Selected slides from

Dynamics and control of space structures

Lorenzo Dozio

Control system design – introduction to state-space methods (part 1)

2020/2021

Outline

Control system design – introduction to state-space methods

Part 1

- Introduction and motivation
- State space fundamentals
- Controllability and observability

Part 2

- Pole placement
- Linear quadratic regulator (LQR) (optimal control)
- Steady-state tracking

Part 3

- Linear observer
- Guidelines for selecting weighting matrices in LQR
- Finite-horizon optimal control

Introduction

- The main goal of this part of the course is to **introduce some basic relevant topics related to the design of feedback control systems using state-space methods** (in the time domain for multiple-inputs-multiple-outputs systems).

- The topics will cover only the essential tools (pole-placement, LQR, linear observers) with the aim of preparing the background to more advanced techniques (e.g., robust control) – not covered in this course.

Good **references** are the following:

B. Friedland,
Control System Design. An Introduction to State-Space Methods, McGraw-Hill

R.L. Williams II, D.A. Lawrence,
Linear state-space control systems, Wiley

The idea of state-space methods comes from the state-variable method of describing differential equations.

In this method a differential equation of order *N* describing a dynamic system is represented as a set of first-order differential equations by using a vector of *N* variables which is called **state of the system**.

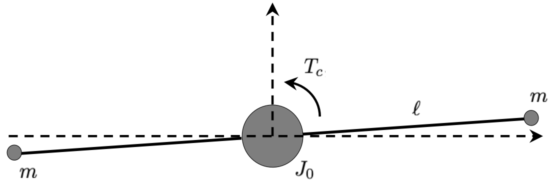
$$\dddot{y} + 6\ddot{y} + 11\dot{y} + 6y = u$$

$$\begin{aligned} x_1 &= y \\ x_2 &= \dot{y} \\ x_3 &= \ddot{y} \end{aligned} \quad \Rightarrow \quad \begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= x_3 \\ \dot{x}_3 &= -6x_3 - 11x_2 - 6x_1 + u \end{aligned} \quad \Rightarrow \quad \mathbf{x} = \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \end{Bmatrix}$$

the state of the system is associated to the number of terms that are required to describe correctly the differential equation of order N.

Control system design – state-space methods
Introduction

Example 1: rigid satellite



Inertia moment of the s/c

Torque

$$J\ddot{\theta}(t) = T_c(t)$$
$$\theta(0) = 0$$
$$\dot{\theta}(0) = 0$$

$$x_1(t) = \theta(t)$$
$$x_2(t) = \dot{\theta}(t)$$

$$\dot{x}_1(t) = \dot{\theta}(t) = x_2(t)$$
$$\dot{x}_2(t) = \ddot{\theta}(t) = \frac{1}{J}T_c(t)$$
$$x_1(0) = 0$$
$$x_2(0) = 0$$

$$\mathbf{x}(t) = \begin{Bmatrix} x_1(t) \\ x_2(t) \end{Bmatrix}$$

$$\dot{\mathbf{x}}(t) = \begin{Bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{Bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{Bmatrix} x_1(t) \\ x_2(t) \end{Bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{J} \end{bmatrix} T_c(t)$$

$\mathbf{x}(0) = \mathbf{0}$ (initial condition)

Control system design – state-space methods
Introduction

The state of a dynamic system often directly describes the **distribution of internal energy in the system**.

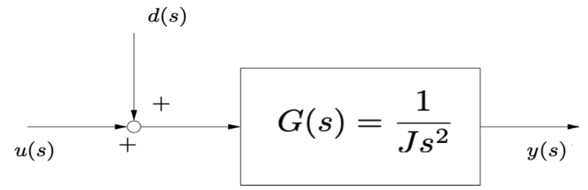
For example, for electro-mechanical systems it is common to select the following as state-variables:
position (potential energy)
velocity (kinetic energy)
capacitor voltage (electric energy)
inductor current (magnetic energy)
thermal systems temperature (thermal energy).

The internal energy can always be computed from the state variables.

We can relate the state to the system inputs and outputs and thus connect the internal variables to the external inputs (control and disturbance) and to the sensed outputs.

In contrast, the transfer function relates only the input to the output and does not show the internal behavior. The state form keeps the latter information, which is sometimes important.

Example 1 (rigid satellite) – input-output representation

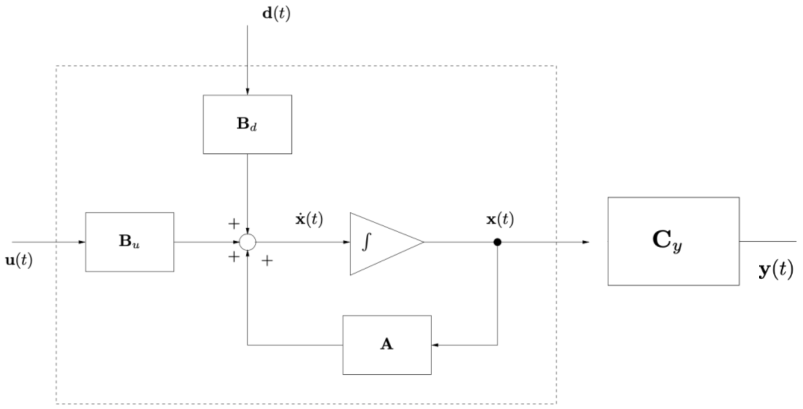


Example 1 (rigid satellite) – state space representation

input
disturbances

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}_u u(t) + \mathbf{B}_d d(t)$$

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \quad \mathbf{B}_u = \begin{bmatrix} 0 \\ \frac{1}{J} \end{bmatrix} \quad \mathbf{B}_d = \begin{bmatrix} 0 \\ \frac{1}{J} \end{bmatrix}$$



$$y(t) = \theta(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} \mathbf{x}(t) = \mathbf{C}_y \mathbf{x}(t)$$

The rotation in the state form was defined on the first state variable ⇒ we can relate the output to the state and to the control.

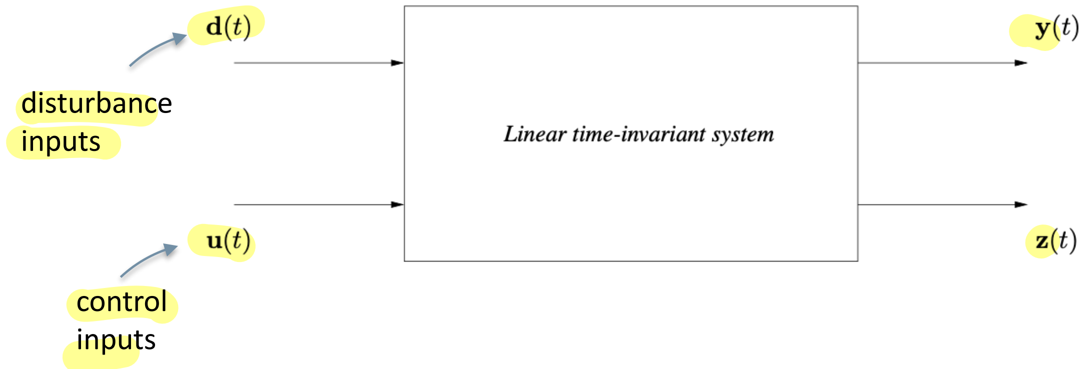
Example 1 (rigid satellite) – state space representation

$$\boxed{\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}_u u(t) + \mathbf{B}_d d(t)\\ y(t) &= \mathbf{C}_y \mathbf{x}(t)\end{aligned}}$$

state equation

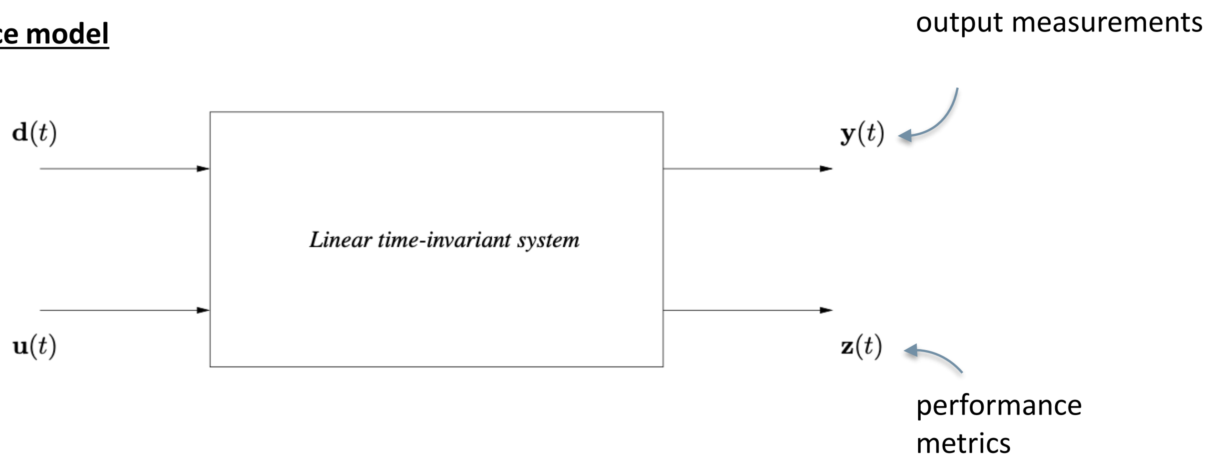
output equation

Generalized state-space model



- disturbance inputs, denoted by $\mathbf{d}(t) \in \mathbb{R}^{n_d \times 1}$, which are typically external prescribed loads, also called exogenous or environmental inputs, that are responsible of exciting the system in an undesired manner;
- control inputs, denoted by $\mathbf{u}(t) \in \mathbb{R}^{n_u \times 1}$, which are controllable variables used to regulate in some way the dynamic response of the system such that the system subjected to $\mathbf{d}$ and $\mathbf{u}$ exhibits a desired dynamic behaviour. The input vector $\mathbf{u}$ is usually associated with the design of a control system.

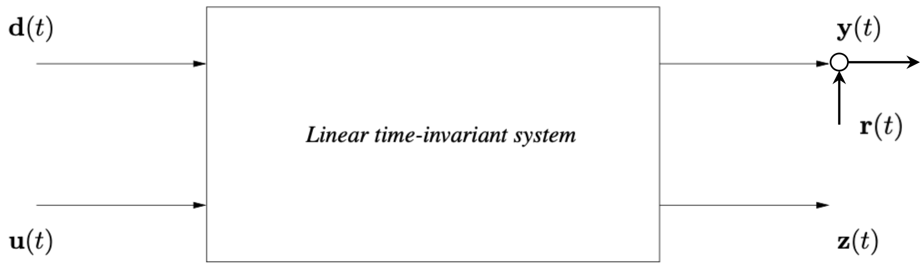
Generalized state-space model



- measured output variables, denoted by $\mathbf{y}(t) \in \mathbb{R}^{n_y \times 1}$, which are physical quantities measured by sensors and representing the system response as perceived by an observer;
- performance variables, denoted by $\mathbf{z}(t) \in \mathbb{R}^{n_z \times 1}$, which are variables related to a performance measure of the structural response and typically introduced as error outputs when a control system is to be designed.

Generalized state-space model

Since measured outputs are signals coming from sensor devices, they are affected by measurement noise, which should be kept as small as possible, but it is never exactly equal to zero. Measurement noise is typically modelled as an additive noise $\mathbf{r}(t) \in \mathbb{R}^{n_r \times 1}$, which is considered as an exogenous variable since it is prescribed (i.e., not controllable).



According to the above notation and the presence of measurement noise, the most general form of the state-space representation of a linear time-invariant system can be written as

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}_u\mathbf{u}(t) + \mathbf{B}_d\mathbf{d}(t) \\ \mathbf{y}(t) &= \mathbf{C}_y\mathbf{x}(t) + \mathbf{D}_{yu}\mathbf{u}(t) + \mathbf{D}_{yd}\mathbf{d}(t) + \mathbf{D}_{yr}\mathbf{r}(t) \\ \mathbf{z}(t) &= \mathbf{C}_z\mathbf{x}(t) + \mathbf{D}_{zu}\mathbf{u}(t) + \mathbf{D}_{zd}\mathbf{d}(t)\end{aligned} \tag{14.58}$$

<u>Generalized state-space model</u>

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}_u\mathbf{u}(t) + \mathbf{B}_d\mathbf{d}(t)$$
$$\mathbf{y}(t) = \mathbf{C}_y\mathbf{x}(t) + \mathbf{D}_{yu}\mathbf{u}(t) + \mathbf{D}_{yd}\mathbf{d}(t) + \mathbf{D}_{yr}\mathbf{r}(t)$$
$$\mathbf{z}(t) = \mathbf{C}_z\mathbf{x}(t) + \mathbf{D}_{zu}\mathbf{u}(t) + \mathbf{D}_{zd}\mathbf{d}(t)$$

- $\mathbf{A}$ is the $n_x \times n_x$ state matrix

  input matrix.
- $\mathbf{B}_u$ is the $n_x \times n_u$ input matrix (the ith column is referred to the influence vector of the ith control input)

  input matrix reflect the influence of disturbances.
- $\mathbf{B}_d$ is the $n_x \times n_d$ disturbance matrix (the ith column is referred to the influence vector of the ith disturbance input)
- $\mathbf{C}_y$ is the $n_y \times n_x$ output matrix (the ith row is referred to the ith output variable)
- $\mathbf{D}_{yu}$ is the $n_y \times n_u$ direct feedthrough matrix (the element (i, j) is the direct feedthrough term of the jth control input on the ith output)
- $\mathbf{D}_{yd}$ is the $n_y \times n_d$ disturbance direct feedthrough matrix (the element (i, j) is the direct feedthrough term of the jth disturbance input on the ith output)

Generalized state-space model

$$
\begin{aligned}
\dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}_u\mathbf{u}(t) + \mathbf{B}_d\mathbf{d}(t) \\
\mathbf{y}(t) &= \mathbf{C}_y\mathbf{x}(t) + \mathbf{D}_{yu}\mathbf{u}(t) + \mathbf{D}_{yd}\mathbf{d}(t) + \mathbf{D}_{yr}\mathbf{r}(t) \\
\mathbf{z}(t) &= \mathbf{C}_z\mathbf{x}(t) + \mathbf{D}_{zu}\mathbf{u}(t) + \mathbf{D}_{zd}\mathbf{d}(t)
\end{aligned}
$$

- $\mathbf{D}_{yr}$ is the $n_y \times n_r$ noise matrix (the element (i, j) relates the jth noise variable on the ith output variable)
- $\mathbf{C}_z$ is the $n_z \times n_x$ performance matrix (the ith row is referred to the ith performance variable)
- $\mathbf{D}_{zu}$ is the $n_z \times n_u$ direct feedthrough matrix of the performance (the element (i, j) is the direct feedthrough term of the jth control input on the ith performance variable)
- $\mathbf{D}_{zd}$ is the $n_z \times n_d$ disturbance direct feedthrough matrix of the performance (the element (i, j) is the direct feedthrough term of the jth disturbance input on the ith performance variable)

Generalized state-space model

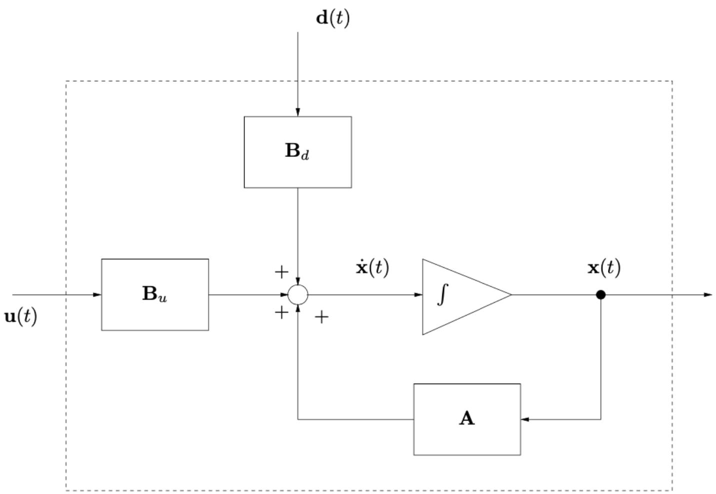


Figure 14.2 Block diagram representation of the state equation.

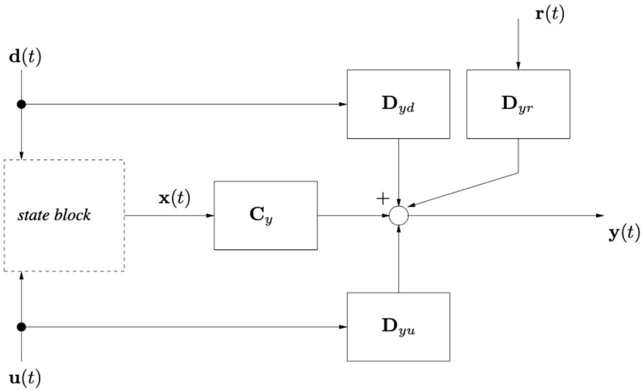


Figure 14.3 Block diagram representation of the output equation.

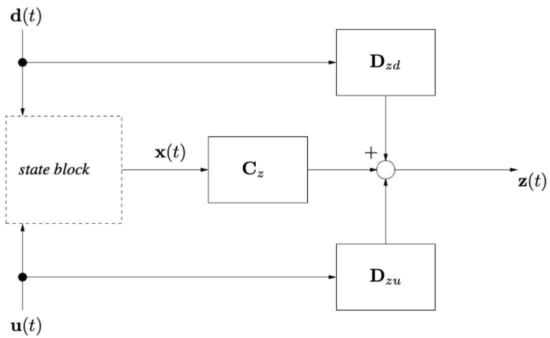
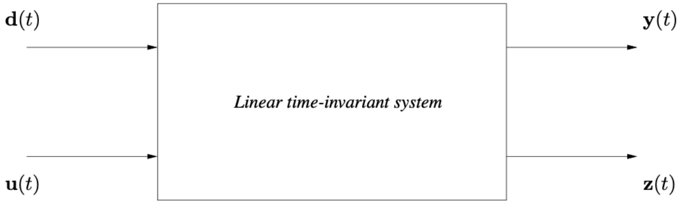


Figure 14.4 Block diagram representation of the performance equation.

State-space control design is the technique in which the control engineer designs a dynamic compensation by working directly with the state-variable description of the system.

Use of the state-space approach has often been referred to as **modern control design**, and use of transfer-function-based methods, such as root locus and frequency response, referred to as classical control design.

Advantages of state-space design are especially apparent when the system to be controlled has more than one control input or more than one sensed output (MIMO).



Control system design – state-space methods
Introduction

- We design the control as if all of the state were measured and available for use in the control law. This provides the possibility of assigning arbitrary dynamics for the system.
- Having a satisfactory control law based on full-state feedback, we introduce the concept of an observer and construct estimates of the state based on the sensed output.
- We then show that these estimates can be used in place of the actual state-variables.
- Finally, we introduce the external reference-command inputs to complete the structure.
- At this point we can recognize that the resulting compensation has the same essential structure as that developed with transform methods.

Before we begin the design using state descriptions,
it is necessary to present develop some results and tools for use throughout this part.

Control system design – state-space methods
State space fundamentals

For the sake of convenience, the generalized form of the state and output equations reported in Eq. (14.58) will be simplified in the following by grouping the control, disturbance and noise inputs into a single input vector denoted as $\mathbf{u}(t)$. Accordingly, the state equation will have a single $\mathbf{B}$ input matrix, without any distinction between control and disturbance related terms, and the output equation will be written in terms of the output matrix $\mathbf{C}$ and a single direct feedthrough matrix $\mathbf{D}$.

$\mathbf{B}_u$ or $\mathbf{B}_d$

$\mathbf{u}(t)$ $\longrightarrow$ $\boxed{\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t)\\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t)\end{aligned}}$ $\longrightarrow$ $\mathbf{u}(t)$

control inputs or disturbance inputs

sensor measurements or performance metrics

$\mathbf{C}_y$ or $\mathbf{C}_z$

$\mathbf{D}_{yu}$ or $\mathbf{D}_{yd}$

$\mathbf{D}_{zu}$ or $\mathbf{D}_{zd}$

Homogeneous solution

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t)$$
$$\mathbf{x}(t_0) = \mathbf{x}_0$$

$$e^{\mathbf{A}t} = \mathbf{I} + \mathbf{A}t + \mathbf{A}^2\frac{t^2}{2} + \mathbf{A}^3\frac{t^3}{3!} + \ldots$$

(exponential matrix)

The solution can be expressed as $\mathbf{x}(t) = e^{\mathbf{A}t}\mathbf{c}$

Putting into the initial condition $\mathbf{x}_0 = e^{\mathbf{A}t_0}\mathbf{c}$ yields $\mathbf{c} = \left(e^{\mathbf{A}t_0}\right)^{-1}\mathbf{x}_0 = e^{-\mathbf{A}t_0}\mathbf{x}_0$

Therefore the solution is

$$\boxed{\mathbf{x}(t) = e^{\mathbf{A}(t-t_0)}\mathbf{x}_0}$$

$$\boxed{\mathbf{y}(t) = \mathbf{C}e^{\mathbf{A}(t-t_0)}\mathbf{x}_0}$$

Complete solution

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t)$$

$$\mathbf{x}(t_0) = \mathbf{x}_0$$

*now the vector **c** is function of time*

$\dot{x}(t) = Ax(t) + Bu(t)$

$\dot{x}(t) = Ax(t)$ → homogeneous solution

Let's assume a solution of the form $$\mathbf{x}(t) = e^{\mathbf{A}t}\mathbf{c}(t)$$

$x(t) = ce^{At}$ general

Substituting this yields $$\mathbf{A}e^{\mathbf{A}t}\mathbf{c}(t) + e^{\mathbf{A}t}\dot{\mathbf{c}}(t) = \mathbf{A}e^{\mathbf{A}t}\mathbf{c}(t) + \mathbf{B}\mathbf{u}(t)$$

$x_0 = ce^{At_0}$ $c = e^{-At_0}x_0$

$$\dot{\mathbf{c}}(t) = e^{-\mathbf{A}t}\mathbf{B}\mathbf{u}(t)$$

$x(t) = e^{A(t-t_0)}x_0$ → homogeneous solution

Particular solution

→ $\dot{x}(t) = Ax(t) + Bu(t)$ $x(t) = e^{A(t-t_0)}c(t)$

$c(t)$ now function of time !

we insert it in

$$\boxed{\mathbf{c}(t) = \int_{\tilde{t}}^{t} e^{-\mathbf{A}\tau}\mathbf{B}\mathbf{u}(\tau)d\tau}$$

the lower limit will be specified later

Complete solution

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t)$$

$$\mathbf{x}(t_0) = \mathbf{x}_0$$

The particular solution has the form

$$\mathbf{x}(t) = e^{\mathbf{A}t} \int_{\tilde{t}}^{t} e^{-\mathbf{A}\tau} \mathbf{B}\mathbf{u}(\tau) d\tau$$
$$= \int_{\tilde{t}}^{t} e^{\mathbf{A}(t-\tau)} \mathbf{B}\mathbf{u}(\tau) d\tau$$

By superposition, the complete solution is given by

$$\mathbf{x}(t) = e^{\mathbf{A}(t-t_0)}\mathbf{x}_0 + \int_{\tilde{t}}^{t} e^{\mathbf{A}(t-\tau)} \mathbf{B}\mathbf{u}(\tau) d\tau$$

For $t = t_0$

$$\mathbf{x}(t_0) = \mathbf{x}_0 + \int_{\tilde{t}}^{t_0} e^{\mathbf{A}(t_0-\tau)} \mathbf{B}\mathbf{u}(\tau) d\tau$$

must be equal to zero, therefore

$$\tilde{t} = t_0$$

Complete solution

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t)$$

$$\mathbf{x}(t_0) = \mathbf{x}_0$$

The complete solution has the form

$$\boxed{\mathbf{x}(t) = e^{\mathbf{A}(t-t_0)}\mathbf{x}_0 + \int_{t_0}^{t} e^{\mathbf{A}(t-\tau)}\mathbf{B}\mathbf{u}(\tau)d\tau}$$

$$\boxed{\mathbf{y}(t) = \mathbf{C}e^{\mathbf{A}(t-t_0)}\mathbf{x}_0 + \int_{t_0}^{t} \mathbf{C}e^{\mathbf{A}(t-\tau)}\mathbf{B}\mathbf{u}(\tau)d\tau + \mathbf{D}\mathbf{u}(t)}$$

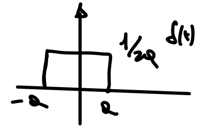
general solution of the time variant system.

convolution integral

Impulse response
($\mathbf{x}_0 = \mathbf{0}$, $t_0 = 0$)

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\boldsymbol{\delta}(t)$$

↳ impulse →



$\delta(t)$ è lim $a \to 0$

$$\begin{cases} 0 & t < -a \\ \frac{1}{2a} & -a \leq t \leq a \\ 0 & t > a \end{cases}$$

The impulse response has the form

$$\mathbf{h}(t) = \int_0^t \mathbf{C}e^{\mathbf{A}(t-\tau)}\mathbf{B}\boldsymbol{\delta}(\tau)d\tau + \mathbf{D}\boldsymbol{\delta}(t)$$

Using the sampling property of the Dirac delta function

$$\mathbf{h}(t) = \mathbf{C}e^{\mathbf{A}t}\mathbf{B} + \mathbf{D}\boldsymbol{\delta}(t)$$

For systems without the direct feedthrough matrix (D = 0)

$$\mathbf{h}(t) = \mathbf{C}e^{\mathbf{A}t}\mathbf{B}$$

Transfer function matrix

The Laplace transform of the state equation is given by

$$s\mathbf{x}(s) = \mathbf{A}\mathbf{x}(s) + \mathbf{B}\mathbf{u}(s)$$

*recall:
transfer function is evaluated for zero initial conditions*

Therefore, the state can be written as a function of the input vector as

$$\mathbf{x}(s) = (s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B}\mathbf{u}(s)$$

Since the Laplace transform of the output equation is given by

$$\mathbf{y}(s) = \mathbf{C}\mathbf{x}(s) + \mathbf{D}\mathbf{u}(s)$$

we have the following input-output relation

$$\mathbf{y}(s) = \mathbf{H}(s)\mathbf{u}(s)$$

where

$$\boxed{\mathbf{H}(s) = \mathbf{C}\,(s\mathbf{I} - \mathbf{A})^{-1}\,\mathbf{B} + \mathbf{D}}$$

$n_y \times n_u$ matrix

↳ mapping of the input to the output → Transfer function matrix proper system = power of term s numerator is equal to the power of term s at the denominator

strictly proper system = power of terms s at the numerator is lower than the power of term s at the denominator.

Transfer function matrix

Note that $\mathbf{h}(t) = \mathbf{C}e^{\mathbf{A}t}\mathbf{B} + \mathbf{D}\boldsymbol{\delta}(t)$

$$\mathbf{H}(s) = \mathbf{C}\left(s\mathbf{I} - \mathbf{A}\right)^{-1}\mathbf{B} + \mathbf{D}$$

$$\mathcal{L}^{-1}\left[\mathbf{H}(s)\right] = \mathbf{h}(t)$$

to compute the poles of the system from state equations

$$\det(s\mathbf{I} - \mathbf{A}) = 0$$

For systems without the direct feedthrough matrix (D = 0)

$$\mathbf{H}(s) = \mathbf{C}\left(s\mathbf{I} - \mathbf{A}\right)^{-1}\mathbf{B}$$

In this case, since the determinant of $s\mathbf{I} - \mathbf{A}$ is of degree n_x, where n_x is the number of state variables, and the adjoint matrix of $s\mathbf{I} - \mathbf{A}$ is of degree $n_x - 1$, it follows that each element of the transfer function matrix is a strictly proper transfer function, i.e., a rational function of s with the numerator of degree $n_x - 1$ (or less) and the denominator of degree n_x.

Transformation of state variables

The same physical system can have many different state space formulations.

Here, we will briefly present the theory related to the transformation of state variables and explain why all the state space formulations are equivalent, in the sense that they share the same input-output relations and, hence, they represent the same physical system.

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

A change of state variables is represented by the following linear transformation:

T must be not singular

$$\mathbf{z}(t) = \mathbf{T}\mathbf{x}(t)$$

where z is the new set of states and T is a non-singular transformation matrix

$$\mathbf{x}(t) = \mathbf{T}^{-1}\mathbf{z}(t)$$

Transformation of state variables

$$\mathbf{z}(t) = \mathbf{T}\mathbf{x}(t)$$

$$\mathbf{x}(t) = \mathbf{T}^{-1}\mathbf{z}(t)$$

Substituting into the state space model yields

$$\mathbf{T}^{-1}\dot{\mathbf{z}}(t) = \mathbf{A}\mathbf{T}^{-1}\mathbf{z}(t) + \mathbf{B}\mathbf{u}(t)$$
$$\mathbf{y}(t) = \mathbf{C}\mathbf{T}^{-1}\mathbf{z}(t) + \mathbf{D}\mathbf{u}(t)$$

Rearranging

$$\dot{\mathbf{z}}(t) = \hat{\mathbf{A}}\mathbf{z}(t) + \hat{\mathbf{B}}\mathbf{u}(t)$$
$$\mathbf{y}(t) = \hat{\mathbf{C}}\mathbf{z}(t) + \hat{\mathbf{D}}\mathbf{u}(t)$$

$$\hat{\mathbf{A}} = \mathbf{T}\mathbf{A}\mathbf{T}^{-1} \qquad \hat{\mathbf{B}} = \mathbf{T}\mathbf{B} \qquad \hat{\mathbf{C}} = \mathbf{C}\mathbf{T}^{-1} \qquad \hat{\mathbf{D}} = \mathbf{D}$$

new matrices of the transformed state space representation

Transformation of state variables

The input-output relation of the new state formulation is represented by the corresponding transfer function

$$
\begin{aligned}
\hat{\mathbf{H}}(s) &= \hat{\mathbf{C}}\left(s\mathbf{I}-\hat{\mathbf{A}}\right)^{-1}\hat{\mathbf{B}}+\hat{\mathbf{D}} \\
&= \mathbf{C}\mathbf{T}^{-1}\left(s\mathbf{I}-\mathbf{T}\mathbf{A}\mathbf{T}^{-1}\right)^{-1}\mathbf{T}\mathbf{B}+\mathbf{D} \\
&= \mathbf{C}\mathbf{T}^{-1}\left(s\mathbf{T}\mathbf{T}^{-1}-\mathbf{T}\mathbf{A}\mathbf{T}^{-1}\right)^{-1}\mathbf{T}\mathbf{B}+\mathbf{D} \\
&= \mathbf{C}\mathbf{T}^{-1}\left[\mathbf{T}\left(s\mathbf{I}-\mathbf{A}\right)\mathbf{T}^{-1}\right]^{-1}\mathbf{T}\mathbf{B}+\mathbf{D} \\
&= \mathbf{C}\mathbf{T}^{-1}\mathbf{T}\left(s\mathbf{I}-\mathbf{A}\right)^{-1}\mathbf{T}^{-1}\mathbf{T}\mathbf{B}+\mathbf{D} \\
&= \mathbf{C}\left(s\mathbf{I}-\mathbf{A}\right)^{-1}\mathbf{B}+\mathbf{D}
\end{aligned}
$$

Therefore, the transfer function matrix of the new state formulation is equal to the transfer function matrix of the original formulation.

This implies that the input-output relations of a LTI system do not depend on how the state variables are defined.

Transformation of state variables

Note also that the original matrix $\mathbf{A}$ and the transformed matrix $\hat{\mathbf{A}}$ are similar. Indeed, they have the same eigenvalues as shown in the following

$$\begin{aligned}
\det\left(s\mathbf{I}-\hat{\mathbf{A}}\right) &= \det\left(s\mathbf{I}-\mathbf{T}\mathbf{A}\mathbf{T}^{-1}\right)\\
&= \det\left(s\mathbf{T}\mathbf{T}^{-1}-\mathbf{T}\mathbf{A}\mathbf{T}^{-1}\right)\\
&= \det\left[\mathbf{T}\left(s\mathbf{I}-\mathbf{A}\right)\mathbf{T}^{-1}\right]\\
&= \det(\mathbf{T})\det\left(s\mathbf{I}-\mathbf{A}\right)\det(\mathbf{T}^{-1})\\
&= \det(\mathbf{T})\det\left(s\mathbf{I}-\mathbf{A}\right)\frac{1}{\det(\mathbf{T})}\\
&= \det\left(s\mathbf{I}-\mathbf{A}\right)
\end{aligned}$$

State space realization (canonical form)

We have shown how to determine the transfer function of a LTI system, given its state space representation. In many cases, it is necessary to go in the opposite direction, i.e., derive the state space form from the input-output relation. This need arises if one wants to use state space methods but one or more subsystems within larger systems are described by a transfer function model.

Let first consider a general transfer function of a single-input, single-output system

$$H(s) = \frac{b_0 s^n + b_1 s^{n-1} + \cdots + b_n}{s^n + a_1 s^{n-1} + \cdots + a_n}$$

After the introduction of the intermediate variable $z(s)$

$$H(s) = \frac{y(s)}{u(s)} = \frac{y(s)}{z(s)}\frac{z(s)}{u(s)}$$

$$\frac{y(s)}{z(s)} = b_0 s^n + b_1 s^{n-1} + \cdots + b_n$$

$$\frac{z(s)}{u(s)} = \frac{1}{s^n + a_1 s^{n-1} + \cdots + a_n}$$

State space realization (canonical form)

$$\frac{z(s)}{u(s)} = \frac{1}{s^n + a_1 s^{n-1} + \cdots + a_n}$$

$$\left(s^n + a_1 s^{n-1} + \cdots + a_n\right) z(s) = u(s)$$

$$\frac{d^n}{dt^n} z(t) + a_1 \frac{d^{n-1}}{dt^{n-1}} z(t) + \cdots + a_n z(t) = u(t)$$

$$\frac{y(s)}{z(s)} = b_0 s^n + b_1 s^{n-1} + \cdots + b_n$$

$$y(s) = \left(b_0 s^n + b_1 s^{n-1} + \cdots + b_n\right) z(s)$$

$$y(t) = b_0 \frac{d^n}{dt^n} z(t) + b_1 \frac{d^{n-1}}{dt^{n-1}} z(t) + \cdots + b_n z(t)$$

State space realization (canonical form)

$$\frac{d^n}{dt^n}z(t) + a_1\frac{d^{n-1}}{dt^{n-1}}z(t) + \cdots + a_n z(t) = u(t)$$

$$y(t) = b_0\frac{d^n}{dt^n}z(t) + b_1\frac{d^{n-1}}{dt^{n-1}}z(t) + \cdots + b_n z(t)$$

defining the following state variables

$$x_1(t) = z(t)$$

$$x_2(t) = \frac{d}{dt}z(t) = \dot{x}_1(t)$$

$$x_3(t) = \frac{d^2}{dt^2}z(t) = \dot{x}_2(t)$$

$$\vdots$$

$$x_n(t) = \frac{d^{n-1}}{dt^{n-1}}z(t) = \dot{x}_{n-1}(t)$$

$$\begin{Bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \vdots \\ \dot{x}_{n-1}(t) \\ \dot{x}_n(t) \end{Bmatrix} = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ & & \dots & & \\ 0 & 0 & 0 & \dots & 1 \\ -a_n & -a_{n-1} & -a_{n-2} & \dots & -a_1 \end{bmatrix} \begin{Bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_{n-1}(t) \\ x_n(t) \end{Bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} u(t)$$

State space realization (canonical form)

By substituting $$\frac{d^n}{dt^n}z(t) + a_1\frac{d^{n-1}}{dt^{n-1}}z(t) + \cdots + a_n z(t) = u(t)$$

into $$y(t) = b_0\frac{d^n}{dt^n}z(t) + b_1\frac{d^{n-1}}{dt^{n-1}}z(t) + \cdots + b_n z(t)$$

yields $$\begin{aligned} y(t) &= b_0\left[-a_1\frac{d^{n-1}}{dt^{n-1}}z(t) - \cdots - a_n z(t) + u(t)\right] + b_1\frac{d^{n-1}}{dt^{n-1}}z(t) + \cdots + b_n z(t) \\ &= (b_1 - a_1 b_0)\frac{d^{n-1}}{dt^{n-1}}z(t) + (b_2 - a_2 b_0)\frac{d^{n-2}}{dt^{n-2}}z(t) + \cdots + (b_n - a_n b_0)\,z(t) + b_0 u(t) \end{aligned}$$

$$y(t) = \begin{bmatrix} b_n - a_n b_0 & b_{n-1} - a_{n-1}b_0 & \ldots & b_2 - a_2 b_0 & b_1 - a_1 b_0 \end{bmatrix} \begin{Bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_{n-1}(t) \\ x_n(t) \end{Bmatrix} + b_0 u(t)$$

State space realization (controller canonical form)

if the function is strictly proper $b_0 = \emptyset$

$$H(s) = \frac{b_0 s^n + b_1 s^{n-1} + \cdots + b_n}{s^n + a_1 s^{n-1} + \cdots + a_n}$$

$\mathbf{A}$ $\mathbf{B}$

state equation

$$\begin{Bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \vdots \\ \dot{x}_{n-1}(t) \\ \dot{x}_n(t) \end{Bmatrix} = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ & & \dots & & \\ 0 & 0 & 0 & \dots & 1 \\ -a_n & -a_{n-1} & -a_{n-2} & \dots & -a_1 \end{bmatrix} \begin{Bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_{n-1}(t) \\ x_n(t) \end{Bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} u(t)$$

$\mathbf{A}$

output equation

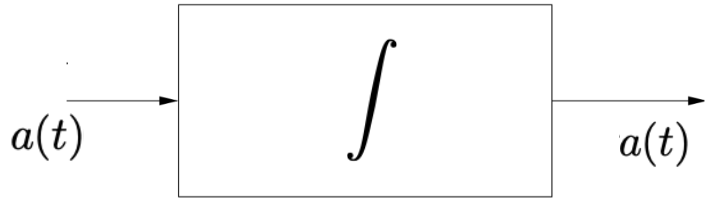
$$y(t) = \begin{bmatrix} b_n - a_n b_0 & b_{n-1} - a_{n-1} b_0 & \dots & b_2 - a_2 b_0 & b_1 - a_1 b_0 \end{bmatrix} \begin{Bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_{n-1}(t) \\ x_n(t) \end{Bmatrix} + b_0 u(t)$$

J_0

Examples
(state space realization)

Ideal integrator

$$\boxed{H(s) = \frac{v(s)}{a(s)} = \frac{1}{s}}$$



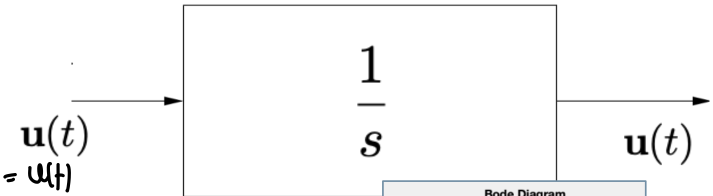
accelerometer providing an acceleration measurement

you want to have the corresponding velocity

$$u(t) = a(t)$$
$$y(t) = v(t)$$

$\frac{y(t)}{z(t)} = 1$

$\frac{z(t)}{u(t)} = \frac{1}{s}$



$z(t)\, s = u(t)$

$\dot{z}(t) = u(t)$ $\quad y(t) = z(t)$

$z(t) = x_1$

$\dot{z}(t) = \dot{x}_1$

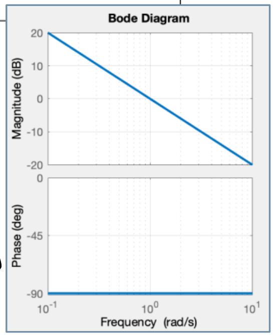
$$b_0 = \cdots = b_{n-1} = 0$$
$$b_n = 1$$
$$a_1 = \cdots = a_n = 0$$

$$\boxed{\begin{aligned}\dot{x}_1(t) &= u(t)\\ y(t) &= x_1(t)\end{aligned}}$$

$\dot{x}_1 = u(t)$

$\underbrace{[0]}_{A}x + \underbrace{[1]}_{B}u$

$\underbrace{[1]}_{C}x + \underbrace{[0]}_{D}$



Examples
(state space realization)

Ideal integrator

$$H(s) = \frac{v(s)}{a(s)} = \frac{1}{s}$$

$$\begin{aligned} \dot{x}_1(t) &= u(t) \\ y(t) &= x_1(t) \end{aligned}$$

MATLAB code 😉

```
>> s=tf('s');
>> H=1/s;
>> sys=ss(H);
>> sys.A

ans =

     0

>> sys.B

ans =

     1

>> sys.C

ans =

     1

>> sys.D

ans =

     0
```

*. matlab does that automatically.

tf

Create transfer function model, convert to transfer function model

ss

Create state-space model, convert to state-space model

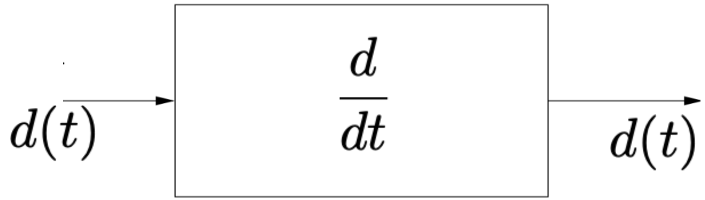
Description

Use ss to create state-space models (ss model objects) with real- or complex-valued matrices or to convert dynamic system models to state-space model form. You can also use ss to create Generalized state-space (genss) models.

Examples
(state space realization)

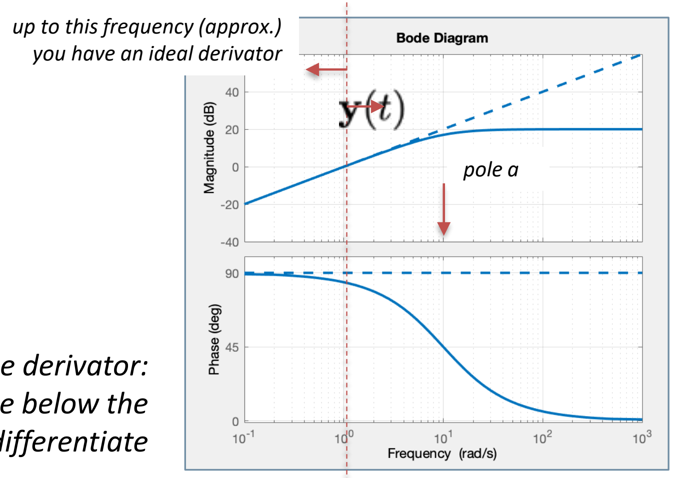
Real derivator

$$H(s) = \frac{v(s)}{d(s)} = \frac{as}{s+a}$$

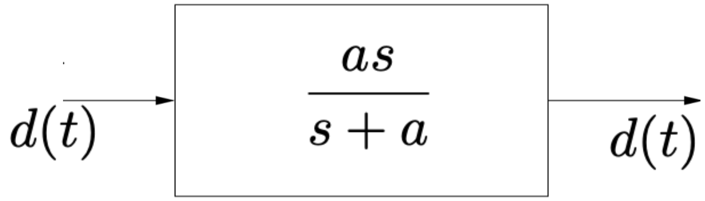


sensor providing a displacement measurement

you want to have the corresponding velocity



Design of the derivator:
select the pole (approx.) one decade below the frequency where you want to stop to differentiate



Examples
(state space realization)

Real derivator

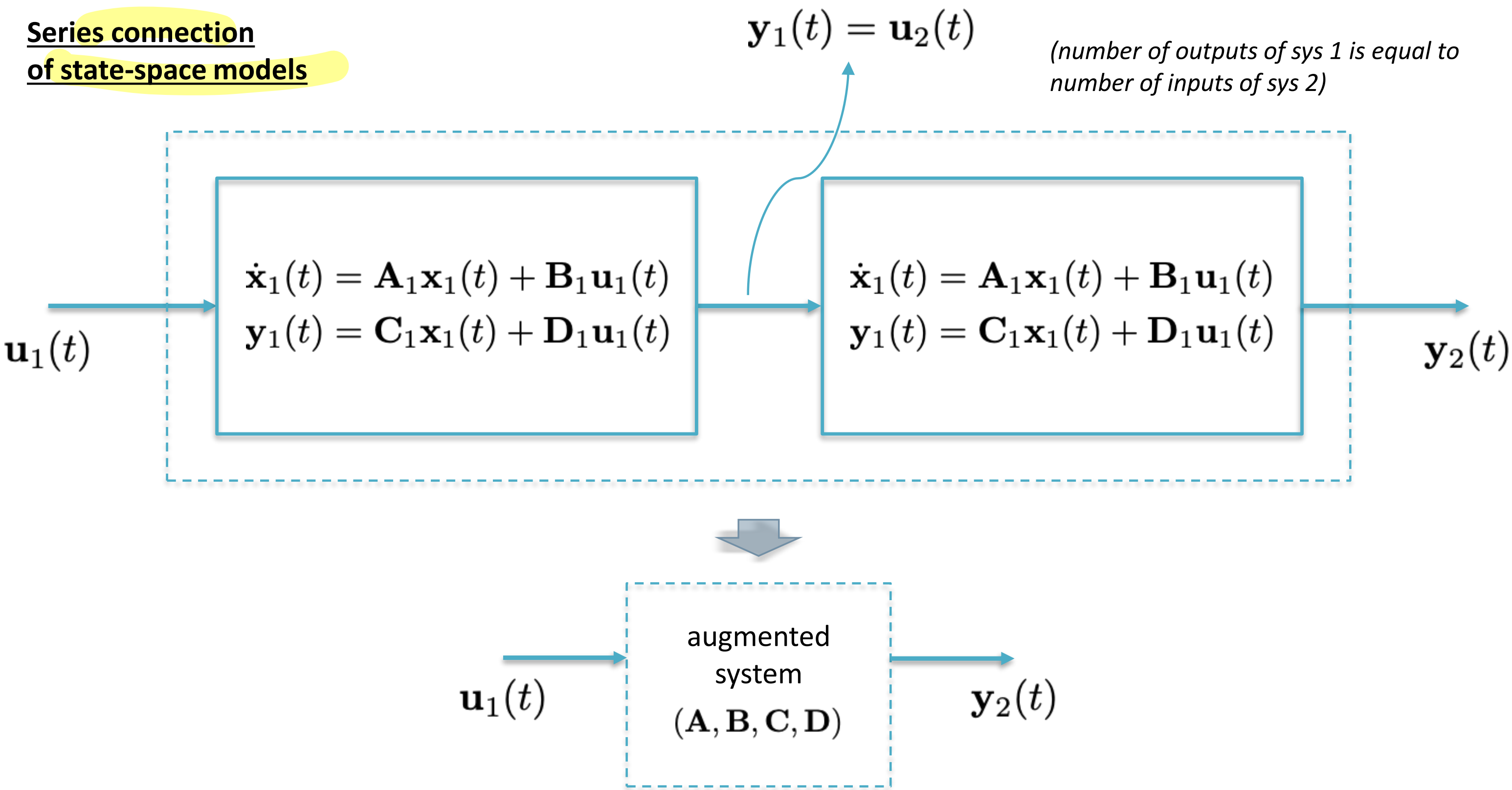
$$H(s) = \frac{v(s)}{d(s)} = \frac{as}{s+a}$$

$$\frac{v(s)}{d(s)} = \frac{v(s)}{z(s)}\frac{z(s)}{d(s)} = \frac{as}{s+a}$$

$$\frac{z(s)}{d(s)} = \frac{1}{s+a} \quad \Rightarrow \quad \dot{z}(t) = \overset{A}{-a}z(t) + \overset{B}{d}(t)$$

$$\frac{v(s)}{z(s)} = as \quad \Rightarrow \quad v(t) = a\dot{z}(t) = \overset{J_0}{-a^2}z(t) + \overset{A}{a}d(t)$$

Series connection of state-space models



Series connection of state-space models

Since $\mathbf{y}_1(t) = \mathbf{u}_2(t)$

substituting…

$$\dot{\mathbf{x}}_1(t) = \mathbf{A}_1\mathbf{x}_1(t) + \mathbf{B}_1\mathbf{u}_1(t)$$
$$\boxed{\mathbf{y}_1(t) = \mathbf{C}_1\mathbf{x}_1(t) + \mathbf{D}_1\mathbf{u}_1(t)}$$

$$\dot{\mathbf{x}}_1(t) = \mathbf{A}_1\mathbf{x}_1(t) + \mathbf{B}_1\mathbf{u}_1(t)$$
$$\mathbf{y}_1(t) = \mathbf{C}_1\mathbf{x}_1(t) + \mathbf{D}_1\mathbf{u}_1(t)$$

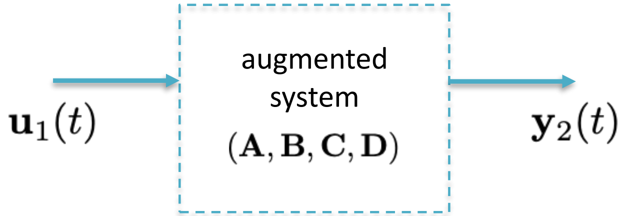
substituting…

yields

$$\dot{\mathbf{x}}_2(t) = \mathbf{A}_2\mathbf{x}_2(t) + \mathbf{B}_2\left[\mathbf{C}_1\mathbf{x}_1(t) + \mathbf{D}_1\mathbf{u}_1(t)\right]$$
$$\mathbf{y}_2(t) = \mathbf{C}_2\mathbf{x}_2(t) + \mathbf{D}_2\left[\mathbf{C}_1\mathbf{x}_1(t) + \mathbf{D}_1\mathbf{u}_1(t)\right]$$

Control system design – state-space methods
State space fundamentals

Series connection of state-space models



Defining an "augmented" state vector collecting $\mathbf{x}_1$ and $\mathbf{x}_2$ as follows

$$\mathbf{x}(t) = \begin{Bmatrix} \mathbf{x}_1(t) \\ \mathbf{x}_2(t) \end{Bmatrix}$$

the state dynamics is governed by

$$\dot{\mathbf{x}}(t) = \begin{Bmatrix} \dot{\mathbf{x}}_1(t) \\ \dot{\mathbf{x}}_2(t) \end{Bmatrix} = \overbrace{\begin{bmatrix} \mathbf{A}_1 & \mathbf{0} \\ \mathbf{B}_2\mathbf{C}_1 & \mathbf{A}_2 \end{bmatrix}}^{\mathbf{A}} \mathbf{x}(t) + \overbrace{\begin{bmatrix} \mathbf{B}_1 \\ \mathbf{B}_2\mathbf{D}_1 \end{bmatrix}}^{\mathbf{B}} \mathbf{u}_1(t)$$

and the output equation is

$$\mathbf{y}_2(t) = \underbrace{\begin{bmatrix} \mathbf{D}_2\mathbf{C}_1 & \mathbf{C}_2 \end{bmatrix}}_{J_0} \mathbf{x}(t) + \underbrace{\mathbf{D}_2\mathbf{D}_1}_{\mathbf{A}} \mathbf{u}_1(t)$$

Series connection of state-space models

In MATLAB…

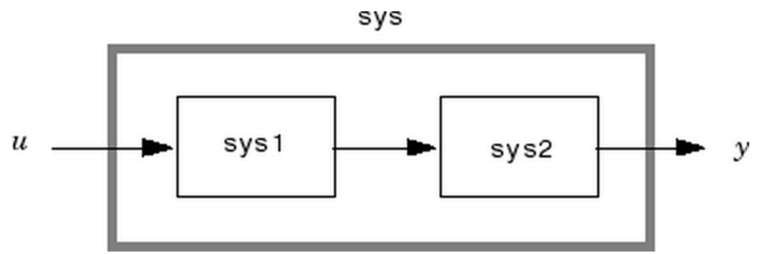
series

Series connection of two models

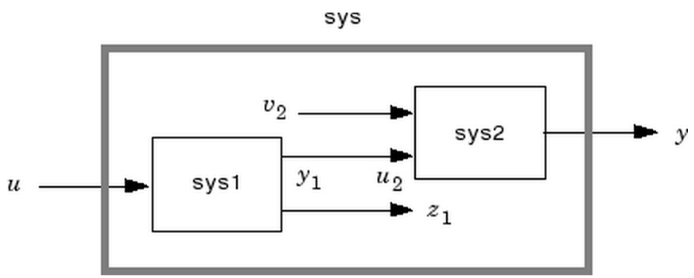
Syntax

```
series
sys = series(sys1,sys2)
sys = series(sys1,sys2,outputs1,inputs2)
```

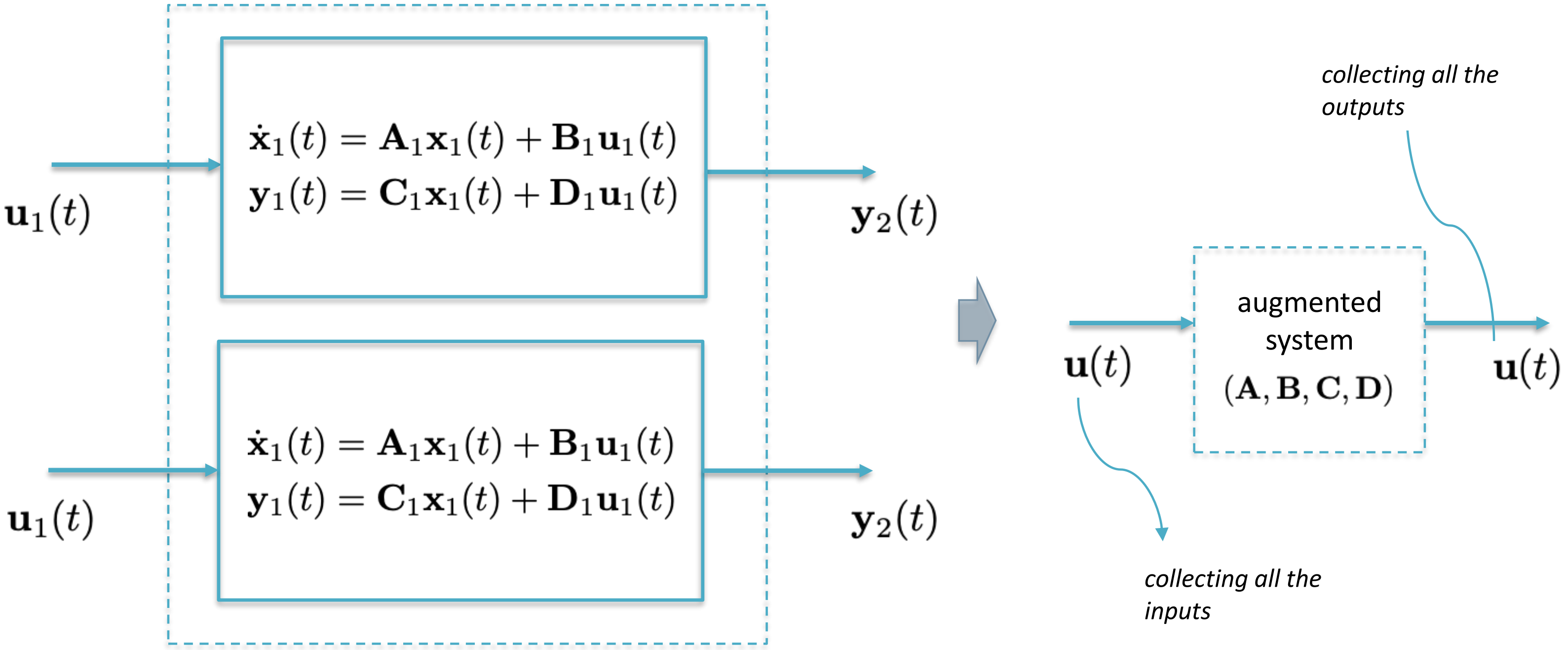
`sys = series(sys1,sys2)` forms the basic series connection shown below.



`sys = series(sys1,sys2,outputs1,inputs2)` forms the more general series connection.



Parallel connection of state-space models



Parallel connection of state-space models

Defining an "augmented" state vector collecting $\mathbf{x}_1$ and $\mathbf{x}_2$ as follows

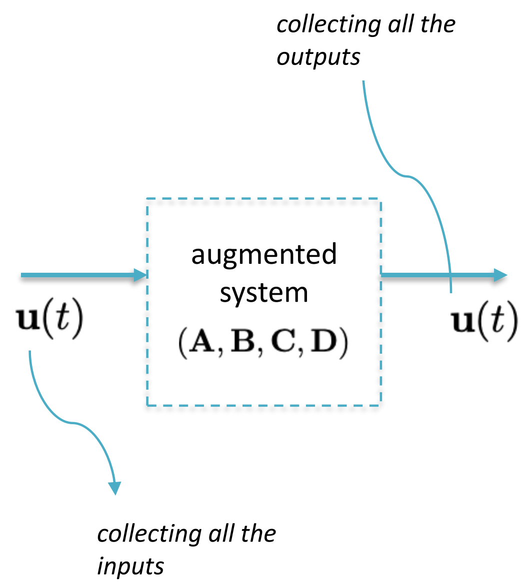
$$\mathbf{x}(t) = \begin{Bmatrix} \mathbf{x}_1(t) \\ \mathbf{x}_2(t) \end{Bmatrix}$$

Defining an "augmented" input vector collecting $\mathbf{u}_1$ and $\mathbf{u}_2$ as follows

$$\mathbf{x}(t) = \begin{Bmatrix} \mathbf{x}_1(t) \\ \mathbf{x}_2(t) \end{Bmatrix}$$

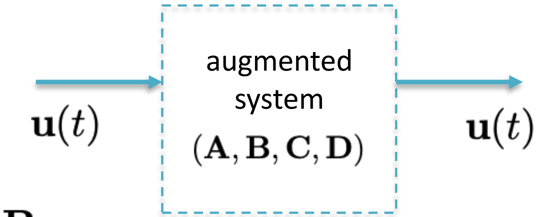
Defining an "augmented" output vector collecting $\mathbf{y}_1$ and $\mathbf{y}_2$ as follows

$$\mathbf{x}(t) = \begin{Bmatrix} \mathbf{x}_1(t) \\ \mathbf{x}_2(t) \end{Bmatrix}$$



Control system design – state-space methods
State space fundamentals

<u>Parallel connection of state-space models</u>



the state dynamics is governed by

$$\dot{\mathbf{x}}(t) = \begin{Bmatrix} \dot{\mathbf{x}}_1(t) \\ \dot{\mathbf{x}}_2(t) \end{Bmatrix} = \overbrace{\begin{bmatrix} \mathbf{A}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_2 \end{bmatrix}}^{\mathbf{A}} \mathbf{x}(t) + \overbrace{\begin{bmatrix} \mathbf{B}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{B}_2 \end{bmatrix}}^{\mathbf{B}} \mathbf{u}(t)$$

and the output equation is

$$\mathbf{y}(t) = \overbrace{\begin{bmatrix} \mathbf{C}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_2 \end{bmatrix}}^{J_0} \mathbf{x}(t) + \overbrace{\begin{bmatrix} \mathbf{D}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_2 \end{bmatrix}}^{\mathbf{A}} \mathbf{u}(t)$$

Parallel connection of state-space models

In MATLAB…

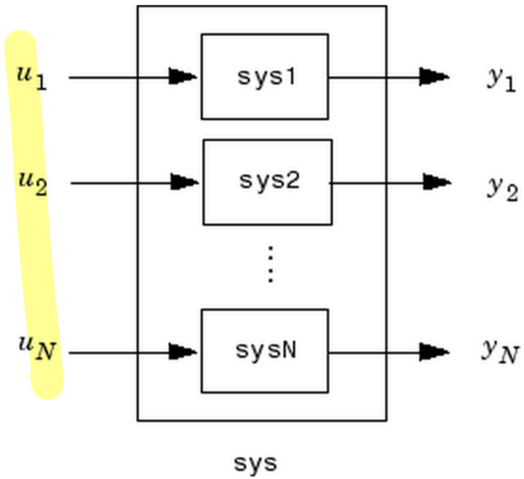
append

Group models by appending their inputs and outputs

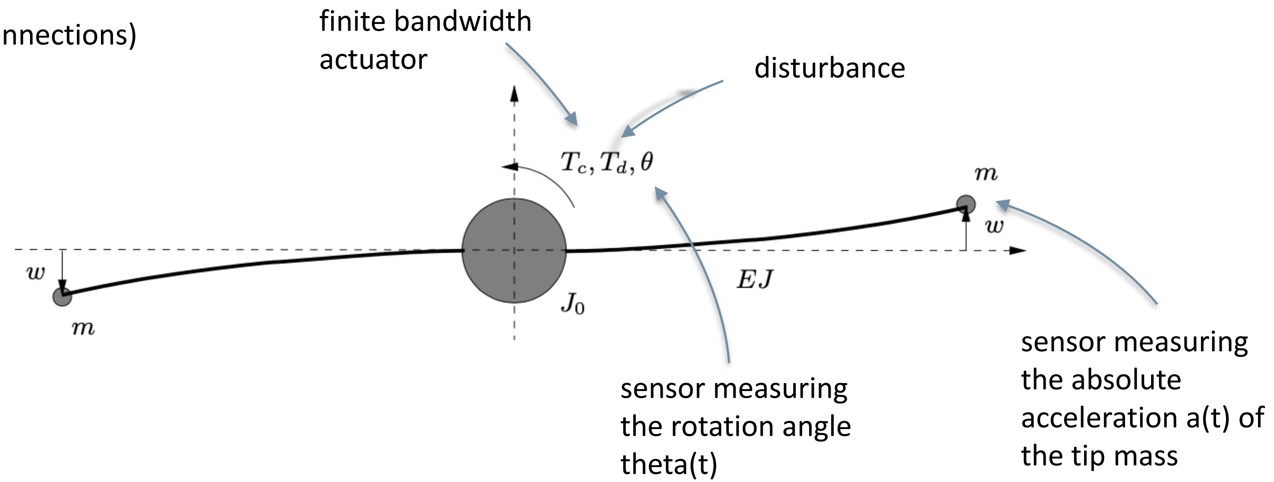
Syntax

```
sys = append(sys1,sys2,...,sysN)
```

`sys = append(sys1,sys2,...,sysN)` appends the inputs and outputs of the models `sys1`,...,`sysN` to form the augmented model `sys` depicted below.



<u>Exercise</u>
(series/parallel connections)



Differential problem:

$$J\ddot{\theta} + 2m\ell\ddot{w} = T_c + T_d$$

$$2m\ddot{w} + 2m\ell\ddot{\theta} + 2c_a\dot{w} + 2k_a w = 0$$

<u>Exercise</u>
(series/parallel connections)

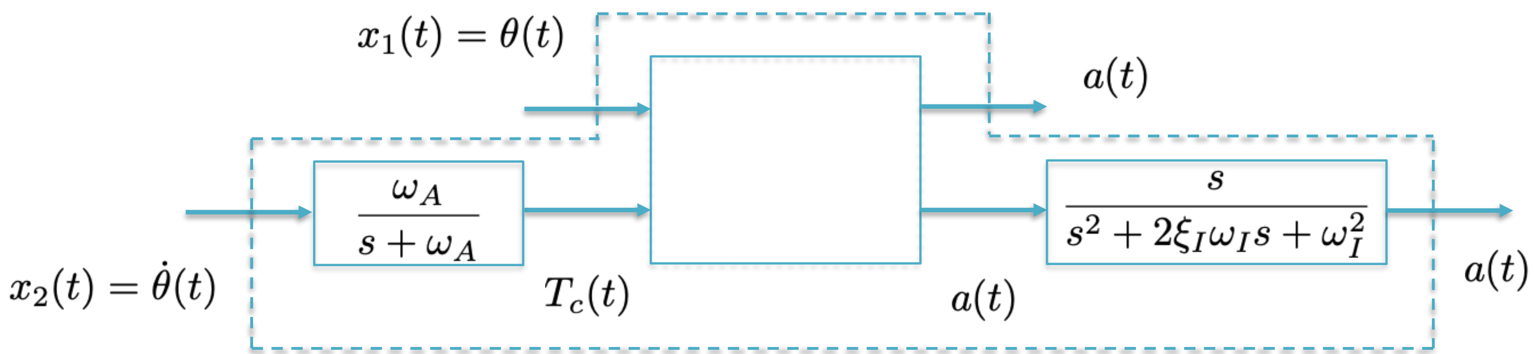
<u>Step 1</u>. Derive the state-space model with control input Tc(t), disturbance input Td(t), and outputs theta(t) and a(t)

$$\mathbf{x}(t) = \begin{Bmatrix} \theta(t) \\ w(t) \\ \dot{\theta}(t) \\ \dot{w}(t) \end{Bmatrix}$$

Exercise
(series/parallel connections)

Step 2. Derive the state-space model by considering the finite bandwidth of the actuator and the output given by theta(t) and the velocity v(t) of the tip mass obtained using an integrator with removal of DC offset.

$$\mathbf{y}(t) = \begin{Bmatrix} \theta(t) \\ v(t) \end{Bmatrix}$$



The idea of controllability

In the following we want to explore the input-to-state interaction of the n-dimensional linear time-invariant state equation, seeking to characterize the extent to which state trajectories can be controlled by the input signal.

Specifically, we are interested in deriving conditions under which, starting anywhere in the state space, the state trajectory can be driven to the origin by piecewise continuous input signals over a finite time interval.

More generally, it turns out that if it is possible to do this, then it is also possible to steer the state trajectory to any final state in finite time via a suitable input signal.

The idea of observability

In our state-space description of linear time-invariant systems, the state vector constitutes an internal quantity that is influenced by the input signal and, in turn, affects the output signal.

It is generally true in practice that the dimension of the state vector, equivalently the dimension of the system modeled by the state equation, is greater than the number of input signals or output signals.
This reflects the fact that the complexity of real-world systems precludes the ability to directly actuate or sense each state variable.

Nevertheless, we are often interested in somehow estimating the state vector because it characterizes the system's complex inner workings and, as we shall see later, figures prominently in state-space methods of control system design.

The fundamental question we address in this chapter is whether or not measurements of the input and output signals of our linear state equation over a finite time interval can be processed in order to uniquely determine the initial state. If so, knowledge of the initial state and the input signal allows the entire state trajectory to be reconstructed according to the state equation solution formula.

Definitions

Definition of controllability. A system is said to be controllable (or completely controllable) if and only if it is possible, by means of the input, to transfer the system from *any* initial state $\mathbf{x}(t_0)$ to *any* other state $\mathbf{x}(T)$ in a *finite* time interval $T - t_0 \geq 0$.

Note the use of words any and finite in the above definition. If it is only possible to transfer the system from some states to some other states, the system is not controllable. Likewise, if the transfer takes an infinite amount of time, the system is not controllable. Note also that the initial time t_0 is arbitrary and the terminal time T is not fixed.

Definition of observability. An unforced system is said to be observable (or completely observable) if and only if it is possible to determine *any* state $\mathbf{x}(t)$ by using only a *finite* record $\mathbf{y}(\tau)$ for $t \leq \tau \leq T$, of the output.

Definitions

[see Example 5A of Friedland's book]

It can be shown that every LTI system in state space form can be decomposed into four subsystems:

1. controllable and observable
2. uncontrollable but observable
3. controllable but unobservable
4. neither controllable nor observable

The transfer function of the system is determined only by the controllable and observable subsystem.
If the TF of a SISO system is of lower degree than the dimension of the state space, then the system must contain an uncontrollable subsystem or an unobservable subsystem, or possibly both.

By convention, if a system contains an uncontrollable subsystem is said to be uncontrollable.
Likewise, if it contains an unobservable subsystem it is said to be unobservable.

Definitions

A distinction is introduced between an uncontrollable system in which the uncontrollable part is stable and one in which the uncontrollable part is unstable.

Definition of stabilizability. A system is said to be stabilizable if the uncontrollable part is stable.

Similarly, there is a distinction between an unobservable system in which the unobservable part is stable and one in which the unobservable part is unstable.

Definition of detectability. A system is said to be detectable if the unobservable part is stable.

Controllability and observability matrices

Controllability and observability can be checked by evaluating the rank of special matrices called controllability and observability matrices. The following results are typically called algebraic test or criterion of controllability and observability. They are given without any proof.

Controllability matrix $$\mathcal{C} = \begin{bmatrix} \mathbf{B} & \mathbf{AB} & \mathbf{A}^2\mathbf{B} & \dots & \mathbf{A}^{n-1}\mathbf{B} \end{bmatrix}$$

If and only if $\mathrm{rank}\,(\mathcal{C}) = n$ (n: number of states),

we say that the pair $d(t)$ is controllable.

If $\mathcal{C}$ is not full rank, the subspace spanned by its columns defines the controllable subsystem.

Controllability and observability matrices

Controllability and observability can be checked by evaluating the rank of special matrices called controllability and observability matrices. The following results are typically called algebraic test or criterion of controllability and observability. They are given without any proof.

Observability matrix

$$\mathcal{O} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \mathbf{CA}^2 \\ \vdots \\ \mathbf{CA}^{n-1} \end{bmatrix}$$

we could transfor A into a diagonal matrix
$\hat{A} = \text{diag}(\lambda_i)$
↳ eigenvalues of A
makes the computation much easier.
because we need to calculate the power of a matrix.

If and only if $\text{rank}(\mathcal{O}) = n$ number of states),

we say that the pair $d(t)$ is observable.

If $\mathbf{A}$ is not full rank, the subspace spanned by its columns defines the observable subsystem.

Controllability and observability matrices

State transformation: $\mathbf{z}(t) = \mathbf{T}\mathbf{x}(t)$

$$\hat{\mathbf{A}} = \mathbf{T}\mathbf{A}\mathbf{T}^{-1} \qquad \hat{\mathbf{B}} = \mathbf{T}\mathbf{B} \qquad \hat{\mathbf{C}} = \mathbf{C}\mathbf{T}^{-1} \qquad \hat{\mathbf{D}} = \mathbf{D}$$

$$\begin{aligned}
\hat{\mathcal{C}} &= \begin{bmatrix} \hat{\mathbf{B}} & \hat{\mathbf{A}}\hat{\mathbf{B}} & \hat{\mathbf{A}}^2\hat{\mathbf{B}} & \dots & \hat{\mathbf{A}}^{n-1}\hat{\mathbf{B}} \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{TB} & \mathbf{TAT}^{-1}\mathbf{TB} & (\mathbf{TAT}^{-1})^2\mathbf{TB} & \dots & (\mathbf{TAT}^{-1})^{n-1}\mathbf{TB} \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{TB} & \mathbf{TAT}^{-1}\mathbf{TB} & (\mathbf{TAT}^{-1})(\mathbf{TAT}^{-1})\mathbf{TB} & \dots & (\mathbf{TAT}^{-1})(\mathbf{TAT}^{-1})\cdot\cdot(\mathbf{TAT}^{-1})\mathbf{TB} \end{bmatrix} \\
&= \mathbf{T}\begin{bmatrix} \mathbf{B} & \mathbf{AB} & \mathbf{A}^2\mathbf{B} & \dots & \mathbf{A}^{n-1}\mathbf{B} \end{bmatrix} \\
&= \mathbf{T}\mathcal{C}
\end{aligned}$$

$$\hat{\mathcal{O}} = \mathcal{O}\mathbf{T}^{-1}$$

It can be shown that the property of controllability/observability is preserved by any nonsingular transformation of the state variables.

Controllability and observability in MATLAB

ctrb

Controllability matrix

Syntax

```
Co = ctrb(A,B)
Co = ctrb(sys)
```

Description

`Co = ctrb(A,B)` returns the controllability matrix:

$$Co = \begin{bmatrix} B & AB & A^2B & \dots & A^{n-1}B \end{bmatrix}$$

where A is an n-by-n matrix, B is an n-by-m matrix, and Co has n rows and nm columns.

Compute controllability matrix.

```
Co = ctrb(A,B);
```

Determine the number of uncontrollable states.

```
unco = length(A) - rank(Co)
```

Controllability and observability in MATLAB

obsv

Observability matrix

Syntax

```
obsv(A,C)
Ob = obsv(sys)
```

Description

`obsv` computes the observability matrix for state-space systems. For an n-by-n matrix `A` and a p-by-n matrix `C`, `obsv(A,C)` returns the observability matrix

$$Ob = \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{bmatrix}$$

with n columns and np rows.

Controllability and observability matrices

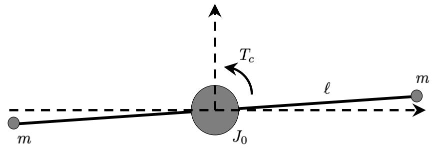
Limitations:

1. estimating the rank of the controllability/observability matrix is ill-conditioned, i.e., it is very sensitive to roundoff errors and errors in the data

2. for systems having a large number of states, computing the highest power of A easily results in numerical overflow

3. the algebraic test on the rank of the controllability/observability matrix is a yes/no answer. Sometimes it could be useful to have a measure of the degree of controllability/observability (i.e., if a state is weakly controllable with respect to the others)

Example
(rigid satellite)



State variables:

$$x_1(t) = \theta(t)$$
$$x_2(t) = \dot{\theta}(t)$$

State space matrices:

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 0 \\ 0.01 \end{bmatrix} \quad \mathbf{C} = \begin{bmatrix} 1 & 0 \end{bmatrix}$$

Controllability matrix:

$$\mathcal{C} = \begin{bmatrix} 0 & 0.01 \\ 0.01 & 0 \end{bmatrix}$$

Controllability test: $\mathrm{rank}(\mathcal{C}) = 2$

The system is controllable!

Observability matrix:

$$\mathcal{O} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Observability test: $\mathrm{rank}(\mathcal{O}) = 2$

The system is observable!