

## Mathematical Methods (10/24.539)

### IV. General Initial Value Problems (IVPs)

#### Introduction

Engineering systems are described in mathematical terms via balance equations (energy balances, mass balances, force balances, etc.) and these conservation relationships invariably lead to differential equations. For systems with only one independent variable, we only have to work with ordinary differential equations (as reviewed in Sections I and II). However, we often have several dependent variables of interest, and these usually give rise to a set of coupled ODEs. The matrix notation and fundamental matrix operations reviewed in Section III give us the tools necessary to deal with a coupled system of differential equations in an efficient manner (the matrix equations can consist of either algebraic or differential equations).

In this section we focus on a particular class of problems known as Initial Value Problems (IVPs) which occur frequently in practical applications. This subject was included indirectly as part of our discussions in Sections I and II, but here we emphasize how to work with coupled systems of first order ordinary differential equations. We can also solve IVPs written in the form of high order ODEs by simply converting the  $n^{\text{th}}$  order system into  $n$  coupled first order equations. This approach gives us a common basis for solving all IVPs with only one independent variable (PDEs which deal with two or more independent variables are discussed in later sections).

IVPs that are linear and have constant coefficients can be solved analytically. This is an important class of problems and applications from this set generally have relatively straightforward solution schemes that are based on the matrix eigenvalue and eigenvector concepts reviewed in Section III. However, many real engineering systems of interest have variable coefficients or nonlinear interactions and, in general, these are not readily amenable to analytical techniques. Thus, this section of notes also emphasizes general numerical schemes that can be applied in most situations that arise. The full treatment of both analytical and numerical solution strategies in tandem should give the student all the necessary tools to handle just about any situation that would arise in practice. Detailed illustrative examples of both the analytical and numerical solution methods are also given to help develop confidence in this area. A good understanding of this subject is essential to engineers and scientists involved in the modeling and simulation of physical systems.

Unlike the first three sections that are included here primarily for review by the individual student, as needed, this section will be treated in some detail in this course. Much of this material is not usually covered in most undergraduate courses. As such, we will spend a little more time and work out a few more examples in full detail to make sure you get a good handle on this material.

This section of notes overviews the basic concepts, analytical and numerical solution techniques, and the accompanying illustrative IVP examples within several subsections, as follows:

**General Notation for Coupled First Order ODEs**

**Conversion of  $n^{\text{th}}$  Order Systems**

**Homogeneous ODEs with Constant Coefficients**

### Non-Homogeneous ODEs with Constant Coefficients

- Method of Undetermined Coefficients
- Matrix Diagonalization
- Variation of Parameters

#### Examples of Analytical Solutions

- **Example 4.1** - Solve  $\underline{y}' = \underline{A}\underline{y}$  with  $\underline{A} = \begin{bmatrix} 0 & 4 \\ 4 & 0 \end{bmatrix}$
- **Example 4.2** - Solve  $\underline{y}' = \underline{A}\underline{y}$  with  $\underline{A} = \begin{bmatrix} 4 & 1 \\ -1 & 2 \end{bmatrix}$
- **Example 4.3** - A Simple Mechanical System with Nonzero  $F(t)$ 
  - **Example 4.3A** - Solution via Undetermined Coefficients
  - **Example 4.3B** - Solution via Matrix Diagonalization
  - **Example 4.3C** - Solution via Variation of Parameters

### Numerical Solution to General IVPs

#### Examples of Numerical Solutions

- **Example 4.4** - Revisit Example 4.3 from above using Matlab
- **Example 4.5** - Semi-Batch Chemical Reactor Dynamics

**Note:** The general subject area of Initial Value Problems (IVPs) is also treated in detail in my **System Dynamics** (24.509) course. A lot of material for this course is available on my website at [www.profjrwhite.com/courses.htm](http://www.profjrwhite.com/courses.htm).

## General Notation for Coupled First Order ODEs

### Overview

Systems of first order ODEs occur frequently in applications, especially in connection with the study of dynamic systems. One can also transform an  $n^{\text{th}}$  order ODE into a coupled set of  $n$  first order ODEs. The resultant systems often fall into the class of problems known as Initial Value Problems (IVPs). This is a very important class of problems, and this section will treat both analytical and numerical solution techniques for general IVPs. The ***State Space Representation*** developed here allows a very general treatment of this subject. The student is referred to a very detailed overview of System Dynamics (24.509) by Prof. J. R. White for a more comprehensive treatment of this subject (with lots of good stuff available at [www.profjrwhite.com/courses.htm](http://www.profjrwhite.com/courses.htm)).

### General Notation

A series of  $n$  1<sup>st</sup> order ODEs can be written as

$$y_1' = f_1(t, y_1, y_2, \dots, y_n)$$

$$y_2' = f_2(t, y_1, y_2, \dots, y_n)$$

or for the  $i^{\text{th}}$  equation, we have

$$y_i' = f_i(t, y_1, y_2, \dots, y_n) \quad (4.1)$$

This sequence of equations can be written in vector-matrix form as,

$$\underline{y}' = \underline{f}(t, \underline{y}) \quad (4.2)$$

where the single underline notation indicates a column vector (of length  $n$ , in this case).

### Linear Systems

If the functions,  $f_i(t)$ , all happen to be linear, then the above general notation reduces to

$$y_1' = a_{11}y_1 + a_{12}y_2 + \dots + a_{1n}y_n + g_1$$

$$y_2' = a_{21}y_1 + a_{22}y_2 + \dots + a_{2n}y_n + g_2$$

and for the  $i^{\text{th}}$  equation, we have

$$y_i' = a_{i1}y_1 + a_{i2}y_2 + \dots + a_{in}y_n + g_i = \sum_j^n a_{ij}y_j + g_i \quad (4.3)$$

Also, in vector-matrix form, this becomes

$$\underline{y}' = \underline{\underline{A}}\underline{y} + \underline{g} \quad (4.4)$$

This notation is referred to as the standard State Space Representation, where  $\underline{\underline{A}}$  is the state matrix (notice that a double underline is used to represent a 2-D array),  $\underline{y}$  is the state vector, and  $\underline{g}$  is the driving or forcing function vector.

The homogeneous form of this equation simply sets the forcing function to zero, giving

$$\underline{y}' = \underline{A}\underline{y} \quad (4.5)$$

Because the system is linear and homogeneous, linear superposition of the individual solutions is used to construct the homogenous solution, or

$$\underline{y}_h = c_1 \underline{y}_1 + c_2 \underline{y}_2 + \cdots + c_n \underline{y}_n = \sum_{i=1}^n c_i \underline{y}_i \quad (4.6)$$

$$\underline{y}_h = \begin{bmatrix} \underline{y}_1 & \underline{y}_2 & \cdots & \underline{y}_n \end{bmatrix} \underline{c} \quad (4.7)$$

$$\underline{y}_h = \begin{bmatrix} y_{11} & y_{12} & \cdots & y_{1n} \\ y_{21} & y_{22} & \cdots & y_{2n} \\ \vdots & \vdots & \cdots & \vdots \\ y_{n1} & y_{n2} & \cdots & y_{nn} \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix} = \underline{Y}\underline{c} \quad (4.8)$$

where the various forms given here are used interchangeably. For nontrivial solutions, we know that the Wronskian, which is given by the determinant of the  $\underline{Y}$  matrix, must be nonzero, or

$$W = \det \begin{bmatrix} \underline{y}_1 & \underline{y}_2 & \cdots & \underline{y}_n \end{bmatrix} = \begin{vmatrix} \underline{y}_1 & \underline{y}_2 & \cdots & \underline{y}_n \end{vmatrix} \neq 0 \quad (4.9)$$

This says that the  $n$  solutions,  $\underline{y}_1, \underline{y}_2, \cdots, \underline{y}_n$ , are linearly independent, and that the  $n$  linearly independent solutions form a basis of solutions on the interval of interest (one should note here that all this discussion is very similar to that given in Section II on Linear ODEs).

Finally, as in our previous discussions, the solution to the non-homogeneous equation is simply the sum of the homogeneous and particular solutions, or

$$\underline{y} = \underline{y}_h + \underline{y}_p \quad (4.10)$$

with the unique solution being determined by application of a set of  $n$  initial conditions that uniquely define the  $n$  coefficients in the homogeneous solution.

### Summary

This section of Math Methods for Engineers (10.539 and 24.539) gives an introduction to both analytical and numerical solution techniques for general IVPs. The analytical solutions are restricted to linear constant coefficient systems, and they address both forced and unforced situations. The numerical solutions are more appropriate for general problems (variable coefficient and/or nonlinear terms) and we will see that a simple extension of our previous discussion of the solution techniques for a single 1<sup>st</sup> order ODE allows the use of the same ODE solution algorithms in Matlab for coupled systems of equations. A series of analytical and numerical examples illustrates how to use these techniques in actual applications.

### Conversion of an $n^{\text{th}}$ Order System into $n$ $1^{\text{st}}$ Order Equations

#### Development of the Method

Consider an  $n^{\text{th}}$  order system written as

$$y^{(n)} + p_{n-1}y^{(n-1)} + p_{n-2}y^{(n-2)} + \cdots + p_1y' + p_0y = r \quad (4.11)$$

Let's make the substitutions  $z_1 = y$ ,  $z_2 = y'$ , etc., up to  $z_n = y^{(n-1)}$ . Then,

$$z_i' = z_{i+1} \quad \text{for } i < n \quad (4.12)$$

and, from eqn. (4.11), we have

$$z_n' = -p_{n-1}z_n - p_{n-2}z_{n-1} - \cdots - p_1z_2 - p_0z_1 + r \quad (4.13)$$

These coupled equations can be put into matrix form,

$$\underline{z}' = \frac{d}{dx} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ \vdots \\ z_n \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ & & \vdots & \ddots & \\ -p_0 & -p_1 & -p_2 & \cdots & -p_{n-1} \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ \vdots \\ z_n \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix} r \quad (4.14)$$

The original  $n^{\text{th}}$  order ODE is now in standard form, where we have  $n$  coupled  $1^{\text{st}}$  order equations.

#### An Example

As an example of using this procedure, consider the following  $3^{\text{rd}}$  order system,

$$y''' + 7xy'' + 3y' - 2x^2y = \sin x - e^x$$

To convert this to state form, let  $z_1 = y$ ,  $z_2 = y' = z_1'$ , and  $z_3 = y'' = z_2'$ . Therefore,

$$z_1' = z_2, \quad z_2' = z_3 \quad \text{and} \quad z_3' = -7xz_3 - 3z_2 + 2x^2z_1 + \sin x - e^x$$

or, in matrix form, this becomes

$$\frac{d}{dx} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 2x^2 & -3 & -7x \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} (\sin x - e^{-x})$$

Therefore,  $\underline{z}' = \underline{A}\underline{z} + \underline{g}$ , as desired.

**Note:** The process illustrated above is valid for both linear and non-linear systems. Linear systems were assumed here since we will focus on linear systems for the subsequent discussion of analytical solution techniques. For linear systems,  $\underline{A} \neq f(\underline{z})$ . For nonlinear systems,  $\underline{A} = f(\underline{z})$  and the system in state form is usually written simply as  $\underline{z}' = \underline{f}(x, \underline{z})$  [as in eqn. (4.2)].

## Homogeneous Linear Systems with Constant Coefficients

### Overview

A homogeneous system of  $n$  linear 1<sup>st</sup> order equations can be written as

$$\underline{y}' = \underline{A}\underline{y} \quad (4.15)$$

Let's assume that  $t$  is the independent variable, giving  $\underline{y} = \underline{y}(t)$ , and that  $\underline{A}$  is a constant coefficient matrix. Now, we recall that the solution for the *scalar case* was obtained quite simply by assuming a solution of exponential form.

Using a similar approach for the *matrix case*, we try

$$\underline{y}(t) = \underline{x}e^{\lambda t} \quad (4.16)$$

as the solution to the system of  $n$  coupled equations, where  $\underline{x}$  is a constant vector of length  $n$ .

Upon substitution into the original equation, this becomes

$$\underline{y}' = \underline{x}\lambda e^{\lambda t} = \underline{A}\underline{y} = \underline{A}\underline{x}e^{\lambda t}$$

or

$$\underline{A}\underline{x} = \lambda\underline{x} \quad (4.17)$$

which is referred to as the characteristic equation. But this is just a standard eigenvalue problem, where  $|\underline{A} - \lambda\underline{I}| = 0$  gives  $n$  eigenvalues and  $(\underline{A} - \lambda_i\underline{I})\underline{x}_i = 0$  gives an eigenvector,  $\underline{x}_i$ , for each distinct eigenvalue,  $\lambda_i$ .

### Distinct Eigenvalues

If the eigenvalues are distinct, then the eigenvectors are linearly independent, and the  $n$  individual solutions,  $\underline{y}_i$ , form a basis in  $n$ -dimensional space, and the full homogeneous solution can be written as a linear combination of the basis solutions, or

$$\underline{y}(t) = \sum_i c_i \underline{y}_i(t) = \sum_i c_i \underline{x}_i e^{\lambda_i t} \quad (4.18)$$

or

$$\underline{y}(t) = \sum_i \underline{x}_i z_i(t) \quad \text{where} \quad z_i(t) = c_i e^{\lambda_i t} \quad (4.19)$$

and

$$\underline{y} = \begin{bmatrix} \underline{x}_1 & \underline{x}_2 & \cdots & \underline{x}_n \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_n \end{bmatrix} = \underline{M}\underline{z} \quad (4.20)$$

where  $\underline{M}$  is referred to as the *modal matrix* and is given by

$$\underline{\underline{M}} = [\underline{x}_1 \quad \underline{x}_2 \quad \cdots \quad \underline{x}_n] \quad (4.21)$$

Thus, knowledge of the distinct eigenvalues and eigenvectors of the state matrix allows one to write the analytical solution,  $\underline{y} = \underline{\underline{M}}\underline{z}$ , to the original system of first order linear homogeneous constant coefficient ODEs. Example 4.1 gives an illustration of how to use the above procedure for the case of distinct roots.

### An Alternate Development for Distinct Eigenvalues

An alternate approach to finding the solution to the homogeneous equation is to assume a solution of the form

$$\underline{y}(t) = \underline{\underline{M}}\underline{z}(t) \quad (4.22)$$

where  $\underline{\underline{M}}$  is the modal matrix (for distinct eigenvalues). Upon substitution into the original equation, this gives

$$\underline{\underline{M}}\underline{z}' = \underline{\underline{A}}\underline{\underline{M}}\underline{z}$$

Premultiplying by  $\underline{\underline{M}}^{-1}$ , we have

$$\underline{z}' = \underline{\underline{M}}^{-1}\underline{\underline{A}}\underline{\underline{M}}\underline{z} = \underline{\underline{D}}\underline{z} \quad (4.23)$$

where we recognize that a similarity transformation (i.e. the  $\underline{\underline{M}}^{-1}\underline{\underline{A}}\underline{\underline{M}}$  combination) with the modal matrix gives a diagonal matrix,  $\underline{\underline{D}}$ , whose elements are simply the  $n$  distinct eigenvalues of the state matrix. Since the similarity transformation guarantees identical eigenvalues, we have an uncoupled system of equations

$$\underline{z}' = \underline{\underline{D}}\underline{z} \quad \text{which implies} \quad z_i' = \lambda_i z_i \quad (4.24)$$

and this has the solution

$$z_i(t) = c_i e^{\lambda_i t} \quad (4.25)$$

Note that both developments give the same result, as expected [see eqn. (4.19)]!

### Repeated Eigenvalues

When particular eigenvalues occur with multiplicity  $m > 1$ , we cannot generate a complete basis from the eigenvectors of the state matrix  $\underline{\underline{A}}$ . In this case, we seek a set of linearly independent basis vectors as follows:

#### *Case 1: The Obvious Choice*

From our experience with the Method of Undetermined Coefficients for linear constant coefficient ODEs, if repeated roots occur, a reasonable choice for a second linearly independent solution is to simply multiply the known solution,  $y_1(t)$ , by the independent variable,  $t$ , to obtain  $y_2(t) = ty_1(t)$ . Applying this approach to the present matrix ODE situation gives the following result.

Start with a single solution where  $\mu$  is a repeated root (of multiplicity 2), or

$$\underline{y}_1 = \underline{x}e^{\mu t} \quad \text{with} \quad \lambda_{1,2} = \mu \quad (4.26)$$

and  $\underline{x}$  is the eigenvector associated with  $\mu$ . Now choose  $y_2$  as

$$\underline{y}_2 = \underline{x}te^{\mu t} \quad (4.27)$$

which gives

$$\underline{y}_2' = \mu \underline{x}te^{\mu t} + \underline{x}e^{\mu t} \quad (4.28)$$

From the original differential equation, we also have

$$\underline{y}_2' = \underline{A}\underline{y}_2 = \underline{A}\underline{x}te^{\mu t} = \mu \underline{x}te^{\mu t} \quad (4.29)$$

Therefore, equating eqns. (4.28) and (4.29) gives  $\underline{x}e^{\mu t} = 0$ , which is not possible! Therefore, the choice of  $\underline{y}_2$  is incorrect (which is not uncommon with the Method of Undetermined Coefficients)! If this happens, we simply need to choose another form for  $\underline{y}_2(t)$ .

### Case 2: A Better Choice

Since our first choice for a second linearly independent solution did not work, let's try a slightly different form for  $\underline{y}_2(t)$  that includes the above guess plus another term containing an unknown vector constant. In particular, given

$$\underline{y}_1 = \underline{x}e^{\mu t} \quad \text{with} \quad \lambda_{1,2} = \mu \quad (4.30)$$

with  $\underline{x}$  being the eigenvector associated with  $\mu$ , let's try

$$\underline{y}_2 = (\alpha \underline{x}t + \underline{z})e^{\mu t} \quad (4.31)$$

where both  $\alpha$  and  $\underline{z}$  are unknown coefficients that need to be determined.

**Note:** For homogeneous equations  $\alpha$  is set to unity, but for non-homogeneous equations it must be included for generality. For consistency, we will carry it along in both developments.

Now, taking the derivative of  $\underline{y}_2(t)$  gives

$$\underline{y}_2' = \alpha \mu \underline{x}te^{\mu t} + \mu \underline{z}e^{\mu t} + \alpha \underline{x}e^{\mu t} \quad (4.32)$$

Also from the defining differential equation, we have

$$\underline{y}_2' = \underline{A}\underline{y}_2 = \underline{A}(\alpha \underline{x}t + \underline{z})e^{\mu t} = \alpha \mu \underline{x}te^{\mu t} + \underline{A}\underline{z}e^{\mu t} \quad (4.33)$$

Therefore, equating eqns. (4.32) and (4.33) gives

$$\underline{A}\underline{z} = \alpha \underline{x} + \mu \underline{z} \quad \text{or} \quad (\underline{A} - \mu \underline{I})\underline{z} = \alpha \underline{x} \quad (4.34)$$

This last equation is similar in form to an eigenvalue problem, but here both  $\mu$  and  $\underline{x}$  are known (these are eigenvalues and eigenvectors of the original state matrix,  $\underline{A}$ ). Also, for homogeneous equations,  $\alpha$  is really arbitrary (usually set to unity), so that the only unknown here is the vector,



$\underline{z}$ . Thus, solution of this matrix equation for  $\underline{z}$  gives the second desired linearly independent solution,  $\underline{y}_2$ , associated with repeated eigenvalue  $\mu$ . Example 4.2 gives an illustration of how to use the above procedure for the case of repeated roots.

## Non-Homogeneous ODEs with Constant Coefficients

### Overview

The general solution to a non-homogeneous linear system of 1<sup>st</sup> order ODEs is determined from a combination of the homogeneous and particular solutions. We have just addressed various analytical solution techniques for linear constant coefficient homogeneous systems, and saw that knowledge of the eigenvalues and eigenvectors of the state matrix is needed to construct these solutions. We now address three techniques for obtaining the particular solution, thus completing all the elements needed for the general solution. In particular, the *three methods* are listed here and described in more detail below:

**Undetermined Coefficients** - used with constant coefficient systems with simple forcing functions.

**Matrix Diagonalization** - used with constant coefficient systems with arbitrary forcing functions. The eigenvalues of the state matrix should be distinct.

**Variation of Parameters** - can be used for variable coefficient systems and general forcing functions. However, because of the complexity, the method is only used routinely for constant coefficient systems.

### Undetermined Coefficients

The method of undetermined coefficients is rather straightforward but it is generally useful only for constant coefficient linear systems. In addition, the forcing function vector,  $\underline{g}(t)$ , must be a relatively simple function (exponential, polynomial, sinusoid, or a linear combination of such functions).

**General rule:** Choose  $\underline{y}_p(t)$  to have the same form as  $\underline{g}(t)$  *and* all its derivatives. Evaluate the unknown coefficient vectors within  $\underline{y}_p(t)$  by substitution into the original system of inhomogeneous ODEs, and simply equate the coefficient vectors of the like terms on both sides of the equations.

**Special rule:** If  $\underline{g}(t)$  contains a term that includes an eigenvalue of  $\underline{A}$ , then modify  $\underline{y}_p$  to be

$$\underline{y}_p(t) = (\alpha \underline{x}t + \underline{z})e^{\lambda t} \quad (4.35)$$

where  $\underline{x}$  is the eigenvector corresponding to  $\lambda$  (with some unknown normalization).

Example 4.3A provides an example of the Method of Undetermined Coefficients.

### Matrix Diagonalization

In our discussion of homogeneous solutions, we saw that, for the case of *distinct eigenvalues*, the homogeneous solution could be written as  $\underline{y} = \underline{M}\underline{z}$ , where  $\underline{M} = [\underline{x}_1 \quad \underline{x}_2 \quad \cdots \quad \underline{x}_n]$  was identified as the *modal matrix*. Now, for the inhomogeneous problem,

$$\underline{y}' = \underline{A}\underline{y} + \underline{g} \quad (4.36)$$

let's assume a solution of the same form, or  $\underline{y} = \underline{M}\underline{z}$ . With this assumption, the original equation becomes

$$\underline{M}\underline{z}' = \underline{A}\underline{M}\underline{z} + \underline{g}$$

and premultiplying by  $\underline{M}^{-1}$  gives

$$\underline{z}' = \underline{M}^{-1}\underline{A}\underline{M}\underline{z} + \underline{M}^{-1}\underline{g}$$

or

$$\underline{z}' = \underline{D}\underline{z} + \underline{h} \quad \text{where} \quad \underline{D} = \underline{M}^{-1}\underline{A}\underline{M} \quad \text{and} \quad \underline{h} = \underline{M}^{-1}\underline{g} \quad (4.37)$$

Since  $\underline{D}$  is a diagonal matrix with the eigenvalues of the state matrix along the diagonal, this last equation represents a set of *uncoupled* first order ODEs which can be written as

$$\frac{d}{dt}z_i - \lambda_i z_i = h_i \quad (4.38)$$

This is a single linear ODE for the  $i^{\text{th}}$  component of  $\underline{z}$  and it has an integrating factor,  $e^{-\lambda_i t}$ , and a general solution

$$z_i(t) = e^{\lambda_i t} \left[ \int e^{-\lambda_i t} h_i(t) dt + c_i \right] \quad (4.39)$$

**Note:** Leaving out the  $c_i$  in this expression gives only a particular solution. However, if the integration constant is carried along, one gets the general solution.

Finally, with  $z_i(t)$  known, the desired solution to the original ODE is

$$\underline{y}(t) = \underline{M}\underline{z}(t) = \sum_i \underline{x}_i z_i(t) \quad (4.40)$$

Example 4.3B illustrates the Matrix Diagonalization method.

### Variation of Parameters

This third method can start with a slightly more general form for the system of 1<sup>st</sup> order ODEs,

$$\underline{y}'(t) = \underline{A}(t)\underline{y}(t) + \underline{g}(t) \quad (4.41)$$

In actual application, however, all the practical solution schemes require that the state matrix be constant. For this situation, the homogeneous solution can be written as

$$\underline{y}_h = \sum_i c_i \underline{y}_i \quad (4.42)$$

where  $\underline{y}_i$  is an individual solution to the homogeneous equation (assumed to be linearly independent), or

$$\underline{y}_h = \begin{bmatrix} \underline{y}_1 & \underline{y}_2 & \cdots & \underline{y}_n \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix} = \begin{bmatrix} \underline{x}_1 e^{\lambda_1 t} & \underline{x}_2 e^{\lambda_2 t} & \cdots & \underline{x}_n e^{\lambda_n t} \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix} = \underline{Y} \underline{c} \quad (4.43)$$

where  $\underline{Y}$  is a matrix whose columns are the solution vectors,  $\underline{y}_i(t)$ .

Now, for the particular solution, let's assume a solution of the form

$$\underline{y}_p = \underline{Y} \underline{u} \quad (4.44)$$

where we have replaced the constant vector with a vector of unknown functions (to be determined).

Noting that  $\underline{Y} = \underline{Y}(t)$ , upon substitution we have

$$\underline{Y}' \underline{u} + \underline{Y} \underline{u}' = \underline{A} \underline{Y} \underline{u} + \underline{g} \quad (4.45)$$

Note also that, from the homogeneous solution, we have

$$\begin{bmatrix} \underline{y}_1' & \underline{y}_2' & \cdots & \underline{y}_n' \end{bmatrix} = \begin{bmatrix} \underline{A} \underline{y}_1 & \underline{A} \underline{y}_2 & \cdots & \underline{A} \underline{y}_n \end{bmatrix} \quad \text{or} \quad \underline{Y}' = \underline{A} \underline{Y} \quad (4.46)$$

Therefore, canceling the terms in eqn. (4.45) associated with this equality, gives

$$\underline{Y} \underline{u}' = \underline{g} \quad (4.47)$$

or

$$\underline{u}' = \underline{Y}^{-1} \underline{g} \quad (4.48)$$

and

$$\underline{u}(t) = \int \underline{Y}^{-1}(\tau) \underline{g}(\tau) d\tau + \underline{c} \quad (4.49)$$

**Note:** As before, if  $\underline{c} = 0$  in the above expression, then

$$\underline{y}_p = \underline{Y} \underline{u} = \underline{Y} \int \underline{Y}^{-1}(\tau) \underline{g}(\tau) d\tau \quad (4.50)$$

If, however,  $\underline{c} \neq 0$ , we get the general solution,

$$\underline{y}(t) = \underline{Y} \underline{c} + \underline{Y} \int \underline{Y}^{-1}(\tau) \underline{g}(\tau) d\tau \quad (4.51)$$

Example 4.3C offers an illustration of the Variations of Parameters method.

**Example 4.1 -- Analytical Solution Methods (Homogeneous System with Distinct Roots)****Problem Description:**

Solve the following set of 1<sup>st</sup> order linear constant coefficient ODEs:

$$y_1' = 4y_2 \quad \text{and} \quad y_2' = 4y_1$$

**Problem Solution:**

1. First, let's write the equations in matrix form, or

$$\underline{y}' = \begin{bmatrix} 0 & 4 \\ 4 & 0 \end{bmatrix} \underline{y}$$

Therefore, we have a homogeneous matrix equation,  $\underline{y}' = \underline{A}\underline{y}$ , where the state matrix is

$$\underline{A} = \begin{bmatrix} 0 & 4 \\ 4 & 0 \end{bmatrix}$$

2. The eigenvalues of the state matrix are given by

$$|\underline{A} - \lambda \underline{I}| = \begin{vmatrix} -\lambda & 4 \\ 4 & -\lambda \end{vmatrix} = \lambda^2 - 16 = (\lambda - 4)(\lambda + 4) = 0$$

or

$$\lambda_1 = -4 \quad \text{and} \quad \lambda_2 = 4$$

and these are indeed distinct so the eigenvectors will be linearly independent. The eigenvectors are determined from  $(\underline{A} - \lambda_i \underline{I})\underline{x}_i = 0$ .

For  $\lambda_1 = -4$ , this gives  $4x_1 + 4x_2 = 0$  and  $4x_1 + 4x_2 = 0$ . Therefore,  $\underline{x}_1 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$  is a valid solution for the first eigenvector (to within some arbitrary constant).

Similarly for  $\lambda_2 = 4$ , we get  $-4x_1 + 4x_2 = 0$  and  $4x_1 - 4x_2 = 0$  or,  $\underline{x}_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

3. Now, with known (distinct) eigenvalues and eigenvectors, the homogeneous solution can be written as

$$\underline{y}(t) = c_1 \underline{x}_1 e^{\lambda_1 t} + c_2 \underline{x}_2 e^{\lambda_2 t} = \underline{M}\underline{z}$$

$$\underline{y}(t) = \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} c_1 e^{-4t} \\ c_2 e^{4t} \end{bmatrix} = \begin{bmatrix} c_1 e^{-4t} + c_2 e^{4t} \\ -c_1 e^{-4t} + c_2 e^{4t} \end{bmatrix}$$

or

$$y_1(t) = c_1 e^{-4t} + c_2 e^{4t} \quad \text{and} \quad y_2(t) = -c_1 e^{-4t} + c_2 e^{4t}$$

**Example 4.2 -- Analytical Solution Methods (Homogeneous System with Repeated Roots)****Problem Description:**

Solve the following set of 1<sup>st</sup> order linear constant coefficient ODEs:

$$y_1' = 4y_1 + y_2 \quad \text{and} \quad y_2' = -y_1 + 2y_2$$

**Problem Solution:**

1. First, let's write the equations in matrix form, or

$$\underline{y}' = \begin{bmatrix} 4 & 1 \\ -1 & 2 \end{bmatrix} \underline{y}$$

Therefore, we have a homogeneous matrix equation,  $\underline{y}' = \underline{A}\underline{y}$ , where the state matrix is

$$\underline{A} = \begin{bmatrix} 4 & 1 \\ -1 & 2 \end{bmatrix}$$

2. The eigenvalues of the state matrix are given by

$$|\underline{A} - \lambda \underline{I}| = \begin{vmatrix} 4 - \lambda & 1 \\ -1 & 2 - \lambda \end{vmatrix} = (4 - \lambda)(2 - \lambda) + 1 = 8 - 6\lambda + \lambda^2 + 1 = 0$$

or

$$\lambda^2 - 6\lambda + 9 = (\lambda - 3)^2 = 0$$

which gives  $\lambda_{1,2} = 3$ , and we have a case with repeated roots. In this case, we only have one solution,

$$\underline{y}_1 = \underline{x}_1 e^{3t}$$

where the eigenvector is determined from  $(\underline{A} - \lambda_1 \underline{I})\underline{x}_1 = 0$ .

For  $\lambda_1 = 3$ , this gives  $x_1 + x_2 = 0$  and  $-x_1 - x_2 = 0$ . Therefore,  $\underline{x}_1 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$  is a valid solution for the first eigenvector (to within some arbitrary constant).

3. Now, we need to determine a second linearly independent solution. Based on our previous development, let's choose  $\underline{y}_2$  as

$$\underline{y}_2 = (\alpha \underline{x}_1 t + \underline{z}) e^{3t}$$

where  $\alpha = 1$  and  $\underline{z}$  is determined by

$$(\underline{A} - \mu \underline{I})\underline{z} = \alpha \underline{x}$$

which, in this case, becomes

$$(\underline{\underline{A}} - 3\underline{\underline{I}})\underline{z} = \alpha \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

or

$$z_1 + z_2 = 1(\alpha) \quad \text{and} \quad -z_1 - z_2 = -1(\alpha)$$

which gives

$$\underline{z} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad \text{and} \quad \underline{y}_2 = \begin{bmatrix} 1 \\ -1 \end{bmatrix} te^{3t} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} e^{3t}.$$

Note that this solution is just one possible choice since there are an infinite number of solutions to this underdetermined system. This was expected since the matrix operator  $(\underline{\underline{A}} - \mu\underline{\underline{I}})\underline{z}$  is singular -- since  $\mu$  is an eigenvalue of  $\underline{\underline{A}}$ .

4. Now putting the two linearly independent solutions together gives the final solution, or

$$\underline{y} = c_1 \underline{y}_1 + c_2 \underline{y}_2 = (c_1 + c_2 t) \begin{bmatrix} 1 \\ -1 \end{bmatrix} e^{3t} + c_2 \begin{bmatrix} 1 \\ 0 \end{bmatrix} e^{3t}$$

or

$$y_1(t) = [c_1 + c_2(1+t)]e^{3t}$$

$$y_2(t) = -[c_1 + c_2 t]e^{3t}$$

This form of the solution (with terms containing  $e^{3t}$  and  $te^{3t}$ ) is exactly what we would expect for a linear 2<sup>nd</sup> order system with repeated roots!

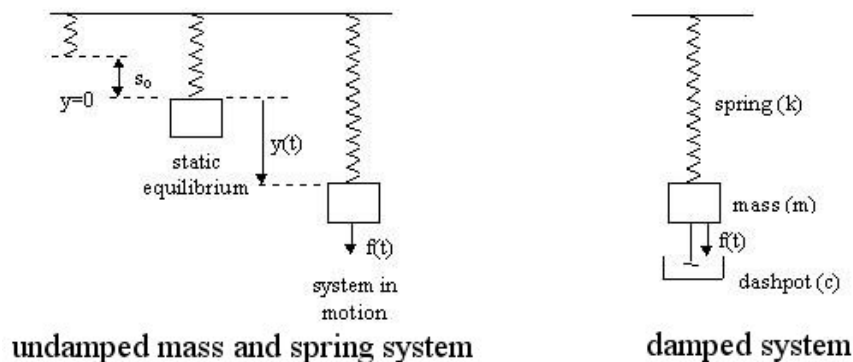
**Example 4.3 -- Analytical Solution Methods (Non-Homogeneous System)**  
*A Simple Mechanical System (One Degree of Freedom)*

**Problem Development:**

Consider a simple mechanical system consisting of a spring, mass, and dashpot (for viscous damping). Let  $y(t)$  be the displacement from equilibrium in the downward direction. Therefore,

$$m \frac{d^2 y}{dt^2} = mg - ks_0 - ky - c \frac{dy}{dt} + f(t)$$

In words, this says that the total force is the sum of the forces due to gravity, the restoring force of the linear spring,  $[-k(s_0 + y)]$ , the viscous damping force, and any external force acting on the mass, where  $s_0$  is the initial stretch due to the mass being attached to the unstretched spring, as shown in the sketch given below:



At equilibrium with no applied force, we have  $y(t) = 0$ , thus  $mg = ks_0$ . This says that the gravity force is exactly cancelled by the spring force associated with stretch  $s_0$ . Therefore, the defining equation for the system becomes

$$m \frac{d^2 y}{dt^2} + c \frac{dy}{dt} + ky = f(t)$$

As a specific case, let  $m = 1$ ,  $k = 2$ , and  $c = 3$  (with appropriate units) and assume that the forcing function is given by  $f(t) = 10e^{-2t}$ . Also choose  $y(0) = 0$  and  $y'(0) = -5$  as initial conditions, making the problem very specific.

**Problem Solution:**

As the first step in the solution, let's convert this problem to state form (two coupled 1<sup>st</sup> order equations). Letting  $x_1 = y$  and  $x_2 = y'$  give

$$x_1' = x_2 \quad \text{and} \quad x_2' = -\frac{k}{m}x_1 - \frac{c}{m}x_2 + \frac{1}{m}f(t)$$

as the defining equations. In matrix form, we have



$$\frac{d}{dt} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{c}{m} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix} f(t)$$

which corresponds exactly to the desired standard state form.

With the numerical data given above, the problem can be written as

$$\frac{d}{dt} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} 10e^{-2t} \quad \text{with} \quad x_1(0) = 0 \quad \text{and} \quad x_2(0) = -5$$

This is solved using three different methods in the following examples. In Example 4.3A, the Undetermined Coefficients method is used to find a solution. Example 4.3B uses the Matrix Diagonalization method and the Variation of Parameters method is illustrated in Example 4.3C.

### Example 4.3A -- Undetermined Coefficients

#### Problem Description:

Solve for  $\underline{y}(t)$  using the method of undetermined coefficients given

$$\frac{d}{dt}\underline{y} = \underline{A}\underline{y} + \underline{g}$$

with

$$\underline{A} = \begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix} \quad \text{and} \quad \underline{g} = \begin{bmatrix} 0 \\ 10 \end{bmatrix} e^{-2t} = \underline{b}e^{-2t}$$

#### Problem Solution:

1. First, let's find the *homogeneous solution*. Setting the right hand side to zero gives

$$\underline{y}' = \underline{A}\underline{y}$$

Now let  $\underline{y} = \underline{x}e^{\lambda t}$ , which gives,

$$\underline{x}\lambda e^{\lambda t} = \underline{A}\underline{x}e^{\lambda t}$$

or

$$(\underline{A} - \lambda \underline{I})\underline{x} = 0$$

which is a *standard eigenvalue problem*. Evaluating the eigenvalues, one has

$$|\underline{A} - \lambda \underline{I}| = \begin{vmatrix} -\lambda & 1 \\ -2 & -3 - \lambda \end{vmatrix} = \lambda(\lambda + 3) + 2 = \lambda^2 + 3\lambda + 2 = (\lambda + 2)(\lambda + 1)$$

or

$$\lambda_1 = -2 \quad \text{and} \quad \lambda_2 = -1$$

The eigenvector associated with  $\lambda_1 = -2$  is given by  $(\underline{A} - \lambda_1 \underline{I})\underline{x}_1 = 0$ , or

$$\begin{bmatrix} 2 & 1 \\ -2 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

or

$$2x_1 + x_2 = 0 \quad \text{and} \quad -2x_1 - x_2 = 0$$

Therefore,  $\underline{x}_1 = \begin{bmatrix} 1 \\ -2 \end{bmatrix}$  is a valid solution for the first eigenvector.

Similarly, for  $\lambda_2 = -1$ , we have

$$\begin{bmatrix} 1 & 1 \\ -2 & -2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

or

$$x_1 + x_2 = 0 \quad \text{and} \quad -2x_1 - 2x_2 = 0$$

and  $\underline{x}_2 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$  is the valid eigenvector for this case.

With the distinct eigenvalues and eigenvectors known, we can simply write the homogeneous solution as

$$\underline{y}_h = \sum_i c_i \underline{x}_i e^{\lambda_i t}$$

$$\underline{y}_h = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}_h = \begin{bmatrix} c_1 e^{-2t} + c_2 e^{-t} \\ -2c_1 e^{-2t} - c_2 e^{-t} \end{bmatrix}$$

2. Now for the **particular solution**, let's use the Method of Undetermined Coefficients. Because the forcing function is a simple exponential, one would generally pick a particular solution of the form,

$$\underline{y}_p = \underline{z} e^{-2t}$$

Normally this would be correct but, because of the specific form of the homogeneous solution for this problem, this choice won't work (because of the  $e^{-2t}$  term in the homogeneous solution). Therefore, we let

$$\underline{y}_p = (\alpha \underline{x}_1 t + \underline{z}) e^{-2t}$$

Then, upon substitution, we have

$$(-2\alpha \underline{x}_1 t + \alpha \underline{x}_1 - 2\underline{z}) e^{-2t} = \underline{A}(\alpha \underline{x}_1 t + \underline{z}) e^{-2t} + \underline{b} e^{-2t}$$

Canceling the  $e^{-2t}$  and noting that  $\underline{A}\alpha \underline{x}_1 t = -2\alpha \underline{x}_1 t$ , gives

$$\alpha \underline{x}_1 - 2\underline{z} = \underline{A}\underline{z} + \underline{b}$$

or

$$(\underline{A} + 2\underline{I})\underline{z} = \alpha \underline{x}_1 - \underline{b}$$

Our goal now is to evaluate this expression for the unknowns,  $\alpha$  and  $\underline{z}$ . Substituting the numerical values gives

$$\left[ \begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix} + \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \right] \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \alpha \begin{bmatrix} 1 \\ -2 \end{bmatrix} - \begin{bmatrix} 0 \\ 10 \end{bmatrix}$$

$$\begin{bmatrix} 2 & 1 \\ -2 & -1 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} \alpha \\ -2\alpha - 10 \end{bmatrix}$$

where we see that  $-\alpha = -2\alpha - 10$ , or  $\alpha = -10$ , gives a consistent, linearly dependent, set of equations. This is expected since the  $(\underline{\underline{A}} - \lambda_1 \underline{\underline{I}})$  operator is singular, which implies redundancy in the equations. Writing the equations explicitly gives

$$2z_1 + z_2 = \alpha \quad \text{and} \quad -2z_1 - z_2 = -2\alpha - 10$$

and with  $\alpha = -10$ , we have

$$z_2 = -10 - 2z_1$$

This gives an infinite number of solutions. Let's choose  $z_1 = -5$  which gives  $z_2 = 0$ , and  $\underline{z} = [-5 \ 0]^T$  becomes a valid result for the unknown coefficient vector.

Thus, the particular solution is given by

$$\underline{y}_p = (\alpha \underline{x}_1 t + \underline{z}) e^{-2t} = \left[ -10t \begin{bmatrix} 1 \\ -2 \end{bmatrix} + \begin{bmatrix} -5 \\ 0 \end{bmatrix} \right] e^{-2t} = \begin{bmatrix} -10t - 5 \\ 20t \end{bmatrix} e^{-2t}$$

3. The **general solution** is formed as the sum of the homogeneous and particular solutions, or

$$\underline{y}(t) = \underline{y}_h(t) + \underline{y}_p(t)$$

$$\underline{y}(t) = \begin{bmatrix} (c_1 - 10t - 5)e^{-2t} + c_2 e^{-t} \\ (-2c_1 + 20t)e^{-2t} - c_2 e^{-t} \end{bmatrix}$$

4. Finally, the **unique solution** is obtained by imposing the specific initial conditions for this case, or

$$\underline{y}(0) = \begin{bmatrix} y_1(0) \\ y_2(0) \end{bmatrix} = \begin{bmatrix} 0 \\ -5 \end{bmatrix} = \begin{bmatrix} c_1 - 5 + c_2 \\ -2c_1 - c_2 \end{bmatrix}$$

Therefore, from the first equation, we have

$$c_1 = 5 - c_2$$

and putting this into the second equation gives

$$-2(5 - c_2) - c_2 = -5$$

or

$$c_2 = 5 \quad \text{and} \quad c_1 = 0$$

Therefore, the final analytical solution for this problem is

$$\underline{y}(t) = \begin{bmatrix} -(10t + 5)e^{-2t} + 5e^{-t} \\ 20te^{-2t} - 5e^{-t} \end{bmatrix}$$

**Example 4.3B -- Matrix Diagonalization****Problem Description:**

Solve for  $\underline{y}(t)$  using the matrix diagonalization method given that

$$\frac{d}{dt}\underline{y} = \underline{A}\underline{y} + \underline{g}$$

with

$$\underline{A} = \begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix} \quad \text{and} \quad \underline{g} = \begin{bmatrix} 0 \\ 10 \end{bmatrix} e^{-2t} = \underline{b}e^{-2t}$$

**Problem Solution:**

In this example we want to use the matrix diagonalization method for solving the state space equations. Therefore, we assume a solution of the form

$$\underline{y} = \underline{M}\underline{z}$$

Substitution of this into the defining equation gives (see above development)

$$z_i(t) = e^{\lambda_i t} \left[ \int e^{-\lambda_i t} h_i(t) dt + c_i \right]$$

where  $\underline{h} = \underline{M}^{-1}\underline{g}$  and  $\underline{M}$  is the modal matrix.

Also from Example 4.3A, we have

$$\lambda_1 = -2 \quad \text{and} \quad \lambda_2 = -1 \quad \text{and} \quad \underline{M} = \begin{bmatrix} 1 & 1 \\ -2 & -1 \end{bmatrix}$$

1. Let's construct the above solution very systematically. First,  $\underline{M}^{-1}$  is given by

$$\underline{M}^{-1} = \frac{1}{-1+2} \begin{bmatrix} -1 & 2 \\ -1 & 1 \end{bmatrix}^T = \begin{bmatrix} -1 & -1 \\ 2 & 1 \end{bmatrix}$$

Therefore,  $\underline{h}$  is simply

$$\underline{h} = \underline{M}^{-1}\underline{g} = \begin{bmatrix} -1 & -1 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 10 \end{bmatrix} e^{-2t} = \begin{bmatrix} -10 \\ 10 \end{bmatrix} e^{-2t}$$

2. Now we can evaluate the functions,  $z_i(t)$ , or,

$$z_1(t) = e^{-2t} \left[ \int e^{2t} (-10e^{-2t}) dt + c_1 \right] = e^{-2t} [(-10t) + c_1] = (c_1 - 10t)e^{-2t}$$

and

$$z_2(t) = e^{-t} \left[ \int e^t (10e^{-2t}) dt + c_2 \right] = e^{-t} [(-10e^{-t}) + c_2] = -10e^{-2t} + c_2e^{-t}$$

3. Now multiplying  $\underline{z}$  by the modal matrix gives

$$\underline{y} = \underline{\underline{M}}\underline{z} = \begin{bmatrix} 1 & 1 \\ -2 & -1 \end{bmatrix} \begin{bmatrix} (c_1 - 10t)e^{-2t} \\ -10e^{-2t} + c_2e^{-t} \end{bmatrix} = \begin{bmatrix} (c_1 - 10 - 10t)e^{-2t} + c_2e^{-t} \\ (-2c_1 + 20t + 10)e^{-2t} - c_2e^{-t} \end{bmatrix}$$

4. Finally, applying the initial conditions gives the unique solution, or

$$\underline{y}(0) = \begin{bmatrix} y_1(0) \\ y_2(0) \end{bmatrix} = \begin{bmatrix} 0 \\ -5 \end{bmatrix} = \begin{bmatrix} c_1 - 10 + c_2 \\ -2c_1 + 10 - c_2 \end{bmatrix}$$

Therefore, from the first equation, we have

$$c_1 = 10 - c_2$$

and putting this into the second equation gives

$$-2(10 - c_2) + 10 - c_2 = -5$$

or

$$c_2 = 5 \quad \text{and} \quad c_1 = 5$$

Therefore, the final analytical solution for this problem is

$$\underline{y}(t) = \begin{bmatrix} -(10t + 5)e^{-2t} + 5e^{-t} \\ 20te^{-2t} - 5e^{-t} \end{bmatrix}$$

which is exactly the same solution as found for Example 4.3A (as it should be).

### Example 4.3C -- Variation of Parameters

#### Problem Description:

Solve for  $\underline{y}(t)$  using the variation of parameters method given that

$$\frac{d}{dt}\underline{y} = \underline{A}\underline{y} + \underline{g}$$

with

$$\underline{A} = \begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix} \quad \text{and} \quad \underline{g} = \begin{bmatrix} 0 \\ 10 \end{bmatrix} e^{-2t} = \underline{b}e^{-2t}$$

#### Problem Solution:

Using the results developed previously for the variation of parameter method, we have

$$\underline{y}(t) = \underline{Y}\left[\underline{c} + \int \underline{Y}^{-1}(\tau)\underline{g}(\tau)d\tau\right] \quad \text{where} \quad \underline{Y} = \begin{bmatrix} \underline{x}_1 e^{\lambda_1 t} & \underline{x}_2 e^{\lambda_2 t} \end{bmatrix}$$

and, for this case, this matrix can be written explicitly as (see Example 4.3A),

$$\underline{Y} = \begin{bmatrix} e^{-2t} & e^{-t} \\ -2e^{-2t} & -e^{-t} \end{bmatrix}$$

1. Again, proceeding systematically to evaluate the above expression, let's first find  $\underline{Y}^{-1}$ . This relation is given by

$$\underline{Y}^{-1} = \frac{1}{-e^{-3t} + 2e^{-3t}} \begin{bmatrix} -e^{-t} & 2e^{-2t} \\ -e^{-t} & e^{-2t} \end{bmatrix}^T = e^{3t} \begin{bmatrix} -e^{-t} & -e^{-t} \\ 2e^{-2t} & e^{-2t} \end{bmatrix} = \begin{bmatrix} -e^{2t} & -e^{2t} \\ 2e^t & e^t \end{bmatrix}$$

and just to check this result we verify that  $\underline{Y}^{-1}\underline{Y} = \underline{I}$ , or

$$\underline{Y}^{-1}\underline{Y} = \begin{bmatrix} -e^{2t} & -e^{2t} \\ 2e^t & e^t \end{bmatrix} \begin{bmatrix} e^{-2t} & e^{-t} \\ -2e^{-2t} & -e^{-t} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

2. Now, the function  $\underline{Y}^{-1}\underline{g}$  can be written as

$$\underline{Y}^{-1}\underline{g} = \begin{bmatrix} -e^{2t} & -e^{2t} \\ 2e^t & e^t \end{bmatrix} \begin{bmatrix} 0 \\ 10 \end{bmatrix} e^{-2t} = \begin{bmatrix} -10 \\ 10e^{-t} \end{bmatrix}$$

3. Performing the integrals gives

$$\int \underline{Y}^{-1}(t)\underline{g}(t)dt = \begin{bmatrix} -10t \\ -10e^{-t} \end{bmatrix}$$

and premultiplying by  $\underline{Y}$  and adding the homogeneous solution gives the general solution, or

$$\underline{y}(t) = \begin{bmatrix} e^{-2t} & e^{-t} \\ -2e^{-2t} & -e^{-t} \end{bmatrix} \begin{bmatrix} c_1 - 10t \\ c_2 - 10e^{-t} \end{bmatrix} = \begin{bmatrix} (c_1 - 10 - 10t)e^{-2t} + c_2e^{-t} \\ (-2c_1 + 20t + 10)e^{-2t} - c_2e^{-t} \end{bmatrix}$$

Note that this is the same solution as derived in Example 4.3B.

4. As before, we complete this problem by applying the initial conditions to get the unique solution, or

$$\underline{y}(0) = \begin{bmatrix} y_1(0) \\ y_2(0) \end{bmatrix} = \begin{bmatrix} 0 \\ -5 \end{bmatrix} = \begin{bmatrix} c_1 - 10 + c_2 \\ -2c_1 + 10 - c_2 \end{bmatrix}$$

Therefore, from the first equation, we have

$$c_1 = 10 - c_2$$

and putting this into the second equation gives

$$-2(10 - c_2) + 10 - c_2 = -5$$

or

$$c_2 = 5 \quad \text{and} \quad c_1 = 5$$

Therefore, the final analytical solution for this problem is

$$\underline{y}(t) = \begin{bmatrix} -(10t + 5)e^{-2t} + 5e^{-t} \\ 20te^{-2t} - 5e^{-t} \end{bmatrix}$$

which is exactly the same solution as found using the method of undetermined coefficients and the matrix diagonalization method (see Example 4.3A and Example 4.3B).



### Numerical Solution of General IVPs

The analytical solution methods just discussed for treating systems of first order ODEs are indeed important and they represent powerful tools that can help solve many problems. However, in almost all cases, these analytical solution methods are restricted to linear time invariant systems, and from practical concerns, they are only useful for relatively low order systems (usually 2 or 3 equations). As we saw previously for the case of a single first order equation, when the analytical methods become tedious, or in some cases, completely inappropriate, we simply turn to numerical techniques for solving the desired equations. And, of course, a wide range of numerical methods incorporated within various computer simulation tools is the standard methodology used in industry for solving real engineering problems.

Fortunately we already know how to numerically integrate first order equations of the form  $y' = f(x, y)$  using adaptive step control techniques. In fact, both the one-step Runge-Kutta formulas and the Adams-Bashford-Moulton multistep method were discussed in some detail, and the specific implementation of the RK23 method within the Matlab *ode23* routine has been described and illustrated via several examples. The reader is encouraged to review this material as needed (see Section I - Numerical Solution Methods).

A convenient feature of the numerical methods discussed previously is that they can also be used to solve the general IVP involving  $n$  equations and  $n$  initial conditions. In this case, one usually starts with a system of coupled 1<sup>st</sup> order equations (recall that an  $n^{\text{th}}$  order system can always be converted into a set of  $n$  1<sup>st</sup> order equations) written in vector form, or

$$\underline{y}' = \underline{f}(x, \underline{y}) \quad (4.52)$$

where the underline notation simply implies vector quantities of length  $n$ . Note that this notation is almost identical, except for the underlines, to that for the single first order equation. Thus, all of our previous development for a single equation can be duplicated for the case of  $n$  coupled equations by simply including the vector notation within the development.

For example, the integration formula for the first order Euler method for a single ODE was derived previously as

$$y_{i+1} = y_i + hf(x_i, y_i) \quad (4.53)$$

where  $h$  is the step size. For the general case of  $n$  equations, the Euler integration formula is

$$\underline{y}_{i+1} = \underline{y}_i + hf(\underline{x}_i, \underline{y}_i) \quad (4.54)$$

Thus, everything we did as review in Section I is still applicable for the general case by simply converting to vector notation. Also, because of the base matrix formalism built into the Matlab package, the Matlab ODE solvers used previously are already set up to handle vector functions. Thus, we already know how to numerically solve general initial value problems (IVPs)!

The only remaining point is to illustrate this general capability via example. Two such cases have been formulated in Example 4.4 and Example 4.5. These examples show the essential features of using *ode23* for solving general IVPs within the Matlab environment. The student is encouraged to study these carefully, since this is such an important class of problems that occurs in all fields of engineering design and analysis.

**Example 4.4 -- Numerical Solution Methods (Example 4.3 Revisited)****Problem Description:**

Solve the following coupled system of equations using Matlab's *ode23* routine and compare the numerical solutions to the analytical solution from Example 4.3 - Analytical Solution Schemes:

$$\frac{d}{dt} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{c}{m} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix} f(t)$$

Let  $m = 1$ ,  $k = 2$ , and  $c = 3$  (with appropriate units) and  $f(t) = 10e^{-2t}$ . Also assume the initial conditions are  $x_1(0) = 0$  and  $x_2(0) = -5$ .

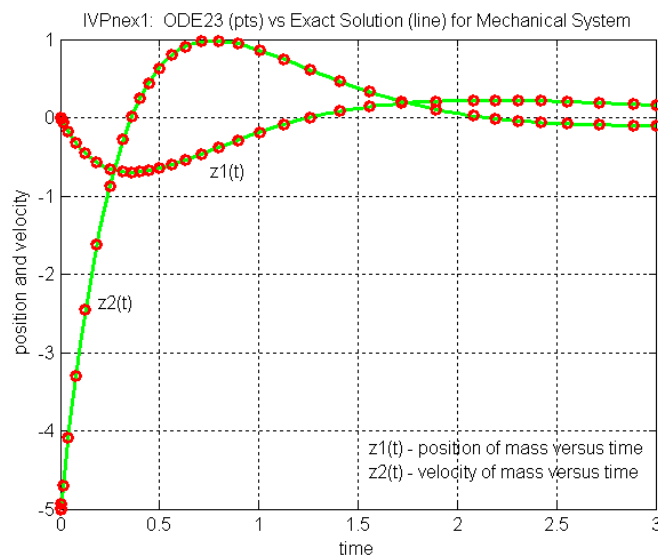
**Problem Solution:**

1. The unique solution for this problem was derived above as (see Ex. 3 - Analytical Examples)

$$\underline{x}(t) = \begin{bmatrix} -(10t + 5)e^{-2t} + 5e^{-t} \\ 20te^{-2t} - 5e^{-t} \end{bmatrix}$$

This vector function was evaluated and plotted over the range  $t = [0, 2]$  in Matlab.

2. A numerical solution to the given IVP was obtained using the *ode23* numerical integration routine in Matlab. The standard state space form is used, and the forcing function is evaluated in a separate routine, making it rather easy to modify the input function if desired.



**Fig. 4.1 Comparison of numerical and analytical solutions for Example 4.4.**

3. The numerical and analytical solutions are compared in Fig. 4.1. Clearly the numerical and analytical solutions show good agreement. These results were obtained using the main Matlab file **ivpnex1.m**. This script file also refers to function files **ivpnex1a.m** and **ivpnex1b.m**, which perform the function evaluations for *ode23* and evaluate the forcing function for any value of the independent variable, respectively. The three Matlab files written to solve this problem are listed in Table 4.1. The reader is encouraged to study these files as needed. These are pretty basic but, if you are new to Matlab, you should take some time to make sure you fully understand the details! This sequence of Matlab m-files represents a good example of *solving IVPs written in state space form* and of *using Matlab function files* for evaluating various functions.

**Table 4.1 - Listing of Matlab files for Example 4.4.**

```
%
% IVPNEX1.M Simulation of a Simple Mechanical System
% (Example for Numerical Integration of System of ODEs (IVPs))
%
% This is a sample problem that considers a simple mechanical system
% consisting of a mass, spring, and dashpot (to represent viscous
% damping). The system also contains a specified forcing function, f(t).
% A force balance on the system gives the following second order ODE:
% my''(t) + cy'(t) + ky(t) = f(t)
%
% For specificity, let
% m = 1, k = 2, c = 3, and f(t) = 10exp(-2t)
% and for initial conditions, let
% y(0) = 0 and y'(0) = -5
% all with appropriate units, of course.
%
% This can be put in matrix form by letting z1 = y and z2 = y', giving
% z1' = z2
% z2' = -(k/m)z1 - (c/m)z2 + (1/m)f(t)
% or
% z' = A*x + B*u (where A & B are state space matrices and u is input)
%
% The goal of this example is to illustrate how to solve problems of this
% type within Matlab using the ODE23 function (uses the RK23 method). The
% numerical solution is also compared to an analytical solution derived in the
% course Lecture Notes for 10/24.539.
%
% This file also uses the following two files:
% ivpnex1a.m - function file for ode23
% ivpnex1b.m - function file for generating value of forcing function
%
% File prepared by J. R. White, UMass-Lowell (Aug. 2003).
%
%
% getting started
% clear all, close all, nfig = 0;
%
% setup problem specific data
% m = 1; k = 2; c = 3;
% yo = 0; ypo = -5;
%
% setup constant matrices for state space formulation
% AA = [0 1; -k/m -c/m]; BB = [0 1/m]';
%
% time domain of interest
% to = 0; tf = 3; te = linspace(to,tf,61);
%
% evaluate and plot forcing function over interval
% ff = ivpnex1b(te);
% nfig = nfig+1; figure(nfig)
% plot(te,ff,'r-','LineWidth',2)
```

```

    title('IVPnex1: Forcing Function Used in Simulation')
    xlabel('time'),ylabel('f(t)'),grid
%
% solve 2x2 system of ODEs (Initial Value Prob -- IVP)
zo = [yo ypo]';
tol = 0.001; options = odeset('RelTol',tol);
[t,z] = ode23('ivpnex1a',[to tf],zo,options,AA,BB);
%
% exact solution (from lecture notes)
te = te'; ze = zeros(length(te),2);
ze(:,1) = -(10*te + 5).*exp(-2*te) + 5*exp(-te);
ze(:,2) = 20*te.*exp(-2*te) - 5*exp(-te);
%
% compare exact and numerical solution
nfig = nfig+1; figure(nfig)
plot(te,ze(:,1),'-g',te,ze(:,2),'-g',t,z(:,1),'or',t,z(:,2),'or','LineWidth',2)
title('IVPnex1: ODE23 (pts) vs Exact Solution (line) for Mechanical System')
xlabel('time'),ylabel('position and velocity'),grid
gtext('z1(t)'),gtext('z2(t)')
%
range = axis;
xt = range(1) + 0.52*(range(2)-range(1));
yt1 = range(3) + 0.13*(range(4)-range(3));
yt2 = range(3) + 0.08*(range(4)-range(3));
text(xt,yt1,'z1(t) - position of mass versus time')
text(xt,yt2,'z2(t) - velocity of mass versus time')
%
% end of problem

%
% IVPNEX1A.M Function evaluation for ODE step in IVPNEX1.M
%
% given differential eqn.  $z' = Az + g$  where  $g = B*u$ 
%
% uses function file ivpnex1b.m to evaluate forcing function at time t
%
% File prepared by J. R. White, UMass-Lowell (Aug. 2003).
%
function zp = odefile(t,z,flag,AA,BB)
u = ivpnex1b(t); g = BB*u;
zp = AA*z + g;
%
% end of function

%
% IVPNEX1B.M Function file to evaluate forcing function at any t
%
% given function:  $f(t) = 10*exp(-2t)$ 
%
% this file is used by ivpnex1.m and ivpnex1a.m
%
% File prepared by J. R. White, UMass-Lowell (Aug. 2003).
%
function f = funfile(t)
f = 10*exp(-2*t);
%
% end of function
%

```

**Example 4.5 -- Numerical Solution Methods (Chemical Reactor Dynamics)**  
*Semi-Batch Chemical Reactor Dynamics*

**Problem Description:**

Model, simulate, and analyze the dynamics of a 3-component semi-batch chemical reactor (see specific description below). Assume the following data in your analysis:

$$Q_o = 10.0 \text{ liters/sec} \quad V_o = 50.0 \text{ liters} \quad C_{A_o} = 1.0 \text{ gmoles/liter}$$

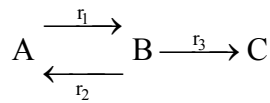
where  $Q_o$ ,  $V_o$ , and  $C_{A_o}$  represent the inlet flow rate, initial reactor volume, and feed concentration of component A, respectively. Also assume that the rate constants for the various processes are

$$k_1 = 0.1 \quad k_2 = 0.5 \quad \text{and} \quad k_3 = 0.05$$

Determine the concentration of components A, B, and C versus time. What is the maximum concentration of component B? How long does it take to reach this state?

**Problem Solution:***Model Development*

Consider the kinetics of a 3-component semi-batch chemical reactor with the following reaction scheme,



where the rates are proportional to the component concentration or the square of the concentration, as follows:

$$r_1 = k_1 C_A^2 \quad r_2 = k_2 C_B \quad \text{and} \quad r_3 = k_3 C_B^2$$

with the concentrations given in gmoles/liter and the reaction rates in gmoles/liter-sec.

Assume that the above reactions occur in a semi-batch reactor -- that is a reactor which has flow in, but no flow out of the system. Initially, the reactor is filled to volume  $V_o$  with an inert liquid. At time zero, a stream containing component A with concentration  $C_{A_o}$  is fed to the reactor at a constant flow rate  $Q_o$ . For this situation, a mole balance on each component gives

$$\frac{d}{dt} n_A = Q_o C_{A_o} + V_R (-r_1 + r_2)$$

$$\frac{d}{dt} n_B = V_R (r_1 - r_2 - r_3)$$

$$\frac{d}{dt} n_C = V_R (r_3)$$

where  $n_i(t)$  is the number of moles of component  $i$  at any time  $t$  and  $V_R(t)$  is the reactor volume versus time and is given by

$$V_R(t) = V_o + Q_o t$$

Using the definition of concentration,  $C_i = n_i/V_R$ , the above kinetics equations can be written in final form as

$$\frac{d}{dt}n_A = Q_o C_{A_o} - \frac{k_1}{V_R}n_A^2 + k_2 n_B$$

$$\frac{d}{dt}n_B = \frac{k_1}{V_R}n_A^2 - k_2 n_B - \frac{k_3}{V_R}n_B^2$$

$$\frac{d}{dt}n_C = \frac{k_3}{V_R}n_B^2$$

where  $n_A = n_B = n_C = 0$  at  $t = 0$ . These equations, along with the expression for  $V_R(t)$ , represent a set of simultaneous, first-order, nonlinear, variable coefficient equations that describe the dynamics of this particular chemical system.

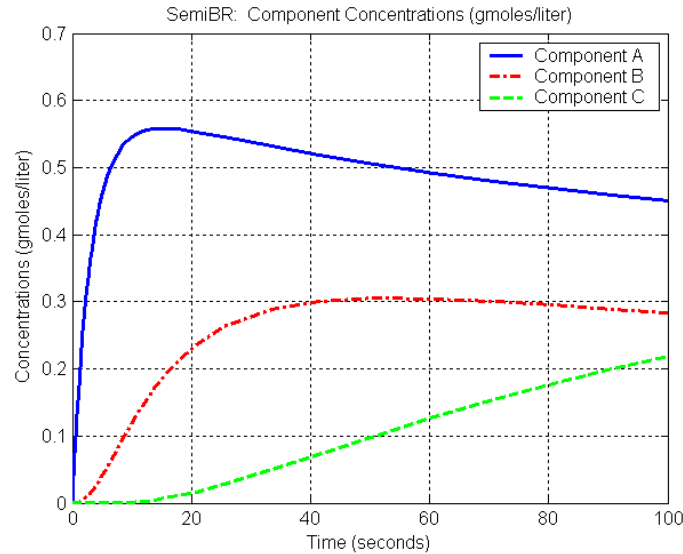
### *Solution Method*

This problem is solved numerically using Matlab. The program logic utilized is outlined as follows:

1. Specify problem data.
2. Use the ***ode23*** routine in Matlab to solve for the number of moles of each component versus time,  $n_i(t)$ .
3. Compute the reactor volume and convert the number of moles to concentration,  $C_i = n_i/V_R$ .
4. Plot the component concentrations versus time and identify the time and magnitude of the peak concentration of component B. In particular, component B reaches a fairly broad peak between 50-60 seconds after startup and has a magnitude of slightly over 0.3 gmoles/liter at maximum concentration. The Matlab plot for  $C_i(t)$  is shown in Fig. 4.2.

### *The Matlab Files*

The data for this example were generated with the main Matlab m-file ***semibr.m***. This file also requires the ODE function file ***semibra.m***. These files are listed in Table 4.2. This set of Matlab m-files illustrates the solution procedure for a realistic IVP. In particular, this problem includes a good example of the ***numerical solution scheme for a coupled set of nonlinear, variable coefficient, first order differential equations*** within the Matlab environment. This class of IVPs is very common in a wide range of engineering disciplines. Thus, a full understanding of this problem should aid in the solution of similar problems in the future.



**Fig. 4.2** Component concentrations generated by the semibr.m program.

**Table 4.2 - Listing of the semibr.m and semibra.m Matlab files.**

```

%
% SEMIBR.M      Simulation of the Dynamics of a Semi-Batch Chemical Reactor
%               (A Typical Initial Value Problem)
%
% This project involves the modeling/solution of the dynamics of a semi-batch
% chemical reactor.  A semi-batch reactor is one that has flow in, but no flow out.
% The specific system of interest has three components and the reactor is fed
% with constant flow and feed concentration of species A.  Components B and C
% are formed from component A.
%
% The defining eqns for the number of moles of each component are:
%   dNA/dt = Qo*CAo - (k1/VR)*NA^2 + k2*NB
%   dNB/dt = (k1/VR)*NA^2 - k2*NB - (k3/VR)*NB^2
%   dNC/dt = (k3/VR)*NB^2
% with
%   VR = Vo + Qo*t
% and initial conditions
%   NAO = NBo = NCo = 0      (all components initially at zero concentration)
%
% These equations will be solved for a specified set of data.  The goal is
% to determine the concentrations versus time, Ci = Ni/VR, with particular
% focus on the time when the second component (B) reaches its maximum
% concentration.
%
% also see SEMIBRA.M - ODE23 function file
%
% File prepared by J. R. White, UMass-Lowell (Aug. 2003)
%
%
% getting started
%   clear all, close all, nfig = 0;
%
% generic and plant specific data
%   Qo = 10.0;      % inlet flow rate (liters/sec)
%   Vo = 50.0;     % initial volume of semi-batch reactor (liters)
%   CAo = 1.0;     % concentration of component A in feed (gmole/liter)
%   K1 = 0.1;  K2 = 0.05;  K3 = 0.05; % rate constants
%
% now simulate system for several seconds (past peak in concentration in B)

```

```

to = 0;    tf = 100;    % time interval
no = [0 0 0]';    % initial conditions
tol = 0.001;    options = odeset('RelTol',tol);
[t,n] = ode23('semibra',[to tf],no,options,Qo,Vo,CAo,K1,K2,K3);

%
% compute concentrations
VR = Vo + Qo*t;    C = zeros(size(n));
for i = 1:3    C(:,i) = n(:,i)./VR;    end

%
% plot primary results
nfig = nfig+1;    figure(nfig)
plot(t,C(:,1),'b-',t,C(:,2),'r-',t,C(:,3),'g--','LineWidth',2),grid
title('SemiBR: Component Concentrations (gmoles/liter)')
xlabel('Time (seconds)'),ylabel('Concentrations (gmoles/liter)')
legend('Component A','Component B','Component C')

%
% end of simulation

%
% SEMIBRA.M    Function for ODE to Evaluate Chemical Reaction Rate Equations
%
% The defining eqns are:
%    dNA/dt = Qo*CAo - (k1/VR)*NA^2 + k2*NB
%    dNB/dt = (k1/VR)*NA^2 - k2*NB - (k3/VR)*NB^2
%    dNC/dt = (k3/VR)*NB^2
% with
%    VR = Vo + Qo*t
%
% File prepared by J. R. White, UMass-Lowell (Aug. 2003)
%
function np = odefile(t,n,flag,Qo,Vo,CAo,K1,K2,K3)
np = zeros(length(n),1);
VR = Vo + Qo*t;
np(1) = Qo*CAo - (K1/VR)*n(1)^2 + K2*n(2);
np(2) = (K1/VR)*n(1)^2 - K2*n(2) - (K3/VR)*n(2)^2;
np(3) = (K3/VR)*n(2)^2;

%
% end of function
%

```