

# Dilbert's problems

Anindya Chatterjee, Mechanical Engineering, IIT Kharagpur  
Please report any errors to anindya100@gmail.com

This version: March 12, 2012

## Abstract

Fifty numerically-oriented, skill-building problems in applied mathematics are offered for engineering graduate students. The intention is to solve slightly unfamiliar problems using widely available tools like Matlab and Maple. These problems may help students with computer programming in addition to nonlinear simultaneous equations, differential equations, boundary value problems, optimization, numerical integration and elementary probability. The student will usually have to think a little about how the problem should be set up for solution. Partial answers are provided.

## 1 Introduction

I have frequently wished for a set of small applied mathematics problems that might help a beginning engineering graduate student develop a few practical skills. I eventually put some together myself. I have used Matlab and Maple to get reasonable numerical answers, as indicated below.

## 2 Problem statements

1. Given  $\ddot{x} + 2.6\dot{x} + 7.9x^3 = 3.2 \sin(xt)$ , and  $x(0) = 1.2$ ,  $\dot{x}(0) = -3.3$ , find numerically the value of  $x(2.3)$ . Use `ode45` in Matlab. Experiment with error tolerances (`odeset`).
2. Assume  $a > 0$ . Given  $\ddot{x} + a\dot{x} + 7.9x^3 = 3.2 \sin(xt)$ , and  $x(0) = 1.2$ ,  $\dot{x}(0) = -3.3$ , and  $x(2.3) = -0.6$ , find numerically all the possible values of  $a$ , each accurate to at least 3 significant digits. Comment on negative solutions for  $a$ .
3. Write your own routine in Matlab for solving systems of nonlinear simultaneous equations using the Newton-Raphson method. Use finite differences to estimate the matrix of partial derivatives. Use your routine to solve the system of equations

$$x + y = 2, \quad x^2 + y^2 = 3.$$

4. Consider the matrix

$$\begin{bmatrix} 7 & a & 0 & 14 \\ -13 & 15 & 12 & 0 \\ i & 1+i & 0 & 0 \\ i & i & i & i \end{bmatrix},$$

and numerically seek real  $a$  for which the matrix has an eigenvalue with magnitude 2.6.

5. Same as above, only now ignore the magnitude and let the eigenvalue have an argument of  $37^\circ$  (see Matlab's `phase`).
6. Same as problem 4, only now seek complex values of  $a$  and plot an appropriate curve (or curves) on the complex plane.

7. Consider the matrix

$$\begin{bmatrix} 1 & a \\ 3 & 6 \end{bmatrix}.$$

Find  $a$  such that the acute angle between its eigenvectors is  $35^\circ$ .

8. Given  $\ddot{x} + 2.6\dot{x} + 7.9x^3 = 3.2 \sin(3.4xt)$ , and  $x(0) = a$ ,  $\dot{x}(0) = b$ , numerically seek  $a$  and  $b$  such that  $x(2.3) = 0.3$ . Plot a curve (or curves) in the  $a$ - $b$  plane.

9. Write your own routine in Matlab to find scalar integrals on an interval using Simpson's rule. Use a high enough degree of refinement to find, accurate to at least 3 significant digits,

$$\int_0^2 \sin\left(\frac{2x^3}{1 + \ln(1+x)}\right) dx.$$

10. Find 5 terms of a power series expansion, for small  $x$ , of

$$\frac{x^{0.7} \cos(1 + \ln(1+x))}{\sqrt{\sin(x)} \ln(1+x)}.$$

11. Using what you learnt from the last two problems find, to at least 3 significant digits,

$$\int_0^2 \frac{x^{0.7} \cos(1 + \ln(1+x))}{\sqrt{\sin(x)} \ln(1+x)} dx.$$

12. Find by integrating on the positive real line

$$\int_0^\infty \frac{\ln(1+x) \cos(1 + \sqrt{x})}{x^{0.6}} dx.$$

The integral converges slowly, so the contribution from  $x \in (A, \infty)$  for some large  $A$  should be computed from a suitable asymptotic approximation. Similarly consider a series for small  $x$ .

13. Using complex variables, reduce the previous integral to one on the imaginary axis. Compare results.

14. Given  $\ddot{x} + \lambda(2 + \cos t)x = 0$  and the boundary conditions  $x(0) = x(2.2) = 0$ , find the first five positive real values of  $\lambda$  for which *nonzero* solutions exist. All ODE solution should be done using Matlab's `ode45`.

15. For problem 14, set up a finite difference scheme in Matlab, and find the first five  $\lambda$  values using Matlab's `eig`.

16. Again for problem 14, assume a solution in the form of a Fourier sine series, and find the first five  $\lambda$  values using Maple.

17. Learn Floquet theory. For the system  $\ddot{x} + \sin(t)\dot{x} + \cos(t)x = 0$ , find the Floquet matrix (monodromy matrix).

18. For the system  $\ddot{x} + \sin(t)\dot{x} + a \cos(t)x = 0$ , seek real values of  $a$  such that the Floquet matrix has an eigenvalue of magnitude 5.

19. Analytically find the solution, for  $x \in (0, 3)$ , of

$$y'' + 0.3y' + 3y = \int_0^3 y dx, \quad y(0) = 0, \quad y(3) = 1.$$

You will need to solve an algebraic equation numerically.

20. Numerically solve, for  $x \in (0, 3)$ ,

$$y'' + 0.3y' + 3y^3 = \int_0^3 e^{-y} dx, \quad y(0) = y(3) = 0.$$

21. Numerically solve, for  $x \in (0, 3)$ ,

$$y''(x) + 0.3y'(x) + 3y(x) + \int_0^x s \cos y(s) ds = \int_0^3 e^{-y(s)} ds, \quad y(0) = y(3) = 0.$$

22. Consider

$$y''(x) + y'(x) + y(x) = \int_x^2 \sqrt{1+s} y(s) ds,$$

with initial conditions  $y(0) = 0$ ,  $y'(0) = 1$ . What is  $y(3)$ ?

23. Given that

$$y''(x) + \cos(x) y(x)^2 y'(x) + y(x)^3 = 1, \quad y(0) = 1,$$

and that there exists one or more real  $a$  such that  $y(a) = y'(a) = 0$ , find  $y$  and plot it for  $x$  between 0 and  $a$ .

24. Given that

$$\ddot{y} + 0.3z\dot{y} + 3y^3 = \sin t,$$

$$\dot{z} + yz = 0,$$

and

$$y(0) = 1, \quad y(1) = 0, \quad z(0.5) = 1,$$

find numerically  $y(t)$  and  $z(t)$  for  $t \in (0, 1)$ .

25. Consider the matrices

$$A = \begin{bmatrix} 1 & 2 & -3 \\ 1 & 3 & 0 \\ 0 & -2 & -3 \\ 1 & 0 & -7 \\ -3 & -1 & 2 \end{bmatrix}, \quad b = \begin{Bmatrix} 1 \\ 0 \\ 0 \\ 4 \\ -2 \end{Bmatrix}.$$

Find the column vector  $x$  which minimizes the error  $e = Ax - b$  in the 2-norm (or least squares sense).

26. Consider the function  $f(x) = \text{abs}(x)$  for  $x \in [-1, 2]$ . Using 300 equally spaced points on this interval, and considering polynomials of fourth order, approximate  $f(x)$  in the least squares sense. Show  $f$  and its approximation on a plot. Separately plot the approximation error. (Use problem 25.)

27. For the matrices in problem 25, consider  $x$  such that  $Ax \leq b$  elementwise. In addition,  $x \geq -6$  and  $x \leq 5$ , also elementwise. Maximize the sum of the first 2 elements of  $x$ . Use Matlab's `linprog`.

28. Reconsider problem 25. Now find the vector  $x$  which minimizes  $e$  in the infinite-norm (maximum error magnitude sense). (Problem 27 is relevant. You can define an auxiliary variable which bounds the errors using linear inequalities, and then minimize *that*.)

29. Reconsider problem 26, only now approximate the function in the maximum absolute error sense. Show  $f$  and its approximation on a plot. Separately plot the approximation error. (Use problem 27.)

30. Reconsider problem 27, only now minimize the sum of the squares of the first 2 elements of  $x$ . Use Matlab's `quadprog`.

31. Consider the van der Pol oscillator,  $\ddot{x} + \mu\dot{x}(x^2 - 1) + x = 0$ . The system has a unique stable periodic solution. Find the time period, accurate to at least 3 significant digits, for  $\mu = 3$ .

32. In problem 31, find  $\mu$  to at least 3 significant digits if the time period is known to be 8.

33. In problem 31, find the steady state RMS value of  $x$ .

34. Using a method of your choice that you implement in Matlab, plot a closed curve in the  $x$ - $y$  plane that satisfies

$$x^4 + 0.1x^3 + 2x^2 - x + y^4 - 0.1y^3 + \frac{y^2}{1+y^2} + 0.1xy = 100.$$

35. Numerically obtain the area and perimeter of the closed curve in problem 34.
36. Consider  $\ddot{x} + \dot{x}^5 + x = 0$ , with  $x(0) = 1000$  and  $\dot{x}(0) = 0$ . Guided by numerics, obtain an analytical estimate of the time taken for  $x$  to first attain the value 1.0.
37. Consider

$$p(x) = x^5 + 2x^4 + 3x^3 + bx^2 + ax + 6.$$

Let  $f_1$  be the largest of the real parts of the roots of  $p$ . Let  $f_2$  be the sum of the real parts of the roots of  $p$ . These are both functions of the real parameters  $a$  and  $b$ . Minimize  $g(a, b) = f_1 + 0.2f_2 + 0.1\text{abs}(a) + 0.1b^2$ . Use Matlab's `fminsearch`. Redo using contour plots in the  $a$ - $b$  plane.

38. Consider all  $2 \times 2$  complex matrices such that the absolute values of their elements add up to 1. What is the maximum absolute difference achievable between the eigenvalues of such a matrix?
39. Consider all  $3 \times 3$  complex matrices such that the absolute values of their elements add up to 1. The eigenvalues of such a matrix define a triangle on the complex plane. Let the area of the triangle be  $A$ . What is the maximum possible value of  $A$ ?
40. Reconsider problem 39 with all  $3 \times 3$  complex matrices, only now the magnitude of *each* element is 1. What is the maximum possible value of  $A$ ?
41. Let  $f(x) = e^{ax+\sqrt{1+x}} \cos(bx^2 + \ln(1+x))$ . Given that

$$\int_0^2 f(x) dx = 2 \quad \text{and} \quad \int_0^2 xf(x) dx = 0,$$

seek solutions for  $a$  and  $b$ .

42. Consider the matrix

$$A = \begin{bmatrix} 1 & 2 & -3 \\ 1 & 3 & 0 \\ a & -2 & -3 \end{bmatrix},$$

and suppose that we know  $e^a + \cos(2a)$  is an eigenvalue of  $A$ . Find possible values of  $a$ .

43. Use some combination of asymptotics and direct summation to find, accurate to at least 2 decimal places, the sum

$$\frac{\ln(2)}{1} + \frac{\ln(3)}{1^{1.1} + 1} + \frac{\ln(4)}{2^{1.1} + 1} + \frac{\ln(5)}{3^{1.1} + 1} \dots$$

44. Let  $A$  be a  $2 \times 2$  matrix whose elements are independent random numbers each uniformly distributed on  $(0, 1)$ , i.e., from Matlab's `rand`. What is the probability that the eigenvalues of  $A$  both lie inside the unit circle? What is the probability of the same if the elements of  $A$  are normally distributed with mean 0.5 and standard deviation 0.5? (Numerics only.)
45. Let  $\ddot{x} + a\dot{x} + 6bx + x^3 = 0$ , with  $x(0) = 1$  and  $\dot{x}(0) = 0$ , where  $a$  and  $b$  are independent random variables uniformly distributed on  $(0, 1)$ . What is the probability that  $|x(4)| < 0.2$ ?
46. Let  $X$  be a real random variable. The probability that  $X < x$  is known to be

$$F(x) = \frac{3}{10} \frac{e^{2x}}{1 + e^{2x}} + \frac{7}{10\pi} \tan^{-1}(2x^3 + 5) + \frac{7}{20}.$$

What is the expected value of  $X$ ?

47. Use Monte Carlo integration to evaluate

$$\int_0^1 \int_0^1 \int_0^1 \int_0^1 \int_0^1 \int_0^1 \ln(1 + x_1 + 2x_2 + 3x_3 + 4x_4 + 5x_5 + 6x_6) dx_1 dx_2 dx_3 dx_4 dx_5 dx_6.$$

48. Use Monte Carlo integration to evaluate

$$\int_0^1 \int_0^{x_6} \int_0^{x_5} \int_0^{x_4} \int_0^{x_3} \int_0^{x_2} \ln(1 + x_1 + 2x_2 + 3x_3 + 4x_4 + 5x_5 + 6x_6) dx_1 dx_2 dx_3 dx_4 dx_5 dx_6.$$

49. Use a Monte Carlo simulation to find the mean and standard deviation of the number of times we need to toss a coin in order to get 7 heads in a row.

50. Consider an equilateral triangle of side 3 units, lying in the  $x$ - $y$  plane with one vertex at the origin, and one edge along the  $x$ -axis. Remove the region inside the circle of radius 0.8 with centre at  $(1.0, 0.5)$ . Further remove the region inside the circle of radius 0.7 with centre at  $(1.0, 1.5)$ . Finally, remove all points below the curve  $y = 0.3x^2$ . Use Monte Carlo integration to find the remaining area.

### 3 Partial answers

1.  $x(2.3) = -0.7735841$ .
2. The ODE is nonlinear. What we can solve easily is an initial value problem for any given  $a$ . Large negative  $a$  might make numerical solutions unreliable, but let us first suppose that Matlab tolerances of `reltol` and `abstol` set to  $10^{-12}$  will give acceptable results. Tighter (smaller) values are often problematic anyway, because roundoff errors may begin to exceed truncation errors.

Given  $a$  and the initial conditions at  $t = 0$ , the computed value  $x(2.3)$  is just some function  $f(a)$ , which we can compute and plot, as shown below (figure 1). A horizontal line at  $-0.6$  is plotted for reference, and every intersection gives a solution for  $a$ . It is seen that there are infinitely many solutions for negative  $a$ , but a few easy ones for  $a > 0$ . In computing  $f(a)$ , it helps to define  $a$  as a global variable inside the derivative-evaluating m-file given to `ode45`. Then  $a$  can be set from the Matlab command window, or from any m-file, as needed.

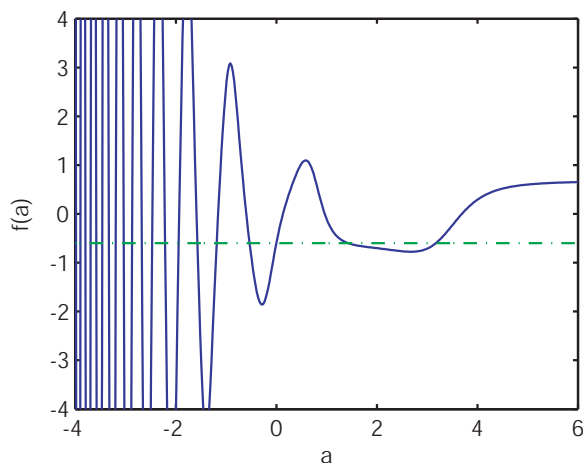


Figure 1: See problem 2.

3. Newton-Raphson method with numerically estimated Jacobians. You may adjust the termination tolerance, size of the finite difference step for estimating the derivative, and allowed number of iterations. `fun` is the name of the m-file which evaluates some vector function  $f(x)$ , where we seek  $x$  for which  $f(x) = 0$ .

```

function x=newton(fun,x)
% numerically estimates derivatives and implements Newton-Raphson method
n=length(x);
epsil=(1e-5*max(1,norm(x)));
pert=eye(n)*epsil;
iter=0; nmax=60;
ee=feval(fun,x);
while (norm(ee)*max(1,norm(x))>1e-10)*(iter<nmax)
    iter=iter+1;
    for k=1:n
        D(:,k)=(feval(fun,x+pert(:,k))-ee)/epsil;
    end
    x=x-(D\ee);
    if nargin == 3
        disp(x)
    end
    ee=feval(fun,x);
end
disp(iter), disp('iterations')
if (iter == nmax)+(abs(x)==inf)
    disp('did not converge')
end

```

Having `newton`, we make a new m-file as follows:

```

function z=p3(x)
z=[x(1)+x(2)-2; x(1)^2+x(2)^2-3];

```

Then, in Matlab, supplying an initial guess to `newton`, we get:

```

>> newton('p3',[0;1])
    5
iterations
ans =
    0.292893218812655
    1.707106781187345

```

4. Consider the following m-file:

```

function z=p4(a)
A=[7,a,0,14; -13,15,12,0; i,1+i,0,0; i,i,i,i];
z=abs(eig(A))-2.6;
[m,n]=min(abs(z));
z=z(n);

```

Note that the magnitude of the eigenvalue could be slightly below or above 2.6 (the desired value), and the sense of the difference should not be lost. So we do *not* want `abs( abs(eig(A)) - 2.6 )`.

Now, with `newton` as above, many random initial guesses fail to give solutions. However, after a few random inputs, I found the following two solutions:

$$-1.9470938, \quad -54.7383329.$$

5. Similar to the previous one. I got a few solutions:

$$-3.56597462, \quad 5.7292335, \quad 84.6838382.$$

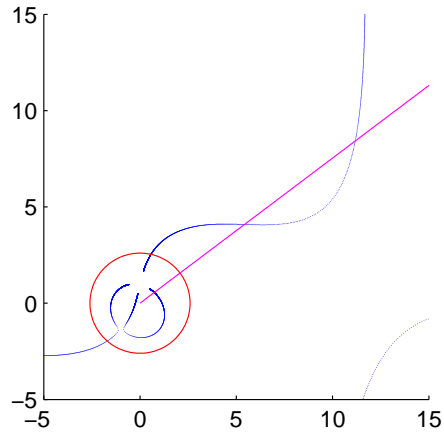


Figure 2: Eigenvalues on the complex plane: 4 curves for 4 eigenvalues. See problems 4 and 5.

You might like to plot all the eigenvalues, in the complex plane, as  $a$  is varied. Superimpose a circle of radius 2.6 (for the previous problem) and a line through the origin at  $37^\circ$  (for this problem). The resulting figure (figure 2) is consistent with the numerical results (2 solutions for problem 4 and 3 solutions for this one).

6. This problem involves numerical continuation of solutions. Since  $a$  is complex, let us write  $a = x + iy$  and note that we have two things to find. Given one solution, like  $-1.9470938 + 0i$ , we want to find a nearby solution (say at a distance of 0.01 on the complex plane). Then use that newly found solution to obtain yet another solution, a further distance of 0.01 away.

To this end, we define  $a_0 = -1.9470938$  to begin with, and use the following m-file:

```
function z=p6(x)
global a0
a=x(1)+i*x(2);
A=[7,a,0,14; -13,15,12,0; i,1+i,0,0; i,i,i,i];
z=abs(eig(A))-2.6;
[m,n]=min(abs(z));
z=z(n);
z=[z;norm(x-a0)-0.01];
```

Then, at the Matlab command,

```
>> a0=[-1.9470938;0];
>> a_new=newton('p6',a0+randn(2,1)/4)
    5
iterations
a_new =
   -1.944288035329471
   -0.009598316759364
>> plot(a0(1),a0(2),'.')
>> hold on
>> for k=1:100; a_guess=2*a_new-a0; a0=a_new; a_new=newton('p6',a_guess);
>> plot(a0(1),a0(2),'.'); end
```

The results obtained are shown in figure 3. Further results can be obtained in the same way.

7. Similar to problem 4. Note only that the dot product of normalized eigenvectors will give you the cosine of the included angle, and you should take the absolute value before you compare with  $\cos 35^\circ$ .

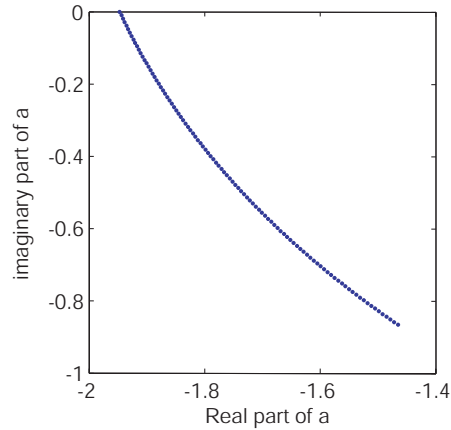


Figure 3: A series of points in the complex- $a$  plane that solve problem 4.

8. Similar to problem 6, except that the evaluation of the function involves numerical integration of a differential equation.
9. Simpson's rule:

```
function z=simp(a,b,N,fun)
% note: "fun" is a string; it is a file name where the integrand is
% evaluated; it should evaluate the function on arrays (be vectorized)
if floor(N/2)*2 == N
    N=N+1;
end
x=linspace(a,b,N); h=x(2)-x(1); y=feval(fun,x);
w=2*ones(1,N-2); w(1:2:end)=2*w(1:2:end); w=[1,w,1]*(h/3);
z=y*w';
```

The integrand:

```
function z=p9(x)
z=sin(2*x.^3./(1+log(1+x)));
```

In Matlab:

```
>> simp(0,2,101,'p9')
ans =
    0.427570776769777
>> simp(0,2,201,'p9')
ans =
    0.427571207725242
```

10. From MAPLE (partially evaluating to numerical values):

$$0.5403023059 x^{-4/5} - 0.5713198318 x^{1/5} - 0.2701511530 x^{6/5} + 0.2502233507 x^{11/5} - 0.2449287360 x^{16/5}$$

11. The above approximation is fine for direct analytical integration from  $x = 0$  to (say)  $x = 0.01$ . The remainder (from 0.01 to 2) can be done using Simpson's rule. There is some arbitrariness in choosing the limit of 0.01. The power series expansion shows that simply throwing away the integral from 0 to even  $10^{-10}$  would lead to an error in the second decimal place.



12. In this problem, there is a weak singularity at  $x = 0$ . Moreover, the domain of integration is infinite. The integrand decays slowly and oscillates with decreasing frequency. Thus, for direct evaluation on the real line, attention is needed both for small  $x$  and large  $x$ .

As shown above, we can use a power series expansion for small  $x$  and evaluate the integral to some suitable small positive value  $\epsilon$ . I get (from MAPLE)

$$\frac{5}{7} \cos(1)\epsilon^{\frac{7}{5}} - \frac{10}{19} \sin(1)\epsilon^{\frac{19}{10}} - \frac{5}{12} \cos(1)\epsilon^{\frac{12}{5}} + \frac{20}{87} \sin(1)\epsilon^{\frac{29}{10}} + \frac{25}{136} \cos(1)\epsilon^{\frac{34}{10}} - \frac{17}{156} \sin(1)\epsilon^{\frac{39}{10}} + \dots,$$

which for  $\epsilon = 0.05$  gives (for the terms shown) 0.0041938.

We also need an approximation for  $A \gg 1$  in

$$f(A) = \int_A^\infty \frac{\ln(1+x) \cos(1+\sqrt{x})}{x^{0.6}} dx.$$

Here, note that

$$\frac{\cos(1+\sqrt{x})}{2\sqrt{x}}$$

is a perfect derivative. So we let  $u = 1 + \sqrt{x}$  to obtain

$$f(A) = \int_{1+\sqrt{A}}^\infty \frac{2 \ln(1+(u-1)^2) \cos u}{(u-1)^{0.2}} du. \quad (1)$$

We could expand

$$h(u) = \frac{2 \ln(1+(u-1)^2)}{(u-1)^{0.2}} \quad (2)$$

for large  $u$  to obtain some further analytical simplification, but it is unnecessary. Integrating by parts in Eq. 1 (see Eq. 2), we obtain

$$f(A) \sim (h(u) \sin(u) + h'(u) \cos u - h''(u) \sin u - h'''(u) \cos u + \dots) \Big|_{1+\sqrt{A}}^\infty.$$

A small trick we can use is to let  $1 + \sqrt{A}$  be  $n\pi$  for a large positive integer  $n$ , so that the sine terms are zero in the expansion above. Using MAPLE to compute five derivatives of  $h$ , we obtain a long expression that decays rather slowly, such that its asymptotic behavior is clearly established for something like  $u > 400\pi$  (a visual estimate from a graph not reproduced here). The corresponding value of  $A$  is inconveniently large:

$$A = 160000\pi^2 + 1.$$

So we do the following (taking an even larger  $A$  for the asymptotic behavior):

$$f(100) \approx \int_{11}^{1200\pi} \frac{2 \ln(1+(u-1)^2) \cos u}{(u-1)^{0.2}} du - 0.000132.$$

Evaluating the integral using Simpson's rule (using 1000000 points, after some convergence checks), we have

$$f(100) \approx 5.8422584.$$

We are finally left with one integral to get our final approximation, which is:

$$0.0041938 + \int_{0.05}^{100} \frac{\ln(1+x) \cos(1+\sqrt{x})}{x^{0.6}} dx + 5.8422584,$$

which (using Simpson's rule again) is

$$\int_0^\infty \frac{\ln(1+x) \cos(1+\sqrt{x})}{x^{0.6}} dx \approx -1.3703.$$

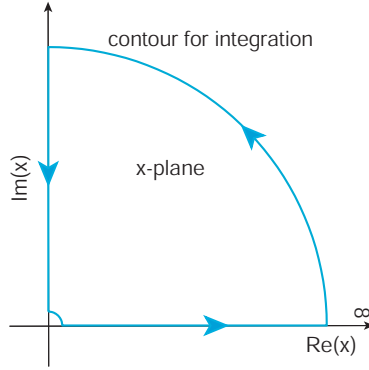


Figure 4: Contour for problem 13.

13. The above evaluation of the integral was laborious. Now we will use contour integration in the complex plane. To this end, we rewrite the integrand as

$$\operatorname{Re} \left( \frac{\ln(1+x)e^{i(1+\sqrt{x})}}{x^{0.6}} \right).$$

The real part can of course be taken after evaluation of the integral; and the evaluation of the integral can use a suitable contour in the complex plane.

Consider the contour in the first quadrant, shown in figure 4, to be traversed counterclockwise. There is a small circular arc near the origin, which makes a negligible contribution. Similarly, the outer circle contributes terms that decay exponentially as its radius grows to infinity. There being no singularities within the contour, the contour integral is zero. The net result is that the integral from zero to infinity along the real line is the same as that from zero to infinity along the positive imaginary axis. Having identified the line integral needed, we can again take real parts before integration, obtaining

$$\int_0^\infty \frac{\ln(1+x) \cos(1+\sqrt{x})}{x^{0.6}} dx = \int_0^\infty \operatorname{Re} \left( \frac{ie^{-0.6\frac{\pi}{2}i} \ln(1+iy)e^{i(1+\sqrt{iy})}}{y^{0.6}} \right) dy.$$

Since evaluation of the integral is numerical, the integrand need not be simplified further. Taking three portions on the  $y$  axis (somewhat arbitrarily from  $10^{-13}$  to  $10^{-3}$ ,  $10^{-3}$  to  $10^3$ , and from  $10^3$  to  $10^6$ ), with about 2 million points each time (takes less than a second anyway), Simpson's rule gives the above integral as

$$\int_0^\infty \frac{\ln(1+x) \cos(1+\sqrt{x})}{x^{0.6}} dx \approx -1.37016.$$

I believe this approximation is much more accurate than the previous one.

14. In this problem, since the boundary conditions are zero, the solution  $x$  can be multiplied by any scalar quantity. It is clear that  $\dot{x}(0) \neq 0$  (because the solution of the initial value problem would then give  $x \equiv 0$ ). By these two observations, we may take  $x(0) = 0$  and  $\dot{x}(0) = 1$ . It remains to use `ode45` to compute  $x(2.2)$ , which may be viewed as a function of  $\lambda$  (call it  $f(\lambda)$ ). We can now easily compute and plot  $f(\lambda)$ . Results are shown in figure 5. Accurate  $\lambda$ -values can be obtained using the Newton-Raphson method.
15. Problem 14 can be solved using finite differences. Taking  $N$  uniformly spaced mesh points with  $t_k - t_{k+1} = h$ ,  $t_0 = 0$  and  $t_{N+1} = 2.2$ ; and  $x_0 = x_{N+1} = 0$ ; we have

$$\frac{x_{k+1} - 2x_k - x_{k-1}}{h^2} = -\lambda(2 + \cos t_k)x_k,$$

which leads to a generalized eigenvalue problem of the form

$$Ax = \lambda Bx.$$

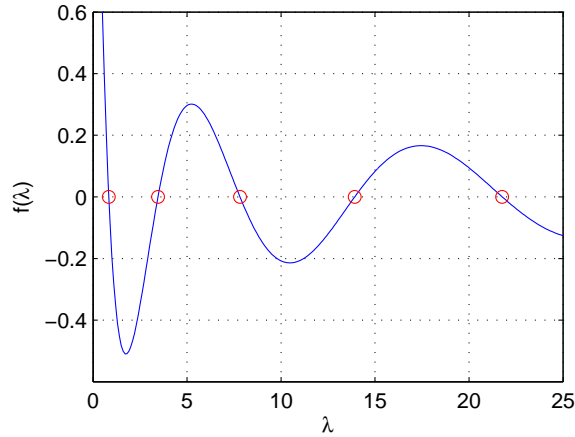


Figure 5: Figure for problem 14. Intersections with the horizontal axis give the required  $\lambda$ -values. The circles were computed as per problem 15.

I used the Matlab code

```
function L=p15(N)
t=linspace(0,2.2,N+2); h=t(2)-t(1); t=t(2:end-1);
A=-2*eye(N); A(2:end,1:end-1)=A(2:end,1:end-1)+eye(N-1);
A(1:end-1,2:end)=A(1:end-1,2:end)+eye(N-1);
A=A/h^2;
B=-diag(2+cos(t));
L=eigs(A,B,5,'sm');
```

For  $N = 500$ , I got the eigenvalues (see also figure 5)

0.8378, 3.4544, 7.8148, 13.9183, 21.7650.

16. Now with Maple. Assume

$$x = \sum_{k=1}^N a_k \sin\left(\frac{k\pi t}{2.2}\right),$$

substitute into the ODE, multiply sequentially by each sine term and set the integral to zero, to obtain  $N$  linear homogeneous coupled equations involving a parameter  $\lambda$ . Then find  $\lambda$  values such that the coefficient matrix is singular.

The following lines of Maple code (using 6 sine terms)

```
N:=6: x:=add(a[k]*sin(k*Pi*t/(22/10)),k=1..N): de:=expand(diff(x,t,t)+lambda*(2+cos(t))*x):
for k from 1 to N do eq[k]:= evalf(integrate(de*sin(k*Pi*t/(22/10)), t=0..22/10)); end:
with(linalg): A:=matrix(N,N): for k from 1 to N do for n from 1 to N do
A[k,n]:=coeff(eq[k],a[n]): end: end: dd:=det(A): evalf(solve(dd,lambda));
```

give the following results for the first 5 eigenvalues (with increasing, but acceptable, inaccuracy):

0.8378, 3.4545, 7.8151, 13.9193, 21.7849.

17. Writing the system in first order form using

$$q = \begin{Bmatrix} \dot{x} \\ x \end{Bmatrix},$$

the Floquet matrix is computed to be

$$\begin{bmatrix} 1 & 0 \\ 21.6237 & 1 \end{bmatrix}.$$

18. Directly evaluating the Floquet multipliers for various values of  $a$ , we can plot the results to see some solutions as in figure 6. The curves with dots are the multipliers, and the horizontal line is at 5. There are 8 solutions within the range  $-5 \leq a \leq 5$ .

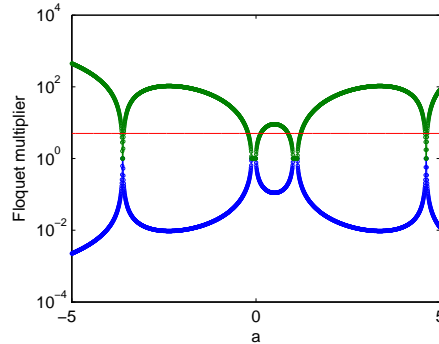


Figure 6: Floquet multipliers (log scale).

19. The point is that although  $y$  varies with  $x$ ,  $\int_0^3 y dx$  is just some number. Call it  $a$ . Then we are really solving

$$y'' + 0.3y' + 3y = a, \quad y(0) = 0, \quad y(3) = 1,$$

with  $a$  as yet undetermined. The solution is analytically easy to find (if somewhat long), and we write it is (say)  $f(a, x)$ . Now we set  $\int_0^3 f(a, x) dx = a$  as originally assumed, and find that  $a = 2.743245$ . Substituting back into  $f(a, x)$ , we obtain

$$y(x) = 0.9144 - 0.6080 e^{-0.1500x} \sin(1.7255x) - 0.9144 e^{-0.1500x} \cos(1.7255x).$$

20. See the previous problem. Now we set  $\int_0^3 e^{-y} dx = a$ , and consider the system of ODEs

$$\begin{aligned} y'' + 0.3y' + 3y^3 &= a, \\ z' &= e^{-y}. \end{aligned}$$

We now integrate these using initial conditions  $y(0) = 0$ ,  $y'(0) = b$  and  $z(0) = 0$ . The aim is to adjust  $a$  and  $b$  so that  $y(3) = 0$  and  $z(3) = a$ . The following Matlab code works:

```
function zdot=p20(t,z)
global a
zdot=[-0.3*z(1)-3*z(2)^3+a; z(1); exp(-z(2))];
```

and

```
function z=p20a(x)
global a
op=odeset('reltol',1e-10,'abstol',1e-10);
a=x(1); b=x(2);
[t,z]=ode45('p20',[0,3],[b;0;0],op);
z=z(end,:);
z=[z(2); z(3)-a];
```

Using the Newton-Raphson method to find a zero of the vector function  $\mathbf{p20a}$ , we obtain many solutions. Some of the  $a$ -values obtained are

$$3.1085, \quad 4.1565, \quad 5.8713, \quad 8.5398, \quad 9.5796, \quad 13.2869.$$

The corresponding  $y(x)$  are plotted in figure 7.

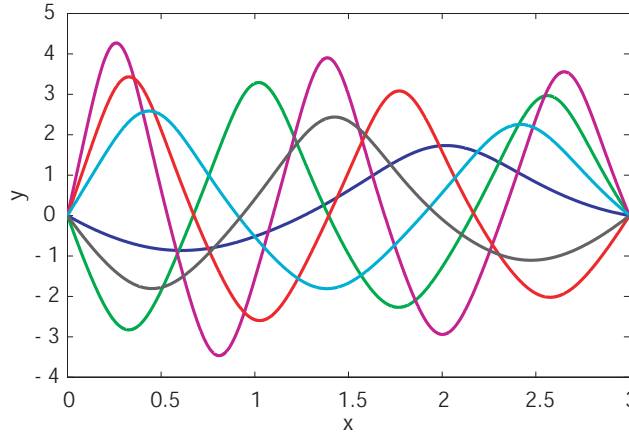


Figure 7: Some solutions for problem 20.

21. This is similar to problem 20 except you need to add on a third differential equation, as follows

$$\begin{aligned} y'' + 0.3y' + 3y + \xi &= a, \\ z' &= e^{-y}, \\ \xi' &= x \cos y. \end{aligned}$$

Now start with initial conditions  $y(0) = 0$ ,  $y'(0) = b$ ,  $z(0) = 0$ ,  $\xi(0) = 0$ . Iteratively adjust  $a$  and  $b$  such that  $z(3) = a$  and  $y(3) = 0$ . Using the Newton-Raphson method, I found

$$a = 2.2136, \quad b = -1.2309.$$

22. Rewrite the problem as

$$y''(x) + y'(x) + y(x) = \int_0^2 \sqrt{1+s} y(s) ds - \int_0^x \sqrt{1+s} y(s) ds.$$

Setting

$$\int_0^2 \sqrt{1+s} y(s) ds = a,$$

solve the system

$$\begin{aligned} y''(x) + y'(x) + y(x) &= a - z, \\ z' &= \sqrt{1+x} y, \end{aligned}$$

with initial conditions  $y(0) = 0$ ,  $y'(0) = 1$  and  $z(0) = 0$ , from  $x = 0$  to  $x = 2$ . Now adjust  $a$  such that  $z(2) = a$ . Finally, knowing  $a$ , integrate one last time to  $x = 3$ . I found  $a = -28.8458$  and  $y(3) = -10.0262$ .

23. This is a second order ODE. The initial condition  $y(0)$  is given, but  $y'(0)$  must be chosen along with  $a$  to satisfy the two conditions  $y(a) = y'(a) = 0$ . Many solutions exist; some are shown (for both positive and negative  $a$ ) in figure 8.

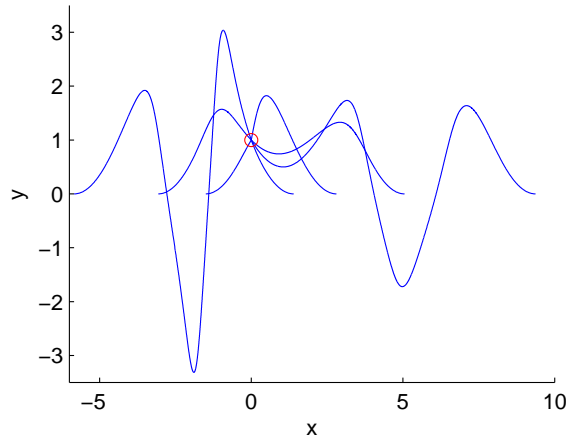


Figure 8: Some solutions for problem 23. The common point  $(0, 1)$  is marked with a circle.

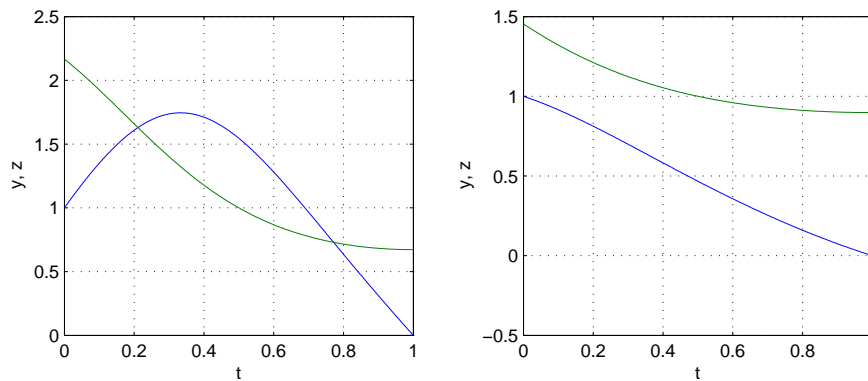


Figure 9: Two solutions for problem 24 (I found others as well). The curves for  $y$  and  $z$  may be distinguished by noting  $z(0.5) = 1$ .

24. In order to solve an initial value problem, we need  $y(0)$  (which we have) along with  $\dot{y}(0)$  and  $z(0)$  (which we do not have). These two have to be adjusted so that  $y(1) = 0$  and  $z(0.5) = 1$ . Multiple solutions exist. Two are shown in figure 9.
25. This is simply  $A \setminus b$  in Matlab. (When  $A$  is square and invertible, it computes the unique solution. When the system is overdetermined as in this problem, it computes the least squares solution.)
26. The following lines in Matlab work. See figure 10.

```
x=linspace(-1,2,300)'; f=abs(x); A=[ones(size(x)),x,x.^2,x.^3,x.^4];
c=A\f; subplot(1,2,1); plot(x,f,x,A*c,'-.'); legend('abs(x)', 'approximation');
xlabel('x'); ylabel('abs(x) and polynomial approximation'); subplot(1,2,2);
plot(x,f-A*c); xlabel('x'); ylabel('approximation error');
```

27. The sum of the first 2 elements turns out to be 3.3333.
28. Impose  $Ax - b \leq c$ , where we understand  $Ax - b$  is a vector, each element of which is less than some scalar quantity  $c$ . Further impose  $Ax - b \geq -c$ , understood similarly. Now minimize  $c$  using `linprog`. I found  $c = 0.85$ . I used the following Matlab program (written compactly for space).

```
function x=infnormsol(A,b)
[n,m]=size(A); f=[1;zeros(n+m,1)]; op=optimset('maxiter',2000);
```

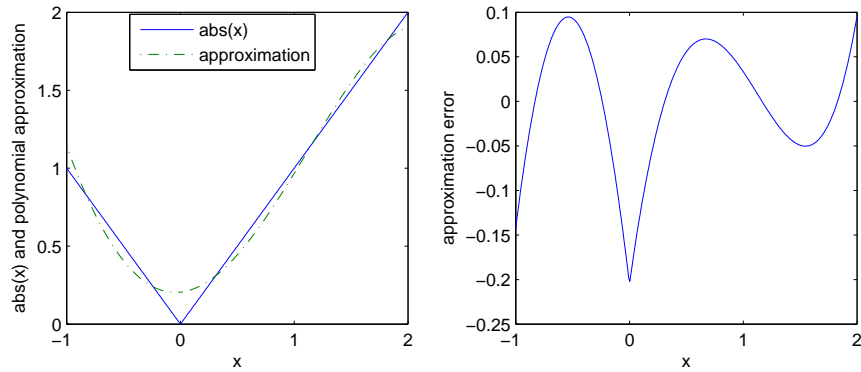


Figure 10: Least squares approximation using a polynomial: problem 26.

```
AA=[zeros(n,1),-eye(n),A]; B=zeros(2*n+1,n+m+1); B(:,1)=-ones(2*n+1,1);
B(2:n+1,2:n+1)=eye(n); B(n+2:2*n+1,2:n+1)=-eye(n); z=zeros(2*n+1,1);
x=linprog(f,B,z,AA,b,[],[],[],op); x=x(n+2:end);
```

29. In approximating  $\text{abs}(x)$  using 300 equally spaced points on  $[-1, 2]$ , a fourth order polynomial, and with the goal of minimizing the maximum error, the maximum error was 0.1245 (compare with figure 10 right, where the maximum error is about 0.2).
30. Try `help quadprog` in Matlab. Take  $H$  to be the  $3 \times 3$  diagonal element with diagonal elements  $(2, 2, 0)$ ; take  $f$  to be the  $3 \times 1$  zero matrix, and  $A$  and  $b$  as already defined. Then `quadprog` gives the answer  $x = \{0.9, -0.3, 0.2\}^T$ .
31. Since there is a periodic solution and the system is autonomous, assume  $\dot{x}(0) = 0$ . If the time period is  $T$ , then on solving the initial value problem with initial conditions  $\dot{x}(0) = 0$  and  $x(0) = a$  (`ode45`), the solution must satisfy  $x(T) = a$  and  $\dot{x}(T) = 0$ . Iteratively adjust  $a$  and  $T$  (Newton-Raphson). I found  $a = 2.0233$  and  $T = 8.8591$ . (A warning: Matlab already has built-in functions called `vdp` and `vdp1`. Avoid clashes.)
32. Very similar to the previous problem, except  $T = 8$  (fixed), and we must adjust  $\mu$  and  $a$ . I found  $\mu = 2.3157$  and  $a = 2.0217$ . I found it helps to have a rough idea in advance about where the solution lies.
33. The periodic solution directly gives the steady state behavior. Upon solving problem 31 you know  $T$ . Now add on a differential equation,  $\dot{z} = x^2$ , and integrate it along with the van der Pol equation with  $z(0) = 0$ , for exactly one period ( $T = 8.8591$ ). Divide by  $T$  and take the square root.
34. See problem 6 for an elementary method of numerical continuation. To use that, you need a starting point. Set  $y = 0$  and solve for  $x$ . Since it is a polynomial in  $x$ , use Matlab's `roots`. There are 4 roots, of which 2 are complex. The real ones are 3.0106 and  $-3.0059$ . Start with the point  $(3.0106, 0)$ , use an arclength of 0.01 (something small but manageable), and find some  $N$  points to begin with. It turns out 2146 points are needed, with the last one slightly overshooting  $y = 0$ . We can drop that last point and insert the starting point in its place to make sure the loop is complete. The resulting plot is shown in figure 11.
35. Perimeter: sum the distances between successive points for a piecewise linear approximation. I got 21.44. Area: think of each line segment along the curve as the base of a triangle with its vertex at the origin. Taking the position vector of each point as  $\mathbf{r}$  and the corresponding little line segment as  $d\mathbf{r}$ , sum  $0.5 |\mathbf{r} \times d\mathbf{r}|$ . I got 34.82.
36. The numerical solution shows an almost linearly varying, non-oscillating solution for large  $x$ . In the equation  $\ddot{x} + \dot{x}^5 + x = 0$ , when  $x$  is large, it has to be balanced by another large term. Being balanced by  $\ddot{x}$  would imply  $\dot{x}^5$  is relatively small, and require oscillatory solutions (not seen). This suggests  $\ddot{x}$  is

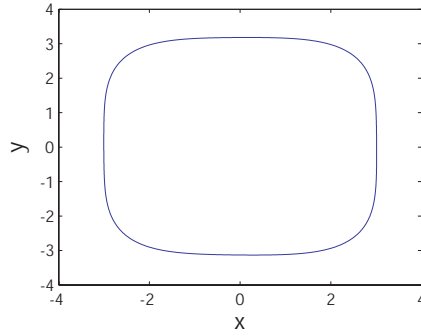


Figure 11: Closed curve plot using continuation: problem 34.

negligible, and  $\dot{x}^5 \approx -x$ . Solving the approximating equation, the time taken for  $x$  to become 1 is about 312.7 units. The match with numerics is good.

37. Using `fminsearch` I find  $a = 0$ ,  $b = 0.21042$ , and  $g = 0.29028$ . The contour plot shows a matching minimum (figure 12).

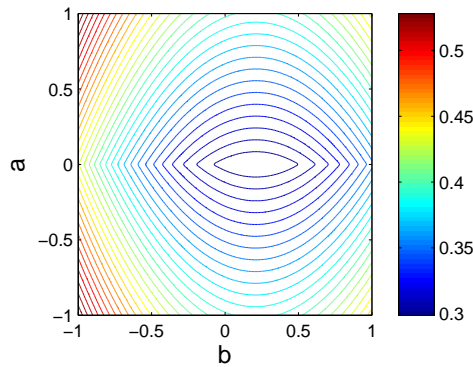


Figure 12: Contour plot of  $g$  for problem 37.

38. Before we begin, we note that the matrix

$$A = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$$

has elements whose magnitudes add up to 1, and the eigenvalues (0,1). Thus, the maximum distance achievable between eigenvalues is at least 1. We will see below that the maximum in fact appears to be 1 (since we use numerical optimization, we will not have a *proof*).

We will use `fminsearch`. How to define the objective function? The sum of absolute values must add up to 1, and this could be used as an explicit constraint. However, if  $A$  is a square matrix and  $a$  is a number, and if  $\lambda$  is an eigenvalue of  $A$ , then  $a\lambda$  is an eigenvalue of  $aA$ . And so we can directly minimize (note the minus sign)

$$g = -\frac{|\lambda_1 - \lambda_2|}{\sum_{i,j} |A_{ij}|}.$$

Now the elements of  $A$  could be complex, so we presently seem to have 8 independent variables. But the first element  $A_{11}$  can be assumed real. This is because if it is complex with phase  $\theta$ , then its phase can be set to zero by multiplying  $A$  with some  $a = e^{-i\theta}$ . This multiplication leads to a rotation of the eigenvalues about the origin in the complex plane, but does not affect their separation. So we seem to have 7 independent variables. However, scaling up all the elements of  $A$  by any real multiplier  $a$  does



not change the objective function  $g$ , as discussed above, so we might as well take  $A_{11} = 1$ , leaving 6 independent variables.

Try the following Matlab function with `fminsearch`:

```
function z=eig_prob_1(x)
% assume x is 6X1 and real, and convert to 3X1 and complex.
x=x(1:3)+i*x(4:6); x=[1;x];
x=[x(1:2),x(3:4)]; z=eig(x); z=-abs(z(1)-z(2))/sum(sum(abs(x)));
```

The minimized objective function turns out to be  $-1$ , but the minimizing  $x$  is very large and changes from run to run. I believe this is because many redundancies remain in the optimization problem formulation. For example, similarity transformations ( $A \rightarrow P^{-1}AP$ ) will leave the eigenvalues unchanged but might affect the sum of the absolute values of the elements of the matrix; and so every optimal  $A$  must have the property that  $\sum_{i,j} |A_{ij}|$  cannot be reduced by any similarity transformation (the optimization could conceivably be carried out over the subset of matrices that possess this property). But I do not know how to use this idea to reduce the number of explicit independent variables.

For our present purposes, the problem as originally stated is solved, and the maximum possible distance is unity.

Comment: In such simple-minded uses of `fminsearch`, I usually try several runs with random initial guesses, and in each case take the offered optimal  $x$  as a new initial guess to run it again, several times in a row. Such persistence does not ensure finding the true optimum, but improves the odds.

39. Let the matrix be called  $B$ . Based on our (my) inability in the previous problem to reduce the number of independent variables to the minimum possible value, let us here choose simplicity and allow every element to be complex and unrestricted; the constraint of  $\sum_{i,j} |B_{ij}| = 1$  will be implicitly incorporated later. Upon computing the 3 eigenvalues ( $\lambda_1, \lambda_2, \lambda_3$ ), we compute 2 complex numbers, namely  $e_1 = \lambda_1 - \lambda_2$  and  $e_2 = \lambda_1 - \lambda_3$ . These two complex numbers may be also viewed as vectors in the plane, and the magnitude of their cross product is twice the area of the triangle. Alternatively, we can rotate the vector  $e_1$  through 90 degrees by multiplying with  $i$ , and then take the dot product of the two vectors ( $ie_1$  and  $e_2$ ). At this point,  $\sum_{i,j} |B_{ij}| \neq 1$ . Scaling the elements of  $B$  uniformly with a factor  $\alpha$  will scale the eigenvalues by  $\alpha$

and the area  $A$  by  $\alpha^2$ . And so the computed area  $A$  should be divided by  $\left(\sum_{i,j} |B_{ij}|\right)^2$ .

Accordingly, consider the following Matlab program:

```
function z=eig_prob_2(x)
% input has 18 elements, convert to 9 complex elements
x=x(1:9)+i*x(10:18); x=[x(1:3),x(4:6),x(7:9)];
z=eig(x); z=[i*(z(1)-z(2));z(1)-z(3)];
z=-abs( prod(z)* cos(phase(z(1))-phase(z(2))) )/(sum(sum(abs(x))))^2/2;
```

As it did for the previous problem, `fminsearch` gives different optimizers for different random initial guesses. But it settles frequently on the value (dropping the minus sign)  $0.14433 = \frac{1}{4\sqrt{3}}$ . The optimal number may be obtained by considering a diagonal matrix, with the diagonal elements equal to one third of each of the cube roots of unity. There is a local optimum at 0.125 which is frequently offered by `fminsearch`, but persistence yields the correct result.

40. For this problem, we can think of each element as  $e^{i\theta}$ , suggesting 9 independent inputs. The number could be reduced, but is not necessary with `fminsearch`. The optimal value obtained (largest area) appears to be  $\frac{9\sqrt{3}}{4}$ .
41. A straightforward use of Simpson's rule and Newton's method, both already covered. Try the Matlab function:

```
function z=p41(q)
N=5001; a=q(1); b=q(2);
x=linspace(0,2,N); h=x(2)-x(1);
w=2*ones(1,N-2); w(1:2:end)=2*w(1:2:end); w=[1,w,1]*(h/3);
y=exp(a*x+sqrt(1+x)).*cos(b*x.^2+log(1+x)); z=y*w'-2;
y=x.*y; z=[z;y*w'];
```

Newton's method gives many solutions. A few are:

$$(a, b) = (-0.2219, 0.3493), \quad (2.8845, 5.1847), \quad (1.0313, -2.5893), \quad \text{and} \quad (2.6977, -5.7385).$$

42. Matlab works fine with complex numbers, and so we can use the following program:

```
function z=p42(a)
% a could be complex
A=[1,2,-3;1,3,0;a,-2,-3]; B=(exp(a)+cos(2*a))*eye(3); z=det(A-B);
```

Trying Newton's method with real initial guesses keeps all iterates real, and gives some real solutions. I found the following two:  $-0.8370$  and  $1.5728$ . But there are infinitely many complex roots as well. Starting with a few random complex initial guesses, I found the following (there are infinitely many more):

$$a = 1.3602 - 1.0887i, -1.5208 + 1.0333i, -4.6718 + 1.1726i.$$

Complex conjugates of the above are roots as well.

43. The series theoretically converges. However, direct summation is hopeless. The sum is on the order of

$$\int_0^{\infty} \frac{\ln(x+2)}{1+x^{1.1}} dx \approx 101 \quad (\text{from Maple}).$$

The numerical sum to  $10^6$  terms is 41.5, and the sum to  $10^7$  terms is 49.2. How many terms will we need for the error to be about 0.01? The answer  $N$  is given by

$$\sum_{k=N}^{\infty} \frac{\ln(k+2)}{1+k^{1.1}} \approx 0.01.$$

Approximately,

$$\int_N^{\infty} \frac{\ln(x+2)}{1+x^{1.1}} dx \approx \int_N^{\infty} \frac{\ln(x)}{x^{1.1}} dx = 10 \frac{10 + \ln N}{N^{1/10}} \approx 0.01.$$

Substituting  $N = 10^p$ , we have

$$10 \frac{10 + p \ln 10}{10^{p/10}} = 0.01.$$

The solution is impossibly large:  $p \approx 51$ .

An asymptotic approximation can be found as follows. Let  $N$  be large (it will become clear how large).

Define  $f(x) = \frac{\ln(x+2)}{1+x^{1.1}}$ . Consider

$$\int_N^{\infty} f(x) dx = \sum_{k=N}^{\infty} \int_k^{k+1} f(k+u) du.$$

Since  $f(x)$  and its derivatives decay successively faster for large  $x$ , we can write

$$\int_N^{\infty} f(x) dx = \sum_{k=N}^{\infty} \int_k^{k+1} \left\{ f(k) + f'(k)u + \frac{f''(k)}{2}u^2 + \dots \right\} du,$$

whence

$$\sum_{k=N}^{\infty} f(k) = \int_N^{\infty} f(x) dx - \frac{1}{2} \sum_{k=N}^{\infty} f'(k) - \frac{1}{6} \sum_{k=N}^{\infty} f''(k) - \dots$$

Applying the above formula for the first sum on the right hand side, we obtain

$$\sum_{k=N}^{\infty} f'(k) = \int_N^{\infty} f'(x) dx - \frac{1}{2} \sum_{k=N}^{\infty} f''(k) - \dots = -f(N) - \frac{1}{2} \sum_{k=N}^{\infty} f''(k) - \dots,$$

or

$$\sum_{k=N}^{\infty} f(k) = \int_N^{\infty} f(x) dx + \frac{1}{2}f(N) + \frac{1}{12} \sum_{k=N}^{\infty} f''(k) + \dots = \int_N^{\infty} f(x) dx + \frac{1}{2}f(N) - \frac{1}{12}f'(N) + \dots.$$

Thus, the desired sum may be approximated as

$$\sum_{k=0}^{\infty} f(k) \approx \sum_{k=0}^{N-1} f(k) + \int_N^{\infty} f(x) dx + \frac{1}{2}f(N) - \frac{1}{12}f'(N).$$

In the above, since  $N$  is large, the integral itself is perhaps best evaluated after expanding the integrand in a series for large  $x$  (using Maple). The result, for  $N = 100$ , is 92.146. Thus

$$\sum_{k=0}^{\infty} f(k) \approx \sum_{k=0}^{99} f(k) + 92.1458 + \frac{1}{2}f(100) - \frac{1}{12}f'(100) = 9.1793 + 92.1458 + 0.0145 - 0.00002 \approx 101.34.$$

44. The following Matlab program can be used, with  $N$  larger if you want a more accurate result.

```
function [p1,p2]=p44
N=1e6; s1=zeros(N,1); s2=s1;
for k=1:N; A=randn(2); s1(k)=max(abs(eig(A)))<1; A=randn(2)*.5+.5;
s2(k)=max(abs(eig(A)))<1; end
p1=sum(s1)/N; p2=sum(s2)/N;
```

I found  $p_1 \approx 0.50$  and  $p_2 \approx 0.46$ .

45. This problem just requires repeated ODE solution. Use the following Matlab program to define the ODE system.

```
function zdot=p45(t,z)
global a b
x=z(1); xdot=z(2);
zdot=[xdot;-a*xdot-6*b*x-x^3];
```

Then, at the Matlab prompt, try

```
global a b
op=odeset('reltol',1e-6,'abstol',1e-6); s=zeros(1000,1);
for k=1:1000; a=rand; b=rand; [t,y]=ode45('p45',[0,4],[1;0],op); s(k)=abs(y(end,1))<0.2; end
mean(s)
```

I got 0.415 and 0.395 on 2 consecutive runs. With 10000 trials instead of 1000, I got 0.4112 and 0.4085 on 2 consecutive runs. I suggest 0.41 may be a reasonable estimate of the required probability.

46. The probability density function of  $X$  is  $p_X(x) = F'(x)$ . The expected value of  $X$  is

$$\int_{-\infty}^{\infty} xp(x) dx \approx -0.8453 \quad (\text{from Maple}).$$

In the above integral (numerically evaluated), it is sufficient to plot the function and choose a finite range for the integral, because  $p$  decays rapidly for large  $x$ .

47. First find the correct answer using Maple. Define

$$f = \ln(1 + x_1 + 2x_2 + 3x_3 + 4x_4 + 5x_5 + 6x_6).$$

Then successively evaluate

$$f_1 = \int_0^1 f dx_1,$$

$$f_2 = \int_0^1 f_1 dx_2,$$

etc. The expressions involved become long. The final integrand, a function of  $x_6$  alone, has 147 terms. Writing  $x$  in place of  $x_6$ , the first few terms are (directly from Maple)

$$\begin{aligned} & \frac{992}{15} x \ln(2) + \frac{21}{4} \ln(1 + 6x)x^3 + \frac{259}{96} \ln(1 + 6x)x + \frac{43}{8} \ln(1 + 6x)x^2 + \frac{198}{5} \ln(2) - \frac{78651}{1600} \ln(3) - \frac{137}{60} \\ & - \frac{130361}{3600} \ln((3 + 2x)^{-1}) - \frac{128}{15} x \ln(4 + 3x) + \frac{16384}{225} \ln(8 + 3x) - \frac{15093}{160} \ln(3)x - \frac{128}{15} \ln((1 + 2x)^{-1})x + \dots \end{aligned}$$

The integral finally is

$$\begin{aligned} & -\frac{26006}{675} \ln(2) - \frac{49}{20} - \frac{1058841}{6400} \ln(7) - \frac{12717}{100} \ln(3) - \frac{844375}{10368} \ln(5) + \frac{12400927}{57600} \ln(11) \\ & - \frac{4826809}{518400} \ln(13) + \frac{24137569}{518400} \ln(17) = 2.41104. \end{aligned}$$

The Monte Carlo integral proceeds as follows. We generate  $N$  points uniformly distributed on the 6-dimensional cube defined by  $x_k \in (0, 1)$ ,  $k = 1, 2, \dots, 6$ . The joint probability density function of these points is some constant  $p$ . Requiring

$$\int_0^1 \int_0^1 \int_0^1 \int_0^1 \int_0^1 \int_0^1 p dx_1 dx_2 dx_3 dx_4 dx_5 dx_6 = 1,$$

we find  $p = 1$  in this case. Being interested in

$$\int_0^1 \int_0^1 \int_0^1 \int_0^1 \int_0^1 \int_0^1 f(x_1, x_2, x_3, x_4, x_5, x_6) dx_1 dx_2 dx_3 dx_4 dx_5 dx_6,$$

we note that it is the same as

$$\int_0^1 \int_0^1 \int_0^1 \int_0^1 \int_0^1 \int_0^1 pf(x_1, x_2, x_3, x_4, x_5, x_6) dx_1 dx_2 dx_3 dx_4 dx_5 dx_6,$$

which in turn is just the expected value of  $f$ , and may be estimated by averaging the  $N$  values of  $f$ . Thus we have the Matlab code:

```
N=1e6; x=rand(N,6); f=1; for k=1:6 f=f+k*x(:,k); end; f=mean(log(f))
```

which estimates the integral as (three random attempts) 2.4111, 2.4113, and 2.4115. In general, the error is inversely proportional to  $\sqrt{N}$ .

48. Maple evaluates the integral. It is

$$-\frac{11573523}{2464000} \ln(3) - \frac{82707247}{17496000} \ln(7) - \frac{49}{14400} + \frac{47045881}{19595520} \ln(19) + \frac{19351049}{21049875} \ln(2) + \frac{1771561}{637875} \ln(11) \approx .0036398.$$

To evaluate the integral using a Monte Carlo method, we can use logical variables as follows. Instead of evaluating

$$\int_0^1 \int_0^1 \int_0^1 \int_0^1 \int_0^1 \int_0^1 f(x_1, x_2, x_3, x_4, x_5, x_6) dx_1 dx_2 dx_3 dx_4 dx_5 dx_6,$$

we evaluate

$$\int_0^1 \int_0^1 \int_0^1 \int_0^1 \int_0^1 \int_0^1 f(x_1, x_2, x_3, x_4, x_4, x_6) g(x_1, x_2, x_3, x_4, x_4, x_6) dx_1 dx_2 dx_3 dx_4 dx_5 dx_6,$$

where

$$g(x_1, x_2, x_3, x_4, x_4, x_6) = (x_1 < x_2)(x_2 < x_3)(x_3 < x_4)(x_4 < x_5)(x_5 < x_6),$$

where in turn the right hand side is a product of logical variables (each takes the value 1 if the inequality holds, and 0 otherwise). The following Matlab code can be used ( $N = 2 \times 10^7$ ):

```
N=2e7; m=6; x=rand(N,m); f=1; for k=1:m f=f+k*x(:,k); end; f=log(f);
for k=1:m-1 f=f.*(x(:,k)<x(:,k+1)); end; s=std(f), F=mean(f)
```

It gives

$$s = 0.098, \quad F = 0.003633.$$

The standard deviation is useful in that the random estimate of the integral (here  $F$ ) is approximately normally distributed with mean equal to its correct value and standard deviation equal to  $s/\sqrt{N} \approx 2 \times 10^{-5}$ , consistent with the exact result above.

49. The following Matlab program simulates a series of coin tosses until 7 heads in a row are obtained, and returns the number of throws needed.

```
function N=p49
% how many coin tosses to get 7 heads in a row?
% let heads=1 and tails=0, with 50% probability each
% use 400 throws at a time
x=[]; n=400; head7=0;
while 1-head7
x=[x;(rand(n,1)>0.5)];
p=x(1:end-6);
for k=1:6
    p=p.*x(1+k:end+k-6);
end
if any(p)
    head7=1; [m,N]=max(p); N=N+6;
end
end
```

Running this many times (30000 times), we get a sample and compute the mean (253.8) and standard deviation (247.9).

Comment: Those expecting an approximation to the exponential distribution (wherein mean and median are identical) may note the difference of about 6 between the mean and the standard deviation: I assume this is because one attempt to get 7 heads in a row requires 7 tosses, which adds 6 to the mean.

50. First write a program which, given a point  $(x, y)$ , will determine whether it lies inside the region of interest. Here is such a program:

```
function z=p50(X)
% to determine whether a given point (x,y) is inside the region of interest
y=X(2); x=X(1); z=1;
if y<0, z=0; end % if below x-axis, then outside
c=polyfit([0,3/2],[0,3*sqrt(3)/2],1);
if y>polyval(c,x), z=0; end % if above first inclined edge, then outside
c=polyfit([3/2,3],[3*sqrt(3)/2,0],1);
```

```

if y>polyval(c,x), z=0; end % if above second inclined edge, then outside
if norm(X-[1,.5])<0.8, z=0; end % if inside first circle, then outside
if norm(X-[1,1.5])<0.7, z=0; end % if inside second circle, then outside
if y < 0.3*x^2, z=0; end % if under parabola, then outside

```

In slightly formal notation, let the region of interest be  $R$ , and define

$$\chi(R; x, y) = \begin{cases} 1 & \text{if } (x, y) \in R, \text{ and} \\ 0 & \text{otherwise.} \end{cases}$$

Then if  $\Omega$  is any portion of the  $x$ - $y$  plane that completely includes  $R$ , then the integral of  $\chi$  over  $\Omega$  equals the area of  $R$ . Choose  $\Omega$  to be a rectangle of length 3 units and height  $\frac{3\sqrt{3}}{2}$  units, with one long side along the  $x$ -axis and its bottom left corner at the origin. The area of the rectangle is  $\frac{9\sqrt{3}}{2}$  units, and so random points uniformly distributed on this rectangle have the constant joint probability density function  $p = \frac{2}{9\sqrt{3}}$ . The expected value of  $\chi(R; x, y)$  is

$$\iint_{\Omega} p\chi \, dx dy = p \iint_{\Omega} \chi \, dx dy = p \times \text{Area of } R.$$

Thus, using 10000 points in Matlab:

```

figure; hold on; x=rand(10000,2); x(:,1)=x(:,1)*3; x(:,2)=x(:,2)*3*sqrt(3)/2;
s=0; for k=1:10000; t=p50(x(k,:)); if t, plot(x(k,1),x(k,2),'.'); end; s=s+t; end

```

generates a figure showing the retained points and returns a value of  $s$ . I got  $s = 813$ , whence the expected value of  $\chi$  is  $813/10000$ , and the area of  $R$  is estimated to be

$$\frac{813}{10000} \times \frac{9\sqrt{3}}{2} = 0.63.$$

The region, with a parabola and two circles drawn in to show excluded points, is shown in figure 13.

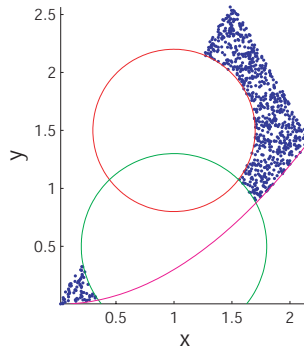


Figure 13: Points retained for problem 50.

Comment: Using 300000 points in place of 10000, I got 0.614 (I do not trust the third digit). We could use a shorter rectangle, e.g., by computing the intersection between the second (right-most) inclined edge of the triangle and the parabola  $y = 0.3x^2$ . I avoided the complication for easier presentation.

## Acknowledgements

Sovan Das and Atanu Mohanty provided useful comments. Additionally, some errors were pointed out by Atanu Mohanty and Saurabh Biswas.