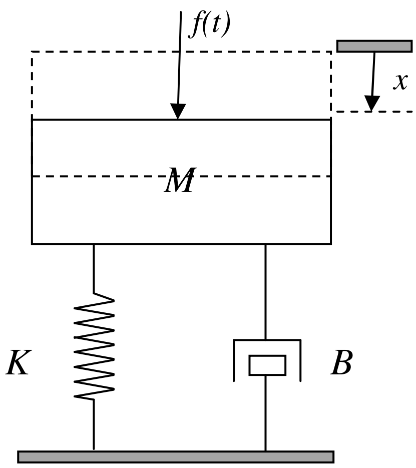
4. State-space representation

4.1 Introduction – by examples

- Previously, we have developed models for a variety of systems, ending up in each case with one or more differential equations, some of which were nonlinear.
- When the equations were linear, the model can be described in terms of a Laplace transfer function.
- Consider for example the spring-mass damper system:



Physical model

$$M\ddot{x} + B\dot{x} + Kx = f(t)$$

or $$M\frac{d^2x}{dt^2} + B\frac{dx}{dt} + Kx = f(t)$$

$$G(s) = \frac{X(s)}{F(s)} = \frac{1}{Ms^2 + Bs + K}$$

- While transfer functions are compact and have several advantages in relation to analysis of dynamical systems, they also have a number of weaknesses as follows:
  - They do not handle initial conditions (assumed to be zero!).
  - Information about internal variables is lost ($\dot{x}$ in the case of the above example).
  - For general *m*-input, *p*-output systems, we would need a total of *m* x *p* transfer functions to fully describe the system.
- An alternative model representation, known as a **state-space model**, overcomes these weaknesses and, as we shall see later, lends itself to computer implementation (simulation).
- The basic idea behind state-space modelling is to write down a set of first-order differential equations in terms of the system state(s) and input(s).

- **Example 4.1:** Develop a state-space model for the second-order differential equation model of the spring-mass damper system:

$$M\ddot{x} + B\dot{x} + Kx = f(t)$$

Solution: We proceed as follows:

Firstly we have a ***second-order*** differential equation; hence we define two states x_1 and x_2:

$$x_1 = x, \quad x_2 = \dot{x}$$

We then obtain expressions for $\dot{x}_1$ and $\dot{x}_2$:

$$\dot{x}_1 = \dot{x} \quad \text{and} \quad \dot{x}_2 = \ddot{x} = \frac{1}{M}\left(f(t) - B\dot{x} - Kx\right)$$

Note the latter expression is obtained for the original mathematical model.

We can rewrite these in terms states x_1 and x_2 only, by replacing x and $\dot{x}$ with their state equivalent. Hence:

$$\dot{x}_1 = x_2 \quad \text{and} \quad \dot{x}_2 = \frac{1}{M}\left(f(t) - Bx_2 - Kx_1\right)$$

Now, we combine these into matrix form as follows:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{K}{M} & -\frac{B}{M} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{M} \end{bmatrix} f(t)$$

Finally, let us define the output as $y = x$, the displacement in the mass. Hence $y = x_1$ giving:

$$[y] = [1 \quad 0]\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + [0]f(t) \Rightarrow [y] = [1 \quad 0]\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Hence our state-space model is expressed, in full, as:

$$\begin{bmatrix} \dot{x}_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{k}{M} & -\frac{B}{M} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{M} \end{bmatrix} f(t)$$

$$[y] = [1 \quad 0]\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

- **Example 4.2:** Develop a state-space model for the first-order differential equation model of the single tank system:

$$A\frac{dh}{dt} = F_{in} - kh$$

Solution:

Here, we only have a ***first-order*** differential equation and hence only one state:

$$x_1 = h$$

We now obtain an expression for $\dot{x}_1$:

$$\dot{x}_1 = \dot{h}$$

We can obtain an expression for $\dot{h}$ from the differential equation, as follows:

$$A\dot{h} = F_{in} - kh \Rightarrow \dot{h} = \frac{1}{A}(F_{in} - kh)$$

Rewriting this equation in terms of state x_1, and letting input $u = F_{in}$, we get:

$$\dot{x}_1 = \frac{-k}{A}x_1 + \frac{1}{A}u$$

Let output $y = h = x_1$. Hence, our state-space model is expressed, in full, as:

$$[\dot{x}_1] = \left[\frac{-k}{A}\right][x_1] + \left[\frac{1}{A}\right][u]$$

$$[y] = [1][x_1]$$

- Note – in this example, all the matrices are simply 1x1 in dimensions, as this is a simple first order system.

- Nevertheless, the example serves to illustrate that all models, irrespective of their order, can be represented in a state-space format.

- **Example 4.3:** Develop a state-space model for the double-mass-spring-damper car suspension system with the equations:

$$M\ddot{x}_1 + B(\dot{x}_1 - \dot{x}_2) + K(x_1 - x_2) = f(t)$$

$$m\ddot{x}_2 + B_t\dot{x}_2 + K_t x_2 = B(\dot{x}_1 - \dot{x}_2) + K(x_1 - x_2)$$

Solution: We can rewrite these equations as:

$$\ddot{x}_1 = -\frac{B}{M}\dot{x}_1 + \frac{B}{M}\dot{x}_2 - \frac{K}{M}x_1 + \frac{K}{M}x_2 + \frac{f(t)}{M}$$

$$\ddot{x}_2 = -\left(\frac{B}{m} + \frac{B_t}{m}\right)\dot{x}_2 + \frac{B}{m}\dot{x}_1 - \left(\frac{K}{m} + \frac{K_t}{m}\right)x_2 + \frac{K}{m}x_1$$

Here, we have ***two second-order*** differential equations and hence we need 4 states:

$$X_1 = x_1$$

$$X_2 = x_2$$

$$X_3 = \dot{x}_1$$

$$X_4 = \dot{x}_2$$

Note, here we're using capital *X* for the states, to avoid confusion with the actual variables in the equations.

We now obtain an expression for $\dot{X}_1, \dot{X}_2, \dot{X}_3$ and $\dot{X}_4$:

$$\dot{X}_1 = \dot{x}_1 = X_3$$

$$\dot{X}_2 = \dot{x}_2 = X_4$$

$$\dot{X}_3 = -\frac{B}{M}X_3 + \frac{B}{M}X_4 - \frac{K}{M}X_1 + \frac{K}{M}X_2 + \frac{f(t)}{M}$$

$$\dot{X}_4 = -\left(\frac{B}{m} + \frac{B_t}{m}\right)X_4 + \frac{B}{m}X_3 - \left(\frac{K}{m} + \frac{K_t}{m}\right)X_2 + \frac{K}{m}X_1$$

Let us define the outputs as the position of both masses, i.e. $y_1 = X_1$ and $y_2 = X_2$.

Hence, our state-space model is:

$$\begin{bmatrix} \dot{X}_1 \\ \dot{X}_2 \\ \dot{X}_3 \\ \dot{X}_4 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -\frac{K}{M} & \frac{K}{M} & -\frac{B}{M} & \frac{B}{M} \\ \frac{K}{m} & -\left(\frac{K+K_t}{m}\right) & \frac{B}{m} & -\left(\frac{B+B_t}{m}\right) \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \frac{1}{M} \\ 0 \end{bmatrix} f(t)$$

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{bmatrix}$$

- Note, with a state-space representation, it is very easy to add other outputs.

- By way of illustration, let's say that for in the last example we also want to consider the output $y_3 = X_1 - X_2$, i.e. the difference in position between the two masses.

- In this instance, we can simply rewrite the output matrix, for the last example, as:

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{bmatrix}$$

4.2 Formal definitions

- The commonly used terms associated with state-space representation are as follows:

  - **State** – the state of a dynamic system is the smallest set of variables (called state variables) such that knowledge of these variables at $t = t_0$, together with knowledge of the input for $t \geq t_0$, completely determines the behaviour of the system for $t \geq t_0$.

  - **State variables** – the variables that make up the state as defined above.

  - **State vector** – when there is more than one state variable, they are normally collected together into a vector called a state vector.

  - **State-space** – the n-dimensional state vector can be viewed as a point moving around in n-dimensional space. This n-dimensional space is known as state-space.

4.3 General state-space form

- The dynamics of a general n^{th} order linear dynamical system, with m inputs, is completely described by a n^{th} order state-space equation of the form:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \vdots \\ \dot{x}_n \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} + \begin{bmatrix} b_{11} & \cdots & b_{1m} \\ b_{21} & \cdots & b_{2m} \\ \vdots & \ddots & \vdots \\ b_{n1} & \cdots & b_{nm} \end{bmatrix} \begin{bmatrix} u_1 \\ \vdots \\ u_m \end{bmatrix}$$

$$\equiv \qquad \dot{\boldsymbol{x}} = \boldsymbol{Ax} + \boldsymbol{Bu} \qquad\qquad \textbf{\textit{state equation}}$$

with a set of initial conditions (one for each state):

$$\boldsymbol{x}(0) = \begin{bmatrix} x_1(0) \\ x_2(0) \\ \vdots \\ x_n(0) \end{bmatrix} = \boldsymbol{x_0} \qquad\qquad \textbf{\textit{initial conditions}}$$

- The model output(s) are given by a linear combination of the states and the inputs. Given p outputs, we obtain:

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_p \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1n} \\ c_{21} & c_{22} & \cdots & c_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{p1} & c_{p2} & \cdots & c_{pn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} + \begin{bmatrix} d_{11} & \cdots & d_{1m} \\ d_{21} & \cdots & d_{2m} \\ \vdots & \ddots & \vdots \\ d_{p1} & \cdots & d_{pm} \end{bmatrix} \begin{bmatrix} u_1 \\ \vdots \\ u_m \end{bmatrix}$$

$$\equiv \qquad \boldsymbol{y} = \boldsymbol{Cx} + \boldsymbol{Du} \qquad\qquad \textbf{\textit{output equation}}$$

- Together, these two equations describe the state-space model of a system.

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{Ax} + \boldsymbol{Bu} \qquad , \boldsymbol{x}(0) = \boldsymbol{x_0} \\ \boldsymbol{y} &= \boldsymbol{Cx} + \boldsymbol{Du} \end{aligned}$$

- The matrices $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$ and $\boldsymbol{D}$ are called the state matrix, input matrix, output matrix and direct transmission matrix respectively:

$\boldsymbol{A}$	-	state matrix (n x n)
$\boldsymbol{B}$	-	input matrix (n x m)
$\boldsymbol{C}$	-	output matrix (p x n)
$\boldsymbol{D}$	-	direct transmission matrix (p x m)

4.4 Transforming system models to state-space form

- It is not always obvious how to choose the system state variables so that we end up with a set of state-equations (i.e. a set of first-order differential equations).

- For example, consider the set of equations obtained from the circuit modelling example that used nodal analysis (see example 3.7), one for each node:

$$\frac{L}{R_1}\dot{v}_1 + v_1 + LC\ddot{v}_1 - v_0 - \frac{L}{R_1}\dot{v}_{in} = 0$$

and

$$v_1 = \frac{L}{R_2}\dot{v}_0 + v_0$$

- For the sake of convenience, let $R = C = L = 1$, giving:

$$\dot{v}_1 + v_1 + \ddot{v}_1 - v_0 - \dot{v}_{in} = 0 \quad \text{or} \quad \ddot{v}_1 = \dot{v}_{in} - \dot{v}_1 - v_1 + v_0$$

and

$$v_1 = \dot{v}_0 + v_0 \quad \text{or} \quad \dot{v}_0 = v_1 - v_0$$

- We can try assigning the states $x_1 = v_1$, $x_2 = \dot{v}_1$ and $x_3 = v_0$, but this doesn't give us a workable set of first-order equations because of the derivative on the input, i.e. $\dot{v}_{in}$:

$$\dot{x}_1 = x_2$$
$$\dot{x}_2 = -x_2 - x_1 + x_3 + \dot{v}_{in}$$
$$\dot{x}_3 = x_1 - x_3$$

We cannot have a derivative of the input in a state-space equation.
We need to choose the states so that this does not occur.

- One possible solution is to bring $\dot{v}_{in}$ over to the left hand side of the equation, i.e.:

$$\ddot{v}_1 - \dot{v}_{in} = -\dot{v}_1 - v_1 + v_0$$

and then choose the states as follows:

$$x_1 = v_1,\ x_2 = \dot{v}_1 - v_{in} \text{ and } x_3 = v_0$$

- This leads to:

$$\dot{x}_1 = \dot{v}_1$$
$$\dot{x}_2 = \ddot{v}_1 - \dot{v}_{in} = -\dot{v}_1 - v_1 + v_0$$
$$\dot{x}_3 = \dot{v}_0 = v_1 - v_0$$

- However: $x_2 = \dot{v}_1 - v_{in} \qquad \Rightarrow \qquad \dot{v}_1 = x_2 + v_{in}$

- Hence, we can write the state equation as follows:

$$\dot{x}_1 = x_2 + v_{in}$$
$$\dot{x}_2 = -(x_2 + v_{in}) - x_1 + x_3$$
$$\dot{x}_3 = \mathrm{x}_1 - x_3$$

or:

$$\boldsymbol{x} = \begin{bmatrix} 0 & 1 & 0 \\ -1 & -1 & 1 \\ 1 & 0 & -1 \end{bmatrix} \boldsymbol{x} + \begin{bmatrix} 1 \\ -1 \\ 0 \end{bmatrix} v_{in}$$

- Note, that the rank of the state matrix (i.e. matrix $\boldsymbol{A}$) is 2.

- In other words, the determinant of matrix $\boldsymbol{A}$ = 0. *Verify this for yourself.*

- The reason for this is that the third equation is not independent of the other two.

- Here, it is a composite of the other two equations, i.e.

equation 3 = – (equation 1 + equation 2)

- The transfer function for the system is:

$$\frac{V_0(s)}{V_i(s)} = \frac{1}{s^2 + 2s + 2} \qquad \text{(\textit{Verify this for yourself})}$$

- This is a second order transfer function.

- Hence the system is second order and, therefore, we should only need two states to describe it.

- Thus, the state model obtained above is ***not a minimal state-space realisation***.

- **The best way to obtain a minimal realisation is to derive the state-space model from the transfer function model of the system.**

- There are two possibilities, leading to two different methods for determining the state variables – one does not involve derivatives in the input, the other one does!

4.4.1 Transfer function → State space, when the input DOES NOT involve derivatives

- Consider the following third-order transfer function:

$$G(s) = \frac{Y(s)}{U(s)} = \frac{k}{s^3 + a_2 s^2 + a_1 s + a_0}$$

- Converting this to the time domain gives:

$$\dddot{y} + a_2 \ddot{y} + a_1 \dot{y} + a_0 y = ku$$

or

$$\dddot{y} = -a_2 \ddot{y} - a_1 \dot{y} - a_0 y + ku$$

- We simply define the states as $x_1 = y, x_2 = \dot{y}$ and $x_3 = \ddot{y}$, leading to the following state model:

$$\begin{aligned}
\dot{x}_1 &= \dot{y} = x_2 \\
\dot{x}_2 &= \ddot{y} = x_3 \\
\dot{x}_3 &= \dddot{y} = -a_2 x_3 - a_1 x_2 - a_0 x_1 + ku
\end{aligned}$$

$$y = x_1$$

giving:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a_0 & -a_1 & -a_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ k \end{bmatrix} u$$

$$y = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

This state-space model is known as the **control *(or controller)* canonical form**.

4.4.2 Transfer function → State space, when the input involves derivatives

- Consider the following third-order transfer function:

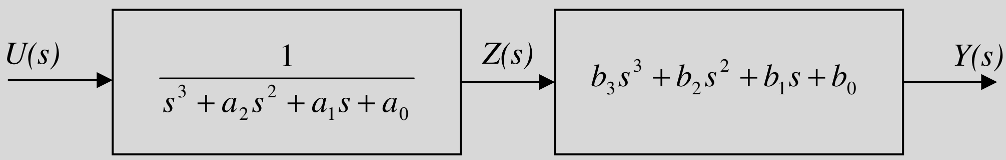
$$G(s) = \frac{Y(s)}{U(s)} = \frac{b_3 s^3 + b_2 s^2 + b_1 s + b_0}{s^3 + a_2 s^2 + a_1 s + a_0}$$

- Firstly, we split the transfer function into two parts by defining an intermediate variable *Z(s)* as follows:

$$\frac{Z(s)}{U(s)} = \frac{1}{s^3 + a_2 s^2 + a_1 s + a_0}$$

and

$$\frac{Y(s)}{Z(s)} = b_3 s^3 + b_2 s^2 + b_1 s + b_0$$



- Converting to the time domain gives:

$$\dddot{z} + a_2 \ddot{z} + a_1 \dot{z} + a_0 z = u$$

and

$$b_3 \dddot{z} + b_2 \ddot{z} + b_1 \dot{z} + b_0 z = y$$

- Setting the states as $x_1 = z, x_2 = \dot{z}$ and $x_3 = \ddot{z}$, we get the state equation as follows:

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= x_3 \\ \dot{x}_3 &= -a_2 x_3 - a_1 x_2 - a_0 x_1 + u \end{aligned}$$

Once again, we have the **control canonical form**.

giving:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a_0 & -a_1 & -a_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u$$

- In terms of the output equation:

  - **if $b_3 = 0$ then:** $y = b_2\ddot{z} + b_1\dot{z} + b_0 z = b_2 x_3 + b_1 x_2 + b_0 x_1$

    giving: $$y = \begin{bmatrix} b_0 & b_1 & b_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$ (*Note, the D matrix is 0 in this case*)

  - **if $b_3 \neq 0$ then:** $y = b_2 x_3 + b_1 x_2 + b_0 x_1 + b_3 \dddot{z}$

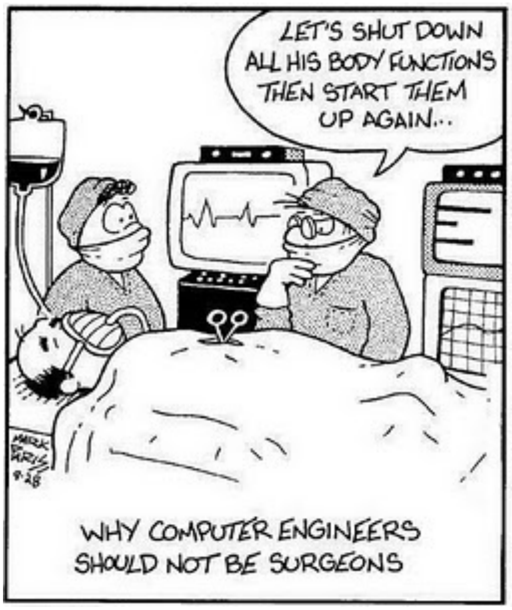
    but: $\dddot{z} = \dot{x}_3 = -a_2 x_3 - a_1 x_2 - a_0 x_1 + u$

    giving: $y = (b_2 - b_3 a_2)x_3 + (b_1 - b_3 a_1)x_2 + (b_0 - b_3 a_0)x_1 + b_3 u$

    therefore: $$y = \begin{bmatrix} (b_0 - b_3 a_0) & (b_1 - b_3 a_1) & (b_2 - b_3 a_2) \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + [b_3]\, u$$

- Note that the state matrix (i.e. matrix ***A***) is exactly the same for both types of transfer function models.

- **This implies that the dynamics depend on the transfer function denominator only**.



4.5 Discrete-time state-space models

- Difference equation models can also be represented in state-space form.

- In this case, the state-space equations take the form:

$$\begin{aligned}\mathbf{x}_{k+1} &= \mathbf{A}_D\mathbf{x}_k + \mathbf{B}_D\mathbf{u}_k \\ \mathbf{y}_k &= \mathbf{C}_D\mathbf{x}_k + \mathbf{D}_D\mathbf{u}_k\end{aligned}$$

or

$$\begin{aligned}\mathbf{x}(k+1) &= \mathbf{A}_D\mathbf{x}(k) + \mathbf{B}_D\mathbf{u}(k) \\ \mathbf{y}(k) &= \mathbf{C}_D\mathbf{x}(k) + \mathbf{D}_D\mathbf{u}(k)\end{aligned}$$

- The *D* subscript simply denotes discrete-time.

- This is similar in structure to the continuous state-space model equations.

- Furthermore, the matrices and vectors have the same dimensions as their continuous counterparts.

- Finally, the same methods can be used to derive the models from difference equation representations.

- **Example 4.4:** Obtain the state-space model from the following difference equation:

$$y_k = 0.5y_{k-1} + 0.3y_{k-2} + 1.5u_{k-1} + 0.5u_{k-2}$$

Solution:

The transfer function (using z-transforms) is given by:

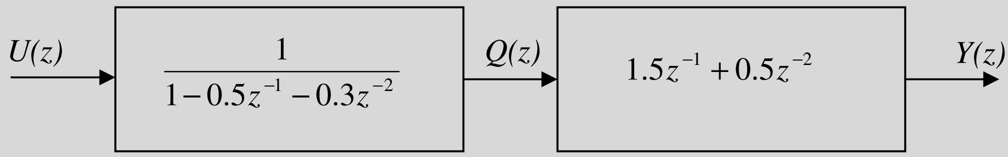
$$\frac{Y(z)}{U(z)} = \frac{1.5z^{-1} + 0.5z^{-2}}{1 - 0.5z^{-1} - 0.3z^{-2}}$$

Note that $y_k \rightarrow Y(z)$, $y_{k-1} \rightarrow z^{-1}Y(z)$, etc. *Verify the transfer function for yourself.*

Let $Q(z)$ be the intermediate variable.

Hence:

$$\frac{Q(z)}{U(z)} = \frac{1}{1 - 0.5z^{-1} - 0.3z^{-2}} \quad \text{and} \quad \frac{Y(z)}{Q(z)} = 1.5z^{-1} + 0.5z^{-2}$$



Therefore:

$$q_k - 0.5q_{k-1} - 0.3q_{k-2} = u_k \qquad \ldots \text{(A)}$$

and

$$1.5q_{k-1} + 0.5q_{k-2} = y_k \qquad \ldots \text{(B)}$$

Choosing the discrete states as $x_1(k) = q_{k-1}$ and $x_2(k) = q_{k-2}$, gives:

$$x_1(k+1) = q_k = 0.5q_{k-1} + 0.3q_{k-2} + u_k$$
$$\Rightarrow x_1(k+1) = 0.5x_1(k) + 0.3x_2(k) + u_k$$

from equation A

and

$$x_2(k+1) = q_{k-1} = x_1(k)$$

Hence:

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \end{bmatrix} = \begin{bmatrix} 0.5 & 0.3 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u$$

$$\equiv \mathbf{x}(k+1) = \mathbf{A}_D \mathbf{x}(k) + \mathbf{B}_D \mathbf{u}(k)$$

In terms of the output equation:

$$y_k = 1.5q_{k-1} + 0.5q_{k-2}$$
$$\Rightarrow y_k = 1.5x_1(k) + 0.5x_2(k)$$

from equation B

Hence:

$$y_k = \begin{bmatrix} 1.5 & 0.5 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix}$$

$$\equiv \mathbf{y}(k) = \mathbf{C}_D \mathbf{x}(k)$$

4.6 Obtaining transfer functions from state-space models

4.6.1 Continuous-time state-space model → transfer function

- Consider the following single-input-single-output continuous-time state-space model:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}u(t) \qquad\qquad y(t) = \boldsymbol{C}\boldsymbol{x}(t)$$

- Taking the Laplace transform gives:

$$s\boldsymbol{X}(s) = \boldsymbol{A}\boldsymbol{X}(s) + \boldsymbol{B}U(s) \qquad\qquad Y(s) = \boldsymbol{C}\boldsymbol{X}(s)$$

- Rearranging the state equation gives:

$$(sI - \boldsymbol{A})\boldsymbol{X}(s) = \boldsymbol{B}U(s) \quad \Rightarrow \quad \boldsymbol{X}(s) = (sI - \boldsymbol{A})^{-1}\boldsymbol{B}U(s)$$

- Substituting this equation into the output equation gives: $Y(s) = \mathbf{C}(sI - \mathbf{A})^{-1}\mathbf{B}U(s)$

- Hence, the transfer function is defined in terms of the state-space equations matrices as:

$$G(s) = \frac{Y(s)}{U(s)} = \mathbf{C}(sI - \mathbf{A})^{-1}\mathbf{B}$$

4.6.2 Discrete-time state-space model → transfer function

- Consider the following single-input-single-output discrete-time state-space model:

$$\boldsymbol{x}(k+1) = \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}u(k) \qquad\qquad y(k) = \boldsymbol{C}\boldsymbol{x}(k)$$

- Taking the z-transform gives:

$$z\boldsymbol{X}(z) = \boldsymbol{A}\boldsymbol{X}(z) + \boldsymbol{B}U(z) \qquad\qquad Y(z) = \boldsymbol{C}\boldsymbol{X}(z)$$

- Rearranging the state equation gives:

$$(zI - \boldsymbol{A})\boldsymbol{X}(z) = \boldsymbol{B}U(z) \quad \Rightarrow \quad \boldsymbol{X}(z) = (zI - \boldsymbol{A})^{-1}\boldsymbol{B}U(z)$$

- Substituting this equation into the output equation gives: $Y(z) = \mathbf{C}(zI - \mathbf{A})^{-1}\mathbf{B}U(z)$

- Hence, the transfer function is defined in terms of the state-space equations matrices as:

$$G(z) = \frac{Y(z)}{U(z)} = \mathbf{C}(zI - \mathbf{A})^{-1}\mathbf{B}$$

- **Example 4.5:** Determine the transfer function for the following system:

$$\dot{\boldsymbol{x}}(t) = \begin{bmatrix} 0 & 2 \\ -3 & 5 \end{bmatrix} \boldsymbol{x}(t) + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u(t), \qquad y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} \boldsymbol{x}(t)$$

Solution:

$$G(s) = \boldsymbol{C}(sI - \boldsymbol{A})^{-1}\boldsymbol{B}$$

$$= \begin{bmatrix} 1 & 0 \end{bmatrix} \left(\begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix} - \begin{bmatrix} 0 & 2 \\ -3 & 5 \end{bmatrix} \right)^{-1} \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} s & -2 \\ 3 & s-5 \end{bmatrix}^{-1} \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

$$= \frac{\begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} s-5 & 2 \\ -3 & s \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix}}{s^2 - 5s + 6}$$

$$= \frac{\begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} s-5 \\ -3 \end{bmatrix}}{s^2 - 5s + 6} \qquad = \frac{s-5}{s^2 - 5s + 6}$$

Hence:

$$G(s) = \frac{Y(s)}{U(s)} = \frac{s-5}{s^2 - 5s + 6}$$

4.7 Nonlinear state-space equations

- The examples of continuous and discrete state-space models considered thus far have all been linear.

- However, these are only special cases of the more general form of the state space equations, which are used to describe general nonlinear dynamical systems:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$$
$$\mathbf{y} = \mathbf{g}(\mathbf{x}, \mathbf{u})$$

i.e.:

$$\begin{bmatrix} \dot{x}_1 = f_1(x_1, x_2, \ldots, x_n, u_1, u_2, \ldots, u_m) \\ \dot{x}_2 = f_2(x_1, x_2, \ldots, x_n, u_1, u_2, \ldots, u_m) \\ \vdots \\ \dot{x}_n = f_n(x_1, x_2, \ldots, x_n, u_1, u_2, \ldots, u_m) \end{bmatrix}$$

n nonlinear state equations

$$\begin{bmatrix} y_1 = g_1(x_1, x_2, \ldots, x_n, u_1, u_2, \ldots, u_m) \\ y_2 = g_2(x_1, x_2, \ldots, x_n, u_1, u_2, \ldots, u_m) \\ \vdots \\ y_p = g_n(x_1, x_2, \ldots, x_n, u_1, u_2, \ldots, u_m) \end{bmatrix}$$

p nonlinear output equations

- Consider, for example, the coupled tank system, which was represented by **a set of nonlinear equations**:

$$\dot{h}_1 = \frac{1}{A_1} f_{in} - \frac{k_1}{A_1}\sqrt{h_1 - h_2} \qquad \rightarrow \qquad \dot{h}_1 = f_1\,(h_1, h_2, f_{in})$$

$$\dot{h}_2 = \frac{k_1}{A_2}\sqrt{h_1 - h_2} - \frac{k_2}{A_2}\sqrt{h_2 - h_3} \qquad \rightarrow \qquad \dot{h}_2 = f_2\,(h_1, h_2, f_{in})$$

- Note, in the second equation, the value associated with f_{in} is 0 in this case!

- **Here, we have two states, h_1 and h_2 and input f_{in}.**

- If we are interested in the flow between the tanks and the height of tank 1, for example, then we can represent the **output equations** as:

$$y_1 = k_1\sqrt{h_1 - h_2} \qquad \rightarrow \qquad y_1 = g_1\,(h_1, h_2, f_{in})$$
$$y_2 = h_1 \qquad \rightarrow \qquad y_2 = g_2\,(h_1, h_2, f_{in})$$

- Although this system is clearly nonlinear, we can nevertheless write down the full state-space model for this system.