EE 4323 – Industrial Control Systems

Module 2: Preliminaries

Dr. James H. Taylor
Department of Electrical & Computer Engineering
University of New Brunswick
Fredericton, New Brunswick, Canada E3B 5A3
e-mail: jtaylor@unb.ca
web page: www.ece.unb.ca/jtaylor/

25 March 2017

Preliminaries / "Review" – Outline

- Physical / Mathematical Modelling
- Equilibria
- Linearization
- More Math Modelling

Motivation: A lot of controls analysis and design work is **model-based.**

Remember:

- *Linear* or *Linearized* models are only good for **"small signals"**
- *Nonlinear* models are only valid within their **"envelope"**

Preliminaries – Modelling

Q 1. What do we have to control?

- We will often deal with *plant models* in **state-space** form:

$$\begin{aligned} \dot{x}(t) &= f(x, u) \quad (1) \\ y(t) &= h(x, u) \quad (2) \end{aligned}$$

  where x is the plant state vector, $\dot{x}$ its time derivative, y is the plant output vector, u is an input vector, and t is time, respectively; this is **nonlinear time-invariant** (NLTI)

- In a sense this is very general; however we will *not* deal with:
  - Discrete-time plants
  - Plants with distributed-parameter behavior
  - Differential-algebraic systems (e.g., $\dot{x} = f(x, u)$ subject to $g(x, u) = 0$)

- **If we are very lucky** we will have a **linear time-invariant** (LTI) plant:

$$\begin{aligned} \dot{x}(t) &= Ax + Bu \quad (3) \\ y(t) &= Cx + Du \quad (4) \end{aligned}$$

- We may also obtain our plant model in *transfer function* or *block diagram form* (more about that later)

- Realistically, this is usually *not* given to you – you must *derive* or *identify* a suitable model

Example: Back-of-the-Envelope Car Model

A simple line of reasoning proceeds as follows:

- An engine delivers torque depending on the specific RPM, ω_e, and for a specific fuel flow rate $w_f \;\rightarrow\; T(\omega_e, w_f)$
- Torque is translated into force, depending on the transmission gear $g \rightarrow F_e(g, \omega_e, w_f)$; note that RPM and velocity v are proportionate, $\omega_e = K_g(g)v$, so for a given g we can write $F_e(v, w_f)$
- Frictional forces are a primary factor in acceleration and speed: $F_f = f_0 \,\mathrm{sign}(v) + f_1\, v + f_2\, v^2 \,\mathrm{sign}(v)$ (static plus viscous plus drag friction) where v represents the velocity of the car
- So Newton's law yields $Ma = M\dot{v} = F_e - f_0 \,\mathrm{sign}(v) - f_1 v - f_2 v^2 \,\mathrm{sign}(v)$
- Note that v is the state variable (corresponds to kinetic energy);

$$\dot{v} = [\, F_e(v, w_f) - f_0 \,\mathrm{sign}(v) - f_1\, v - f_2\, v^2 \,\mathrm{sign}(v) \,] \,/\, M$$

- **With this simple model you can study the effect of speed on gas mileage. You will observe that the drag force greatly increases fuel consumption if you cruise at 120 km/hr rather than 90 km/hr.**

Preliminaries – Modelling (Cont'd)

For any system modelling builds on basic foundations:

- Start from first principles: $F = M\,a\,,\ v = L\,di/dt\,,\ \ldots$
- System order and state variables are governed by **independent energy storage elements**:
  - $E_L = \frac{1}{2}Li_L^2\,;\ \ v_L = L\dfrac{di_L}{dt}$ (inductor)
  - $E_C = \frac{1}{2}Cv_C^2\,;\ \ i_C = C\dfrac{dv_C}{dt}$ (capacitor)
  - KE $= \frac{1}{2}Mv^2\,;\ F = M\dfrac{dv}{dt}$ (rotation: $= \frac{1}{2}J\omega^2\,;\ T = J\dfrac{d\omega}{dt}$)
  - PE $= \frac{1}{2}K_Sx^2\,;\ \ v = \dfrac{dx}{dt}$ (rotation: $= \frac{1}{2}K_S\theta^2\,;\ \ \omega = \dfrac{d\theta}{dt}$)
  - TE $= \frac{1}{2}C_T\theta^2\,;\ \ q_T = C_T\dfrac{d\theta}{dt}$ (thermal energy; $\theta =$ degrees celsus)
- $F = Ma$ (or $T = J\ddot{\theta}$) and interaction laws are the starting points for mechanical subsystems
- Kirchhov's element and circuit laws are the starting points for electrical subsystems (also, by analogy, thermal ones)
- Energy storage is an important consideration in deciding whether to include or neglect dynamics – e.g., if the energy stored in a motor's field inductance is very small compared with that in the rotor plus load inertia, then neglect it.

An Integrated Modelling Approach

- Use element and connection laws to derive a "raw model"
- Decide on model format:
  - Input/output model (transfer function) – *if it's linear!*
  - State-space model – *nonlinearities easy to add!*
    (state-space models have other advantages as well, e.g., the physical structure of the plant is retained in the model)
- process the raw model equations to produce the end result, a transfer function or state-space model

A brief but intensive coverage of modelling follows, based on pages were taken from Close, Frederick and Newell[1]

[1]Close, Frederick and Newell, *Modeling and Analysis of Dynamic Systems*, John Wiley & Sons, Third Edition.

A Simple Mechanical System

EXAMPLE 3.2

Find the state-variable equations for the system shown in Figure 3.2(a), which is identical to Figure 2.13(a). Write output equations for the tensile force f_{K_2} in the spring K_2 and for the total momentum m_T of the masses.

SOLUTION An appropriate choice of state variables is x_1, v_1, x_2, and v_2, because we can express the velocity of each mass and the elongation of each spring in terms of these four variables and because none of these variables can be expressed in terms of the other three. Because $\dot{x}_1 = v_1$ and $\dot{x}_2 = v_2$, two of the four state-variable equations are available immediately.

The free-body diagrams for the two masses are repeated from Example 2.2 in Figure 3.2(b) and Figure 3.2(c), with all forces labeled in terms of the state variables and the input. By D'Alembert's law,

$$M_1\dot{v}_1 + K_1x_1 - K_2(x_2 - x_1) - B(v_2 - v_1) = 0$$
$$M_2\dot{v}_2 + K_2(x_2 - x_1) + B(v_2 - v_1) = f_a(t)$$

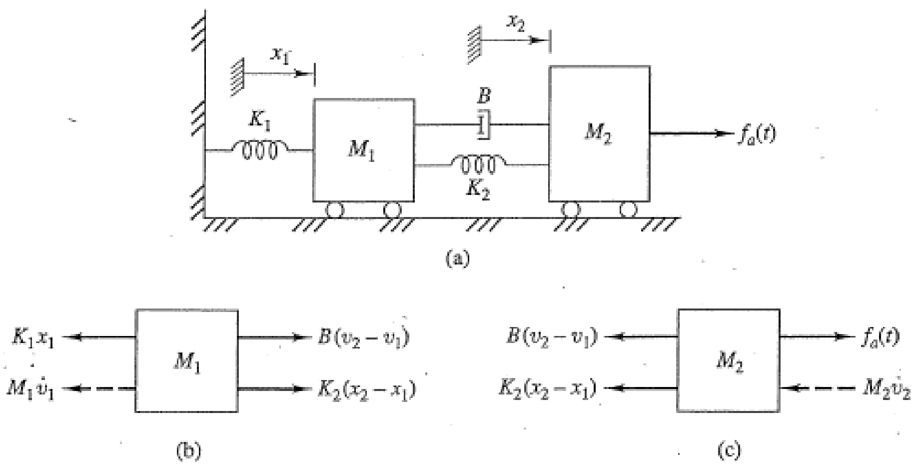


Figure 3.2 (a) Translational system for Example 3.2. (b), (c) Free-body diagrams.

which may be solved for $\dot{v}_1$ and $\dot{v}_2$, respectively. The state-variable equations are

$$\dot{x}_1 = v_1 \tag{3a}$$
$$\dot{v}_1 = \frac{1}{M_1}[-(K_1 + K_2)x_1 - Bv_1 + K_2x_2 + Bv_2] \tag{3b}$$
$$\dot{x}_2 = v_2 \tag{3c}$$
$$\dot{v}_2 = \frac{1}{M_2}[K_2x_1 + Bv_1 - K_2x_2 - Bv_2 + f_a(t)] \tag{3d}$$

If we know the element values, the input $f_a(t)$ for $t \geq 0$, and the initial conditions $x_1(0)$, $v_1(0)$, $x_2(0)$, and $v_2(0)$, then we can solve this set of simultaneous first-order differential equations for x_1, v_1, x_2, and v_2 for all $t \geq 0$. The output equations are

$$\begin{aligned} f_{K_2} &= K_2(x_2 - x_1) \\ m_T &= M_1v_1 + M_2v_2 \end{aligned} \tag{4}$$

Another Mechanical System

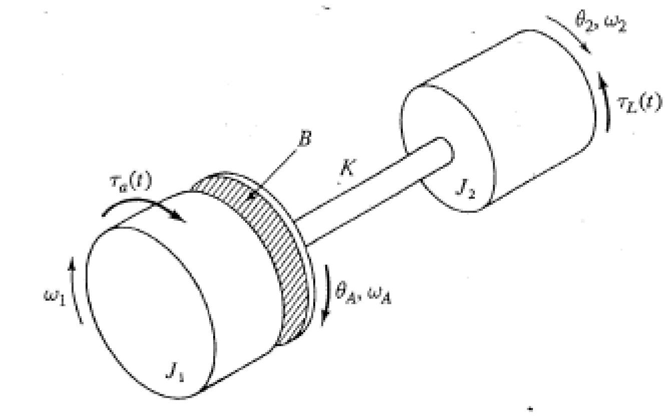


Figure 5.19 Rotational system for Example 5.5.

EXAMPLE 5.5

The system shown in Figure 5.19 consists of a moment of inertia J_1 corresponding to the rotor of a motor or a turbine, which is coupled to the moment of inertia J_2 representing a propeller. Power is transmitted through a fluid coupling with viscous-friction coefficient B and a shaft with stiffness constant K. A driving torque $\tau_a(t)$ is exerted on J_1, and a load torque $\tau_L(t)$ is exerted on J_2. If the output is the angular velocity ω_2, find the state-variable model and also the input-output differential equation.

SOLUTION There are three independent energy-storing elements, so we select as state variables ω_1, ω_2, and the relative displacement θ_R of the two ends of the shaft, where

$$\theta_R = \theta_A - \theta_2 \tag{36}$$

Note that the equation

$$\dot{\theta}_R = \omega_A - \omega_2 \tag{37}$$

is not yet a state-variable equation because of the symbol ω_A on the right side.

Next we draw the free-body diagrams for the two inertia elements and for the shaft, as shown in Figure 5.20. Note that the moment of inertia of the right side of the fluid coupling element is assumed to be negligible. The directions of the arrows associated with the torque $B(\omega_1 - \omega_A)$ are consistent with the law of reaction torques and also indicate that the frictional torque tends to retard the relative motion.

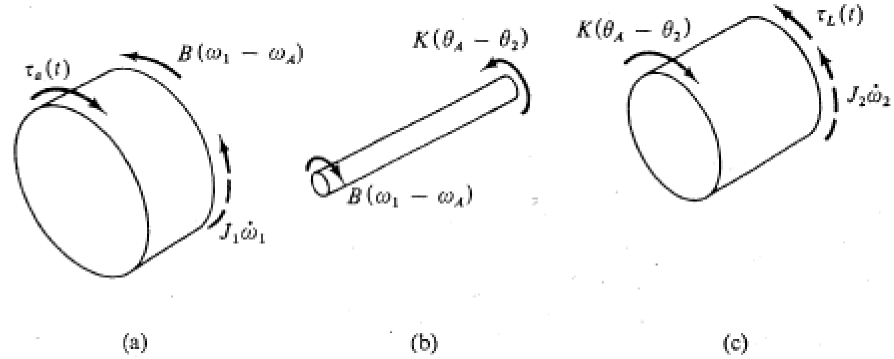


Figure 5.20 Free-body diagrams for Example 5.5.

Another Mechanical System (Cont'd)

▶ Rotational Mechanical Systems

Setting the algebraic sum of the torques on each diagram equal to zero yields the three equations

$$J_1\dot{\omega}_1 + B(\omega_1 - \omega_A) - \tau_a(t) = 0 \quad (38a)$$
$$B(\omega_1 - \omega_A) - K(\theta_A - \theta_2) = 0 \quad (38b)$$
$$J_2\dot{\omega}_2 - K(\theta_A - \theta_2) + \tau_L(t) = 0 \quad (38c)$$

Using (36), we can rewrite (38) as

$$J_1\dot{\omega}_1 + B(\omega_1 - \omega_A) - \tau_a(t) = 0 \quad (39a)$$
$$B(\omega_1 - \omega_A) = K\theta_R \quad (39b)$$
$$J_2\dot{\omega}_2 - K\theta_R + \tau_L(t) = 0 \quad (39c)$$

Substituting (39b) into (39a) and repeating (39c) give

$$\begin{aligned} J_1\dot{\omega}_1 + K\theta_R - \tau_a(t) &= 0 \\ J_2\dot{\omega}_2 - K\theta_R + \tau_L(t) &= 0 \end{aligned} \quad (40)$$

Also from (39b),

$$\omega_A = \omega_1 - \frac{K}{B}\theta_R \quad (41)$$

We substitute (41) into (37) and rearrange (40) to obtain the three state-variable equations

$$\dot{\theta}_R = -\frac{K}{B}\theta_R + \omega_1 - \omega_2 \quad (42a)$$
$$\dot{\omega}_1 = \frac{1}{J_1}[-K\theta_R + \tau_a(t)] \quad (42b)$$
$$\dot{\omega}_2 = \frac{1}{J_2}[K\theta_R - \tau_L(t)] \quad (42c)$$

Because the specified output is one of the state variables, a separate output equation is not needed as part of the state-variable model.

To obtain the input-output equation, we first rewrite (38) in terms of the angular velocities ω_1, ω_2, and ω_A and the torques $\tau_A(t)$ and $\tau_L(t)$. Differentiating (38b) and (38c) and noting that $\dot{\theta}_2 = \omega_2$ and $\dot{\theta}_A = \omega_A$, we have

$$J_1\dot{\omega}_1 + B(\omega_1 - \omega_A) = \tau_a(t)$$
$$B(\dot{\omega}_1 - \dot{\omega}_A) - K(\omega_A - \omega_2) = 0$$
$$J_2\ddot{\omega}_2 - K(\omega_A - \omega_2) + \dot{\tau}_L = 0$$

By using the p-operator technique to eliminate ω_A and ω_1 from these equations, we can obtain the input-output equation

$$\begin{aligned} &\dddot{\omega}_2 + \frac{K}{B}\ddot{\omega}_2 + K\left(\frac{1}{J_1} + \frac{1}{J_2}\right)\dot{\omega}_2 \\ &= \frac{K}{J_1J_2}\tau_a(t) - \frac{1}{J_2}\ddot{\tau}_L - \frac{K}{BJ_2}\dot{\tau}_L - \frac{K}{J_1J_2}\tau_L(t) \end{aligned} \quad (43)$$

Although this result can be viewed as a second-order differential equation in $\dot{\omega}_2$, we will need three initial conditions if we are to determine ω_2 rather than the acceleration $\dot{\omega}_2$. Note that if the load torque $\tau_L(t)$ were given as an algebraic function of ω_2, as it would be in practice, ω_2 would appear in (43). Then the input-output equation would be strictly third order.

A Mechanical System with Gears

EXAMPLE 5.10

Find the state-variable equations for the system shown in Figure 5.26(a), in which the pair of gears couples two similar subsystems.

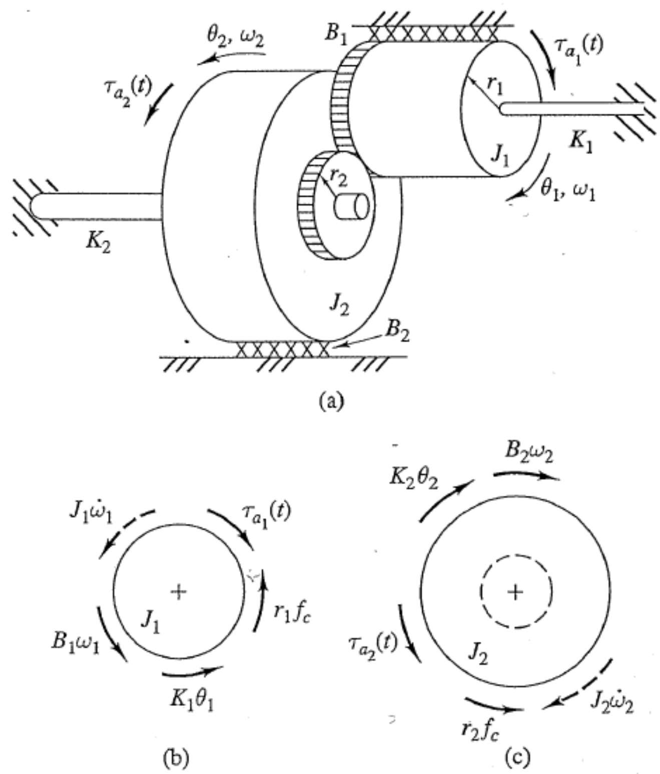


Figure 5.26 (a) System for Example 5.10. (b), (c) Free-body diagrams.

A Mechanical System with Gears (Cont'd)

SOLUTION Because of the two moments of inertia and the two shafts, it might appear that we could choose ω_1, ω_2, θ_1, and θ_2 as state variables. However, θ_1 and θ_2 are related by the gear ratio, as are ω_1 and ω_2. Because the state variables must be independent, either θ_1 and ω_1 or θ_2 and ω_2 constitute a suitable set.

The free-body diagrams for each of the moments of inertia are shown in Figure 5.26(b) and Figure 5.26(c). As in Example 5.9, f_c represents the contact force between the two gears. Summing the torques on each of the free-body diagrams gives

$$J_1\dot{\omega}_1 + B_1\omega_1 + K_1\theta_1 + r_1 f_c = \tau_{a_1}(t) \tag{55a}$$

$$J_2\dot{\omega}_2 + B_2\omega_2 + K_2\theta_2 - r_2 f_c = \tau_{a_2}(t) \tag{55b}$$

By the geometry of the gears,

$$\begin{aligned} \theta_1 &= N\theta_2 \\ \omega_1 &= N\omega_2 \end{aligned} \tag{56}$$

where $N = r_2/r_1$.

Selecting θ_2 and ω_2 as the state variables, we can write $\dot{\theta}_2 = \omega_2$ as the first state-variable equation and combine (55) and (56) to obtain the required equation for $\dot{\omega}_2$ in terms of θ_2, ω_2, $\tau_{a_1}(t)$, and $\tau_{a_2}(t)$. We first solve (55b) for f_c and substitute that expression into (55a). Then, substituting (56) into the result gives

$$(J_2 + N^2 J_1)\dot{\omega}_2 + (B_2 + N^2 B_1)\omega_2 + (K_2 + N^2 K_1)\theta_2 - N\tau_{a_1}(t) - \tau_{a_2}(t) = 0 \tag{57}$$

At this point, it is convenient to define the parameters

$$\begin{aligned} J_{2_{eq}} &= J_2 + N^2 J_1 \\ B_{2_{eq}} &= B_2 + N^2 B_1 \\ K_{2_{eq}} &= K_2 + N^2 K_1 \end{aligned} \tag{58}$$

which can be viewed as the combined moment of inertia, damping coefficient, and stiffness constant, respectively, when the combined system is described in terms of the variables θ_2 and ω_2. For example, it is common to say that $N^2 J_1$ is the equivalent inertia of disk 1 when that inertia is reflected to shaft 2. Similarly, $N^2 B_1$ and $N^2 K_1$ are the reflected viscous-friction coefficient and stiffness constant, respectively. Hence the parameters $J_{2_{eq}}$, $B_{2_{eq}}$, and $K_{2_{eq}}$ defined in (58) are the sums of the parameters associated with shaft 2 and the corresponding parameters reflected from shaft 1.

With the new notation, we can rewrite (57) as

$$J_{2_{eq}}\dot{\omega}_2 + B_{2_{eq}}\omega_2 + K_{2_{eq}}\theta_2 - N\tau_{a_1}(t) - \tau_{a_2}(t) = 0 \tag{59}$$

and the state-variable equations are

$$\begin{aligned} \dot{\theta}_2 &= \omega_2 \\ \dot{\omega}_2 &= \frac{1}{J_{2_{eq}}}\left[-K_{2_{eq}}\theta_2 - B_{2_{eq}}\omega_2 + N\tau_{a_1}(t) + \tau_{a_2}(t)\right] \end{aligned} \tag{60}$$

Note that the driving torque $\tau_{a_1}(t)$ applied to shaft 1 has the value $N\tau_{a_1}(t)$ when reflected to shaft 2.

An Electro-mechanical System

EXAMPLE 10.1

Derive the state-variable equations for a dc motor that has a constant field voltage E_F, an applied armature voltage $e_i(t)$, and a load torque $\tau_L(t)$. Also obtain the input-output equation with ω as the output, and determine the steady-state angular velocities corresponding to the following sets of inputs: $e_i(t) = E$, $\tau_L(t) = 0$ and $e_i(t) = 0$, $\tau_L(t) = T$.

SOLUTION The basic motor diagram in Figure 10.12 is repeated in Figure 10.13, with the specified field and armature input voltages added. The field voltage E_F is constant, so the field current will be the constant $i_F = E_F/R_F$. We can write the electromechanical driving torque τ_e and the induced voltage e_m as

$$\begin{aligned} \tau_e &= \alpha i_A \\ e_m &= \alpha\omega \end{aligned} \tag{28}$$

where α is a constant defined by

$$\alpha = \gamma\,\phi(i_F) \tag{29}$$

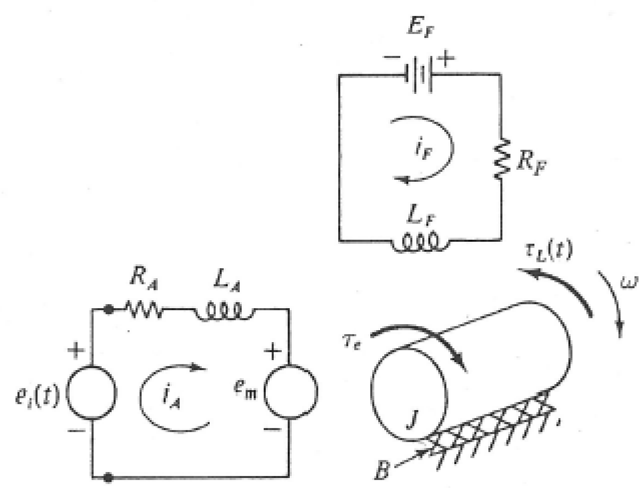


Figure 10.13 DC motor with a constant field current.

Equation (28) provides the connection between the electrical and mechanical domains – formulating the model simply requires applying Kirchhoff's law to the circuit and Newton's law to the rotor.

An Electro-mechanical System (Cont'd)

and γ is given by (25). We select i_A and ω as the state variables and write a voltage equation for the armature circuit and a torque equation for the rotor. Then, using (28) and solving for the derivatives of the state variables, we find the state-variable equations to be

$$\begin{aligned} \frac{di_A}{dt} &= \frac{1}{L_A}[-R_A i_A - \alpha\omega + e_i(t)] \\ \dot{\omega} &= \frac{1}{J}[\alpha i_A - B\omega - \tau_L(t)] \end{aligned} \tag{30}$$

In order to find the system's transfer functions, we apply the Laplace transform to (30) and assume that there is no initial stored energy. With $i_A(0) = 0$ and $\omega(0) = 0$, we have

$$L_A s I_A(s) = -R_A I_A(s) - \alpha\Omega(s) + E_i(s)$$

$$Js\Omega(s) = \alpha I_A(s) - B\Omega(s) - \tau_L(s)$$

Eliminating $I_A(s)$ from this pair of algebraic equations and solving for the transformed output $\Omega(s)$, we find that

$$\Omega(s) = H_1(s)E_i(s) + H_2(s)\tau_L(s)$$

where

$$H_1(s) = \frac{\alpha/JL_A}{P(s)} \tag{31a}$$

$$H_2(s) = \frac{-(1/J)s - (R_A/JL_A)}{P(s)} \tag{31b}$$

$$P(s) = s^2 + \left(\frac{R_A}{L_A} + \frac{B}{J}\right)s + \left(\frac{R_A B + \alpha^2}{JL_A}\right) \tag{31c}$$

The quantity $H_1(s)$ is the transfer function relating the output velocity and input voltage when $\tau_L(t) = 0$. $H_2(s)$ relates $\Omega(s)$ and $\tau_L(s)$ when $e_i(t) = 0$. The general input-output differential equation is

$$\ddot{\omega} + \left(\frac{R_A}{L_A} + \frac{B}{J}\right)\dot{\omega} + \left(\frac{R_A B + \alpha^2}{JL_A}\right)\omega = \frac{\alpha}{JL_A}e_i(t) - \frac{1}{J}\dot{\tau}_L - \frac{R_A}{JL_A}\tau_L(t) \tag{32}$$

As expected, both the electrical and the mechanical parameters contribute to the system's undamped natural frequency ω_n and to the damping ratio ζ.

To solve for the steady-state motor speed when the voltage source has the constant value $e_i(t) = E$ and when $\tau_L(t) = 0$, we omit all derivative terms in (32) and substitute these values for $e_i(t)$ and $\tau_L(t)$, obtaining

$$\omega_{ss} = \frac{\alpha E}{R_A B + \alpha^2} \tag{33}$$

In physical terms, the motor will run at a constant speed such that the driving torque $\tau_e = \alpha i_A$ exactly balances the viscous-frictional torque $B\omega_{ss}$. However, the steady-state armature current is $i_A = (E - e_m)/R_A$, where $e_m = \alpha\omega_{ss}$. Making the appropriate substitutions, we again obtain (33). We get the same expression by examining $H_1(0)$.

When $e_i(t) = 0$ and $\tau_L(t)$ has the constant value T, the steady-state solution to (32) is

$$\omega_{ss} = -\frac{R_A T}{R_A B + \alpha^2} \tag{34}$$

which indicates that the motor will be driven backward at a constant angular velocity. We can also obtain (34) by noting that $\omega_{ss} = TH_2(0)$. To understand the behavior of the motor

An Electro-mechanical System (Cont'd)

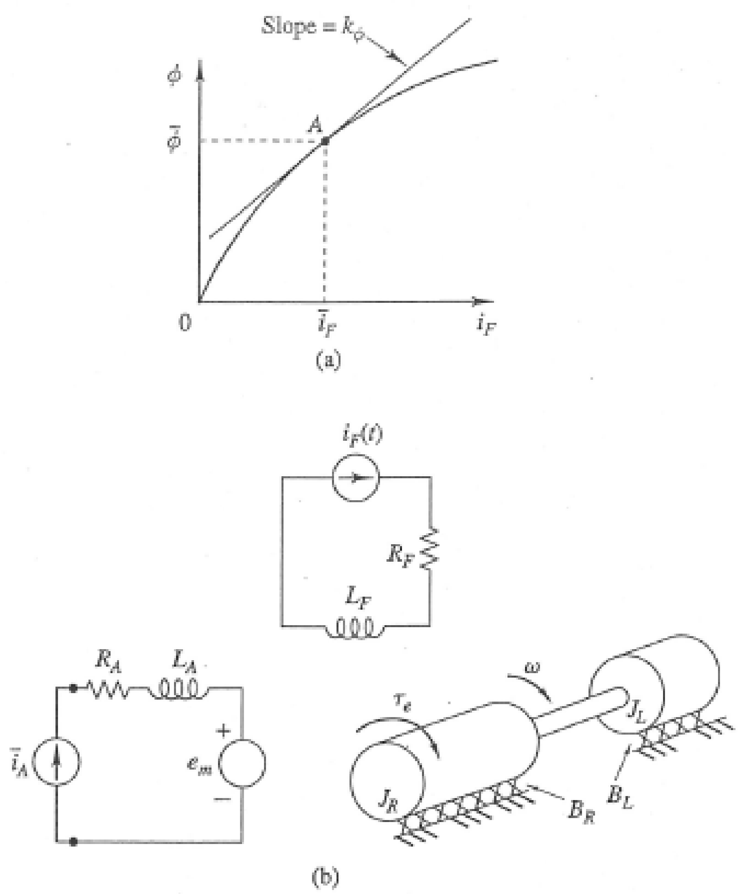


Figure 10.14 DC motor for Example 10.2. (a) Nonlinear field characteristic. (b) Diagram used for analysis.

under this condition, we observe that the electromechanical torque, $\tau_e = \alpha i_A$, must balance the sum of the load torque T and the viscous-frictional torque $B\omega_{ss}$. Thus an armature current must flow that will make $\alpha i_A = T + B\omega_{ss}$. Furthermore, because the applied armature voltage is zero and $e_m = \alpha\omega_{ss}$, it follows that $i_A R_A = -\alpha\omega_{ss}$. As anticipated, solving these two equations for ω_{ss} results in (34).

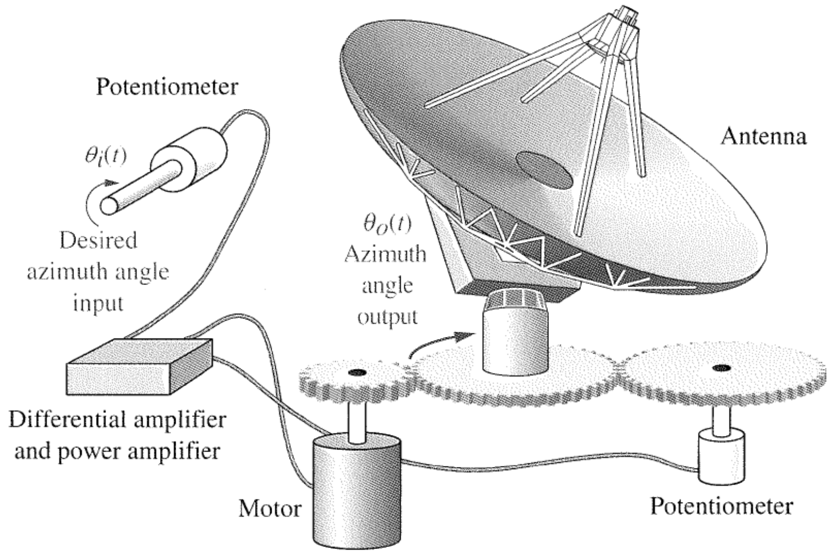
In this situation, the motor is acting as a **generator** connected to a load of zero resistance. Part of the mechanical power supplied by the load torque is being converted to electrical form and dissipated in the armature resistance R_A. Basically, we may think of a generator as a motor that is being driven mechanically and that delivers a portion of the power to an electrical load connected across the armature terminals.

You can verify that when the constant applied voltage is $e_i(t) = E$ and the constant load torque is $\tau_L(t) = \alpha E/R_A$, the steady-state motor speed is $\omega_{ss} = 0$. In this condition, the electromechanical driving torque τ_e exactly matches the load torque and the motor is stalled.

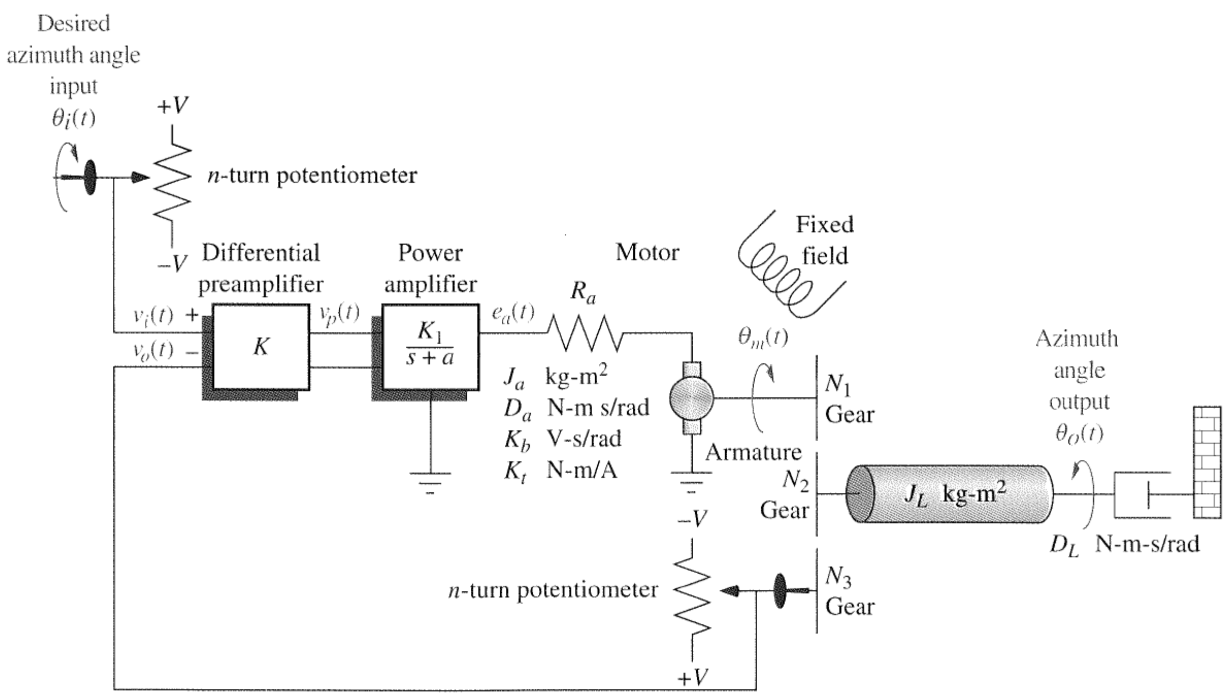
Modelling Something Real

Antenna Azimuth Position Control System

Layout



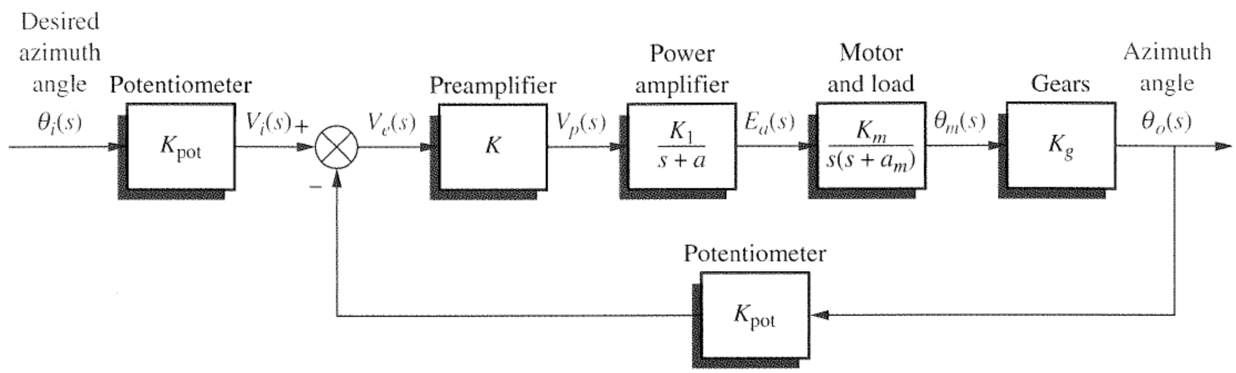
Schematic



Let's discuss the sensor ...

Modelling Something Real (Cont'd)

Block Diagram



We can derive a state-space model from the block diagram as follows:

- Choose the states: $x_1 = E_a$, $x_2 = \theta_m$, $x_3 = \dot{\theta}_m$
- For the motor: $\dot{x}_3 + a_m x_3 = K_m E_a$ or

$$\begin{aligned} \dot{x}_2 &= x_3 \\ \dot{x}_3 &= K_m x_1 - a_m x_3 \end{aligned}$$

- For the power amplifier: $(s + a)X_1 = V_p$ or $\dot{x}_1 = V_p - a x_1$
- Using the entire block diagram: $V_p = KV_e = KK_{pot}(\theta_i - \theta_m)$
- Therefore we have:

$$\begin{aligned} \dot{x}_1 &= -a x_1 + KK_{pot}(\theta_i - \theta_m) \\ \dot{x}_2 &= x_3 \\ \dot{x}_3 &= K_m x_1 - a_m x_3 \end{aligned}$$

Modelling Considerations Revisited

When constructing a mathematical plant model, the most basic rule is: Make it detailed enough to be realistic *in the context of your problem*, but **no more detailed than that**. There are many issues to consider – for example:

- Model order (linear **or** nonlinear) - should you include:
  - parasitic capacitances and inductances in an electronic circuit? Which ones?
  - flexible modes in a mechanical link? How many?
  - motor lags (RL) in a robotic manipulator?
  - transport or computational delay (include using a Padé approximation [2])?
  - *Et cetera*

  These decisions are made based on common sense and experience, with some trial and error that can be done by looking at simulations and/or frequency responses. It will probably be necessary to iterate as:

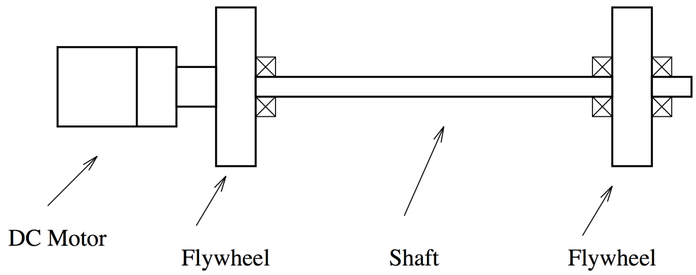
  - the problem becomes better understood
  - control system design proceeds ($\rightarrow$ operating regime)
  - you need to determine why something doesn't work
  - *Et cetera*

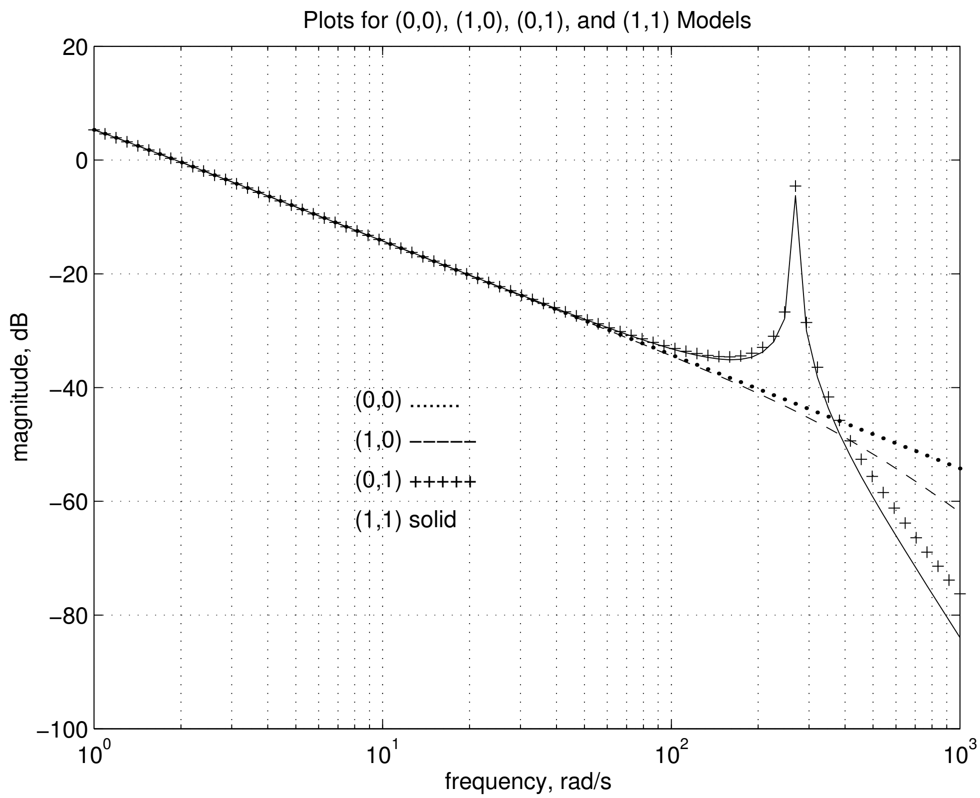
---

[2] That is: $e^{-sT} \cong (1 - sT)$ **or** $\cong 1/(1 + sT)$ **or** $\cong (1 - sT/2)/(1 + sT/2)$ – e.g., Ralston & Wilf, Mathematical Methods for Digital Computers, Wiley, 1967. This approximation approach is very useful for simulation and design (e.g., root-locus methods).

Modelling Considerations (Cont'd)

Example of determining model order:





(1,1) $\rightarrow$ with motor lag and flexible mode

- For a low-gain "sluggish" control system the (0,0) model suffices
- The flexible mode is more important than the motor lag over this frequency range
- For high-performance control flexible modes **do** matter!

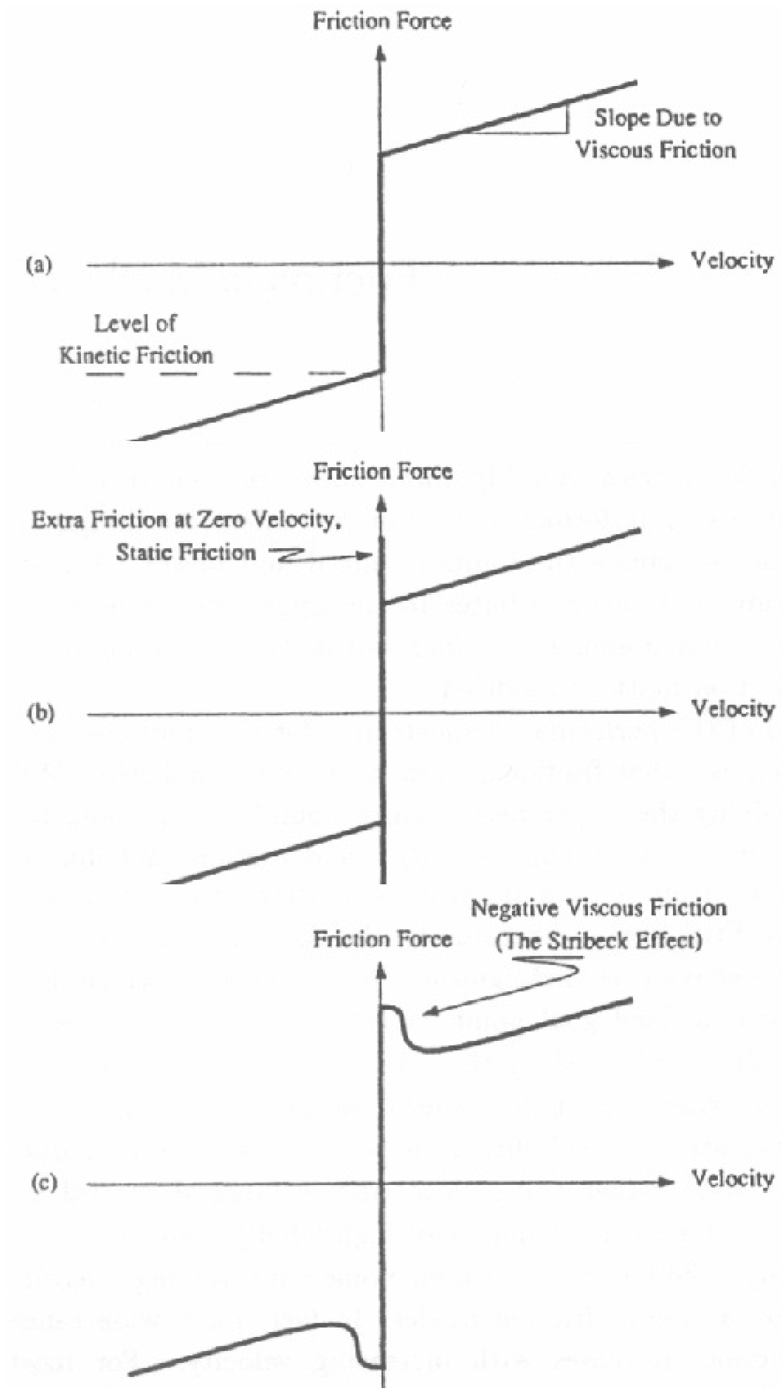
Modelling Considerations (Cont'd)

- Model nonlinearities - should you include:
  - inductor / spring nonlinearities
  - nonlinear resistance / friction
  - motor saturation
  - relays with hysteresis
  - gear backlash
  - *Et cetera*
- Considerations:
  - **every** system saturates, causing the most basic limitation to control system performance!
  - discontinuities at the operating point (e.g., relays) are almost always important
  - small-signal linearizations of relays, deadzones, backlash, ... are **undefined**;
  - most include/exclude decisions depend on **expected signal amplitude**.

Bottom line: It's often best to have several models: a higher-order, more nonlinear "high-fidelity" or **"truth" model** and one or more **"simplified" or linearized models**; work primarily with validated simplified models, checking against the "truth" model sparingly as needed.

Modelling Considerations (Cont'd)

Friction is particularly "interesting":



Modelling Considerations (Cont'd)

Did I mention that friction is "interesting"?

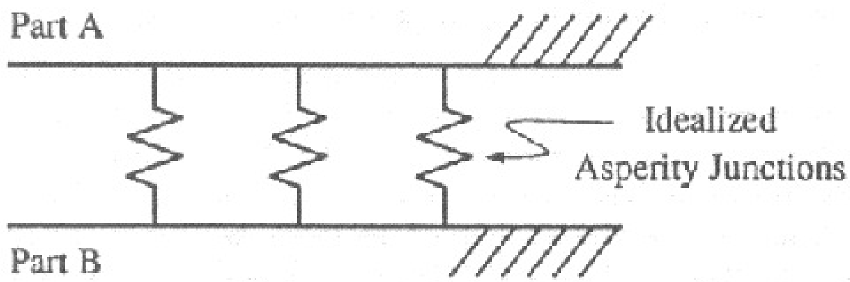


Figure 2.6 Idealized Contact Between Engineering Surfaces in Static Friction. Asperity Contacts Behave Like Springs.

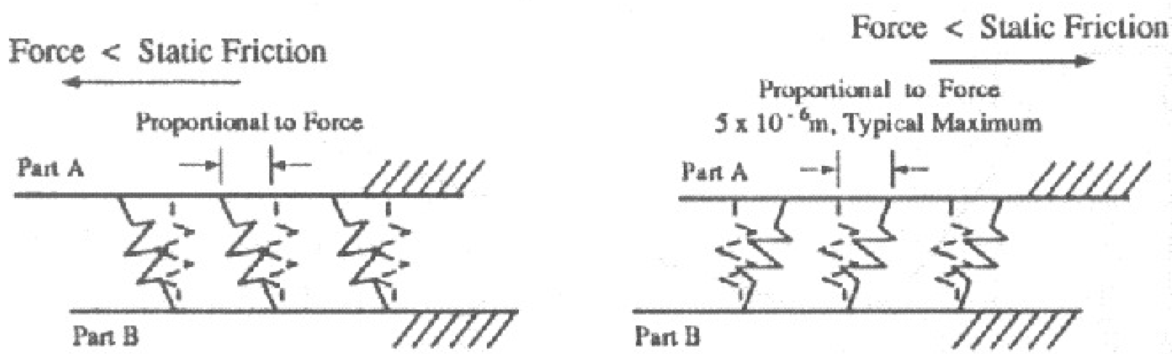
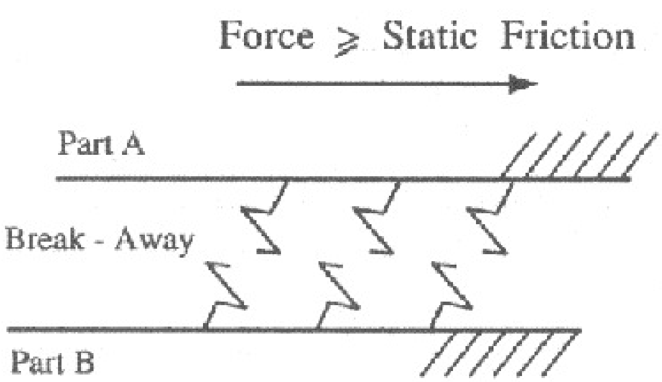
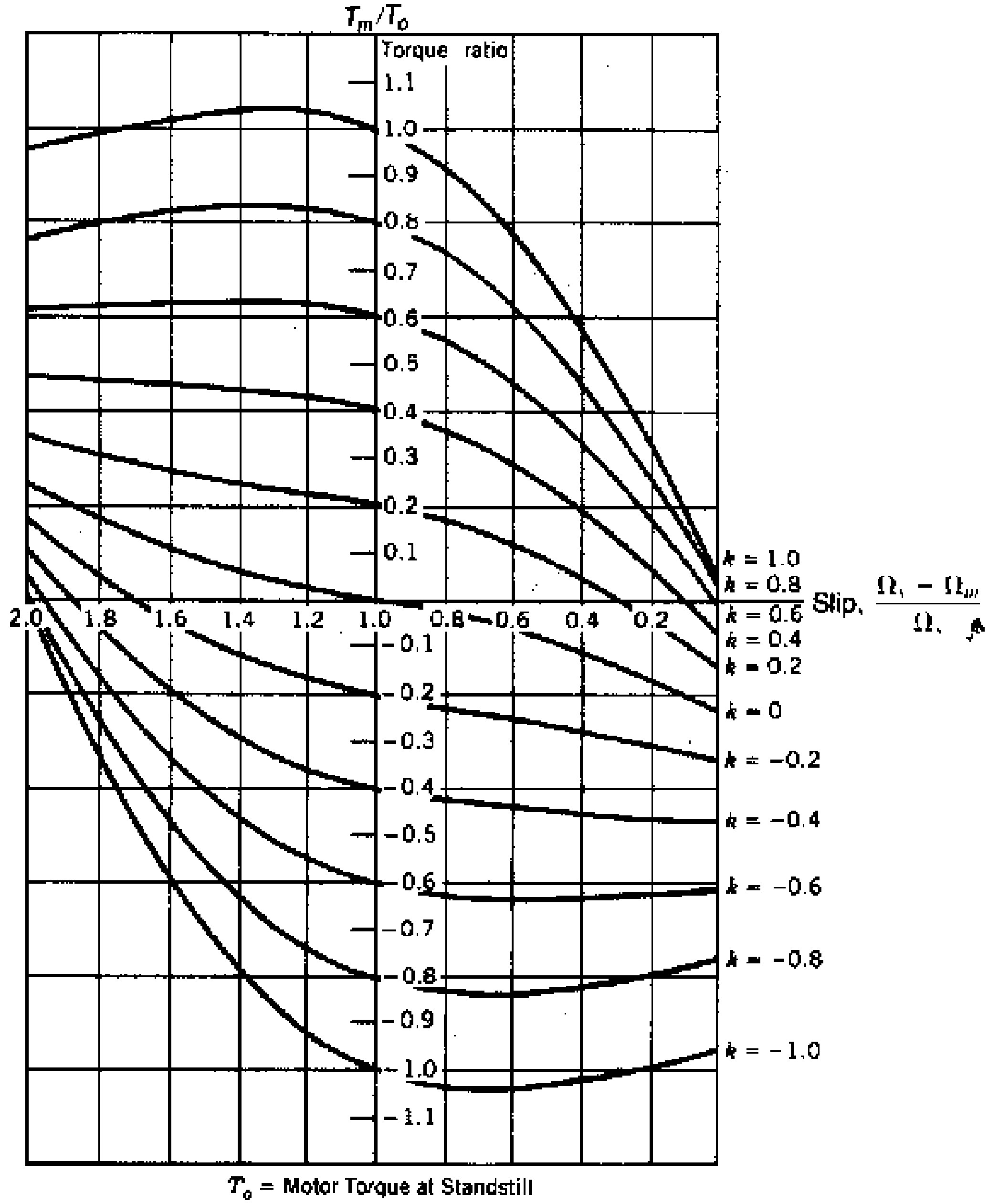


Figure 2.7 Asperity Deformation under Applied Force, the Dahl Effect.



Modelling Considerations (Cont'd)

Actuators are **always** nonlinear:



(AC servomotor characteristic) ... and so are sensors. They are also **limited** (saturate)

Modelling Considerations (Cont'd)

Finally, some **convenient model manipulations** are:

1. Higher-order scalar systems to state-space form: Given

$$\zeta^{(n)} = \Phi(\zeta, \dot{\zeta}, \ddot{\zeta}, \ldots, u, t)$$

one may define $x = [\zeta \ \ \dot{\zeta} \ \ \ddot{\zeta}, \ldots]^T$ to obtain:

$$\dot{x} = \begin{bmatrix} \dot{\zeta} \\ \ddot{\zeta} \\ \vdots \\ \zeta^{(n)} \end{bmatrix} = \begin{bmatrix} x_2 \\ x_3 \\ \vdots \\ \Phi(x_1, x_2, x_3, \ldots, u, t) \end{bmatrix}$$

This can be done whenever it is possible to solve for the highest-order derivative as a function of lower-order derivatives; for example, for a mechanical system you can use Newton's law to obtain $M\,\ddot{\zeta} = \ldots$

Modelling Considerations (Cont'd)

2. Transfer function models to state-space form: Given

$$G(s) = \frac{p(s)}{q(s)} = d + \frac{c_n s^{n-1} \cdots + c_2 s + c_1}{s^n + a_n s^{n-1} \cdots + a_2 s + a_1}$$

one may simply construct *a particular* corresponding state-space model as follows:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & \cdot\ \cdot & 0 \\ 0 & 0 & 1 & 0 & \cdot\ \cdot & 0 \\ 0 & 0 & 0 & 1 & \cdot\ \cdot & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot\ \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot\ \cdot & \cdot \\ 0 & 0 & 0 & 0 & \cdot\ \cdot & 1 \\ -a_1 & -a_2 & -a_3 & -a_4 & \cdot\ \cdot & -a_n \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \cdot \\ \cdot \\ 0 \\ 1 \end{bmatrix} \qquad (5)$$

$$C = [c_1 \;\; c_2 \;\; c_3 \;\; c_4 \;\; \cdot \;\; \cdot \;\; c_n] \;,\;\; D = [d] \qquad (6)$$

this *phase-variable canonical form* (PVCF) (also called *controllable canonical form*) is very common and useful. Note that **improper** transfer functions (order($p(s)$) > order($q(s)$)) **cannot** be represented in state-space form; such systems are also **not physically realizable** – appropriate high-frequency roll-off must be added. Also, if the transfer function is strictly proper (order($p(s)$) < order($q(s)$)) we have $d = 0$ – **this represents the real world**.

This is *just one possible* state-space model – a "real" state-space model (based on physics) is true to the structure of the system; a PVCF model (as well as a transfer function model) is not.

Modelling Considerations (Cont'd)

3. Given:

$$G(s) = \frac{5(s^2 + 2s + 4)}{s^2 + 5s + 6} \tag{7}$$

4. Find: an equivalent state-space model –

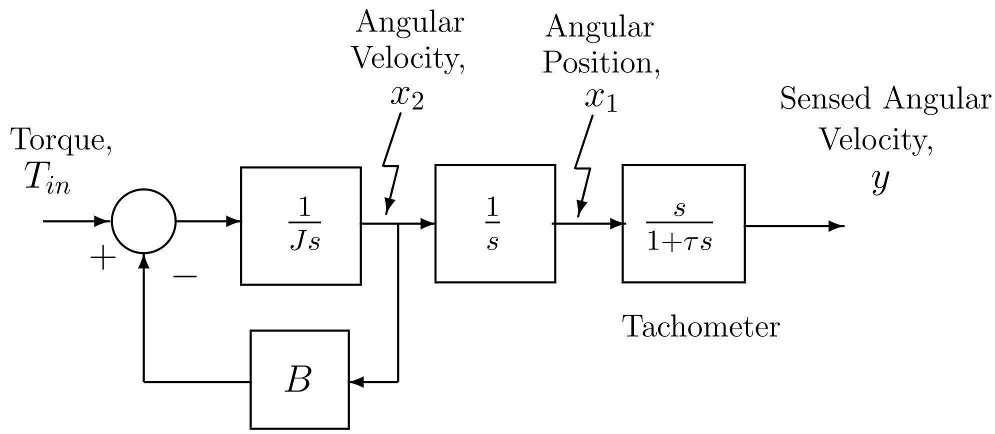
$$\begin{aligned} G(s) &= \frac{5(s^2 + 5s + 6) + (10 - 25)s + (20 - 30)}{s^2 + 5s + 6} \\ &= 5 - \frac{15s + 10}{s^2 + 5s + 6} \end{aligned}$$

5. Therefore:

$$\begin{aligned} \dot{x} &= \begin{bmatrix} 0 & 1 \\ -6 & -5 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \\ y &= \begin{bmatrix} -10 & -15 \end{bmatrix} x + 5\,u \end{aligned}$$

Modelling Considerations (Cont'd)

6. Using the transfer function model to state-space transform for one block in a system: Given



   (a) Identify x_3 as the tachometer state variable

   (b) Manipulate $G(s) = s/(1 + \tau s) = \left(\frac{1}{\tau} - \frac{1/\tau^2}{(s+1/\tau)}\right)$

   (c) Therefore $a = -\frac{1}{\tau}$, $b = 1$, $c = -\frac{1}{\tau^2}$ and $d = \frac{1}{\tau}$

   (d) Now, note that b is the direct input gain from x_1 to $\dot{x}_3$ and d is the direct output gain from x_1 to y. Thus these elements can be substituted into your linear state-space model as follows:

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -\frac{B}{J} & 0 \\ 1 & 0 & -\frac{1}{\tau} \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ \frac{1}{J} \\ 0 \end{bmatrix} \tag{8}$$

$$C = \begin{bmatrix} \frac{1}{\tau} & 0 & -\frac{1}{\tau^2} \end{bmatrix}, \quad D = [\,0\,] \tag{9}$$

7. This can also be done in a **nonlinear** model setting (e.g., if friction is $B_1\,x_2 + B_0\,\mathrm{sign}(x_2)$ in the model above).

Equilibria (Operating Points)

Q 2. What is the steady-state operating point?

- Finding **simple equilibria** involves determining steady-state points for $u = u_0$:
  - LTI – solutions of $0 = Ax_0 + Bu_0$ to get x_0 for a given u_0 = constant; assumes A is nonsingular, so $x_0 = -A^{-1}Bu_0$
  - NLTI – solutions of $0 = f(x_0, u_0)$ to get x_0 for a given u_0; involves solving n simultaneous equations in n unknowns; simple equilibria exist if one or more **isolated** solutions exist

  Note that the properties of an LTI system *do not depend* on the equilibrium x_0; in the NLTI case this is **not so!**

- Systems may have **nonsimple equilibria** – examples:
  - If $\text{rank}(A) = \tilde{n} < n$ then there is an $\tilde{n}$-dimensional subspace wherein an equilibrium exists; in the remaining $(n - \tilde{n})$-dimensional subspace the states are arbitrary
  - Similar effects occur in nonlinear systems – example: $M\ddot{x} + f(\dot{x}) = F_e(t)$ for $F_e(t) = F_0$ has a constant velocity equilibrium v_0 satisfying $f(v_0) = F_0$; therefore the nonsimple equilibrium is $x(t) = x_0 + v_0 t$ where x_0 is arbitrary (e.g., your position when $F_e(t)$ first became F_0)

Equilibria (Cont'd)

- Equilibria can be found using several approaches:
  - **If** the system is **very simple** (e.g., our "car model" or a low-order linear model), then solve for x_0 analytically
  - **If** the system $\dot{x} = f(x, u)$ is **asymptotically stable**, then simply set $u = u_0$ and simulate to steady state
  - **If** the equilibrium equation can be written as a **polynomial** then write the polynomial in MATLAB (vector) form, e.g., for $K_1 x_0 + K_3 x_0^3 = F_0$ then `poly = [ K_3 0 K_1 -F_0 ]` then `roots(poly)` will provide the answer, assuming K_1 *etc.* are defined in MATLAB's workspace
  - **If** you have a MATLAB model you can use the MATLAB function `fminsearch` for equilibrium-finding:
    - Given: a simulation model (ODE) in file `car_mdl.m`
    - Create: a "shell" file called `xdot_mag`, to achieve correct input/output form for MATLAB:

      ```
      function objective = mag_sq_xdot(x)
      xdot = car_mdl(0,x);  objective = xdot'*xdot;
      ```
    - Invoke `fminsearch`:
      `x_eq = fminsearch('xdot_mag',[ 0 0 ])`. Note that `[ 0 0 ]` is your initial guess here; give your best estimate. If there are more than one equilibrium then different "guesses" will yield different results

    Note: the model input must be simply defined $u =$ constant in file `car_mdl.m`

- **Pause: this is hard to comprehend without a concrete example, which I will present now.**

Properties Near Equilibria

Given a system $\dot{x} = f(x, u)$ and some value for u_0 and the corresponding equilibrium $x_0 \rightarrow$ one also has a corresponding output value, $y_0 = h(x_0, u_0)$. Define the perturbation variables $\delta x = x - x_0$, $\delta u = u - u_0$, $\delta y = y - y_0$.

Then, *if* the perturbations are small *and if* continuous partial derivatives exist at (x_0, u_0) the behavior of the original system near x_0 is similar to that of $\dot{\delta x} = A\,\delta x + B\,\delta u$, $\delta y = C\,\delta x + D\,\delta u$ where

$$A = \left[\frac{\partial f}{\partial x}\right]_{x_0,u_0} \quad , \quad B = \left[\frac{\partial f}{\partial u}\right]_{x_0,u_0} \quad , \tag{10}$$

$$C = \left[\frac{\partial h}{\partial x}\right]_{x_0,u_0} \quad , \quad D = \left[\frac{\partial h}{\partial u}\right]_{x_0,u_0} \tag{11}$$

The procedure in Eqns (10,11) is called **small signal linearization** (SSL), for obvious reasons. The A-matrix reveals the dynamic nature of the system, through its **eigenvalues**.

As a specific example, we will see later (using Lyapunov stability theory) that if A is asymptotically stable then $\dot{x} = f(x, u)$ is also AS *for sufficiently small perturbations* about $(x_0, u_0) \Rightarrow$ "**infinitesimal stability**"; the same is true for instability but **not** for marginal stability.

Generating SSL Models

The arrays in Eqns (10,11) can be found in three ways:

- Analytically, by hand
- Analytically, by symbolic manipulation – MAPLE, MACSYMA

(these methods may be more insightful and can be machine coded, but the size and class of problems that can be treated is limited);

- Numerically –

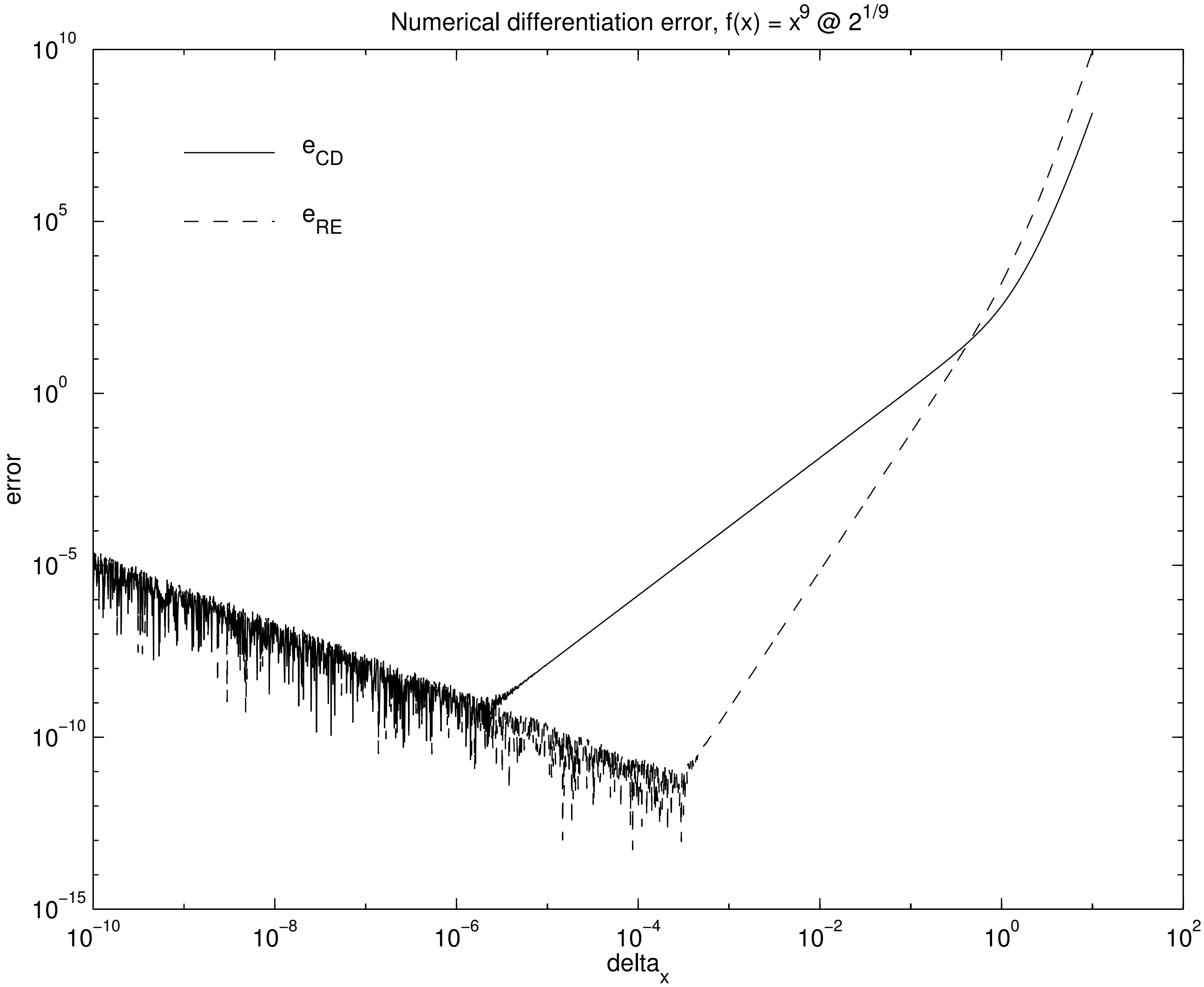
$$Df_{CD}(\delta) = \frac{f(x_0 + \delta) - f(x_0 - \delta)}{2\delta} \qquad (12)$$

  - This **central difference formula** is exact for terms up to x^2 (i.e., has *truncation error* $0(\delta^2)$)
  - Please be aware that numerical differentiation is **not** trivial, due to truncation and round-off error[3] – yes, engineers have to be alert when using numerical methods (algorithms)! This was **really** an issue decades ago with low-precision arithmetic; not so much now.

[3] J. H. Taylor and A. J. Antoniotti, "Linearization Algorithm for Computer-Aided Control Engineering", *IEEE Control Systems, Vol. 13,* No. 2, pp. 58-64, April 1993.

Generating SSL Models (Cont'd)

Example of numerical differentiation error:



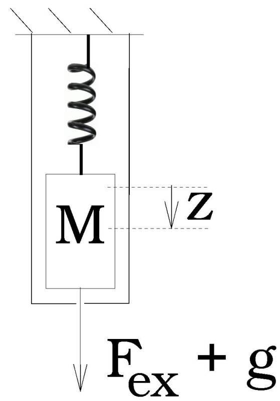
A more accurate formula based on "Richardson extrapolation" is

$$Df_{RE} = [4Df_{CD}(\delta) - Df_{CD}(2\delta)]/3 \tag{13}$$

which is exact for terms up to x^4 (has truncation error $0(\delta^4)$; see plot e_{RE} on the plot above). Note that the straight-line segments have slopes based on $0(\delta^2)$ or $0(\delta^4)$.

Equilibria and SSL Model for a Hanging Mass

Given: a mass M is hanging by a spring that exerts a linear plus cubic force when stretched z units, $f_s = K_1 z + K_3 z^3$ and which moves in a fluid-filled cylindar that yields a friction force that is viscous plus drag (square-law), $f_f = B_v \dot{z} + B_d \dot{z}|\dot{z}|$. In addition, a downward force F_{ex} acts on the mass. According to Newton's law the equation of motion is:



$$M\ddot{z} + B_v \dot{z} + B_d \dot{z}|\dot{z}| + K_1 z + K_3 z^3 = F_{ex} + Mg \tag{14}$$

If we choose $x^T = [z \;\; \dot{z}]$ then the nonlinear state-space model is:

$$\dot{x} = \begin{bmatrix} \dot{z} \\ \ddot{z} \end{bmatrix} = \begin{bmatrix} x_2 \\ g + \frac{1}{M}[F_{ex} - K_1 x_1 - K_3 x_1^3 - B_v x_2 - B_d x_2 |x_2| \end{bmatrix} \tag{15}$$

"By inspection" the equilibrium is found by setting $F_{ex} = F_0$, then we require $\dot{z} = x_{20} = 0$ (velocity is zero) and $Mg + F_0 - K_1 x_{10} - K_3 x_{10}^3 = 0$ (which yields the spring extension at rest).

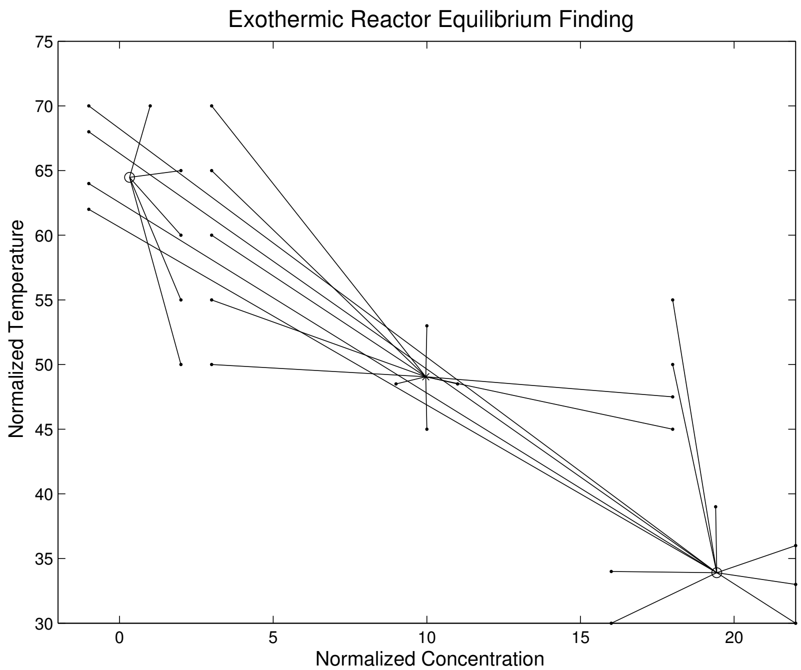
Based on this, we take the partial derivatives of nonlinear state-space model to obtain:

$$A = \begin{bmatrix} 0 & 1 \\ -(K_1 + 3K_3 x_{10}^2) & -(B_v + 2B_d|x_{20}|) \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ \frac{1}{M} \end{bmatrix} \tag{16}$$

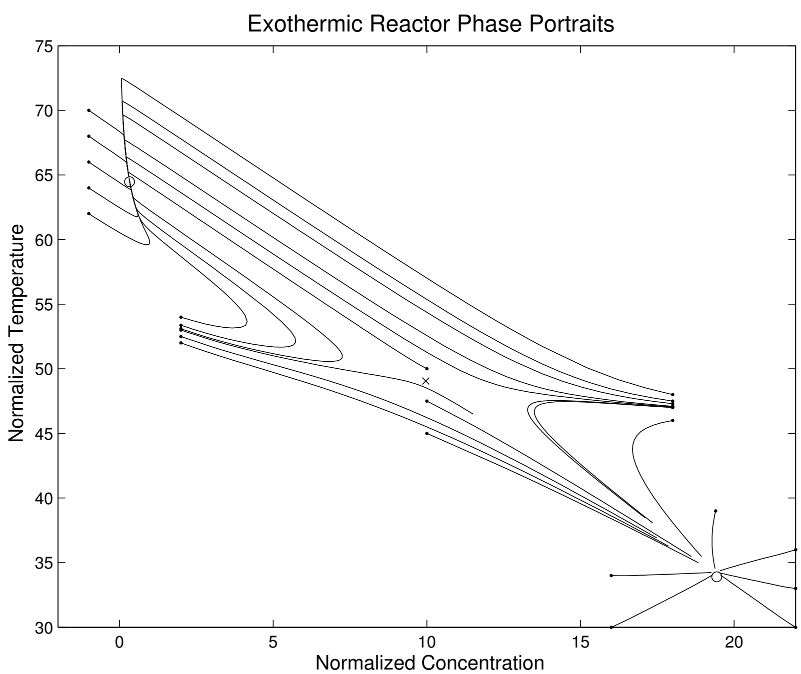
Note that the Mg term has a direct influence on the equilibrium displacement $x_{10} = z_0$ as well as an indirect impact on the small-signal linearization model due to the value of the stretched spring constant $K_1 + 3K_3 x_{10}^2$.

Value of Equilibria & SSL Models

Application: Understanding the behavior of an **exothermic reactor** – The equilibrium-finding and eigenvectors of the SSL A matrix of the reactor reveal a lot about its dynamic behavior:



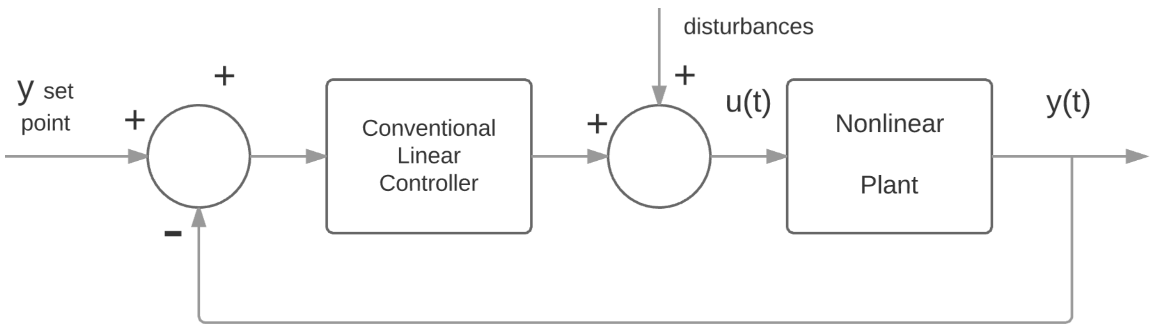
…and a large number of simulations confirms that information:



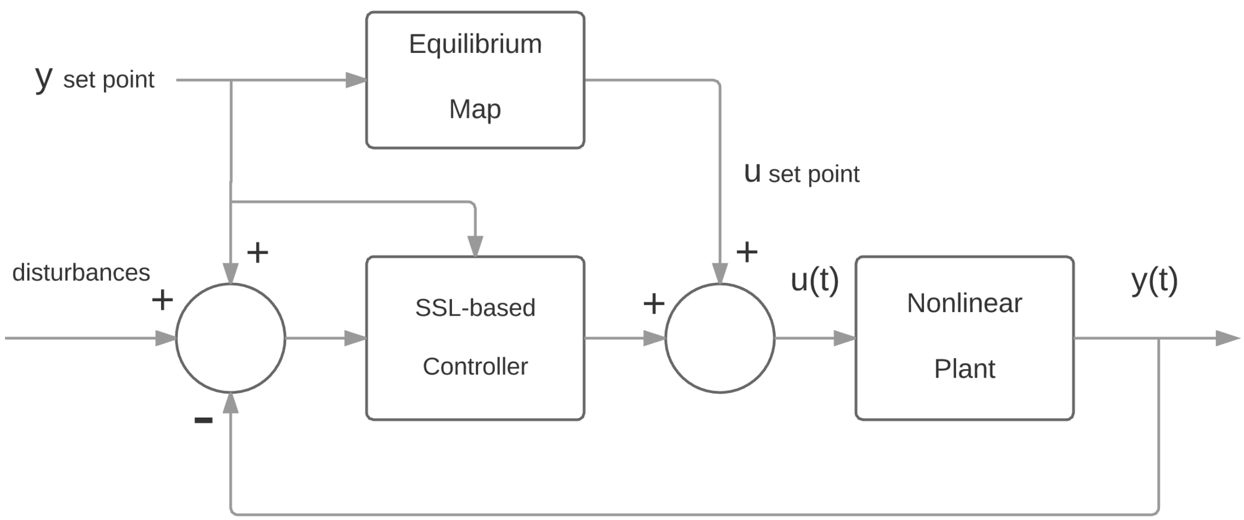
The central **unstable** equilibrium is the desired operating point …

Role of Equilibria & SSL Models in Controls

- This is the usual concept of control systems implementation:



- ... but robust control of a nonlinear process often requires full use of equilibria and linearized models:



A typical example application is aircraft flight control – the controller deals only with **perturbations (small signals)**; the controller is **gain-scheduled** according to $y_{\text{set point}}$.