State variables and SISO/MIMO control systems

1. State variable model for a dynamic system
This type of model consists of a set of simultaneous first-order differential equations, called the State equation:

\[
\frac{d}{dt} x = [A]x + [B]i(t)
\]

and the output equation (relating \( o(t) \) to the state vector \( x \)):

\[
o(t) = [C]x + Di(t)\]

Here we define:
- \( x \) as the state vector,
- \( A \) as the system matrix (square, \( N \times N \) for \( N \) states),
- \( B \) as the input matrix (\( N \) rows x 1 column for a single-input, single output (SISO) system),
- \( C \) as the output matrix (one row x \( N \) columns for a SISO system),
- \( D \) as the feedforward matrix (1 x 1 for a SISO system).

As shown in the Topic Overview for the course, you can obtain transfer functions in either the \( s \)- or the \( z \)-domain for a system from the state-variable model. For a SISO system, these transfer functions take the forms:

\[
H(s) = [C(sI - A)^{-1}B + D], \quad \text{and} \quad H(z) = [C(zI - A)^{-1}B + D],
\]

where \( I \) is the identity matrix (remember that the elements of \( A, B, C, \) and \( D \) will not be the same for the continuous and the sampled models, and that in the sampled model they will depend on sampling frequency)

2. Obtaining a state variable model
You can get a state variable model for a system in two ways: from detailed knowledge of the internal dynamics of the system, and from a transfer function (which might have been obtained by system identification or by other means not requiring knowledge of the internal dynamics).

2a. from the internal dynamics of the system: an example
Suppose we use an electromagnet coil to position permanent bar magnet of mass \( M \) attached to a wall through a spring of spring constant \( K \). The coil has negligible inductance and a wire resistance of \( R \). We drive this coil with an input voltage, \( v(t) \), as shown here:
Frictional forces in this system are represented by an effective coefficient of friction, \( W \). The force on the bar magnet is proportional to the current in the coil with a constant of proportionality, \( \alpha \). Equating this magnetic force to the sum of inertial, frictional, and spring forces gives an equation of motion (in the \( s \)-domain),

\[
(Ms^2 + Ws + K)X(s) = \frac{\alpha V(s)}{R}
\]

and a transfer function from applied voltage to position,

\[
G(s) = \frac{X(s)}{V(s)} = \frac{\alpha}{RM^2 + RWS + RK},
\]

and a second-order differential equation,

\[
RM \frac{d^2x}{dt^2} + RW \frac{dx}{dt} + RKx = \alpha v(t)
\]

For a second-order system, two states are required to model it. Choosing a set of states:

\[
\begin{aligned}
x &= \begin{bmatrix} x_1 = \text{position} \\ x_2 = \text{velocity} \end{bmatrix}
\end{aligned}
\]

the system differential equation becomes

\[
RM \frac{dx_2}{dt} + RWx_2 + RKx_1 = \alpha v(t)
\]

This equation, and another obvious one,

\[
\frac{dx_1}{dt} = x_2,
\]

provide enough information to write the state equation,

\[
\frac{d}{dt} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -K/M & -W/M \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \alpha/\text{RM} \end{bmatrix} v(t)
\]

and an output equation,
Important general property: the poles of the transfer function are the eigenvalues of the system matrix, $A$. Setting the denominator of the transfer function to zero gives

$$Ms^2 + Ws + K = 0 \Rightarrow s = \frac{-W \pm \sqrt{W^2 - 4MK}}{2M}.$$ 

Finding the eigenvalues of $A$ using the standard matrix method of setting $\det[sI - A] = 0$ yields

$$s^2 + \frac{W}{M}s + \frac{K}{M} = 0,$$

the same result. This property can be shown to be true for all linear systems with state-variable models.

2b. From the transfer function of the system: another example

Signal-flow graphs

In order to introduce some key ideas in state-variable system modeling, we need to use signal-flow graphs. These graphs allow for only three types of operations:

- Addition of incoming signals at a node: Here the node is a small circle. Any signal flowing out of a node is the sum of all the signals flowing in.
- Amplification by a fixed factor, positive or negative: the gain is just written above the signal path.
- Integration: This is denoted by a box containing the integral sign, or $1/s$.

The state variable model for any linear system is a set of first-order differential equations. Therefore, the outputs of each integrator in a signal-flow graph of a system are the states of that system. For any system, an infinite number of signal graphs are possible, but only a few are of interest. Let’s look at some processes for obtaining a signal-flow graph for a given system. This is best done by means of a specific example. Consider the transfer function, and its equivalent differential equation:
since this is a second-order system, its state model will have two states, which will appear at the outputs of the two integrators in any signal flow graph. Next, we will consider three forms of the state model for this system, each of which results from a slightly different approach:

**Control canonical form:** This form gets its name from the fact that all of the states are fed back to the input in the signal flow graph. For this state-variable model, solve the differential equation for the highest-order derivative of the output as

$$\frac{d^2 o}{dt^2} = -\frac{a_1}{a_2} \frac{do}{dt} - \frac{a_0}{a_2} o + \frac{b_0}{a_2} i + \frac{b_1}{a_2} \frac{di}{dt}$$

(This solution is for a particular second order system, but you can see how to extend this idea to a higher-order system) To begin to draw the signal graph, we connect two integrators in cascade, and identify the output and its two derivatives (using primes to denote differentiation), which gives

Note that the signals are drawn immediately to the right of (on the output side of) the nodes. The differential equation for the highest derivative of $o(t)$ identifies this derivative as the sum of several terms. Two of these terms depend on lower-order derivatives of the output, and one depends on the input. You draw the signal paths corresponding to the lower output derivatives as feedback loops as shown here

This diagram obviously represents, in a graphic way, the first two terms on the right side of the equation for $o''(t)$. Just like any diagram of a “signal-processing” system, the input should be on the left and the output should be on the right. Putting the input into the diagram gives
All terms of the equation for $o''(t)$ but the last one are now represented in the diagram. In order to use only integration, addition and multiplication in our signal graph, we have to represent terms which are proportional to first (and higher) derivatives in the following way: suppose we rewrite the transfer function as

$$\frac{-B_0 - B_1 s}{a_0 + a_1 s + a_2 s^2} = \frac{B_0}{a_0 + a_1 s + a_2 s^2} + \frac{B_1}{a_0 + a_1 s + a_2 s^2} s$$

This clearly shows that the output arises from two terms, and the first of these terms could be obtained from the signal graph we have so far. The second term is proportional to the derivative of the first one. The signal graph has a node from which we can get the derivative of the output, namely $o'(t)$. To finish our signal graph, we just move the input gain to the output side, and take an additional signal proportional to $o'(t)$ to the new output via a feed-forward loop with the required proportionality constant. The result is

You can now identify each state with an integrator output, to yield the states $x_1$ and $x_2$, as shown next:
The first derivative of each state is the signal just back on the upstream side of each integrator. Thus, we can write two differential state equations and an additional equation called the “output equation”, which relates the states to the system output, as

$$\frac{dx_i}{dt} = x_2$$

$$\frac{dx_2}{dt} = -\frac{a_1}{\alpha_2} x_2 - \frac{a_0}{\alpha_2} x_1 + u$$

$$o(t) = \frac{b_0}{\alpha_2} x_1 + \frac{b_1}{\alpha_2} x_2$$

These equations can be organized into a compact set of matrix equations which look like this:

$$\begin{bmatrix} \frac{dx_1}{dt} \\ \frac{dx_2}{dt} \end{bmatrix} = \begin{bmatrix} A \\ B \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$$

$$o = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ D \end{bmatrix} u$$

And have the general form,

$$\frac{dx}{dt} = Ax + Bu$$

$$o = Cx + Du$$

In this general form for the state equation model, if there are \( n \) states, \( r \) inputs, and \( p \) outputs, then the matrices will have the following names and forms (rows x columns):

- System matrix, \( A \): \( n \times n \),
- Input matrix, \( B \): \( n \times r \),
- Output matrix, \( C \): \( p \times n \),
- Feed-forward matrix, \( D \): \( p \times r \).

**Note on transfer function normalization:** Notice how the highest-order \( \alpha \) for this transfer function keeps appearing in denominators everywhere. Transfer function coefficients are not unique, and you can always divide numerator and denominator of any
transfer function by the highest-order $\alpha$ to obtain a normalized transfer function of the form
\[
G(s) = \frac{\beta_0 + \ldots + \beta_m s^m}{\alpha_0 + \alpha_1 s + \ldots + s^n},
\]
where the highest-order $\alpha$ is unity. This obviously makes for cleaner matrices. If you normalize the transfer function first, the control canonical form state equations look like this (for a normalized 4th-order system. Extension to higher order is straightforward):

\[
A = \begin{bmatrix}
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
-\alpha_0 & -\alpha_1 & -\alpha_2 & -\alpha_3 \\
0 & 0 & 0 & 1
\end{bmatrix}, \\
B = [\beta_0 \ \beta_1 \ \beta_2 \ \beta_3], \\
C = \begin{bmatrix}
0 \\
0 \\
1
\end{bmatrix}, \\
\]
and $D$ contains only zeros.

**Modal (or modal canonical) form**

Suppose you converted the second order transfer function we are using as an example here into pole residue form,

\[
G(s) = \frac{A_1}{s - p_1} + \frac{A_2}{s - p_2}
\]

You could convert any one of the terms into the following simple signal flow graph and state- and output equation:

For a transfer function with $n$ distinct poles (two poles with the same values can be artificially separated by some very small difference), you would get the following graph and state- and output equation:

\[
\frac{d\vec{x}}{dt} = \rho \vec{x} + \vec{I} \\
\vec{o} = \mathbf{A} \vec{x}
\]

The interesting thing about this form is the appearance of the system poles as the elements in a diagonalized system matrix. This says that the eigenvalues of the system
matrix (regardless of what form it is in) are the poles of the transfer function. Also, note that in this form, the coefficients in the equations will generally be complex. This was not the case for the control canonical form earlier, since the coefficients in the equations there were ratios of (real) transfer function coefficients.

Here is the general matrix modal form for a fourth-order system:

\[
\begin{bmatrix}
    p_1 & 0 & 0 & 0 \\
    0 & p_2 & 0 & 0 \\
    0 & 0 & p_3 & 0 \\
    0 & 0 & 0 & p_4
\end{bmatrix}, \quad
\begin{bmatrix}
    1 \\
    1 \\
    1 \\
    1
\end{bmatrix},
\begin{bmatrix}
    A_1 & A_2 & A_3 & A_4
\end{bmatrix}
\]

with \( D \) containing all zeros.

**Observer canonical form**

There is one more special form of the state equations that is of interest. In this case the feedback is from the output to the state variables. For this form, we start with a normalized, 3rd-order transfer function,

\[
G(s) = \frac{\beta_0 + \beta_1 s + \beta_2 s^2}{\alpha_0 + \alpha_1 s + \alpha_2 s^2 + s^3},
\]

and draw the following signal-flow graph:

![Signal-flow graph](image)

which leads to the matrix forms,

\[
\begin{bmatrix}
    -\alpha_2 & 1 & 0 \\
    -\alpha_1 & 0 & 1 \\
    -\alpha_0 & 0 & 0
\end{bmatrix}, \quad
\begin{bmatrix}
    \beta_1 \\
    \beta_0
\end{bmatrix},
\begin{bmatrix}
    1 & 0 & 0
\end{bmatrix},
\]

and \( D \) contains all zeros.

For the control canonical form, we justified the form of the signal-flow graph by solving the differential equation for the highest-order derivative of the output. For the modal form, we did this by first looking at a single term of the residue-pole form of the transfer function, then adding similar terms. For the observer canonical form, suppose we multiply the normalized transfer function by \( s \) raised to that power, thereby creating a rational polynomial in \((1/s)\) as follows:
\[
G(s) = \frac{O(s)}{I(s)} = \frac{\beta_0 s^{-3} + \beta_1 s^{-2} + \beta_2 s^{-1}}{\alpha_0 s^{-3} + \alpha_1 s^{-2} + \alpha_2 s^{-1} + 1}.
\]

This leads to the following Laplace transform equation relating the input, \(I(s)\) to the output \(O(s)\):

\[
O(s) = (\beta_0 I(s) - \alpha_0 O(s))s^{-3} + (\beta_1 I(s) - \alpha_1 O(s))s^{-2} + (\beta_2 I(s) - \alpha_2 O(s))s^{-1}.
\]

This equation corresponds exactly to the signal-flow graph: The first term on the right side gets integrated three times, the second twice, and the third once.

**Transformation to other state-space representations**

How are the different state-space representations related, other than in representing the same physical system? If a linear system can be represented by two state vectors, \(u\) and \(v\), the two vectors must be related through a transformation \(T\) by

\[
u = Tv, \quad \text{and} \quad v = T^{-1}u.
\]

The inverse of \(T\) must exist, that is \(T\) must be non-singular.

We can use this relation to transform the state and output equations as well, for example, if with one state vector,

\[
\dot{u} = Gu + Hi, \quad o = Pu + Qi,
\]

then using the transformation, \(T\),

\[
\dot{u} = GTv + Hi \quad \text{and} \quad o = PTv + Qi.
\]

Pre-multiplying by the inverse of \(T\) gives a new set of state equations,

\[
\dot{v} = T^{-1}\dot{u} = T^{-1}GTv + T^{-1}Hi \quad \text{and} \quad o = PTv + Qi.
\]

where \(A, B,\) and \(C\) are respectively the new system, input, and output matrices for the system using the state vector \(v\).

The availability of the transformation, \(T\), means that an infinite number of state representations for a system are possible. Only a few of these are interesting.

**It is left as a problem** to show that the transfer function, calculated using the matrices from the state variable model using

\[
H(s) = [C(sI - A)^{-1}B + D]
\]

is the same as the one at the beginning of this example. The extension to plants of order higher than 2 is straightforward.

**3. Pole placement using state feedback**
Suppose you can measure all the states of a plant with sensors that provide signals corresponding to each. Such a sensor is called an observer. In the automatic control scheme called full state feedback, a weighted linear combination of all the states is fed back and subtracted from the command signal, \( r(t) \), as shown here:

To see how this controller works, just substitute the new input,

\[
\begin{bmatrix}
3 \\
2 \\
1 \\
\end{bmatrix} x_1 - \begin{bmatrix}
K_1 \\
K_2 \\
K_3 \\
\end{bmatrix} x_2 = r(t)
\]

into the state equation. The result is a new state equation for the closed-loop system with the state feedback controller:

\[
\frac{dx}{dt} = [A - BK]x - Br(t).
\]

Clearly, the new state matrix for the closed-loop system is \([A-BK]\), and its poles (eigenvalues) can be adjusted at will by the designer by adjusting the components of \(K\). The analytic method for this pole placement is called Ackermann’s Formula, and a computer implementation is available. Unfortunately, just placing poles where you want them is usually insufficient to insure that the closed-loop system’s time- and frequency-domain responses are as you wish. Nevertheless, a state-variable analysis of a proposed controller is still a powerful alternative design tool.

**Multiple input multiple output (MIMO) systems**

All through this discussion, we have been explicitly carrying along the capability to analyze systems with multiple inputs and outputs (MIMO systems). The link between the state variable model and the transfer function is the transfer function matrix (see below) which is 1 x 1 for a single-input single output (SISO) system and of higher dimensionality for a MIMO system.

**Can we “guarantee” observable states?**

When using state variables to design a control system, you have to be sure you can sense or measure the states you are using for your model, or you won’t be able to carry out the multiplication,
in the feedback loop. Regardless of whether or not any particular set of states are observable, you can always count on being able to measure the following things about your plant:

- all of its outputs, comprising the components of the vector, \( \mathbf{o}(t) \),
- all of its inputs, comprising the components of the vector, \( \mathbf{i}(t) \).

Now, suppose that in addition to these measurements, you have used either a theoretical model or system identification techniques to obtain a transfer function matrix for your plant that relates each output to each input (this example has two inputs and two outputs),

\[
\begin{bmatrix}
G_{11}(s) & G_{12}(s) \\
G_{21}(s) & G_{22}(s)
\end{bmatrix} = \mathbf{G}(s)
\]

and that each element of this matrix is in rational polynomial form. Finally, suppose you have found all of the poles of each element in \( \mathbf{G} \), and have rewritten each element so that all elements share a common denominator. Further, assume you have normalized each rational polynomial. If the denominators in \( \mathbf{G} \) were 3\text{rd} order, each element in the transfer function matrix would have the following form (generalization to higher order rational polynomials is straightforward):

\[
G_{ij}(s) = \frac{\beta_{0ij} + \beta_{1ij}s + \beta_{2ij}s^2}{\alpha_0 + \alpha_1s + \alpha_2s^2 + s^3}.
\]

A useful way to interpret \( G_{ij} \) is as the transfer function to output \( i \) from input \( j \). If you have either a theoretical model or the results of system identification for \( \mathbf{G} \), then in addition to the inputs and outputs, you know:

- all of the \( \alpha \) and \( \beta \) coefficients that make up the plant transfer function, \( \mathbf{G}(s) \)

You can use the Observer Canonical form of each element in \( \mathbf{G} \) to construct a set of states that are, in principle, guaranteed to be observable. Consider first the transfer function from input 1 to output 1, \( G_{11} \). For a 3\text{rd} order system such as our example, the signal-flow graph would look like this:

![Signal-flow graph](image)

The first subscript of the \( \beta \) coefficients refers to the order in the numerator of the rational polynomial. The later pair of subscripts refers to the element in the \( \mathbf{G} \)-matrix for which
we are making the signal-flow graph. To take account of the contribution of both inputs
to output 1 (the top row of the transfer function matrix), you would need to bring in input
2 as shown below:

If there were more than two elements in the first row of $G$, you would need more sets of
inputs, and more $\beta$ coefficients than the ones shown here in order to show the
contribution of all the inputs to output 1.

A unique feature of the Observer Canonical form of the signal flow graph is that you can
calculate all of the states in terms of the known $\alpha$ and $\beta$ coefficients and the measurable
inputs and outputs. Note that, no matter how many inputs you might have, the states are
given as:

$$
x_1 = o_1;
$$

$$
x_2 = \dot{x}_1 + \alpha_2 o_1 - (\beta_{211}i_1 + \beta_{212}i_2);
$$

$$
x_3 = \dot{x}_2 + \alpha_1 o_1 - (\beta_{111}i_1 + \beta_{112}i_2)
$$

Thus, if you can measure and differentiate the outputs and inputs to this system, you can
always obtain the states. To look at the second output, repeat the above process,
recognizing that since the output is different (it is output 2, not output 1) the states will be
different also. The resulting signal flow graph looks like this:
From the signal-flow graph for output 1, you can write the state equations for the first triplet of states,
\[
\begin{align*}
\frac{d}{dt} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} &= \begin{bmatrix} -\alpha_2 & 1 & 0 \\ -\alpha_1 & 0 & 1 \\ -\alpha_0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} \beta_{211} & \beta_{212} \\ \beta_{111} & \beta_{112} \\ \beta_{011} & \beta_{012} \end{bmatrix} \begin{bmatrix} i_1 \\ i_2 \end{bmatrix} \quad \text{and} \quad o_1 = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}.
\end{align*}
\]

From the signal flow graph for output 2, you can write state equations for the second triplet of states,
\[
\begin{align*}
\frac{d}{dt} \begin{bmatrix} x_4 \\ x_5 \\ x_6 \end{bmatrix} &= \begin{bmatrix} -\alpha_2 & 1 & 0 \\ -\alpha_1 & 0 & 1 \\ -\alpha_0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} \beta_{221} & \beta_{222} \\ \beta_{121} & \beta_{122} \\ \beta_{021} & \beta_{022} \end{bmatrix} \begin{bmatrix} i_1 \\ i_2 \end{bmatrix} \quad \text{and} \quad o_2 = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_4 \\ x_5 \\ x_6 \end{bmatrix}.
\end{align*}
\]

For the combined MIMO system, this state-space representation requires six states, and looks like this:

- the state equation:
\[
\begin{align*}
\frac{d}{dt} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix} &= \begin{bmatrix} -\alpha_2 & 1 & 0 & 0 & 0 & 0 \\ -\alpha_1 & 0 & 1 & 0 & 0 & 0 \\ -\alpha_0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\alpha_2 & 1 & 0 \\ 0 & 0 & 0 & -\alpha_1 & 0 & 1 \\ 0 & 0 & 0 & -\alpha_0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix} + \begin{bmatrix} \beta_{211} & \beta_{212} \\ \beta_{111} & \beta_{112} \\ \beta_{011} & \beta_{012} \\ \beta_{221} & \beta_{222} \\ \beta_{121} & \beta_{122} \\ \beta_{021} & \beta_{022} \end{bmatrix} \begin{bmatrix} i_1 \\ i_2 \end{bmatrix}.
\end{align*}
\]
- the output equation:

\[
\begin{bmatrix}
o_1 \\
o_2
\end{bmatrix} =
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0
\end{bmatrix}
\begin{bmatrix}
x_1 \\
x_2 \\
x_3 \\
x_4 \\
x_5 \\
x_6
\end{bmatrix}
\]

From this discussion with an example having two inputs, two outputs, and a 3\textsuperscript{rd} order common denominator for the plant transfer function matrix (yielding 6 states = denominator order x \# outputs), you can see how to generalize this approach to higher order systems and different numbers of outputs and inputs. It is not guaranteed to yield a system that is controllable, but it is guaranteed to yield a set of states that can be calculated from measurements on the inputs and outputs, together with the rational polynomial coefficients of the known plant transfer function. Using this final state-space model, you can at least attempt to design a controller for the plant using state feedback.