulator problem

So far we discussed the deterministic, linear, optimal regulator problem. The solution of this problem allowed us to tackle purely transient problems, where a linear process has a perturbed initial state, and it is required to return the process to the zero state as quickly as possible while limiting the input amplitude. Practical problems exist, which can be formulated in this manner but much more common are problems where there are disturbances which act uninterruptedly upon the process, and that tend to drive the state away from zero. The problem is then to design a feedback configuration through which initial effects are reduced as quickly as possible but which feedback also counteracts the effects of disturbances as much as possible in the steady state situation. If the disturbances can be interpreted as white noise acting on the states, one can roughly imagine that at each moment the noise forces the state into a new initial state. We have learned that the deterministic regulator is able to quickly reduce the initial state to zero. In steady state this is done by a constant state feedback L. Since superposition holds, we can imagine that each moment is a initial moment for a new problem with all problems having the same solution in terms of the controller L. It is precisely this characteristic that enables us to show that the deterministic regulator similarly functions as an optimal stochastic regulator.

Problem definition.

The effect of disturbances can be accounted for by suitably extending the process description:

$$\begin{array}{rcccccc} \dot{x} & = & Ax & + & Bu & + & v \\ y & = & Cx & + & w & & \end{array} \tag{5.102}$$

We suppose that v and w are white noise sequences defined by the expectations:

$$E[v(t)] = 0 \quad E[w(t)] = 0 \tag{5.103}$$
$$E[v(t)v^T(t+\tau)] = R_v\delta(\tau) \tag{5.104}$$
$$E[w(t)w^T(t+\tau)] = R_w\delta(\tau) \tag{5.105}$$
$$E[v(t)w^T(t+\tau)] = R_{vw}\delta(\tau) \tag{5.106}$$

The variance matrices R_* are of proper dimensions and $\delta(\tau)$ is the Dirac function. If the actual state disturbance is *filtered* white noise, the filter itself should be represented in state space and the corresponding states be added to the state representation of the deterministic part of the process. Furthermore, as we still assume that the full state vector can be observed without any disturbance, the effect of w is irrelevant here and will be studied in the chapter concerning the optimal observers.

We would like to minimise the effect of the disturbance v in the state x by minimising the steady state error criterion:

$$\tilde{J} = \frac{1}{2}E[x^TQx + u^TRu] \tag{5.107}$$

where Q and R are the familiar weighting matrices and under the constraint:

$$u(t) = -Lx(t) \tag{5.108}$$

It appears that the solution of this stochastic regulator problem is exactly the same as the solution to the deterministic steady state regulator problem provided that we indeed restrict the solution to a *linear* regulator $u = -Lx$. Note that this restriction was not necessary for the deterministic regulator problem. If the noise is Gaussian, this restriction is unnecessary too and the linear controller appears to be the optimal one among all possible also nonlinear controllers. We will not prove this and confine to linear controllers altogether.

Proof.

As we know the structure of the controller $u = -Lx$ we can rewrite the deterministic and stochastic criteria:

$$2J = \int_0^\infty (x^TQx + x^TL^TRLx)dt = \int_0^\infty (x^TWx)dt = \text{trace}(W\int_0^\infty xx^Tdt) \tag{5.109}$$
$$2\tilde{J} = E[x^TQx + x^TL^TRLx] = E[x^TWx] = E[\text{trace}(Wxx^T)] = \text{trace}(W\ E[xx^T]) \tag{5.110}$$

where by definition $W = Q + L^TRL$ and remembering: trace ("spoor" in Dutch) is a linear operator yielding the sum of the diagonal elements with the property trace(ABC) = trace(CAB) = trace(BCA). Furthermore linear operators may be interchanged such as trace, Σ, $\int$, $\frac{d}{dt}$, E, multiplication by a constant.

Trivially we have in Laplace domain

- for the deterministic case: $x(s) = [sI - A + BL]^{-1}x(0)$
- for the stochastic case: $x(s) = [sI - A + BL]^{-1}v(s)$

By inverse Laplace transform we can obtain the transition matrix H_x:

$$H_x(t) = \mathcal{L}^{-1}[(sI - A + BL)^{-1}] \tag{5.111}$$

and thereby define x as follows:

- in the deterministic case: $x(t) = H_x(t)x(0)$
- in the stochastic case: $x(t) = \int_0^\infty H_x(\tau)v(t-\tau)d\tau$

Substitution in respectively 5.109 and 5.110 yields:

- $$J = \frac{1}{2}\,\mathrm{trace}(W\int_0^\infty H_x(\tau_1)x(0)x^T(0)H_x^T(\tau_1)d\tau_1) \tag{5.112}$$

- $$\tilde{J} = \frac{1}{2}\,\mathrm{trace}(WE[\int_0^\infty H_x(\tau_1)v(t-\tau_1)d\tau_1 \int_0^\infty v^T(t-\tau_2)H_x^T(\tau_2)d\tau_2]) = \tag{5.113}$$
$$\frac{1}{2}\,\mathrm{trace}(W\int_0^\infty\int_0^\infty H_x(\tau_1)E[v(t-\tau_1)v^T(t-\tau_2)]H_x^T(\tau_2)d\tau_1 d\tau_2) = \tag{5.114}$$
$$\frac{1}{2}\,\mathrm{trace}(W\int_0^\infty\int_0^\infty H_x(\tau_1)R_v\delta(\tau_1-\tau_2)H_x^T(\tau_2)d\tau_1 d\tau_2) = \tag{5.115}$$
$$\frac{1}{2}\,\mathrm{trace}(W\int_0^\infty H_x(\tau_1)R_vH_x^T(\tau_1)d\tau_1) \tag{5.116}$$

Note that J contains $x(0)x^T(0)$ where $\tilde{J}$ has R_v. This is the only difference between J and $\tilde{J}$. The optimal control solution L for J is independent of $x(0)$ so we can take any value for x(0) we like. Think of n different initial values $x_i(0)$ corresponding to n deterministic criteria J_i. Each J_i is optimised by the same L. Consequently a supercriterion J_s consisting of the sum of the respective criteria J_i will also be optimised by the same L. We have all freedom to choose the various $x_i(0)$ and propose $x_i(0)$ such that:

$$\sum_{i=1}^{n} x_i(0)x_i(0)^T = R_v \tag{5.117}$$

Then it simply holds that:

$$J_s = \sum_{i=1}^{n} J_i = \tilde{J} \tag{5.118}$$

again because of interchanging of linear operators.

QED

Finally, from above proof we can easily derive a simple expression for the optimum criterion $\tilde{J}_{min}$:

$$\tilde{J}_{min} = \sum_{i=1}^{n} J_{i,min} = \frac{1}{2}\sum_{i=1}^{n} x_i(0)^T\bar{P}x_i(0) = \tag{5.119}$$
$$\frac{1}{2}\,\mathrm{trace}(\bar{P}\sum_{i=1}^{n} x_i(0)x^T(0)) = \frac{1}{2}\,\mathrm{trace}(\bar{P}R_v) \tag{5.120}$$

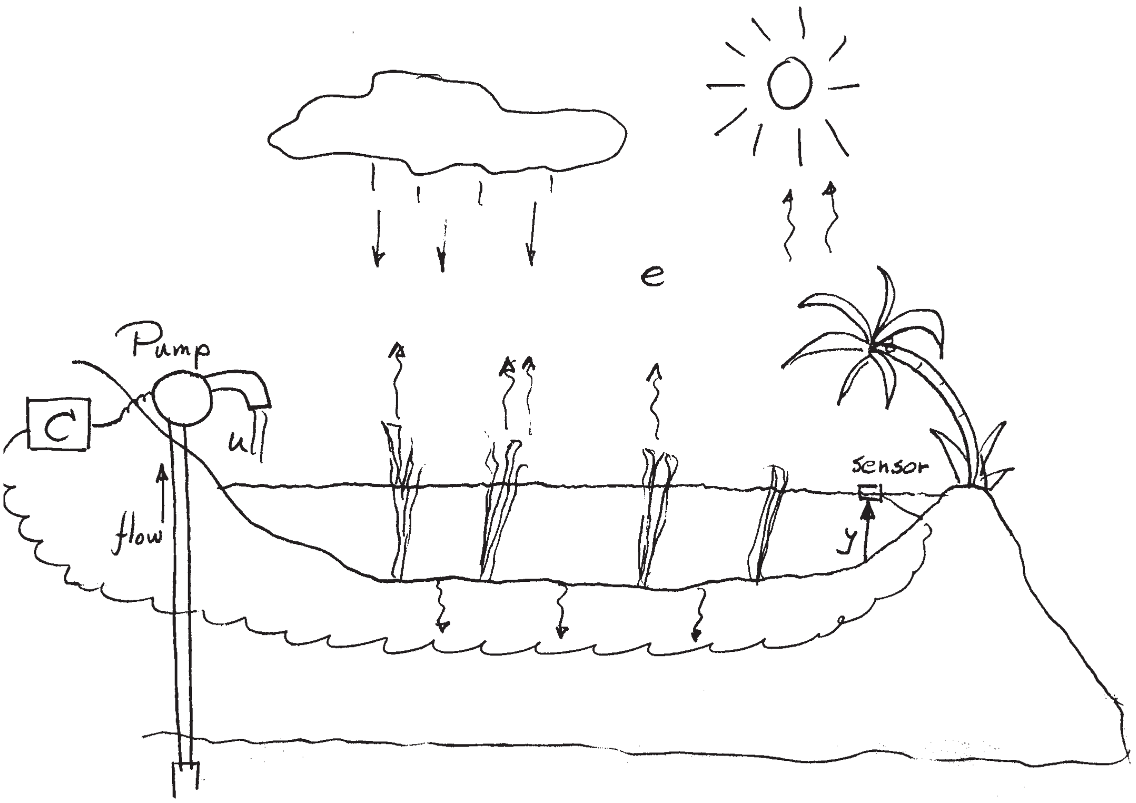


Figure 5.18: Water management for a rice field.

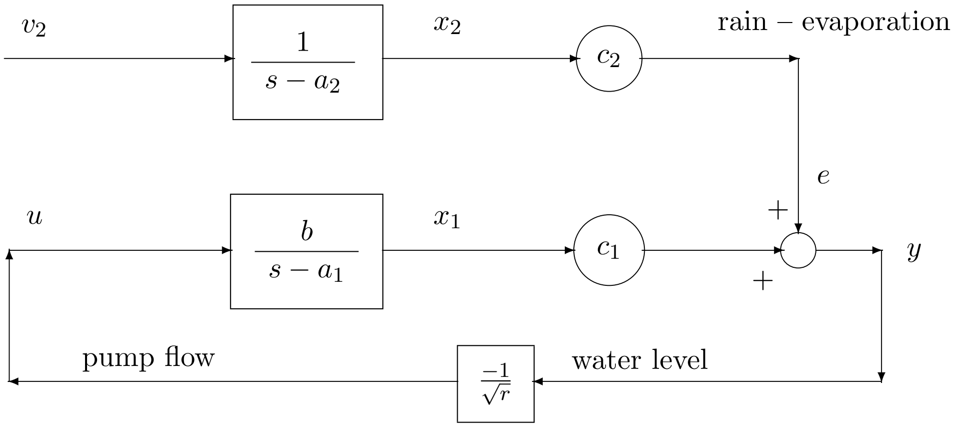


Figure 5.19: A simple irrigation model

5.4.5 Example: irrigation.

As an example to show the stochastic control we present the water management problem of a rice field in Fig. 5.18 and propose a simple model of an irrigation as shown in Fig. 5.19. The lower block $b/(s-a_1)$ together with gain c_1 represents the dynamics of the water-level in a field if there were no input flow. Like the water vessel in the practical training set up, this can be approximated by a first order process with a time constant equal to the "capacity" of the field multiplied by the "resistance" of the soil for the water leakage. We can measure the water-level y and control it by a proper controller in a feedback together with a pump yielding an input water flow u. The disturbing water supply is caused by rainfall and evaporation and represented by low pass $(c_2/(s-a_2))$ filtered, white noise v_2. Low pass, because it takes the weather hours, if not days, to substantially change the water flow "from above". So we note that the actual, disturbance free process can be described by only one state x_1 but we need an extra state x_2 to represent the low pass character of the disturbance. It will be clear that this last state x_2 is not controllable $(b_2 = 0)$, while the first state x_1 has no disturbance $(v_1 = 0)$.

The total process is described by the following state space description:

$$\begin{pmatrix} \dot{x_1} \\ \dot{x_2} \end{pmatrix} = \begin{pmatrix} a_1 & 0 \\ 0 & a_2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} b \\ 0 \end{pmatrix} u + \begin{pmatrix} 0 \\ v_2 \end{pmatrix} \tag{5.121}$$

$$y = \begin{pmatrix} c_1 & c_2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \tag{5.122}$$

where we have taken $b = 1$, $c_1 = 1$, $c_2 = 1$, $a_1 = -1$, $a_2 = -2$ and v_2 is white, zero mean, Gaussian noise. Its autocorrelation is $R_v\delta(\tau)$ and we take $R_v = 1$. This implies theoretically an infinitely large variance because all frequencies contribute equally. Compare the course "Stochastic signal Theory". In practice, here the simulation in Simulink, it is sufficient to take the bandwidth of the flat spectrum far beyond the low pass filter bound of the plant, thus $>> \omega_B = a_1$.

Surely, this set of linear equations describes the process in an equilibrium point. We want to keep the water level constant on the operating value, which implies that we want to keep y as close to zero as possible given the limits of the pump. The criterion is given by:

$$\tilde{J} = \frac{1}{2} E[y^2 + ru^2] \tag{5.123}$$

where r will be varied until a satisfactory control is obtained. Using equation 5.122 turns the x-weighting matrix Q into:

$$Q = \begin{pmatrix} c_1^2 & c_1c_2 \\ c_1c_2 & c_2^2 \end{pmatrix} \tag{5.124}$$

so that the ARE can be written as:

$$\begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} p_1 & p_0 \\ p_0 & p_2 \end{pmatrix} \begin{pmatrix} a_1 & 0 \\ 0 & a_2 \end{pmatrix} + \begin{pmatrix} a_1 & 0 \\ 0 & a_2 \end{pmatrix} \begin{pmatrix} p_1 & p_0 \\ p_0 & p_2 \end{pmatrix} + \\ \begin{pmatrix} c_1^2 & c_1c_2 \\ c_1c_2 & c_2^2 \end{pmatrix} - \begin{pmatrix} p_1 & p_0 \\ p_0 & p_2 \end{pmatrix} \begin{pmatrix} b \\ 0 \end{pmatrix} 1/r \begin{pmatrix} b & 0 \end{pmatrix} \begin{pmatrix} p_1 & p_0 \\ p_0 & p_2 \end{pmatrix} \tag{5.125}$$

by parametrising $\bar{P}$ as:

$$\bar{P} = \begin{pmatrix} p_1 & p_0 \\ p_0 & p_2 \end{pmatrix} \tag{5.126}$$

Above *three* (because of symmetry) quadratic equations in p_0, p_1 and p_2 can be solved analytically, yielding:

$$p_1 = \frac{a_1 r}{b^2}\left(1 \pm \sqrt{1 + \frac{c_1^2 b^2}{a_1^2 r}}\right) \tag{5.127}$$

$$p_0 = \frac{-c_1 c_2}{a_1 + a_2 - p_1 b^2/r} \tag{5.128}$$

$$p_2 = \frac{b^2 p_0^2/r - c_2^2}{2a_2} \tag{5.129}$$

The positive definite solution $\bar{P}$ is obtained, according to Sylvester, if we take $p_1 > 0$ and $\det(\bar{P}) > 0$. This leads us to the plus sign in equation 5.127. Consequently, the optimal feedback law defines:

$$u(t) = -Lx(t) = -\frac{1}{r}\begin{pmatrix} b & 0 \end{pmatrix}\begin{pmatrix} p_1 & p_0 \\ p_0 & p_2 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \tag{5.130}$$

The upper plots of Figs. 5.20 and 5.21 show the resultant water-levels y and necessary control flows u for respectively $r = 10^{-2}$ and $r = 10^{-4}$. Note that for smaller r the flow u can and will be bigger, resulting in a smaller and higher frequent level y. This can be explained as follows.

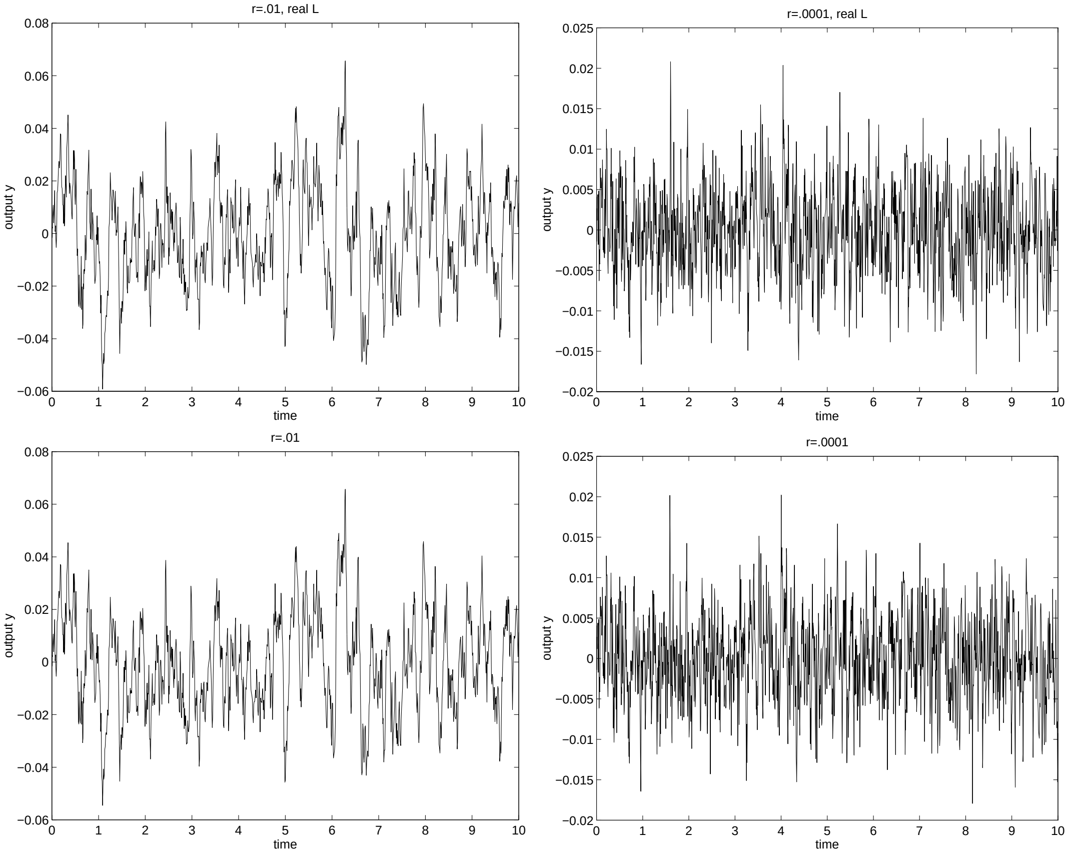


Figure 5.20: The reduced disturbance on the output y for $r = 10^{-2}$ at the left and $r = 10^{-4}$ at the right. The upper plots represent the optimum, while the lower plots show the result for the approximate control of equation 5.134.

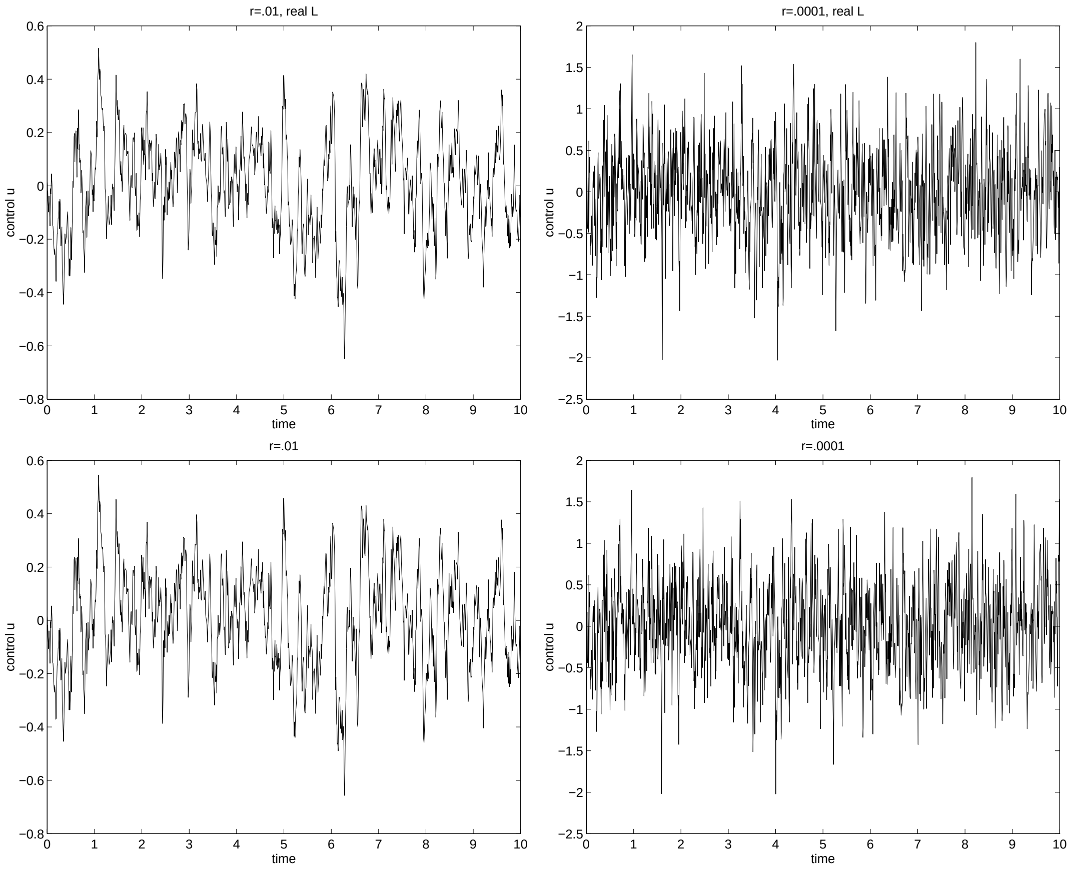


Figure 5.21: The necessary control input u for $r = 10^{-2}$ at the left and $r = 10^{-4}$ at the right. The upper plots represent the optimum, while the lower plots show the result for the approximate control of equation 5.134.

Because r is very small compared to the entries in Q we may in approximation take the zero and first order terms of a Taylor expansion in $\sqrt{r}$ which results into:

$$p_0 \approx \frac{c_2}{b}\sqrt{r} \tag{5.131}$$

$$p_1 \approx \frac{c_1}{b}\sqrt{r} \tag{5.132}$$

$$p_2 \approx 0 \tag{5.133}$$

so that the feedback gain L becomes:

$$L = \frac{1}{r}\begin{pmatrix} b & 0 \end{pmatrix}\begin{pmatrix} \frac{c_1\sqrt{r}}{b} & \frac{c_2\sqrt{r}}{b} \\ \frac{c_2\sqrt{r}}{b} & 0 \end{pmatrix} = \frac{1}{\sqrt{r}}\begin{pmatrix} c_1 & c_2 \end{pmatrix} \tag{5.134}$$

This is very fortunate because

$$y = \begin{pmatrix} c_1 & c_2 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \tag{5.135}$$

so that we have a simple output feedback:

$$u(t) = -\frac{1}{\sqrt{r}}y(t) \tag{5.136}$$

as represented in Fig. 5.19. Usually we have to build an observer to obtain an estimate of the state vector x(t) from output measurements y, contaminated with measurement noise, as we will discuss in the next chapter. Just here, where we ignore the measurement noise and where we apply approximated formulas in $\sqrt{r}$, we arrive at such a simple expression. In Figs. 5.20 and 5.21 we may compare in the lower plots the effect of the approximation and we conclude that there is hardly any difference. The approximation highly facilitates the analysis of the control as we deal with a simple SISO feedback. The component e of y represents the full disturbance and its frequency contents is simply given by:

$$\frac{c_2}{j\omega - a_2} = \frac{1}{j\omega + 2} \tag{5.137}$$

This is the low pass curve in Fig. 5.22.

The closed loop process functions as a (sensitivity) filter :

$$\frac{y}{e} = \frac{j\omega - a_1}{j\omega - a_1 + bc_1/\sqrt{r}} = \frac{j\omega + 1}{j\omega + 1 + 1/\sqrt{r}} \tag{5.138}$$

Note that this filter will let all frequencies pass above its pole, thus $\omega > |-1-1/\sqrt{r}|$, but this pole shifts to higher and higher values for smaller r. The lower frequencies will be filtered out especially below its zero, so for $\omega < |-1|$, where the reduction is approximately $\sqrt{r}$. This is the high pass filter shown in Fig. 5.22 for $r = 10^{-4}$. Evidently, the smaller r the more the lower frequencies are filtered out and the broader the filter band will be, but the capacity of the pump will put an upper limit on this. The heavier and more expensive the pump will be, the better the ultimate controlled behaviour can become.

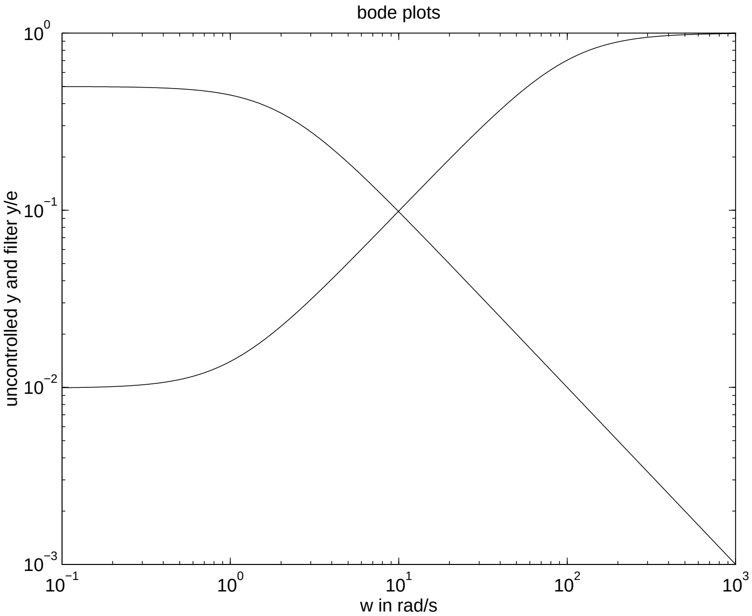


Figure 5.22: Bodeplots of disturbance e (low pass) and filter transfer y/e (high pass) for $r = 10^{-4}$ with approximate feedback of equation 5.134.

CRITICS!

Although this example excellently backs the theory, two restrictive remarks should be made:

- The water from the pump is indistinguishable from the water from rain as soon as both are contained in the rice field. So x_1 cannot be measured independently from x_2, but it also means that the modelling is altogether wrong. Also the water from the rain leaks trough the soil! Ergo the disturbance e should not be added to the output of the process but to the input. If you make this correction it turns out that the same, approximate, proportional controller results. Later on in the text we will still use the uncorrected model as "parallel" states are easier to analyse than "serial" states, that occurs when state x_2 precedes state x_1 in case of the correctly modeled input disturbance.

- The found controller may function well theoretically, in practice there are severe drawbacks. For a reasonable flow the pump should be of high power so that the "servo amplifier" becomes extremely expensive. For lowering the costs we have to use a simple switch as an actuator before the pump. Then we are back to the nonlinear systems that we analysed with the describing functions and where we synthesised a controller by means of Pontryagin. This is a nice theoretical exercise. A (nonoptimal) solution could be:

  if $y < r - \varepsilon$ **then** $u = u_{max}$ **else** $u = 0$

  If the level is too high a simple overflow at the palm-tree in Fig. 5.18 will do. This is the kind of control at your toilet and is very cheap and sufficient.

Nevertheless there remain plenty of applications where the proposed LQR-control performs much better e.g. the control of wafer-steppers (ASML) where we combine high demands with very good modeling and less contraints at costs.

Chapter 6

The continuous, optimal observer problem.

In section 5.3 we showed that, if a realisation $\{A, B, C\}$ is controllable (better: reachable), then state-variable feedback, $u = -Lx$, can modify the poles, i.e. the eigenvalues of $A - BL$ at will. The problem is the acquisition of the state-values at each moment t. Even in the exceptional case that we are able and willing to measure all states, which is very expensive, we will have to deal with measurement noise. In general we measure only a limited set of outputs $y_i(t)$ with inevitable noise. We shall now discuss the problem of actually obtaining (an estimate of) the states for the particular realisation in use from knowledge only of the system input $u(t)$ and system output $y(t)$ and a (perfect) model of the plant. In section 5.2 we already alluded to the possibility of obtaining all states by taking a sufficient number of higher time derivatives of the outputs, provided that the realisation $\{A, B, C\}$ is observable (or better: detectable). This technique is clearly impractical because the inevitable noise would soon dominate in the derivatives. In the next section we will therefore develop a more realistic state estimator, which is usually known as an **asymptotic observer**. The name arises from the fact that the states can only be obtained with an error, but one that can be made to go to zero at any specified exponential rate.

In chapter 7 we shall discuss the use of such state estimates in place of the unavailable, true states. We shall find that the design equations for the controller are not affected by the fact that approximate states are being used instead of the true states. More crucially, however, we shall find the important and a priori nonobvious fact that the overall observer-controller configuration is internally stable, which was an issue not completely faced by classical design methods. However, use of the estimated instead of the true states, for feedback, may lead in general to a deterioration of the transient response.

6.1 Asymptotic observers for state measurement

We shall now begin to explore the question of methods of actually determining the states of a realisation:

$$\begin{array}{rcrcl} \dot{x}(t) & = & Ax(t) & + & Bu(t) \\ y(t) & = & Cx(t) & & \end{array} \qquad x(0) = x_0 \tag{6.1}$$

given knowledge only of $y(t)$ and $u(t)$, i.e. x_0 is **not** known. Further on, we suppose to deal with a **minimum** realisation, which says that all n state-variables are both controllable (reachable) and observable (detectable) so that the controllability matrix Γ and the observability matrix Δ have full rank:

$$\Gamma = \begin{pmatrix} B & AB & A^2B & \ldots & A^{n-1}B \end{pmatrix} \qquad \Delta = \begin{pmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{pmatrix} \tag{6.2}$$

When we reflect on the fact that we know $A, B, C, u(t)$ and $y(t)$, which is really quite a lot, we wonder why $x(t)$ cannot be reconstructed by forming a dummy system $\{A, B\}$ and driving it with $u(t)$. The problem is, of course, that we do not know the initial condition $x(0) = x_0$. We will reconstruct that x_0 implicitly by adding later, in a closed loop, the knowledge of $y(t)$. Let us analyse the two consecutive steps:

An open loop observer. If we just consider the modeled behaviour of the real system, we can simply simulate that system and excite this simulation with the same input signal $u(t)$ as the real system as outlined in Fig. 6.1.

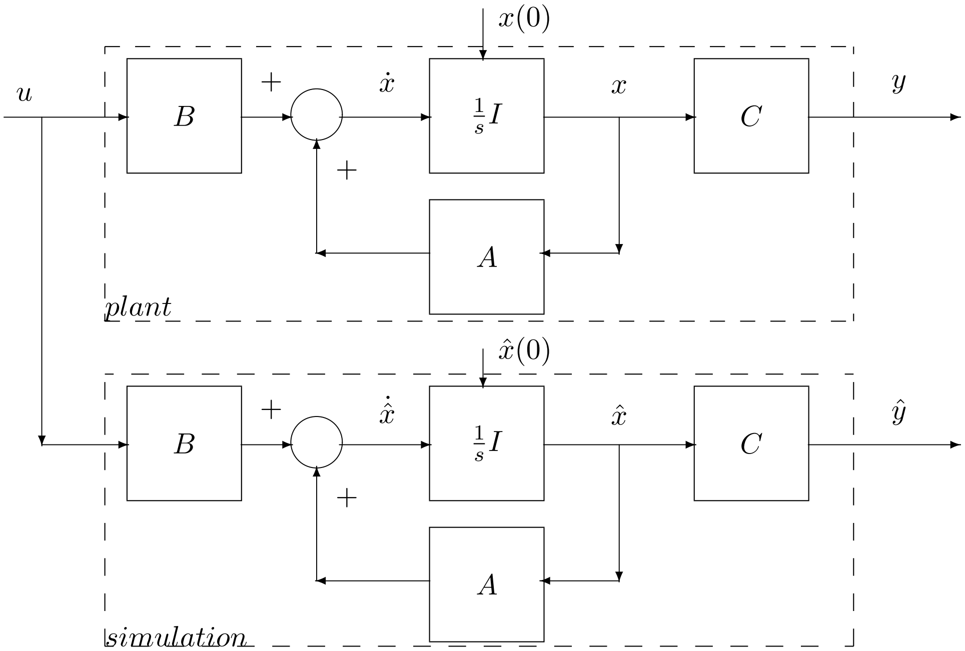


Figure 6.1: An open loop observer

Note that we assume exactly the same realisation $\{A, B, C\}$ for both the real plant and the model. All possible deviations have to be treated as disturbances, which will be discussed under the forthcoming "Kalman filter". The effect of the input $u(t)$ on the real state $x(t)$ and on the estimated state $\hat{x}(t)$ is perfectly well modeled. This is obvious, if we subtract both describing equations:

$$\begin{array}{rcccrc} & \dot{x}(t) & = & Ax(t) & + & Bu(t) \\ & \dot{\hat{x}}(t) & = & A\hat{x}(t) & + & Bu(t) \\ \hline \dot{\tilde{x}}(t) = & \dot{x}(t) - \dot{\hat{x}}(t) & = & A(x(t) - \hat{x}(t)) & = & A\tilde{x}(t) \end{array} \tag{6.3}$$

We have lost the contribution of $u(t)$ in the **error in the states** defined as $\tilde{x}(t) = x(t) - \hat{x}(t)$. Nevertheless the initial state effect is still there and will fade out with the system's own time constants according to:

$$\tilde{x}(t) = H_x(t)\tilde{x}(0) \tag{6.4}$$

where the initial error is given by the misfit in initial states:

$$\tilde{x}(0) = x(0) - \hat{x}(0) \tag{6.5}$$

and the transition matrix is the inverse Laplace transform:

$$H_x(t) = \mathcal{L}^{-1}[(sI - A)^{-1}] \tag{6.6}$$

Clearly, if the system is unstable (recall that we are interested in determining states to be fed back to achieve stabilisation), then the error $\tilde{x}(t)$ will become arbitrarily large as $t \to \infty$, no matter how small the initial error is. Less dramatically, even if the system is stable but some some eigenvalues have real parts that are very small, the effects of errors in the initial estimates will take a long time to die out as shown in Fig. 6.2.

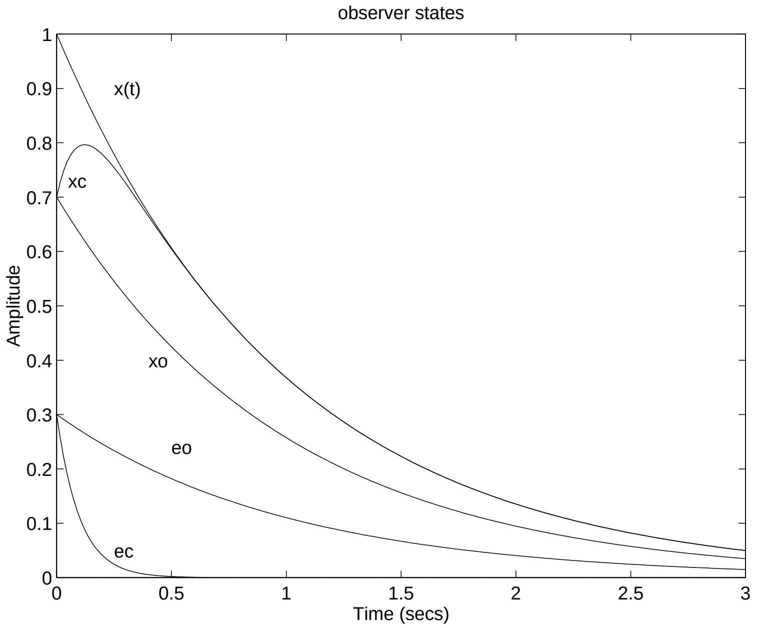


Figure 6.2: State, estimated state and state error for a first order plant. $xo = \hat{x}$, $eo = \tilde{x}$ for open loop observer: $A = -1$; $xc = \hat{x}$, $ec = \tilde{x}$ for closed loop observer: $A = -1$, $C = 1$, $K = 10$

The problem is that the error in the states goes to zero with exactly the same time constant as the state itself. In the curves marked as xc and ec it is shown how a speed-up of the error dynamics would yield a much more acceptable estimate. This speed-up can be accomplished by a feedback scheme as shown in Fig. 6.3.

A closed loop observer. As the missing information in open loop observer concerns the unknown initial value x_0 and since only the measured output y bears information on this aspect, it is obvious that we should involve the measurement y into the observer. In the next Fig. 6.3 the measured output y is compared to the estimated output $\hat{y}$ yielding the output error $e(t) = y(t) - \hat{y}(t)$. This output error is a measure for the

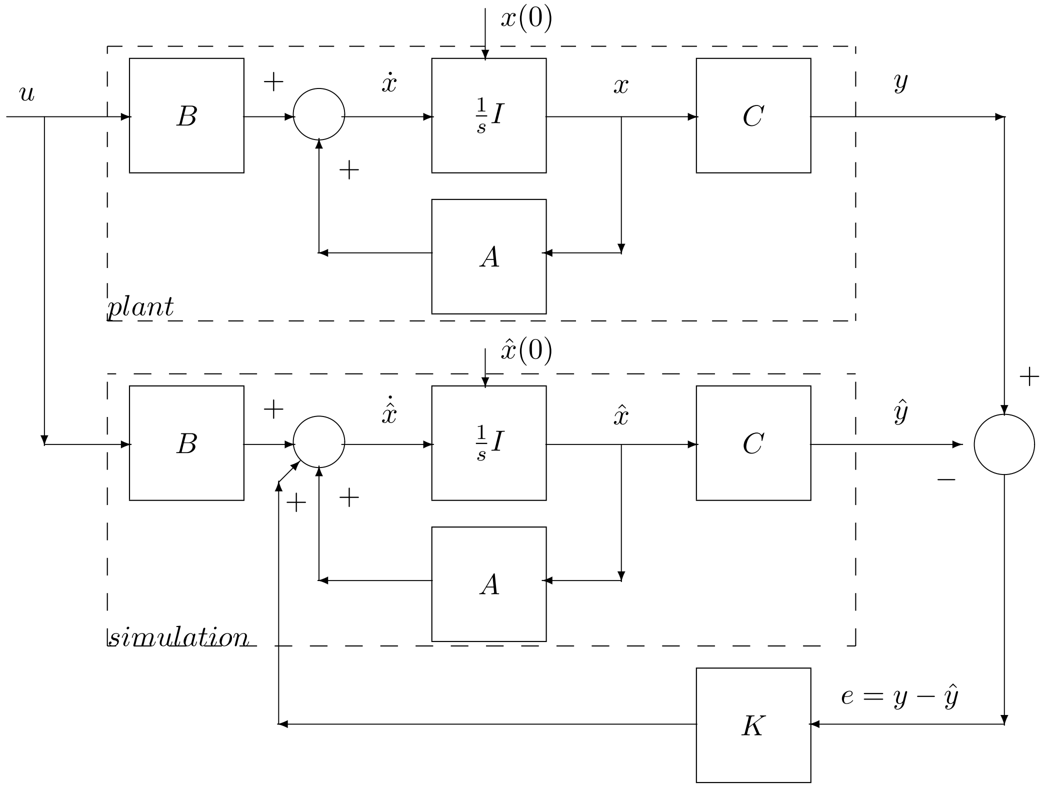


Figure 6.3: Block diagram of an asymptotic observer.

misfit in state estimate $\hat{x}(t)$ and thus is this error fed to "the estimated state input point" with an appropriate scaling matrix K.

In state equations the effect is very straightforward:

$$\begin{array}{rrcl} & \dot{x}(t) & = & Ax(t) + Bu(t) \\ & \dot{\hat{x}}(t) & = & A\hat{x}(t) + Bu(t) + K(y(t) - \hat{y}(t)) \\ \cline{2-4} \dot{\tilde{x}}(t) = & \dot{x}(t) - \dot{\hat{x}}(t) & = & A(x(t) - \hat{x}(t)) - KC(x(t) - \hat{x}(t)) = (A - KC)\tilde{x}(t) \end{array} \tag{6.7}$$

Obviously the dynamics for the error in the states $\tilde{x}(t)$ is now governed by the state matrix $A - KC$ where (theoretically) the coefficient matrix K can be chosen at will. For instance in Fig. 6.2 we have taken $A = -1$, $C = 1$ and $K = 10$ so that the closed loop observer has a pole at $A - KC = -11$. The reduced time constant from an open loop $\tau = 1$ to the closed loop $\tau = 1/11$ is clearly recognisable.

In the multivariable case the effect is completely similar. The matrix K has dimensions $[nxq]$, where q is the number of outputs y_i which is less or equal to the number of states n. The poles of the observer, better of the error in the states $\tilde{x}$, is determined by the determinant of $sI - A + KC$. As the determinant of a matrix or of its transpose is the same we get:

$$\det(sI - A + KC) = \det(sI - A^T + C^T K^T) :: \det(sI - A + BL) \tag{6.8}$$

We observe a dualism between the previous control problem and the actual observer problem by comparing:

controller problem	*observer problem*
A	A^T
B	C^T
L	K^T

(6.9)

As a consequence, by proper choice of K we can position the poles of $A - KC$ wherever we want, because we could do this similarly with the control problem by choosing L. The proof is completely dual.

The same dualism holds for the effect of state disturbance v. If the real state x is permanently disturbed by white noise $v(t)$ we get for the state equation:

$$\dot{x}(t) = Ax(t) + Bu(t) + v(t) \tag{6.10}$$

and by substitution in equation 6.7 we obtain:

$$\begin{array}{rcl} \dot{x}(t) & = & Ax(t) + Bu(t) + v(t) \\ \dot{\hat{x}}(t) & = & A\hat{x}(t) + Bu(t) + K(y(t) - \hat{y}(t)) \\ \hline \dot{\tilde{x}}(t) & = & A(x(t) - \hat{x}(t)) - KC(x(t) - \hat{x}(t)) + v(t) = (A - KC)\tilde{x}(t) + v(t) \end{array} \tag{6.11}$$

so that finally the state error is given by :

$$\tilde{x}(s) = (sI - A + KC)^{-1}v(s) \tag{6.12}$$

As $v(t)$ has a flat power spectrum, being white noise, it is beneficial to choose ”big” values in K so that the poles of $A - KC$ lie far away from the origin in the left half s-plane. The larger the distance to the origin, the smaller $|(j\omega I - A + KC)^{-1}|$ will be so that the effect of the disturbance is less. At the same time the misfit in the initial state will decrease faster. Again we want to position the poles as far as possible to the left as we did for the optimal control problem. There we were confronted with unrealistically high values for $u(t)$ thus saturating the actuator. So these inputs $u(t)$ have put a bound on the pole placement, which we then implicitly defined by a proper control criterion J also weighting $u(t)$. In the observer design the limiting factor is, of course, the output $y(t)$. If we increase K in order to obtain a fast decrease of the error in the initial state estimate, we will put extreme confidence in $y(t)$. A possible measuring error in $y(t)$ will then cause an enormous effect on the estimated state, that we want to avoid of course. Consequently the choice of K should once more be a compromise, which we can define again by a proper observation criterion weighting the measurement noise w on y as well. In the next section we will discuss this criterion. Let us analyse here how the measurement noise $w(t)$ disrupts the picture so far. Suppose that we have to deal with white measurement noise $w(t)$ so that the real measurement becomes:

$$y(t) = Cx(t) + w(t) \tag{6.13}$$

and consequently the state error:

$$\begin{array}{rcl} \dot{x}(t) & = & Ax(t) + Bu(t) + v(t) \\ \dot{\hat{x}}(t) & = & A\hat{x}(t) + Bu(t) + K(Cx(t) - \hat{y}(t)) + Kw(t) \\ \hline \dot{\tilde{x}}(t) & = & (A - KC)\tilde{x}(t) + v(t) - Kw(t) \end{array} \tag{6.14}$$

so that finally the state error is given by :

$$\tilde{x}(s) = (sI - A + KC)^{-1}(v(s) - Kw(s)) \tag{6.15}$$

It will be clear that increasing K for the faster reduction of the initial state misfit and the better reduction of the state disturbance v, will increase the effect of the measurement noise as it has the K as its coefficient.

6.2 The Kalman-Bucy filter.

Let us start by updating the asymptotic observer block diagram with the state disturbance $v(t)$ and the measurement noise $w(t)$ in Fig. 6.4.

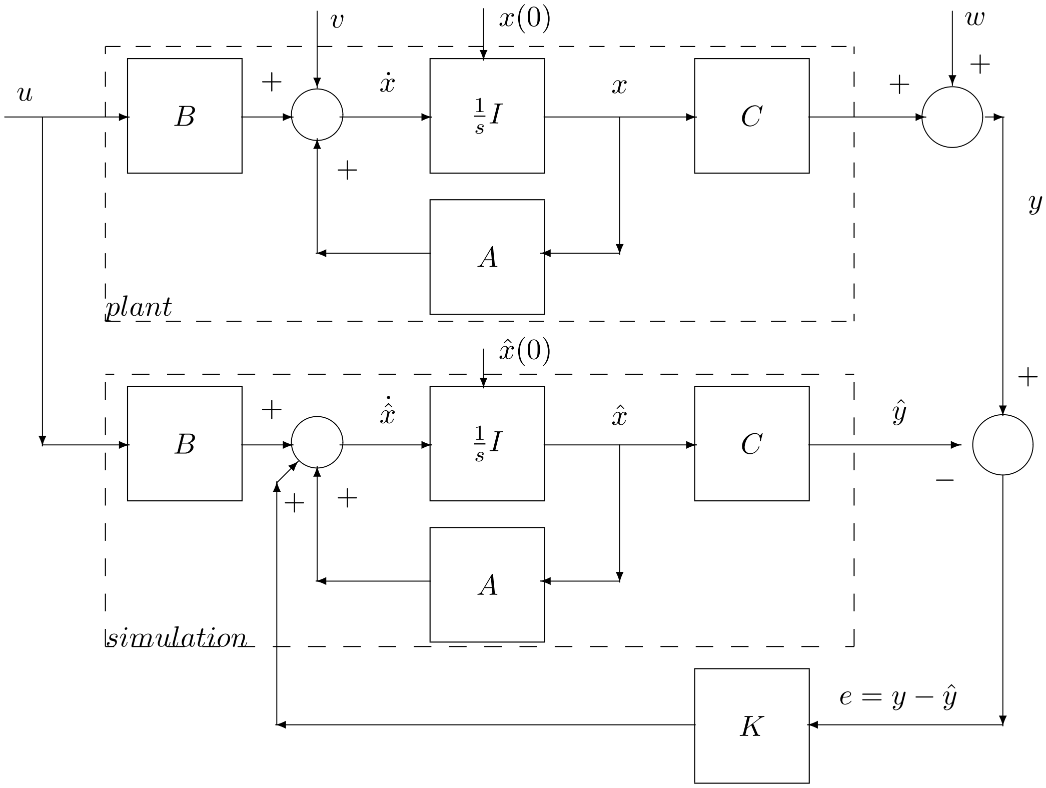


Figure 6.4: Block diagram of an asymptotic observer.

The corresponding, describing equations are:

$$\begin{array}{rcl} \dot{x}(t) & = & Ax(t) + Bu(t) + v(t) \\ y(t) & = & Cx(t) + w(t) \end{array} \tag{6.16}$$

where:

$$\begin{array}{rclrcl} E\{v(t)\} & = & 0 & E\{w(t)\} & = & 0 \\ E\{v(t)v^T(t+\tau)\} & = & R_v\delta(\tau) & E\{w(t)w^T(t+\tau)\} & = & R_w\delta(\tau) \\ E\{v(t)w^T(t+\tau) & = & R_{vw}\delta(\tau) & E\{x(0)\} & = & \bar{x}_0 \end{array} \tag{6.17}$$

The initial state is supposed to be uncorrelated with the state disturbance $v(t)$ and the measurement noise $w(t)$. Furthermore the variance of the initial states (about its expectation) is given by:

$$E\{(x(0) - \bar{x}_0)(x(0) - \bar{x}_0)^T\} = P_0 \tag{6.18}$$

Denoting the misfit in the state estimate again by:

$$\tilde{x}(t) = \hat{x}(t) - x(t) \tag{6.19}$$

while

$$\dot{\hat{x}} = A\hat{x} + Bu + K(y - C\hat{x}) \tag{6.20}$$

we obtain as before:

$$\dot{\tilde{x}} = (A - KC)\tilde{x} + v - Kw \tag{6.21}$$

The optimal stochastic observer can now be defined by:

$$\min_{K,\hat{x}(0)} E\{\tilde{x}^T\tilde{x}\} \tag{6.22}$$

The solution of this minimisation of the quadratic error in the state estimate, for the case that $R_{vw} = 0$, is given by:

- Choose the initial estimate $\hat{x}(0)$ equal to the expectation of the real state $x(0)$, thus:

  $$\hat{x}(0) = \bar{x}_0 \tag{6.23}$$

  This appealing condition appears to effect that $\forall t : E\{\hat{x}(t)\} = x(t)$ or equivalently: $\forall t : E\{\tilde{x}(t)\} = 0$

- The variance of the $\tilde{x}(t)$ denoted by:

  $$E\{\tilde{x}(t)\tilde{x}(t)^T\} = P(t) \tag{6.24}$$

  is minimal (in 2-norm) if we solve the following Riccati equation:

  $$\dot{P}(t) = P(t)A^T + AP(t) + R_v - P(t)C^T R_w^{-1} CP(t) \tag{6.25}$$

  with *initial* condition:

  $$E\{\tilde{x}(0)\tilde{x}(0)^T\} = P(0) = P_0 \tag{6.26}$$

  while we take

- the output error feedback as:

  $$K(t) = P(t)C^T R_w^{-1} \tag{6.27}$$

In steady state condition, where we don't bother about the initial conditions and transients, so for large t, we get simply:

- $$E\{\tilde{x}(t)\} = 0 \tag{6.28}$$

- $$E\{\tilde{x}(t)\tilde{x}(t)^T\} = \bar{P} \tag{6.29}$$

  is constant and minimal (in 2-norm) if we solve the following algebraic Riccati equation:

  $$0 = \bar{P}A^T + A\bar{P} + R_v - \bar{P}C^T R_w^{-1} C\bar{P} \tag{6.30}$$

  while we take

- the output error feedback constant as:

  $$K = \bar{P}C^T R_w^{-1} \tag{6.31}$$

Before analysing the solution and deriving the proof, it is worthwhile to note the dualism which exists between the optimal control and the optimal observer problem. We list all comparable equations:

$$\begin{array}{c|c}
\mathit{controller} & \mathit{observer} \\
\dot{x} = Ax + Bu + v = Ax - BLx + v & \dot{\tilde{x}} = A\tilde{x} + v + K(y - \hat{y}) \\
\dot{x} = (A - BL)x + v & \dot{\tilde{x}} = (A - KC)\tilde{x} + v - Kw \\
\min_L E\{x^T Qx + u^T Ru\} & \min_K E\{\tilde{x}^T \tilde{x}\} \\
L = R^{-1}B^T P & K = PC^T R_w^{-1} \\
(-\dot{P} =)PA + A^T P + Q - PBR^{-1}B^T P = 0 & (\dot{P} =)PA^T + AP + R_v - PC^T R_w^{-1} CP = 0 \\
(t \leftarrow P_f) & (P_0 \rightarrow t)
\end{array} \tag{6.32}$$

The solutions expressed in matrices are completely equivalent if we apply the following dual transforms:

$$\begin{array}{c|c}
\mathit{controller} & \mathit{observer} \\
A & A^T \\
Q & R_v \\
B & C^T \\
R & R_w \\
L & K^T \\
P & P
\end{array} \tag{6.33}$$

Note that the same symbol P is used for both the control and the observer problem. In both cases P is the solution of a Riccati equation. The meaning is completely different though.

If K is taken as above, following the Riccati equation, it is called the Kalman gain and the corresponding observer is indicated as the Kalman-Bucy filter: a filter producing the estimate $\hat{x}$ in a least squares sense. In steady state it is equivalent with the Wiener filter.

6.2.1 Proof of Kalman filter

For the proof it is useful to derive first three lemmas for the following general stochastic system:

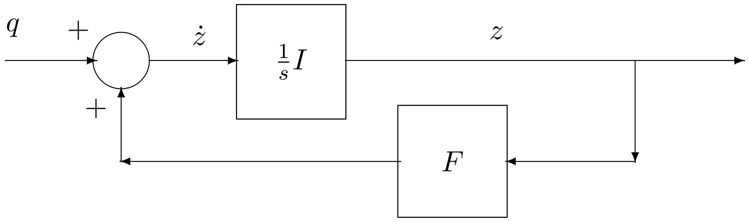


Figure 6.5: General stochastic system.

$$\begin{array}{ccc}
\dot{z}(t) = Fz(t) + q(t) & E\{q(t)\} = 0 & E\{z(t)\} = \bar{z}(t) \\
E\{(z(t) - \bar{z}(t))(z(t) - \bar{z}(t))^T\} = R_z(t) & E\{q(t)q^T(t + \tau)\} = R_q\delta(\tau) & E\{z(0)q^T(0)\} = 0
\end{array} \tag{6.34}$$

In words it simply says that the state vector z is disturbed by white noise disturbance q, having zero mean and constant variance R_q. The state z is characterised by its time-dependent mean $\bar{z}(t)$ and time-dependent variance $R_z(t)$. The following three properties hold, to be used as lemmas:

- **lemma 1**

$$E\{z^T(t)z(t)\} = \bar{z}^T(t)\bar{z}(t) + \text{trace}\,(R_z(t)) =' bias' +' variance' \tag{6.35}$$

Ergo, if we want to minimise the square of z we have to minimise both the mean $\bar{z}$ and the variance R_z.

- **lemma 2**

$$\bar{z}(t) = H_z(t)\bar{z}(0) \tag{6.36}$$

where the transition matrix $H_z(t)$ represents the impulse response from q to z thus:

$$H_z(t) = \mathcal{L}^{-1}\{(sI - F)^{-1}\} \tag{6.37}$$

The mean of z behaves as a deterministic signal would.

- **lemma 3**

$$\dot{R}_z(t) = FR_z(t) + R_z(t)F^T + R_q \tag{6.38}$$

The variance of z obeys a kind of Riccati equation without the quadratic term. Note that the variance depends linearly on the state matrix F as does the mean. For symmetry in matrices (variance!) we get the sum of two terms one of which is transposed. In steady state ($\dot{R}_z = 0$), this equation represents the discussed Lyapunov equation!

The proofs of the lemmas follow next. They are straightforward and can be skipped by lazy believers.

- **proof of lemma 1:** For each time t (argument t has been skipped) the following holds:

$$R_z = E\{(z - \bar{z})(z - \bar{z})^T\} = E\{zz^T - z\bar{z}^T - \bar{z}z^T + \bar{z}\bar{z}^T\} = E\{zz^T\} - \bar{z}\bar{z}^T \qquad \Rightarrow \tag{6.39}$$

So the correlation function $\Psi(t)$ satisfies:

$$\Psi(t) \overset{def}{=} E\{z(t)z^T(t)\} = \bar{z}(t)\bar{z}^T(t) + R_z \qquad \Rightarrow \tag{6.40}$$

$$\text{trace}\{E\{zz^T\}\} = E\{z^Tz\} = \text{trace}\{\bar{z}\bar{z}^T\} + \text{trace}\{R_z\} = \bar{z}^T\bar{z} + \text{trace}\{R_z\} \tag{6.41}$$

- **proof of lemma 2:**

$$z(t) = H_z(t)z(0) + \int_0^t H_z(t-\tau)q(\tau)d\tau \tag{6.42}$$

$$\bar{z}(t) = E\{z(t)\} = H_z(t)E\{z(0)\} + \int_0^t H_z(t-\tau)E\{q(\tau)\}d\tau = H_z(t)\bar{z}(0) \tag{6.43}$$

- **proof of lemma 3:** The correlation function $\Psi(t)$, as defined in equation 6.40 and by substituting equation 6.42, can be written as follows:

$$\begin{gathered}\Psi(t) = E\{z(t)z^T(t)\} = H_z(t)E\{z(0)z^T(0)\}H_z^T(t) + \\ \int_0^t\int_0^t H_z(t-\tau_1)E\{q(\tau_1)q^T(\tau_2)\}H_z^T(t-\tau_2)d\tau_1 d\tau_2\end{gathered} \tag{6.44}$$

where we have used the fact that $z(0)$ and $q(t)$ are uncorrelated. The variance of q can be used as we know that $E\{q(\tau_1)q^T(\tau_2)\} = R_q\delta(\tau_1 - \tau_2)$.

$$\Psi(t) = H_z(t)\Psi(0)H_z^T(t) + \int_0^t H_z(t-\tau_1)R_qH_z^T(t-\tau_1)d\tau_1 \tag{6.45}$$

Note that, like in equation 6.40, we have a bias term due to $\bar{z}(t)$ and a variance term. Consequently, as the first term indeed equals $E\{\bar{z}(t)\bar{z}^T(t)\}$, the second term is the variance $R_z(t)$. In fact above equation is the integral form of the differerential equation we want to proof. By differentiation we get:

$$\begin{array}{c}\dot{\Psi}(t) = \dot{H}_z(t)\Psi(0)H_z^T(t) + H_z(t)\Psi(0)\dot{H}_z^T(t)\\ +H_z(t-\tau_1)R_qH_z^T(t-\tau_1)|_{\tau_1=t}\\ +\int_0^t \dot{H}_z(t-\tau_1)R_qH_z^T(t-\tau_1)d\tau_1\\ +\int_0^t H_z(t-\tau_1)R_q\dot{H}_z^T(t-\tau_1)d\tau_1\end{array} \tag{6.46}$$

The derivative of the transition matrix $H_z(t)$ can be obtained by noting that $H_z(t)$ satisfies the state equation (like a deterministic state and the mean $\bar{z}(t)$ according to lemma 1):

$$\begin{array}{c}\mathcal{L}\{\dot{H}_z(t)\} = sH_z(s) - H_z(0) = sH_z(s) - I =\\ sI(sI-F)^{-1} - (sI-F)(sI-F)^{-1} = F(sI-F)^{-1} = FH_z(s) = \mathcal{L}\{FH_z(t)\}\end{array} \tag{6.47}$$

Substitution of $\dot{H}_z(t) = FH_z(t)$ and $H_z(0) = I$ yields:

$$\begin{array}{c}\dot{\Psi}(t) = FH_z(t)\Psi(0)H_z^T(t) + H_z(t)\Psi(0)H_z^T(t)F^T + R_q\\ +\int_0^t FH_z(t-\tau_1)R_qH_z^T(t-\tau_1)d\tau_1\\ +\int_0^t H_z(t-\tau_1)R_qH_z^T(t-\tau_1)F^Td\tau_1\end{array} \tag{6.48}$$

By rearranging we obtain:

$$\begin{array}{c}\dot{\Psi}(t) = F\{H_z(t)\Psi(0)H_z^T(t) + \int_0^t H_z(t-\tau_1)R_qH_z^T(t-\tau_1)d\tau_1\}\\ +\{H_z(t)\Psi(0)H_z^T(t) + \int_0^t H_z(t-\tau_1)R_qH_z^T(t-\tau_1)d\tau_1\}F^T + R_q\end{array} \tag{6.49}$$

and we finally obtain:

$$\dot{\Psi}(t) = F\Psi(t) + \Psi(t)F^T + R_q \tag{6.50}$$

So the correlation function $\Psi(t)$ obeys the equation of lemma 3. Note that this correlation function is developed for each moment t and that the time shift (τ), which is usually its argument, is zero. By using lemma 1, in particular equation 6.40, we can easily show that the lemma also holds for the variance $R_z(t)$:

$$\Psi = R_z + \bar{z}\bar{z}^T \;\Rightarrow\; \dot{\Psi} = \dot{R}_z + \dot{\bar{z}}\bar{z}^T + \bar{z}\dot{\bar{z}}^T \tag{6.51}$$

and from lemma 2 we derive:

$$\dot{\bar{z}}(t) = \dot{H}_z(t)\bar{z}(0) = FH_z(t)\bar{z}(0) = F\bar{z}(t) \tag{6.52}$$

Substitution yields:

$$\dot{R}_z + F\bar{z}\bar{z}^T + \bar{z}\bar{z}^TF^T = FR_z + F\bar{z}\bar{z}^T + R_zF^T + \bar{z}\bar{z}^TF^T + R_q \tag{6.53}$$

The equal bias terms on both sides can be skipped so that the lemma 3 has been proved.

End of lemma proofs.

Now we are in the position to apply the lemmas to the equation:

$$\dot{\tilde{x}}(t) = (A - KC)\tilde{x}(t) + (v(t) - Kw(t)) \tag{6.54}$$

Consequently we have:

$$\begin{array}{lcr} z(t) = \tilde{x}(t) & \textit{indeed} & E\{q(t)\} = 0 \; \textit{because}: \\ F = A - KC & & E\{v(t)\} = 0 \\ q(t) = v(t) - Kw(t) & & E\{w(t)\} = 0 \end{array} \tag{6.55}$$

and in particular:

$$\begin{aligned} R_q = E\{qq^T\} &= E\{vv^T - vw^TK^T - Kwv^T + Kww^TK^T\} \\ &\Rightarrow R_q = R_v + KR_wK^T \end{aligned} \tag{6.56}$$

because R_{vw}=0.

We have to minimise $E\{\tilde{x}(t)^T\tilde{x}(t)\}$ and from lemma 1:

$$E\{\tilde{x}(t)^T\tilde{x}(t)\} = \bar{\tilde{x}}^T(t)\bar{\tilde{x}}(t) + \text{trace}(R_{\tilde{x}}) \tag{6.57}$$

Minimisation thus boils down to minimisation of the bias and of the variance term:

- **minimisation of bias:** The (first) bias term is obviously minimal when $\bar{\tilde{x}}(t) = 0$. This can obviously be achieved by effecting $\bar{\tilde{x}}(0) = 0$ since by lemma 2 $\bar{\tilde{x}}(t)$ obeys the homogeneous differential equation. We can easily fulfil this condition:

$$\bar{\tilde{x}}(0) = E\{\tilde{x}(0)\} = E\{x(0)\} - \hat{x}(0) = 0 \tag{6.58}$$

by the trivial choice:

$$\hat{x}(0) = E\{x(0)\} = \bar{x}(0) = x_0 \tag{6.59}$$

- **minimisation of the variance:**

From lemma 3 we have by substitution:

$$\dot{R}_{\tilde{x}} = (A - KC)R_{\tilde{x}} + R_{\tilde{x}}(A - KC)^T + R_v + KR_wK^T \tag{6.60}$$

From this equation we can compute the variance of the misfit in the state estimation at any moment:

$$R_{\tilde{x}}(t) = E\{(x(t) - \hat{x}(t))(x(t) - \hat{x}(t))^T\} \tag{6.61}$$

because we have eliminated the bias by the choice $\hat{x}(0) = x_0$. We may integrate equation 6.60 from $t = 0$ on because we know:

$$R_{\tilde{x}}(0)\} = E\{(x(0) - \bar{x}(0))(x(0) - \bar{x}(0))^T\} = E\{(x(0) - x_0)(x(0) - x_0\} = P_0 \tag{6.62}$$

By convention from literature, we rename $R_{\tilde{x}}(t)$ by $P(t)$, not to be confused with the "Control"-P of the previous chapter! As a consequence equation 6.60 can be rewritten with some rearrangements as:

$$\dot{P} = AP + PA^T + R_v + KR_wK^T - KCP - PC^TK^T \tag{6.63}$$

The dependence of K can be expressed as a nonnegative, quadratic term:

$$\dot{P} = AP + PA^T + R_v - PC^TR_w^{-1}CP + (K - PC^TR_w^{-1})R_w(K - PC^TR_w^{-1})^T \tag{6.64}$$

Indeed, the last term is nonnegative, because R_w is positive definite by definition (no output can be measured without measurement noise). As we have already eliminated the bias error we have to minimise:

$$E\{\tilde{x}^T(t)\tilde{x}(t)\} = \text{trace}(\tilde{x}(t)\tilde{x}^T(t) = \text{trace}(P(t)) \tag{6.65}$$

At $t = 0$ we are confronted with given initial misfit P_0 by an error in the initial state by guessing it as x_0. Afterwards the variance is increased according to equation 6.64. We can take the trace of all terms:

$$\begin{array}{c} \text{trace}(\dot{P}(t)) = \frac{d\,\text{trace}(P)}{dt} = \\ \text{trace}(AP + PA^T + R_v - PC^TR_w^{-1}CP) + \\ \text{trace}((K - PC^TR_w^{-1})R_w(K - PC^TR_w^{-1})^T) \end{array} \tag{6.66}$$

The first term at the right hand side represents the increase of P ($\dot{P}$ at left hand side) which is inevitable and due to both v and w. The increase (positive definite!) due to the second, right sided term can be annihilated though by choosing the correct Kalman gain :

$$K = PC^TR_w^{-1} \tag{6.67}$$

In that case the variance of the state error is governed by:

$$\dot{P} = AP + PA^T + R_v - PC^TR_w^{-1}CP \tag{6.68}$$

with initial condition: $P(0) = P_0$.

This ends the proof of the Kalman filter.

Summary: Criterion to be minimised is:

$$\boxed{\min_K\{\tilde{x}(t)^T\tilde{x}(t)\}}$$

Solution: Take as initial estimated state:

$$\boxed{\hat{x}(0) = E\{x(0)\} = x_0}$$

The Kalman gain is given by:

$$\boxed{K = P(t)C^TR_w^{-1}}$$

where:

$$\boxed{P(t) = E\{\tilde{x}(t)\tilde{x}(t)^T\}}$$

is the minimal covariance and obtained from the Riccati equation:

$$\boxed{\dot{P} = AP + PA^T + R_v - PC^TR_w^{-1}CP}$$

with initial condition:

$$\boxed{P(0) = E\{(x(0) - x_0)(x(0) - x_0)^T\} = P_0}$$

Remarks:

- In steady state the effect of a misfit in the initial estimate $\hat{x}(0)$ has died out. The expectation of the misfit $E\{\tilde{x}(t)\}$ is zero and the variance $P(t) = E\{\tilde{x}(t)\tilde{x}(t)^T\} = \bar{P}$ is constant and due to the stationary noises $v(t)$ and $w(t)$. So the steady state solution is simply given by the algebraic Riccati equation (ARE):

  $$\boxed{0 = A\bar{P} + \bar{P}A^T + R_v - \bar{P}C^T R_w^{-1} C\bar{P} \qquad\qquad K = \bar{P}C^T R_w^{-1}}$$

  The steady state covariance $\bar{P}$ is independent of the initial covariance P_0. This means that for *any* symmetric and nonnegative definite P_0 the solution of the ARE will converge to the same steady state $\bar{P}$. As P is a covariance, it is both symmetric and nonnegative definite. It can be proved that the symmetric, nonnegative solution of the ARE is unique. Furthermore the optimum filter will be stable for this $\bar{P}$ as it produces a finite covariance $\bar{P}$ for its state $\tilde{x}(t)$. Consequently $A - KC$ has stable eigenvalues/poles. This is quite important, because it tells us that a possible mismatch in $\tilde{x}(t) = x(t) - \hat{x}(t)$ at any time t will allways die out. Also the steady state optimum filter yields unbiased estimates $\hat{x}(t)$ even in the case of erroneous initial estimates.

- The state matrices A, B, C and the covariance matrices R_v and R_w may be time dependent. The optimal Kalman filter is still found according to the given general Riccati equation (RE). Certainly the Kalman gain $K(t)$ will be time dependent.

- In the derivation of the optimal observer we have inverted the covariance R_w of the measurement noise and therefore supposed it to be nonsingular. From $K = PC^T R_w^{-1}$ we may expect that the Kalman gain becomes very large if R_w is nearly singular. The explanation is straightforward. In fact the measurement noise was the limiting factor for the increase of the Kalman gain K. Without measurement noise $w(t)$ the K can be made extremely large causing a very fast tracking of $x(t)$ by $\hat{x}(t)$. If the measurement $y(t)$ is less reliable, implying a large R_w, less confidence can be put into it so that K is decreased. Large and small are fuzzy qualifications though. The derived formulas tell us exactly how the trade off should be made exactly conform the particular noise amplitudes on the various outputs.

- The observer problem is completely dual to the control problem. Consequently for a first order SISO example we have the same effects as discussed in the optimal control section. So the poles governing the steady state observer are given by:

  $$s_{1,2} = \pm\sqrt{a^2 + c^2 \frac{r_v}{r_w}} \tag{6.69}$$

  Obviously a large ratio r_v/r_w is beneficial for a fast tracking of the state. The better the sensor is (low noise), the better the observer can function.

6.2.2 An example: detection of a constant amidst white noise.

An extremely stripped but nevertheless illustrative example is formed by the problem of detecting a *constant* value x amidst a measurement signal $y(t)$ disturbed by white noise $w(t)$. The state space description is:

$$\begin{aligned} \dot{x} &= 0 & x(0) &= x_0 = 2 \\ y(t) &= x + w(t) & r_w &= .01 \end{aligned} \tag{6.70}$$

The problem constants are given by:

$$a = 0 \quad b = 0 \quad c = 1 \quad r_v = 0 \tag{6.71}$$

Suppose that we know practically nothing about the unknown value $x(t) = x_0$, then it is wise to take a uniform probability distribution for x with mean $\bar{x}(0) = 0 = \hat{x}(0)$ and variance P_0 large, say 10. First the general observer structure in various perspectives has been drawn in Fig. 6.6.

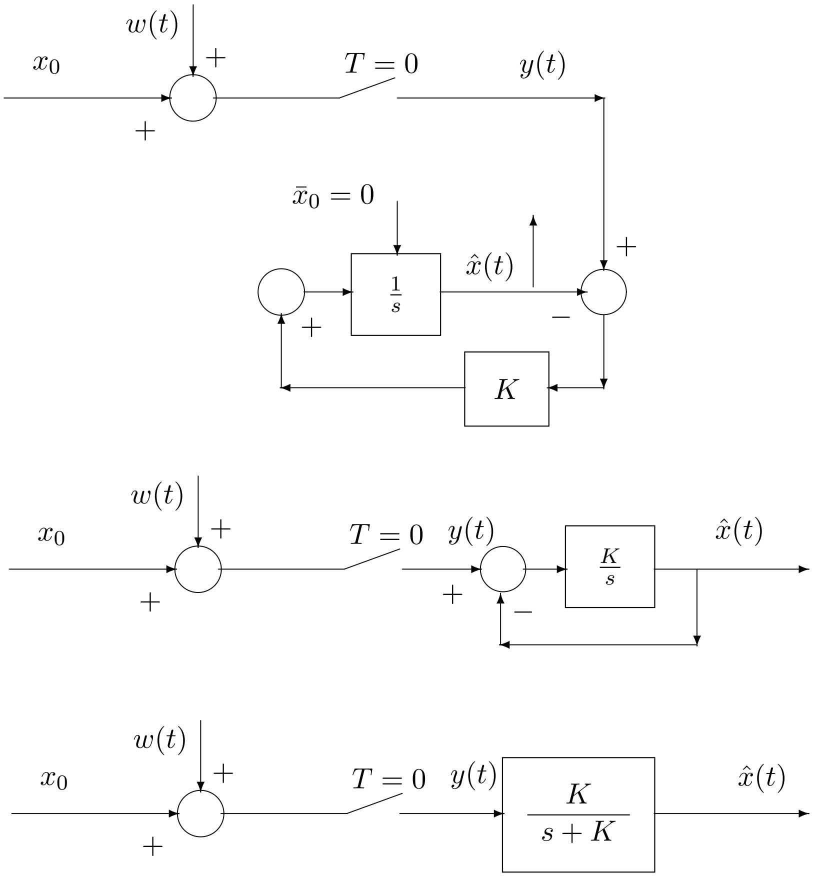


Figure 6.6: Asymptotic observer for constant amidst noise.

The upper block scheme is simply the stripped version of the general scheme as displayed in Fig. 6.4. Rearrangement shows, in the middle block scheme, how from time instant $T = 0$ on the measured signal is simply filtered to produce an estimate $\hat{x}(t)$. The lower block scheme finally shows how the filter is a simple first order low pass filter. The pole is given by $-K = (a - Kc)$! The higher the K, the broader the pass band of this low pass filter is. The sooner the misfit in $\hat{x}(0)$ will be corrected but also the more frequencies of the white measurement noise $w(t)$ will pass and disturb the steady state $\hat{x}(t)$. If we decrease K, the opposite will occur: slow recover from initial misfit but a substantially better steady state estimate because of a smaller pass band. This is illustrated in Fig. 6.7 where the estimates for K=.2, 1 and 10 are shown.

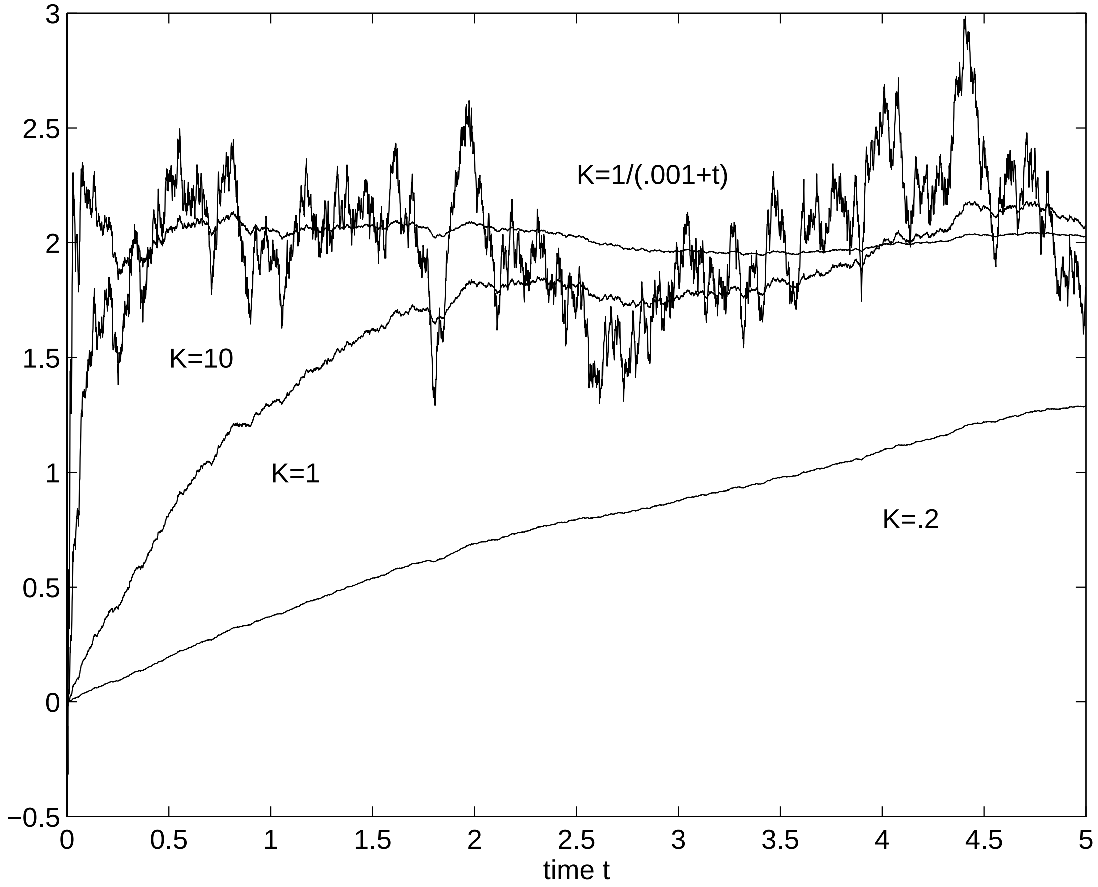


Figure 6.7: Estimates $\hat{x}(t)$ of $x_0 = 2$ for $K = .2, K = 1, K = 10$ and the optimal, time varying Kalman gain $K(t) = 1/(.001 + t)$.

It is clear that small K, e.g. 1, shows a slow convergence paired to a small final variance of the error. An increased K, e.g. 10, increases the convergence at the price of a high final variance. The optimal choice, being the Kalman gain as a solution of the Riccati equation, is time dependent and starts initially as very big to obtain a fast convergence, while it decreases later on effecting a small, even zero, final variance. The computation of above curves can be done in two ways. First of all we can simulate one of the block schemes of Fig. 6.6. An alternative is to start with Fig. 6.5 redrawn in the next Fig. 6.8 with appropriate states and parameters.

According to lemma 2 the mean of $\tilde{x}(t)$ will be given by:

$$\tilde{x}(t) = \tilde{x}(0)e^{(a-Kc)t} = 2e^{-Kt} \tag{6.72}$$

The larger K is, the sooner the average misfit fades away: cf. Fig. 6.7.

The variance $R_{\tilde{x}} = p$ is given by lemma 3 or explicitly by formula 6.63:

$$\begin{array}{ll} \dot{p} = K^2 r_w - 2cKp & p(0) = p_0, r_w = .01 \\ \Rightarrow \dot{p} = .01K^2 - 2Kp & p_0 = 10 \end{array} \tag{6.73}$$

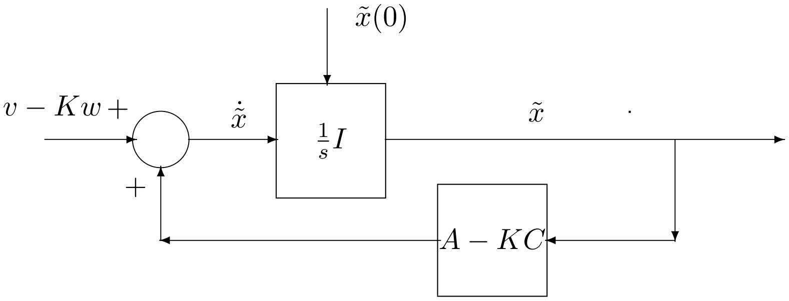


Figure 6.8: State error stochastic system.

The time dependent variance can be derived as:

$$\begin{array}{rcll}
\int \frac{dp}{2p - Kr_w} & = & \int -K dt & \\
\ln(2p - Kr_w) & = & -Kt + C_0 & \forall 2p - \frac{K}{r_w} \geq 0 \\
2p - Kr_w & = & e^{-Kt} e^{C_0} & \\
p & = & \frac{1}{2}(e^{C_0} e^{-Kt} + Kr_w) & \\
\frac{1}{2}(e^{C_0} + Kr_w) & = & p_0 & \\
e^{C_0} & = & 2p_0 - Kr_w & \\
\Rightarrow p & = & (p_0 - \frac{Kr_w}{2}) e^{-Kt} + \frac{Kr_w}{2} & \\
p & = & (10 - .005K) e^{-Kt} + .005K &
\end{array} \tag{6.74}$$

The first exponential term is the transient of erroneous estimate $\hat{x}(0)$ while the constant second term is the asymptotic, steady state term. Note that indeed for larger K the decrease of the variance has a time constant $\tau = 1/K$ and can be made very fast. Unfortunately the steady state variance is very large as it is given by $p = Kr_w/2$! For small K it is just the other way around.

The optimal Kalman gain will appear to be essentially *time variant* such that at the beginning the K is large in order to effect a sharp decrease of both initial misfit and initial variance. The K then smoothly decreases to guarantee a small steady state variance which is actually zero but only ultimately obtained for $t = \infty$. The algebra behind it develops as follows.

The filter is described by:

$$\begin{array}{l}
\hat{x}(t) = K(t)(y(t) - \hat{x}(t)) \quad \hat{x}(0) = \bar{x}_0 \\
K(t) = p(t)/r_w
\end{array} \tag{6.75}$$

while the Riccati equation has been reduced to :

$$\dot{p}(t) = -\frac{p^2(t)}{r_w} \qquad p(0) = p_0 \tag{6.76}$$

This simple nonlinear differential equation is easily solved by:

$$\frac{dp}{p^2} = -\frac{dt}{r_w} \qquad \Rightarrow \qquad -\frac{1}{p} = -\frac{t}{r_w} - \frac{1}{p_0} \tag{6.77}$$

which results in:

$$p(t) = \frac{r_w}{(r_w/p_0) + t} \qquad \Rightarrow \qquad K(t) = \frac{1}{(r_w/p_0) + t} \tag{6.78}$$

Note that both the variance $p(t)$ and the Kalman gain $K(t)$ show the same decreasing behaviour. This behaviour is exactly the optimal Kalman gain that effects the fast initial decrease of misfit and the minimisation of the final variance as displayed in Fig.6.7. For $t \to \infty$: $p(t) \to 0$, so that we are dealing with an asymptotic efficient estimator.

As we have put

$$\hat{x}(0) = \bar{x}(0) = 0 \tag{6.79}$$

(no a priori information) and

$$p_0 = \infty \tag{6.80}$$

or at least very large, we obtain by substitution into 6.75:

$$(\frac{r_w}{p_0} + t)\dot{\hat{x}}(t) + \hat{x}(t) = y(t) \tag{6.81}$$

This nonlinear differential equation is exactly the describing equation of the 'Maximum Likelihood Estimator (MLE)'. Even if one is not familiar with these estimation techniques, the following estimate (being the MLE) will look very appealing: Suppose that one has k samples $y(i)$ available. The average is then:

$$\hat{x}(k) = \frac{1}{k}\sum_{i=1}^{k} y(i) \tag{6.82}$$

The continuous time equivalent is trivially:

$$\hat{x}(t) = \frac{1}{t}\int_0^t y(\tau)d\tau \tag{6.83}$$

Neglecting the effects at $t = 0$ for a moment, differentiation yields:

$$\dot{\hat{x}}(t) = \frac{y(t)}{t} - \frac{1}{t^2}\int_0^t y(\tau)d\tau = \frac{y(t)}{t} - \frac{\hat{x}(t)}{t} \tag{6.84}$$

which leads to:

$$t\dot{\hat{x}}(t) + \hat{x}(t) = y(t) \tag{6.85}$$

which is exactly the differential equation 6.81 describing the Kalman filter for $p_0 \to \infty$.

6.2.3 More about state disturbance and measurement noise.

Till now the system definitions for observers was quite abstract, in particular with respect to the state disturbance v and measurement noise w, which were both taken as white noise sources. Practice will not directly provide such a scheme, but we can show how the modeling can be adapted. Let us take the irrigation system of the previous chapter as an example to discuss the characteristics of state disturbance and measurement noise in practice. Below, the irrigation problem is sketched again in Fig. 6.9

Note that we have added measurement noise f on the water level sensor being filtered white noise. Note also that the noise filter is biproper. This is necessary because we have to have all measurements to be disturbed. If we had a strictly proper filter the measurement at frequency $\omega = \infty$ would be undisturbed. For that frequency, how unrealistically high it may be, the measurement would be infinitely reliable and the R_w would be zero. This is not allowed; think of the need for R_w^{-1}!

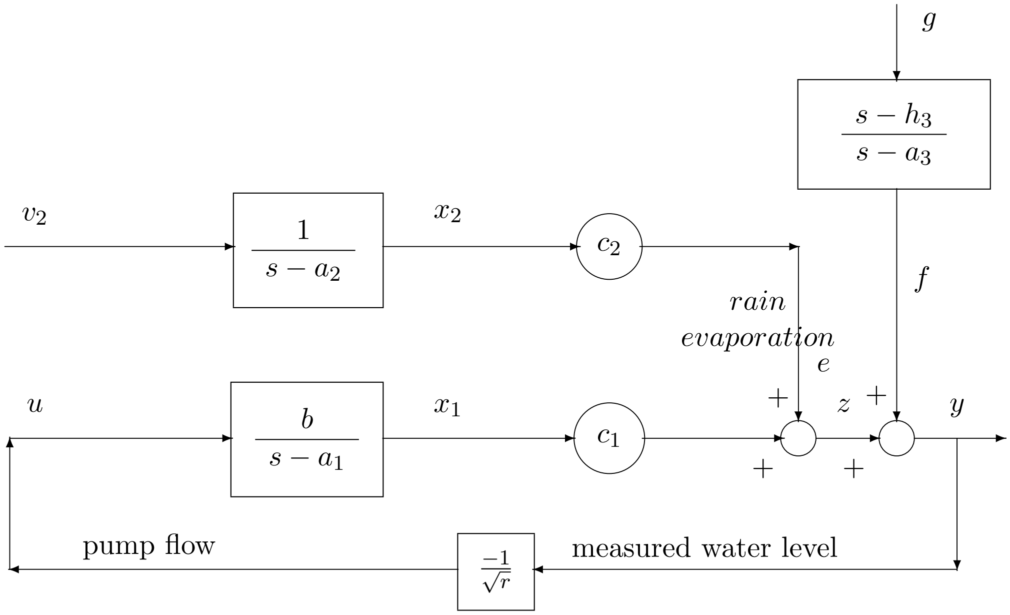


Figure 6.9: A simple irrigation model with (colored) measurement noise.

The real state disturbance is e, but its character is filtered white noise. Therefor, we added an extra state x_2 for the representation of the coloring of the source disturbance v_2. This trick can be used for the coloring of the measurement noise as well:

$$\frac{s-h_3}{s-a_3} = \frac{s-a_3+a_3-h_3}{s-a_3} = 1 + \frac{a_3-h_3}{s-a_3} \tag{6.86}$$

so the measurement noise filter in state space can be written as:

$$\begin{array}{rcrcl} \dot{x}_3 & = & a_3 x_3 & + & (a_3-h_3)g \\ f & = & x_3 & + & g \end{array} \tag{6.87}$$

where:

$$E\{g(t)\} = 0 \qquad\qquad E\{g(t)g(t+\tau)\} = \sigma_g^2\delta(\tau) \tag{6.88}$$

Merging the process and measurement state spaces we get:

$$\begin{pmatrix} \dot{x_1} \\ \dot{x_2} \\ \dot{x_3} \end{pmatrix} = \begin{pmatrix} a_1 & 0 & 0 \\ 0 & a_2 & 0 \\ 0 & 0 & a_3 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} + \begin{pmatrix} b \\ 0 \\ 0 \end{pmatrix} u + \begin{pmatrix} 0 \\ v_2 \\ (a_3-h_3)g \end{pmatrix} \tag{6.89}$$

$$y = \begin{pmatrix} c_1 & c_2 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} + g \tag{6.90}$$

Consequently the familiar covariance matrices become:

$$R_v = \begin{pmatrix} 0 & 0 & 0 \\ 0 & \sigma_v^2 & 0 \\ 0 & 0 & (a_3-h_3)^2\sigma_g^2 \end{pmatrix} \qquad\qquad R_w = \sigma_g^2 \tag{6.91}$$

If we still want to minimise the actual height, previously called y now renamed as:

$$z = \begin{pmatrix} c_1 & c_2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \tag{6.92}$$

the weighting matrix of the states in the control design should be:

$$Q = \begin{pmatrix} c_1^2 & c_1 c_2 & 0 \\ c_1 c_2 & c_2^2 & 0 \\ 0 & 0 & 0 \end{pmatrix} \tag{6.93}$$

One problem is created, though. By creating a state related to the measurements we have obtained a cross correlation between the state disturbance v and the measurement noise w:

$$R_{vw}\delta(\tau) = E\{v(t)w^T(t+\tau)\} = E\{\begin{pmatrix} 0 \\ v_2 \\ (a_3 - h_3)g \end{pmatrix} g\} = \begin{pmatrix} 0 \\ 0 \\ (a_3 - h_3)^2\sigma_g^2 \end{pmatrix} \delta(\tau) \tag{6.94}$$

In the derivation and proof of the Riccati equation for observers we assumed that $R_{vw} = 0$ so that we have to correct for $R_{vw} \neq 0$. It will appear that the Riccati equation and the expression for optimal Kalman-gain can easily be adapted as follows.

The crucial equation for state error dynamics and its general equivalent were given by:

$$\begin{array}{ccccccc} \dot{\tilde{x}} & = & (A - KC) & \tilde{x} & + & v - Kw \\ \dot{z} & = & F & z & + & q \end{array} \tag{6.95}$$

The variance of the noise $q = v - Kw$ is computed as:

$$E\{qq^T\} = E\{vv^T + Kww^TK^T - vw^TK^T - Kwv^T\} = \tag{6.96}$$

$$R_q = R_v + KR_wK^T - R_{vw}K^T - KR_{vw}^T \tag{6.97}$$

The differential equation describing the variance then turns into:

$$\dot{P} = (A - KC)P + P(A - KC)^T + R_v + KR_wK^T - R_{vw}K^T - KR_{vw}^T \tag{6.98}$$

Again we have to combine the terms in K as an explicit quadratic form which yields:

$$\begin{array}{rcc} \dot{P} & = & (A - R_{vw}R_w^{-1}C)P + P(A - R_{vw}R_w^{-1}C)^T + \\ & & R_v - R_{vw}R_w^{-1}R_{vw}^T - PC^TR_w^{-1}CP + \\ & & +(PC^T + R_{vw} - KR_w)R_w^{-1}(PC^T + R_{vw} - KR_w)^T \end{array} \tag{6.99}$$

As before, we minimise the increase in the eigenvalues of P by putting the last term equal to zero so that:

$$K = (PC^T + R_{vw})R_w^{-1} \tag{6.100}$$

and the adapted Riccati equation is:

$$\begin{array}{rcc} \dot{P} & = & (A - R_{vw}R_w^{-1}C)P + P(A - R_{vw}R_w^{-1}C)^T + \\ & & R_v - R_{vw}R_w^{-1}R_{vw}^T - PC^TR_w^{-1}CP \end{array} \tag{6.101}$$

Consequently the solution is analogous to the solution for $R_{vw} = 0$ by taking the following adaptations:

$$\begin{array}{lccl} A - R_{vw}R_w^{-1}C & \mathit{instead} & \mathit{of} & A \\ R_v - R_{vw}R_w^{-1}R_{vw}^T & \mathit{instead} & \mathit{of} & R_v \\ K = (PC^T + R_{vw})R_w^{-1} & \mathit{instead} & \mathit{of} & K = PC^TR_w^{-1} \end{array} \tag{6.102}$$

It is simple to see that for $R_{vw} = 0$ the equivalence is complete.

Chapter 7

Complete LQG-control

In section 5.4 we have derived an optimal state controller of the form:

$$u = -Lx + u^* \tag{7.1}$$

Where u^* is some exogenous input.

In chapter 6 we have obtained a state observer producing a state estimate $\hat{x}$. It is now time to combine the controller and the observer by simply putting:

$$u = -L\hat{x} + u^* \tag{7.2}$$

and study the consequences. In Fig. 7.1 we can see how the feedback loop is closed.

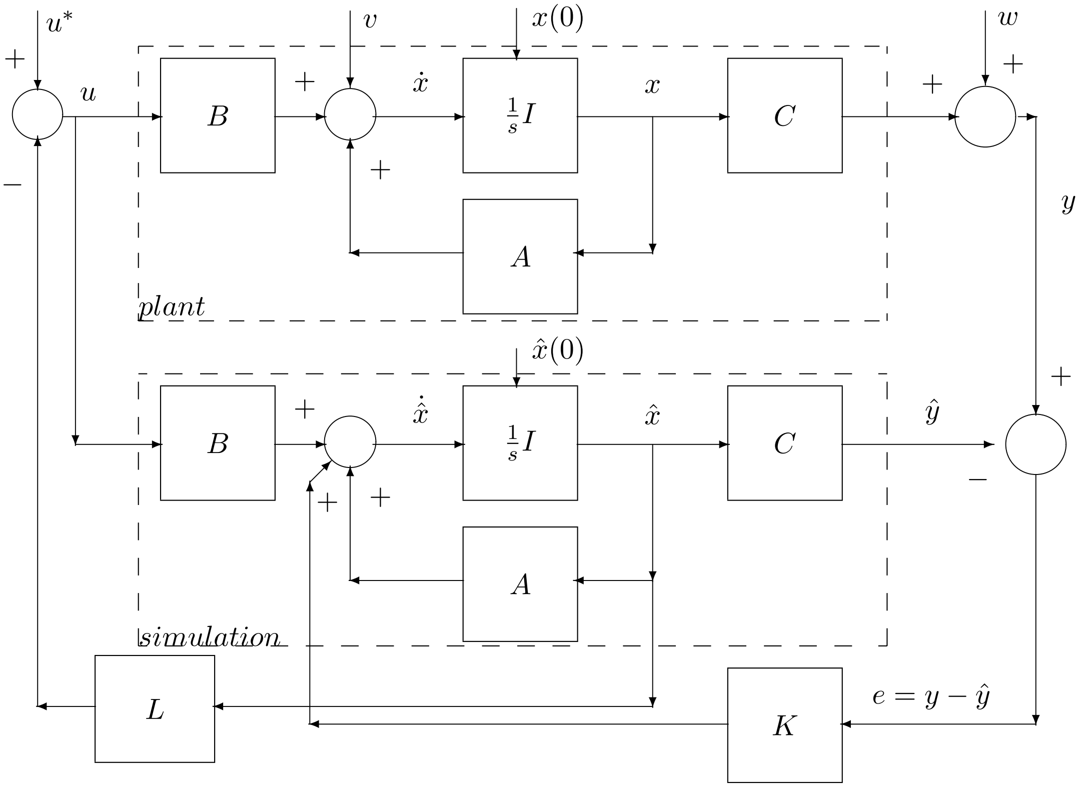


Figure 7.1: Block diagram of complete LQG-control.

Let us first analyse what such a structure effects for any stabilising pair L and K, not necessarily optimal but :

$$|sI - A + BL| = 0 \qquad \textit{and} \qquad |sI - A + KC| = 0 \tag{7.3}$$

yield stable poles. The governing set of equations is given by:

$$\dot{x} = Ax + Bu + v \tag{7.4}$$
$$y = Cx + w \tag{7.5}$$
$$u = -L\hat{x} + u^* \tag{7.6}$$
$$\dot{\hat{x}} = A\hat{x} + Bu + K(y - C\hat{x}) \tag{7.7}$$

Elimination of u and y by simple substitution yields:

$$\begin{pmatrix} \dot{x} \\ \dot{\hat{x}} \end{pmatrix} = \begin{pmatrix} A & -BL \\ KC & A - KC - BL \end{pmatrix} \begin{pmatrix} x \\ \hat{x} \end{pmatrix} + \begin{pmatrix} I & 0 \\ 0 & K \end{pmatrix} \begin{pmatrix} v \\ w \end{pmatrix} + \begin{pmatrix} B \\ B \end{pmatrix} u^* \tag{7.8}$$

Consequently, we can describe the closed loop system in the more compact form of Fig. 7.2 where we can clearly distinguish the controller $C(s)$ in the feedback loop. The second state equation reads as:

$$\dot{\hat{x}} = (A - KC - BL)\hat{x} + K(Cx + w) + Bu^* \Rightarrow \tag{7.9}$$
$$(sI - A + KC + BL)\hat{x} = Ky + Bu^* \Rightarrow \tag{7.10}$$
$$\hat{x} = (sI - A + KC + BL)^{-1}(Ky + Bu^*) \tag{7.11}$$

Because we had $u = -L\hat{x} + u^*$ the feedback controller is given by:

$$C(s) = L(sI - A + KC + BL)^{-1}K \tag{7.12}$$

Ergo, the poles of the controller are given by the eigenvalues of $A - KC - BL$. Note that these poles can be unstable. Some plants can only be stabilised by unstable controllers. In fact these plants have intermittently poles and zeros on the real, positive axis like the example in equation 5.33 in section 5.2.(The solution of this particular problem is straightforward and left to the reader.)

The instability of the controller can do no harm in the way we implemented the controller in Fig. 7.1. If we write:

$$u = -L(sI - A + KC + BL)^{-1}Ky + (I - L(sI - A + KC + BL)^{-1}B)u^* \tag{7.13}$$

and implement the controller accordingly. That is, if we would have fed the input u^* via the prefilter $I - L(sI - A + KC + BL)^{-1}B$ to the comparator from the feedback-loop, the complete system would be unstable. Consequently this is not allowed if the prefilter happens to be unstable, i.e. if $|sI - A + BL + KC|$ yields unstable poles.

Apart from this implementation problem it is not so important what the poles of the controller are. Of crucial importance is the effect of the controller, i.e. what are the poles of the closed loop system.

7.1 Preservation of controller and observer poles.

Above equations suggest that the poles of the closed loop system are complex functions of the controller coefficient L and the observer gain K. It is easy to show that this is not

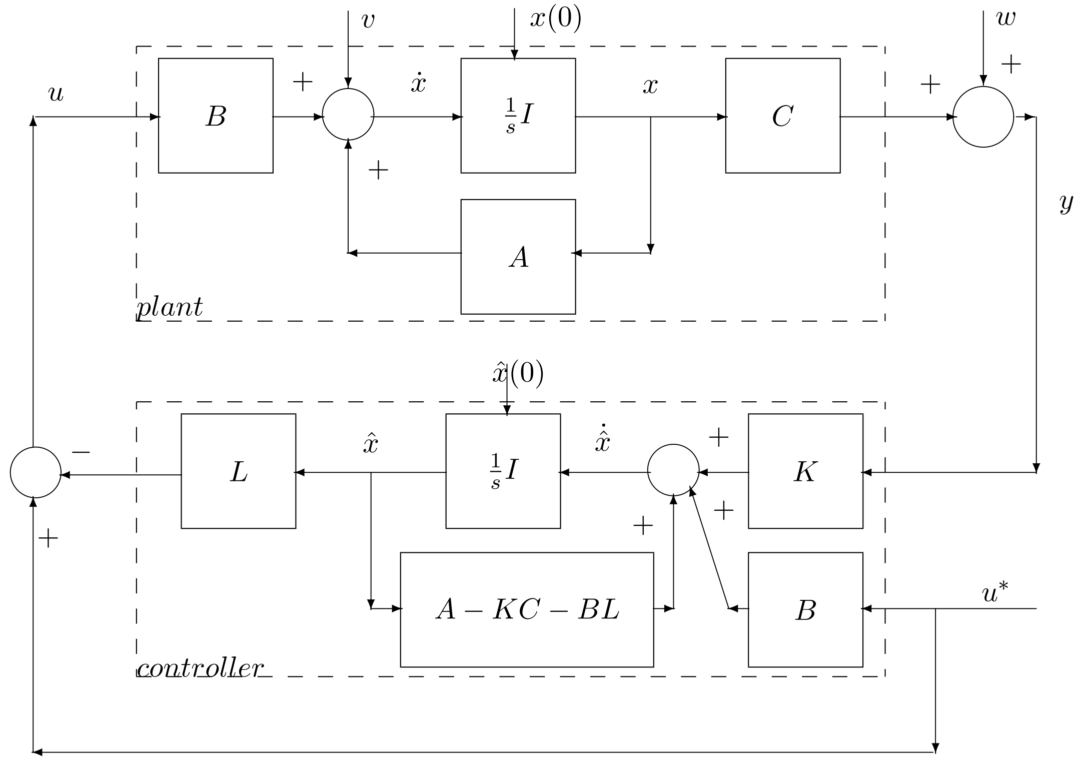


Figure 7.2: Feedback loop with plant $P(s)$ and controller $C(s)$.

the case by taking the state error $\tilde{x}$ as part of the state vector instead of the estimated state $\hat{x}$. If we subtract $\hat{x}$ from x in order to obtain the state error $\tilde{x}$ we simply have:

$$\dot{x} = Ax - BL\hat{x} + v + Bu^* = (A - BL)x + BL(x - \hat{x}) + v + Bu^* \tag{7.14}$$

$$\dot{\hat{x}} = (A - BL)\hat{x} + KC(x - \hat{x}) + Kw + Bu^* \Rightarrow \tag{7.15}$$

$$\dot{x} - \dot{\hat{x}} = (A - BL)(x - \hat{x}) + BL(x - \hat{x}) - KC(x - \hat{x}) + v - Kw \Rightarrow \tag{7.16}$$

$$\dot{\tilde{x}} = (A - KC)\tilde{x} + v - Kw \tag{7.17}$$

so the complete state description becomes:

$$\begin{pmatrix} \dot{x} \\ \dot{\tilde{x}} \end{pmatrix} = \begin{pmatrix} A - BL & BL \\ 0 & A - KC \end{pmatrix} \begin{pmatrix} x \\ \tilde{x} \end{pmatrix} + \begin{pmatrix} I & 0 \\ I & -K \end{pmatrix} \begin{pmatrix} v \\ w \end{pmatrix} + \begin{pmatrix} B \\ 0 \end{pmatrix} u^* \tag{7.18}$$

Because of the zero block in the state matrix we can obtain the poles from:

$$\det \begin{pmatrix} sI - A + BL & -BL \\ 0 & sI - A + KC \end{pmatrix} = \det(sI - A + BL)\det(sI - A + KC) = 0 \tag{7.19}$$

So it says that the poles of the closed loop system are exactly the poles obtained before in the separate state control problem and the state observer problem!! Above equations can be visualised in the next figure. Indeed, the state x depends on the external input u^* and the state noise v exactly as whether the real x was fed back, apart from the influence of the lower signal $\tilde{x}$. Surely we have fed back $\hat{x}$ instead of x and we can write this as:

$$u = -L\hat{x} + u^* = -Lx + Lx - L\hat{x} + u^* = -Lx + L\tilde{x} + u^* \tag{7.20}$$

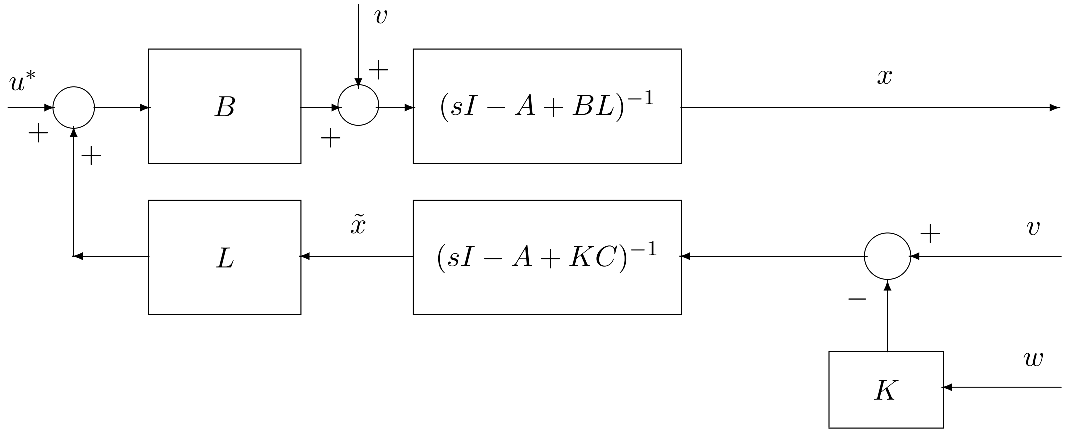


Figure 7.3: Controller and observer design poles exactly in closed loop system

The effect of the second term $L\tilde{x}$ is exactly represented by the lower part of Fig. 7.3. The question arises whether the optimal designed controller (L) and the optimal observer (K) remain optimal designs in this combined implementation. Surely, the poles remained the same but what about optimality? The so called **separation principle** says that indeed the optimality is preserved and poles take over.

7.2 Proof of the separation principle

We have proved in the previous section by deriving a state representation of the closed loop system in x and $\tilde{x}$ that indeed poles are preserved. For optimality we have to prove for the closed loop configuration that:

- **Optimal control problem:** $\min_L E\{x^TQx + u^TRu\}$ with $u = -L\hat{x}$
- **Optimal observer problem:** $\min_K E\{\tilde{x}^T\tilde{x}\}$ with $\dot{\hat{x}} = A\hat{x} + Bu + K(y - C\hat{x})$

yields the same L and K as for the originally separate problems. The easiest part is the observer. Because both the real plant and the observer have the same input u and because both have the same representation in A, B, C, the governing equation for the state error has the effect of the input completely annihilated as indicated in e.g. equation 7.17. Consequently for the observer problem it is irrelevant what the input u is so that also a feed back does not influence the optimality. The optimal K is not depending on L as reflected also in Fig. 7.3. This figure also reveals that optimality of L is not so obvious as L is the coefficient of the state error $\tilde{x}$. For that reason alone, L should be as small as possible. It will turn out in the sequel that this term $-L\tilde{x}$ is not influencing optimality because of its independent, white noise character.

By substitution of $u = -Lx$ in the original, stand alone controller problem we obtained:

$$\min_L E\{x^T(Q + L^TRL)x\} \tag{7.21}$$

$$\dot{x} = (A - BL)x + v \tag{7.22}$$

In the closed loop configuration we obtain by substitution of $u = -L\hat{x}$:

$$\min_L E\{x^TQx + \hat{x}^TL^TRL\hat{x}\} \tag{7.23}$$

$$\dot{\hat{x}} = (A - BL)\hat{x} + K(y - C\hat{x}) \tag{7.24}$$

From $\tilde{x} = x - \hat{x}$ we obtain $x = \tilde{x} + \hat{x}$ and substitution yields:

$$\min_L E\{\hat{x}^T(Q + L^T RL)\hat{x} + \tilde{x}^T Q\tilde{x} + 2\tilde{x}^T Q\hat{x}\} \tag{7.25}$$

$$\dot{\hat{x}} = (A - BL)\hat{x} + K(C\tilde{x} + w) \tag{7.26}$$

The term $E\{\tilde{x}^T Q\tilde{x}\}$ is independent of L and in fact minimised by K.

The term $E\{\tilde{x}^T Q\hat{x}\}$ is zero. The formal proof can be found in Kwakernaak [4][section 5.3.2, pages 391-393]. We will skip the algebra here and explain the effect in geometric terms. The minimisation $\min_K E\{\tilde{x}^T\tilde{x}\}$ concerns a quadratic criterion (Hilbert space) and is as such completely comparable to the minimal distance problem in the familiar Euclidean space. We thus search for the minimal distance $\tilde{x}(t)$ between $x(t)$ and the estimate $\hat{x}$, which is confined by the observer structure. The minimal distance can be viewed at as the radius of the smallest sphere with $x(t)$ at its center and containing a possible $\hat{x}(t)$. By the "orthogonality principle" in quadratic minimisation problems the minimal distance is then found by a projection as illustrated in Fig. 7.4. The "error" $\tilde{x}(t)$ is perpendicular to the optimising estimate $\hat{x}(t)$ as it is a projection onto the space of allowed $\hat{x}(t)$, represented in the figure by the full horizontal axis. Orthogonality in the actual space defined by the criterion means that:

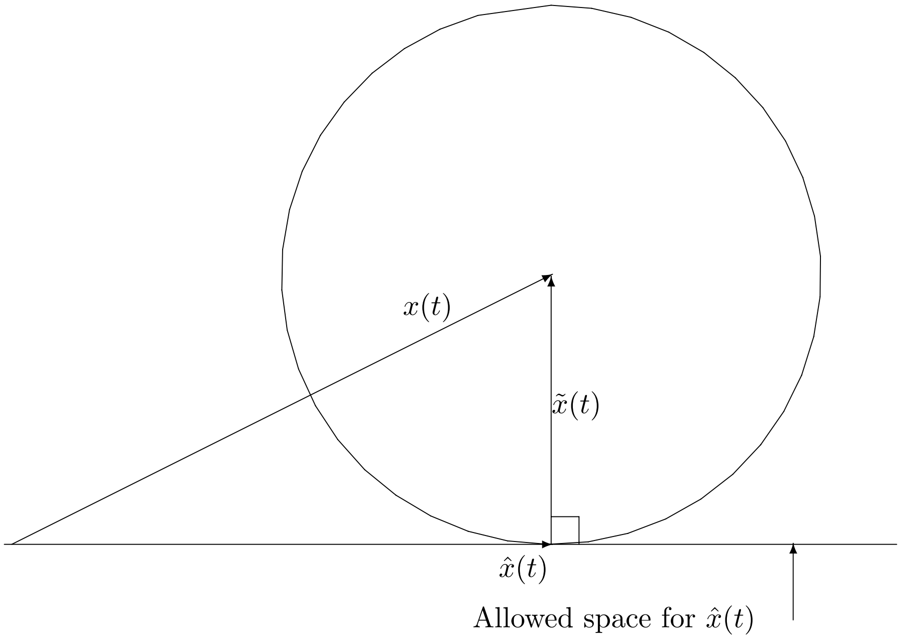


Figure 7.4: Illustration of the orthogonality principle in Hilbert space.

$$E\{\hat{x}\tilde{x}(t)^T\} = 0 \tag{7.27}$$

Then the actual term under study is zero because:

$$E\{\tilde{x}^T Q\hat{x}\} = E\{\text{trace}\{\tilde{x}^T Q\hat{x}\}\} = E\{\text{trace}\{\hat{x}\tilde{x}^T Q\}\} = \text{trace}\{E\{\hat{x}\tilde{x}^T\}Q\} = 0 \tag{7.28}$$

Finally it turns out that the state error influence, by means of the term $C\tilde{x}$, is a white noise term for optimal K. The formal proof can be found again in Kwakernaak [4][section 5.3.3, page 401 and section 4.3.6, pages 361-363] . The proof is again quite abstract and will be skipped here. The explanation is as follows. The observer produces at any moment an estimate $C\tilde{x}(t)$ of the actual output $y(t) = Cx(t) + w(t)$. The error $e(t) = y(t) - C\hat{x}(t) = C\tilde{x}(t) + w(t)$ should be a white noise sequence for optimal K because only in that case all information contained in $y(t)$ was optimally used: If it were not a white noise, something in $y(t)$ was not predicted by $C\hat{x}(t)$ but still dependent on previous outputs $y(t - \tau)$, so it could have been predicted. Then obviously the job was not well done. If, on the other hand, $e(t)$ is indeed a white noise sequence, it says that every new output $y(t)$ provides completely independent, new information for the observer on top of previous information in $y(t - \tau)$ already contained in the optimally tuned observer. The ”new” information $e(t) = y(t) - C\hat{x}(t)$ is therefore called the **innovation**. In the next chapter discussing the sampled or discrete time LQG-problem we will discuss this topic further. For the moment we conclude: $e(t) = y(t) - C\hat{x}(t)$ is white noise. So the minimisation turns into:

$$\min_L E\{\hat{x}^T (Q + L^T RL)\hat{x}\} \tag{7.29}$$

$$\dot{\hat{x}}(t) = (A - BL)\hat{x}(t) + Ke(t) \tag{7.30}$$

which is completely comparable with the original problem, be it that we deal with $\hat{x}$ instead of x and that the white noise term $v(t)$ has changed into the white noise term $e(t)$. Ergo, optimality of the controller is preserved under the condition that the state error $\tilde{x}$ is independent of the state estimate $\hat{x}$ and the output error $e(t)$ is white noise. This is true if K is the optimal Kalman gain.

This completes the ”proof” of the separation principle saying that controller gain L and Kalman gain K can be determined for the separate controller and observer problem and still preserve their optimality for the combined system. Also the stable poles of control and observer loops are preserved in the combined system.

7.3 Reference input, final error

The LQG-design was intended to decrease the disturbances on a system, as represented by the (white) state noise $v(t)$, as far as possible. The sensor by its measurement noise $w(t)$ and the actuator by its limited range have put bounds on the obtained performance. Apart from disturbance reduction a control aim is tracking. The question is whether the LQG-design, which yielded at least a stable closed loop system, can be used for tracking as well.

If we take the exogenous input u^* as the reference signal $r(t)$, we read from Fig. 7.3 that the output $y(t)$ depends on the reference $r = u^*$ according to:

$$y(s) = C(sI - A + BL)^{-1}Bu^*(s) \tag{7.31}$$

Indeed stable and as fast as the actuator allowed, i.e. with the closed loop controller poles. However, DC-gain will not be one so that the final step error won't be zero. For a SISO plant we could scale the input accordingly by taking:

$$u^* = (C(-A + BL)^{-1}B)^{-1}r \tag{7.32}$$

The final step error is then (theoretically) zero indeed, but the accuracy depends on the accurate inverse of the closed loop DC-gain as above scaling represents. Furthermore, for MIMO plants this approach only holds if there are at least as many inputs as there are outputs by taking pseudo-inverses.

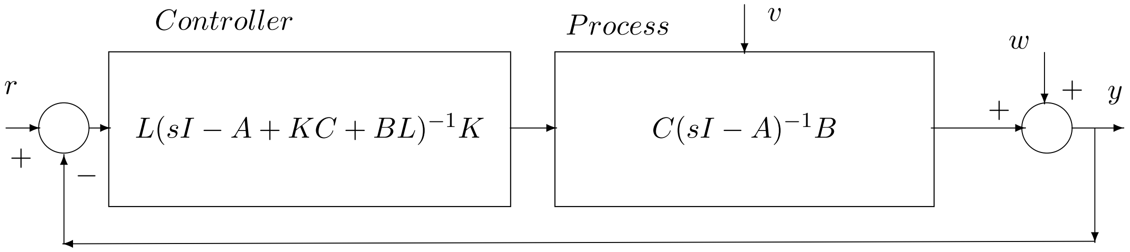


Figure 7.5: The standard tracking scheme

Another option is to compare the output $y(t)$ with the reference $r(t)$ before feeding it back as suggested in the next Fig.7.5.

In this option, which leads to the standard tracking configuration, we notice that the state x will depend on $w(t)$ and $-r(t)$ in exactly the same manner. Consequently the transfer from r to y can be read from Fig. 7.3 again by substituting $-r(t)$ for $w(t)$ so that:

$$y = C(sI - A + BL)^{-1}BL(sI - A + KC)^{-1}Kr \tag{7.33}$$

We notice that both sets of stable closed loop poles are involved and final step errors won't be zero.

Often, the final step error is forced to zero by artificially adding an integrator in the closed loop as illustrated in Fig. 7.6.

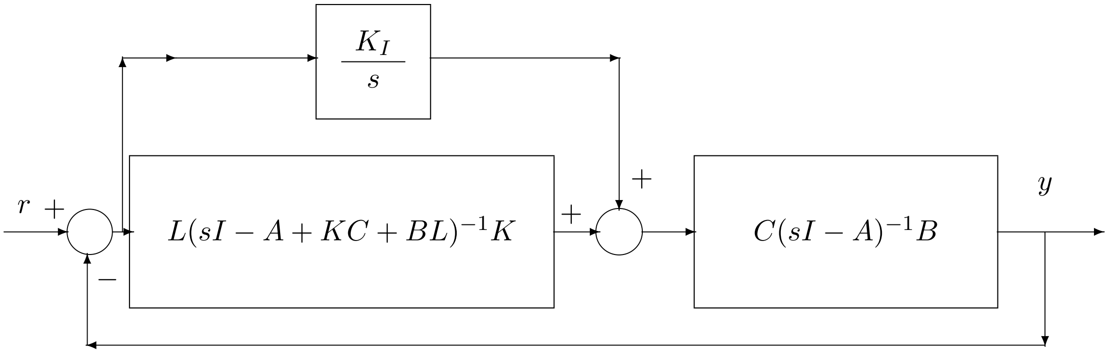


Figure 7.6: LQG with integrational feedback

The integration constant K_I should then be chosen small enough so that the integration hardly influences the frequency band where the LGQ-controller is active. If this is not conceivable, the LQG-controller can be (re)designed with the integration feedback actualised. Restructuring Fig. 7.6 then yields Fig. 7.7. Now P^* is the plant for which LQG-control should be designed.

Note that the reference r has been skipped and that w will also act as state disturbance on the state of the integration feedback.

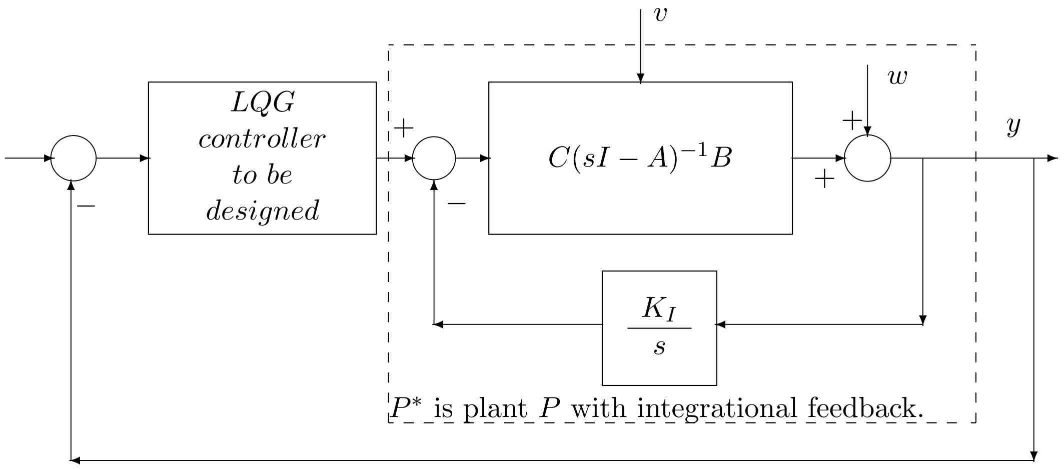

v
w
LQG controller to be designed
$+$
$-$
$C(sI - A)^{-1}B$
$+$
$+$
y
$-$
$\frac{K_I}{s}$
P^* is plant P with integrational feedback.

Figure 7.7: LQG with integrational feedback

7.4 Example: irrigation.

As an example the irrigation plant of section 5.4.5 is suitable. We may compare the full LQG-control with the LQR-control used in section 5.4.5, where we derived that for small enough r a simple proportional feedback with $-1/\sqrt{(r)}$ would suffice as illustrated in Fig. 7.8.

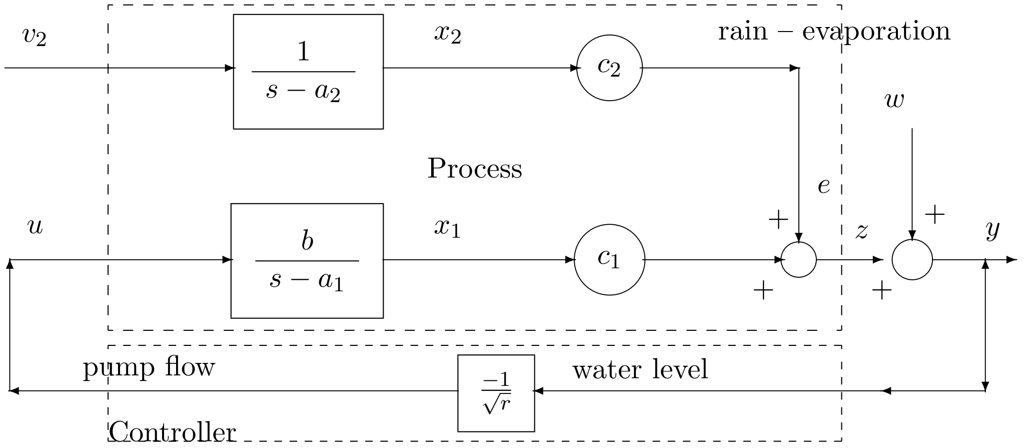


Figure 7.8: LQR control (without observer).

Note that we have to deal here with the measurement noise w.

The water from the pump is indistinguishable from the water from rain as soon as both are contained in the rice field. So x_1 cannot be measured independently from x_2. An observer is indispensable. Based upon the model, the known input u and the measured output we can reconstruct the states.

The real output to be minimised is called z and thus $z = c_1x_1 + c_2x_2 = x_1 + x_2$ so that the weights for the states remain the same, viz.:

$$Q = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$$

The same parameter values are taken for the parameters and in particular:

$r = .0001$, $R_v = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}$, $R_w = .01$, $R_{vw} = 0$

The LQG-feedback can simply be given by $L(sI - A + BL + KC)^{-1}K$ but it is better illustrated in the detailed state representation as displayed in Fig. 7.9.

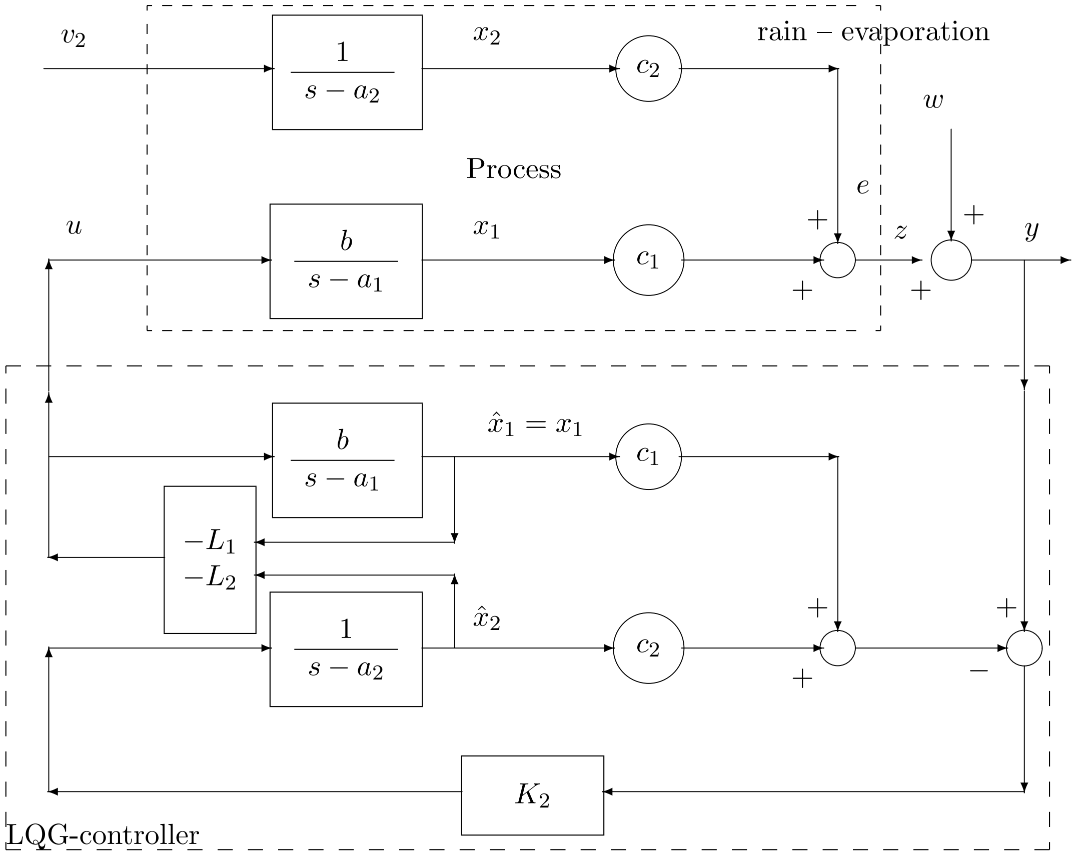


Figure 7.9: LQG control (with observer).

Note that only state x_2 gets a feedback from the output error $y - \hat{y}$. It thus appears that entry K_1 is zero. This follows from the observer Riccati equation and is a reflection of the fact that in the real process only state x_2 is disturbed (by v_2). This illustrates that the Kalman feed back tries in fact to imitate the real disturbance which has only a component v_2 on x_2.

The optimal L and K have been computed from the appropriate Riccati equations. In the next Fig. 7.10 the output signals z and the necessary control signals u are shown.

The following analysis can be made. The result for the complete LQG-control (right upper plot) is obviously better than the control without an observer (left upper plot). The last controller, a LQR-controller without observer, as proposed in section 2.4.5, fully ignores the measurement noise w. As a consequence much measurement noise is fed back which is responsible for the higher noise level in z. For the LQG-controlled system the observer with the optimal Kalman gain offers the best $\hat{x}$ (best in 2-norm) for feedback. That is: an optimal choice has been made in the trade off between fastest following of x and least effect of the measurement noise w. The observer certainly pays off in the final

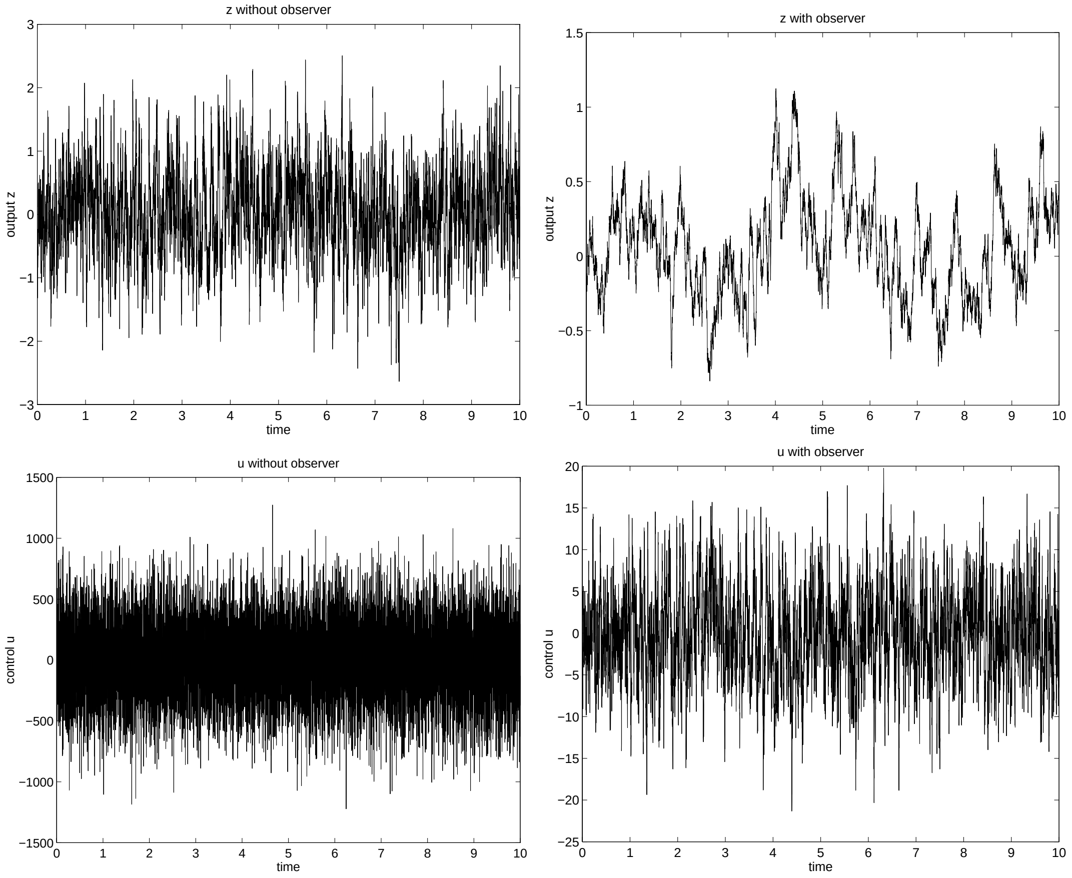


Figure 7.10: The controlled water levels z in the upper plots and the necessary control inputs u underneath. The left plots represent the LQR control from Fig. 7.8 without observer, while the right plots show the result for the full LQG-control of Fig. 7.9.

performance of the closed loop system.

But there is more to gain with the observer. In the lower plots we compare the actual control signals for both controllers. Astonishingly, for the worse control we need an amplitude of roughly twenty (sic!) times the better control. As the optimal control gain L was based on the given range of the actuator, which aligns with the LQG-controlled system, we may expect that a factor twenty will certainly saturate the actuator with all its disastrous effects. So in practice the results without an observer will be even worse. So we may safely conclude that an observer is indispensable for proper control.

Experiments for tracking a step with zero final error are left to the reader to test his understanding of the offered material and to test his ability to perform such a task in matlab/simulink.

Chapter 8

Time discrete controller and observer

8.1 Introduction

In the previous chapters we dealt, in considerable detail, with the linear controller and observer theory for continuous-time systems. In this chapter we give a condensed review of the same theory for discrete-time systems.

Since the theory of linear discrete-time systems very closely parallels the theory of linear continuous-time systems, many of the results are similar. For this reason the comments in the text are brief, except in those cases where the results for discrete-time systems deviate noticeably from the continuous-time situation. For the same reason many proofs are omitted. In order to observe the simularity between the continuous-time and discrete-time systems at a glance, we used the same symbols for corresponding quantities.

8.2 Structure of linear controller and observer.

In equivalence with continuous-time the discrete-time state equations are given by:

$$\begin{array}{ccccccc} x(k+1) & = & Ax(k) & + & Bu(k) & + & v(k) \\ y(k) & = & Cx(k) & + & w(k) & & \end{array} \tag{8.1}$$

where:

$$\begin{array}{rclrcl} E\{v(k)\} & = & 0 & E\{w(k)\} & = & 0 \\ E\{v(k)v^T(k+\tau)\} & = & R_v\delta(\tau) & E\{w(k)w^T(k+\tau)\} & = & R_w\delta(\tau) \\ E\{v(k)w^T(k+\tau) & = & R_{vw}\delta(\tau) & E\{x(0)\} & = & \bar{x}_0 \end{array} \tag{8.2}$$

and note that for discrete-time systems τ is integer and:

$$\delta(\tau) = \begin{cases} 1 & for \quad \tau = 0 \\ 0 & for \quad \tau \neq 0 \end{cases} \tag{8.3}$$

Furthermore, for the observer we need the following extra information. The initial state is supposed to be uncorrelated with the state disturbance $v(k)$ and the measurement noise $w(k)$. The variance of the initial states (about its expectation) is given by:

$$E\{(x(0) - \bar{x}_0)(x(0) - \bar{x}_0)^T\} = P_0 \tag{8.4}$$

For the ideal, linear, state control $u(k) = -Lx(k)$ we obtain:

$$x(k+1) = (A - BL)x(k) + v(k) \tag{8.5}$$

For the structure of the linear, state observer we obtain simular equations. Denoting the misfit in the state estimate again by:

$$\tilde{x}(k) = \hat{x}(k) - x(k) \tag{8.6}$$

while

$$\hat{x}(k+1) = A\hat{x}(k) + Bu(k) + K(y(k) - C\hat{x}(k)) \tag{8.7}$$

we obtain as before:

$$\tilde{x}(k+1) = (A - KC)\tilde{x}(k) + v(k) - Kw(k) \tag{8.8}$$

It is easy to verify that equations 8.5 and 8.8 completely correspond to those describing the continuous time situation and all conclusions drawn there can be applied here. By proper choice of L and K, desired pole placements can be accomplished. If we implement the realistic feedback $u(k) = -L\hat{x}(k)$, the **separation** principle applies, indicating that the closed loop system is governed by both the poles of the controller (from $A - BL$) and the poles of the observer (from $A - KC$). Similarly both the controller and the observer can be designed independently, which brings us to the **optimal** design methods of the next sections.

8.3 Optimal, linear, quadratic controller design.

The quadratic criterion is defined as:

$$J = \frac{1}{2}\Sigma_{k=0}^{N-1}\{x^T(k)Qx(k) + u^T(k)Ru(k)\} + \frac{1}{2}x^T(N)P_N x(N) \tag{8.9}$$

Ergo the Hamiltonian becomes:

$$H = \frac{1}{2}\{x^T(k)Qx(k) + u^T(k)Ru(k)\} + \lambda^T(k+1)\{Ax(k) + Bu(k)\} \tag{8.10}$$

Note that the end index of the summation is $N-1$ and the extra penalty at N weighted by P_N. The index of the Lagrange operator λ is $k+1$. One may choose k as well, which causes just a time shifted λ, but the resulting formulas become less elegantly symmetric.

The Euler Lagrange equations yield:

$$\lambda(k) = Qx(k) + A^T\lambda(k+1) \tag{8.11}$$

$$Ru(k) = -B^T\lambda(k+1) \tag{8.12}$$

By use of the process state equations and eliminating u we get:

$$\begin{pmatrix} x(k+1) \\ \lambda(k) \end{pmatrix} = \begin{pmatrix} A & -BR^{-1}B^T \\ Q & A^T \end{pmatrix} \begin{pmatrix} x(k) \\ \lambda(k+1) \end{pmatrix} \tag{8.13}$$

Let us compare this with the corresponding continuous-time equations:

$$\begin{pmatrix} \dot{x} \\ \dot{\lambda} \end{pmatrix} = \begin{pmatrix} A & -BR^{-1}B^T \\ -Q & -A^T \end{pmatrix} \begin{pmatrix} x \\ \lambda \end{pmatrix} \tag{8.14}$$

Notice that the minus signs for the continuous time equivalent are skipped now. These minus signs caused the poles to be mirrored with respect to the imaginary axis, the boundary between stable and unstable behaviour. In discrete time this boundary is represented by the unit circle in the z-domain. For proper analogue the poles should be mirrored with respect to this unit circle and indeed they do. This is caused by the time shift of λ at the right hand side. This is simple to show remembering that z represents a forward time shift so that we can write:

$$\begin{pmatrix} zI - A & BR^{-1}B^T \\ -Q & z^{-1}I - A^T \end{pmatrix} \begin{pmatrix} x(k) \\ \lambda(k+1) \end{pmatrix} = 0 \tag{8.15}$$

where the continuous time equivalent is:

$$\begin{pmatrix} sI - A & BR^{-1}B^T \\ Q & sI + A^T \end{pmatrix} \begin{pmatrix} x \\ \lambda \end{pmatrix} = 0 \tag{8.16}$$

In the scalar continuous-time case we had:

$$\det \begin{pmatrix} s-a & b^2/r \\ q & s+a \end{pmatrix} = s^2 - a^2 - b^2\frac{q}{r} = 0 \quad \rightarrow \quad s_{1,2} = \pm\sqrt{a^2 + b^2\frac{q}{r}} \tag{8.17}$$

indeed poles mirrored with respect to the imaginary axis. For the discrete-time case we have:

$$\det \begin{pmatrix} z-a & b^2/r \\ -q & z^{-1}-a \end{pmatrix} = (z-a)(z^{-1}-a) + b^2\frac{q}{r} = 0 \tag{8.18}$$

It is clear that if z_1 is a solution, so is $z_2 = z_1^{-1}$.

This is quite fortunate, as we may now use the same reasoning as for the continuous case and conclude that if $N \rightarrow \infty$ neither x nor λ may contain unstable modes. So proper boundary values for λ have to accomplish that unstable modes are eliminated and we thus write again:

$$\lambda(k) = \bar{P}x(k) \tag{8.19}$$

Substition into equation 8.13 leads to the Discrete (time) Algebraic Riccati Equation (DARE):

$$\boxed{\bar{P} = A^T\bar{P}A + Q - A^T\bar{P}B(R + B^T\bar{P}B)^{-1}B^T\bar{P}A} \tag{8.20}$$

The desired solution for $\bar{P}$ is again the positive definite, symmetric solution which can be proved to be unique. The solution of this DARE leads to the control :

$$u(k) = -R^{-1}B^T\lambda(k+1) = -R^{-1}B^T\bar{P}x(k+1) \tag{8.21}$$

This puts us to a paradoxical, time problem: we need the future state x(k+1) to compute the input $u(k)$, where $x(k+1)$ indeed heavily depends on $u(k)$. Fortunately we know the complete future deterministically and therefore solve $u(k)$ expressed in earlier states $x(k)$ which yields:

$$\boxed{u(k) = -R^{-1}B^T\bar{P}x(k+1) = -(R + B^T\bar{P}B)^{-1}B^T\bar{P}Ax(k) = -Lx(k)} \tag{8.22}$$

Finally we state without proof that if N does not tend to infinity, we have to solve the following Discrete (time) Riccati Equation backwards in time:

$$\boxed{P(k) = Q + A^TP(k+1)(A - BL(k+1))} \tag{8.23}$$

with:

$$\boxed{L(k) = (R + B^TP(k)B)^{-1}B^TP(k)A} \tag{8.24}$$

and of course:

$$\boxed{u(k) = -L(k)x(k)} \tag{8.25}$$

The terminal condition appears to be:

$$P(N) = P_N \tag{8.26}$$

It is easy to verify what the general DRE is by substitution of $L(k+1)$:

$$\boxed{P(k) = A^TP(k+1)A + Q - A^TP(k+1)B(R + B^TP(k+1)B)^{-1}B^TP(k+1)A} \tag{8.27}$$

Clearly this is a recurrent expression for $P(k)$ backwards in time as expected. The stationary DARE results by convergence and thus by taking $P(k+1) = P(k) = \bar{P}$

Remarks:

- The DRE (Discrete Riccati Equation) can also be used for the time varying case, when A, B, C, Q and R are time dependent.
- Again the stochastic regulator is equivalent to the deterministic regulator for $N \rightarrow \infty$.

8.4 Example : Car parking.

Suppose that we want to design a controller for car parking, where the car is just moving along a straight trajectory. If we neglect the friction in bearings and tires, we can simply model this process by Newton's law:

$$F = m\ddot{x} \tag{8.28}$$

Here m is the mass of the car and $\ddot{x}$ its acceleration. The force F is effected by the driving engine and the brakes. This combined actuator can be controlled by both gas and brake pedals. By norming on mass (say 1 ton) and actuator gain we are simply left with a double integrator. So renaming input $F = u$ we have:

$$\ddot{x} = u \tag{8.29}$$

By taking as states $x_1 = x$ the position and $x_2 = \dot{x}$ the velocity the state space description is:

$$\begin{pmatrix} \dot{x_1} \\ \dot{x_2} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u \tag{8.30}$$

We try to solve a simple linear quadratic regulator (LQR) problem defined as illustrated in Fig. 8.1. The car's position is 1 unit of length in front of the parking place, how to park "gently". This simple problem can serve as an example for discrete-time controller design by first transforming the car's dynamics to discrete time domain with the configuration of Fig. 8.2 in mind. The zero order hold (impulse response equivalent) transform yields (check for yourself):

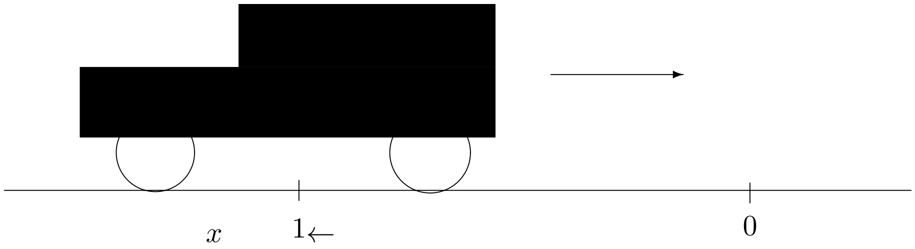


Figure 8.1: The car parking problem

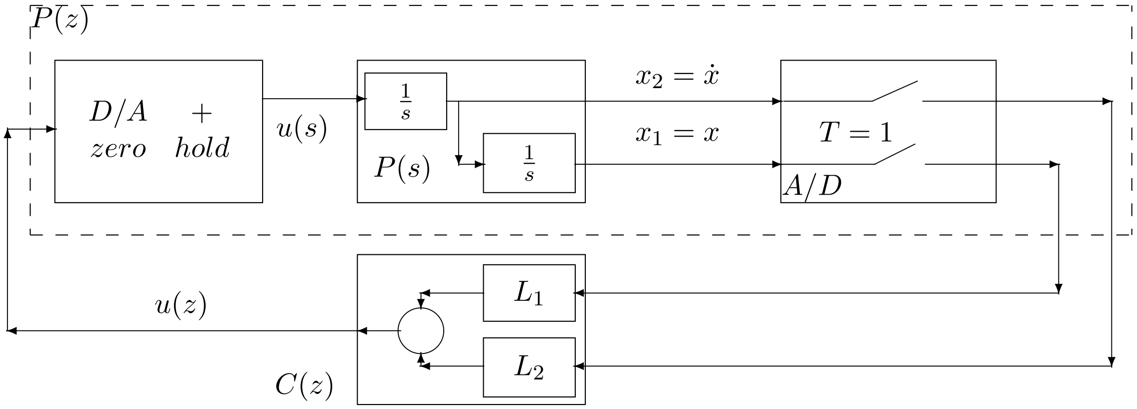


Figure 8.2: The sampled car parking problem

$$\begin{pmatrix} x_1(k+1) \\ x_2(k+1) \end{pmatrix} = \begin{pmatrix} 1 & T \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x_1(k) \\ x_2(k) \end{pmatrix} + \begin{pmatrix} \frac{1}{2}T^2 \\ T \end{pmatrix} u(k) \tag{8.31}$$

For simplicity we choose a sampling period $T = 1$ which is allowed as the transfer for the sampling frequency then amounts $1/(2\pi)^2 << 1$. For the optimality criterion we take:

$$J = \frac{1}{2}\Sigma_{k=1}^{\infty}\{x^T(k)Qx(k) + ru^2(k)\} \tag{8.32}$$

$N = \infty$, so that we will arrive at the steady state solution of the DARE. The choice for Q is:

$$Q = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \tag{8.33}$$

Note that the speed x_2 is not penalised. For the control effort several weights r are taken to illustrate its effect. For a "gentle" parking it is clear that the control effort, i.e. the gas and brake pedals positions, should be modest. This is what Fig. 8.3 shows in the upper left plot for $r = 1$. The block curve represents the engine and brake force $u(t)$.

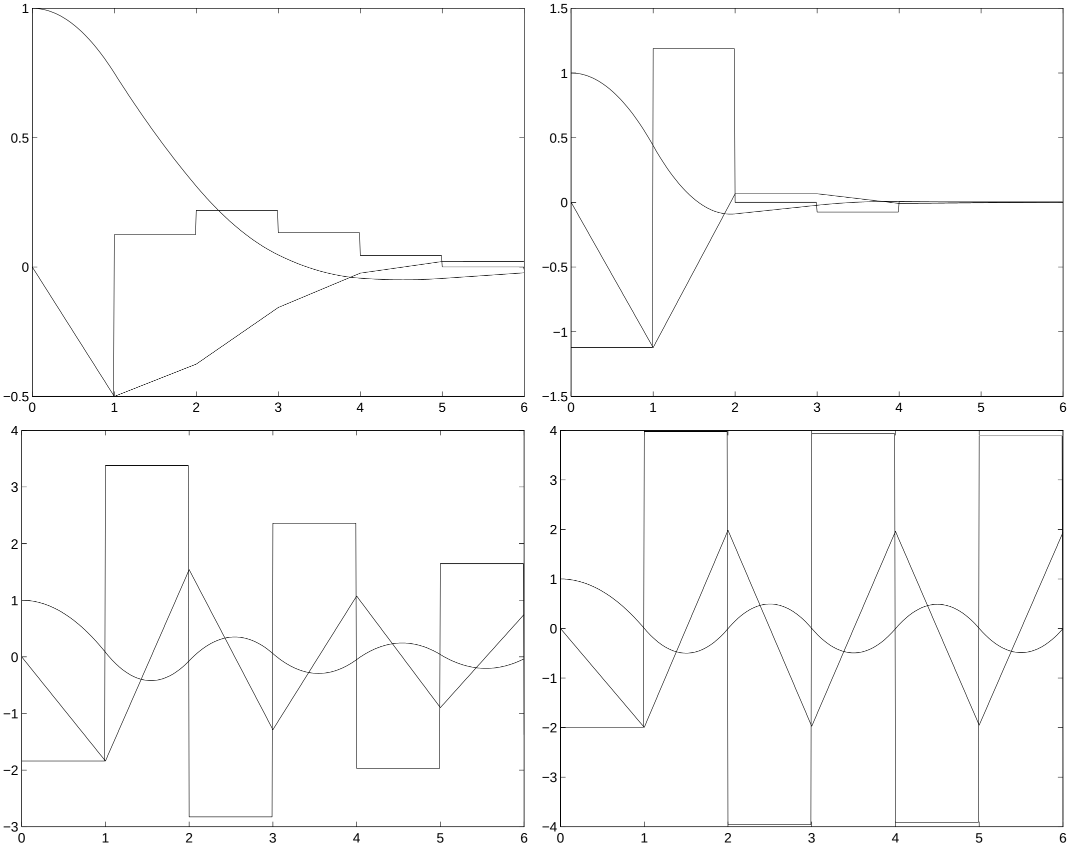


Figure 8.3: Position, velocity and control signal as a function of time for $r = 1$ (upper left), $r = .05$ (upper right), $r = .0005$ (lower left, $r = .0000005$ (lower right) and $Q_{i,j} = 0$ except $Q_{11} = 1$ weighting position x_1.

Surely, initially we drive backwards (negative force) and then brake smoothly (positive force). The speed $x_2(t)$ is the integral of the force so that we distinguish it as the straight line pieces. Again integration leads to the position $x_1(t)$ which is the smooth line built up by parabolic segments. So far the solution appears quite reasonable. Now suppose that we decrease r under the pressure of hurry or indifference for petrol consumption and environmental pollution. The resulting driving "as a sport" can easily be recognised by the increase of speed and braking in the right upper plot without gaining much time though. If we exaggerate this behaviour by still decreasing the force weight r we run into very strange behaviour, ultimately an oscillatory move about parking position 0. This intersample oscillation is caused by the fact that we wanted to decrease the time for the parking action beyond all bounds. One of the bounds was the frequency band determined by sampling period 1. At the sample instants the position control is perfect. In one sample period the car is at its aimed position and stays there for all subsequent sample instants. If one would look at the car with a stroboscopic illumination of 1 Hz, indeed the car would seemingly stand still. This looks like a dead beat control which it actually is. One pole is shifted to the origin of z-plane for the one sample delayed response and the other pole appears at -1 which is allowed as the position $x_1(k)$ does not contain its effect. Only the speed $x_2(k)$ has the consequent oscillatory behaviour, but this state was not penalised. See also Fig. 8.4.

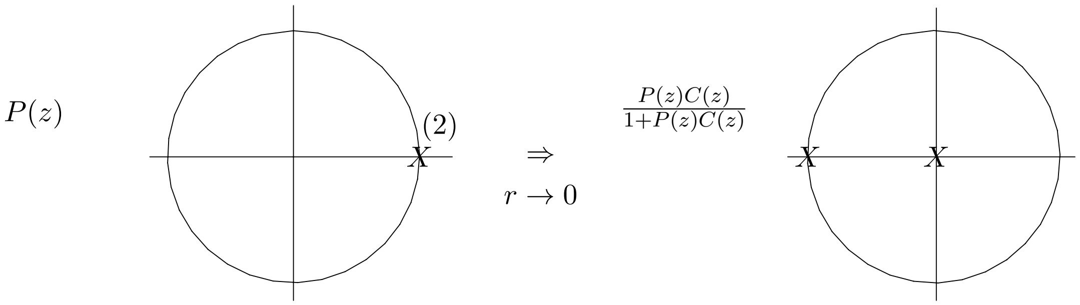


Figure 8.4: The shifted poles for $r \to 0$ and not weighted speed.

Knowing the cause of the problem we can now easily obtain a satisfactory solution by weighting the speed as well by taking:

$$Q = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \tag{8.34}$$

The resulting signals of the controlled system are shown in Fig. 8.5.

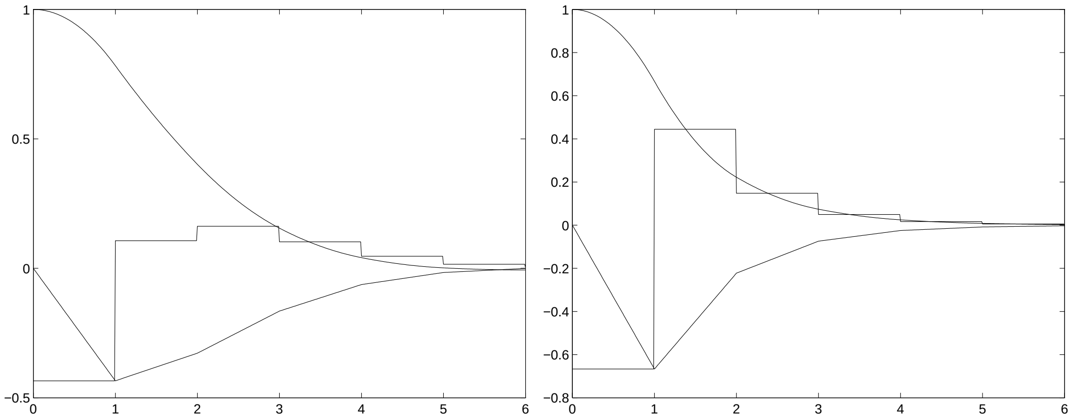


Figure 8.5: Position, velocity and control signal as a function of time for $r = 1$ (left), $r = .0000005$ (right) and Q is identity thus weighting both position and velocity.

Note that even for $r \to 0$ the behaviour is acceptable as the poles are shifted both to the stable part of the real axis in z-domain as Fig. 8.6 illustrates.

8.5 Optimal, linear, quadratic observer design

For the Kalman-Bucy filter design in discrete-time domain we have been given the equation:

$$\tilde{x}(k+1) = (A - KC)\tilde{x}(k) + v(k) - Kw(k) \tag{8.35}$$

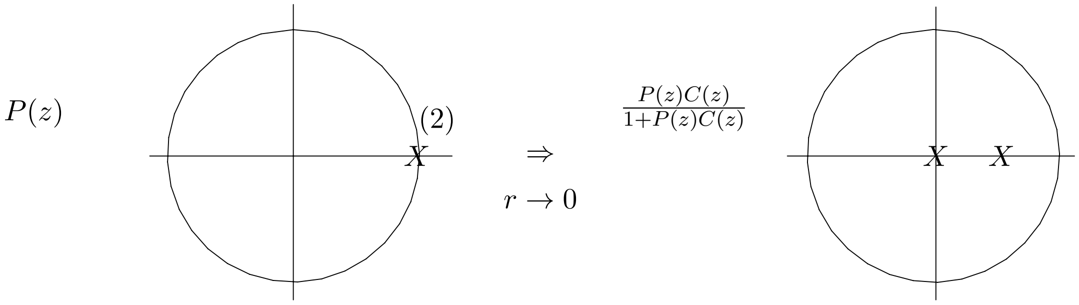


Figure 8.6: The shifted poles for $r \to 0$ and weighted speed.

Following the strategy of the continuous-time domain, we redefine this equation as:

$$z(k+1) = Fz(k) + q(k) \tag{8.36}$$

The continuous-time lemmas have their straightforward equivalents in discrete-time except the third lemma which deviates in a nontrivial manner:

$$\begin{gathered} E\{z(k+1)z^T(k+1)\} = \Psi(k+1) = \\ E\{(Fz(k)+q(k))(Fz(k)+q(k))^T = \\ FE\{z(k)z^T(k)\}F^T + E\{q(k)q^T(k)\} = \\ F\Psi(k)F^T + R_q \end{gathered} \tag{8.37}$$

since $q(k)$ is white noise with $E\{q(k)\} = 0$ and uncorrelated with $z(k)$. Substitution of the first lemma $\Psi = R_z + \bar{z}\bar{z}^T$ yields:

$$R_z(k+1) + \bar{z}(k+1)\bar{z}^T(k+1) = FR_z(k)F^T + F\bar{z}(k)\bar{z}^T(k)F^T + R_q \tag{8.38}$$

and because of the second lemma $\bar{z}(k+1) = F\bar{z}(k)$ we finally have:

$$R_z(k+1) = FR_z(k)F^T + R_q \tag{8.39}$$

By substitution of:

$$\begin{gathered} R_z(k) = R_{\tilde{x}}(k) = E\{\tilde{x}(k)\tilde{x}^T(k)\} = P(k) \\ R_q = R_v + KR_wK^T - KR_{vw}^T - R_{vw}K^T \\ F = A - KC \end{gathered} \tag{8.40}$$

we arrive at:

$$P(k+1) = (A-KC)P(k)(A-KC)^T + R_v + KR_wK^T - KR_{vw}^T - R_{vw}K^T \tag{8.41}$$

Along the same lines as for the continuous case the dynamical Kalman gain is then given by:

$$\boxed{K(k) = (AP(k)C^T + R_{vw})(R_w + CP(k)C^T)^{-1}} \tag{8.42}$$

While the covariance is obtained in a forward recursive formula:

$$\boxed{P(k+1) = (A - K(k)C)P(k)A^T + R_v - K(k)R_{vw}^T} \tag{8.43}$$

Note that again these two above equations are the dual form of the discrete time optimal control equations. Also the general DRE (discrete time Riccati equation) can be obtained by substitution of $K(k)$:

$$\boxed{P(k+1) = AP(k)A^T + R_v - (AP(k)C^T + R_{vw})(R_w + CP(k)C^T)^{-1}(CP(k)A^T + R_{vw})} \tag{8.44}$$

end the steady state solution simply follows by putting $P(k+1) = P(k) = \bar{P}$ yielding the DARE:

$$\boxed{\bar{P} = A\bar{P}A^T + R_v - (A\bar{P}C^T + R_{vw})(R_w + C\bar{P}C^T)^{-1}(C\bar{P}A^T + R_{vw})} \tag{8.45}$$

Remarks:

- The DRE can simply be solved forwards in time. The initial value amounts:

  $$P(0) = E\{\tilde{x}(0)\tilde{x}^T(0)\} \tag{8.46}$$

  if we take for the initial value of the observer state:

  $$\hat{x}(0) = E\{x(0)\} \tag{8.47}$$

  In steady state, when $N \to \infty$, these initial values are irrelevant again, as all effects die out due to necessary stable poles. The DARE is then used for the positive definite, symmetric $\bar{P}$.

- The DRE holds as well for time varying parameters $A(k), C(k), R_v(k), R_{vw}(k)$.

8.6 The innovations representation

The full, discrete time, LQG control has been depicted in Fig. 8.7 because the separation principle holds again.

All that has been said for the continuous time case can be repeated here without restriction. But on top of that we can pursue the analysis even further for the discrete time case. We will therefor first add an extra index to the state estimate by defining $\hat{x}(l/m)$ as the state estimate at instant l as before, while m indicates that for this estimate all real, measured outputs $y(k)$ till $k = m$ are used. The same holds for the dependent signals $\hat{y}(l/m) = C\hat{x}(l/m)$ and $u(l/m) = u^*(l) - L\hat{x}(l/m)$. For the observer part of Fig. 8.7 we can apply this as illustrated in Fig.8.8 by splitting the addition point for $\hat{x}(k+1)$ into two addition points. The equations corresponding to the addition points under discussion are:

$$\hat{x}(k+1/k-1) = A\hat{x}(k/k-1) + Bu(k/k-1) \tag{8.48}$$

and

$$\hat{x}(k+1/k) = \hat{x}(k+1/k-1) + K(y(k) - C\hat{x}(k/k-1)) \tag{8.49}$$

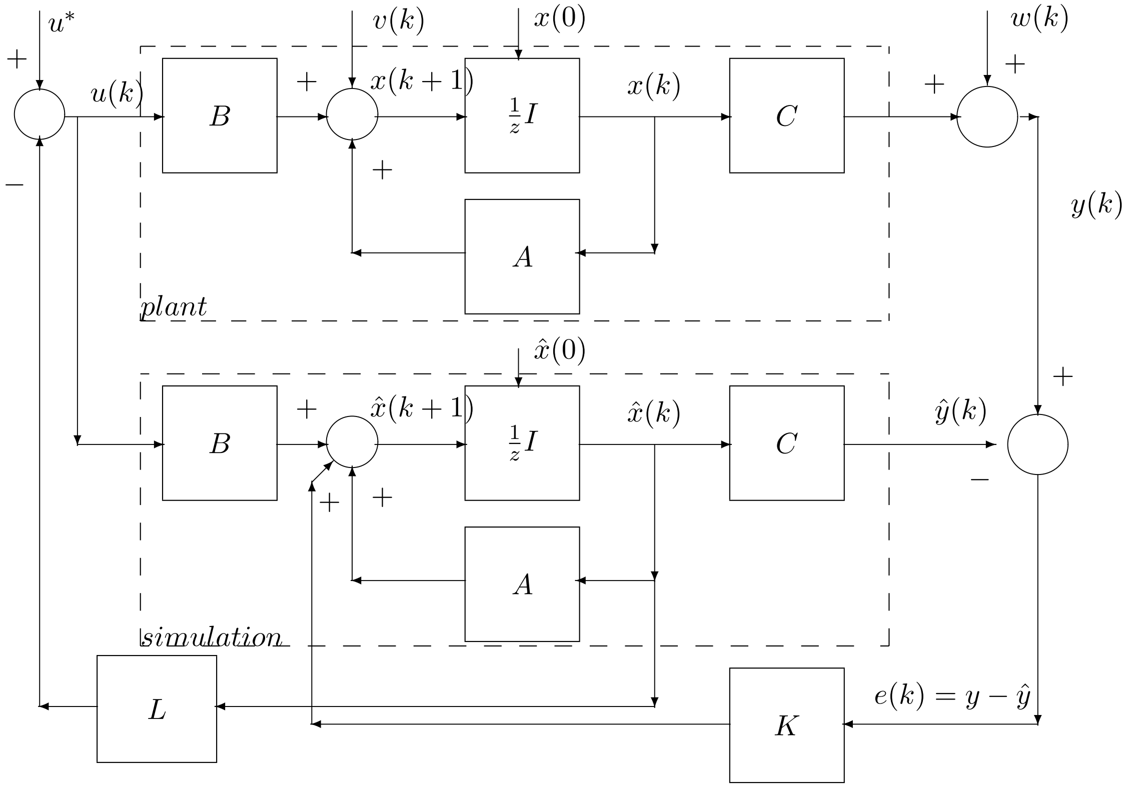


Figure 8.7: Discrete time LQG-control.

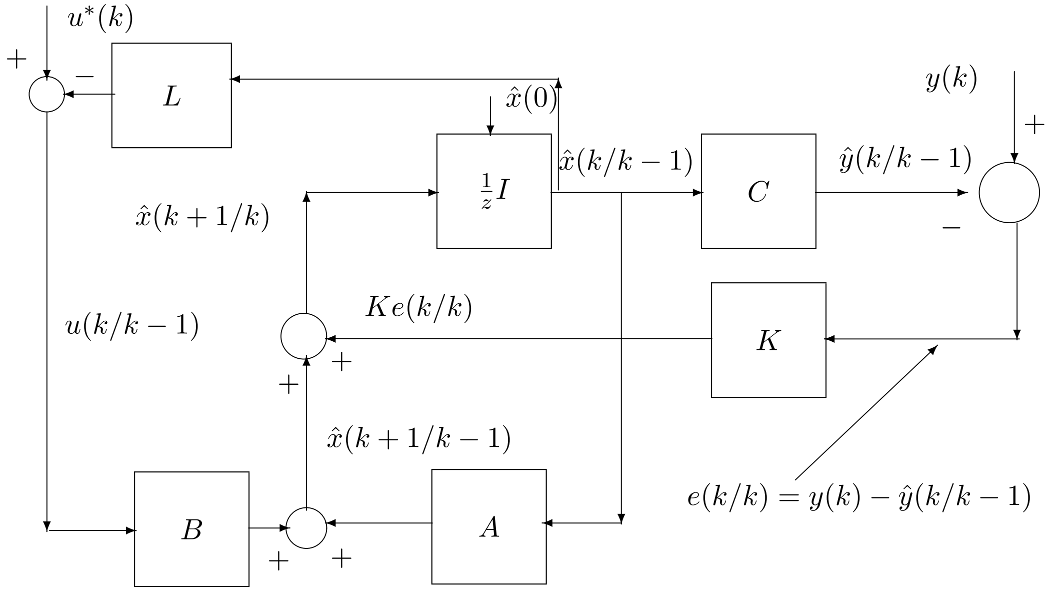


Figure 8.8: State prediction with indication of used measurements.

The last equation, shifted back in time over one sample, can be substituted in the previous yielding:

$$\hat{x}(k+1/k) = A\hat{x}(k/k-1) + Bu(k/k-1) + K(y(k) - C\hat{x}(k/k-1)) \tag{8.50}$$

We clearly deal with one step ahead **predicted** state estimates.

The whole problem of state estimation can be defined in the frame work of state **filtering** as well by first updating the estimate $\hat{x}(k/k-1)$ by $K^*e(k/k)$. The Kalman gain K^* will be different so that we gave it the superfix asterisk. The appropriate structure is depicted in Fig. 8.9. The state estimate updating equations now become:

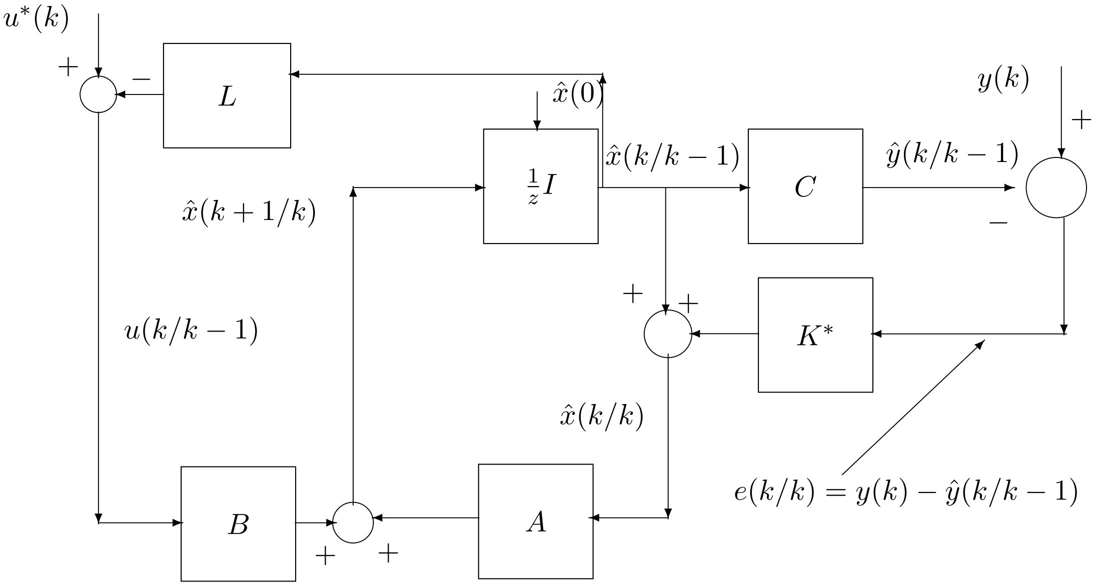


Figure 8.9: State filtering with indication of used measurements.

$$\hat{x}(k+1/k) = A\hat{x}(k/k) + Bu(k/k-1) \tag{8.51}$$

and

$$\hat{x}(k/k) = \hat{x}(k/k-1) + K^*(y(k) - C\hat{x}(k/k-1)) \tag{8.52}$$

The last equation can be substituted in the previous equation yielding:

$$\hat{x}(k+1/k) = A\hat{x}(k/k-1) + Bu(k/k-1) + AK^*(y(k) - C\hat{x}(k/k-1)) \tag{8.53}$$

which is equivalent with the prediction updating equation 8.50 if $K = AK^*$. This is indeed true and can be proved under the condition that K^* is optimised by minimising the following criterion:

$$\min_{K^*} E\{\tilde{x}^T(k)\tilde{x}(k)\} = \text{trace}\,(Q) \tag{8.54}$$

where

$$Q = E\{\tilde{x}(k)\tilde{x}^T(k)\} \tag{8.55}$$

and

$$\tilde{x}(k) = \hat{x}(k/k) - x(k) \tag{8.56}$$

which is completely comparable to the prediction where we had:

$$\min_{K} E\{\tilde{x}^T(k)\tilde{x}(k)\} = \text{trace}\,(P) \tag{8.57}$$

where

$$P = E\{\tilde{x}(k)\tilde{x}^T(k)\} \tag{8.58}$$

and

$$\tilde{x}(k) = \hat{x}(k/k-1) - x(k) \tag{8.59}$$

As we use one more sample $y(k)$ for the filtering, $\hat{x}(k/k)$ will be more accurate than the prediction $\hat{x}(k/k-1)$ so that also:

$$\text{trace}\,(Q) < \text{trace}\,(P) \tag{8.60}$$

but the ultimate LQG control is exactly the same since $K = AK^*$. From above figures this can easily be deducted.

Remark: In Matlab the *filter* Kalman gain K^* is computed!!

Exegesis: One can of course go through all the necessary formulas to prove above, but we prefer here to explain the essential machination. In *both* cases we deal with a misfit in the output *prediction* :

$$e(k) = y(k) - \hat{y}(k/k-1) \tag{8.61}$$

Certainly $\hat{y}(k/k-1)$ is the best prediction of $y(k)$ (in 2-norm) if all that can be predicted from the past is being used appropriately. That is, no more can be predicted so that the difference with the real output $y(k)$ expressed as $e(k)$ is zero mean, white noise! Indeed this is accomplished both for the prediction and the filtering method. This was hinted at for the continuous time case as well. As this error $e(k)$ attributes a completely new piece of information, independent of the past, we call $e(k)$ an **innovations** sequence. The whiteness of this "innovation" lends itself for a very simple trick. If we have :

$$e(k) = y(k) - \hat{y}(k) \tag{8.62}$$

we may also write:

$$y(k) = \hat{y}(k) + e(k) \tag{8.63}$$

so that we have an alternative state space representation: the innovations representation:

$$\begin{array}{rcl} \hat{x}(k+1) & = & A\hat{x}(k) + Bu(k) + Ke(k) \\ y(k) & = & C\hat{x}(k) + e(k) \end{array} \tag{8.64}$$

illustrated as well in Fig. 8.10.

Note that at the end addition the signals $y(k)$ and $e(k)$ have simply interchanged roles with respect to previous schemes according to equations 8.62 and 8.63. This innovations representation is an *exact* description of the process, though one has to keep in mind that we no longer deal with the actual state disturbance $v(k)$ and measurement noise $w(k)$. Although the state matrices are the original ones, the states (estimated states) are disturbed by the innovations $e(k)$ that represent both the state and the measurement noise, but surely $Ke(k) \neq v(k)$ and $e(k) \neq w(k)$! How can *one* sequence $e(k)$ represent implicitly two noise sources $v(k)$ and $w(k)$? Simply by the fact that, if we consider only the output $y(k)$, any spectrum of y caused by many noise sources in superposition can always

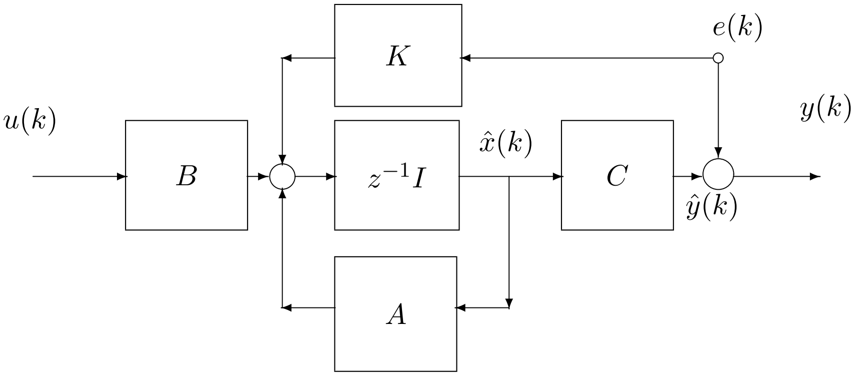


Figure 8.10: Innovations representation.

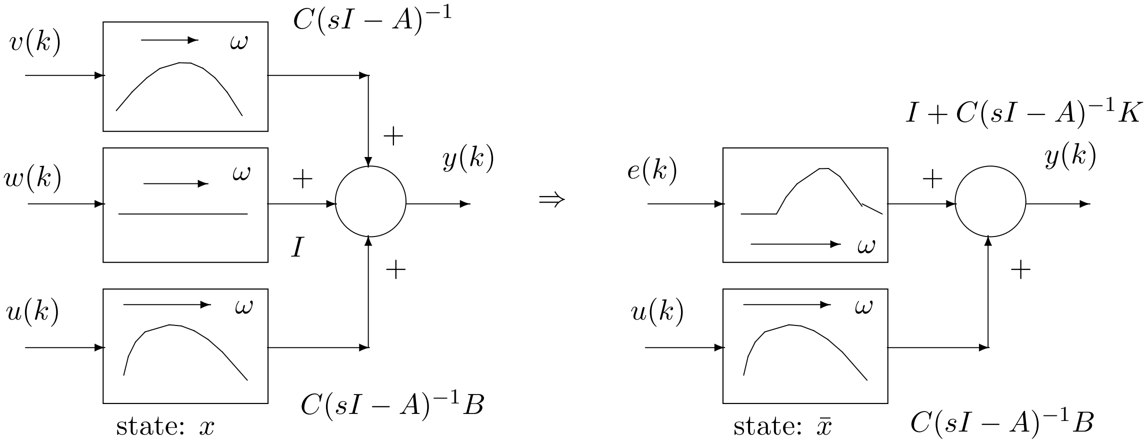


Figure 8.11: Innovations representation of filtered noise sources.

be represented by filtered white noise, which is in fact the innovations representation. Only $q = \dim(y)$ independent, white noise sources can be distinguished, which is precisely the dimension of the innovation sequence $e(k)$. Fig. 8.11 illustrates this effect.

By taking the original observer equations we can obtain the actual innovations as outputs:

$$\begin{aligned} \hat{x}(k+1) &= (A-KC)\hat{x}(k) + u(k) + Ky(k) \\ e(k) &= y(k) - C\hat{x}(k) \end{aligned} \tag{8.65}$$

where $u(k)$ and $y(k)$ are obtained from the real process. This operation is the so called whitening filter as depicted in Fig. 8.12. Such a whitening filter is frequently used to estimate the Kalman gain by changing its entries until $\epsilon(k)$ as an estimate of $e(k)$ becomes a white sequence. It appears that we then just have to minimise:

$$\min_{\hat{K}} \Sigma_k \epsilon^T(k)\epsilon(k) \tag{8.66}$$

In this way we need not to have numerical values for the state space noise v and the measurement noise w! Also there is no need for solving any Riccati equation. This whitening filter can also be used to obtain an estimate of the plant matrices if these are not known beforehand. Therefor we need to minimise (according to course "Stochastic systems theory"):

$$\min_{\hat{A},\hat{B},\hat{C},\hat{K}} \Sigma_k \epsilon^T(k)\epsilon(k) \tag{8.67}$$

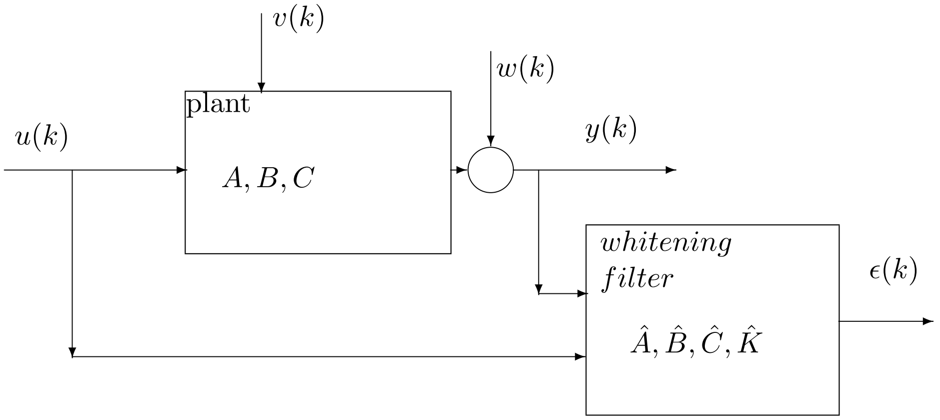


Figure 8.12: The whitening filter.

Indeed a very powerful tool for both estimating plant parameters and Kalman gain!

Finally, it will be clear that $e(k)$ is zero mean, otherwise its mean would be the best prediction. For its variance we can derive:

$$y(k) = Cx(k) + w(k) \tag{8.68}$$

$$y(k) = C\hat{x}(k) + e(k) \tag{8.69}$$

Elimination of $y(k)$ yields:

$$e(k) = C\tilde{x}(k) + w(k) \tag{8.70}$$

so that

$$E\{e(k)e^T(k)\} = CE\{\tilde{x}(k)\tilde{x}^T(k)\}C^T + E\{w(k)w^T(k)\} = CPC^T + R_w \tag{8.71}$$

because $\tilde{x}(k)$ is not correlated with $w(k)$ since:

$$\tilde{x}(k+1) = (A - KC)\tilde{x}(k) + v(k) - Kw(k) \tag{8.72}$$

and $w(k)$ is white.

Bibliography

[1] Isidori, A., "Nonlinear Control Systems: an Introduction", Lecture Notes in Control and Information Sciences, 72, Springer, 1985.

[2] Kailath, T., "Linear Systems", Prentice Hall inc., Engelwood Cliffs, New Yersey .1980.

[3] Khalil, H.K., "Nonlinear Systems", Macmillan Publishing Company, New York, 1992.

[4] Kwakernaak, H. and R.Sivan, "Linear Optimal Control Systems", John Wiley and Sons inc., New York, 1972.

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