MAE 305
Engineering Mathematics I
Princeton University

Assignment # 5

October 3, 1997

Due on Friday, 2PM, October 17, 1997

1. **Where we are.** We are now experts on ODE probelms with one single dependent variable. The first order case was dealt with in Chapter 2. We solve linear inhomogeneous first order ODE's with time dependent coefficients (integration factor). Aside from this, there is a short list of special first order ODE's, including some nonlinear ones, that we know special tricks. We worried about existence and uniqueness, and was given the assurance that if the right hand side of a quasi-linear first order ODE satisfies the Lipschitz condition in a box enclosing the initial point, we are OK in the box. Of course we now also know what is a quasi-linear ODE whatever the order of the ODE is (the highest order derivative term can be made to appear in a linear fashion; you don't care how the lower order derivatives and the unknown itself appear). We also learned quite a bit about second order linear ODEs in Chapter 3. There are some tricks to make some special second order ODE into first order ODEs; and then there is the trick on the second order *Euler Equation* which can easily be generalized to work for any order Euler Equation. We learned about the Wronskian (the determinant of the matrix we called **W** in class, and it provides the litmus test on whether a bunch of given solutions are linearly independent), and of course you understand the concept of linear independence of solutions (is there a faker in the collection of solutions?). We represent the solution of a linear nonhomogeneous problem as the sum of its *homogeneous part* and its *nonhomogeneous part*—and look for each part separately. Then there is the method of undetermined coefficients: have an inspired guess of the form of the solutions (usually you pick them among sine, cosine, polynominals, and exponentials and their products) and see if your

guess works. Remember the power of the method of superposition!!! (Work on the inhomogeneous solution one group of terms at a time). Then there is the method of variation of parameters which allows you to reduce the order of your nonhomogeneous problem if you somehow know (at least one or all of) the homogeneous solutions The simplest and important case is the constant coefficient case. You learned the "physical meaning" of the coefficients (damping coefficient and natural frequency). You learned that the homogeneous solution for a constant coefficient second order ODE can be expressed as exponentials, the "root" r which shows up in the exponent can sometimes be complex (if so, they come in complex conjugate pairs). And you learned what happened when the forcing term is some periodic function of time, and what happens to the amplitude of the steady-state response when you vary the forcing frequency. I, of course, hope that you remember something about my description of the nonlinear Duffing equation.

For second order linear constant coefficient ODEs, there are only two roots for r. When these two roots are idential—when the root is a repeated root—the solution is $t * \exp(rt)$. This result is generalized in Chapter 4, which deals with linear equations of higher orders. For example, when the root is a triple repeat, the two needed 'additional' homogeneous solutions are $t * \exp(rt)$ and $t^2 \exp(rt)$.

And, of course, you are totally competent to get the solutions to any of the problems numerically using computer software.

2. **New Readings**. Boyce and DiPrima, Chapter 7: System of First order linear equations. First of all, lets agree on notations. We are focusing our attention on equation (15) in Boyce and Diprima:

$$\begin{aligned}
\frac{dx_1}{dt} &= p_{11}x_1 + \ldots + p_{1n}x_N + g_1(t), \\
\frac{dx_2}{dt} &= p_{21}x_1 + \ldots + p_{2n}x_N + g_2(t), \\
&\vdots \\
\frac{dx_n}{dt} &= p_{n1}x_1 + \ldots + p_{nn}x_N + g_n(t).
\end{aligned}$$

You must agree that

$$\frac{dx_n}{dt} = P_{nn'}(t)x_{n'} + g_n(t) \tag{1}$$

using the Einstein Summation Convention is the clear choice of a lazy-person to represent the same thing. Note n' is the dummie index because it is repeated. This equation can also be represented by

$$\frac{d\mathbf{x}}{dt} = \mathbf{P}\mathbf{x} + \mathbf{g}. \tag{2}$$

where the bold-face is indicated on the blackboard by an overbar. Don't you agree this is even better? Here, **x** and **g** are *column vectors*, and **P** is a square matrix. Now, (2) looks just like a single first order ODE for a single unknown. But it is NOT. It is a system of n ODE's for the n elements of the **x** vector which are our n dependent variables. For most of this Chapter, we will be dealing with a real constant **P** matrix.

Now *if* **P** is a diagonal matrix, the problem is then trivial: you just have n *uncoupled* first order linear ODE's (integrating factor!). So the big deal is to learn what to do when **P** is not diagonal. In many math courses, (2) is sometimes studied under the restriction that **P** is *symmetric*. Indeed symmetric **P** is a lot easier to deal with—but we want to be able to handle eq.(2) wheter **P** is symmetric or not. Our task this week is to deal with **P** in all its glories: all we know is that it is real. It may be symmetric, or it may not. It may be diagonal (thank God!), but then it most probably is not. Obviously, to pass this course, we will need to take on all comers.

You may asked: if the **P** given to me is not diagonal, why don't we just diagonalize it? Yes, indeed. This is the main theme (when **P** is a constant matrix)! Diagonalize! Diagonalize! Diagonalize! (now, after all the shouting, what does "diagonalize" mean?) But is it always possible to diagonalize a given arbitrary matrix (what did your instructor in Math 202/204 say?)? Unfortunately, you may remember that not all square matrix can be diagonalized (Aha! I remember when the eigenvalues of the matrix **P** are repeated I *may* be in trouble). And of course you vaguely remember a fellow by the name of Jordan and the form that bears his name. This is where all these wonderful stuff comes together! So cheer up.

3. **Comments on Readings and Homeworks.**

§**7.1: Introduction.** pp.335-339. Light readings. Trying to convince you that system of linear constant coefficient ODE's actually describe real problems. Then it gives you the mumbo-jumbo like Theorem 7.1.1 (everything is OK if the *Jacobian* of F_i with respect to x_j exists and is continuous—it satisfies the Lipschitz Condition), followed by the jumbo-mumbo Theorem 7.1.2. (Ask me again in class what is a Jacobian if you don't remember). Basically, what Boyce and Diprima is trying to tell you is: whenever Equation (15) on page 339 is not pathological, we can guarantee uniqueness and existence of iniitial value problems.

Do the following two problems.

- pp. 340-344, Problem 11.
- Problem 12.

You are asked to find the solution analytically. When you are asked to "graph" the solution, the book wants you to graph x_1 versus x_2 (I took a peek at the back of the book). If you want to, you can get the plot by using Matlab. After computing the solution with

```
[t,x]=ode23('problem',0,20,[-2 2]);
```

you can plot x_2 versus x_1 with a red line by:

```
plot(x(:,1),x(:,2),'r')
```

Try it.

§**7.2 Review of Matrices** pp. 344-351. Of course, you all passed Math 202 or 204. So this is just a review. But now you have Matlab. Start Matlab, and create two matrices:

```
john=[1 -2  0; 3 2 1; -2 1 3]; mary=[4 -2 3; -1 5 0; 6 1 2];
```

Now you can find `john+mary` and `john*mary` using matlab just by typing what you want! Now: `john*mary` means the *matrix multiplication* of john and mary. If you want the transpose of john, ask for `john'` . Yes, a simple prime post-fix gives the transpose. To get the inverse of `john` , ask for `inv(john)` and you will get it.

- pp. 351-353, Problem 1(a,b,c,d). Note: BA and AB (matrix multiplication) gives different answers!

- Problem 3(a,b,c,d).
- Problem 7. Pick $\alpha = 1.111$ and $\beta = 9.999$. Use Matlab to 'prove' 7(a,b,c,d,e,f).
- Problems 15 and 17. The matrix in problem 15 is called a 'band matrix.' Do you see why?

§**7.3: System of Algebraic Equations** and **Linear Independence, Eigenvalues, Eigenvectors.** pp.353-363. This section is devoted to solving

$$\mathbf{Ax} = \mathbf{b}.$$

Go to Matlab, and type

help slash

to learn about the backslash operator and its application to solving this linear algebraic equation (and its relation to the inverse operator). The only complication arises when $\mathbf{A}$ is *singular* (*i.e.* when at least one of the eigenvalues is precisely zero). I will talk about it in class.

Again, in Matlab, type

help eig

and minutes later you are an expert in finding eigenvalues and eigenvectors! (I trust you did homework in Math 202 or 204 to do this chore long hand). Eigenvectors are helpful in diagonalizing a matrix!!!

- pp. 363-364, Problem 1: use the backslash.
- Problem 24: use Matlab's built-in function, eig.

§**7.4, Basic Theory** ... pp.365-368. Totally straightforward generalization from what you knew before. The Wronskian is here again. No problem assigned.

§**7.5, Homogeneous ..., pp.370-378;** §**7.6, Complex ..., pp.381-287;** §**7.7**, **Repeated ...,** pp.390-396. Until further notice, a Hermitian real matrix is simply a symmetric matrix. Physicists who work with quantum mechanics must deal with complex matrices all the time. In this course, we shall confine our attention to real matrices. But real matrices could have complex eigenvalues.

Matlab handles complex numbers in stride. Go to Matlab, say

$$z = 3 + 4i \qquad \text{or} \qquad z = 3 + 4j$$

and Matlab knows to interpret either 'i' or 'j' as the square root of minus one 'i'. You can ask for the square of z the normal way (as if z is real). You can get the complex conjugate of z by asking for conj(z) .

So long as you do not have repeated eigenvalues, everything is straightforward.

- pp. 378, Problem 17. Do this analytically. Doing it once, and you can do them all (assuming the eigenvalues are real and distinct).
- pp. 387, Problem 1. The eigenvalues are complex pairs.
- pp. 397, Problem 11. Where is this fellow Jordan? He appeared only in the footnote on page 344. I will give him his due credit in my lectures. Don't wait for Jordan (did you notice his 'form?'). Study the two examples to guide your work here.
- pp.396-398, Problem 13: Straightforward. Follow the example.

§**7.8, Fundamental Matrices, pp. 398-399;** §7.9, **Nonhomog**... pp. 405-411. In §7.8, Read enough to know what the Fundamental Matrix is (matrix notation is such a neat way to save a lot of writing!). In §7.9, we have three ways of doing things. First, diagonalize! and the rest is SO simple afterwards. Second, undetermine coefficients! needs inspirations (from a limited pools of candidates). And finally, variation of parameters again, using the fundamental matrix.

- pp.411-413, Problem 13.

As you can see, you are exercising your Matlab capabilities. You can now create vectors and matrices, transpose them, get the inverse of a non-singular square matrix, solve $\mathbf{Ax} = \mathbf{b}$ using the backslash, find the eigenvalues and eigenvectors of a square matrix using eig, and handle complex numbers. Everything is pretty straightforward, except when this Jordan thing comes along (when we have repeated eigenvalues).