3 Nonlinear Control Design for a Three-Tank System

M.Sc. Laboratory Advanced Control (WS 22/23)

Prof. Dr.–Ing. habil. Thomas Meurer, Chair of Automation and Control

The goal of this experiment is the controller design for nonlinear multivariable systems using exact input-output linearization.

Two different examples are considered. Exact input-output linearization is first considered for an introductory example incorporating the software tool MATLAB / SIMULINK, where the manipulation of the algebraic expressions can be done using the SYMBOLIC MATH TOOLBOX. This nonlinear control design method is furthermore applied to control the three-tank system shown in Figure 3.1.

For the first part of this lab, please work on the exercises that are given below. Answers to questions are to be answered in MATLAB.

Fig. 3.1: Three-tank system

For any questions regarding these experiments, please contact

- Denis Ratke (dera@tf.uni-kiel.de).

3.1 Lie-Algebra

The design of a nonlinear controller using exact input-output linearization requires a frequent use of algebraic operations from the *Lie algebra* such as *Lie derivative* and the *Lie bracket*. Therefore, the implementation of the operations as functions facilitates computer-aided controller design. In the following a brief description of the *Lie derivative* and the *Lie brackets* is presented by means of nonlinear SISO-system

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{g}(\boldsymbol{x})u, \quad && t > 0, \qquad \boldsymbol{x}(0) = \boldsymbol{x}_0 \\ y &= h(\boldsymbol{x}), && t \geq 0, \end{aligned} \tag{3.1}$$

where $\boldsymbol{f}(\boldsymbol{x}) \in \mathbb{R}^n$ is the drift vector field, $\boldsymbol{g}(\boldsymbol{x}) \in \mathbb{R}^n$ the input vector field and $h(\boldsymbol{x}) \in \mathbb{R}$ the output function. For detailed explanation of the *Lie derivative* and the *Lie brackets* see [3, Chapter 3].

3.1.1 Lie derivative

The *Lie derivative*

$$L_{\boldsymbol{f}} h(\boldsymbol{x}) = \sum_{j=1}^{n} \frac{\partial h}{\partial x_j} f_j(\boldsymbol{x}) = \frac{\partial h}{\partial \boldsymbol{x}} \boldsymbol{f}(\boldsymbol{x}), \tag{3.2a}$$

is the directional derivative of a smooth function $h(\boldsymbol{x}) : \mathcal{M} \to \mathbb{R}$ in the direction of the vector field $\boldsymbol{f}(\boldsymbol{x}) \in \mathcal{T}\mathcal{M}$ at point $\boldsymbol{x} = [x_1, \ldots, x_n]^T$, with the relationships

$$L_{\boldsymbol{f}}^k h = L_{\boldsymbol{f}}(L_{\boldsymbol{f}}^{k-1} h), \qquad L_{\boldsymbol{f}}^0 h = h \tag{3.3a}$$

$$L_{\boldsymbol{g}} L_{\boldsymbol{f}} h = \frac{\partial L_{\boldsymbol{f}} h}{\partial \boldsymbol{x}} \boldsymbol{g}, \tag{3.3b}$$

hold true for given vector field $\boldsymbol{g}(\boldsymbol{x}) \in \mathcal{T}\mathcal{M}$.

3.1.2 Lie bracket

The *Lie bracket*

$$[\boldsymbol{f}, \boldsymbol{g}] = \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}} \boldsymbol{f} - \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}} \boldsymbol{g}. \tag{3.4a}$$

defines the rate of change of a vector field $\boldsymbol{f}(\boldsymbol{x}) \in \mathcal{T}\mathcal{M}$ along the integral curve of another vector field $\boldsymbol{g}(\boldsymbol{x}) \in \mathcal{T}\mathcal{M}$. Another notation of the *Lie bracket* is given by $\mathrm{ad}_{\boldsymbol{f}}$-operator

$$\mathrm{ad}_{\boldsymbol{f}} \boldsymbol{g} = [\boldsymbol{f}, \boldsymbol{g}] \tag{3.5a}$$

$$\mathrm{ad}_{\boldsymbol{f}}^k \boldsymbol{g} = [\boldsymbol{f}, \mathrm{ad}_{\boldsymbol{f}}^{k-1} \boldsymbol{g}], \qquad \mathrm{ad}_{\boldsymbol{f}}^0 \boldsymbol{g} = \boldsymbol{g}. \tag{3.5b}$$

Exercise 3.1. *Consider the nonlinear SISO system*

$$\dot{\boldsymbol{x}}(t) = \begin{bmatrix} -x_3^2 \\ x_1 x_2 \\ x_2 \end{bmatrix} + \begin{bmatrix} x_2 \\ 1 \\ 0 \end{bmatrix} u, \qquad t > 0, \qquad \boldsymbol{x}(0) = \boldsymbol{x}_0 \in \mathcal{M} \tag{3.6a}$$

$$\boldsymbol{y} = h(\boldsymbol{x}) = x_3, \qquad t \geq 0. \tag{3.6b}$$

Solve the following tasks using MATLAB *in conjuction with* SYMBOLIC MATH TOOLBOX.

~~(i)~~ *Implement a function* `Lfh(f, h, x, k)` *that calculate the* `k`*-th Lie derivative of a smooth function* `h` *along a vector field* `f` *in local coordinates* `x`.

~~(ii)~~ *Implement a function* `LgLfh(f, g, h, x, k)` *which takes the* `k`*-th Lie derivative of the function* `h` *and vector field* `f` *and calculate a Lie derivative along the vector field* `g` *in local coordinates* `x`.

~~(iii)~~ *Implement a function* `adfg(f, g, x, k)` *that calculate the* `k`*-th Lie bracket of two vector fields* `f` *and* `g` *in local coordinates* `x`.

~~(iv)~~ *Test the functions implemented in task* $(i)-(iii)$ *using the SISO-system 3.6. Therefore:*

(a) *Determine the relative degree* `r` *of the system 3.6 using the functions implemented in* (i) *and* (ii).

(b) *Determine the Lie bracket of the vector fields* $\boldsymbol{f}$ *and* $\boldsymbol{g}$ *using the function implemented in* (iii) *for* $k = 0, 1, 2$.

Remark 3.1

The short definition of the relative degree r of the SISO-system is given by

$$\begin{aligned} L_{\boldsymbol{g}} L_{\boldsymbol{f}}^k h(\boldsymbol{x}) &= 0, \text{ for } k = 0, \dots, r-2 \\ L_{\boldsymbol{g}} L_{\boldsymbol{f}}^{r-1} h(\boldsymbol{x}) &\neq 0 \end{aligned}$$

3.2 Exact Input-Output Linearization

The basic idea of exact input-output linearization is to determine a diffeomorphic coordinate transformation for a nonlinear multivariable system of the form

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{f}(\boldsymbol{x}) + \sum_{j=1}^{m} \boldsymbol{g}_j(\boldsymbol{x}) u_j, \quad && t > 0, \qquad \boldsymbol{x}(0) = \boldsymbol{x}_0 \\ \boldsymbol{y} &= \boldsymbol{h}(\boldsymbol{x}), && t \geq 0, \end{aligned} \tag{3.8}$$

to decouple input and output and to stabilize the input-output behaviour using linear techniques due to feedback compensation. Detailed derivations of the exact input-output linearization for both single and multivariable cases can be found in [3, Chapter 4].

In the following, $\boldsymbol{f}(\boldsymbol{x}) \in \mathbb{R}^n$ is called drift vector field and $\boldsymbol{g}_j(\boldsymbol{x}) \in \mathbb{R}^n$ are called input vector fields. The vector of functions $\boldsymbol{h}(\boldsymbol{x})$ represents a mapping of the manifold $\mathcal{M}$ to the space $\mathbb{R}^p$. It is further assumed that the input and output dimensions are identical, i.e. $\dim(\boldsymbol{u}) = \dim(\boldsymbol{y}) = m$.

The method of exact input-output linearization can be divided into several steps. First, the so-called relative degree is determined. For $m = 1$ the relative degree r is a scalar and specifies the number of time derivatives of the system output y until the input u appears explicitly for the first time, i.e.

$$\begin{aligned} L_{\boldsymbol{g}} L_{\boldsymbol{f}}^k h(\boldsymbol{x}) &= 0 \quad \text{for} \quad k = 0, \dots, r-2 \\ L_{\boldsymbol{g}} L_{\boldsymbol{f}}^{r-1} h(\boldsymbol{x}) &\neq 0. \end{aligned} \tag{3.9}$$

For $m > 1$, the relative degree becomes a vector $(r_1, \ldots, r_m)$ and describes for each system output the number of time derivatives until any input appears for the first time. The system (3.8) has the vectorial relative degree $(r_1, \ldots, r_m)$ at a point $\overline{\boldsymbol{x}} \in U$ if

$$L_{\boldsymbol{g}_j} L_{\boldsymbol{f}}^k h_i(\boldsymbol{x}) = 0, \quad i, j = 1, \ldots, m \text{ for } k = 0, 1, \ldots, r_i - 2 \text{ and all } \boldsymbol{x} \in U \tag{3.10}$$

and the decoupling matrix

$$B(\boldsymbol{x}) = \begin{bmatrix} L_{\boldsymbol{g}_1} L_{\boldsymbol{f}}^{r_1-1} h_1(\boldsymbol{x}) & \cdots & L_{\boldsymbol{g}_m} L_{\boldsymbol{f}}^{r_1-1} h_1(\boldsymbol{x}) \\ \vdots & & \vdots \\ L_{\boldsymbol{g}_1} L_{\boldsymbol{f}}^{r_m-1} h_m(\boldsymbol{x}) & \cdots & L_{\boldsymbol{g}_m} L_{\boldsymbol{f}}^{r_m-1} h_m(\boldsymbol{x}) \end{bmatrix} \tag{3.11}$$

is regular at the point $\boldsymbol{x} = \overline{\boldsymbol{x}}$. This results in the following representation

$$\begin{bmatrix} y_1^{(r_1)} \\ \vdots \\ y_1^{(r_m)} \end{bmatrix} = \underbrace{\begin{bmatrix} L_{\boldsymbol{f}}^{r_1\cancel{-1}} h_1(\boldsymbol{x}) \\ \vdots \\ L_{\boldsymbol{f}}^{r_m\cancel{-1}} h_m(\boldsymbol{x}) \end{bmatrix}}_{\boldsymbol{a}(x)} + \underbrace{\begin{bmatrix} L_{\boldsymbol{g}_1} L_{\boldsymbol{f}}^{r_1-1} h_1(\boldsymbol{x}) & \cdots & L_{\boldsymbol{g}_m} L_{\boldsymbol{f}}^{r_1-1} h_1(\boldsymbol{x}) \\ \vdots & & \vdots \\ L_{\boldsymbol{g}_1} L_{\boldsymbol{f}}^{r_m-1} h_m(\boldsymbol{x}) & \cdots & L_{\boldsymbol{g}_m} L_{\boldsymbol{f}}^{r_m-1} h_m(\boldsymbol{x}) \end{bmatrix}}_{B(\boldsymbol{x})} \boldsymbol{u}, \tag{3.12}$$

which allows us to decouple the input-output behaviour if the decoupling matrix $B(\boldsymbol{x})$ is regular. In the second step a state transformation is determined, which converts the original system (3.8) into the Byrnes-Isidori normal form. Under the assumption that

$$\sum_{j=1}^{m} r_j = r < n, \tag{3.13}$$

there exist $(n - r)$ functions $\Phi_{r+1}(\boldsymbol{x}), \ldots, \Phi_n(\boldsymbol{x})$ such that in an open neighbourhood U of the point $\overline{\boldsymbol{x}}$ the local diffeomorphism is given by

$$\boldsymbol{z} = \begin{bmatrix} \boldsymbol{\xi} \\ \boldsymbol{\eta} \end{bmatrix} = \begin{bmatrix} \xi_{1,1} \\ \vdots \\ \xi_{1,r_1} \\ \hline \vdots \\ \hline \xi_{m,1} \\ \vdots \\ \xi_{m,r_m} \\ \hline \eta_1 \\ \vdots \\ \eta_{n-r} \end{bmatrix} = \boldsymbol{\phi}(\boldsymbol{x}) = \begin{bmatrix} h_1(\boldsymbol{x}) \\ \vdots \\ L_{\boldsymbol{f}}^{r_1-1} h_1(\boldsymbol{x}) \\ \hline \vdots \\ \hline h_m(\boldsymbol{x}) \\ \vdots \\ L_{\boldsymbol{f}}^{r_m-1} h_m(\boldsymbol{x}) \\ \hline \Phi_{r+1}(\boldsymbol{x}) \\ \vdots \\ \Phi_n(\boldsymbol{x}) \end{bmatrix}. \tag{3.14}$$

It should be noted that the choice of the elements $\Phi_{r+1}(x)$ to $\Phi_n(x)$ is generally restricted by the requirement that $\boldsymbol{\Phi}(x)$ must be a diffeomorphism.

With (3.12) the Byrnes-Isidori normal is given by

$$\Sigma_1: \begin{cases} \dot{\xi}_{1,1} &= \xi_{1,2} \\ &\vdots \\ \dot{\xi}_{1,r_1-1} &= \xi_{1,r_1} \\ \dot{\xi}_{1,r_1} &= a_1(\boldsymbol{\xi},\boldsymbol{\eta}) + \boldsymbol{b}_1^T(\boldsymbol{\xi},\boldsymbol{\eta})\boldsymbol{u} \\ &\vdots \\ \dot{\xi}_{m,1} &= \xi_{m,2} \\ &\vdots \\ \dot{\xi}_{m,r_m-1} &= \xi_{m,r_m} \\ \dot{\xi}_{m,r_m} &= a_m(\boldsymbol{\xi},\boldsymbol{\eta}) + \boldsymbol{b}_m^T(\boldsymbol{\xi},\boldsymbol{\eta})\boldsymbol{u} \end{cases} \tag{3.15}$$

$$\Sigma_2: \begin{cases} \dot{\eta}_1 &= p_1(\boldsymbol{\xi},\boldsymbol{\eta}) + q_1^T(\boldsymbol{\xi},\boldsymbol{\eta})\boldsymbol{u} \\ &\vdots \\ \dot{\eta}_{n-r} &= p_{n-r}(\boldsymbol{\xi},\boldsymbol{\eta}) + q_{n-r}^T(\boldsymbol{\xi},\boldsymbol{\eta})\boldsymbol{u} \end{cases} \tag{3.16}$$

$$\boldsymbol{y} = \begin{bmatrix} \xi_{1,1} & \dots & \xi_{m,1} \end{bmatrix}^T, \tag{3.17}$$

with

$$\begin{aligned} a_j(\boldsymbol{\xi},\boldsymbol{\eta}) &= L_{\boldsymbol{f}}^{r_j} h_j(\boldsymbol{x}) \circ \boldsymbol{\Phi}^{-1}(\boldsymbol{z}), & b_{j,l}(\boldsymbol{\xi},\boldsymbol{\eta}) &= L_{g_l} L_{\boldsymbol{f}}^{r_j-1} h_j(\boldsymbol{x}) \circ \boldsymbol{\Phi}^{-1}(\boldsymbol{z}), & & j,l = 1,\dots,m, \\ p_k(\boldsymbol{\xi},\boldsymbol{\eta}) &= L_{\boldsymbol{f}} \Phi_{r+k}(\boldsymbol{x}) \circ \boldsymbol{\Phi}^{-1}(\boldsymbol{z}), & q_{k,l}(\boldsymbol{\xi},\boldsymbol{\eta}) &= L_{g_l} \Phi_{r+k}(\boldsymbol{x}) \circ \boldsymbol{\Phi}^{-1}(\boldsymbol{z}), & & \begin{cases} k = 1,\dots,n-r \\ l = 1,\dots,m \end{cases}. \end{aligned} \tag{3.18}$$

The subsystems Σ_1 and Σ_2 describe the input-output dynamics and the internal dynamics of the system (3.8). Furthermore the relation

$$B(\boldsymbol{\xi},\boldsymbol{\eta}) = \begin{bmatrix} \boldsymbol{b}_1^T(\boldsymbol{\xi},\boldsymbol{\eta}) \\ \vdots \\ \boldsymbol{b}_m^T(\boldsymbol{\xi},\boldsymbol{\eta}) \end{bmatrix} = B(\boldsymbol{x}) \circ \boldsymbol{\Phi}^{-1}(\boldsymbol{z}) \tag{3.19}$$

between the decoupling matrix $B(\boldsymbol{x})$ and the inhomogeneous terms in $\boldsymbol{u}$ of the Byrnes-Isidori normal form can be derived. Using the nonlinear input transformation

$$\boldsymbol{u} = B^{-1}(\boldsymbol{\xi},\boldsymbol{\eta})(-\boldsymbol{a}(\boldsymbol{\xi},\boldsymbol{\eta}) + \boldsymbol{v}) \tag{3.20}$$

the nonlinear system (3.8) is decoupled and transformed into a system with linear input-output behaviour, where the vector $\boldsymbol{v}$ represents the new input of the system. The input-output behaviour of the subsystem Σ_1 ($\boldsymbol{v}$ to $\boldsymbol{y}$) can be described by the transfer matrix

$$\mathbb{G}(s) = \operatorname{diag}\left\{\frac{1}{s^{r_j}}\right\}_{j=1,\dots,m}. \tag{3.21}$$

The stability of the resulting system depends on the stability of the internal dynamics Σ_2 and the zero dynamics. The zero dynamics describe the behaviour of (3.16) for $\boldsymbol{\xi} = \boldsymbol{0}$ and $\boldsymbol{v} = \boldsymbol{0}$ for all t. This corresponds to the internal dynamics, where the initial values $\boldsymbol{\eta}(0) = \boldsymbol{\eta}_0$ and $\boldsymbol{\xi}(0) = 0$ are chosen such that the output $\boldsymbol{y}$ is identical to zero for all time. With (3.16) and (3.20) it applies that

$$\dot{\boldsymbol{\eta}} = \boldsymbol{p}(\boldsymbol{\xi},\boldsymbol{\eta}) + Q(\boldsymbol{\xi},\boldsymbol{\eta})B^{-1}(\boldsymbol{\xi},\boldsymbol{\eta})(-\boldsymbol{a}(\boldsymbol{\xi},\boldsymbol{\eta}) + \boldsymbol{v}), \tag{3.22}$$

where it holds that

$$Q(\boldsymbol{\xi},\boldsymbol{\eta}) = \begin{bmatrix} \boldsymbol{q}_1^T(\boldsymbol{\xi},\boldsymbol{\eta}) \\ \vdots \\ \boldsymbol{q}_{n-r}^T(\boldsymbol{\xi},\boldsymbol{\eta}) \end{bmatrix}. \tag{3.23}$$

For the zero dynamics it follows that

$$\dot{\boldsymbol{\eta}} = \boldsymbol{p}(\boldsymbol{0},\boldsymbol{\eta}) - Q(\boldsymbol{0},\boldsymbol{\eta})B^{-1}(\boldsymbol{0},\boldsymbol{\eta})\boldsymbol{a}(\boldsymbol{0},\boldsymbol{\eta}). \tag{3.24}$$

Provided that the zero dynamics are asymptotically (exponentially) stable, the control law

$$\boldsymbol{u} = B^{-1}(\boldsymbol{x}) \begin{bmatrix} -L_{\boldsymbol{f}}^{r_1} h_1(\boldsymbol{x}) - \sum_{l=0}^{r_1-1} p_{1,l} L_{\boldsymbol{f}}^{l} h_1(\boldsymbol{x}) \\ \vdots \\ -L_{\boldsymbol{f}}^{r_m} h_m(\boldsymbol{x}) - \sum_{l=0}^{r_m-1} p_{m,l} L_{\boldsymbol{f}}^{l} h_m(\boldsymbol{x}) \end{bmatrix} \tag{3.25}$$

results in an asymptotically stable closed loop system, if the coefficients $p_{j,l}$ with $j = 1,\dots,m$ and $l = 0,\dots,r_j - 1$ are coefficients of a Hurwitz polynomial.

If the system is supposed to follow a given reference trajectory, the control law can be extended to

$$\boldsymbol{u} = B^{-1}(\boldsymbol{x}) \begin{bmatrix} -L_{\boldsymbol{f}}^{r_1} h_1(\boldsymbol{x}) + y_1^{*(r_1)} - \sum_{l=0}^{r_1-1} p_{1,l} \left(L_{\boldsymbol{f}}^{l} h_1(\boldsymbol{x}) - y_1^{*(l)}\right) \\ \vdots \\ -L_{\boldsymbol{f}}^{r_m} h_m(\boldsymbol{x}) + y_m^{*(r_m)} - \sum_{l=0}^{r_m-1} p_{m,l} \left(L_{\boldsymbol{f}}^{l} h_m(\boldsymbol{x}) - y_m^{*(l)}\right) \end{bmatrix}, \tag{3.26}$$

where $\boldsymbol{y}^*(t) = \left[y_1^*(t),\dots,y_m^*(t)\right]$ represents the vector of the output reference trajectory, where all elements must adhere to the differentiability requirement $y_j^* \in C^{r_j}(\mathbb{R})$. With this control law the system output will asymptotically converge towards $\boldsymbol{y}^*$.

Exercise 3.2. *Consider the nonlinear multivariable input affine system*

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \end{bmatrix} = \begin{bmatrix} -x_1 + x_2^2 \\ x_1 x_3 + x_4 \\ -x_1 + x_3 \\ x_5 \\ -x_5 + x_3^2 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \\ x_2 \end{bmatrix} u_1 + \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} u_2, \tag{3.27a}$$

$$\boldsymbol{y} = \begin{bmatrix} x_3 \\ x_2 \end{bmatrix}. \tag{3.27b}$$

Solve the following tasks using a MATLAB *in conjunction with the* SYMBOLIC MATH TOOLBOX.

(i) *Determine the vectorial relative degree of the system* (3.27).

(ii) *Examine the internal dynamics and make a statement about the stability of the zero dynamics.*

(iii) *Design an output control with the method of exact input-output linearization. Use the reference trajectory*

$$\boldsymbol{y}^* = \begin{bmatrix} \left(\frac{3t^2}{T^2} - \frac{2t^3}{T^3}\right)(y_{1T} - y_{10}) + y_{10} \\ \left(\frac{3t^2}{T^2} - \frac{2t^3}{T^3}\right)(y_{2T} - y_{20}) + y_{20} \end{bmatrix} \quad \textit{with} \quad y_{10} = y_{20} = 0,\ y_{1T} = 2,\ y_{2T} = 3,\ T = 10. \tag{3.28}$$

(iv) *Test your results using a* MATLAB / SIMULINK *simulation.*

Remark 3.2

- Implement the model using a `Matlab-Function` and an integrator.
- Use a `Matlab-Function` to implement the control law.
- With *matlabFunctionBlock('blockname', symexpr)* provide MATLAB a function to convert the symbolic expression to MATLAB-FUNCTION block.

3.3 Three-Tank System

In the following, the three-tank system is described in more detail and the arising physical effects are explained.

3.3.1 Setup and Components

The system to be controlled is the laboratory setup DTS 200 three-tank system of Amira GmbH. The system is shown schematically in Figure 3.2. The three-tank system consists of the following components:

- A water depot
- Three cylindrical water tanks, hereinafter referred to as tank 1, 2 and 3
- Six valves

- Two feed pumps P_1, P_2 for filling tanks 1 and 3
- A power amplifier for the feed pumps
- Three piezoresistive pressure sensors for the water level measurement x_1, x_2, x_3
- A measuring amplifier for the sensor signals in the dSPACE-system.

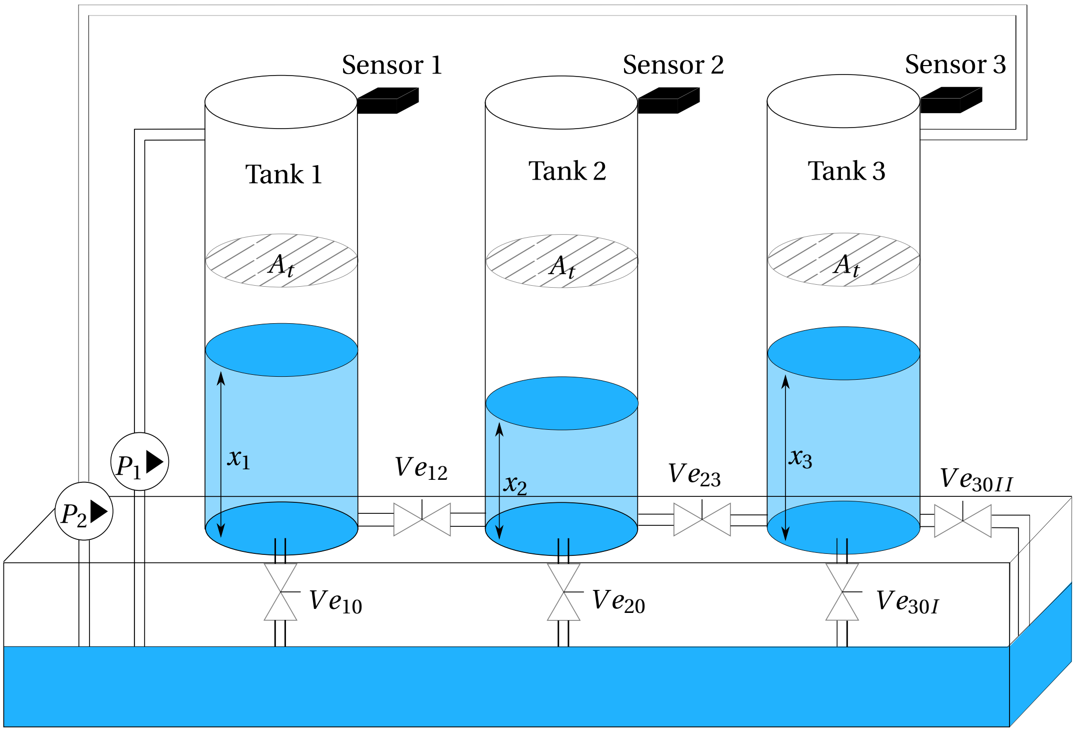

Fig. 3.2: Schematics of the three-tank system

The three tanks are identical in their geometry and dimensions so that the tank cross-section A_t and the tank volume V_1, V_2 and V_3 are the same for each tank. The tanks are connected by the connecting valves Ve_{12} and Ve_{23}, as shown in Figure 3.2. Tanks 1 and 3 have an additional adjustable drain valve Ve_{10} and Ve_{30I} that connect the tanks to the water depot. For the rest of this exercise the valves Ve_{10}, Ve_{30I}, Ve_{12} and Ve_{23} are not considered as control inputs to the system and are open at any time. Tank 2 and tank 3 have additional drain valves Ve_{20} and Ve_{30II}, which are used to vary the amount of drainage that acts as a disturbance to the system. The opening of these valves corresponds to the application of a disturbance signal to the system. In the nominal case, which is assumed for the controller design, the valves Ve_{20} and Ve_{30II} are closed, i.e. the disturbance is zero.

The water levels of the tanks x_1, x_2 and x_3 represent the state variables of the system. These quantities are measured by pressure sensors mounted in the upper part of the tanks (see Figure 3.2). The sensors transmit their measurement information via the measurement amplifier to a dSPACE-system according to Figure 3.3.

Tanks 1 and 3 are supplied with liquid from the depot by the feed pumps. The pumps are diaphragm pumps with a maximum flow rate of 7 l/m.

The flow rates of the two pumps Q_{P1} and Q_{P2} correspond to the control inputs u_1 and u_2. A change in the flow rate of the pumps is achieved by changing the supply voltage of the pumps P_1 and P_2.

The goal of the control system is to set and maintain defined water levels in the tanks 1 and 3.

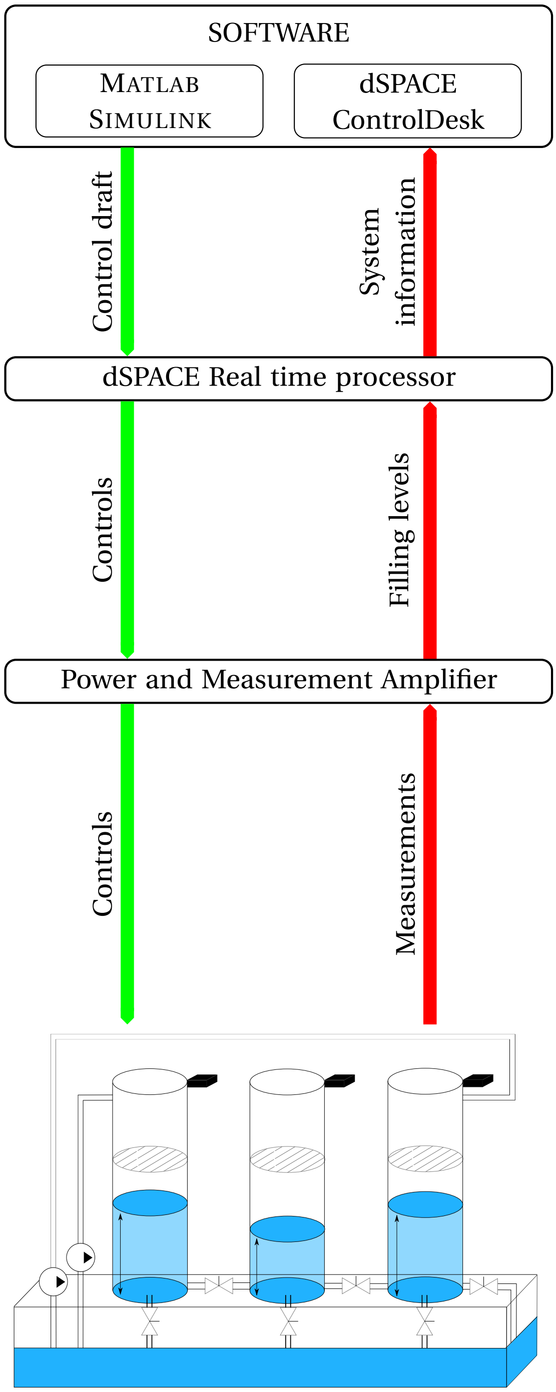

Fig. 3.3: Information flow

3.3.2 Dynamic Modelling

The mathematical modelling is based on balancing the individual volume flows of the three-tank system, i.e. the change in the level of a tank over time corresponds to the difference in the volumes flowing into or out of a tank per unit of time, i.e.

$$Q_{Ti} = \frac{dV_{Ti}}{dt} = A_t \dot{x}_i = Q_{Ti,\text{inflow}} - Q_{Ti,\text{outflow}}, \qquad i \in \{1,2,3\}, \tag{3.29}$$

where A_t represents the cross-sectional area of the tank.

Three different types of volume flows must be considered:

(i) The volume flows of the feed pumps Q_{P1}, Q_{P2}

(ii) The volume flows of the tanks into the water depot $Q_{10}, Q_{20}, Q_{30I}, Q_{30II}$

(iii) The volume flows between the tanks Q_{12}, Q_{23}

3.3.2.1 Volume Flow of the Feed Pumps

The volume flows of the feed pumps represent the control variables u_1 and u_2 of the system. For this experiment

$$Q_{Pi} = u_i, \qquad i \in 1,2,3 \tag{3.30}$$

applies for modelling and the subsequent controller design.

3.3.2.2 Volume Flow from the Tank to the Depot

The volume flows from the tanks into the water depot are assumed to be the ideal volume flow in the nominal case and are calculated according to [1] in dependence on the outflow velocity v_2 (see Figure 3.4) as

$$Q_{\text{ideal}}(t) = A_v v_{i2}(t). \tag{3.31}$$

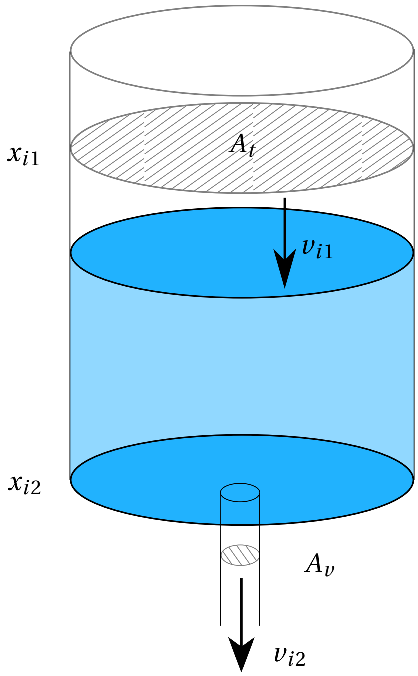

Fig. 3.4: Volume outflow from a tank to the water depot

Since the outflow velocity v_{i2} cannot be measured directly, it must be determined from the state variables, whereby

$$\dot{x}_i = v_{i1} \tag{3.32}$$

holds. For the outflow velocity of a liquid from a vessel the expression

$$\underbrace{p_{i1} + \frac{1}{2}\rho v_{i1}^2 + \rho g x_{i1}}_{p_{i1,ges}} = \underbrace{p_{i2} + \frac{1}{2}\rho v_{i2}^2 + \rho g x_{i2}}_{p_{i2,ges}} \tag{3.33}$$

holds by means of the Bernoulli equation, the continuity equation and the assumption of identical pressures at points x_{i1}, x_{i2}, where

$$p_{i1} = p_{i2} = p_0 \tag{3.34}$$

applies to static pressure and p_0 denotes the atmospheric pressure.

The velocity of the level change is significantly smaller than the outflow velocity $v_{i1} \ll v_{i2}$ due to the cross-sectional ratios $A_t \gg A_v$ and can therefore be neglected.

This results in the law of Torricelli according to [1, 2] and provides the desired relationship for the outflow velocity

$$v_{i2} = \sqrt{2\frac{1}{\rho}\rho g(x_{i1} - x_{i2})} = \sqrt{2g(x_{i1} - x_{i2})} = \sqrt{2g\Delta x_{i0}}. \tag{3.35}$$

The actual flow cannot be achieved in reality due to internal friction of the fluid, pipe friction and deformation of the emerging water jet. To take this into account, the ideal flow is calculated from (3.31) and extended by the correction factor a_{i0} such that

$$Q_{i0}(t) = a_{i0} A_v \sqrt{2g\Delta x_{i0}(t)}, \tag{3.36}$$

where $\Delta x_{i0}(t)$ is the water level responsible for the prevailing hydrostatic pressure (see also Figure 3.4). This level is given by the difference of the points x_{i1} and x_{i2}. For the volume flows from the tanks to the water depot $x_{i2} = 0$ holds, since the outflow is located at the bottom of the tank, thus

$$Q_{i0} = a_{i0} A_v \sqrt{2gx_i}, \qquad i \in \{1,2,3\}. \tag{3.37}$$

3.3.2.3 Volume Flow between the Tanks

The volume flow resulting from the tank connections cannot be directly assigned to an inflow or outflow. Depending on the level of the connected tanks, the flow direction may change.

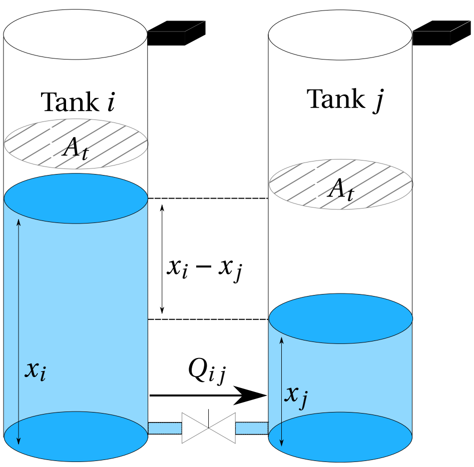

Fig. 3.5: Volume flow between two tanks

The volume flow is proportional to the square root of the difference of the water levels which are responsible for the pressure difference $\sqrt{|x_i - x_j|}$. This level is shown in Figure 3.5. The reason for this relationship is the balance of the weight of the liquids in the tanks

$$Q_{ij} = a_{ij} A_v \operatorname{sgn}(x_i - x_j)\sqrt{2g|x_i - x_j|}, \qquad i, j \in \{(1,2), (2,1), (2,3), (3,2)\}. \tag{3.38}$$

Exercise 3.3.
Solve the following tasks using MATLABs SYMBOLIC MATH TOOLBOX *and* MATLAB / SIMULINK.

(i) Set up a mathematical model of the three-tank system. Balance the volume inflows and outflows for the respective tanks.

(ii) Create a simulation model of the three-tank system in MATLAB / SIMULINK *using the state equations. Use a* `Matlab-Function` *and an integrator and compare the model that you the obtained with the private* `S-Function` `acon_dts_plant.p`*. The values of the required parameters are listed in the Table 3.1.*

Remark 3.3

To include the private `S-Function` `acon_dts_plant.p`, it has to be used as a `Level-2 Matlab S-Function`. The function title must be `acon_dts_plant`. The function is called with only the initial states as parameters (as an array).

As first output the `S-Function` returns the water levels of tanks 1 and 3. The second output includes all states.

It only works if both the function and the parameter names are correct!

Tab. 3.1: Parameter of the three-tank system.

Parameter	Parameter name (Matlab)	Value	Description
A_t	`At`	0.0154 m	Cross sectional area of the tanks
A_V	`Av`	$\pi \cdot (0.0040\text{m})^2$	Cross sectional area of the valves
a_{12}	`a12`	0.3485	Correction factor
a_{23}	`a23`	0.3485	Correction factor
a_{10}	`a10`	0.6677	Correction factor
a_{30I}	`a30I`	0.6591	Correction factor
g	`g`	9.80665 m s^{-2}	Acceleration of gravity

References

[1] M. Stohrer E. Hering R. Martin. „Physik für Ingenieure“. In: *Springer, Berlin* (2007) (cit. on pp. 10, 11).

[2] N. Aksel J. Spruk. „Strömungslehre - Einführung in die Theorie der Strömungen“. In: *Springer, Berlin* (2010) (cit. on p. 11).

[3] T. Meurer. „Nonlinear Control Systems - Lecture Notes“. In: `https://www.control.tf.uni-kiel.de/en/teaching/summer-term/nonlinear-control-systems` (2020) (cit. on pp. 2, 3).