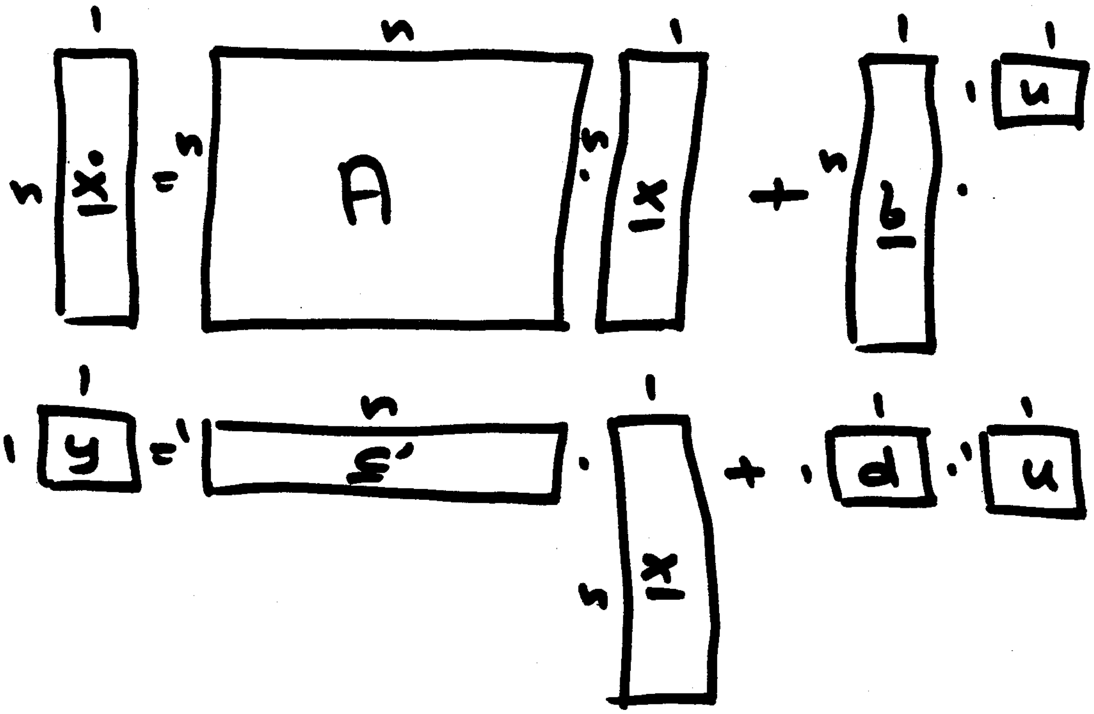
Thus, we can write <u>all</u> linear time-continuous single-input / single-output (SISO) systems in the form:

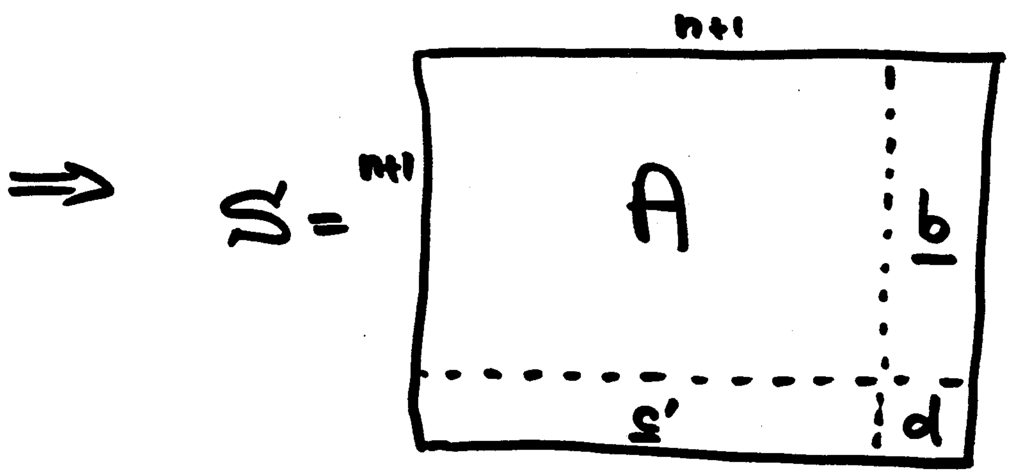
$$\left| \begin{array}{l} \underline{\dot{x}} = A \cdot \underline{x} + \underline{b} \cdot u \\ y = \underline{c}' \cdot \underline{x} + d \cdot u \end{array} \right|$$

$\underline{x} \in \mathbb{R}^n$ is called the state vector of the system. Its size (n) is determined by the number of elements in the system that can store energy independently.

For dimensional reasons:

$$\left| \begin{array}{l} A \in \mathbb{R}^{n \times n} \\ \underline{b} \in \mathbb{R}^{n \times 1} \\ \underline{c}' \in \mathbb{R}^{1 \times n} \\ d \in \mathbb{R}^{1 \times 1} \end{array} \right|$$





Notation:

horizontal concatenation

$$S = [\, A \,,\, \underline{b} \,;\, \underline{c}' \,,\, d \,]$$

vertical concatenation

Example:

$$G(s) = \frac{5s^2 + 7s + 9}{s^3 + 8s^2 + 6s + 2} = \frac{P(s)}{Q(s)}$$

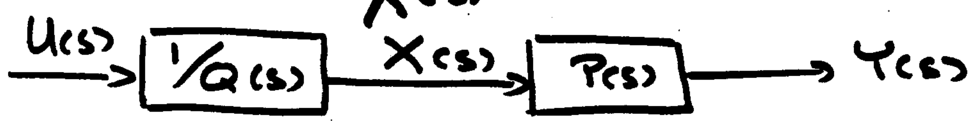
Question: Can we find a state-space representation for this system?

Recipe:

$$\frac{Y(s)}{U(s)} = G(s) = \frac{Y(s)}{X(s)} \cdot \frac{X(s)}{U(s)}$$

$$= P(s) \cdot \frac{1}{Q(s)}$$

We set: $\dfrac{Y(s)}{X(s)} \equiv P(s)$; $\dfrac{X(s)}{U(s)} = \dfrac{1}{Q(s)}$



Example continued:

$$\frac{X(s)}{U(s)} = \frac{1}{Q(s)} = \frac{1}{s^3 + 8s^2 + 6s + 2}$$

$$\Rightarrow \left[s^3 + 8s^2 + 6s + 2 \right] X(s) = U(s)$$

$$\dddot{x}(t) + 8\ddot{x}(t) + 6\dot{x}(t) + 2x(t) = u(t)$$

$$\Rightarrow \dddot{x}(t) = -2x(t) - 6\dot{x}(t) - 8\ddot{x}(t) + u(t$$

We let:

$$x_1 = x \; ; \quad x_2 = \dot{x} \quad ; \quad x_3 = \ddot{x}$$

$$\Rightarrow \left| \begin{array}{l} \dot{x}_1 = x_2 \\ \dot{x}_2 = x_3 \\ \dot{x}_3 = -2x_1 - 6x_2 - 8x_3 + u \end{array} \right|$$

or:

$$\underline{\dot{x}} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -2 & -6 & -8 \end{bmatrix} \underline{x} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u$$

$$\frac{Y(s)}{X(s)} = P(s) = 5s^2 + 7s + 9$$

$$\Rightarrow Y(s) = \left[5s^2 + 7s + 9\right] X(s)$$

$$\Rightarrow y(t) = 5\ddot{x}(t) + 7\dot{x}(t) + 9x(t)$$

$$\Rightarrow y = 9x_1 + 7x_2 + 5x_3$$

or: $$y = \begin{bmatrix} 9 & 7 & 5 \end{bmatrix} \underline{x} + [\emptyset]\, u$$

$$\Rightarrow \left| \begin{array}{l} \dot{\underline{x}} = \begin{bmatrix} \emptyset & 1 & \emptyset \\ \emptyset & \emptyset & 1 \\ -2 & -6 & -8 \end{bmatrix} \underline{x} + \begin{bmatrix} \emptyset \\ \emptyset \\ 1 \end{bmatrix} u \\ y = \begin{bmatrix} 9 & 7 & 5 \end{bmatrix} \underline{x} + [\emptyset]\, u \end{array} \right|$$

is our desired state-space representation.

In general:

$$G(s) = \frac{P(s)}{Q(s)} = \frac{b_m s^m + b_{m-1} s^{m-1} + \cdots + b_1 s +}{s^n + a_{n-1} s^{n-1} + \cdots + a_1 s + a}$$

where: $m < n$

(that is: $\mathrm{ord}(P) < \mathrm{ord}(Q)$)

$$\Downarrow$$

$$\dot{\underline{x}} = \begin{bmatrix} 0 & 1 & 0 & \cdots & \cdots & 0 \\ 0 & 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 1 & \cdots & 0 \\ \vdots & & & & \ddots & \vdots \\ & & & & & 1 \\ -a_0 & -a_1 & -a_2 & \cdots & \cdots & -a_{n-1} \end{bmatrix} \underline{x} + \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} u$$

$$y = \left[b_0 \; b_1 \; b_2 \; \cdots \; b_m \; \vdots \; 0 \cdots \right] \underline{x} + [0]\, u$$

There are som
zeroes iff:
$m \le n-2$

Terminology: If $m < n$:

$G(s)$ is "strictly proper"

<u>We notice</u>: If $G(s)$ is strictly proper, we can get a state-space representation at once by filling the negative denominator coefficients into the lowermost row of A, the numerator coefficients into $\underline{c}'$, and putting a superdiag of 1-elements into A.

We also realise that:

$$\operatorname{ord}(\underline{x}) = \operatorname{ord}(Q)$$

the <u>order</u> of the system.

Example:

$$G(s) = \frac{5s^2 + 7s + 9}{s^2 + 2s + 15} = \frac{P(s)}{Q(s)}$$

$\Rightarrow$ $m = n$: $G(s)$ is proper, but not strictly proper.

Recipe: We divide $P(s) : Q(s)$

$$(5s^2+7s+9) : (s^2+2s+15) = 5$$

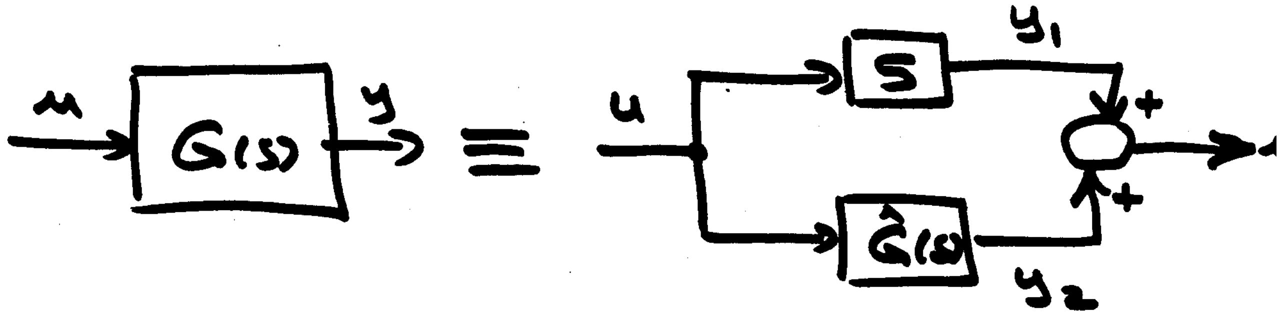
$$\underline{-5s^2+10s+75}$$

$$-3s-66$$

$$\Rightarrow G(s) = 5 + \hat{G}(s)$$

$$\hat{G}(s) = \frac{-3s-66}{s^2+2s+15}$$

$\hat{G}(s)$ is strictly proper



$$y_1 = 5u$$

$$\dot{\underline{x}} = \begin{bmatrix} 0 & 1 \\ -15 & -2 \end{bmatrix} \underline{x} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$$

$$y_2 = \begin{bmatrix} -66 & -3 \end{bmatrix} \underline{x} + \begin{bmatrix} 0 \end{bmatrix} u$$

$$y = y_1 + y_2$$

$$\Rightarrow \quad \dot{\underline{x}} = \begin{bmatrix} 0 & 1 \\ -15 & -2 \end{bmatrix} \underline{x} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$$

$$y = \begin{bmatrix} -66 & -3 \end{bmatrix} \underline{x} + \begin{bmatrix} 5 \end{bmatrix} u$$

is the desired state-space representation.

Example:

$$G(s) = \frac{5s^2 + 7s + 9}{3s + 4} = \frac{P(s)}{Q(s)}$$

$G(s)$ is <u>not proper</u>:

$$P(s) : Q(s) = (5s^2 + 7s + 9) : (3s + 4) = \frac{5}{3}s +$$

$$-\quad 5s^2 + \frac{20}{3}s$$

$$\frac{1}{3}s + 9$$

$$-\quad \frac{1}{3}s + \frac{4}{9}$$

$$\frac{77}{9}$$

$$\Rightarrow G(s) = \underbrace{\left(\frac{5}{3}s + \frac{1}{9}\right)}_{d \,\equiv\, d(s)} + \underbrace{\frac{77/9}{3s + 4}}_{\text{strictly proper}}$$

$\Rightarrow$ There does not exist a state-space representation for non-proper transfer functions.

We cannot avoid true differentiations $\Rightarrow$ <u>bad system.</u>

<u>Warning</u>: The state-space representation is not unique

<u>Proof</u>: Given: $\left| \begin{array}{l} \underline{\dot{x}} = A\underline{x} + \underline{b}u \\ y = \underline{c}'\underline{x} + du \end{array} \right|$

Choose any non-singular matri[x] T such that:

$$\underline{\xi} = T \cdot \underline{x}$$

$$\Rightarrow \underline{x} = T^{-1} \cdot \underline{\xi} \quad ; \quad \underline{\dot{x}} = T^{-1} \cdot \underline{\dot{\xi}}$$

$$\Rightarrow \left| \begin{array}{l} T^{-1} \underline{\dot{\xi}} = A \cdot T^{-1} \underline{\xi} + \underline{b}u \\ y = \underline{c}' T^{-1} \underline{\xi} + du \end{array} \right|$$

$$\Rightarrow \left| \begin{aligned} \dot{\underline{\xi}} &= TAT^{-1}\underline{\xi} + T\underline{b}\,u \\ y &= \underline{c}'T^{-1}\underline{\xi} + d\,u \end{aligned} \right|$$

Let:
$$\left| \begin{aligned} \hat{A} &= TAT^{-1} \\ \hat{\underline{b}} &= T\underline{b} \\ \hat{\underline{c}}' &= \underline{c}'T^{-1} \\ \hat{d} &= d \end{aligned} \right|$$

$$\Rightarrow \left| \begin{aligned} \dot{\underline{\xi}} &= \hat{A}\underline{\xi} + \hat{\underline{b}}\,u \\ y &= \hat{\underline{c}}'\underline{\xi} + \hat{d}\,u \end{aligned} \right|$$

has the same form, and i therefore another state-space representation is the new state vector $\underline{\xi}$.

The two representations have state-vectors that are linear combinations of each other. The two representations are called <u>similar</u>, and the transformation

$$\underline{x} \xrightarrow{T} \underline{s}$$

is called a <u>similarity transformation</u> with T.

- Any similarity transformation with a non-singular matrix T leads to a similar state-space representation.

__Question__: Given a state-spa[ce]
representation:

$$\left| \begin{array}{l} \dot{\underline{x}} = A\underline{x} + \underline{b}\,u \\ y = \underline{c}'\underline{x} + d\,u \end{array} \right|$$

Can we find the equivalent transfer function?

__Recipe__:

$$\underline{x}(t) \circ\!\!-\!\!\!-\!\!\bullet\ \underline{X}(s)$$

$$\dot{\underline{x}}(t) \circ\!\!-\!\!\!-\!\!\bullet\ s\underline{X}(s)$$

$$\Rightarrow \left| \begin{array}{l} s\underline{X}(s) = A\underline{X}(s) + \underline{b}\,U(s) \\ Y(s) = \underline{c}'\underline{X}(s) + d\,U(s) \end{array} \right|$$

$$\Rightarrow \left| \begin{array}{l} (sI - A)\underline{X}(s) = \underline{b}\,U(s) \\ Y(s) = \underline{c}'\underline{X}(s) + d\,U(s) \end{array} \right|$$

$$\Rightarrow \left| \begin{array}{l} \underline{X}(s) = (sI-A)^{-1}\underline{b}\,U(s) \\ Y(s) = \underline{c}'\underline{X}(s) + d\,U(s) \end{array} \right|$$

$$\Rightarrow Y(s) = \left[\underline{c}'(sI-A)^{-1}\underline{b} + d \right] U(s)$$

$$= G(s) \cdot U(s)$$

$$\Rightarrow \underline{\underline{G(s) = \underline{c}'(sI-A)^{-1}\underline{b} + d}}$$

Example:

$$\left| \begin{array}{l} \dot{\underline{x}} = \begin{bmatrix} 73 & -31 \\ 184 & -78 \end{bmatrix} \underline{x} + \begin{bmatrix} 2 \\ 5 \end{bmatrix} u \\ y = \begin{bmatrix} -9 & 4 \end{bmatrix} \underline{x} + [\emptyset] u \end{array} \right|$$

$$\Rightarrow (sI-A) = \begin{bmatrix} (s-73) & 31 \\ -184 & (s+78) \end{bmatrix}$$

$$\Rightarrow \det(sI-A) = (s-73)(s+78) + 31 \cdot 184$$

$$= s^2 + 5s - 5694 + 57\emptyset4$$

$$= s^2 + 5s + 1\emptyset$$

$$\Rightarrow \quad (sI-A)^{-1} = \frac{(sI-A)^{+}}{|sI-A|} = \frac{\operatorname{adj}(sI}{\det(sI}$$

$$= \frac{\begin{bmatrix} (s+78) & -31 \\ 184 & (s-73) \end{bmatrix}}{s^2+5s+10}$$

$$\Rightarrow \quad (sI-A)^{-1}\underline{b} = \frac{\begin{bmatrix} (2s+1) \\ (5s+3) \end{bmatrix}}{s^2+5s+10}$$

$$\Rightarrow \quad \underline{\underline{G(s)}} = \underline{c}'(sI-A)^{-1}\underline{b} = \underline{\underline{\frac{2s+3}{s^2+5s+10}}}$$

For larger systems, this get soon very cumbersome though. We shall need something better!

If the system were in the special form:

$$\left|\begin{array}{l} \dot{\underline{x}} = \begin{bmatrix} 0 & 1 & & \emptyset & \\ & & 1 & & \\ & \emptyset & & \ddots & \\ & & & & 1 \\ * & * & * & \cdots & * \end{bmatrix} \underline{x} + \begin{bmatrix} \emptyset \\ \vdots \\ \emptyset \\ 1 \end{bmatrix} u \\[2ex] y = \begin{bmatrix} * & * & * & * & \cdots & * \end{bmatrix} \underline{x} + [*] u \end{array}\right.$$

we could read out the transfer function directly. This special representation has therefore a name. It is called: <u>controller - canonical form</u>, and it is one of several such special (that is: "canonical") representations.

Thus, we can reformulate our previous question. Given an state-space representation,

can we find the transform-
ation matrix T that can
be used in a similarity
transformation to transform
the system into controller-
canonical form ?

<u>Recipe</u>:

(1) We compute the "controllability
matrix"

$$Q_c = \left[\underline{b}, A \cdot \underline{b}, \ldots, A^{n-1} \underline{b} \right]$$

$$Q_c \in \mathbb{R}^{n \times n}$$

(2) We compute its inverse:

$$Q_c^{-1} = \text{inv}(Q_c)$$

and extract its last row $\underline{q}$

(3) We build the "observability matrix" assuming $\underline{q}'$ to be the output vector:

$$T = [\,\underline{q}' \,;\, \underline{q}'A \,;\, \dots \,;\, \underline{q}'A^{n-1}\,]$$

(4) We perform a similarity transformation with T.

Example continued:

$$\dot{\underline{x}} = \begin{bmatrix} 73 & -31 \\ 184 & -78 \end{bmatrix} \underline{x} + \begin{bmatrix} 2 \\ 5 \end{bmatrix} u$$

$$y = [\,-9 \quad 4\,]\,\underline{x}$$

$$A \cdot \underline{b} = \begin{bmatrix} -9 \\ -22 \end{bmatrix}$$

$$\Rightarrow Q_c = \left[\begin{array}{c:c} 2 & -9 \\ 5 & -22 \end{array}\right] \Rightarrow \det(Q_c) = 1$$

$$\Rightarrow Q_c^{-1} \equiv \mathrm{adj}(Q_c) = \begin{bmatrix} -22 & 9 \\ -5 & 2 \end{bmatrix}$$

$$\Rightarrow \underline{q}' = [-5 \quad 2] \; ; \quad \underline{q}'A = [3 \quad -1]$$

$$\Rightarrow T = \begin{bmatrix} -5 & 2 \\ 3 & -1 \end{bmatrix} \Rightarrow T^{-1} = \begin{bmatrix} 1 & 2 \\ 3 & 5 \end{bmatrix}$$

$$\Rightarrow \hat{A} = T \cdot A \cdot T^{-1} = \begin{bmatrix} 0 & 1 \\ -10 & -5 \end{bmatrix}$$

$$\hat{\underline{b}} = T \cdot \underline{b} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$\hat{\underline{c}}' = \underline{c}' \cdot T^{-1} = [3 \quad 2]$$

$$\Rightarrow \left| \begin{aligned} \dot{\underline{\xi}} &= \begin{bmatrix} 0 & 1 \\ -10 & -5 \end{bmatrix} \underline{\xi} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \\ y &= [3 \quad 2] \underline{\xi} \end{aligned} \right|$$

is in controller-canonical form

$$\Rightarrow G(s) = \frac{2s + 3}{s^2 + 5s + 10}$$

Also this can get pretty boring when the system is sufficiently large. Now, we need to invert twice, but at least, these are normal matrix inversions, and not polynomial matrix inversions.

- To solve such problems conveniently, we need a computer and some userfriendly software. One such system is <u>MATLAB</u>, which is available on all our Unix systems as well as on the PCs.

- The previous problem could e.g. have been solved in Matlab using the code:

```
>> A = [73, -31; 184, -78];
>> b = [2; 5];
>> c = [-9, 4];
>> Qc = [b, A*b];
>> Qcin = inv(Qc);
>> q = Qcin(2, :);
>> T = [q; q*A];
>> An = T*A/T
>> bn = T*b
>> cn = c/T
```

Of course, Matlab offers built-in functions to compute transfer functions from state-space descriptions as part of its <u>control systems toolbox</u>.