Systems Theory Summary

1. Linear systems theory

What are systems? And how do we respresent them/deal with them? That is what this summary is all about. In the first chapter, we look at what systems are, and how we deal with basic linear systems. In the second chapter, we discuss system properties like stability and controllability. Thirdly, we give some thoughts on how we can use feedback in controlling systems. And finally, in the fourth chapter, we'll examine transfer functions in detail. But first, we start by asking the fundamental question: what is systems theory?

1.1 Basic systems theory principles

1.1.1 What is systems theory?

A **system** is a part of reality that can be seen as a separate unit. The reality outside the system is known as the **surroundings**. Of course, the system and the surroundings influence each other. The environment influences the system by **input**, denoted by the **input vector** $\mathbf{u}(t)$. Similarly, the system influences the environment by means of the **output** $\mathbf{y}(t)$.

Mathematical **systems theory** (sometimes also called **system theory**) concerns the study and control of systems. In particular, input/output phenomena are examined.

1.1.2 Modelling principles

Before we can actually concern ourselves with systems, we need to know how we can model them. This is done, using **modelling principles**. The three most important ones are the following principles.

- **Conservation laws** state that certain quantities (like mass or energy) are conserved.
- **Physical laws** describe important relations between variables. (Think of Newton's laws, or the laws of thermodynamics.)
- **Phenomenological principles** are principles that are known from experience. (Examples are Ohm's law of electrical resistance and Fourier's law of heat conduction.)

By using these principles, often a model of the system can be achieved. This model can then be used to examine and control the system.

1.2 Making differential systems linear

The most easy type of differential systems is the linear one. What are linear differential systems? And how do we get them?

1.2.1 What are linear differential systems?

A **linear differential system** is a system that can be written as

$$\dot{\mathbf{x}}(t) = A(t)\mathbf{x}(t) + B(t)\mathbf{u}(t), \tag{1.2.1}$$
$$\mathbf{y}(t) = C(t)\mathbf{x}(t) + D(t)\mathbf{u}(t). \tag{1.2.2}$$

These equations are known as the **state equation** and the **output equation**, respectively. $\mathbf{x}(t)$ denotes the $(n \times 1)$ **state vector** of the system, $\mathbf{u}(t)$ is the $(m \times 1)$ **input vector** and $\mathbf{y}(t)$ is the $(p \times 1)$ **output vector**. Also, $A(t)$ is the $(n \times n)$ **state matrix**, $B(t)$ is the $(n \times m)$ **input matrix**, $C(t)$ is the $(r \times n)$ **output matrix** and $D(t)$ is the $(r \times m)$ **direct (transmission) matrix**. If the matrices A, B, C and D do not depend on time, then the system is said to be **time-invariant**.

To solve the output $\mathbf{y}(t)$ of the system, two things need to be known. First of all, the initial state of the system $\mathbf{x}(0)$ (often also denoted as $\mathbf{x_0}$) needs to be known. Second, the input $\mathbf{u}(t)$ must be given. If these two things are given (and have the right properties), then the output $\mathbf{y}(t)$ is well defined.

1.2.2 Linearization

Linear systems are quite important in systems theory. This is mainly because they are easy to work with. Non-linear systems are more difficult to work with. Luckily, non-linear systems can be approximated by a linear system. This is called **linearization**. But how does it work?

Let's suppose we have a non-linear system, described by

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}, \mathbf{u}) \qquad \text{and} \qquad \mathbf{y}(t) = \mathbf{g}(\mathbf{x}, \mathbf{u}). \tag{1.2.3}$$

We also suppose that we have a solution $\tilde{\mathbf{x}}(t)$ and $\tilde{\mathbf{y}}(t)$ for given initial conditions $\tilde{\mathbf{x}}_\mathbf{0}$ and input $\tilde{\mathbf{u}}(t)$. Now, let's suppose that we have a problem with a slightly different input function $\tilde{\mathbf{u}}(t) + \mathbf{v}(t)$. ($\mathbf{v}(t)$ is thus the deviation from the original input function.) This will then give a solution $\tilde{\mathbf{x}}(t) + \mathbf{z}(t)$ for the state and $\tilde{\mathbf{y}}(t) + \mathbf{w}(t)$ for the output. (Again, $\mathbf{z}(t)$ and $\mathbf{w}(t)$ are deviations!) It can now be shown (using a Taylor-expansion about the original solution) that we can write the system of equations as

$$\dot{\mathbf{z}}(t) = A(t)\mathbf{z}(t) + B(t)\mathbf{v}(t), \tag{1.2.4}$$
$$\mathbf{w}(t) = C(t)\mathbf{z}(t) + D(t)\mathbf{v}(t). \tag{1.2.5}$$

In other words, we have linearized the system! The matrices A, B, C and D can be found, using

$$A(t) = \frac{\partial f}{\partial x}(\tilde{\mathbf{x}}, \tilde{\mathbf{u}}) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}, \qquad B(t) = \frac{\partial f}{\partial u}(\tilde{\mathbf{x}}, \tilde{\mathbf{u}}) = \begin{bmatrix} \frac{\partial f_1}{\partial u_1} & \cdots & \frac{\partial f_1}{\partial u_m} \\ \vdots & & \vdots \\ \frac{\partial f_n}{\partial u_1} & \cdots & \frac{\partial f_n}{\partial u_m} \end{bmatrix}, \tag{1.2.6}$$

$$C(t) = \frac{\partial g}{\partial x}(\tilde{\mathbf{x}}, \tilde{\mathbf{u}}) = \begin{bmatrix} \frac{\partial g_1}{\partial x_1} & \cdots & \frac{\partial g_1}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial g_n}{\partial x_1} & \cdots & \frac{\partial g_n}{\partial x_n} \end{bmatrix}, \qquad D(t) = \frac{\partial f}{\partial u}(\tilde{\mathbf{x}}, \tilde{\mathbf{u}}) = \begin{bmatrix} \frac{\partial g_1}{\partial u_1} & \cdots & \frac{\partial g_1}{\partial u_m} \\ \vdots & & \vdots \\ \frac{\partial g_n}{\partial u_1} & \cdots & \frac{\partial g_n}{\partial u_m} \end{bmatrix}. \tag{1.2.7}$$

In other words, you first find the derivative matrices (like $\frac{\partial f}{\partial x}$). You then insert the original solution $\tilde{\mathbf{x}}, \tilde{\mathbf{u}}$ into this matrices, and the linearization is complete!

However, it is very important to remember that you are dealing with deviations from the original solution. In other words, your quantities $\mathbf{v}$, $\mathbf{z}$ and $\mathbf{w}$ are not the real physical quantities. And, to make things even worse, books often use the symbols $\mathbf{u}$, $\mathbf{x}$ and $\mathbf{y}$ to indicate both the normal physical quantities, and the deviations from the original solution. Always make sure that you know what each symbol exactly means!

1.2.3 Solving linear differential systems

Let's suppose we have a linear differential system $\dot{\mathbf{x}}(t) = A(t)\mathbf{x}(t) + B(t)\mathbf{u}(t)$. We now would like to find a solution. To do this, we need to perform four steps.

1. Find the n independent solutions $\boldsymbol{\xi_1}(t), \boldsymbol{\xi_2}(t), \ldots, \boldsymbol{\xi_n}(t)$ of the so-called **autonomous state equation** $\dot{\mathbf{x}} = A(t)\mathbf{x}$.

2. Assemble the **fundamental matrix** $Y(t)$, according to

$$Y(t) = \begin{bmatrix} \boldsymbol{\xi_1}(t) & \boldsymbol{\xi_2}(t) & \ldots & \boldsymbol{\xi_n}(t) \end{bmatrix}. \tag{1.2.8}$$

3. Assemble the **transition matrix** $\Phi(t, s)$ according to

$$\Phi(t, s) = Y(t)Y^{-1}(s). \tag{1.2.9}$$

   This matrix has several interesting properties, like

$$\Phi(t, t) = I, \qquad \Phi(t_2, t_0) = \Phi(t_2, t_1)\Phi(t_1, t_0) \qquad \text{and} \qquad \Phi^{-1}(t, s) = \Phi(s, t). \tag{1.2.10}$$

4. Find the solution $\mathbf{x}(t)$ of the state of the system, using

$$\mathbf{x}(t) = \Phi(t, t_0)\,\mathbf{x_0} + \int_{t_0}^{t} \Phi(t, s)\,B(s)\,u(s)\,ds \tag{1.2.11}$$

1.3 Time-invariant systems

A **time-invariant system** is a system described by $\dot{\mathbf{x}}(t) = A\mathbf{x}(t) + B\mathbf{u}(t)$. In other words, the matrices do not depend on time. How do we deal with this kind of systems?

1.3.1 The solution for time-invariant systems

Time-invariant systems are a lot more easy to solve than time-dependent systems. Normally, it can be very time-consuming to find the transition matrix Φ. However, for time-invariant systems, we simply have

$$\Phi = e^{(t-s)A}. \tag{1.3.1}$$

By the way, the exponential of a matrix is defined as an infinite series, according to

$$e^{tA} = I + tA + \frac{t^2A^2}{2!} + \frac{t^3A^3}{3!} + \ldots = \sum_{i=0}^{\infty} \frac{t^iA^i}{i!}. \tag{1.3.2}$$

There's just one problem. To find e^{tA}, we have to compute an infinite series. And this usually takes quite a long time. But luckily, there are some tricks. But before we examine those tricks, we need to recap on linear algebra.

1.3.2 Linear algebra recap

Let's suppose we have an $n \times n$ matrix A. This matrix has k eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_k$. Each of these eigenvectors λ_i has an **algebraic multiplicity** m_i and a **geometric multiplicity** g_i. The algebraic multiplicity m_i is the number of times which λ_i appears as a root of the **characteristic polynomial**

$\det(\lambda I - A)$. The geometric multiplicity g_i is the number of eigenvectors $\mathbf{q_{ij}}$ corresponding to the eigenvalue λ_i. In other words, it is the number of linearly independent solutions $\mathbf{q_{ij}}$ to the equation

$$(\lambda_i I - A)\,\mathbf{q_{ij}} = \mathbf{0}. \tag{1.3.3}$$

The geometric multiplicity is never bigger than the algebraic multiplicity. So, $g_i \leq m_i$. Also, the sum of all algebraic multiplicities equals the size of the matrix n. So, $\sum_{i=1}^{k} m_i = n$.

We say that the matrix A is **diagonalizable** if there exists an invertible matrix T such that $T^{-1}AT = D$, where D is a diagonal matrix. It can be shown that A is only diagonalizable, if $g_i = m_i$ for all eigenvalues λ_i. If this is indeed the case, then D is the matrix of eigenvalues $D = \operatorname{diag}(\lambda_1, \ldots, \lambda_k)$. In other words, it is the matrix with the k eigenvalues on its diagonal. (By the way, if an eigenvalue λ_i has a multiplicity of m_i, then it also appears m_i times in D. So, D is an $n \times n$ matrix.) Similarly, T is the matrix with as columns the corresponding eigenvectors $\mathbf{q_i}$.

1.3.3 Solutions for diagonalizable systems

Now it's time to find an alternate expression for e^{tA}. Let's suppose that A is diagonalizable. Then, we have $T^{-1}AT = D$. It can now be shown that

$$e^{tA} = Te^{t\left(T^{-1}AT\right)}T^{-1} = Te^{tD}T^{-1} = T\begin{bmatrix} e^{\lambda_1 t} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & e^{\lambda_k t} \end{bmatrix} T^{-1}. \tag{1.3.4}$$

Let's denote the rows of T by $\mathbf{w_1}, \ldots, \mathbf{w_n}$. The columns are still denoted by $\mathbf{q_1}, \ldots, \mathbf{q_n}$.We can now also write

$$A = \sum_{i=1}^{n} \lambda_i \mathbf{q_i w_i}, \qquad \text{and similarly,} \qquad e^{tA} = \sum_{i=1}^{n} e^{t\lambda_i} \mathbf{q_i w_i}. \tag{1.3.5}$$

Let's suppose that the system has no input. (Thus, $\mathbf{u}(t) = \mathbf{0}$.) The solution $\mathbf{x}(t)$ of the system is now known as the **free response** of the system. It is given by

$$\mathbf{x}(t) = e^{(t-t_0)A}\mathbf{x_0} = \sum_{i=1}^{n} e^{(t-t_0)\lambda_i}\mathbf{q_i w_i x_0} = \sum_{i=1}^{n} \mu_i e^{(t-t_0)\lambda_i}\mathbf{q_i}, \tag{1.3.6}$$

where $\mu_i = \mathbf{w_i x_0}$. (Remember that $\mathbf{w_i}$ is a row vector.) The above equation implies an interesting fact. The free response of a system $\mathbf{x}(t)$ can be decomposed along the eigenvectors $\mathbf{q_i}$. The solution corresponding to one eigenvector $\mathbf{q_i}$ is called a **mode** of the system. A system will be in mode i, if the initial vector $\mathbf{x_0}$ is aligned with the eigenvector $\mathbf{q_i}$.

1.3.4 The Jordan form

We now know what to do with systems if A is diagonalizable. But what if A is not diagonalizable? In this case, we don't have $T^{-1}AT = D$. Instead, we will use $T^{-1}AT = J$. In this equation, J is the so-called **Jordan form** of A. It has a block-diagonal structure $J = \operatorname{diag}(J_1, J_2, \ldots, J_k)$. Every submatrix J_i is an $m_i \times m_i$ matrix and corresponds to the eigenvalue λ_i. In fact, it has the form

$$J_i = \begin{bmatrix} \lambda_i & \gamma_{i1} & 0 & \cdots & 0 \\ 0 & \lambda_i & \gamma_{i2} & \ddots & \vdots \\ 0 & 0 & \lambda_i & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \gamma_{i(m_i-1)} \\ 0 & \cdots & 0 & 0 & \lambda_i \end{bmatrix}. \tag{1.3.7}$$

In this equation, some of the values γ_{ij} are 0, while others are 1. This, in fact, depends on how T is build up. We can remember that $T^{-1}AT = J$ and thus $AT = TJ$. If the individual columns of T are written as $\mathbf{q_1}, \ldots, \mathbf{q_n}$, then we have

$$A\mathbf{q_i} = \lambda\mathbf{q_i} + \gamma\mathbf{q_{i-1}}, \tag{1.3.8}$$

with λ the corresponding eigenvalue in J. Note that, if $\gamma = 0$, then $\mathbf{q_i}$ is an eigenvector of A, corresponding to the eigenvalue λ. If, instead, $\gamma = 1$, then we have

$$\mathbf{q_{i-1}} = (A - \lambda I)\,\mathbf{q_i}. \tag{1.3.9}$$

The vector $\mathbf{q_i}$ is known as a **generalized eigenvector**. It can be derived from the (possibly generalized) eigenvector $\mathbf{q_{i-1}}$.

We can now see a relation between the form of J and T. The zeroes in J correspond with the positions of the normal eigenvectors in T. Similarly, the ones in J correspond to the positions of the generalized eigenvectors in T.

1.4 System response

We now know how the state of a system behaves. But what about the output? That's what we'll look at now.

1.4.1 The impulse response

The output of the system is given by

$$\mathbf{y}(t) = C(t)\mathbf{x}(t) + D(t)\mathbf{u}(t) = C(t)\Phi(t, t_0)\mathbf{x_0} + \int_{t_0}^{t} C(t)\Phi(t, s)B(s)\mathbf{u}(s)\,ds + D(t)\mathbf{u}(t). \tag{1.4.1}$$

To simplify matters, we define $K(t, s)$ such that

$$K(t, s) = C(t)\Phi(t, s)B(s). \tag{1.4.2}$$

Let's assume that there is some time t_0 for which $\mathbf{x_0} = \mathbf{0}$. This eliminates the left term in the output equation. What we remain with is some sort of **mapping function**: we map the output onto the input, without having to know anything about the state. This thus gives us an **external description** of the system.

Now, let's assume that $D(t) = 0$. Also, the input function is given by $\mathbf{u(t)} = \delta(t - t_1)\mathbf{e_i}$. In this relation, δ denotes the **unit impulse function** and $\mathbf{e_i}$ denotes the i'th basis vector. The output $\mathbf{y}(t)$ is now given by

$$y(t) = \int_{t_0}^{t} K(t, s)\delta(s - t_1)\mathbf{e_i}\,ds = K(t, t_1)\mathbf{e_i} = \text{the } i\text{'th column of } K(t, t_1). \tag{1.4.3}$$

The matrix $K(t, t_1)$ can thus be seen as the response to an impulse function. For this reason, $K(t, t_1)$ is known as the **impulse response matrix**.

1.4.2 The step response

After examining the impulse function, we will now look at the step function. This time, we assume that the input is given by $\mathbf{u}(t) = H(t - t_1)\mathbf{e_i}$, where $H(t)$ is known as the **unit step function**. This results in an output

$$\mathbf{y}(t) = \int_{t_0}^{t} K(t, s)H(s - t_1)\mathbf{e_i}\,ds = \int_{t_1}^{t} K(t, s)\mathbf{e_i}\,ds = S(t, t_1)\mathbf{e_i}. \tag{1.4.4}$$

The matrix $S(t, t_1)$ is known as the **step response matrix**. It is related to $K(t, s)$ according to

$$S(t, t_1) = \int_{t_1}^{t} K(t, s)\, ds \qquad \text{and} \qquad \frac{d}{ds} S(t, s) = \frac{d}{ds} \int_{s}^{t} K(t, \tau)\, d\tau = -K(t, s). \tag{1.4.5}$$

You might wonder why the minus sign on the right side is present. In short, this is because s is the lower limit of the integration.

2. System properties

There are many ways to characterize a system. We can for example divide systems in stable/unstable systems, controllable/uncontrollable systems or observable/unobservable systems. This chapter discusses the meanings of these terms.

2.1 Stability

2.1.1 Definitions

Let's consider the time-invariant systems $\dot{\mathbf{x}} = f(\mathbf{x})$. Given an initial point $\mathbf{x_0}$, the system will have a solution $\mathbf{x}(t, \mathbf{x_0})$. Now it's time to make some definitions.

- We say that a vector $\bar{\mathbf{x}}$, which satisfies $\dot{\bar{\mathbf{x}}} = f(\bar{\mathbf{x}}) = \mathbf{0}$, is an **equilibrium point**. If we take $\mathbf{x_0} = \bar{\mathbf{x}}$, then $\mathbf{x}(t, \mathbf{x_0}) = \bar{\mathbf{x}}$.
- An equilibrium point $\bar{\mathbf{x}}$ is called **stable** if, for every boundary $\varepsilon > 0$, there is a distance $\delta > 0$ such that, if $||\mathbf{x_0} - \bar{\mathbf{x}}|| < \delta$, then $||\mathbf{x}(t, \mathbf{x_0}) - \bar{\mathbf{x}}|| < \varepsilon$ for all $t \geq 0$. An equilibrium point that is not stable is termed **unstable**.
- A stable equilibrium point is also called **asymptotically stable** if $\lim_{t\to\infty} ||\mathbf{x}(t, \mathbf{x_0}) - \bar{\mathbf{x}}|| = 0$, given that $||\mathbf{x_0} - \bar{\mathbf{x}}|| < \delta$ for some $\delta > 0$.

2.1.2 Eigenvalue method

The question is, how do we know if an equilibrium point is stable? There are several methods to find that out. We start with the **eigenvalue method**. This trick works for linear time-invariant systems, with $\dot{\mathbf{x}} = A\mathbf{x}$. These systems have $\bar{\mathbf{x}} = \mathbf{0}$ as equilibrium solution. (There are also other equilibrium solutions if $\det A = 0$, but we won't go into detail on that.)

Let's denote the k eigenvalues of A by $\lambda_1, \lambda_2, \ldots, \lambda_k$. We now have to look at the real parts $\text{Re}\ \lambda_i$ of these eigenvalues.

- The point $\bar{\mathbf{x}} = \mathbf{0}$ is asymptotically stable if and only if $\text{Re}\ \lambda_i < 0$ for all eigenvalues λ_i.
- The point $\bar{\mathbf{x}} = \mathbf{0}$ is stable if $\text{Re}\ \lambda_i \leq 0$ for all eigenvalues λ_i. Also, for the eigenvalues λ_i with $\text{Re}\ \lambda_i = 0$, the algebraic multiplicity m_i must equal the geometric multiplicity g_i.
- In any other case, the point $\bar{\mathbf{x}} = \mathbf{0}$ is unstable. So, either there is an eigenvalue λ_i with $\text{Re}\ \lambda_i > 0$, or there is an eigenvalue λ_i with $\text{Re}\ \lambda_i = 0$ for which $g_i < m_i$.

Now we can ask ourselves, what do we do with a non-linear system $\dot{\mathbf{x}} = f(\mathbf{x})$ with equilibrium point $\bar{\mathbf{x}} = \mathbf{0}$? Well, we simply linearize it. In other words, we write it as $\dot{\mathbf{x}} = A\mathbf{x} + h(\mathbf{x})$, where $h(\mathbf{x})$ only contains higher-order terms of $\mathbf{x}$. The stability of $\bar{\mathbf{x}}$ now depends on the matrix A. In other words, if the eigenvalues of A all have negative real parts, then the equilibrium point $\bar{\mathbf{x}} = \mathbf{0}$ is asymptotically stable. There is one side-note though: $h(\mathbf{x})$ has to be real and continuous for $\mathbf{x}$ near $\bar{\mathbf{x}} = \mathbf{0}$.

2.1.3 Routh's criterion

To apply the eigenvalue method, we need to find the eigenvalues λ_i. They are usually found, by solving the characteristic polynomial

$$\det(A - \lambda I) = a_0 + a_1\lambda + a_2\lambda^2 + \ldots + a_n\lambda^n. \tag{2.1.1}$$

Especially for big n, it can be very difficult to solve the above equation. But, in fact, we don't need to know the exact eigenvalues. We only want to know whether they're all in the negative (left) part of the complex plane! And to find this out, we can use **Routh's criterion**. According to Routh's criterion, we first construct the **Routh table**.

$$\begin{array}{cccc} a_n & a_{n-2} & a_{n-4} & \dots \\ a_{n-1} & a_{n-3} & a_{n-5} & \dots \\ b_{n-2} & b_{n-4} & b_{n-6} & \dots \\ c_{n-3} & c_{n-5} & c_{n-7} & \dots \\ d_{n-4} & d_{n-6} & d_{n-8} & \dots \\ \vdots & \vdots & \vdots & \ddots \end{array} \tag{2.1.2}$$

There are a few important rules. Any number with a negative coefficient (like for example a_{-1}) is equal to zero. You also continue calculating this pattern, until you wind up with only zeroes. (This happens after at most $n+1$ rows.) For the rest, you can find the coefficients b_i, c_i, d_i, etc. using

$$b_{n-2} = \frac{a_{n-1}a_{n-2} - a_n a_{n-3}}{a_{n-1}}, \qquad b_{n-4} = \frac{a_{n-1}a_{n-4} - a_n a_{n-5}}{a_{n-1}}, \tag{2.1.3}$$

$$c_{n-3} = \frac{b_{n-2}a_{n-3} - a_{n-1}b_{n-4}}{b_{n-2}}, \qquad c_{n-5} = \frac{b_{n-2}a_{n-5} - a_{n-1}b_{n-6}}{b_{n-2}}. \tag{2.1.4}$$

Do you see the pattern? To calculate a number x, you always use four numbers. Two of these numbers come from the column just right of x. The other two numbers come from the leftmost column. All four numbers come from the two rows just above x.

We still have one question remaining. What does this tell us about the eigenvalues? Well, according to Routh's criterion, we have to look at the left column. There must be $n+1$ nonzero numbers in this column. Also, all these numbers must have the same sign. Only if this is the case, then the eigenvalues of A all have a negative real part. And this then of course implies that the system is asymptotically stable.

2.1.4 Lyapunov stability

There is another way in which we can determine whether a system $\dot{\mathbf{x}} = A\mathbf{x}$ is asymptotically stable. For this method, we first have to define some kind of **'generalized' energy** $V(\mathbf{x}(t)) = \mathbf{x}^T P\mathbf{x}$, where P is a positive-definite matrix. (Positive-definite means that $\mathbf{x}^T P\mathbf{x} > 0$ for all vectors $\mathbf{x} \neq \mathbf{0}$. So, V is always positive, unless $\mathbf{x} = \mathbf{0}$.) The derivative of V is

$$\frac{d}{dt}V(\mathbf{x}(t)) = \dot{\mathbf{x}}^T P\mathbf{x} + \mathbf{x}^T P\dot{\mathbf{x}} = \mathbf{x}^T \left(PA + A^T P\right)\mathbf{x} = -\mathbf{x}^{\mathbf{T}} Q\mathbf{x}, \tag{2.1.5}$$

where Q is defined as

$$Q = -\left(PA + A^T P\right). \tag{2.1.6}$$

Now let's look at what happens if Q is also positive definite. In this case, we have $dV/dt < 0$ as long as $\mathbf{x} \neq \mathbf{0}$. ($V$ can only decrease.) But we also have $V > 0$ as long as $\mathbf{x} \neq \mathbf{0}$. This implies that $\lim_{t\to\infty} V(\mathbf{x}(t)) = 0$. In other words, V eventually becomes zero, and so does $\mathbf{x}$.

The conclusion? Let's suppose we can find any positive definite matrices P and Q that satisfy the so-called **Lyapunov equation** (2.1.6). In this case, $\mathbf{x}$ will converge to $\mathbf{0}$ and the system is thus asympotically stable.

2.1.5 Interval stability

Let's again examine the polynomial

$$p(\lambda) = a_0 + a_1\lambda + a_2\lambda^2 + \ldots + \lambda^n. \tag{2.1.7}$$

(Note that this time we have divided out a_n. This of course does not change the roots of the equation.) Sometimes, the coefficients a_i are not exactly known. Instead, we only know the interval $[a_i^-, a_i^+]$ in which the coefficient a_i must lie. We usually denote this by the **interval polynomial**

$$p(\lambda, [\mathbf{a}^-, \mathbf{a}^+]) = [a_0^-, a_0^+] + [a_1^-, a_1^+]\lambda + [a_2^-, a_2^+]\lambda^2 + \ldots + [a_{n-1}^-, a_{n-1}^+]\lambda^{n-1} + \lambda^n. \tag{2.1.8}$$

Associated to this polynomial are four polynomials, called the **Kharitonov polynomials**. They are

$$\begin{aligned}
p_{--}(\lambda) &= a_0^- + a_1^-\lambda + a_2^+\lambda^2 + a_3^+\lambda^3 + a_4^-\lambda^4 + a_5^-\lambda^5 + a_6^+\lambda^6 + \ldots + \lambda^n, && (2.1.9)\\
p_{+-}(\lambda) &= a_0^+ + a_1^-\lambda + a_2^-\lambda^2 + a_3^+\lambda^3 + a_4^+\lambda^4 + a_5^-\lambda^5 + a_6^-\lambda^6 + \ldots + \lambda^n, && (2.1.10)\\
p_{-+}(\lambda) &= a_0^- + a_1^+\lambda + a_2^+\lambda^2 + a_3^-\lambda^3 + a_4^-\lambda^4 + a_5^+\lambda^5 + a_6^+\lambda^6 + \ldots + \lambda^n, && (2.1.11)\\
p_{++}(\lambda) &= a_0^+ + a_1^+\lambda + a_2^-\lambda^2 + a_3^-\lambda^3 + a_4^+\lambda^4 + a_5^+\lambda^5 + a_6^-\lambda^6 + \ldots + \lambda^n. && (2.1.12)
\end{aligned}$$

Note that all these polynomials alternately have two pluses and two minuses. The subscripts of p simply indicate the start of the series.

The Karitonov polynomials are important when determining the stability of the system. Let's suppose that all eigenvalues of all these four polynomials p have a negative real part. Only in this case will all eigenvalues of every polynomial $p(\lambda, \mathbf{a})$ (with parameter vector $\mathbf{a} \in [\mathbf{a}^-, \mathbf{a}^+]$) have a negative real part. In other words, if the Karitonov polynomials are asymptotically stable, then every polynomial within the interval is asymptotically stable, and vice versa.

2.1.6 Input-output stability

Let's examine a system in state-space form. So,

$$\begin{aligned}
\dot{\mathbf{x}}(t) &= A(t)\mathbf{x}(t) + B(t)\mathbf{u}(t), && (2.1.13)\\
\mathbf{y}(t) &= C(t)\mathbf{x}(t) + D(t)\mathbf{u}(t). && (2.1.14)
\end{aligned}$$

Let's suppose that the input $\mathbf{u}(t)$ is bounded. (There is a constant c such that $||\mathbf{u}(t)|| \leq c$ for all t.) If the output $\mathbf{y}(t)$ also remains bounded ($||\mathbf{y}(t)|| \leq k$ for some k), then the system is said to be **BIBO stable**. (BIBO stands for bounded input, bounded output.)

BIBO stability is often also referred to as **external stability**. (It concerns the external parts $\mathbf{u}$ and $\mathbf{y}$ of the system.) On the other hand, the stability of $\dot{\mathbf{x}} = A\mathbf{x}$ is referred to as **internal stability**. A system that is internally stable is always externally stable as well. But the opposite is not always true.

2.2 Controllability

2.2.1 Definitions

Next to stability, also controllability is an important aspect of systems. Let's examine a system (A, B, C, D). If both the initial state $\mathbf{x_0}$ and the input $\mathbf{u}(t)$ at any time t are given, then the solution $\mathbf{x}(t, \mathbf{x_0}, \mathbf{u})$ is fully determined.

Now let's suppose that we have a system (A, B, C, D) and a desired state $\mathbf{x_1}$. We say that the system is **controllable** if there is an input $\mathbf{u}(t)$ such that the state of the system $\mathbf{x}(t, \mathbf{x_0}, \mathbf{u})$ equals $\mathbf{x_1}$ at some finite time $t_1 > 0$. (This must hold for every desired state $\mathbf{x_1}$.)

Sometimes it is assumed that $\mathbf{x_1} = \mathbf{0}$ with $\mathbf{x_0} \neq \mathbf{0}$: we want to achieve/keep a zero state. In this case, we are talking about **null-controllability**. On the other hand, sometimes it is assumed that $\mathbf{x_1} \neq \mathbf{0}$ but $\mathbf{x_0} = \mathbf{0}$: we want to reach a certain point $\mathbf{x_1}$ starting from zero. In this case, we are talking about **reachability**. However, it can be shown that the controllability-problem, the null-controllability-problem and the reachability problem are equivalent. (If a system is controllable, it is also reachable and vice verse.) So, we will only consider ourselves with the controllability problem.

2.2.2 The controllability matrix

An important aspect in controllability is the **controllability matrix** R. It is defined as the $n \times nm$ matrix

$$R = \begin{bmatrix} B & AB & A^2B & \dots & A^{n-1}B \end{bmatrix}. \tag{2.2.1}$$

Now let's look at the image of R, denoted by im R. (The image of R means the column space of R. It consists of all vectors $\mathbf{b}$ that can be reached by linear combinations of the columns of R: $R\mathbf{a} = \mathbf{b}$.) It can be shown that this image consists of all reachable vectors. In other words, all linear combinations of R are reachable. For this reason, im R is also known as the **reachable subspace** or the **controllable subspace**.

Based on the above fact, we can find something very important. If the matrix R has rank n (meaning that all n rows of R are linearly independent), then the system (A, B) is controllable! This condition is known as the **rank condition** for controllability. (Note that we have denoted the system by (A, B). This is because C and D don't influence the controllability of the state.)

2.2.3 The modal form

If A is diagonalizable, we can apply a little trick. In this case, we can write $A = TDT^{-1}$. Now, let's define

$$\tilde{A} = D = T^{-1}AT, \qquad \tilde{B} = T^{-1}B, \qquad \tilde{C} = CT, \qquad \text{and} \qquad \tilde{D} = D. \tag{2.2.2}$$

If we also define $\tilde{\mathbf{x}} = T^{-1}\mathbf{x}$, then we can rewrite the state space representation to

$$\dot{\tilde{\mathbf{x}}} = \tilde{A}\tilde{\mathbf{x}} + \tilde{B}\mathbf{u} \qquad \text{and} \qquad \mathbf{y} = \tilde{C}\tilde{\mathbf{x}} + \tilde{D}\mathbf{u}. \tag{2.2.3}$$

This system has exactly the same properties as the old (A, B, C, D) system. (Only the state parameters have changed.) But now, we have a matrix A which is diagonal! For diagonal matrices, it's not hard to find the controllability matrix. So, solving this problem further won't be difficult.

2.2.4 Seperating noncontrollable parts

Let's suppose that we have a noncontrollable matrix. What do we do now? Well, there is a way in which we can separate the noncontrollable part from the controllable part. We'll examine that method now.

First, we find the matrix R. Let's denote the dimension of im R as $k = \text{rank } R < n$. We can find a basis $\{\mathbf{q_1}, \dots, \mathbf{q_k}\}$ of k linearly independent vectors $\mathbf{q_i}$ for the subspace im R. Let's add $(n - k)$ more vectors $\mathbf{q_{k+1}}, \dots, \mathbf{q_n}$ to this basis, such that we have n linearly independent vectors. We now put all these vectors in a matrix. So,

$$T = \begin{bmatrix} \mathbf{q_1} & \mathbf{q_2} & \dots & \mathbf{q_n} \end{bmatrix}. \tag{2.2.4}$$

Just like in the previous paragraph, we can define $\tilde{A} = T^{-1}AT$ and $\tilde{B} = T^{-1}B$. This again gives us an equivalent system. But we now have

$$\tilde{A} = \begin{bmatrix} \tilde{A}_{11} & \tilde{A}_{12} \\ 0 & \tilde{A}_{22} \end{bmatrix} \qquad \text{and} \qquad \tilde{B} = \begin{bmatrix} \tilde{B}_1 \\ 0 \end{bmatrix}. \tag{2.2.5}$$

In this equation, A_{11} has size $k \times k$. (The rest of the matrices are sized accordingly.) The controllability matrix of our new system is

$$\tilde{R} = TR = \begin{bmatrix} \tilde{B} & \tilde{A}\tilde{B} & \tilde{A}^2\tilde{B} & \dots & \tilde{A}^{n-1}\tilde{B} \end{bmatrix} = \begin{bmatrix} \tilde{B}_1 & \tilde{A}_{11}\tilde{B}_1 & \tilde{A}_{11}^2\tilde{B}_1 & \dots & \tilde{A}_{11}^{n-1}\tilde{B}_1 \\ 0 & 0 & 0 & \dots & 0 \end{bmatrix}. \tag{2.2.6}$$

So, we have split up our system into a controllable part and an uncontrollable part.

2.3 Observability

2.3.1 Definitions and theorems

Let's suppose that we have some system (A, B, C, D), of which we do not know the initial state $\mathbf{x_0}$. But we are observing the output $\mathbf{y}(t)$, and also the input $\mathbf{u}(t)$ is known. If, after some time t_1, we are always able to uniquely determine the initial state $\mathbf{x_0}$ of the system, then the system is called **observable**. This must hold for every input function $\mathbf{u}(t)$.

Just like controllability completely depends on A and B, so does observability completely depend on A and C. We therefore usually talk about the system (C, A). (Note that this time A is mentioned last.) Now, let's define the **observability matrix** W as

$$W = \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{bmatrix}. \tag{2.3.1}$$

The size of W is $np \times n$ (with p the height of C). It can be proven that the system (C, A) is observable if and only if rank $W = n$. (This condition is known as the **rank condition** for observability.) In other words, this is the case if all n columns of W are linearly independent.

Let's look at the kernel ker W of W. (The kernel is also known as the null-space: all $\mathbf{x}$ for which $W\mathbf{x} = \mathbf{0}$.) It can be shown that ker W is the **non-observable subspace**. In other words, any vectors $\mathbf{x_0}$ and $\mathbf{x_1}$ in the kernel of W (thus satisfying $W\mathbf{x_0} = W\mathbf{x_1} = \mathbf{0}$) can not be destinguished from each other by just looking at the output.

2.3.2 Seperating nonobservable parts

Previously, we have separated the noncontrollable part of a system. Now, we will separate the nonobservable part from a nonobservable system. We denote the dimension of ker W as $k = n - \text{rank } W$. We can find a basis $\{\mathbf{q_1}, \ldots, \mathbf{q_k}\}$ of k linearly independent vectors $\mathbf{q_i}$ for the subspace ker W. Let's again add $(n - k)$ more vectors $\mathbf{q_{k+1}}, \ldots, \mathbf{q_n}$ to this basis, such that we again have n linearly independent vectors. Once more, we put all these vectors in a matrix $T = [\mathbf{q_1} \;\; \mathbf{q_2} \;\; \ldots \;\; \mathbf{q_n}]$. This time we have

$$\bar{A} = T^{-1}AT = \begin{bmatrix} \bar{A}_{11} & \bar{A}_{12} \\ 0 & \bar{A}_{22} \end{bmatrix} \qquad \text{and} \qquad \bar{C} = CT = \begin{bmatrix} 0 & \bar{C}_2 \end{bmatrix}. \tag{2.3.2}$$

Again, A_{11} has size $k \times k$. (The rest of the matrices are sized accordingly.) The observability matrix of our new system now is

$$\bar{W} = WT = \begin{bmatrix} \bar{C} \\ \bar{C}\bar{A} \\ \bar{C}\bar{A}^2 \\ \vdots \\ \bar{C}\bar{A}^{n-1} \end{bmatrix} = \begin{bmatrix} 0 & \bar{C}_2 \\ 0 & \bar{C}_2\bar{A}_{22} \\ 0 & \bar{C}_2\bar{A}_{22}^2 \\ 0 & \vdots \\ 0 & \bar{C}_2\bar{A}_{22}^{n-1} \end{bmatrix}. \tag{2.3.3}$$

And we have split up the system into an observable part and a nonobservable part.

2.3.3 The Hautus test

There are plenty of other ways in which to check if a system is controllable/observable. Although we won't treat all of them, we will treat the **Hautus test**.

- The Hautus test for controllability states that the pair (A, B) is controllable if and only if, for all (possibly) complex s, we have
$$\text{rank } \begin{bmatrix} (sI - A) & B \end{bmatrix} = n. \tag{2.3.4}$$
- Similarly, the Hautus test for observability states that the pair (C, A) is controllable if and only if, for all (possibly) complex s, we have
$$\text{rank } \begin{bmatrix} (sI - A) \\ C \end{bmatrix} = n. \tag{2.3.5}$$

It is nice to note that $sI - A$ already has rank n for all s unequal to an eigenvalue λ_i. So, you only have to check the above matrices for values of s equal to one of the eigenvalues of A. That should save us some work. Finally, it is interesting to note that, if the pair (A, B) is controllable, then the pair (B^T, A^T) is observable and vice verse.

3. Feedback control

When controlling a system, the trick is to provide the right input. But what input will control the system? And how do we find it? That's what we'll look at in this chapter.

3.1 State feedback

3.1.1 Definitions

Let's suppose we're controlling a boat. When doing this, we use the output (the current heading) to determine the rudder position. In this case, the output is used to determine the input. When this is the case, we are dealing with a **feedback control** system, also known as a **closed-loop system**. When the input is independent of the output, then we have an **open-loop system**. In this chapter, we will examine closed-loop systems.

The important question is, how do we choose our input? Well, let's suppose that $C = I$ and $D = 0$. Thus, the output $\mathbf{y}(t)$ equals the state $\mathbf{x}(t)$. In this case, we can take $\mathbf{u}(t) = F\mathbf{x}(t)$, with F a matrix. This is called **state feedback**. The equation $\mathbf{u} = F\mathbf{x}$ is called a **control law**. In fact, this specific control law is known as a **static compensator**. (This is because F does not vary with time.)

3.1.2 Stabilizability

With the static compensator, our system equation turns into $\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u} = (A + BF)\mathbf{x}$. We should choose our F, such that the system $A + BF$ is stable. In other words, the eigenvalues of $A + BF$ should all have a negative real part. We call the system **stabilizable** if there is a matrix F, such that $A + BF$ is stable.

You may wonder, when is a system stabilizable? To find this out, we can use the **Hautus test** for stabilizability. This test states that the system (A, B) is stabilizable if and only if, for all (possibly) complex s with $\mathrm{Re}(s) \geq 0$, we have

$$\text{rank} \begin{bmatrix} (sI - A) & B \end{bmatrix} = n. \tag{3.1.1}$$

Again, note that the rank of the above matrix always equals n, when s is not an eigenvalue. We thus only need to check the above equation for values $s = \lambda_i$, where λ_i is (in this case) an unstable eigenvalue. (Also note that stable systems are stabilizible by default, since they don't have any unstable eigenvalues. Similarly, controllable systems are also always stabilizable.)

3.1.3 The pole-assignment theorem

Let's suppose that the system (A, B) is controllable. The **pole-assignment theorem** now states that we can actually choose the poles of $(A + BF)$ ourselves! This implies that the system is surely stabilizable. (We can simply choose poles with negative real parts.) All we need to do is choose F in the right way. But how do we choose F? That is, however, not very easy. But we will describe the process here for the single-input case.

The first step is to put the matrices A and B into so-called **controller (canonical) form**, being

$$\bar{A} = T^{-1}AT = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 1 \\ -p_0 & -p_1 & \cdots & -p_{n-2} & -p_{n-1} \end{bmatrix} \quad \text{and} \quad \bar{B} = T^{-1}B = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}. \tag{3.1.2}$$

In this equation, the coefficients $p_1, \ldots, p_{n-1}$ are the coefficients of the characteristic polynomial of A. In fact, we have

$$\det(\lambda I - A) = \det(\lambda I - \bar{A}) = p(\lambda) = \lambda^n + p_{n-1}\lambda^{n-1} + \ldots + p_1\lambda + p_0. \tag{3.1.3}$$

The only difficult part here is finding the right transformation matrix T. Luckily, T can be constructed. We simply have $T = [\mathbf{q_1} \;\; \mathbf{q_2} \;\; \ldots \;\; \mathbf{q_n}]$, where the vectors $\mathbf{q_1}, \ldots, \mathbf{q_n}$ are defined as

$$\begin{aligned} \mathbf{q_n} &= B, && (3.1.4)\\ \mathbf{q_{n-1}} &= AB + p_{n-1}B = A\mathbf{q_n} + p_{n-1}\mathbf{q_n}, && (3.1.5)\\ \mathbf{q_{n-2}} &= A^2B + p_{n-1}AB + p_{n-2}B = A\mathbf{q_1} + p_{n-2}\mathbf{q_n}, && (3.1.6)\\ &\vdots && (3.1.7)\\ \mathbf{q_1} &= A^{n-1}B + p_{n-1}A^{n-2}B + \ldots + p_1B = A\mathbf{q_2} + p_1\mathbf{q_n}. && (3.1.8) \end{aligned}$$

Once we have put A and B into the controller form, we can continue with step 2. Let's suppose that we want to have a certain set of poles. These poles then fully determine the characteristic equation of $A + BF$, being

$$\det(\lambda I - (A + BF)) = r(\lambda) = \lambda^n + r_{n-1}\lambda^{n-1} + \ldots + r_1\lambda + r_0. \tag{3.1.9}$$

Now, all we need to do is define

$$\bar{F} = \begin{bmatrix} p_0 - r_0 & p_1 - r_1 & \ldots & p_{n-1} - r_{n-1}. \end{bmatrix} \tag{3.1.10}$$

This immediately gives $\bar{A} + \bar{B}\bar{F}$ the required eigenvalues, and thus also $A + BF$. (Linear transformations don't change the eigenvalues of a matrix.) Once we know $\bar{F}$, finding F itself is not difficult. We just use $F = \bar{F}T^{-1}$.

3.2 The observer

3.2.1 Detectability

Previously, we have assumed that $C = I$. In other words, the state $\mathbf{x}(t)$ was known. But this is of course not always the case. What do we do if we don't know the state $\mathbf{x}$? One option is to approximate it, by using an **observer**. This observer uses the input $\mathbf{u}(t)$ and the output $\mathbf{y}(t)$ to make an estimate $\hat{\mathbf{x}}$ of the state $\mathbf{x}$. The observer can now be modeled as a separate system. This system has as output the estimated state $\hat{\mathbf{x}}$ of the system. We thus have

$$\dot{\hat{\mathbf{x}}} = P\hat{\mathbf{x}} + Q\mathbf{u} + K\mathbf{y} \qquad \text{and} \qquad \hat{\mathbf{x}} = I\hat{\mathbf{x}}. \tag{3.2.1}$$

The question remains, what should P, Q and K be? We use two simple rules to derive them. First of all, if we have a correct estimate of the state at some time, then we would like to keep the estimate correct. So, if $\mathbf{x} = \hat{\mathbf{x}}$, then we want to have

$$\frac{d}{dt}(\mathbf{x} - \hat{\mathbf{x}}) = \mathbf{0} = A\mathbf{x} + B\mathbf{u} - P\hat{\mathbf{x}} - Q\mathbf{u} - K\mathbf{y} = (A - KC)\mathbf{x} - P\hat{\mathbf{x}} + (B - Q)\mathbf{u}, \tag{3.2.2}$$

where we have used $\mathbf{y} = C\mathbf{x}$. The above equation should hold for every $\mathbf{u}(t)$. By using this fact, and by also using $\mathbf{x} = \hat{\mathbf{x}}$, we can find that

$$A - KC = P \qquad \text{and} \qquad B = Q. \tag{3.2.3}$$

Second, we would like the **error** $\mathbf{e} = \mathbf{x} - \hat{\mathbf{x}}$ to decrease to zero. The derivative of this error is given by

$$\dot{\mathbf{e}} = \frac{d}{dt}(\mathbf{x} - \hat{\mathbf{x}}) = (A - KC)(\mathbf{x} - \hat{\mathbf{x}}) = (A - KC)\,\mathbf{e}. \tag{3.2.4}$$

This means that, if $A - KC$ is asymptotically stable, then the error will converge to $\mathbf{0}$. The observer will thus be able to estimate the state $\mathbf{x}$. We therefore say that a system is **detectable** if there is a matrix K such that the eigenvalues of $A - KC$ all have negative real parts.

The question remains, when is a system detectable? Again, there is a test. A system is detectable if and only if, for all (possibly) complex s with $\mathrm{Re}(s) \geq 0$, we have

$$\text{rank} \begin{bmatrix} (sI - A) \\ C \end{bmatrix} = n. \tag{3.2.5}$$

Again, we only need to check the above equation for values $s = \lambda_i$, where λ_i is an unstable eigenvalue. (Also note that stable and observable systems are always detectable.)

3.2.2 Again the pole-assignment theorem

Let's suppose that (C, A) is observable. In this case, we can again select the poles of $A - KC$. In fact, for every polynomial $w(\lambda) = \lambda^n + w_{n-1}\lambda^{n-1} + \ldots + w_1\lambda + w_0$, there is a K such that $\det(\lambda I - (A - KC)) = w(\lambda)$. Thus, observable systems are always detectable. We just need to select K in the correct way. But how do we find K?

First, we remember that, if (C, A) is observable, then (A^T, C^T) is controllable. The system (A^T, C^T) is thus also stabilizable. There thus is an F such that $\det\left(\lambda I - (A^T + C^T F)\right) = w(\lambda)$. (To find F, use the set of steps discussed earlier in this chapter.) If we now take $K = -F^T$, then there is of course also a K such that

$$\det\left(\lambda I - (A^T - C^T K^T)\right) = \det(\lambda I - (A - KC)) = w(\lambda). \tag{3.2.6}$$

So, in this way, the correct matrix K can be chosen for the observer.

3.3 The dynamic compensator

3.3.1 Combining state feedback with an observer

Let's summarize matters. In the first part of this chapter, we saw how we can stabilize a system, if we know the state. In the second part, we saw how we can estimate the state. What if we use the estimate of the state to stabilize the system? We simply connect the observer to the static compensator. Doing this will give us a **dynamic compensator** (also known as a **dynamic controller**). This dynamic compensator can be seen as a separate system. Its input will be the output $\mathbf{y}$ of the original system, while its output is the input $\mathbf{u}$ to the original system. The dynamic compensator can thus be described by

$$\dot{\hat{\mathbf{x}}} = (A + BF - KC)\hat{\mathbf{x}} + K\mathbf{y}, \tag{3.3.1}$$

$$\mathbf{u} = F\hat{\mathbf{x}}. \tag{3.3.2}$$

By connecting the dynamic compensator to the original system, we actually get a **combined system** without any inputs and outputs. This combined system can be described by $\dot{\mathbf{x}}_\mathbf{c} = A_c\mathbf{x}_\mathbf{c}$, where

$$\mathbf{x_c} = \begin{bmatrix} \mathbf{x} \\ \hat{\mathbf{x}} \end{bmatrix} \qquad \text{and} \qquad A_c = \begin{bmatrix} A & BF \\ KC & A + BF - KC \end{bmatrix}. \tag{3.3.3}$$

3.3.2 Stability of the combined system

Let's ask ourselves, is the combined system stable? To answer that question, we need to examine the eigenvalues of the matrix A_c. These are given by $\det(\lambda I - A_c)$. Interestingly enough, it can be shown that

$$\begin{bmatrix} \dot{\mathbf{x}} \\ \dot{\mathbf{e}} \end{bmatrix} = \begin{bmatrix} A + BF & -BF \\ 0 & A - KC \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{e} \end{bmatrix}. \tag{3.3.4}$$

This implies that

$$\det(\lambda I - A_c) = \det(\lambda I - (A + BF)) \cdot \det(\lambda I - (A - KC)). \tag{3.3.5}$$

In other words, the set of eigenvalues of A_c is the union of the set of eigenvalues of the state feedback $A + BF$ and the set of eigenvalues of the observer $A - KC$. This means that the state feedback and the observer can be designed independently. (This is known as the **separation principle**.) It also means that, if the system is both stabilizable and detectable, then there are F and K such that A_c is stable.

4. Transfer functions

Systems theory is all about input and output. One common way to relate them to each other, is by using transfer functions. In this chapter, we'll go more into depth on what these functions really are.

4.1 Transfer function basics

4.1.1 The Laplace transform

When we are dealing with systems in the normal way, we say we're working with the system in the time domain. There is an alternative: the **Laplace domain**. To use this domain, we have to make use of the **Laplace transform**. Let's examine a function $f(t)$ in the time domain. Its Laplace transform $F(s)$ is given by

$$F(s) = \mathscr{L}(f(t)) = \int_0^\infty f(t)e^{-st}dt. \tag{4.1.1}$$

Now let's examine a system with impulse response matrix $G(t-\tau)$. If we assume that the input $\mathbf{u}(\tau)$ is zero for $\tau < 0$, then we have

$$\mathbf{y}(t) = \int_{-\infty}^{t} G(t-\tau)\mathbf{u}(\tau)d\tau = \int_0^t G(t-\tau)\mathbf{u}(\tau)d\tau. \tag{4.1.2}$$

The **convolution theorem** now states that this equation is equivalent to $Y(s) = H(s)U(s)$. Here, $U(s)$ and $Y(s)$ are the Laplace transforms of $\mathbf{u}(t)$ and $\mathbf{y}(t)$. (Transforming vectors and matrices is simply done element-wise.) Also, $H(s)$ is the **transfer matrix** of the system.

So, to find $H(s)$, we can simply use $H(s) = \mathscr{L}(G(t)) = \mathscr{L}(Ce^{At}B)$. But there is also another way to find $H(s)$. For that, we have to use the property that, for any $\mathbf{x}(t)$, we have $\mathscr{L}(\dot{x}) = sX(s) - \mathbf{x_0}$. Applying this to the state space equation of a linear time-invariant system gives

$$sX(s) - \mathbf{x_0} = AX(s) + BU(s) \qquad \Rightarrow \qquad X(s) = (sI - A)^{-1}\mathbf{x_0} + (sI - A)^{-1}BU(s). \tag{4.1.3}$$

If we assume that $\mathbf{x_0} = \mathbf{0}$ and $D = 0$, then

$$H(s) = \mathscr{L}(Ce^{At}B) = C(sI - A)^{-1}B. \tag{4.1.4}$$

The matrix function $(sI - A)^{-1} = \mathscr{L}(e^{At})$ is known as the **resolvente** of the matrix A.

4.1.2 Connecting series

Let's suppose that we want to connect two systems. When we use state space representations, this will give us a lot of work. This is where the Laplace transform saves time. Because connecting systems is easy with the Laplace transform.

Let's suppose that we want to connect two systems $H_1(s)$ and $H_2(s)$. What will the resulting transfer matrix $H(s)$ be? If we connect the systems in **series** (figure 4.1, left), then we will have $H(s) = H_2(s)H_1(s)$. (Note that the order of multiplication is important, since we are using matrices.) If we connect them in **parallel** (figure 4.1, middle), then we have $H(s) = H_1(s) + H_2(s)$. And if we connect them in a **feedback connection** pattern (figure 4.1, right), we will get $H(s) = (I + H_1(s)H_2(s))^{-1}H_1(s)$.

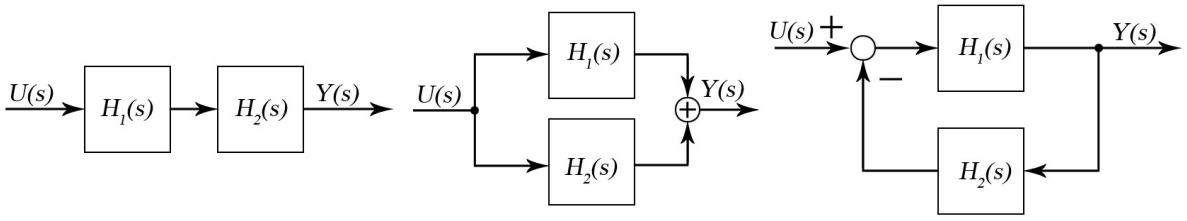

Figure 4.1: Three ways to connect two systems.

4.1.3 Transfer function properties

We can assign several properties to transfer functions. We start with the property of rationality. We say that a function $H(s)$ is a **rational function** if it can be written as

$$H(s) = \frac{q(s)}{p(s)} = \frac{q_k s^k + q_{k-1} s^{k-1} + \ldots + q_1 s + q_0}{s^n + p_{n-1} s^{n-1} + \ldots + p_1 s + p_0}. \tag{4.1.5}$$

In other words, $H(s)$ can be written as the ratio of two polynomials. Let's take a closer look at the numerator $q(s)$ and the denominator $p(s)$. The roots of $q(s)$ are called the **zeroes** of the function, whereas the roots of $p(s)$ are called the **poles**. We also define the **degree** of a polynomial as the highest power of the variable s. Thus, in the above equation we have $\deg(q) = k$ and $\deg(p) = n$.

We say that a transfer function $H(s)$ is **proper** if it has $\deg(q) \leq \deg(p)$. (This implies that $\lim_{s\to\infty} H(s) =$ constant.) And a function is **strictly proper** if $\deg(q) < \deg(p)$. (We now have $\lim_{s\to\infty} H(s) = 0$.) We can always transform a linear time-invariant state space system to a transfer function. If we do this, we always get a proper transfer function $H(s)$. And, if $D = 0$, we even get a strictly proper transfer function.

4.2 State space realizations

4.2.1 The controller form

Let's examine a proper transfer function $H(s)$. We can always transform this transfer function to a state space representation. We will illustrate this process for a single-input single-output system $h(s)$. (Later on, we examine multi-dimensional systems.) First, let's assume that $h(s)$ is strictly proper, and thus $k < n$. This implies that $D = 0$. We can now find the so-called **controller form** (also known as the **standard controllable realization**). It is given by

$$A = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 1 \\ -p_0 & -p_1 & \cdots & -p_{n-2} & -p_{n-1} \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}, \quad \text{and} \quad C = \begin{bmatrix} q_0 & q_1 & \cdots & q_{n-2} & q_{n-1} \end{bmatrix}. \tag{4.2.1}$$

(The reason why they call this the controllable realization is because it is easy to see that the system is controllable.) You may still wonder, what if $k = n$? In this case, we first have to rewrite $h(s)$ as

$$h(s) = q_n + \frac{\bar{q}(s)}{p(s)} = q_n + \frac{(q_{k-1} - q_k p_{k-1}) s^{k-1} + \ldots + (q_1 - q_k p_1) s + (q_0 - q_n p_0)}{s^n + p_{n-1} s^{n-1} + \ldots + p_1 s + p_0}. \tag{4.2.2}$$

We now take $D = q_n$ and use the strictly proper function $\bar{q}(s)/p(s)$ to find A, B and C in the normal way. Do note though, that a system can always be represented by infinitely many different state space

representations. We have just found one of them.

4.2.2 Other forms

Next to the standard controllable realization, we also have a **standard observable realization**. To find it, we have exactly the same process as previously. But now, we have the matrices

$$A = \begin{bmatrix} 0 & 0 & \cdots & 0 & -p_0 \\ 1 & 0 & \cdots & 0 & -p_1 \\ 0 & 1 & \ddots & \vdots & \vdots \\ \vdots & \vdots & \ddots & 0 & -p_{n-2} \\ 0 & 0 & \cdots & 1 & -p_{n-1} \end{bmatrix}, \quad B = \begin{bmatrix} q_0 \\ q_1 \\ \vdots \\ q_{n-2} \\ q_{n-1} \end{bmatrix}, \quad \text{and} \quad C = \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 \end{bmatrix}. \tag{4.2.3}$$

The last representation that we will discuss is the **diagonal realization**. To find it, we first have to write

$$h(s) = D + \frac{q(s)}{p(s)} = D + \frac{\gamma_1}{s - a_1} + \frac{\gamma_2}{s - a_2} + \ldots + \frac{\gamma_n}{s - a_n}. \tag{4.2.4}$$

This is possible as long as $h(s)$ has n different poles. If this is the case, then we will find that

$$A = \begin{bmatrix} a_1 & 0 & \cdots & 0 \\ 0 & a_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_n \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}, \quad \text{and} \quad C = \begin{bmatrix} \gamma_1 & \gamma_2 & \cdots & \gamma_n \end{bmatrix}. \tag{4.2.5}$$

You may wonder, what if some factor $(s - a_i)$ occurs more than once in $p(s)$? In this case, A won't be a diagonal matrix. Instead, ones will appear in A and disappear in B. For example, let's examine the function

$$h(s) = \frac{\gamma}{s - 1} + \frac{\delta}{(s - a)^2} + \frac{\alpha}{(s - a)^3}. \tag{4.2.6}$$

This will give the state space matrices

$$A = \begin{bmatrix} a & 1 & 0 \\ 0 & a & 1 \\ 0 & 0 & a \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \quad \text{and} \quad C = \begin{bmatrix} \alpha & \delta & \gamma \end{bmatrix}. \tag{4.2.7}$$

4.2.3 Multiple inputs/outputs

We can also turn systems with multiple inputs/outputs into a state space representation. Let's examine the $p \times m$ transfer matrix $H(s)$. We can write it as

$$H(s) = D + \frac{1}{p(s)} Q(s) = D + \frac{1}{p(s)} \left(Q_{n-1} s^{n-1} + Q_{n-2} s^{n-2} + \ldots + Q_1 s + Q_0 \right). \tag{4.2.8}$$

In this equation, $Q(s)$ denotes a $p \times m$ transfer matrix and Q_i (with $i = 0, 1, \ldots, n-1$) denotes a constant $p \times m$ matrix. The matrices A, B and C are now given by

$$A = \begin{bmatrix} 0 & I & 0 & \cdots & 0 \\ 0 & 0 & I & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & I \\ -p_0 I & -p_1 I & \cdots & -p_{n-2} I & -p_{n-1} I \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ I \end{bmatrix}, \quad \text{and} \quad C = \begin{bmatrix} Q_0 & Q_1 & \cdots & Q_{k-2} & Q_{k-1} \end{bmatrix}. \tag{4.2.9}$$

A now is a matrix of size $mn \times mn$. Note that the above matrices are in some sort of controller form. In a similar way, we can put a multiple input/output system in an observable realization.

4.2.4 The McMillan degree

Now we know how to write a multiple input/output system in state space form. But we did need nm state variables for that. We could ask ourselves, can't we use less state variables? Well, that actually depends on the transfer function $H(s)$. Let's call the minimum amount of state variables needed z. This number z is also known as the **McMillan degree** of $H(s)$. To find it, we can examine

$$H(s) = L_0 + L_1 s^{-1} + L_2 s^{-2} + \ldots. \tag{4.2.10}$$

The McMillan degree z of $H(s)$ is now given by the rank of the matrix L (so $z = \text{rank}(L)$), where

$$L = \begin{bmatrix} L_1 & L_2 & \cdots & L_r \\ L_2 & L_3 & \cdots & L_{r+1} \\ \vdots & \vdots & \ddots & \vdots \\ L_r & L_{r+1} & \cdots & L_{2r-1} \end{bmatrix}. \tag{4.2.11}$$

The parameter r is the degree of the least common multiple of all denominators of $H(s)$.

When we use the minimum amount of state variables to create a state space representation, then we say that we have a **minimum realization**. If we have a minimum realization system, then it can be shown that the system is always both controllable and observable. Equivalently, if the system is either not controllable or not observable (or both), then it is not a minimum realization system.