Lafayette College
ME 352: Dynamics of Physical Systems and Electric Circuits

Problem Set No. 6
Due Friday, March 9, 2001 (if your here, since it will be a comp. Day)
or Thursday March 8, 2001 (if you will be leaving for Spring Break)

STATE-SPACE REPRESENTATION OF DYNAMIC SYSTEMS

Overview

Our method to determine the response of first order dynamic systems to step inputs is to:

1. Draw a linear graph.
2. Write an equation list of mathematical statements of physical truth
3. Reduce the equation list to a single differential system equation by eliminating all power variables except the input and output variable.
4. Solve the differential equation for the response of the desired output variable.

This is a very efficient method for first order systems because once we have identified the time constant from the system equation for one output variable we can then write the response equations for all of the other power variables after only identifying their initial and final values. However, most machines and processes have more than one independent energy storage and, consequently, are higher order. The effort to reduce the equation list to a differential system equation increases roughly with the square of the number of the branches in the linear graph, which tends to increase with increasing order. A reduction for a third order system is roughly nine times the effort of a first order system. The state variable method allows us to perform a partial reduction of the equation list by keeping the input variable and a set of "state variables", rather than just the input and the output variables of a complete reduction. The partial reduction of the equation list produces a set of simulataneous first order differential equations instead of a single higher differential equation. We will use Mathematica or Mathcad to solve this set of simultaneous differential equations for the "state variables" from which we can calculate every power variable in the system. In addition to easing the reduction of higher order systems, the numerical solution of the system formulated using the state-variable method frees us from the restriction of using linear elemental equations imposed by our analytical method.

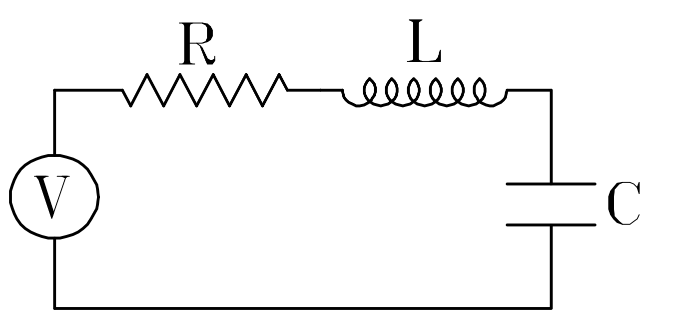
State Variables

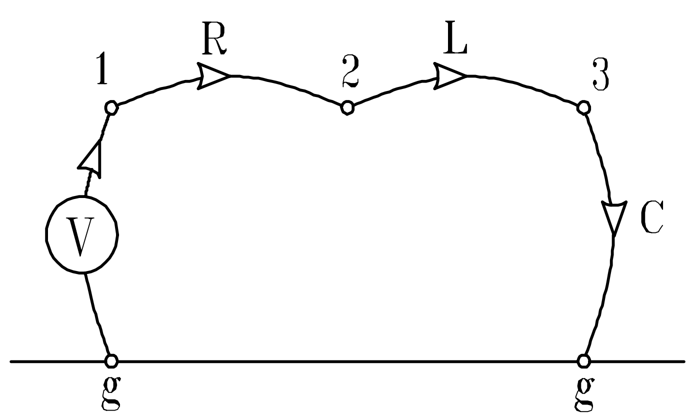
One meaning of the word "state" is "condition". A physician describes the "state" of a patient's health by his or her "vital statistics" which are variables, such as temperature, blood pressure, and white blood cell count. One could refer to the "vital statistics" as "state variables" because these are the variables needed to determine the state of a person's health. The word "state" also means "condition" when used in systems dynamics. The "condition" of condition of a dynamic system is value of the power variables, the pressure in a fluid capacitor in a hydraulic system is analogous to blood pressure in a person.

We calculate the initial values of power variables in first order systems using the values of the input variable and the energy storage variable at time $t = 0^+$. We know the value of the input variable over time because we control the input source. We also know that $\mathbf{E}(0^-) = \mathbf{E}(0^+)$, which allows us to calculate the energy storage variable at time $t = 0^+$ if we know it at $t = 0^-$. If we can use the input and the energy storage variable to calculate the any other power variable in the system at time $t = 0^+$, then, if we know the value of the energy storage variable over time, we can use the same logic to calculate any power variable in any element at anytime. Consequently, **THE ENERGY STORAGE VARIABLES ARE STATE VARIABLES** for the dynamic system. The define the state of the system because if we know these variables as functions of time we can solve for any other power variable for any time.

Second Order Dynamic System Example: RLC Circuit.

There are two unknown power variables in every dynamic element, except sources which have only one unknown power variable since we control the other. The unknown power variables in the RLC circuit shown below are: i, v_{12}, i_R, v_{23}, i_L, v_{3g}, and i_C. There are two independent energy storages in this system, the capacitor and the inductor. The state variables are the energy storage variables, v_{3g} and i_L. We will first formulate the state equations to find state variables v_{3g} and i_L. We will then formulate the output equations to calculate all of the remaining unknowing power variables using the input v, and the state variables v_{3g} and i_L.





Equation List

Compatibility

$v(t) = v_{12} + v_{23} + v_{3g}$

Continuity

Node 1: $i - i_R = 0$

Node 2: $i_R - i_L = 0$

Node 3: $i_L - i_C = 0$

Elemental

Resistor: $v_{12} = Ri_R$

Inductor: $v_{23} = L\frac{di_L}{dt}$

Capacitor: $i_C = C\frac{dv_{3g}}{dt}$

Energy

System: $\mathbf{E} = \mathbf{E}_C + \mathbf{E}_L$

Capacitor: $\mathbf{E}_C = \frac{1}{2}Cv_{3g}^2$

Inductor: $\mathbf{E}_L = \frac{1}{2}Li_L^2$

State Equations

We will now perform a partial reduction of the equation list to a set of two first order differential equations written in terms of the input variable, the state variables, the elemental parameters and time. These first order differential equations are called "state equations".

The reduction method is always the same:

1. **ALWAYS START A STATE EQUATION REDUCTION WITH THE ELEMENTAL EQUATION OF AN ENERGY STORAGE ELEMENT.** You will have as many state equations as there are independent energy storages in the system.

2. Rearrange the energy storage elemental equation to place the derivative of the state variable on the left side by itself.

3. Proceed to eliminate all power variables except for the input variable and the state variables.

IMPORTANT: ELIMINATE THE OUTPUT VARIABLES UNLESS THEY ARE A STATE VARIABLE. We will calculate the output variable after calculating the state variables.

It is always a good idea to list the input and state variables before beginning the reduction so you don't loose track of the variable you wish to keep.

STATE-EQUATION REDUCTION

Input Variable: v State Variables: i_L, v_{3g}

Inductor Elemental Eq.:

$$v_{23} = L\frac{di_L}{dt}$$

Rearrange to put the derivative of the state variable i_L on the left side.

$$\frac{di_L}{dt} = \frac{v_{23}}{L}$$

Eliminate v_{23} because it is neither the input nor a state variable.

$$\frac{di_L}{dt} = \frac{1}{L}\left(v - v_{12} - v_{3g}\right)$$

Eliminate v_{12} because it is neither the input nor a state variable.

$$\frac{di_L}{dt} = \frac{1}{L}\left(v - Ri_R - v_{3g}\right)$$

Eliminate i_R because it is neither the input nor a state variable.

$$\frac{di_L}{dt} = \frac{1}{L}\left(v - Ri_L - v_{3g}\right)$$

This is a state equation; v is the input, v_{3g} and i_L are state variables.

Capacitor Elemental Eq.:

$$i_C = C\frac{dv_{3g}}{dt}$$

Rearrange to put the derivative of state variable v_{3g} on the left side.

$$\frac{dv_{3g}}{dt} = \frac{i_C}{C}$$

Eliminate i_C since it is neither the input nor a state variable.

$$\frac{dv_{3g}}{dt} = \frac{i_L}{C}$$

This is a state equation; v_{3g} and i_L are state variables.

List the state equations, rearranging the right side so that the terms with the state variables are first and in the same order, if present, in each equation, followed by the term with the input variable. This is the form we will use for the vector-matrix notation.

Summary of State Equations:

$$\frac{di_L}{dt} = -\frac{R}{L} i_L - \frac{1}{L} v_{3g} + \frac{1}{L} v$$

$$\frac{dv_{3g}}{dt} = \frac{1}{C} i_L$$

This is a set of simultaneous first order differential equations. It is a set of simultaneous differential equations because, since both equations share the state variables, they are coupled to one another and both must be satisfied if either is to be true. We will use a common numerical method for solving differential equations available in Mathcad as a function to solve this set of equations for the state variables as functions of time.

Output Equations:

If we know the source v and the state variables v_{3g} and i_L as functions of time, then we can calculate every unknown power variable as functions of time. Referring to the equation list, use the same logic used to derive initial conditions, except use i_L rather than $i_L(0+)$ and v_{3g} rather than $v_{3g}(0+)$ since $i_L(t)$ and $v_{3g}(t)$ are determined (by Mathcad) using the state equations before the output equations are evaluated. Write the output equations using the state variables and input variable in the same order on the right side as in the state equations.

The output equations for i_C and i are simple since all of the currents are equal to the current through the inductor, which is a state variable.

$$i_C = i_L$$

$$i_R = i_L$$

$$i = i_L$$

The output equation for the voltage drop across the resistor requires a substitution.

$$v_{12} = Ri_R$$

therefore

$$v_{12} = Ri_L$$

The output equation for the voltage drop across the inductor requires two substitutions and rearranging the state variable and the input variable terms:

$$v = v_{12} + v_{23} + v_{3g}$$

$$v_{23} = v - Ri_L - v_{3g}$$

Rearrange as:

$$v_{23} = -Ri_L - v_{3g} + v$$

Summary of the Output Equations for the unknown power variables:

$$i_C = i_L$$

$$i_R = i_L$$

$$i = i_L$$

$$v_{12} = Ri_L$$

$$v_{23} = -Ri_L - v_{3g} + v$$

State-Space:

"State-Space" is a very impressive term for a simple concept. Mathematically, a space is defined by the coordinates axes that allow you to plot a vector in that space. All of the planes and spaces you have worked with in engineering are abstractions, with the exception of purely geometric spaces used to draw parts. You used force spaces in statics to perform equilibrium analysis. The coordinates of a force space are the force components F_X, F_Y, and F_Z. Although a force space is an abstraction, it is a comfortable one because it is easy to visualize a force vector in a three dimensional geometric space since the force axes can be aligned with the position axes.

A two dimensional geometric "space", i.e. a plane, is formed by defining two coordinate axes, a three dimensional space by defining three coordinate axes, a four dimensional space by defining four coordinate axes, etc. Although we can not visualize spaces above three dimensions and call such spaces "hyperspace", which means "beyond space", hyperspace follows the same mathematical rules as does two and three dimensional space. Mathematically, you can define a space to have as many dimensions as you want. The only restriction is that the axes must be independent. Axes are independent if you can not plot a point on one axis using the other axes, except at the origin. In the most fundamental sense, a space is defined by its coordinate coordinates. If we define coordinates, then we define a space. The axes of the space do not need to have the same units. We often use two dimensional spaces with time or position as the horizontal axis and a different physical quantity as the vertical axis.

A "state-space" is a space defined by coordinates which are the state variables of a system. It is not a "real" space in a geometric sense, but neither is a force space. The energy storage variables satisfy the mathematical requirement for axes because they are independent. The values of the state variables at any moment in time are the coordinates of a point (a vector from the origin) in the state-space of that dynamic system. We can, and do, define hyper state-spaces. A dynamic system with four independent energy storage elements has four state variables. Consequently, its state-space is four dimensional because its state-vector has four components. Is it easier to work with three dimensional state-spaces than four dimensional state-spaces? Not really. It makes little difference how many dimensions there are in the state-space in practice because our visualization will be two dimensional plots of individual state variables or output variables vs. time. Our major concern with higher order systems will be with the accuracy of our model, which will degrade with increased complexity.

Vector-Matrix Representation of the State Equations.

We will use vectors and matrices to express the state equations and the output equations in the simple and compact form of Linear Algebra, which was developed to handle systems of simultaneous equations. The term "algebra" means a set of rules to manipulate mathematical symbols. There are many different types of algebra. Linear algebra is the set of rules that defines the allowable operations on symbols that represent systems of simultaneous equations. The easiest way to understand the manipulations of linear algebra is to work with a system of equations consisting of two equations with two unknowns. The rules that apply to this system of equations apply to any system of equations. Consider the system of equations:

$$\begin{aligned} y_1 &= a_{11}x_1 + a_{12}x_2 \\ y_2 &= a_{21}x_1 + a_{22}x_2 \end{aligned} \qquad (1)$$

where x_i and y_i are variables and a_{ii} are constants. This system of equation can be written in vector-matrix notation as:

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \qquad (2)$$

where $\begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$ and $\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ are column vectors and $\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$ is a matrix. The right side of this vector-matrix equation is a multiplication which follows the rule: **MULTIPLY ROW TIMES COLUMN, element by element and sum the product.** Example:

$$y_1 = \begin{bmatrix} a_{11} & a_{12} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = a_{11}x_1 + a_{12}x_2$$

Eq. (1) and (2) can be written more compactly by defining

$$\mathbf{y} \equiv \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$$
$$\mathbf{A} \equiv \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$
$$\mathbf{x} \equiv \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

where bold is used to denote a vector or matrix. Conventionally, vectors are denoted by lower case and matrices by upper case, where possible. This leads to:

$$\mathbf{y} = \mathbf{Ax} \qquad (3)$$

Eqs. (1), (2), and (3) are increasing abstract representations of the same equation. Eq. (3) is understood by expanding backwards to the less abstract form of Eq. (2) and finally to Eq. (1).

The product of a vector or matrix and scalar (or a single valued function) does not have the element by element summation of matrix times matrix multiplication, e. g.

If

$$\mathbf{B} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$$

and u is a scalar then

$$\mathbf{B}u = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} u = \begin{bmatrix} b_1 u \\ b_2 u \end{bmatrix}.$$

Consequently, the system of equations

$$y_1 = a_{11}x_1 + a_{12}x_2 + b_1 u$$
$$y_2 = a_{21}x_1 + a_{22}x_2 + b_2 u$$

can be written as

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} u$$

or

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{B}u$$

Similarly, distributive operators, such as differentiation can be treated as scalar-matrix multiplication, e.g.

$$\frac{d}{dt}\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} \frac{dy_1}{dt} \\ \frac{dy_2}{dt} \end{bmatrix}$$

We can now write the state equations and the output equations in vector-matrix form.

The general vector-matrix form of the state equations is:

$$\frac{d}{dt}\mathbf{x} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$$

where **x** is the vector of state variables and **u** is the vector of inputs. In the above example, we have just one input, so u would be a single function. The state equations for the RLC system are:

$$\frac{di_L}{dt} = -\frac{R}{L} i_L - \frac{1}{L} v_{3g} + \frac{1}{L} v$$
$$\frac{dv_{3g}}{dt} = \frac{1}{C} i_L$$

which can be written as:

$$\frac{d}{dt}\begin{bmatrix} i_L \\ v_{3g} \end{bmatrix} = \begin{bmatrix} -\frac{R}{L} & -\frac{1}{L} \\ \frac{1}{C} & 0 \end{bmatrix} \begin{bmatrix} i_L \\ v_{3g} \end{bmatrix} + \begin{bmatrix} \frac{1}{L} \\ 0 \end{bmatrix} v$$

Vector-Matrix Form of the Output Equations

The general vector-matrix form of the output equations is:

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

where **y** is the vector of output variables, **x** is the vector of state variables and **u** is the vector of inputs. The output equations are:

$$i_C = i_L$$

$$i_R = i_L$$

$$i = i_L$$

$$v_{12} = Ri_L$$

$$v_{23} = -Ri_L - v_{3g} + v$$

which can be in written vector-matrix form as:

$$\begin{bmatrix} i_C \\ i_R \\ i \\ v_{12} \\ v_{23} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ R & 0 \\ -R & -1 \end{bmatrix} \begin{bmatrix} i_L \\ v_{3g} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} v$$

Sequence of the Numerical Solution

We have two very different vector-matrix equations for the state equations and the output equations:

$$\frac{d}{dt}\mathbf{x} = \mathbf{Ax} + \mathbf{Bu} \qquad \text{State Equations}$$

$$\mathbf{y} = \mathbf{Cx} + \mathbf{Du} \qquad \text{Output Equations}$$

The state equations are coupled simultaneous first order differential equations. They are first order differential equations since the vector **x** of state variables is differentiated with respect to time. They are coupled simultaneous equations because they represent the response of a physical system in which the state variables and the input determine the future state of the system. All of the equations must be simultaneously true to describe the system. Omitting a state equation, and the corresponding state variable, from the set of state equations is similiar to omitting physical elements from the system. The output equations are algebraic equations, not differential equations. They are also simultaneous equation, they must all be true simultaneously, but they are not coupled equations in the sense that we are free to omitt any output equation we are not interested in. In fact, we do not need output equations at all if the only variables in the system we are interested in are state variables.

The sequence of the solution is to solve the state equations first and then use the results to solve the output equations. The state equations are solved numerically for the state variable vector **x** using the Runge-Kutta method in either Mathematica or Mathcad. Mathematica's numerical solution of the state-equations produces "interpolating functions". Mathcad's numerical solution produces an output matrix, the columns of which are the values of the state variables for each time step in the solution. In either case, the numerical solution of the state equations produces the state variable's history over the duration of the solution. This information is then used to evaluate the output variables. When analyzing mechanical systems, we are often interested in either displacement or acceleration, not our state variable velocity, and either integrate or differentiate numerically.

Problem Set No. 6

Due Friday, March 9, 2001 (if your here since it will be a comp. Day)
or Thursday March 8, 2001 (if you will be leaving for Spring Break)

Reading

Rowell and Wormley: pp. 120-126. Skim the remainder of Chapter 5 which presents an algorithm for reducing a linear graph to a set of state equations. The algorithm can has been used in dynamic system analysis software. However, we do not need it to produce a set of state equations using a manual reduction of the equation list.

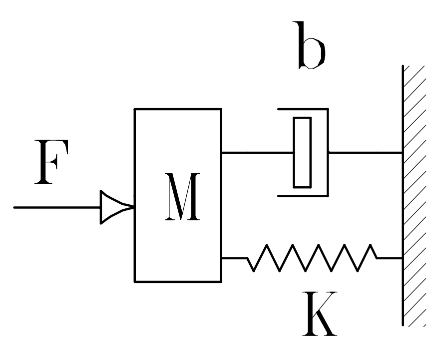
Rowell and Wormley Appendix A: Introduction to Matrix Algebra.

Problems

Problem 1: A translational mechanical system consisting of a spring, a mass, and a damper acted upon by an applied force is shown in the schematic below.

(a) Derive the second order system equation for the velocity of the mass. Note: You can not solve this equations until you are given the forcing function F(t) and the initial conditions, so do not try to.

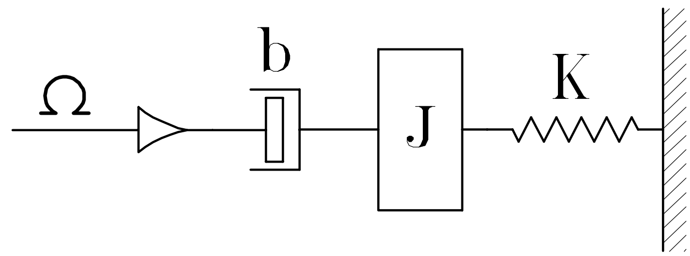
(b) Derive the state equations for this system with the output variables the state variables and the force in the damper. Express the state equations and the output equations in vector-matrix form.



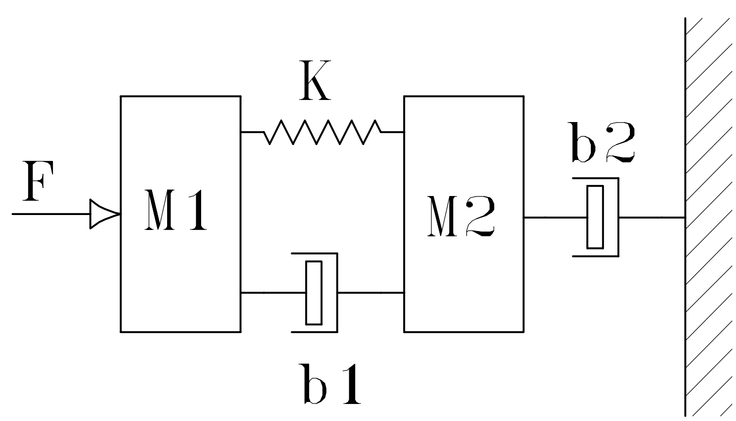
Problem 2: A rotational mechanical system consisting of a spring, an inertia, and a damper acted upon by an applied torque is shown in the schematic below.

(a) Derive the second order system equation for the torque in the spring.

(b) Derive the state equations for this system. Derive the output equations for the angular velocity of the inertia, the torque accelerating the inertia, and the torques in the spring and the damper. Express the state equations and the output equations in vector-matrix form.



Problem 3: A translational mechanical system consisting of a spring, two masses, and two dampers acted upon by an applied force is shown in the schematic below. Derive the state equations for this system. Derive the output equations for the velocity of the masses, the force in the spring, and the forces in the dampers. Express the state equations and the output equations in vector-matrix form.



Problem 4: A dynamic system shown schematically below. A torque source is connected to a rigid shaft which drives a gear pump. The gear pump delivers volume *Vp* for each 2π rad rotation of the shaft. A pipe connects the pump to a valve, with fluid resistance R. The valve is connected to a fluid capacitor, with capacitance C, and to a hydraulic cylinder. The area of the piston in the cylinder is A. A rigid rod connects the piston to mass M. There is negligible energy stored or dissipated in the pump, the pipe, the hydraulic cylinder, and the rod connecting the piston to the mass.

Derive the state equations for this system. Derive the output equations for the pressure in the capacitor, the velocity of the mass, the force applied to the mass, and the flow rate from the pump. Express the state equations and the output equations in vector-matrix form.