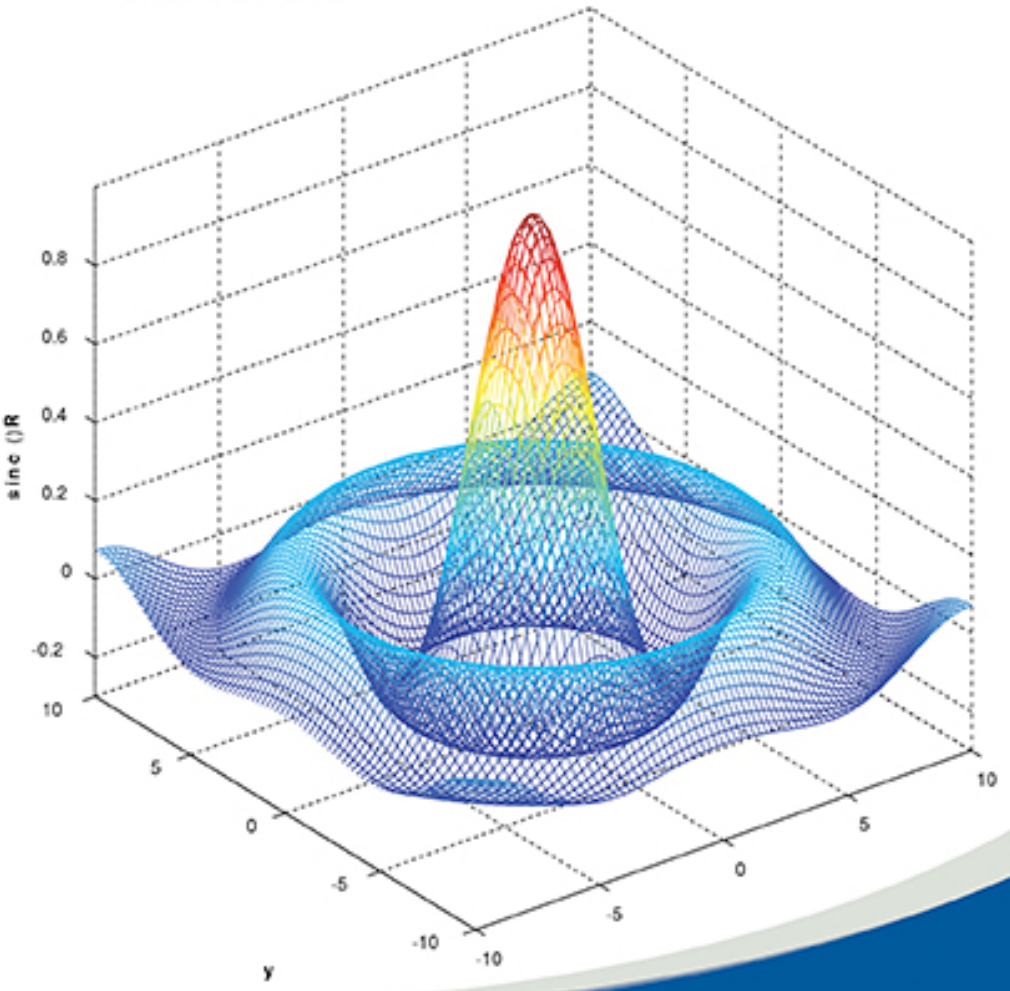
ONE HUNDRED PHYSICS VISUALIZATIONS USING MATLAB

Dan Green



World Scientific

ONE HUNDRED PHYSICS VISUALIZATIONS USING MATLAB

This page intentionally left blank

ONE HUNDRED PHYSICS VISUALIZATIONS USING MATLAB

Dan Green
Fermi National Accelerator Laboratory, USA

World Scientific
NEW JERSEY • LONDON • SINGAPORE • BEIJING • SHANGHAI • HONG KONG • TAIPEI • CHENNAI

Published by

World Scientific Publishing Co. Pte. Ltd.
5 Toh Tuck Link, Singapore 596224
USA office: 27 Warren Street, Suite 401-402, Hackensack, NJ 07601
UK office: 57 Shelton Street, Covent Garden, London WC2H 9HE

British Library Cataloguing-in-Publication Data
A catalogue record for this book is available from the British Library.

ONE HUNDRED PHYSICS VISUALIZATIONS USING MATLAB (With DVD-ROM)

ISBN 978-981-4518-43-7
ISBN 978-981-4518-44-4 (pbk)

Printed in Singapore by Mainland Press.

"The soul should always stand ajar, ready to welcome the ecstatic experience."

— **Emily Dickinson**

"The effort to understand the Universe is one of the very few things that lifts human life above the level of farce, and gives it some of the grace of tragedy."

— **Steven Weinberg**

This page intentionally left blank

Preface

"Computers are useless. They can only give you answers."
— **Pablo Picasso**

"The purpose of computing is insight, not numbers."
— **Richard Hamming**

There are only a very few solvable problems in physics. They are extremely useful because the equations for the solutions can be plotted and the parameters defining the solutions can be varied in order to explore the dependence of the solutions on the variables of the problem. In that way the student can build up an intuition about the Kepler problem, for example.

However, this can only be done in a few cases and even then the effort needed is tedious. For the others, numerical methods are needed and the computation becomes somewhat cumbersome. As a result, it is more difficult to vary the inputs to the problem numerically rather than symbolically and develop an intuition about the dependence of the solution on those parameters. In particular time development is often obscure and "movies" can be a welcome tool in improving physical intuition.

Nevertheless, the advent of powerful personal computing has considerably reduced the difficulties. Indeed, the aim of this book is to use the ensemble of symbolic and numeric tools available in the MATLAB suite of programs to illustrate representative numerical solutions to more than one hundred problems spanning several

physics topics. The student typically works through the demonstration and alters the inputs through a menu driven script. In that way the user driven menu allows for parametric variation.

MATLAB is a good vehicle for the computational tasks. It has a compiler, editor and debugger which are very useful and user friendly. The HELP utility is very extensive. The MATLAB language is similar to a modern C++ language and it is vectorial/matrix which makes coding simpler than older languages such as FORTRAN. Data is easily imported and exported in a variety of formats.

MATLAB contains many special functions. Matrices and linear algebra are covered well. Curve fitting, polynomials and fast Fourier transforms are supplied. Numerical integration packages are available. Differential equations, symbolic, ordinary and partial, as well as numerical solutions are available for both initial value and boundary value versions.

As an additional package, MATLAB has symbolic mathematics. Within that package, calculus, linear algebra, algebraic equations and differential equations are covered. It is easy to combine a symbolic treatment of a problem with a numerical display of the solution when that is desirable. In this way converting from symbols to numbers is easily achieved.

Finally, and very importantly, MATLAB has an extensive suite of display packages. One can make bar, pie, histogram and simple data plots. There are several contour and surface plots which are possible. The time evolution of solutions can be made into "movies" that illustrate the speed of a process. These extensive visualization tools are crucial in that the student can plot, vary and then re-plot. There are two- and three-dimensional plots of all types available. Complex as well as real data can be shown.

The aim of using these tools is to create intuition, not to solve a specific problem or to complete a specific number crunching exercise. Indeed, the aim of the text is not to teach physics but to give the user a sense of how the solutions of a given physics problem depend on the parameters of that problem and to show the connections between, say, wave optics and quantum mechanics.

The script for these demonstrations is made available. Using that material the student can write his/her own additions and explorations with the supplied scripts as jumping off points. In this way, a path is available to extend well beyond the specific demonstrations enclosed in the book itself, making the search for further possible insights open ended.

This page intentionally left blank

Contents

This page intentionally left blank

Chapter 1

Symbolic Mathematics and Math Tools

"If people do not believe that mathematics is simple, it is only because they do not realize how complicated life is."

— **John von Neumann**

"Pure mathematics is, in its way, the poetry of logical ideas."

— **Albert Einstein**

"There cannot be a language more universal and more simple, more free from errors and obscurities ... more worthy to express the invariable relations of all natural things [than mathematics]. [It interprets] all phenomena by the same language, as if to attest the unity and simplicity of the plan of the universe."

— **Joseph Fourier**

1.1. MATLAB Functions

The first section will deal with mathematical tools, mostly using the MATLAB symbolic math package. Although not strictly physics, the tools of mathematics are crucial because they are the language of physics and physics cannot be understood well without a facility in that language. Some of these tools will be invoked later in the more physics oriented demonstrations found in the following sections.

MATLAB has a very large suite of special functions which are available to the user. They can be found for the MAPLE symbolic functions, by invoking the command "mfunlist". The first page of symbolic functions is shown in Figure 1.1.

The complete set of MATLAB functions is available using the HELP tab in the Command Window. The sequence is HELP/ MATLAB/functions. There are ten headings under functions and by using them all, the MATLAB functions are available for examination. There are two other useful headings, examples and demos, which give useful aid in understanding some applications of these functions.

```
MATLAB has several very useful symbolic functions
  mfunlist Special functions for MFUN.
    The following special functions are listed in alphabetical order
    according to the third column. n denotes an integer argument,
    x denotes a real argument, and z denotes a complex argument. For
    more detailed descriptions of the functions, including any
    argument restrictions, see the documentation of the active symbolic engine

 bernoulli      n       Bernoulli Numbers
 bernoulli      n,z     Bernoulli Polynomials
 BesselI        x1,x    Bessel Function of the First Kind
 BesselJ        x1,x    Bessel Function of the First Kind
 BesselK        x1,x    Bessel Function of the Second Kind
 BesselY        x1,x    Bessel Function of the Second Kind
 Beta           z1,z2   Beta Function
 binomial       x1,x2   Binomial Coefficients
 EllipticF -    z,k     Incomplete Elliptic Integral, First Kind
```

Figure 1.1: First entries for MATLAB symbolic functions.

Specific descriptions and examples follow when the help function is explicitly queried. One of the strengths of MATLAB is that there are so many supported special functions and that they are described using the help files. These utilities make the use of these functions quite transparent. An example is given in Figure 1.2, where the path to drill down the function tree to "acos" is shown.

A full list is invoked using the .m script "MATLAB_Functions" in the Command Window, which gives the complete list of symbolic functions and also prints the path to retrieve all the MATLAB numerical special functions. Tools that are useful with symbolic math are: "sym", "factor", "simplify", "pretty", "simple" and "eval". These tools can be used to simplify the symbolic strings and "eval" is used to convert them for numerical evaluations.

1.2. Symbolic Differentiation

A first demonstration of the use of symbolic math is to evaluate derivatives. As with most of the demonstrations in this text, there is a recurring format. First, explanatory text is printed by invoking "help" in the script, an example is given, and then there is a menu driven prompt which asks the user to try other functions or additional options. The example is plotted in order to see the result

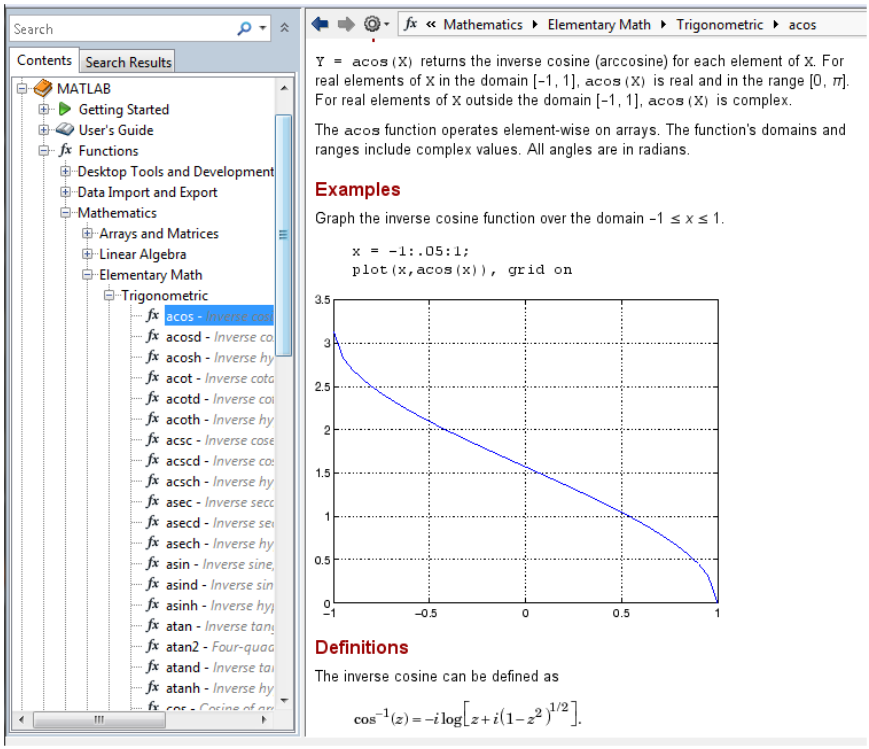


Figure 1.2: Result of using the help facility to find the description of the acos function.

of the operations. The MATLAB script "diff" is the core of the script "SM_Diff". The plot of the printout of the example is shown in Figure 1.3, while the function and derivative is shown in Figure 1.4. Most of the exercises have explanatory printout as the initial response to starting the specific script. The format of the script used in this text is made as uniform as possible within the different physics being explored so as to make the script easy to use, understand, and ultimately be modified by the users to follow their interests.

1.3. Symbolic Integration

A similar script performs symbolic integration, with an example followed by possible user input functions with resulting plots. The

```
Symbolic Math Differentiation: enter f(x), + - * / ^ cos sin tan sqrt,x a b n
 An Example, Plot Function and Derivative

fin =

sin(x^2)


           2
  2 x cos(x )
```

Figure 1.3: Printout for the symbolic differentiation script.

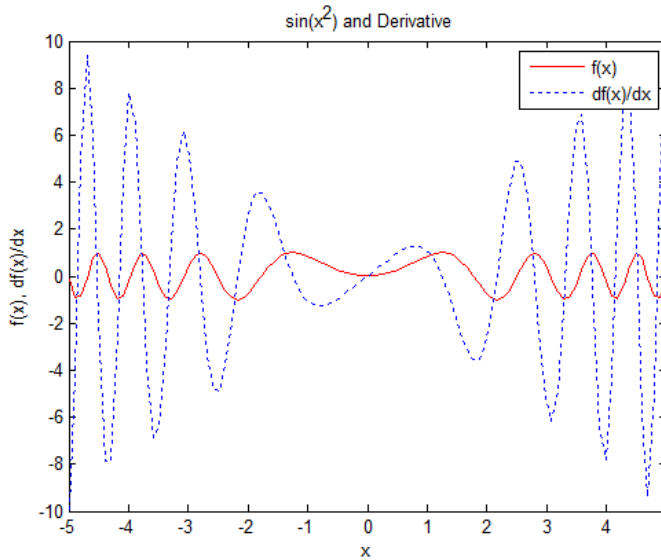


Figure 1.4: Plot of the example function and the derivative. Other functions can be input by the user in symbolic form as often as is desired in one session of the script use.

example provided is shown in Figure 1.5. The script, "SM_Int" is set up to perform indefinite integrals. However, MATLAB has other options using the "int" script, such as defining the variable not to be x, but a user defined variable or supplying the limits for a definite integral. The reader is encouraged to further explore the available options in MATLAB should they be interested in using the "int" function in greater depth. Invoking "help int" in the Command Window yields examples and options for symbolic

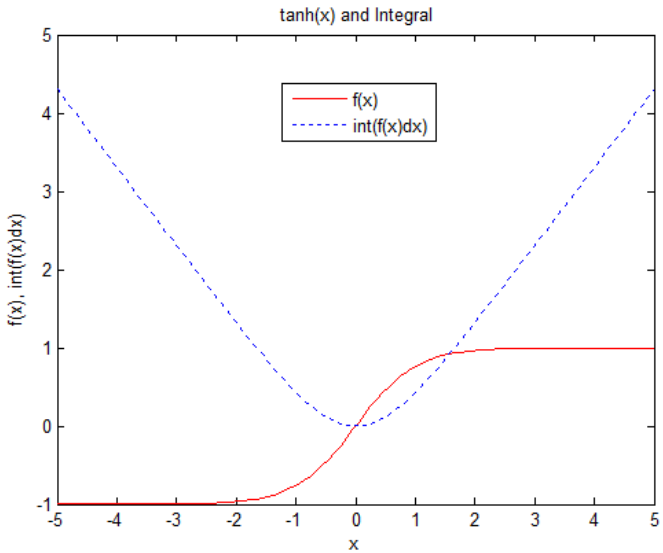


Figure 1.5: Example for the symbolic integration of $\tanh(x)$ as an indefinite integral.

integration. In particular, all of the symbolic functions indicated in Figure 1.1 are available as integrand candidates.

1.4. Taylor Expansion

Power law expansions are useful tools in several applications. They are available using the MATLAB function "taylor", where the expansion point and the number of terms in the series can be selected. The printout from the script "SM_Taylor" for the example and the menu for the user are shown in Figure 1.6. The plotting output for the example expansion is given in Figure 1.7. Clearly, an intuition can be built up fairly quickly as to the domain of validity of the expansion as to how well it approximates the function. It is clear that the expansion of the cos function with five terms is a fair representation for $|x| < 2$. As with other MATLAB scripts, a full description is available from the Command Window via the help query or using the help tab.

The user chooses the function, the number of terms, and the offset, or expansion point in the variable x. The results are displayed

```
Symbolic Math Taylor Expansion: enter f(x),
  An Example

fin =

cos(x)


   4    2
  x    x
  -- - -- + 1
  24   2
 Enter f(x): exp(x)
 Enter Number of Terms: 6
Enter a, Expansion About x = a: 0

   5     4    3    2
  x     x    x    x
  --- + -- + -- + -- + x + 1
  120   24   6    2
```

Figure 1.6: Printout for the "SM_Taylor" script with the example of $\cos(x)$ and the menu choice of $\exp(x)$.

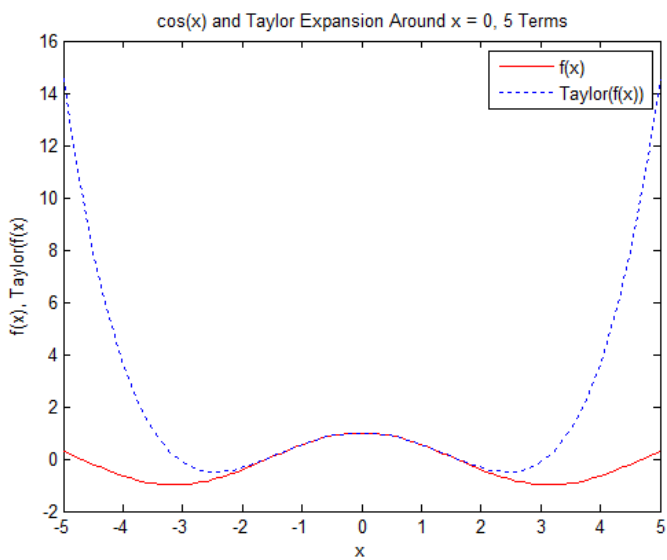


Figure 1.7: Example from the "SM_Taylor" script for the Taylor expansion of $\cos(x)$.

graphically in the figure. That figure can be printed or edited using the Figure Window and the editing tabs supplied in that window.

1.5. Series Summation

There is the ability in MATLAB to symbolically sum a series using the "symsum" script. This script has been used with a wrapper script called "SM_Sum_Series". An example and a user input series is shown in Figure 1.8. As always, a user driven menu is provided so that any series can be input and summed. More details and other examples can be accessed with the input "help symsum" in the Command Window.

```
>> SM_Sum_Series
  Symbolic Math - Series Summation


Symbolic Math Series Summation: enter f(x), + - * / ^ cos sin tan sqrt,x
  An Example, x^k k=0 to inf

          /                    1              \
  piecewise| Inf if 1 <= x, - ----- if |x| < 1 |
          \                  x - 1            /
 Other Examples; k , k^2 and k ^3 from 1 to n, also 1/k^2 from 1 to inf
 Enter Term of the Series f(k): 1/k^2
Indefinite = 1 or Definite = 0: 0
 Enter Starting Index: 1
 Enter Ending Index: inf

    2
  pi
  ---
   6
```

Figure 1.8: Output of the series summing script for an example and then for a user defined series with the terms being possibly definite or indefinite.

1.6. Polynomial Factorization

Polynomials can be factorized symbolically. The script "factor" is used in the wrapper script, "SM_Factor" and an example with user input is shown in Figure 1.9. Numbers can also be factored.

1.7. Equation Solving

The MATLAB script "solve" performs the symbolic solution of a set of equations with multiple variables. The initial dialogue for the

```
Symbolic Math Polynomial Factorization: enter f(x), + - * / ^ x, y
  An Example

fin =

x^2 - y^2


fact =

(x - y)*(x + y)

 Enter Polynomial to Factor: x^9 - 1

fact =

(x - 1)*(x^2 + x + 1)*(x^6 + x^3 + 1)
```

Figure 1.9: Printout of the initial dialogue when the factorization script is used and an example of user input using the menu provided.

"SM_SolveEq" wrapper script is shown in Figure 1.10. This example illustrates a simple quadratic solution of a single equation. Much more complex problems are easily solvable. Extensive use of "solve" and related scripts will be made later on in the text.

```
>> SM_SolveEq
  Symbolic Math - Solve an Equation


Symbolic Math Solution of Equation: enter f(x), + - * / ^ cos sin tan
  An Example, Solves f(x) = 0

fin =

a*x^2 + b*x + c


  +-                          -+
  |           2          1/2  |
  |    b + (b  - 4 a c)       |
  |  - -------------------    |
  |             2 a           |
  |                           |
  |           2          1/2  |
  |    b - (b  - 4 a c)       |
  |  - -------------------    |
  |             2 a           |
  +-                          -+
```

Figure 1.10: The result of the example provided by the "SM_SolveEq" script showing the symbolic solution for a single quadratic equation.

1.8. Inverse Functions

The MATLAB script, "finverse" provides a symbolic solution for the inverse of a function. A wrapper is provided; "SM_Finverse" which gives an example and asks for a user defined input. The printout is shown in Figure 1.11 and the plot provided for a user chosen function is shown in Figure 1.12.

```
Symbolic Math Inverse Function: enter f(x), + - * / ^ cos sin tan sqrt
  An Example

fin =

1/tan(x)


     / 1 \
  atan| - |
     \ x /
 Enter f(x): cos(x^2)

        1/2
  acos(x)
```

Figure 1.11: Dialogue on inverse functions. The user choice was $\cos(x^2)$ while the example was $1/\tan(x)$.

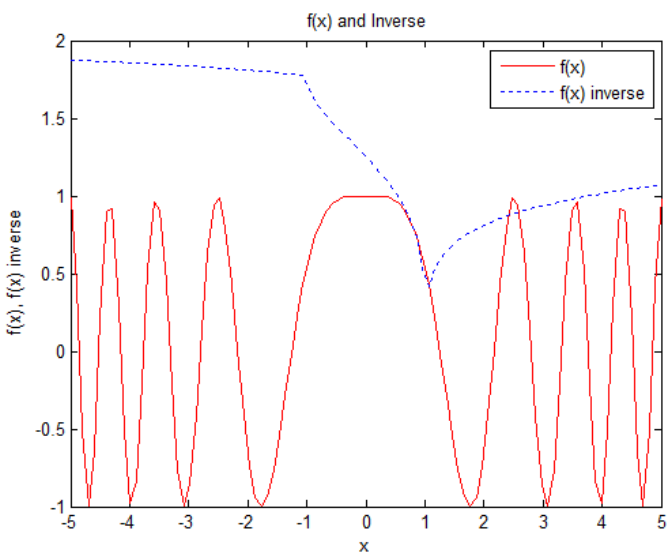


Figure 1.12: User defined input function, $\cos(x^2)$ and the inverse, shown as a function of x.

1.9. Matrix Inversion

In solving eigenvalue and eigenvector problems in physics, it is required to evaluate the inverse of a matrix. The MATLAB script "inv" is available and can be used to evaluate the inverse of any matrix symbolically. In particular, the wrapper script "Matrix_Inv" can be used. A general 2×2 matrix example is shown in Figure 1.13. Note that in MATLAB, for many operations, one need not specify the indices. MATLAB is a matrix language, hence the name. For example $C = A * B$, yields the product matrix of the matrices A and B.

```
Matrix Inversion, 2 x 2 and 3 x 3

b =

[ b11, b12]
[ b21, b22]


ans =

[  b22/(b11*b22 - b12*b21), -b12/(b11*b22 - b12*b21)]
[ -b21/(b11*b22 - b12*b21),  b11/(b11*b22 - b12*b21)]
```

Figure 1.13: Printout of the general matrix inverse for a 2×2 symbolic matrix.

1.10. Matrix Eigenvalues

Matrix operations are integral to many problems in physics. Fortunately, MATLAB is a matrix language which simplifies much of the coding of indices which is needed with older languages. There is also a suite of matrix scripts available which enables the user to evaluate matrix quantities. The provided wrapper script is "SM_Eigen" which uses the MATLAB functions "det", "inv" and "eig". An example printout for the case of a symbolic rotation matrix is shown in Figure 1.14.

MATLAB utilities will be used for matrices in several applications later in the text including determinant, inverse, eigenvalues and eigenvectors.

1.11. Ordinary Differential Equations

Ordinary differential equations and systems of such equations may be solved symbolically using the MATLAB script "dsolve". Some

```
Symbolic Matrices: Eignevalues and Eigenvectors
  An Example - Rotation by x
A =

[ cos(x), -sin(x)]
[ sin(x),  cos(x)]

Determininat of A

  1
Inverse of A

  +-                  -+
  |   cos(x), sin(x)   |
  |                    |
  | -sin(x), cos(x)    |
  +-                  -+
Eigenvalues of A, Diagonals

  +-                      -+
  |  exp(-x i),     0      |
  |                        |
  |      0,      exp(x i)  |
  +-                      -+
```

Figure 1.14: Rotation matrix, the determinant, inverse matrix, and eigenvalues of that matrix.

examples are given in the script "SM_ODE2", where equations familiar in physics are given as a menu choice, as seen in Figure 1.15. The printout for a specific case is displayed in Figure 1.16, where a Taylor expansion is shown using the script to evaluate the expansion of the

Figure 1.15: Menu for the choice of a differential equation to solve for in "SM_ODE2".

```
>> SM_ODE2
  Symbolic Math - Simple Ordinary Diffferential Equations, Up to Second Order


Symbolic Math ODE - a Few Solvable Examples

    g exp(-k t) - g + g k t
  - -----------------------
                2
               k
x(t) for x(0)=0, Dx(0) = 0, free fall in uniform gravity field, velocity force k
 Taylor Expansion of x

       2  4        3      2
    g k  t    g k t    g t
  - ------- + ------ - ----
      24         6       2
v(t)

  g exp(-k t)   g
  ----------- - -
       k        k
```

Figure 1.16: Symbolic solution of the equation of a particle falling in a uniform gravity field with a velocity dependent dissipative force, both the exact solution and a Taylor expansion for the position $x(t)$. At large times a terminal velocity g/k is reached.

position as a function of time. Much use will be made of "dsolve" in situations in physics where a closed form of the solution is possible.

In the script "SM_ODE3", the user can arbitrarily choose the ordinary differential equation to explore and can either define initial conditions on the function or the derivative of the function of not. A simple example of the use of the script is shown in Figure 1.17, in the case of the choice of simple harmonic motion with unspecified initial position and velocity. In this case, there are integration constants C2 and C3 in the solution which will need to be evaluated.

1.12. Fourier Series

Fourier series is a powerful tool that is used to understand how large are the frequency components which are needed to sufficiently approximate an arbitrary waveform. In fact, any function can be synthesized using the harmonic Fourier series. The more localized in position the function is the larger is the span of frequencies needed to synthesize the function.

```
>> SM_ODE3
  Program to symbolically solve ODE


Enter Single Differential Eq to Solve y(t); e.g., D2y+a*y=0
: D2y+a*y==0
Symbolic Solution, y(t) and v(t)

y =

C2*exp((-a)^(1/2)*t) + C3*exp(-(-a)^(1/2)*t)


v =

C2*(-a)^(1/2)*exp((-a)^(1/2)*t) - C3*(-a)^(1/2)*exp(-(-a)^(1/2)*t)
```

Figure 1.17: Printout for the case of symbolically solving the simple harmonic oscillator when the initial position and velocity are not specified.

The script "SM_fourier_ex" gives three examples of a Fourier series for a square wave, a triangular wave and for a saw tooth waveform. In all cases, the coefficients are printed out for the first six terms of the series and plots of the function and the approximate series representation for the first six terms is plotted, which shows how the function is better approximated when more terms are added. The integral formulae for the Fourier coefficients are displayed in Equation (1.1).

The function is called x and is built out of sine and cosine functions of time with frequencies $k\omega$, k an integer, and coefficients in the series of frequencies a_k, b_k. The function is periodic with lowest frequency $\omega = 2\pi/T$ and T is the period of the function. The series coefficients are determined by evaluating integrals, which is a function well suited to use of the MATLAB script "int".

$$
\begin{aligned}
x &= a_o/2 + \sum_k [a_k \cos(k\omega t) + b_k \sin(k\omega t)] \\
a_k &= 2 \int x(u) \cos(2\pi\, ku) du \\
b_k &= 2 \int x(u) \sin(2\pi\, ku) du \\
u &= t/T,\ [-1/2, 1/2] \\
\omega &= 2\pi/T,\ \omega t = 2\pi u
\end{aligned}
\tag{1.1}
$$

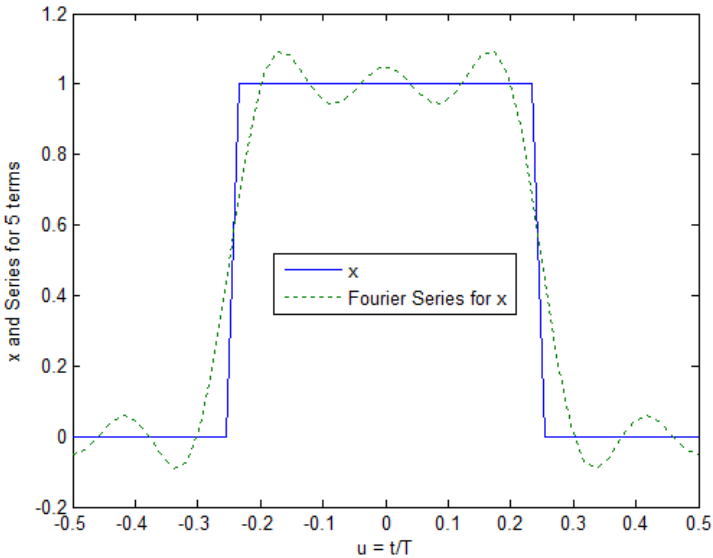


Figure 1.18: Fourier approximation to a square wave after 5 terms (odd terms are zero by symmetry).

Results for a user choice of the square wave are shown in Figure 1.18 after the series is five terms long. The function is even about $t = 0$ so that the sin coefficients are all zero. Plots of the series for all terms are provided in order to give the user a feeling about how the series approaches the function as the number of terms increases. Printout for this choice is given in Figure 1.19.

There is another script called "SM_Fourier_try" which does not have examples to choose from, but rather has user defined functions with symbolic input. Any function can be attempted. The results for a function $t\cos(t)$ with five terms in the series are shown in Figure 1.20 showing the coefficients and Figure 1.21 showing the function and the Fourier series. The user supplies a fully symbolic input of the function of time in two half periods and also the number of terms in the series. In distinction to "SM_fourier_ex" which plots each additional term in the series for one of three examples, "SM_fourier_try" plots the full series for an arbitrary function and a user defined number of terms.

```
>> SM_fourier_ex
  Fourier - fourier series for functions, period = [-1/2,1/2]


x = ao/2 + sumk( ak*cos(kwt) + bk*sin(kwt) )
ak = 2*int(x(u)*cos(2*pi*k*u)du, bk = 2*int(x(u)*sin(2*pi*k*u)du
u = t/T, [-1/2,1/2], w = 2*pi/T, w*t = 2*pi*u
z = pi*k/2, bk = 0, ak = sin(z)/z
Cos Coefficients

a =

    0.6366    0.0000   -0.2122   -0.0000    0.1273

Sin Coefficients

b =

     0     0     0     0     0

z = pi*k, bk = 0, ak = 2*(1-cosz)/z^2
```

Figure 1.19: Printout of "SM_fourier_ex" for the menu choice of a square wave. The series coefficients are printed in symbolic form.

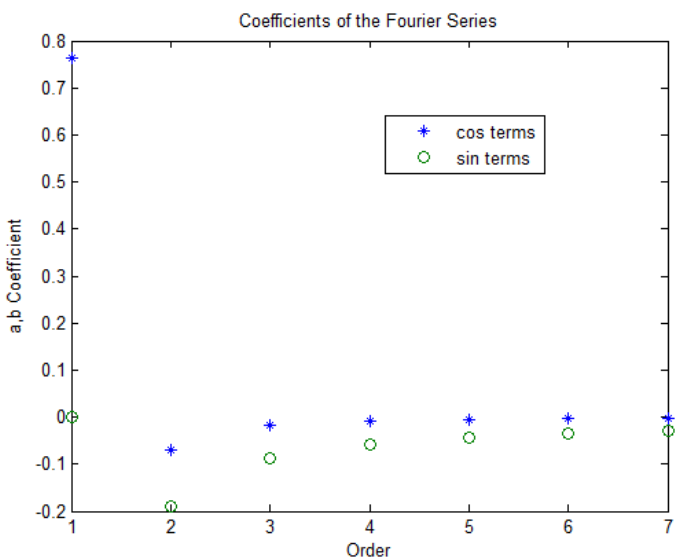


Figure 1.20: Fourier coefficients for the function $t^*\cos(t)$ using "SM_fourier_try".

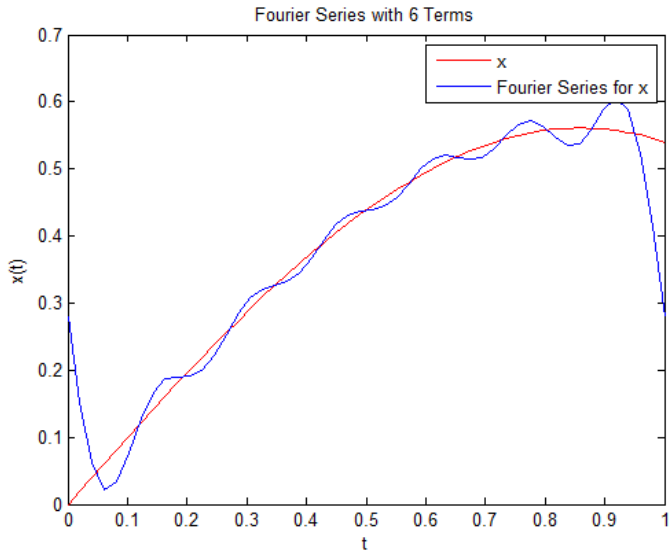


Figure 1.21: Function $t\cos(t)$ and the Fourier series approximation after summing six terms.

1.13. Data Fitting

One indispensable tool in physics is the ability to fit experimental data to some hypothesis. MATLAB has tools for this and additional script has been developed to treat the problem. First, consider a set of data y_i at locations x_i. These points can be fit to hypothesized functions, for example polynomials. The script "Data_Fits" has two sets of experimental data stored in the script. These data are plotted and a second order polynomial is fit to them using the MATLAB "polyfit" function. The result for one data set is shown in Figure 1.22.

To compare to the printout from that fit, tools from MATLAB can be used — the tab "tools" in the Figure Window gives the "basic fitting" option for Figure 1.23. Choosing the "show eqs" and "plot residuals" options yields the plot in Figure 1.23. The data shown in Figure 1.22 can be fit to, and the results compared to the fit shown in Figure 1.23 for a different order of polynomials. All the errors are assumed to be the same, as plotted.

A more general problem can be approached with the script called "Least SquaresFit2" which performs a least squares fit to a straight

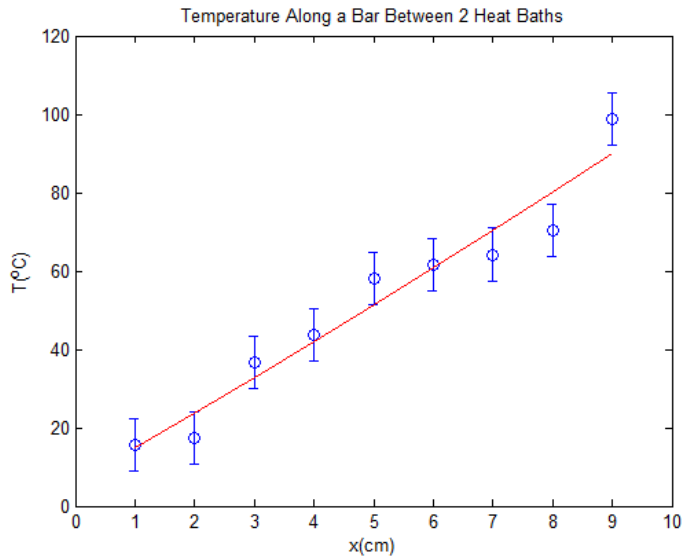


Figure 1.22: Fit to stored data using "polyfit" in the script "Data_Fits" with the assumed shape of a second order polynomial.

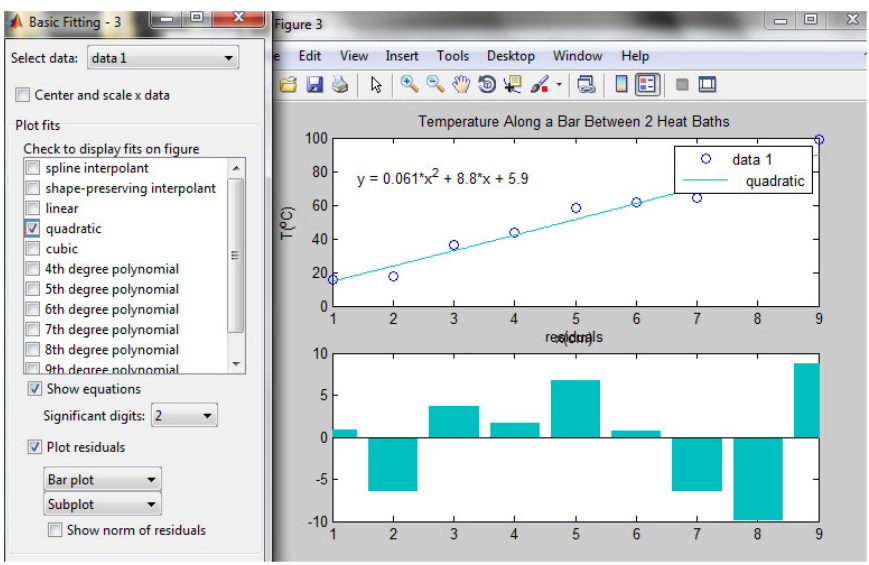


Figure 1.23: Screenshot of the use of MATLAB basic curve fitting using the Figure editing tools provided by MATLAB.

line in the case when the points y_i have different errors. The "polyfit" script assumes all points have the same statistical weight, whereas the least squares fit uses the weight assigned to each data point.

The output contains the slope, intercept, the chi squared, Equation (1.2), of the fit, the number of degrees of freedom of the fit, the error of the slope, a, and intercept, b. The user can use the algebra contained in the script to apply a straight line fit to any data of their choice. The results for a straight line fit to the data shown in Figure 1.22, which displays the assumed errors on each individual point, are presented in Figure 1.24.

$$\begin{aligned} \chi^2 &= \sum_i^n (y_i - y(a,b))^2/\sigma_i^2 \\ y &= ax + b \\ n_{dof} &= n - 2 \end{aligned} \tag{1.2}$$

A still more complex tool is contained in the script "Fits_Chisq_Errors" which is a wrapper script using the MATLAB function "fminsearch" to minimize the chisquared as a function of several variables, taking proper account of the error matrix of those variables. A fit to a cubic polynomial with properly weighted errors is shown in Figure 1.25. The fit is visually better than the straight line fit or the quadratic fit discussed previously. Any arbitrary function can be fit to by adopting the script.

The printout for a fit to some Monte Carlo generated data with a simple Gaussian is shown in Figure 1.26. The plot of the data and the best fit, characterized by a normalized number of events (1000 generated) a mean (0 generated) and a standard deviation (1 generated) is shown in Figure 1.27. Note that the number of events, mean, and standard deviation are all as generated within the quoted error estimates. Very approximately, with N events, the percent error is $1/\sqrt{N}$, so that with 1000 generated events an error of about 3.2% is expected, which can be compared to the diagonal error matrix elements shown in Figure 1.26.

The script of "Fits_Chisq_Errors" contains the function "Fit_Fun" which defines how functions to be fit are defined. These

```
Data Fit - Straight Line, Variable, Uncorrelated Errors

a =

    9.4453

b =

    4.6503

chi =

   21.3378

dof =

     7

err =

    0.1923   -0.6627   -0.6627    3.3706
```

Figure 1.24: Printout of the script applied to the data of Figure 1.22 with variable errors in a least squares fit to a straight line. The fitted slope is a, while the intercept is b. The error matrix for a and b, is "err" in the printout.

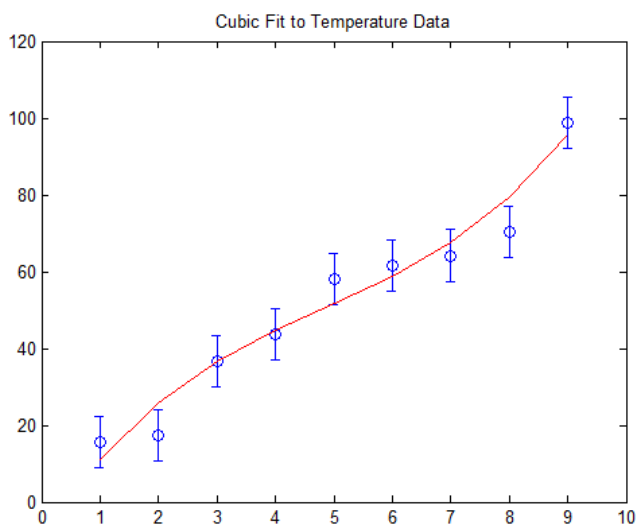


Figure 1.25: Chisquared minimization of a cubic fit to the temperature data fit above to a unweighted quadratic curve and a properly weighted linear straight line fit.

```
 Gaussian Fit to Monte Carlo "Data" -

chs =

    6.2354


dof =

    15


afit =

  994.7965   -0.0276    1.0122

diag =

   31.5963    0.0326    0.0240
```

Figure 1.26: Printout of the fit of Monte Carlo data to a Gaussian. The fitted parameters, a, are the number of entries, the mean and the standard deviation of the fitted Gaussian function. The "diag" refers to the diagonal elements of the error matrix.

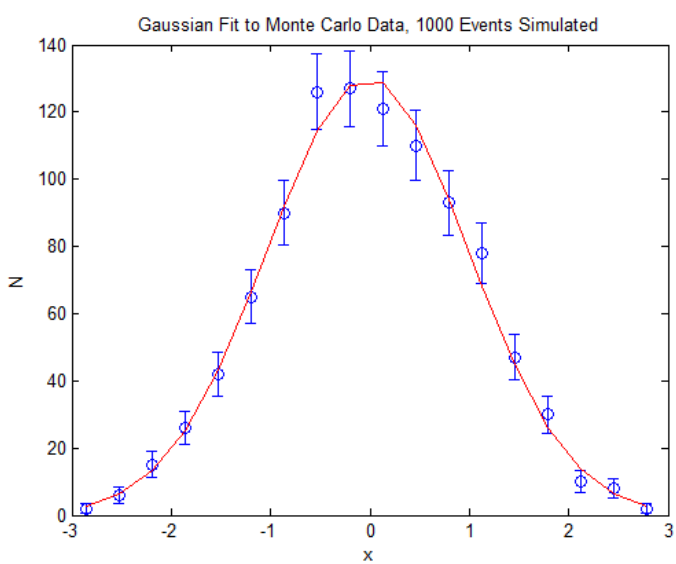


Figure 1.27: The Monte Carlo data shown with statistical errors and the Gaussian fit to the data.

two packaged scripts are available to the user to do fits to any function that is already defined or which can be defined. The entries are the diagonal elements of the error matrix of the fit chosen and the fit function which depends on parameters in any defined fashion. The script "Fits_Chisq_Errors" is set up to fit data to a polynomial, a Gaussian, a Poisson distribution or an exponential. The printout and plots supplied are cubic fits to the temperature distribution and the voltage distribution, as well as the Gaussian fit to the stored Monte Carlo data. The user with some acquaintance with MATLAB should be able to add script for any other desired functional form.

1.14. MATLAB Utilities

MATLAB has many utilities and features and here only the surface can be scratched. The desktop provides a workspace, but also windows with the command history and the command window, providing a history of commands issued in the present and prior sessions and variables currently in memory.

There is a help browser with both an index and a search facility. Help for a particular function can be invoked from the command window, e.g. "help plot". The editor and debugger make writing and running scripts quite easy. Plots can be edited as well as fit using the Figure tab options.

There is a full suite of array operations for vectors including, "max", "min", "length", "mean", "std" (standard deviation), "sum", "diff" and "sort". Matrices can use the "gradient" function which will be used to derive fields from potentials.

There are arithmetic, e.g. +, relational, e.g. >, and logical, e.g. ==, operators. Script flow is controlled by the use of: "if", "while", "for", "end" and "break". The nested loops are conveniently indented by the compiler. Any incorrect script is indicated in red by the compiler as it is typed in.

Equations can be solved using "solve" for algebraic equations, and "dsolve" for ordinary differential equations. Partial differential equations in one dimension are solved using "pdepe". If the solutions

are not possible, ordinary differential equations can be integrated numerically using "ode45". General numerical integrations are handled using "quad".

There is a full suite of data handling utilities for importing and exporting data, but they will not be discussed in this text.

Chapter 2

Classical Mechanics

"The squares of the periodic times are to each other as the cubes of the mean distances."

— **Johannes Kepler**

"Two things that matter to me; emotional resonance and rocket launchers."

— **Joss Whedon**

"The earth doesn't move backward (very much) when you walk only because it's much more massive than you are."

— **K.C. Cole**

The first section was devoted to the mathematical tools which are of general use in this text. Typically there was a script written with a user menu which wrapped a specific MATLAB function. Now the demonstrations for real physics problems begin. In general, the same format for the scripts is in place; an introductory printout, an example and then a user menu to enable the user to build up an intuition about the problem. The displays and plots aim to be "movies" whenever possible, to give a sense of the dynamics of the problem as it evolves in time. A "movie" of the time development allows the user to appreciate both the development of the system in position and the time dependent velocity.

2.1. Simple Harmonic Oscillator

Galileo began the study of physics described mathematically. The story goes that he measured the periods of chandeliers in church using his pulse to measure the time and thus found that the period depended on the length of the device. It is now accepted almost universally that the Universe can be apprehended mathematically which is a great mystery why that should be so.

One of the simplest problems in mechanics is the harmonic oscillator realized as a mass, m, on a spring with spring constant k moving in one dimension, x, in time t. The spring restoring force is a representation of the situation for small oscillations of any bound system, since it represents the first term in the Taylor expansion of the force binding the particle. The one-dimensional differential equation is:

$$d^2x/d^2t = -(k/m)x - b/m(dx/dt) + B\cos(\omega t) \tag{2.1}$$

where b is a damping factor and B is a harmonic driving term with driving frequency ω. The damped frequency in the absence of damping and driving terms is $\omega_o^2 = k/m$ and the motion is oscillatory, $x \sim e^{\pm i\omega_o t}$.

This second order equation needs initial values for position and velocity to be defined in order for the solution to be fully determined. In this exercise, supplied by the script "cm_osc" an initial displacement is supplied equal to A and the initial velocity is defined to be zero. There are three cases which are considered; no damping, no driving term, then damped motion without driving forces, and then damped and driven motion. All cases are solved symbolically using the "dsolve" script introduced previously. The results can be displayed by making the input y, yd, ydr on the keyboard in the Command Window.

To simplify, units where the oscillation frequency is $k/m = 1$ are used. The under damped natural frequency and the resonant response to a driving term at long times are:

$$\begin{aligned} \omega_o &= \sqrt{k/m} \\ \omega_d &= \sqrt{\omega_o^2 - (b/2m)^2} \\ \omega_{dr} &= \sqrt{\omega_o^2 - (b/m)^2/2} \end{aligned} \tag{2.2}$$

In the over damped case, the solutions are exponentials, while in the under damped case, the solutions are oscillatory. The damped frequency is less than the undamped frequency, while the driven resonant frequency is different from both. The frequency half width of the resonant response to the driving force is approximately, $b/2m$.

Less damping means that the resonant response to a driving force is more sharply peaked in frequency.

The script "cm_osc" first asks to choose if the initial position and $A = 1$ is a reasonable choice for simplicity. The damping term is next and $b/m = 0.1$ gives an under damped solution. The damped frequency is 0.999, only slightly shifted from the natural frequency. A movie of the un-damped and damped position is then shown. Finally, a driving amplitude and frequency is asked as input, where $B = 1$ and $\omega = 0.9$ are used as the example. The resonant frequency is approximately 0.997.

When another menu choice is requested, the over damped case can be illustrated by the choice of $b/m = 2.1$. All the plotted results are shown in Figure 2.1 through Figure 2.4. In Figure 2.1, it is clear that the damped frequency is slightly less than that for the un-damped case, when the free and damped motion of $b/m = 0.1$ is compared. The amplitude in the damped case is reduced with time and with respect to the un-damped case.

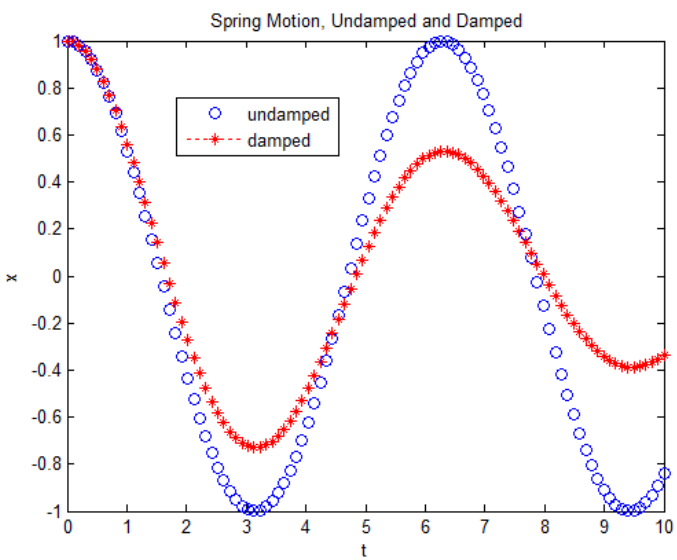


Figure 2.1: $x(t)$ for free and damped motion with $b/m = 0.1$.

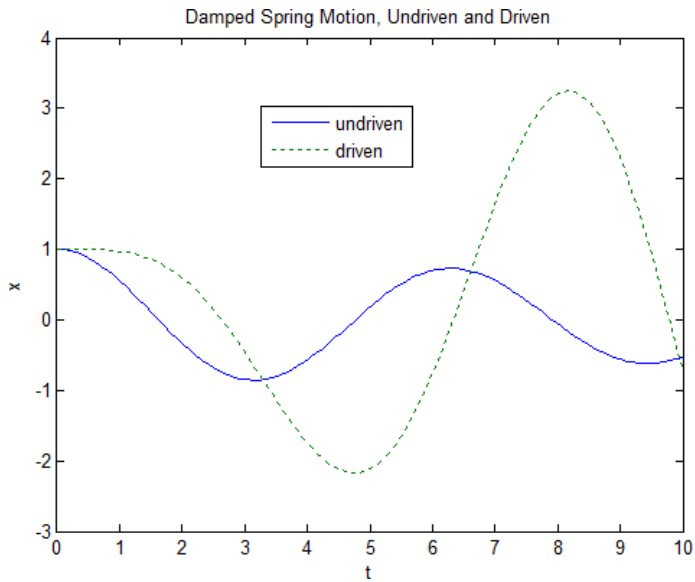


Figure 2.2: Damped motion, driven and un-driven with driving amplitude = 1 and frequency 0.9.

In Figure 2.2, the driven oscillation begins to have a frequency approaching the driving frequency at longer times for a driving frequency of 0.9.

In Figure 2.3, the maximum amplitude, $x(t)$ in the driven case with amplitude of one and with damping factor $b/m = 0.1$ is plotted for the exact solution as a function of the frequency of the driving force. The approximate resonant frequency is also shown, as well as the approximate width of the resonant frequency response. The exact solution differs from the approximate case. However, the expected resonant behavior is seen near the resonant frequency and the resonant width is very approximately what is indicated on the figure. Finally, the un-driven damped response in the over damped case, $b/m = 2.1$, is displayed in Figure 2.4. In that case, the solution is a decaying exponential compared to the under damped case where there is both an oscillatory and an exponential component of the solution, as seen in Figure 2.1.

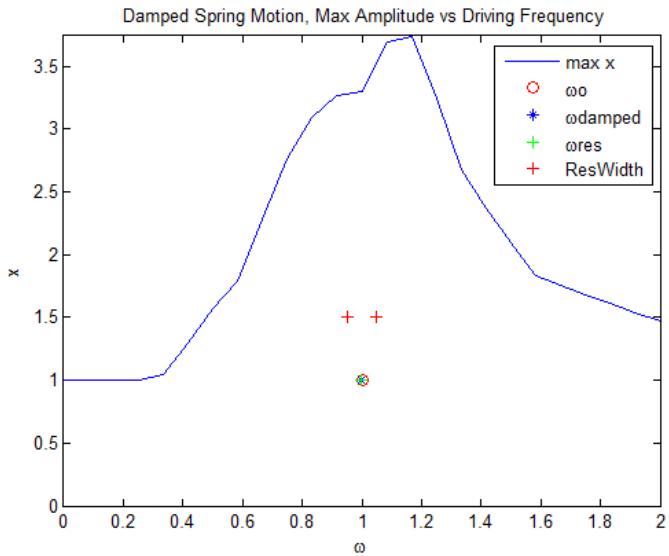


Figure 2.3: Maximum $x(t)$ as a function of driving frequency — under damped, $b/m = 0.1$.

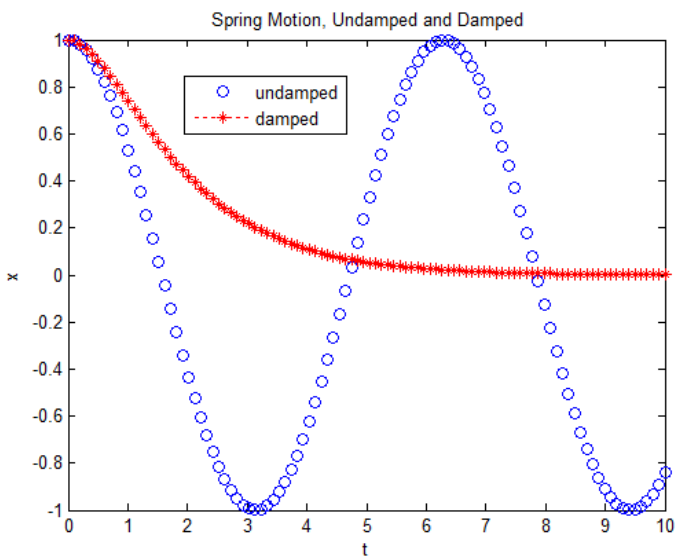


Figure 2.4: $x(t)$ for free motion and in the over damped case, $b/m = 2.1$.

There are many parameters which the solutions depend on. The user can vary them all and see how the solutions vary in response.

2.2. Coupled Pendulums

A more complex harmonic example is the description of two pendulums with a spring coupling between them. The coupled differential equations, with pendulums defined by k and m and with a coupling K are:

$$\begin{aligned} d^2x_1/d^2t &= -(k/m)x_1 - (K/m)(x_1 - x_2) \\ d^2x_2/d^2t &= -(k/m)x_2 + (K/m)(x_1 - x_2) \end{aligned} \tag{2.3}$$

In this case, using the script "cm_s2sho", the MATLAB function "dsolve" is used for the system of two coupled differential equations. The user has a menu to specify k, m and K along with the initial displacements of the two pendulums. This problem can easily be treated as an eigenvalue problem, and the two eigenfrequencies are:

$$\begin{aligned} \omega_1^2 &= (k/m) \\ \omega_2^2 &= (k/m) + 2(K/m) \end{aligned} \tag{2.4}$$

The eigenfrequencies correspond to solutions, eigenvectors exhibiting simple harmonic motion. The eigenvectors correspond first to the case where the 2 pendulums are in phase and the coupling spring, K, is not displaced. The second eigenvector occurs with the two pendulums out of phase. The values of x_1 and x_2 for an example with initial displacements [2, 1], are shown in Figure 2.5 and clearly, in this case, the motion is not simple harmonic. The eigenvectors, illustrating the single eigenfrequencies are plotted in Figure 2.6.

There is a movie which shows the time evolution of the system. The user supplies the values of k and K and the initial positions of the two pendulums. With these options, the behavior of the eigenvectors can easily be seen with the proper choice of initial positions. A frame of the movie for the specific example of Figure 2.5 appears in Figure 2.7. Indeed, that fact can be checked by setting [1, 1] and then [1, −1] as the initial displacements and watching the resulting movie and associated plots.

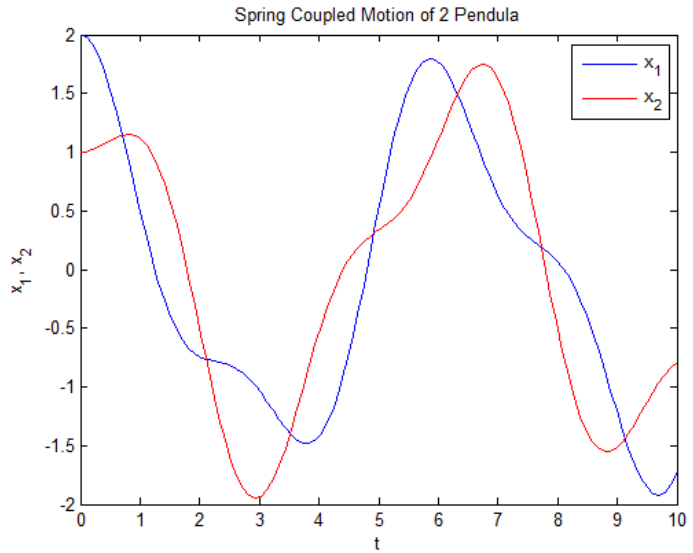


Figure 2.5: Displacements of the two pendulums for the case where $k = m = 1$, $K = 2$ and with initial displacements of [2, 1].

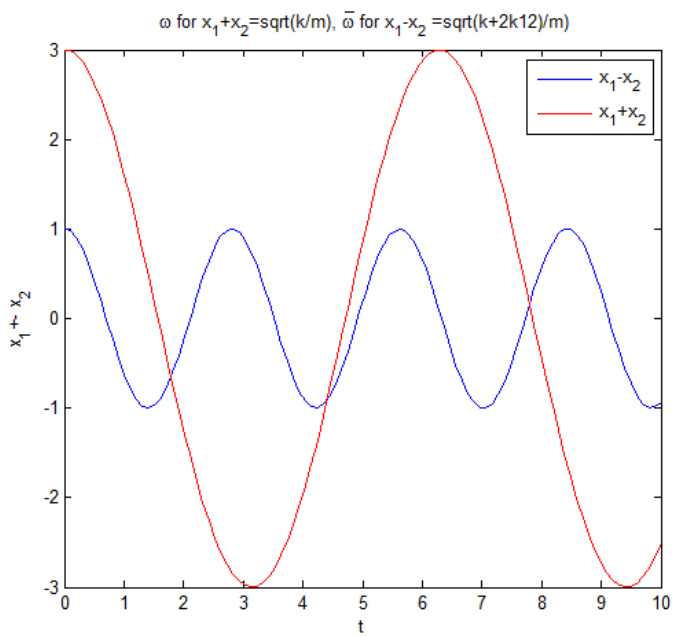


Figure 2.6: Time dependence of the sum and difference of the displacements of the pendulums for the example specified in Figure 2.5. These eigenvectors are simple harmonic.

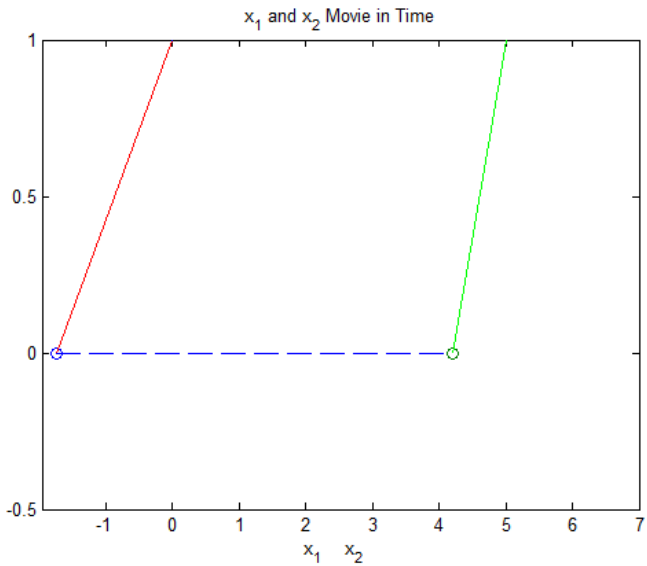


Figure 2.7: Movie of the time evolution of a coupled pendulum system for $k = m = 1$ and $K = 2$, with initial condition of [1, 2] for the positions.

2.3. Triatomic Molecule

One more demonstration with eigenvectors has been worked out, that for the motion of a linear triatomic molecule with two atoms of mass m on the left and right edges and an atom of mass M in the center with two springs coupling the atoms together which simulates atomic bonds. The system of equations to be solved for the three displacements is, taking $k = m = 1$:

$$\begin{aligned} d^2x_1/d^2t &= (x_2 - x_1) \\ d^2x_2/d^2t &= b(-2x_2 + x_1 + x_3) \\ d^2x_3/d^2t &= (x_2 - x_3) \\ b &= m/M \end{aligned} \tag{2.5}$$

The eigenvalue equation is solved using the MATLAB functions “det” and “factor” in the script “cm_triatomic”. The matrix Aw is derived assuming eigenfrequencies for the motion and substituting into Equation (2.5). The solution of the set of equations occurs when the

determinant is zero. The three eigenfrequencies are:

$$\omega_1^2 = 0,\ \omega_2^2 = k/m,\ \omega_3^2 = k/m + 2k/M \tag{2.6}$$

The first case corresponds to an eigenvector with uniform translation of the entire molecule. The second case is the "breathing mode", where the central atom of mass M remains at rest while the outer two atoms have equal and opposite displacements. The printout for this script is shown in Figure 2.8.

```
>> cm_triatomic
  Program to symbolically solve ODE for linear molecule - 3 masses, 2 springs
  k/m = 1, (m/M) ratio = b, outer atoms have mass m, central atom has M


Symbolic Solution Initial Velocities = 0, Initial Positions x(i)
 Aw is the Oscillation Matrix for the 3 Atoms, y = w^2

Aw =

[ 1 - y,       -1,     0]
[    -1, 2 - y/bb,    -1]
[     0,       -1, 1 - y]

Eigenfrequencies = 0, 1, and sqrt(1+2b)
 The Determinant of Aw has Roots y = w^2 of the Eigenfrequencies in sqrt(k/m) Units

ans =

(y*(y - 1)*(2*bb - y + 1))/bb
```

Figure 2.8: Printout for the script "cm_triatomic". The eigenvalues are found using the functions "det" and "factor". The exact motion is found using the MATLAB function "dsolve" for the motion of the three coupled masses.

In general, the molecule has a complex oscillatory behavior. The result for initial displacements of $[-1, 2, 1]$ are shown in Figure 2.9 and a movie is displayed in Figure 2.10. Finally, the simple harmonic behavior where the central atom stays at rest can be invoked with the user choice of $[-1, 0, 1]$ for initial displacements. In this way, the user can confirm the eigenvectors of the problem. All three should be tried.

2.4. Scattering Angle and Force Laws

In physics, one way to understand the forces which act in a given situation is to scatter a probe particle off the force center. Indeed,

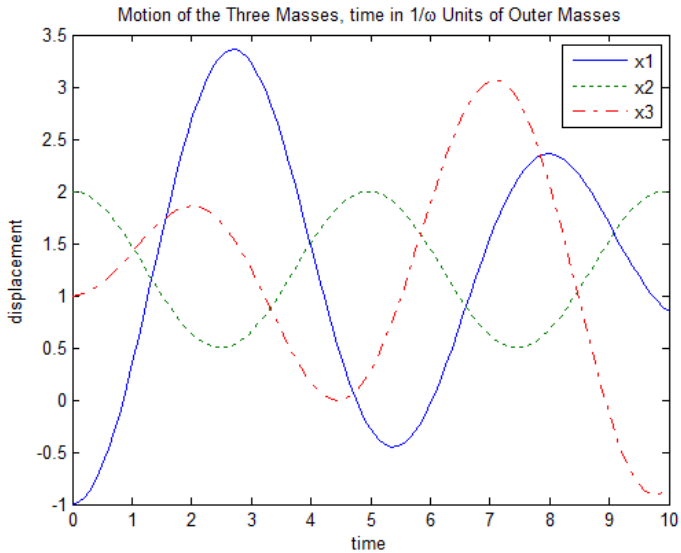


Figure 2.9: Time development of the solution for initial positions of $[-1, 2, 1]$, $m/M = 0.3$.

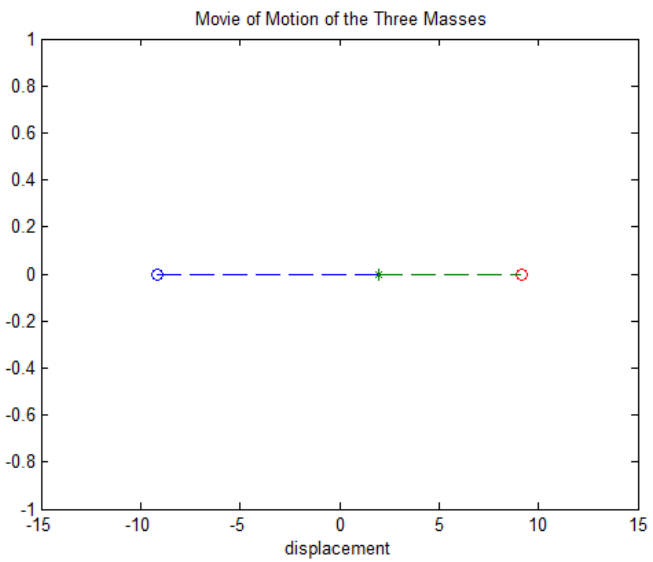


Figure 2.10: Movie snapshot for initial positions of the three atoms for the initial conditions $[-1, 2, 1]$ and $m/M = 0.3$.

in that way, Rutherford discovered that an atom was mostly empty space with the protons all clustered into a compact nucleus which strongly scattered alpha particles when they were used as a probe.

A graphical look at different force laws is provided by the script "Scatt_Force_Law". Central forces that go as inverse power laws with powers from one to four and with attractive and repulsive options are provided in the user menu. Other force laws could be added to the script by making small modifications to the script. A movie of the trajectory with a defined initial impact parameter, b, is shown and then the suite of trajectories is shown. The movie is in equal time steps, so that a feeling for the velocity as a function of time can be obtained. The relationship between impact parameter, b, and scattering angle is plotted in a separate graph.

The MATLAB tool, "ode45" is used which is a numerical solver for a set of ordinary differential equations. In this case, there are four unknowns, the x and y position and the x and y velocity, which are called $y(i)$ in the script. The initial conditions are that $x = -10$ and $y = b$, the impact parameter with initial x velocity $= v_o = \sqrt{2}$ and initial y velocity $= 0$. The mass and kinetic energy of the classical probe particle are taken to be equal to one. The equations as input to "ode45" are;

$$\begin{aligned} dx/dt &= v_x,\ dy(2)/dt = y(1) \\ dy/dt &= v_y,\ dy(4)/dt = y(3) \\ dv_x/dt &= (x/r)(q/r^n) = dy(1)/dt \\ dv_y/dt &= (y/r)(q/r^n) = dy(3)/dt \end{aligned} \tag{2.7}$$

The sign of q defines whether the force is attractive or repulsive. The power n is chosen by the user via a provided menu as is q. The ode45 solver finds all four unknowns numerically. Results for a $n = 2$ attractive force are given in Figure 2.11, while results for $n = 2$ repulsive are shown in Figure 2.12. In general, more localized forces with larger n, give larger deflections at small impact parameters than forces with a weaker r dependence. The user can explore these characteristics by watching all eight of the possible movies. In the special case of an inverse square law, the attractive and repulsive orbits are

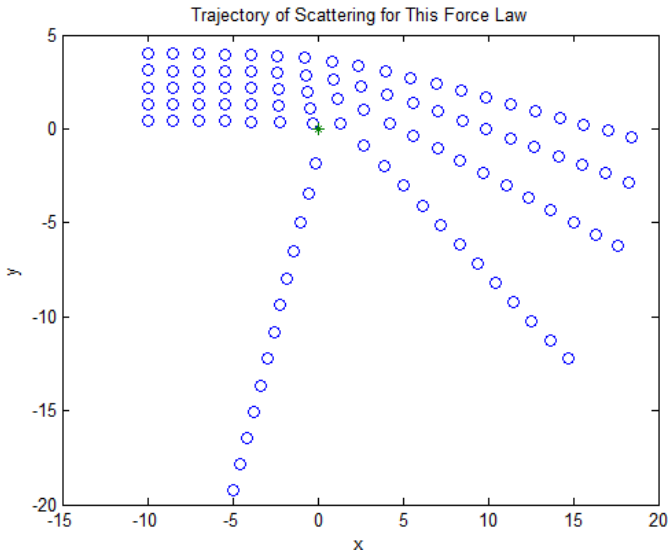


Figure 2.11: Scattering trajectories for different b for a $1/r^2$ attractive force.

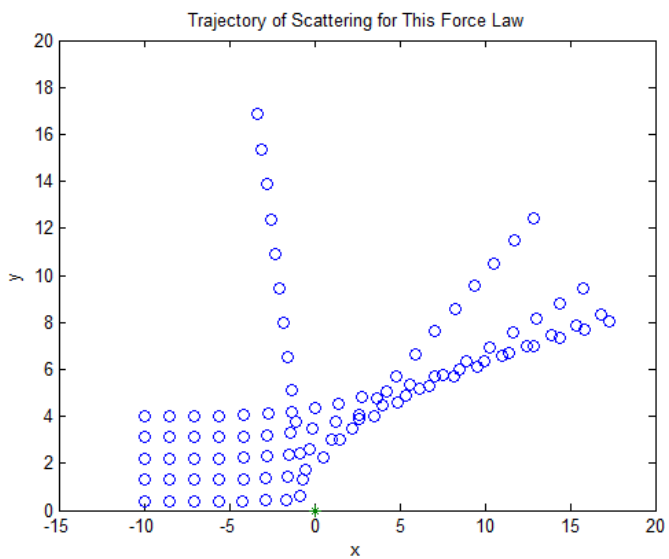


Figure 2.12: Scattering trajectories for different b in the case of a $1/r^2$ repulsive force.

on the two distinct hyperbolic trajectories which are explained in more depth later in the section on Keplerian orbits.

Finally, a plot of the scattering angle as a function of impact parameter is given in Figure 2.13. Clearly, the small impact parameter trajectories see a stronger force and; therefore, have larger scattering angles. The experimenter cannot, alas, aim the probe so that all impact areas are equally probable and the cross section is just proportional to the area of a ring of impact parameters of area $2\pi bdb$. The resulting scattering angle distribution is easily found, since there is a one to one relationship between scattering angle and impact parameter;

$$d\sigma \sim bdb = b(db/d\theta)d\theta \tag{2.8}$$

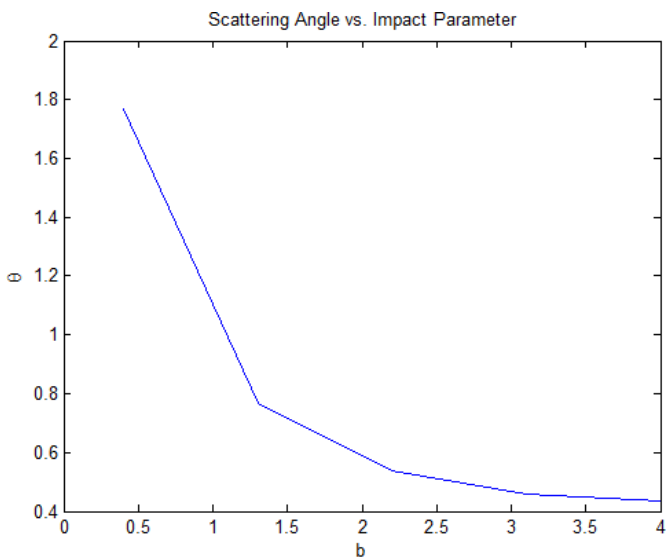


Figure 2.13: Relationship of the scattering angle to the impact parameter for the case of a $1/r^2$ repulsive force.

Because of the equally probable impact parameter areas, bdb, most b are large which means most angles are small. In that case, the experimentally observed angular distribution will be peaked at small scattering angles. An example is Rutherford scattering, which falls

as the fourth power of the scattering angle for small angles. A rough estimate for Coulomb scattering in the Rutherford case reproduces that dependence.

$$\begin{gathered} d\sigma/d\Omega \sim 1/\theta(d\sigma/d\theta) \sim b/\theta(db/d\theta) \\ F \sim 1/b^2,\ t \sim b/v \rightarrow b \sim 1/\theta \\ d\sigma/d\Omega \sim 1/\theta^4 \\ d\Omega = d\phi d\cos\theta = 2\pi \sin\theta d\theta \sim 2\pi\theta d\theta \end{gathered} \tag{2.9}$$

The Coulomb force, F, is large for $r \sim b$ and the time it acts, t, goes as $\sim b$ divided by the incident particles velocity, v. The scattering angle due to the momentum impulse, Ft, is then $\sim 1/b$, so that in this case, the inverse fourth power is obtained for the angular distribution. Other forces would give other predictions for the observed scattering angle distribution. Note that the attractive and repulsive orbits are on the two arms of hyperbolae for an inverse square law. With the addition of negative charge, both possibilities open up as opposed to gravitational attraction only.

2.5. Classical Hard Sphere Scattering

Kinematics plays a big role in scattering beyond that of the dynamics which were previously discussed. The script to explore hard sphere scattering is contained in "cm_NR_scatt". The scattering is $m + M \rightarrow m + M$, where m is the projectile mass taken to be one and M is the target mass. The user menu consists of the choice of M. Once M is known, the kinematics for different scattering angles of the projectile and recoil angles of the target are explored. The conservation of kinetic energy, $T = mv^2/2$, and vector momentum is, $m = 1$:

$$\begin{aligned} v_{in}^2 &= v_2^2 M + v_1^2 \\ \vec{v}_{in} &= \vec{v}_1 + \vec{v}_2 \end{aligned} \tag{2.10}$$

These equations can be solved for the recoil velocity as a function of the angle of the recoiling target, ϕ.

$$v_2 = 2\cos\phi/(1 + M/m) \tag{2.11}$$

Once the recoil angle is chosen, the recoil velocity is solved for, and then the scattered projectile velocity, v_1 follows from momentum conservation as does the scattering angle θ of the projectile.

$$\begin{aligned} v_1 &= \sqrt{1 + v_2^2} - 2v_2 \cos\phi \\ v_1 \sin\theta &= v_2 \sin\phi \end{aligned} \tag{2.12}$$

The results for representative recoil angles are shown as a "movie" which indicates the initial velocity, the scattered velocity and angle, and the recoil angle and velocity. In this way, an intuition is built up as regards the kinematics of scattering.

In Figure 2.14, a frame of the movie for the case of $M = 1$ is shown. In this case, the projectile can transfer all its velocity to the target, which is familiar in billiards. Note that the angle between the scattered projectile and the recoiling target in this equal mass case is always ninety degrees. In Figure 2.14, the target recoils with

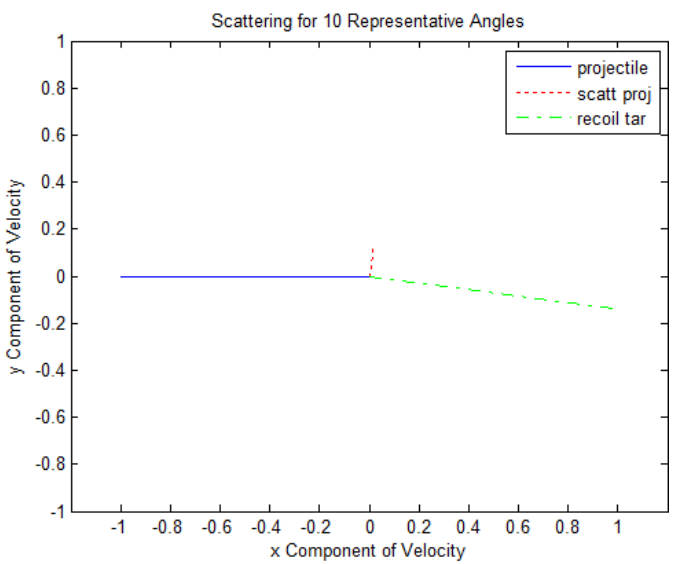


Figure 2.14: Scattering of a projectile and the recoil momentum and angle for the case of equal target and projectile mass. This is a snapshot of a movie covering several scattering angles.

almost the full velocity of the projectile. This fact for scattering has an application in neutron moderation. Only for free protons will neutrons slow down significantly, so that neutron moderators normally contain molecular hydrogen or light elements with large capture cross sections such as boron.

The complementary case is shown in Figure 2.15, where $M = 10$. In that case, the recoil target never attains more than 18% of the velocity of the projectile, while the scattered projectile always retains at least 82% of the projectile velocity. This behavior is also familiar to pool players when using the "bumpers". A light particle cannot transfer velocity to a heavy target because it is momentum that is transferred not velocity. This is well understood by car drivers. A Mack truck colliding with a Smart car will not suffer a large recoil.

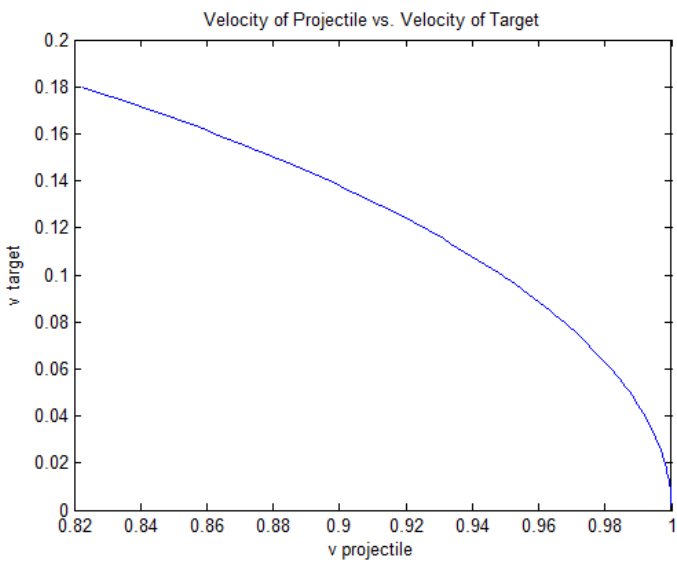


Figure 2.15: Target velocity as a function of scattered projectile velocity in the case of $M = 10$. There is a maximum velocity less than the projectile, which the target can attain that depends on M.

The user can vary the masses and see how the velocity partition between recoil and projectile particles is altered when M is varied.

2.6. Ballistics and Air Resistance

Motion of a ballistic projectile in a uniform gravity field, g, near the surface of the earth is a practical problem and one which needs to take air resistance into account. The equations which are used are solvable and are computed symbolically using the MATLAB function "dsolve".

$$\begin{gathered} d^2x/d^2t + kdx/dt = 0 \\ d^2y/d^2t + kdy/dt + g = 0 \\ x(0) = 0 = y(0) \\ v_x(0) = v_o \cos\alpha, \quad v_y(0) = v_o \sin\alpha \end{gathered} \tag{2.13}$$

The x coordinate is horizontal and y is the vertical position. The constant k specifies the velocity dependent air resistance. The initial angle of the projectile is α. The printout for the script "cm_ballis_sym" is shown in Figure 2.16.

Note that there is a terminal velocity at long times, where $v = g/k$ and where the acceleration is zero. This phenomenon is familiar for sky divers and others when the air resistance exerts a force matched to the acceleration of gravity. At long times $y \sim (g/k)t$.

A movie of the projectile motion, y as a function of x, is displayed and the complete trajectory is plotted in Figure 2.17, comparing the cases with and without resistance. The user menu has a choice of initial velocity and the initial angle of the projectile. In this way, the user can confirm the well-known fact that the maximum range obtains when the projectile starts at 45 degrees.

2.7. Rocket Motion — Symbolic and Numerical

The motion of a rocket is defined by the exhaust velocity, v_o, or the velocity at which material of mass dm is ejected from the rocket with respect to the rocket. For a present rocket mass m, conservation of momentum leads to a rocket velocity change dv:

$$mdv = -v_o dm \tag{2.14}$$

Integration of Equation (2.14) leads to the result that the velocity of the rocket depends logarithmically on the ratio of the rocket mass

```
>> cm_ballis_sym
  Program to compute the trajectory of a projectile with air resistance


Projectile Motion, Air Resistance  - Acceleration = k*dy/dt
 Air resistance k (sec^-1), Initial Angle/Velocity alf, vo -  x(t) and

  vo cos(alf)   vo exp(-k t) cos(alf)
  ----------- - ---------------------
       k                  k
x(t) With No Resistance

  t vo cos(alf)

  g + k vo sin(alf) - g k t   exp(-k t) (g + k vo sin(alf))
  ------------------------- - ------------------------------
              2                              2
             k                              k
y(t) With No Resistance

                     2
                  g t
  t vo sin(alf) - ----
                   2
y Velocity With Air Resistance

  exp(-k t) (g + k vo sin(alf))   g
  ----------------------------- - -
               k                  k
```

Figure 2.16: Printout showing the $x(t)$ and $y(t)$ solutions with and without air resistance.

to its initial value, m_o, or the payload ratio:

$$v(m) = v_o \ln(m/m_o) \tag{2.15}$$

Assuming a constant burn rate of fuel, $\dot{m} = dm/dt$, and a total burn time, T, if all the fuel were exhausted, with no payload, the differential equation for a simple rocket as a function of time follows:

$$m = m_o - \dot{m}t$$
$$T = m_o/\dot{m}$$

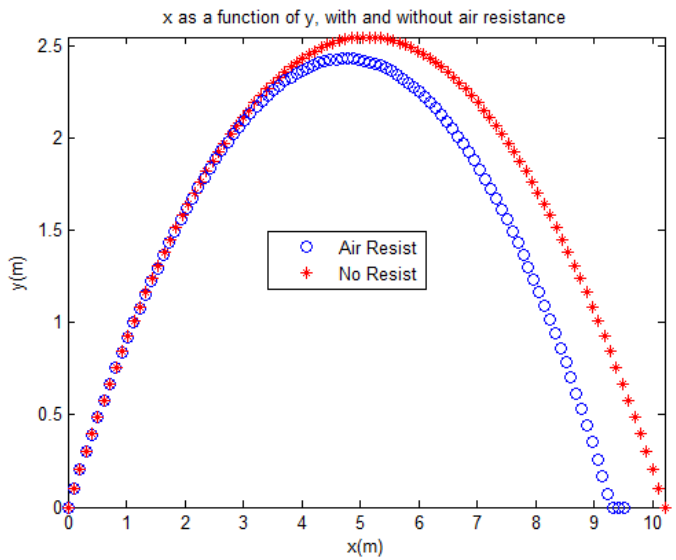


Figure 2.17: Plots of the trajectory of a projectile starting at 45 degrees and with a muzzle velocity of 10 m/sec with and without air resistance, $k = 0.1\,\mathrm{sec}^{-1}$.

$$d^2y/d^2t = -v_o/(T-t)$$
$$d^2y/d^2t = -v_o/(T-t) + g \tag{2.16}$$

The addition of a uniform gravity field, g, makes the equation somewhat more complex but still solvable. A closed form solution for the presence of a real inverse square gravity field does not exist, however.

The rocket script is contained in "cm_rocket_sym". The printout of the symbolic solutions to the equations uses the MATLAB "dsolve" function which appears in Figure 2.18.

Integrating the rocket equation, the relationship of the payload mass ratio to the initial mass, defines the final velocity ratio to the exhaust velocity as in Equation (2.15).

A movie is provided to the user after the payload ratio is chosen. The rapid increase in distance as the rocket nears the end of the burn time is evident when viewing the movie of altitude as a function of time. Numerical results for a 1% payload ratio are shown for the rocket acceleration in Figure 2.19 and the rocket altitude as

```
  Solve non-relativistic rocket, symbolically - no friction or forces


Solve d2y/dt2 = vo /(T-t), vo = exhaust velocity w.r.t. rocket, T is Burn Time = mo/dmdt

v =

vo*(log(T) - log(T - t))


y =

(T - (T - t)*(log(T) - log(T - t) + 1))*vo

Final Velocity = vo*ln(mo/mp), mp = Payload Mass - Works with Multi-Stage Analysis
Solve With Rocket in a Uniform Gravity Field - g

v =

(log(-T) - log(t - T))*vo - g*t


y =

(t + T*log(t - T) - t*log(t - T) - T*log(-T) + t*log(-T))*vo - (g*t^2)/2

Numerical Results: total possible burn time = T
Payload ratio mp/mo => Payload burn time tp = T(1-mp/mo)
vo = exhaust velocity, acceleration in vo/T units, velocity in vo units
Distance at the end of payload burn in voT units
```

Figure 2.18: Symbolic solutions to the simple rocket equation and the equation in a uniform gravity field.

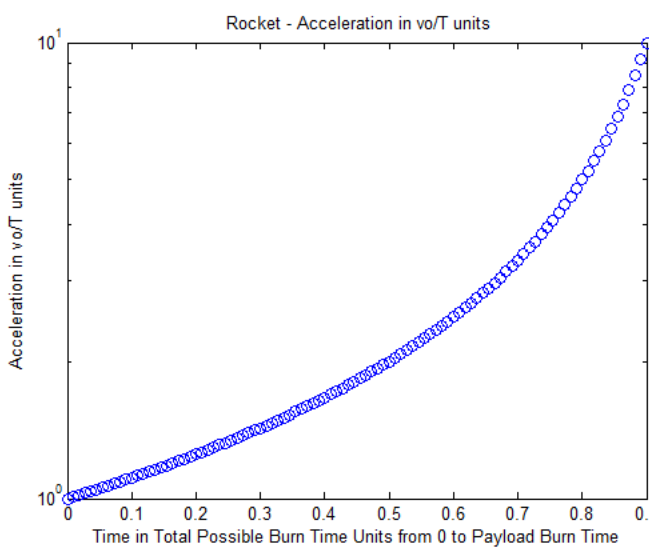


Figure 2.19: Rocket acceleration as a function of time. There is a rapid increase in acceleration as the rocket approaches the end of a burn with a small payload ratio.

a function of burn time in Figure 2.20. It is interesting to observe that rocket velocities in excess of the exhaust velocity are possible, if the rocket has a small payload ratio. In the 1% payload case, the final rocket velocity is about five times that of the exhaust velocity. For small payload ratios, the altitude approaches the exhaust velocity times the total burn time.

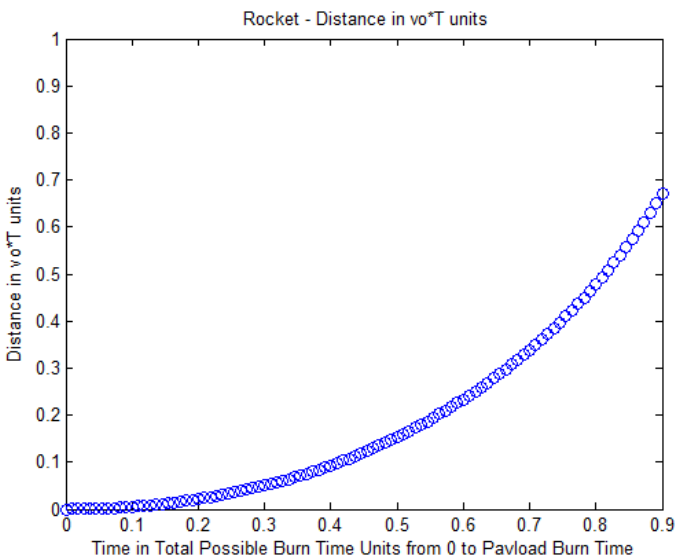


Figure 2.20: Rocket altitude as a function of time for a 1% payload. The movie shows the time development of the altitude which illustrates the sharp increase in acceleration and velocity at late times.

A more purely numerical calculation appears in the script, "cm_rocket_num2". The solutions found symbolically in the script "cm_rocket_sym" are evaluated in the particular context of a Saturn V rocket. As context to set the scales, the low earth orbital velocity of 7.9 km/sec, the Earth escape velocity of 11.2 km/sec, and the escape velocity of the solar system of 42.1 km/sec are printed out. The equatorial launch velocity of 0.46 km/sec is also printed to remind the user of the reason why rocket launches in the US were placed in Florida. The effect is, however, small and ignored in what follows. Escape velocities needed to escape from a mass M starting

from a radius r are:

$$v_{es} = \sqrt{2GM/r} \tag{2.17}$$

A user menu of scaling to a Saturn V is invoked. For a rocket two times heavier, with only a 200 kg payload and a 4 km/sec exhaust velocity, a one stage rocket can escape from the solar system. The acceleration of this scaled up Saturn is shown in Figure 2.21. Note that it takes almost two minutes, 125 sec, for the rocket to build up sufficient acceleration to lift off by overcoming the acceleration of gravity. This is familiar to those who watch rocket launches. A movie of the rocket distance as a function of time is provided to the user. The velocity as a function of time appears in Figure 2.22.

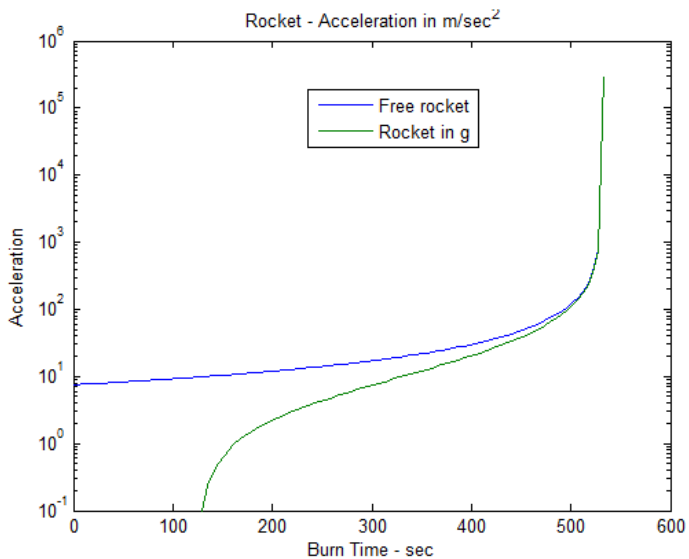


Figure 2.21: Acceleration for a free rocket and a rocket in a uniform gravity field with acceleration g. Note the delay in the gravity field case which is needed to overcome the gravity well of the Earth.

The script gives some added printout for the user defined rocket. The burn time in this example is 533 sec. Payload ratios for a given final velocity are evaluated using Equation (2.15). The rocket could put 489,000 kg into an Earth escape orbit, or a 6.1% payload ratio.

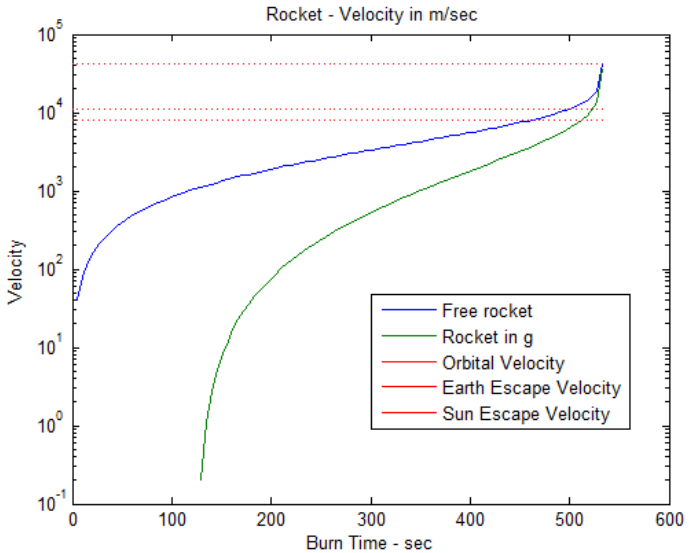


Figure 2.22: Velocity of the scaled up Saturn V rocket with a 200 kg payload. Note that this small payload is just small enough to attain escape velocity from the solar system.

However, to escape the sun, this rocket can only have a 215 kg payload or a payload ratio of 0.000027. A realistic scale is set by the actual Saturn rocket and escape module with a payload ratio of 0.3%.

2.8. Taking the Free Subway

It is an amusing thought experiment to imagine a subway shaft cut along a chord connecting two points. Because of the Gauss law that the important thing is the mass between the center of a body and the test mass, there is simple harmonic motion of that test body moving along such a chord. For a uniform density, ρ, sphere of radius R, the distance along the chord is x_o, the maximum depth of the subway is d, the angle subtended by the chord is θ, and the distance along the Earth's surface is s:

$$\begin{aligned} x_o &= 2R\sin\theta \\ d &= R(1-\cos\theta) \\ s &= 2R\theta \end{aligned} \tag{2.18}$$

The circular frequency of the motion, ω_o, is independent of the particular chord and depends only on constants which define the gravity field of the Earth in the uniform density approximation.

$$\omega_o^2 = G\rho(4\pi/3) = g/R \tag{2.19}$$

The trip time is then, $T = \pi/\omega_o$, which for the Earth is 2533 sec, or about 42 minutes, independent of the trip distance. The script which was written to cover this problem appears in "cm_subway2". It provides some numerical printout, reproduced in Figure 2.23, a movie of the motion in order to provide insight into the velocity as a function of time, and the trajectory along the chord as a function of time, given in Figure 2.24. Who says there is no free lunch? We can use the gravity field of the earth to travel for "free."

```
"Free" Subway - Earth Radius = 6.38e+06 (m)
Enter Subway Distance in km : 20000
Free Subway Max Depth = 6.38e+06 (m)
Free Subway Distance Along Earth = 2.00434e+07 (m)
Circular Frequency = 0.00124001 Trip Time = 2533.53 (sec)
```

Figure 2.23: Printout for the user supplied distance along the subway showing the subway depth, surface distance, and travel time.

However, the velocity of the trip increases with the distance and the heat encountered in a "deep" trip might be a bit impractical. Still, the study of the use of the properties of the inverse square law for gravity is amusing.

2.9. Large Angle Oscillations — Pendulum

The pendulum equation which describes simple harmonic motion is correct only in the small angle approximation. The approximate frequency is $\omega_o = \sqrt{g/L}$, where L is the length of the pendulum. In general, the equation is nonlinear and therefore the motion is not harmonic.

$$d^2y/d^2\theta = (g/L)\sin\theta \rightarrow (g/L)\theta \tag{2.20}$$

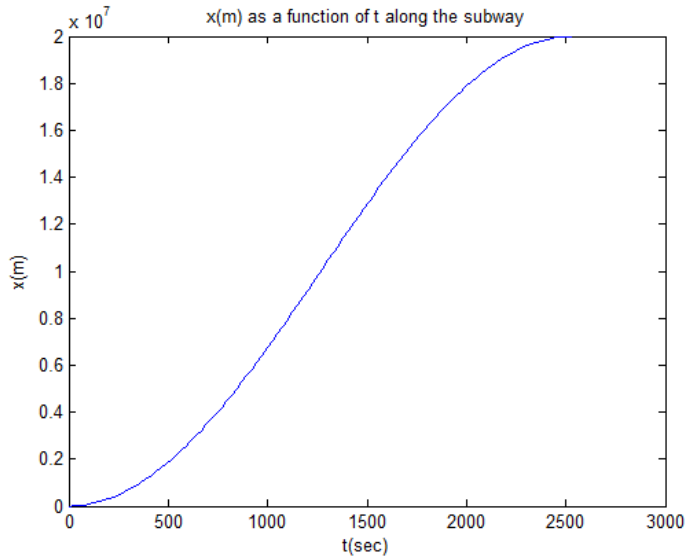


Figure 2.24: For a chord distance of 20,000 km the travel time is 2533 sec. The slope (velocity) is greatest at the midpoint of the trip.

The script provided to study this problem is "cm_pendul" which uses the MATLAB numerical differential equation solver "ode45". The printout for a particular example is shown in Figure 2.25, while the plot of the angular velocity in this case is shown in Figure 2.26. A movie of the angular position in the small angle case and also the general case is provided to the user in order to build up an intuition as to the regions of approximate validity of the small angle solution. In general, the large angle period is increased, and the printout gives the first term in the series expansion for the solution.

```
Pendulum - Large Oscillations
Enter Initial Angle (degrees): 150
Enter Initial Angular Velocity (degrees/sec): 0
Enter g/L in MKS units: 5
Small Angle Circular Frequency = 2.23607 1/sec
Small Angle Period (sec) = 2.80993
Period = 2 * pi *sqrt(L/g) Increased by Factor 1 + thetao^2/16
```

Figure 2.25: Printout for the "cm_pendul" script. The particular example has a large initial angle but no initial angular velocity.

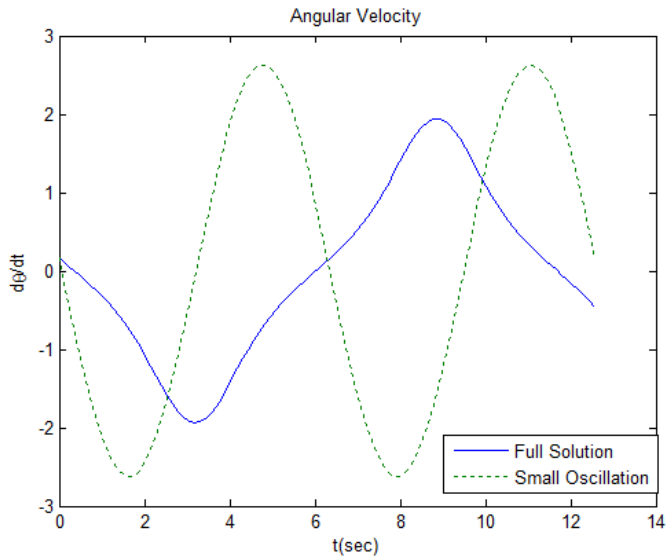


Figure 2.26: Angular velocity in the simple harmonic case and the solution for a particular large oscillation with parameters defined in Figure 2.25.

2.10. Double Pendulum

Large angle oscillations of two coupled pendulums are treated in the script "cm_chaotic". The motion is nonlinear because the oscillations have a large amplitude and they are called chaotic. The equations of motion are four fold, for the angular location of the two mass points and for the velocities of the two points. They are not very transparent, but can be studied by examining the script provided. The masses, m, are taken to be equal to one as are the g/L ratios. The length, L, of both pendulums is set to be one also. The inputs are the initial angles of the two mass points. The initial velocities are fixed at zero.

Plots of the angular velocity and position as a function of time of the two mass points are made and a movie of the motion is played. The position of the mass points appears in Figure 2.27, while a frame from the movie appears in Figure 2.28. The MATLAB function "ode_45" was used to create the numerical solutions of these nonlinear and coupled equations. The movie viewer may be reminded of the motion of nunchucks in kung fu movies.

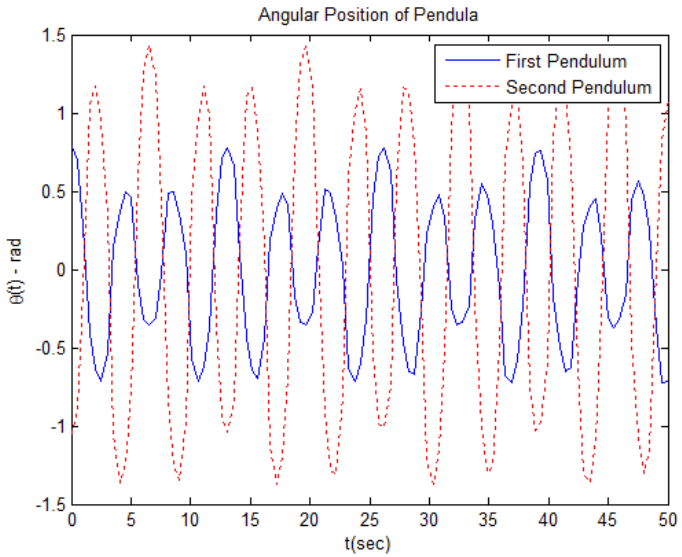


Figure 2.27: Plot of the positions of the two pendula for initial positions of 45 and −60 degrees.

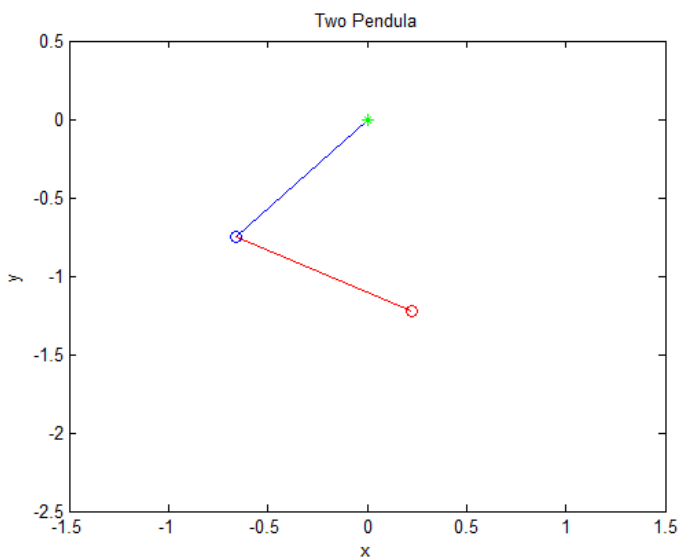


Figure 2.28: A frame from the movie of the pendulum's positions as a function of time for initial angles of 45 and −60 degrees.

2.11. Coriolis Force

There are several solvable problems in physics arising in accelerated reference systems. One such problem concerns the "fictitious" forces that arise on Earth due to its rotation with circular frequency, ω. In particular, there is the Coriolis force which is treated in the supplied script "cm_coriolis" for the special case of dropping a particle with no initial velocity components. The resulting solution is found using the MATLAB symbolic tool "dsolve" for the time development in the vertical (z) and East (y) direction at a latitude defined by θ. The printout is shown in Figure 2.29.

```
Coriolis Force, Nothern Hemisphere, w is w*cos, Latitude
 z is Vertical, x is South and y is East

zz =

(g*t^2)/2


yy =

-(g*t^3*w)/3

Enter Initial Free Fall Height (m): 10000
Enter Latitude (deg): 45
Total Eastward Deflection (m) = 15.4868
```

Figure 2.29: Dialogue for the script which explores Coriolis force. In this simple case, y as a function of z is easily determined. The quantity w is the angular velocity of the Earth.

The coupled equations are:

$$\begin{aligned} d^2y/d^2t &= -2\omega\cos\theta(dz/dt) \\ d^2z/d^2t &= g \end{aligned} \tag{2.21}$$

The solutions shown in Figure 2.29 can be used to remove the time dependence to find the trajectory $-y(z)$. The numerical value for a drop of 10,000 m from 45 degrees of north latitude is shown in Figure 2.30. In some real sense you aim at $y = 0$, but the Earth rotates away from you by 15.5 m during the time of the fall.

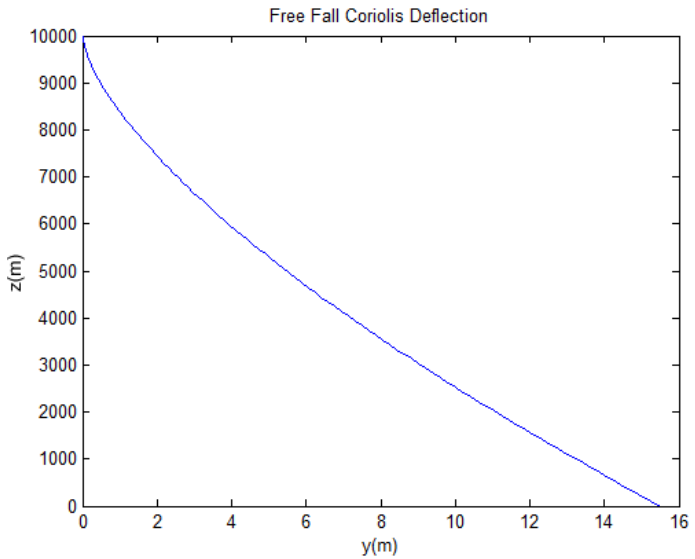


Figure 2.30: Trajectory of y as a function of z during free fall of an object released from an altitude of 10,000 m at 45 degrees north latitude.

2.12. Kepler Orbits — Numerical

The problem of the numerical evaluation of orbits in a central inverse square law is treated in the script "cm_kepl3". The core computation is done with the use of "ode45", the MATLAB numerical integrator for a set of ordinary differential equations. The initial dialogue, in response to a user defined choice of distance to the sun, is a request for the initial velocity, both radial and tangential. The velocity for a circular orbit and the escape velocity at that distance are provided to set the scale. The values for Earth are also given, so that Kepler's laws can be checked numerically.

The motion defined by these initial conditions is displayed as a (x, y) movie. The time spacing is uniform, so that an intuition for orbital velocity can be attained by trying different configurations. Radial velocity is largest nearest the sun, as expected. Results for initial radial velocity $= 0$ and tangential velocity in AU/yr of 6.28, 8 and 9.5 are shown in Figures 2.31, 2.32 and 2.33 respectively in the specific case of an initial radius of one astronomical unit or AU. They correspond to the possible Keplerian orbits of a circle, ellipse

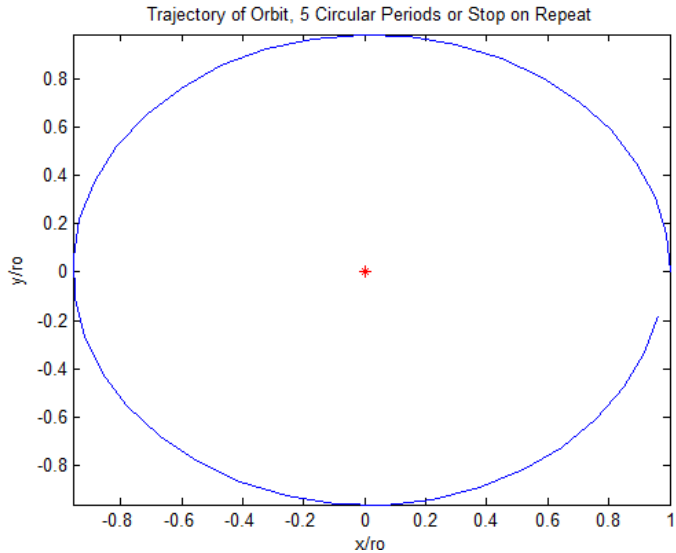


Figure 2.31: Plot of the (x, y) trajectory for a circular orbit. The Sun is at the origin, marked by *. The orbits in an inverse square force are, in fact, re-entrant but the numerical integration is not exact.

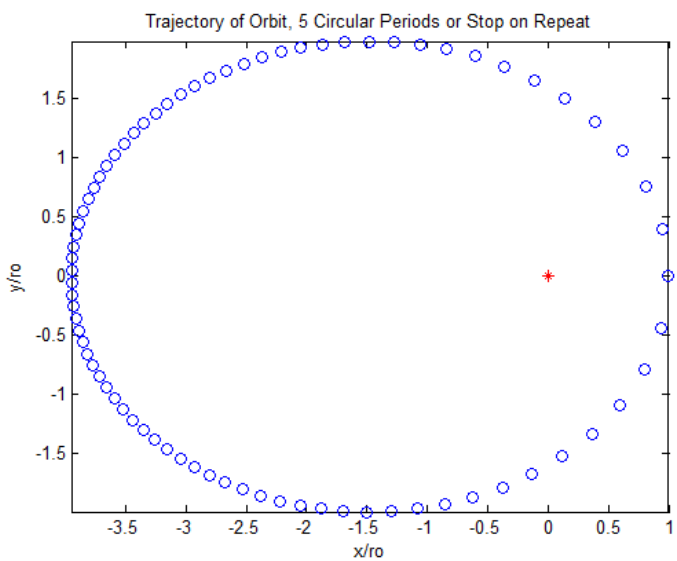


Figure 2.32: Plot of the (x, y) trajectory for an elliptical orbit. The sun is located at a focus of the ellipse and is marked by *. The speed of the orbit is largest when nearest the sun.

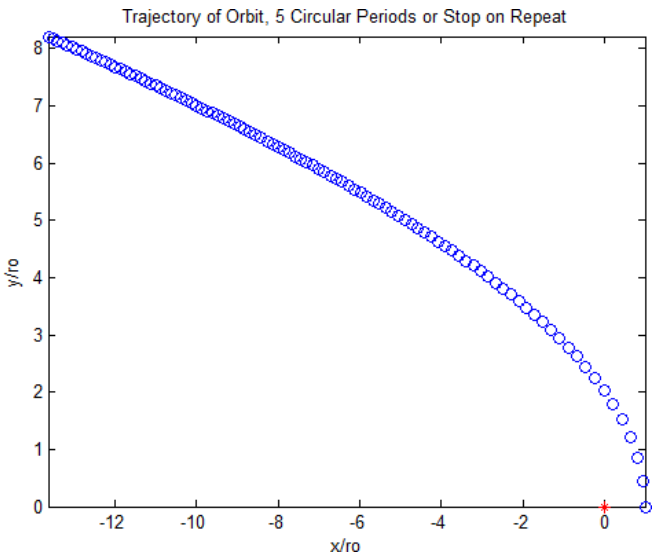


Figure 2.33: Plot of the (x, y) trajectory for a hyperbolic orbit. The location of the sun is indicated by *.

and a hyperbola. In the case of the ellipse and hyperbola, the last movie frame is shown in order to illustrate the larger velocity near the sun in those cases.

The numerical integrations work in general, but the script choices by the user may set plot limits which may be violated in the case of orbits which fall into the Sun or which escape very rapidly out of orbit. For the circular orbit, the sun is at the center, while for the elliptical, the sun is at one of the foci. Other orbits are possible and the user should explore the possibilities. The time scale for integration is the time span of five circular orbits at the starting radius. Note, in particular that a circular or elliptical orbit is "re-entrant" — it repeats in time so that the orientation of the ellipse in the plane does not change from orbit to orbit. This is a property of the inverse square law and is not true for other forces.

2.13. Analytic Kepler Orbits — Energy Considerations

Gravity is a central force. This means that orbital motion is confined to a plane, (r, θ) called the ecliptic for the solar system. Furthermore,

the angular momentum, L, is conserved which allows for the reduction of the problem to an effective one dimensional equation of motion. The angular momentum is:

$$L = mr^2 d\theta/dt \tag{2.22}$$

The effective one-dimensional conserved energy is, $E = T_r + V_{\text{eff}}$, where T_r is the radial kinetic energy and V_{eff} is the effective radial potential energy. Setting the orbiting mass m to be one:

$$E = (dr/dt)^2/2 + [L^2/(2r^2) - GM/r] \tag{2.23}$$

The energy is the sum of the radial kinetic energy, the gravitational potential energy, and an effective repulsive inverse cube force law.

The numerical orbital script, "cm_kepl3" could be modified to cover different force laws by any user willing to change a few lines of code. In the case of the symbolic script "cm_kepl", the effort has been focused on analytical solutions to the inverse square problem. Some of the relevant equations for a user choice of radius $= r_o$ are:

$$\begin{aligned} L_c^2 &= r_o GM \\ E_c &= -GM/2r_o \\ q &= E/|E_c| \\ e &= \sqrt{1+q} \\ r/r_o &= 1/(1+e\cos\theta) \\ v/v_c &= (dr/dt)/\sqrt{GM/r_o} = \sqrt{q+2(1+e\cos\theta)} \end{aligned} \tag{2.24}$$

The angular momentum for a circular orbit of radius r_o, is L_c. The energy for a circular orbit of that radius is E_c. The controlling parameter is the energy of the orbit, E, which is normalized to the energy of the circular orbit by defining the parameter q. The eccentricity of the orbit is e. The one-dimensional problem of r as a function of t can be transformed into the orbit, r as a function of polar angle, θ, by using the fact that L is a constant of the motion. The solution for r given above corresponds to an initial angle of zero, which initially yields the smallest radius defined to be the perihelion. The radial velocity as a function of angle is also solved for, and is

normalized to, the velocity of a circular orbit, v_c, in Equation (2.24). The effective potential is:

$$V_{eff} = GM[r_o/2r^2 - 1/r] \tag{2.25}$$

There is a competition between the attractive gravitational energy which goes inversely with radius and the repulsive centrifugal potential that goes as the inverse square of the radius. Depending on the relative strengths of these two effects, the orbits will be closed ellipses or open hyperbolae. The minimum of the potential occurs at r_o and is $-1/2(GM/r_o)$ which is the energy of a circular orbit. The energy of such an orbit is less than zero since the state is bound. It satisfies the virial theorem.

The solution for the orbit, r, and the orbital velocity, v, in circular velocity units depends on the energy of the orbit. The ellipse major axis is $r_o/|q|$ and the minor axis is the square root of the product of the major axis and r_o. The turning points for an ellipse where the radial velocity is zero are at:

$$\begin{aligned} x_1/r_o &= 1/q(-1+e) \\ x_2/r_o &= 1/q(-1-e) \end{aligned} \tag{2.26}$$

Therefore q fully defines the solutions; $q < -1$ means no solution, $q = -1$ is a circular orbit, q between -1 and zero is an elliptical orbit, $q =$ zero is a parabolic orbit, and q greater than zero is a hyperbolic orbit. Very elongated ellipses correspond to cometary orbits, which are bound to the sun but with very low energies and very large eccentricities.

The printout for the script "cm_kepl" is shown in Figure 2.34. The dialogue gives escape velocity, circular orbit parameters, and, for elliptical orbits, the major and minor axes, the turning point radii, the orbital period and the eccentricity.

For each choice of initial radius and energy, the effective potential energy and the (x, y) orbit is plotted, as well as the orbital time and orbital velocity as a function of the orbital angle. The effective potential, with contributions from the Sun and from the repulsive centrifugal potential, is shown in Figure 2.35. The potential for a

```
>> cm_kepl
  Program to compute solar system orbits, closed and open


Enter Initial Distance ro(AU): 1
L^2 is GMro/m^2 for a Circular Orbit at Radius ro
Escape Velocity (km/sec) = 42.3155 at ac (au) = 1
For circular orbit, v(km/sec) = 29.9216, Period (yr) = 0.992144
Enter Total Energy in Units of Circular Energy -G*M*m/2*ro, = q > -1: -0.5
Elliptical Orbits
For Elliptical Orbit, Major/Minor Axes (au) = 2 , 1.41421
For Elliptical Orbit, Turning Points in ro units = 0.585786 , 3.41421
For Elliptical Orbit, Orbital Period (yr) = 2.80621
For Elliptical Orbit, Eccentricity = 0.707107
```

Figure 2.34: Printout in the dialogue for the script "cm_kepl" for a user chosen example with an elliptical orbit.

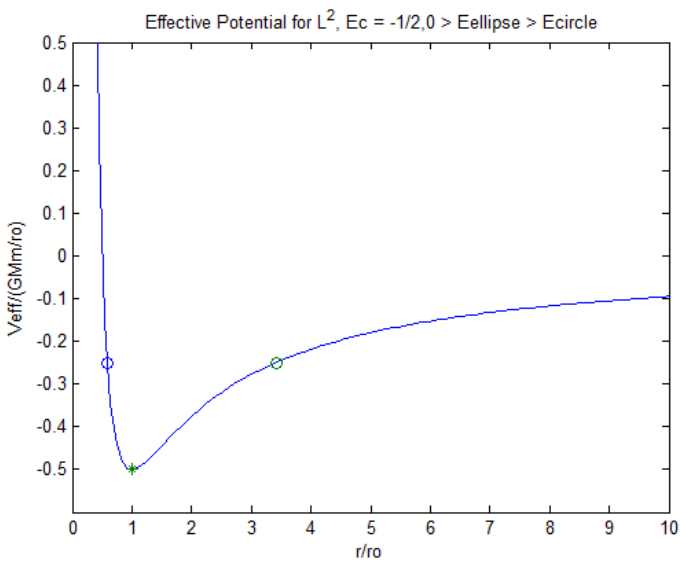


Figure 2.35: Effective potential energy for an elliptical orbit. The circular orbit is indicated by *, while the turning points for the chosen energy of, $q = -0.5$ are indicated by o.

circular orbit and the location of the turning points for the chosen energy are also shown. In the particular case the limits for the ellipse with an initial r of one AU and total energy of $-1/2$ is shown. The elliptical orbit is shown in Figure 2.36, where the major and minor

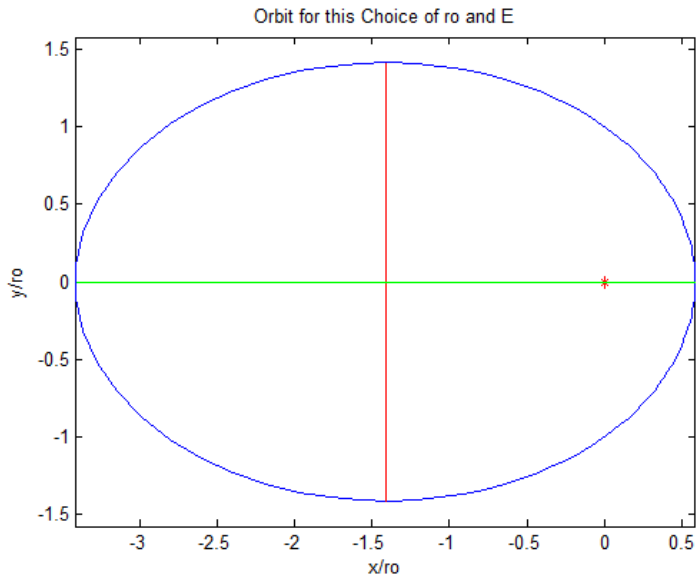


Figure 2.36: Elliptical orbit for the parameters printed in Figure 2.34. The distance between the turning points is the length of the elliptical major axis. The major and minor axes are shown in green and red.

axes are shown in green and red and the major axis length corresponds to the turning points printed in Figure 2.34.

The orbital velocity for the chosen elliptical orbit is shown in Figure 2.37. Clearly, the velocity is not constant as it is for a circular orbit. The orbital velocity is larger when the orbit is near the focus at the Sun and smaller, by a substantial factor, when the orbit is at a larger distance from the Sun. As indicated in the printout, the orbital eccentricity is 0.707 in this example.

The user can also make inputs to the menu which result in parabolic or hyperbolic orbits. At the boundary between a very eccentric ellipse and a hyperbola, the parabolic input of $q = 0$ results in the trajectory shown in Figure 2.38. A hyperbolic orbit is shown with $q = 0.015$ in Figure 2.39. The orbit rapidly approaches a straight line with an unbound trajectory which escapes the gravitational binding force. The straight lines are called the asymptotes of the orbit. In all the plots displayed here, the initial radius was 1 AU, the chosen L

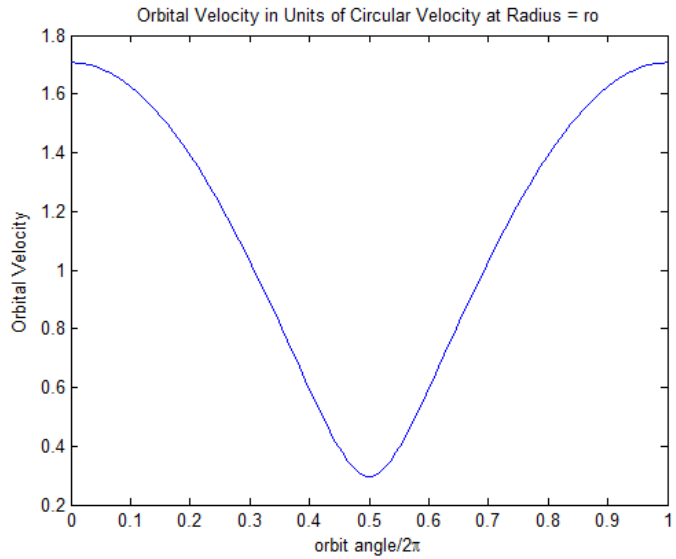


Figure 2.37: Orbital velocity as a function of the orbital angle for an elliptical orbit in units of the velocity for a circular orbit with radius r_o. Parameters are as quoted in Figure 2.34. Perihelion is at angle = 0, while aphelion is at an angle of 180°.

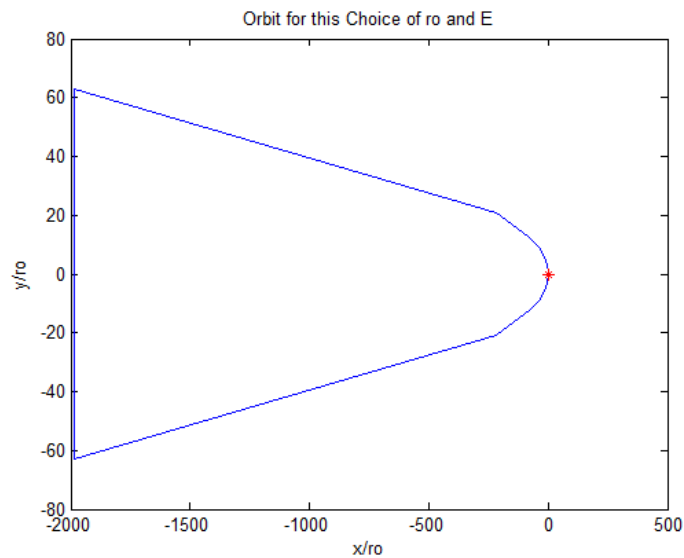


Figure 2.38: Trajectory (x, y) for a parabolic orbit with $q = 0$. The scale is too large to separate the force center and the perihelion.

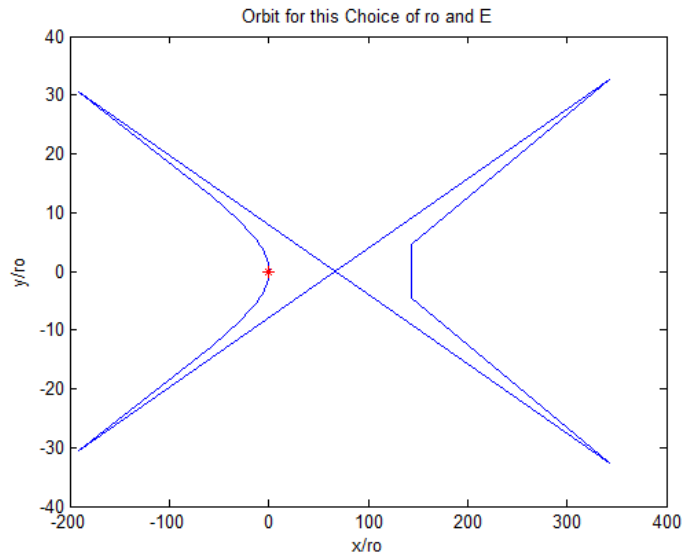


Figure 2.39: Hyperbolic orbit showing both the orbit and the asymptotes of the hyperbola for $q = 0.015$. The scale is too large to separate the force center and the perihelion. The turning point is at 0.5 AU. A repulsive force of gravity would have an orbit on the other segment of the hyperbolae.

value was that of a circular orbit at that radius and the orbital type was controlled by the conserved energy.

It is clear in making these plots that the bound state orbits are re-entrant. That is, they repeat in time and the orbital paths do not change with time averaged over many periods. This fact is unique to the fact that gravity is an inverse square force law.

2.14. Stable Orbits and Perihelion Advance

It is of interest to explore whether orbits which are perturbed are also stable. The script which was written to explore the question is "cm_circl_orbit". In fact, circular orbits are possible for most central forces obeying a power law in the radius from the force center. For a circular orbit, radius a, the centrifugal force is equal to the attractive force, $L^2 = a^3 F(a)$, which is a generalization of the Kepler formulation discussed already. The force law is assumed to be an attractive

power law, $F(r) \sim 1/r^n$, which competes with the repulsive centrifugal inverse cubed force. The restoring force for small perturbations, $r = a+x$, follows from Taylor expansions of the centrifugal force and the attractive force. The equation for a small displacement leads to simple harmonic motion as shown in Equation (2.27):

$$\begin{aligned} d^2x/d^2t &= [L^2/r^3 - 1/r^n] \sim [F(a)(3-n)]x/a \\ \omega_o^2 &= (F(a)/a)(3-n) = \omega^2(3-n) \end{aligned} \tag{2.27}$$

First, the perturbations on circular orbits for a given force law are plotted where the power is chosen by the user via a menu. The perturbed orbit follows x around $r = a$. The result is simple harmonic motion for small perturbations for the case $n < 3$, where the unperturbed circular frequency is $\omega^2 = F(a)/a$. Only the inverse square law, $n = 2$, has closed orbits which repeat, as mentioned previously. The perturbed orbit for $n = 1$ is shown in Figure 2.40. Note that

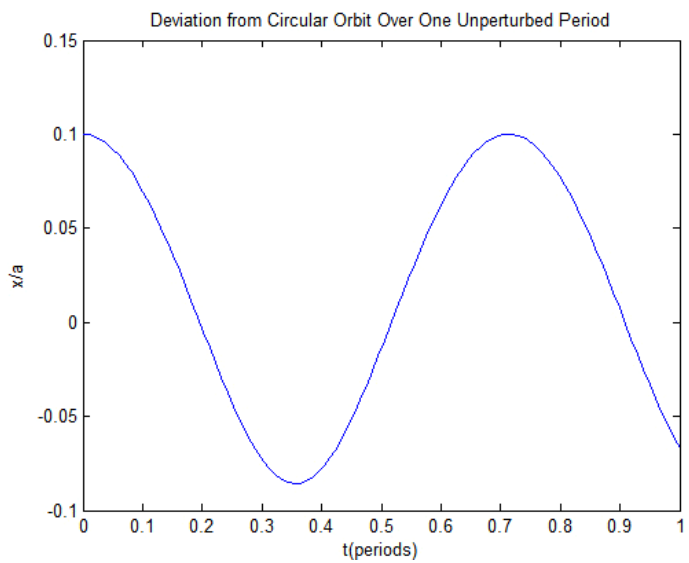


Figure 2.40: Perturbed orbit in the case of a $1/r$ force law.

the perturbation does not close with the basic orbital frequency. The script uses the MATLAB function "ode45" to compute the trajectory numerically.

Therefore, any failure to close a solar orbit means that gravity is not a pure inverse square force law, contradicting Newton. Indeed, this was known for Mercury (the advance of the perihelion) and brilliantly confirmed Einstein's general theory of relativity (GR). That theory predicted a small, relativistic addition to the law of gravity with an inverse fourth power. The short range of the additional force meant that the effect was large only for planets at small orbital radii. The user has a choice of the amount of added force and the resulting orbit for a coefficient of 0.2 is shown in Figure 2.41, while the

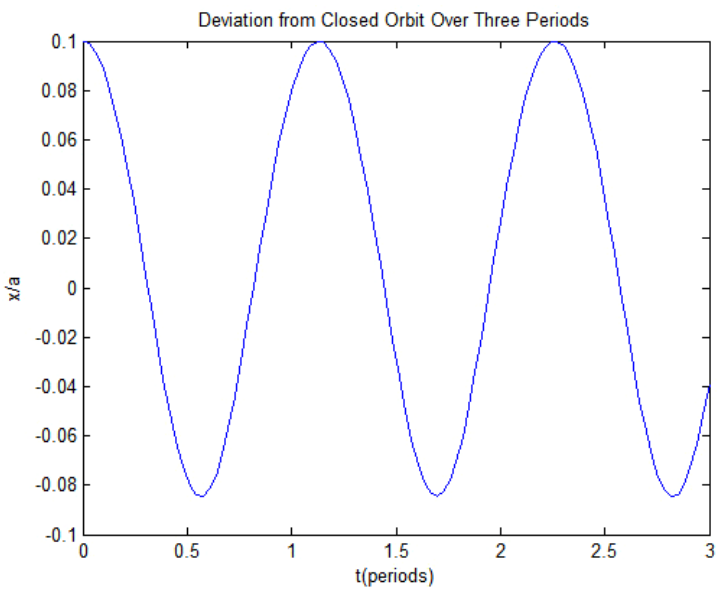


Figure 2.41: Advancing perihelion for a 20% addition of an inverse fourth power force.

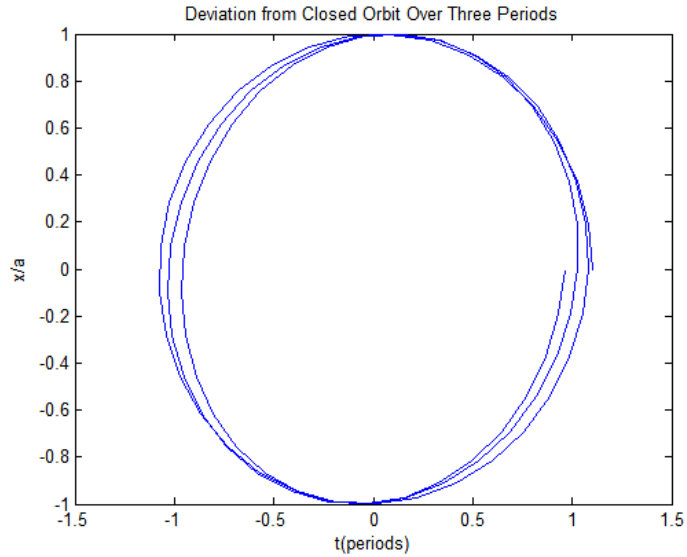


Figure 2.42: Perturbed orbit for three circular periods showing the advance of the perihelion with time.

advancing perihelion is shown more graphically in Figure 2.42 for numerical integration of three orbits. Note that the effect is vastly exaggerated for visual purposes. The actual advance is only 43 seconds of arc per century for Mercury.

Chapter 3

Electromagnetism

> "Little you know the subtle electric fire that for your sake is playing within me."
>
> — **Walt Whitman**

> "It was like bouncing tennis balls off a mystery piece of furniture and deducing, from the direction in which the balls ricocheted, whether it was a chair or a table or a Welsh dresser."
>
> — **Marcus Chown**

There are many possible candidates for numerical results in electromagnetism. The first exercises shown here are in the realm of statics which are then followed by dynamical demonstrations. In this section, the symbol E denotes the electric field, while in others E denotes the non-relativistic total energy or the relativistic total mass-energy. The reader will be warned about these unfortunate, but customary, changes of notation.

3.1. Electric Potential for Point Charges

The analytic solution for a point charge is well known. Since electromagnetism is a linear theory, the field and potential for a collection of point charges follows by superposition of the point charge solution. The supplied script is "Point_Elec_Static", which finds the potential for a set of charges which are user supplied giving (x, y) positions and individual charges. The electric field is derived from the potential using the MATLAB utility "gradient".

The results for a dipole configuration of $[x\ y\ q] = [-10\ 0\ 200]$ and $[10\ 0\ -200]$ are shown for the potential in Figure 3.1 and for the x component of the field in Figure 3.2.

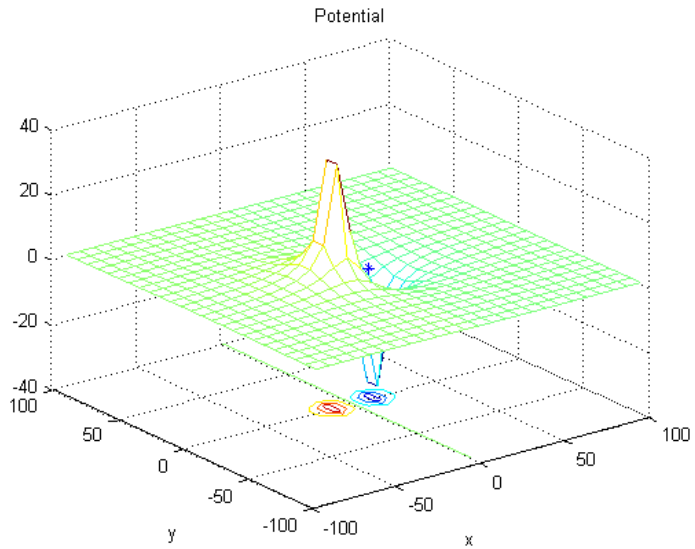


Figure 3.1: Plot of the dipole potential using the MATLAB function "meshc" which displays both the values of the potential and the equipotential contours. The typical dipole pattern of the contours is evident.

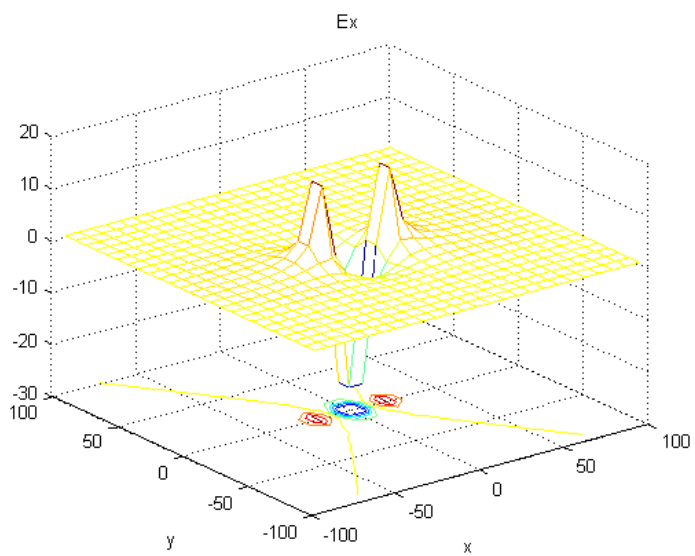


Figure 3.2: Field component Ex for the dipole configuration. The dipole field between the two charges is evidently large, as is the rapid falloff with distance r from the origin.

As can be seen from Figure 3.2, the field Ex is strongest between the two charges and then falls off rapidly as the distance to the observation point increases.

3.2. Image Charge for a Grounded Sphere

There are many ways to solve boundary value problems in electromagnetism. One method is to place an image charge which provides the proper boundary conditions; for example, the vanishing of the electric field parallel to a conductor surface. A simple example of this technique is to look at an infinite grounded plane. The boundary conditions for a point charge placed near the plane are satisfied by creating an "image charge" of the same magnitude, but different sign placed in a virtual location behind the plane at the same distance from the plane as the point charge. That choice makes the plane an equipotential and the electric field normal at the plane.

In the special case of a grounded sphere of radius a under the influence of an external charge, q, placed at $z = c$, the image charge, q_i, and image location, z_i, are;

$$\begin{aligned} q_i &= -aq/c \\ z_i &= a^2/c \end{aligned} \tag{3.1}$$

This configuration induces a charge density on the sphere which follows from the requirement that the field has only a normal component at the radius a. For the numerical case where $a = q = 1$, the script "Image_Charge_Sphere2" asks for a z location of the charge. A "movie" for the potential and the field is then shown for image charges at image locations from $z = 0.1$ to 0.9. The user can then watch the boundary conditions approach the solution. The plots for an image location of $z = 0.7$ are shown in Figure 3.3. The equipotential, in blue, is close to the locus of the circle, shown in green. The external and internal (image) charges are indicated by a red *. The electric field at that image charge location is almost normal to the sphere as seen in the plot on the right.

The induced charge density is shown in Figure 3.4 for several choices of charge location, $z = c$. The induced charge as a function of

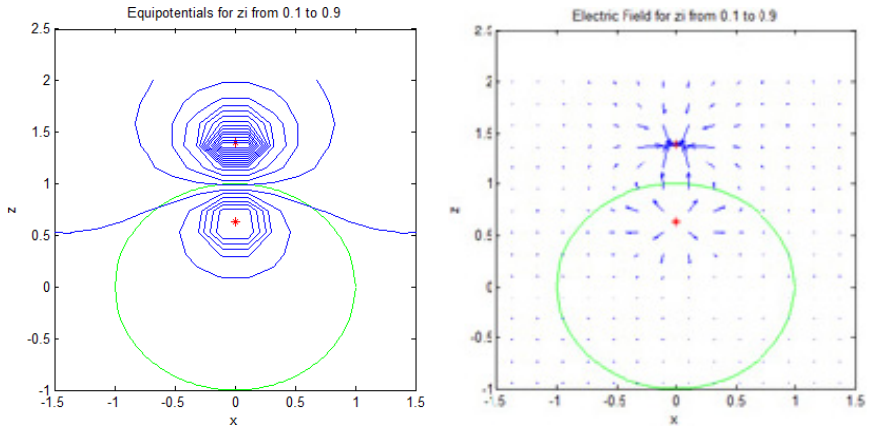


Figure 3.3: Equipotential contours (left) and electric field (right) for an external charge located at $z = 1.4$ and with image charge at $z = 0.7$ for a sphere with radius 1.

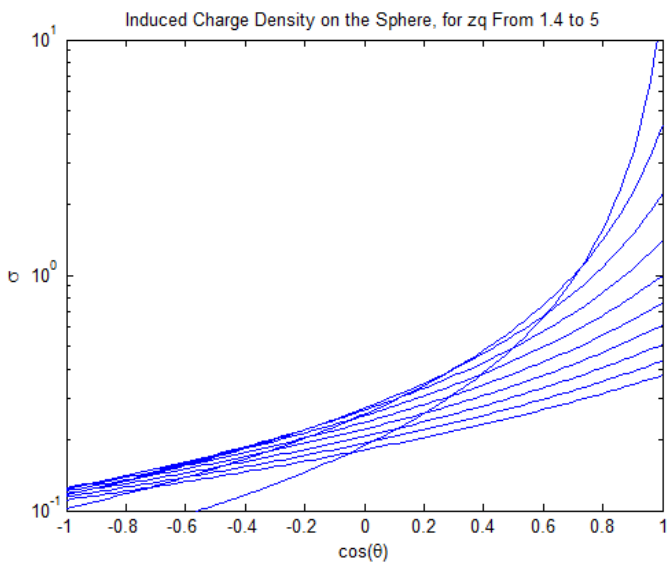


Figure 3.4: Induced surface charge density as a function of the observation angle with respect to the external charge located at $z = c$ for several values of c/a, varying from 1.4 to 5.

angle for cases where c/a varies from 1.4 to 5 indicates how strongly the result depends on that ratio. The charge scale is logarithmic. The script provides several choices of c to show the dependence. Also in the script "Image_Ch_Sphere2", the user menu allows for any choice of c in order to explore other values.

3.3. Magnetic Current Loop

Another static example comes from magnetostatics, in this case the magnetic field of a current loop. At distances much greater than the size of the current loop, the field approaches dipole behavior and falls as the inverse cube of the radius. The exact solutions at any distance are elliptic integrals, which, although they are available in MATLAB, add little to the intuition for the problem which is to be built up. Because of that, the solution is achieved numerically.

The script to display the magnetic fields far from the loop is contained in "Current_Loop". The printout from that exercise is shown in Figure 3.5. The field is expected at large values of r/a, where a is the loop radius, to go as:

$$\begin{aligned} B_r &\to \cos\theta/r^3 \\ B_\theta &\to \sin\theta/r^3 \end{aligned} \tag{3.2}$$

```
>> Current_Loop
  Program to plot B field of a current loop, far from the loop
  numerically - all point using Biot-Savert


 B Field for a Current Loop, Radius a, for r >> a
 Dipole Moment = pi*I*a^2
 Current Loop in x,y Plane. Theta is the Angle w.r.t. the z Axis
 B Field for All Points - Biot-Savert
```

Figure 3.5: Printout for the current loop demonstration which defines the coordinates of the demonstration.

The field has both radial and angular components. That reflects the fact that there are no magnetic "charges" so that the field must always close on itself.

The produced field, plotted as a function of coordinates scaled to the loop radius, is plotted for the radial component in Figure 3.6. At large r/a, the enhancement of the radial field in the z direction, expected from the behavior shown in Equation (3.2), is evident. The polar angle magnetic field is also calculated and displayed for the user to inspect.

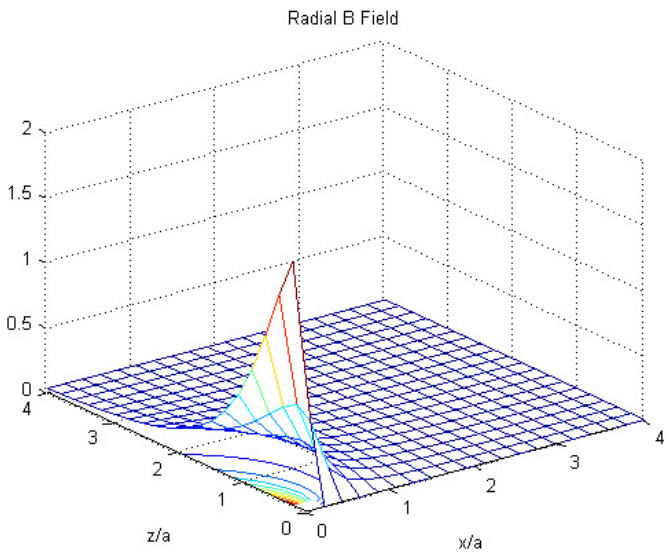


Figure 3.6: Radial magnetic field for a current loop in the approximation that $r \gg a$. The contours display the rapid falloff of the field with r and the dipole enhancement at larger z/a values.

The complete solution is most easily approached using the Biot-Savert law which relates the differential source current to the differential field increment:

$$d\vec{B} \sim (d\vec{x}\ x\ d\vec{r})/r^3 \tag{3.3}$$

The current element is located at $\vec{x}$, while $\vec{r}$ is the vector from the location of the current to the point where the field is evaluated. Integrating over all source points of the field in the case of the current loop, the z field is shown in Figure 3.7. Far from the current loop the field approaches a dipole pattern. The integral can be done explicitly

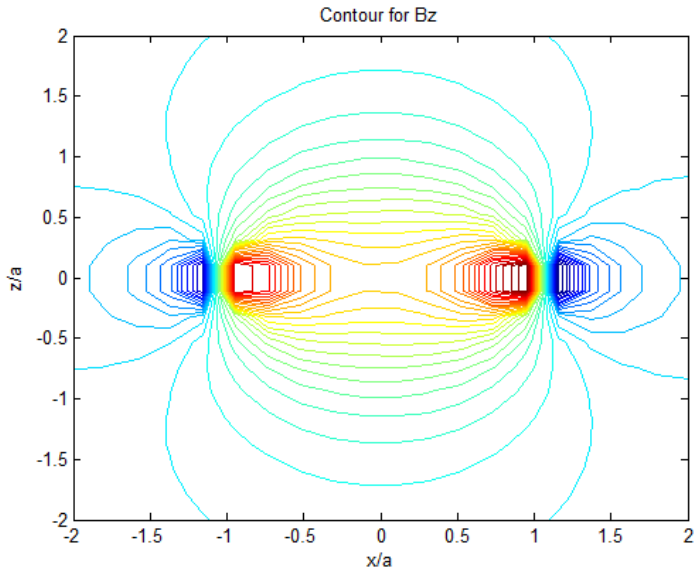


Figure 3.7: Contour of the z field as a function of x and z. The loop has radius = 1. The evolution to a dipole field at large z values is evident.

and the exact solution appears in the script. It is cumbersome and will not be given in this text.

3.4. Helmholtz Coil

Having solved the problem of a current loop at z equal to zero, it is fairly simple to extend the solution to locations at non-zero values of z. The result is the analogue of superimposing electric point charge solutions. The simplest case is that of two current loops with current flow such as to reinforce the field between the loops. The script is "Helmholtz_Coil".

The user chooses the distance between the loops, where the radius is taken to be equal to one. The field contours are returned for that choice. Compare the contours of two loops to one loop. The coils are used to provide a reasonably constant field between the loops over a reasonably large field volume. The contour plot for Bz appears in Figure 3.8, while the surface plot is shown in Figure 3.9.

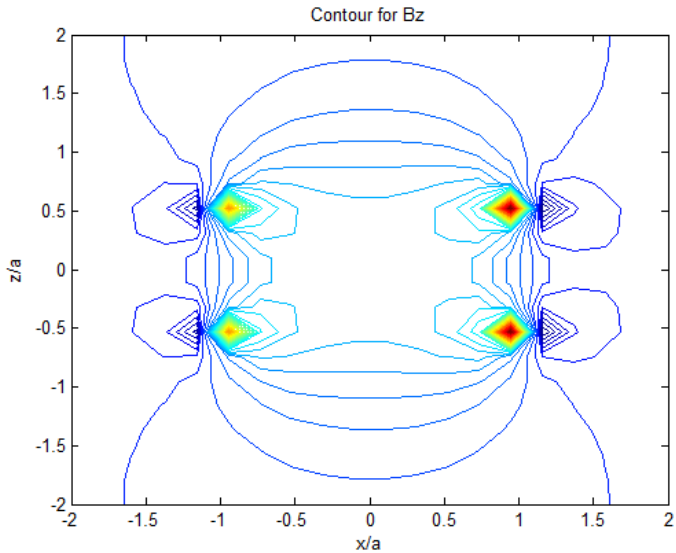


Figure 3.8: Contour plot for Bz due to two current loops separated by a distance equal to their radius.

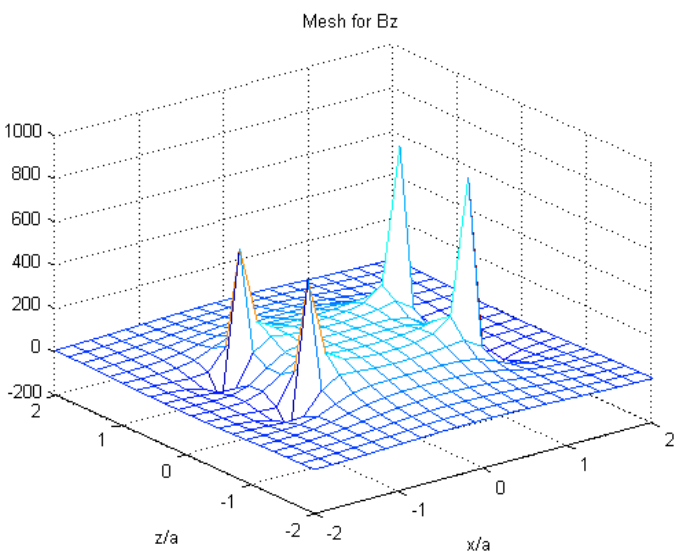


Figure 3.9: Surface mesh plot for the field contours shown in Figure 3.8. Note that a region of roughly constant field between the current loops has been established.

3.5. Magnetic Shielding

It is well known that shielding from electric fields is possible in the static case for an interior volume, by using grounded and closed conductors (Section 3.1). For high frequencies, the skin depth in good conductors is quite small so that time varying electric fields can also be effectively shielded against, as a glance at your microwave window confirms with its mesh of thin metallic shielding.

What about magnetic fields? It is important to shield objects like compasses from stray fields. Consider a prototype shield consisting of a thin shell of high magnetic permeability metal of inner radius a and outer radius b immersed in a magnetic field, B_o, directed along the z axis.

This problem can be solved by using standard methods in magnetostatics by expanding in powers of the Legendre polynomials. In fact, only the first power of the cosine of the polar angle is needed, and radial factors going linearly with r and as the inverse square of r are the only ones needed. The magnetic potentials, Φ are:

$$\begin{aligned} \Phi_{out} &= -B_o r\cos\theta + (\alpha/r^2)\cos\theta \\ \Phi_{in} &= \delta r\cos\theta \\ \Phi_{\mu} &= \beta r\cos\theta + (\gamma/r^2)\cos\theta \end{aligned} \tag{3.4}$$

where "out" refers to $r > b$, "in" refers to $r < a$, and μ refers to the permeability value of the metal shield existing for $a < r < b$. The analytic results for the parameters are somewhat tedious and appear in the script "Magnetic_Shield", which serves as a reference for the interested user.

The user menu asks for the value of the permeability and the geometry, b/a. For the values 10 and 1.2, respectively, the potential is shown in Figure 3.10. Far from the sphere the uniform field is visible. Inside the sphere, the magnetic field is much reduced, and magnetic shielding is achieved. This problem is of great practical interest, and the user is encouraged to explore different choices of the shielding parameters. For thick shields, the ratio of exterior to interior fields is approximately $9/2(\mu)$. The boundaries of the spherical shell are shown as red and green (inner and outer) circles in the figures.

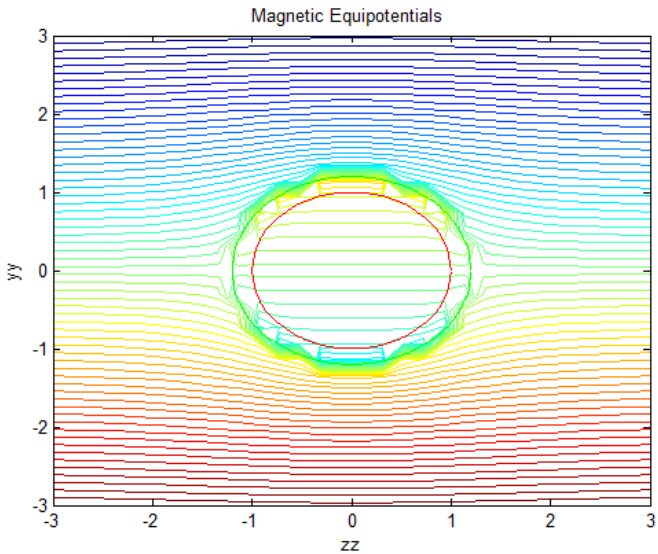


Figure 3.10: Potential for a metallic sphere immersed in a uniform magnetic field oriented along the z axis for $b/a = 1.2$ and $\mu = 10$.

To derive the fields, the MATLAB function "gradient" is used again but, in addition, the MATLAB display tool "quiver" lets the user observe the magnitude and direction of the field as indicated by arrows having both length and orientation. The fields are displayed in Figure 3.11. Thicker shielding or shielding materials with larger permeability reduces the fields. This effect is shown in Figure 3.12 for the same geometry, $b/a = 1.2$, but with a permeability of 100 in this case. The field is reduced approximately ten fold, as expected.

3.6. Potentials and Complex Variables

Two-dimensional potential problems can also be solved using complex variable techniques. In two dimensions, any analytic function of the complex variable, z, will satisfy the Laplace equation for both the real and the imaginary parts of the function. Therefore, knowing the result in a simple case and finding the appropriate transformation, one can find the solution in the more complex geometric situation specified by the transformation.

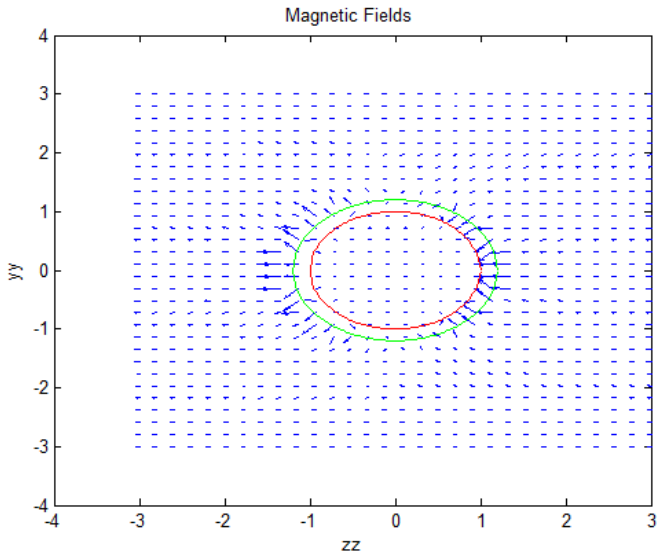


Figure 3.11: Magnetic fields for the potential shown in Figure 3.10 is reduced. The interior field is much attenuated.

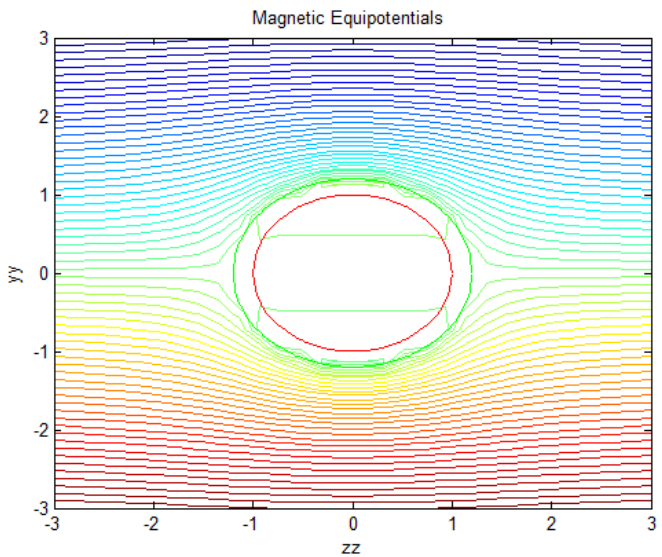


Figure 3.12: Potential for the geometry of Figure 3.10 but with a permeability ten times larger. The shell boundaries are shown in red and green.

Several examples are provided by the script "Laplace_z". The user has five different transformations to choose from. In each case, the potential contours, the x and y fields, found using the "gradient" tool, and the "quiver" plot of the fields in two dimensions are supplied. The dialogue is shown for all five choices in Figure 3.13.

```
>> Laplace_z
  Laplace_z - Laplace Electrostatics - BV, Laplace Eq. Potential, E Field
  Use Complex Variables


Complex Potential  - Modulus of psi is the Potential V, grad(V) = Electric Field
Enter a Number for an Electrostatic Example- 1 to 5: 1
Point Charge at the Origin - 2D
Enter a Number for an Electrostatic Example- 1 to 5: 2
Two Charges - Image Charge for Conducting Plane
Enter a Number for an Electrostatic Example- 1 to 5: 3
Two Half Circles - Interior Solution
Enter a Number for an Electrostatic Example- 1 to 5: 4
Two Coaxial Cylinders at V = 0,1 - Interior Solution
Enter a Number for an Electrostatic Example- 1 to 5: 5
Two Line Charges at V = 0,1 - 1/2 Plane
```

Figure 3.13: Printout of "Laplace_z" for a user making all five choices sequentially.

A plot of the fields for a charge at $x = 0$ and $y = -0.5$, with a grounded conductor at $y = 0$ is displayed in Figure 3.14. The upper half plane has no field, and the field is perpendicular to the grounded plane at $y = 0$. The fields in the case of conductors at $r = 1$, where the voltage is V for $y > 0$ on the circle and $-V$ for $y < 0$ are shown in Figure 3.15. Only the internal fields are plotted. The fields are clearly strongest at the $y = 0$ boundary where the voltage gradient is largest. As a last example, the equipotentials in the upper half plane for the case of line charges on the x axis with $V = 0$ for $x < 0$ and $y = 0$ and $V = 1/2$ for $x > 0$ and $y = 0$ are given in Figure 3.16. The field peaks at $x = 0$ where the voltage gradient is largest.

3.7. Numerical Solution — Laplace Equation

More complex topologies than those given in the last section can be solved numerically. A Cartesian version of the solution in two dimensions is supplied by "EM_Laplace_Test2". MATLAB has tools

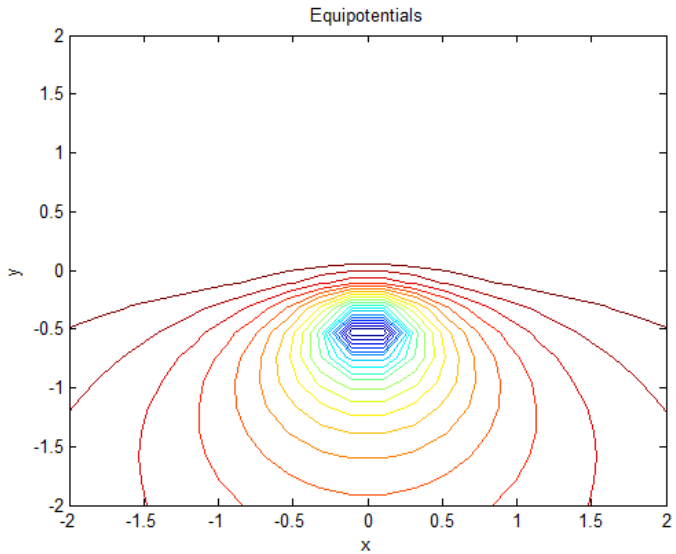


Figure 3.14: Equipotentials in the case of a charge in the lower half plane at $(0, -0.5)$ and a grounded conductor at $y = 0$.

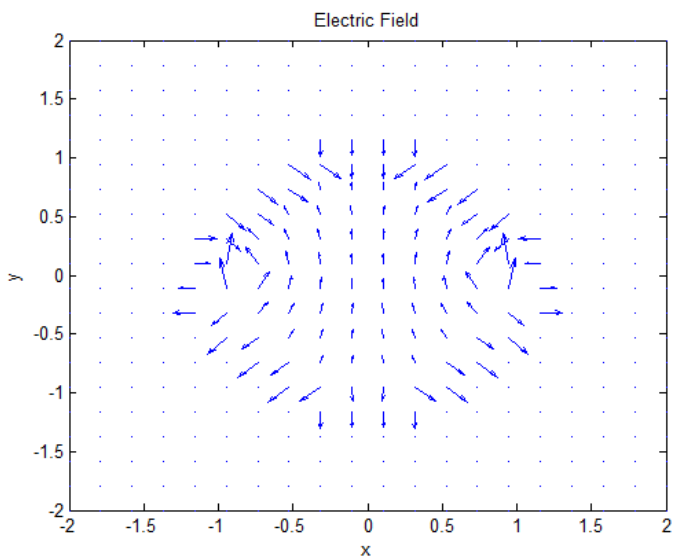


Figure 3.15: Interior fields for the case of a half circle at potential V for $y > 0$ and $-V$ for $y < 0$.

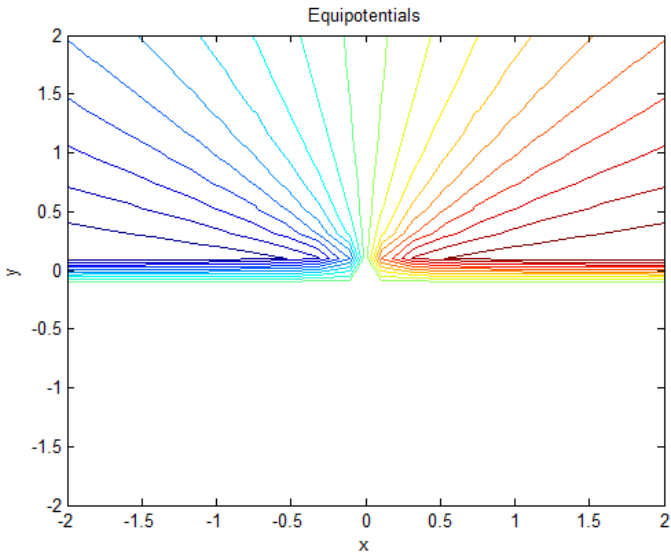


Figure 3.16: Equipotentials in the case where $V = 0$ at $y = 0$ and $x < 0$ and $V = 1/2$ for $y = 0$ and $x > 0$.

to solve one-dimensional, partial differential equations, but higher dimensionality problems with specified boundary conditions are not yet available. The script which is available uses the Gauss-Seidel method, given explicitly in the printout, to solve a two dimensional problem on a grid of points. The expression given in the printout is simply the finite difference expression for the vanishing of the second partial derivative in x and y on a Cartesian grid.

The user dialogue asks for the grid size and then the voltages on the four boundary surfaces of the square. The boundary values may be constants or functions of x or y, depending on the boundary.

An example shown here is for a 25×25 grid with voltages on the left, right, top and bottom as defined in the printout given in Figure 3.17. The basic finite difference grid computation of the solution also appears in the printout. The whole grid is iterated ten times using the Gauss-Seidel method. The script returns the potential, the x and y fields, using the "gradient" tool, and the "quiver" plot of the combined vector electric field. The potential for the example above appears in Figure 3.18, the fields in Figure 3.19.

```
>> EM_Laplace_Test2
  Solve static Laplace Eq. using Gauss Seidel, Cartesian, Boundary Voltages


Solve Finite Difference Eq: 4Vi,j = Vi,j+1 + Vi,j-1 + Vi+1,j + Vi-1,j
Input the Square Grid Number of Points, 0<x<1,0<y<1: 25
Input the Voltage Function on the Left Boundary,f(y)
: sin(pi*y)
Input the Voltage Function on the Right Boundary, f(y)
: -sin(pi*y)
Input the Voltage Function on the Top Boundary, f(x)
: sin(pi*x)
Input the Voltage Function on the Bottom Boundary, f(x)
: -sin(pi*x)
```

Figure 3.17: Printout of the dialogue and input for "EM_Laplace_Test".

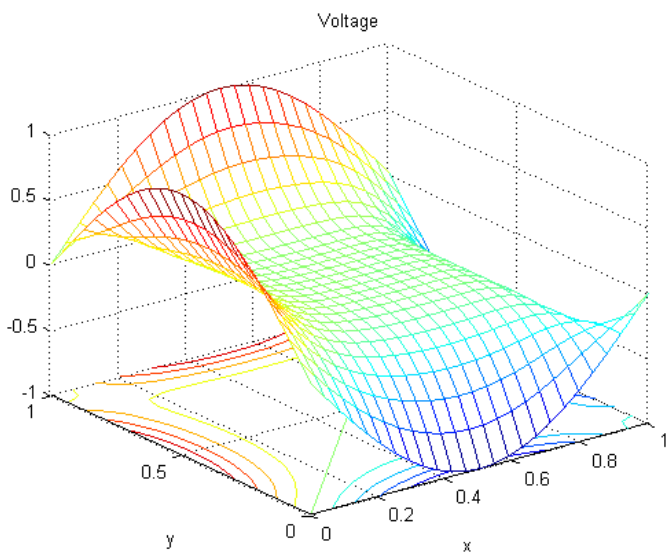


Figure 3.18: The interior solution for the potential defined by the boundary values input in the example above, Figure 3.17.

3.8. Numerical Solution — Poisson Equation

The Laplace equation treated in the last section holds in the absence of sources of the field. The Poisson equation, Equation (3.5), applies in the case where the solution is defined by actual sources, rather than

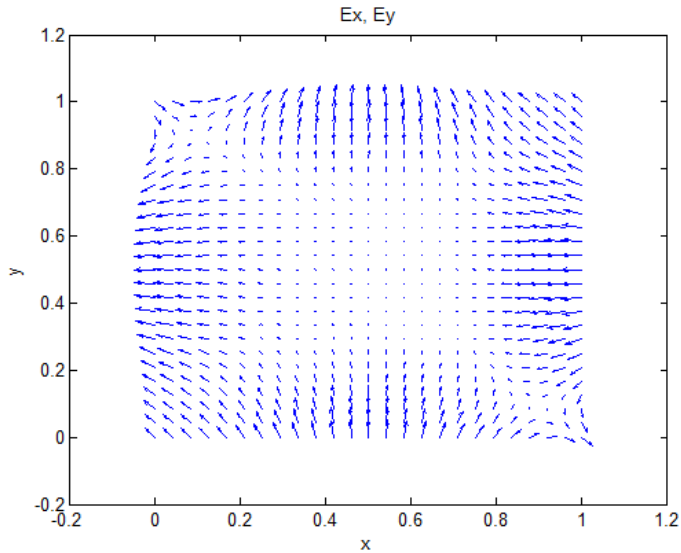


Figure 3.19: x and y electric fields for the potential shown in Figure 3.18.

fixed boundary conditions enclosing a space without any sources. The solution for the potential Φ is, rather, defined by the location and strength of the sources as defined by the charge density ρ which exists in the interior space.

$$\nabla^2 \Phi = \rho \tag{3.5}$$

The numerical solution in two Cartesian dimensions is encapsulated in "EM_Poisson_Test" which wraps the MATLAB tools for fast Fourier transforms, "FFT", "fft2" and "ifft2". The Gauss-Seidel method could be used with a simple extension, but the method adopted here uses fast Fourier transforms, because script for them is available in MATLAB. These tools allow the potential to be solved for, Equation (3.6), in the transformed space where the charge density is transformed using "fft2". The Fourier transform of the potential is then inverted back to the original space at the fixed grid points using "ifft2". The grid spacing is δ and the number of grid points, assuming a square grid, is N^2. Note that here the boundary conditions on the space require the potential to vanish, so that the boundaries should be far enough away from the interior sources so

as not to unduly influence the potential near them.

$$\Phi_{i,j}^{FT} = \rho_{i,j}^{FT}\delta^2/[2(\cos x_i + \cos y_j - 2)]$$
$$x_i = 2\pi i/N, \qquad y_i = 2\pi j/N \tag{3.6}$$

The dialogue with the user in the example of a model of a capacitor is shown in Figure 3.20.

```
>> EM_Poisson_Test
  Solve static Poisson Eq. using FFT for Periodic BC, Cartesian - MATLAB


Solve Finite Difference Eq: 4Vi,j = Vi,j+1 + Vi,j-1 + Vi+1,j + Vi-1,j - rhoi,j*del*del
Input the Square Grid Number of Points, 0<x<1,0<y<1: 50
Input Number of Point Charges: 0
Input Number of Charged Rectangles: 2

 For Rectangle 1
Enter Top Right [x,y] Position: [0.8 0.7]
Enter Bottom Left [x,y] Position: [0.2 0.6]
Input Voltage on Rectangular Charge: 100

 For Rectangle 2
Enter Top Right [x,y] Position: [0.8 0.4]
Enter Bottom Left [x,y] Position: [0.2 0.3]
Input Voltage on Rectangular Charge: -100
 FT of Diff eq for Vij is rhoij*del*del/[2(cos(2*pi*i/N) + cos(2*pi*j/N)-2)]
```

Figure 3.20: Dialogue for an example of the use of "EM_Poisson_Test" for the capacitor model.

The resulting plots are the potential, the x and y electric fields and the "quiver" display of the vector electric field. The voltage potential appears in Figure 3.21. The rapid gradient between the plates of the capacitor is clearly evident as are the non-zero values of the fields outside the regions between the plates.

The electric vector field derived from the solution shown in Figure 3.21 appears in Figure 3.22. The strong field between the capacitor plates is a major feature as are the fringe fields at the ends of the plates and the rapid falloff at large distances from the center of the object.

3.9. Light Pressure and Solar Sailing

Light exerts pressure as the photons collide and transfer momentum to the struck object. Understanding these phenomena, science fiction

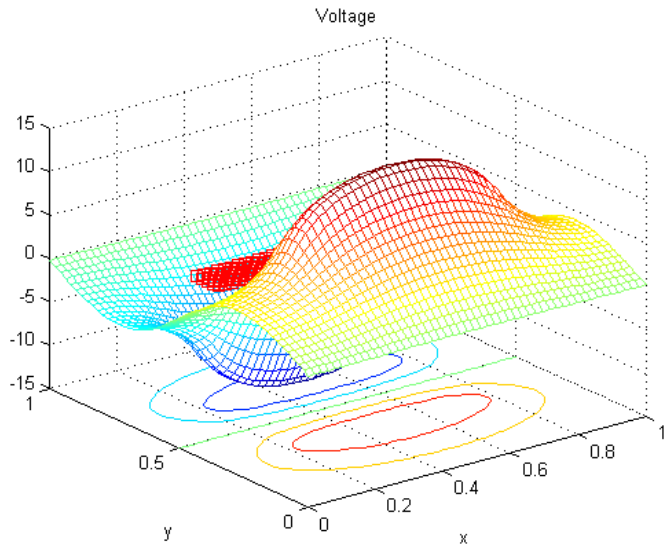


Figure 3.21: Voltage distribution for the capacitor example using the numerical Poisson equation solver. The plate locations are indicated by red squares.

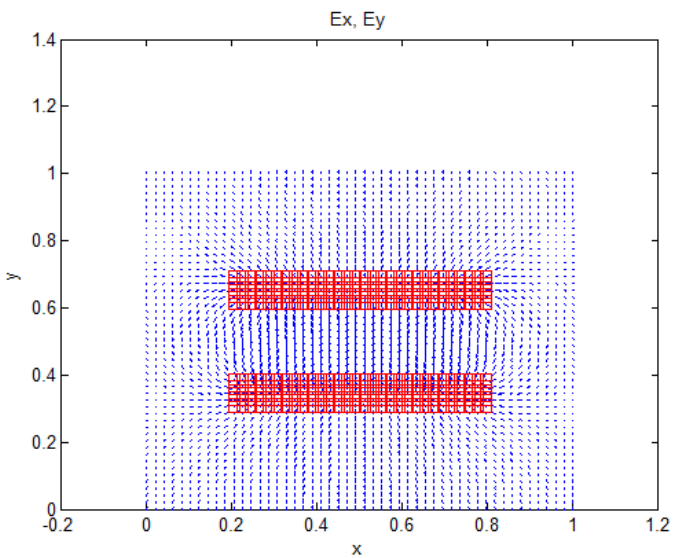


Figure 3.22: Electric field for the capacitor example. The field is strong between the plates, as expected. The plates are constructed using red grid sized pixels.

writers have created the idea of solar sailing within the solar system and for exploration to nearby stars. Solar sails have actually been deployed in space by U.S. funding agencies. Here is an idea whose time has come. The pressure, p, for perfect reflection due to a point source at distance r with luminosity L is:

$$p = (2L/c)/4\pi r^2 \tag{3.7}$$

The basic equation of motion follows from the expression for light pressure and the competing attraction of the sun. If a small payload is ignored, the acceleration depends on the luminosity of the sun, L_o, the distance, r, from the sun, the density, ρ, and thickness, d, of the solar sail. With a finite payload, the expression is modified by the replacement, $\rho d \rightarrow \rho d + m_p/A_s$ where m_p is the payload mass and A_s is the sail area.

$$d^2r/d^2t = [2L_o/(4\pi c\rho d) - GM]/r^2 \tag{3.8}$$

The acceleration needed for a stellar voyage must overcome the pull of gravity from the sun, with a mass, M, and propel a "sail". The acceleration does not depend on the size of the "sail", but payload considerations argue for a large construction of light material, such as very thin Mylar. Perfect reflection is assumed. The acceleration does not depend on the area of the sail, only density and thickness, and varies inversely with the square of the distance from the Sun.

The script is "Solar_Sail2" and the printout of a user dialogue is shown in Figure 3.23. The specific sail is sufficiently light and thin to overcome the acceleration of the sun. If it is not, the sail will fall into the sun and the "ode45" code will generate errors. The equation is not solvable explicitly, so "ode45" is used in the script to provide a numerical result. The acceleration falls off rapidly with radius, which means the velocity builds up quickly. The approximate time to go five light years is 52,900 years with a small payload of 314 kg. Note that a launch closer to the sun is better. At a starting radius of 0.01 AU, the same trip takes only 5244 years, a gain of a factor of ten. The user is encouraged to try different launch positions, sail areas, thicknesses of Mylar sail, and payloads. The velocity builds up quickly over the first year in this example as seen in Figure 3.24. After ten years,

```
>> Solar_Sail2
  solar sailing using the momentum of light


Light Pressure is P = (2*L)/(4*pi*r*r*c)
Light Pressure at 1 AU = 9.19562e-06 (nt/m^2)
Enter Launch Radius (au): 1
Escape Velocity (m/sec) = 41106.4
Initial Solar Accel (m/sec^2) = 0.00563244
Enter Solar Sail Thickness in um : 1
Enter Circular Solar Sail Radius in km : 1
Enter Payload Ratio x 10^-6 : 0.1
Sail Mass (kg) = 3.14159e+09 , Payload Mass (kg) = 314.159
Initial Sail Acceleration(m/sec^2) = 0.00835965
Velocity After 10 Years (m/sec) = 28354.1
Distance After 10 Years (AU) = 57.9171
Time (yr) to go 5 Light Years Assuming Rapid Terminal Velocity = 52902.4
```

Figure 3.23: Dialogue for a particular sail and launch position.

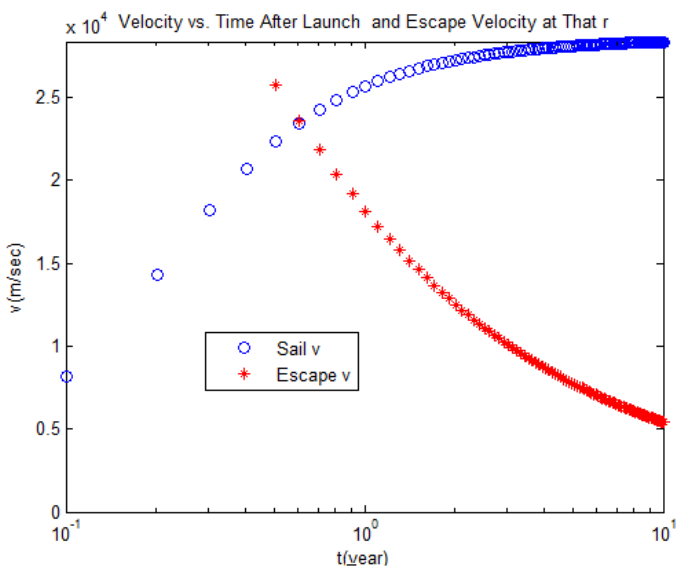


Figure 3.24: Velocity as a function of time for the first year after the launch of a solar sail from an initial position at 1.0 AU as defined in Figure 3.23.

the ship is at 58 AU and going 28 km/sec. The ship has attained escape velocity less than one year after the launch. This behavior is different from that of a rocket which quickly attains its final velocity, as discussed previously.

3.10. Motion in Electric and Magnetic Fields

To continue with the dynamics, motion in combined magnetic and electric fields by charged particles is explored. The script which is used is "ExB_ODE_NR". The motion is assumed to be non-relativistic and the Lorentz force equation is integrated numerically using the MATLAB "ode45" tool. Units with charge, q, and mass, m, equal to one are used. The initial position is at $x = y = z = 0$. The initial velocity is input by the user as is the magnetic field magnitude, assumed to be oriented along the z axis and the x, y, and z components of the electric field. The Lorentz force equations are:

$$d^2\vec{x}/d^2t = q/m[\vec{E} + (d\vec{x}/dt)x\vec{B}] \tag{3.9}$$

The trajectory in three dimensions is provided by the script. There are four plots which are produced. The first is the x, y, and z velocity as a function of time, the second is the x vs y velocity and the third is the x, y and z position as a function of time. Finally, the fourth is a plot of the (x, y) position, where a movie of the time development of (x, y) is also shown in order to get a feeling about the velocities in the x and y directions for the specified setup of fields and initial velocities.

The results for an example with $E = [1\ 0\ 0]$, $B = 1$ and $v = [1\ 1\ 1]$ are shown in Figures 3.25, 3.26 and 3.27. The electric field here is only along x, so that the velocity along z (the magnetic field direction) is constant. The velocities in the (x, y) plane rotate due to the magnetic field, with an increasing velocity along the x direction due to the electric field, as observed in Figure 3.25. The user can also look at the special cases where there is only an electric field or a magnetic field. Another possibility is to vary the initial velocity conditions.

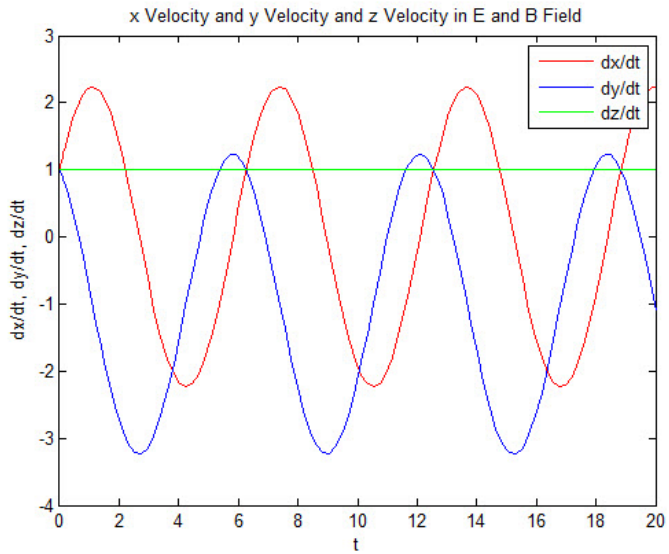


Figure 3.25: The three velocity components as a function of time. The z component of velocity is constant because the magnetic force is absent in this direction and there is no electric field along z in this example.

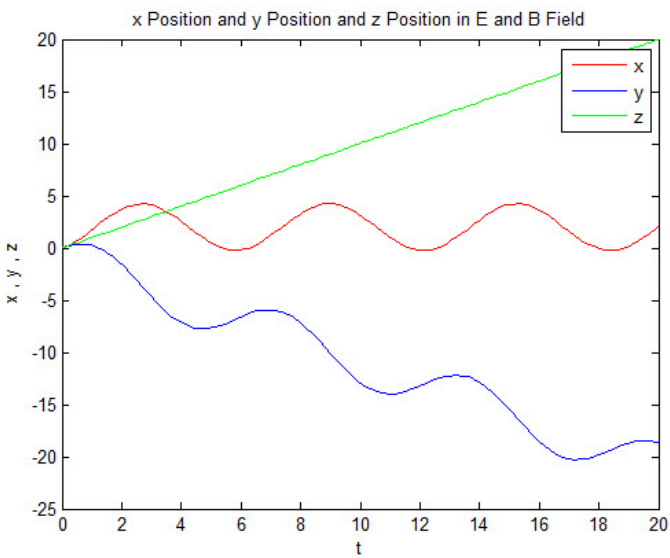


Figure 3.26: The three position components as a function of time. The basic circular motion of the x and y positions is evident.

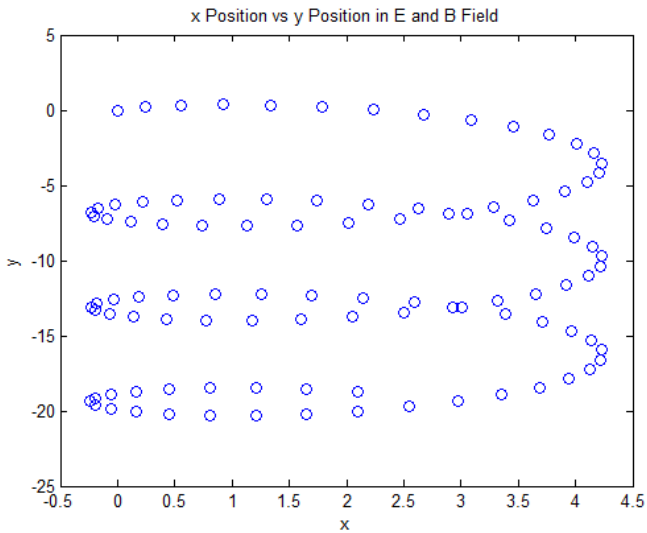


Figure 3.27: Movie frames of the trajectory of (x, y) as it evolves in time.

Because of the magnetic field, the particle tends to have a circular (x, y) trajectory with a constant radius in this non-relativistic case. The completely relativistic case will be treated in a distinct script to be explored later in the text. In the present case, the Larmor frequency for circular motion does not depend on the momentum, and is $\omega = qB/m$.

3.11. The Cyclotron

The cyclotron is a device to accelerate charged particles in electric and magnetic fields. The magnetic field causes the particle to rotate in a circular orbit with an angular frequency, ω, and radius, r. For non-relativistic particles, the frequency depends only on the magnetic field, the charge and the mass of the particle. For protons, it is 95.5 MHz for a field of 1 T or 10 kG. As the energy increases, the radius increases proportional to the velocity perpendicular to the magnetic field, v_T.

$$\begin{aligned} \omega &= qB/m \\ r &= v_T/\omega \end{aligned} \tag{3.10}$$

A schematic of a cyclotron appears in Figure 3.28. The magnetic field is supplied by Helmholtz coils which are not shown but which make a field perpendicular to the plane of the figure. The two half circles, called "dees," are charged + and − so as to accelerate the protons. However, it is clear they need to be reversed in polarity for each half revolution, at the frequency quoted above.

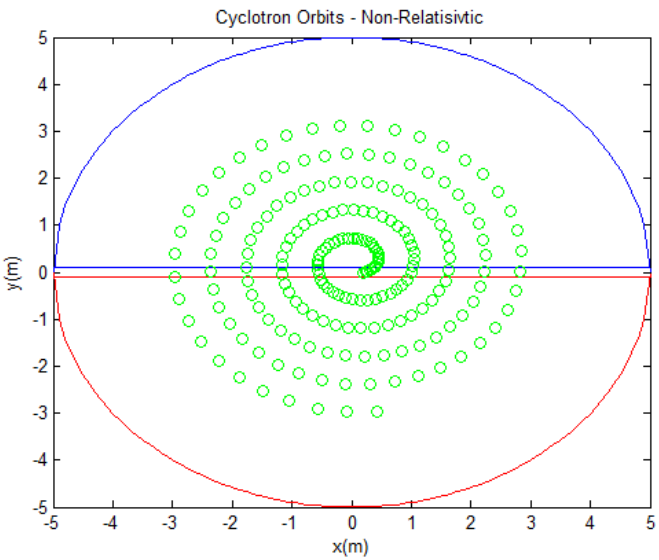


Figure 3.28: End of the movie for a charged particle in a cyclotron with 10 half revolutions and with an energy kick of 0.3 at each crossing of the "dees".

A movie is provided by the script "Cyclotron" of the proton path after the user chooses the number of half revolutions and the energy kick supplied by the electric field. The last frame of the movie for a specific choice, 10 half circles and a kick of 0.3 is given in Figure 3.28. The increasing radius and velocity with time are very clear. Obviously, higher energy particles require larger and more expensive devices. Indeed that fact historically limited the energies of beams accelerated by cyclotrons.

A few numbers are instructive. To achieve a velocity of 0.1 that of light for a proton, or a proton kinetic energy of 4.7 MeV requires the radius of the cyclotron to be 0.31 m. In general, the radius needed

is $r = 10.4 \times 10^{-9}(v_T/B)$ where r is in m, transverse velocity is in m/sec and B is in Tesla.

3.12. Dipole Radiation

The static dipole electric field has already appeared in Equation (3.2). In the dynamic case, accelerated charges create electric and magnetic fields that fall as the inverse of the radius rather than the inverse cube of the radius. Since the flux of energy through a surface then goes as the square of the radius, radiative solutions carrying constant flux over a surface of arbitrary size are possible. That realization was the great discovery of Maxwell, along with many other major contributions to many areas of physics.

The lowest order multipole is the dipole, called d here. This order is possible because electromagnetism has both positive and negative charges. By contrast gravity has a quadrupole moment as the lowest multipole because gravity is always attractive. The fields can be expanded in terms of the wave number, k, and the radius as shown in Equation (3.11).

$$\begin{aligned} E_r r^3 &= d(2z/r)(1 - ikr)e^{ikr} \\ E_\theta r^3 &= d(x/r)[1 - ikr - (kr)^2]e^{ikr} \end{aligned} \tag{3.11}$$

At small values of kr the solutions for the electric fields approach the static dipole case, where d is the dipole moment of the charge distribution which is here oriented along the z axis. However, at large values of kr, the static terms have fallen rapidly with r and the only remaining term has a transverse quality. The radial field has a near zone static piece and a piece that falls as the square of the radius. The transverse filed has a static piece going as the inverse cube of the radius, an intermediate piece and a radiative piece that falls only as the inverse of the radius,.

The fields, given in Equation (3.11) are evaluated in the script "EM_Dipole_Rad". They are plotted as surfaces as a function of kx and kz so that the static near zone, $kr \ll 1$ and the radiative far zone, $kr \gg 1$ can both be observed. The surface for the transverse radiative field is shown in Figure 3.29. The small kx and kz regions

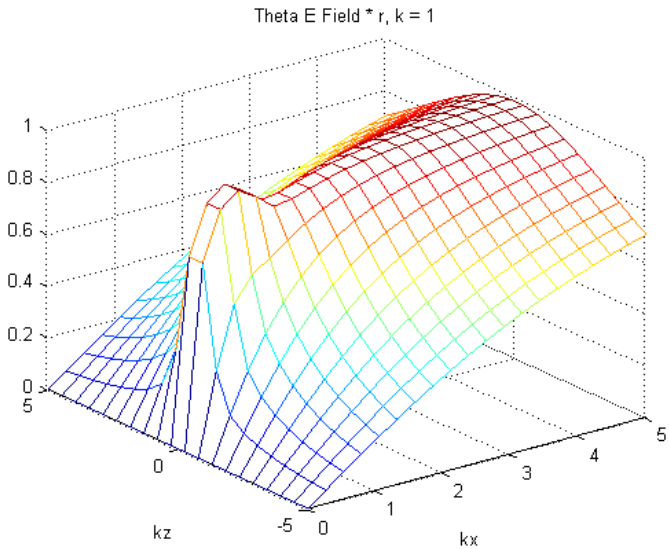


Figure 3.29: Contours for the transverse theta field, E_θ, as a function of kx and kz. The static region is at small kr, while the radiative behavior dominates at large kr.

should be compared to the static contours displayed in Figure 3.6. The surface for the radial field is shown in Figure 3.30. In this figure the region $r < 1$ is set to zero field for the purposes of a better display of the large r regions. The radial field falls off rapidly and is essentially the static field. Both fields are multiplied by r so that in the region of large r the radial field falls as $1/r$ while the transverse field approaches a constant in r.

In Figure 3.29 it is evident that the radiative behavior becomes strong when $kr > 1$ and that the radiative part of the wave is transverse to the wave vector k. The field approaches $E_\theta \to d \sin\theta(k^2/r)$. The magnitude of the field is proportional to the acceleration of the charge distribution, a, where $a \sim \omega^2$, so that the radiated power goes as the fourth power of the oscillation frequency. The energy flux through a sphere surrounding the source is independent of radius which indicates a radiative solution, $4\pi r^2 E_\theta^2 \sim 4\pi(d \sin\theta k^2)^2$.

These considerations can be applied to the situation where low energy photons are scattered by electrons in materials. The incident

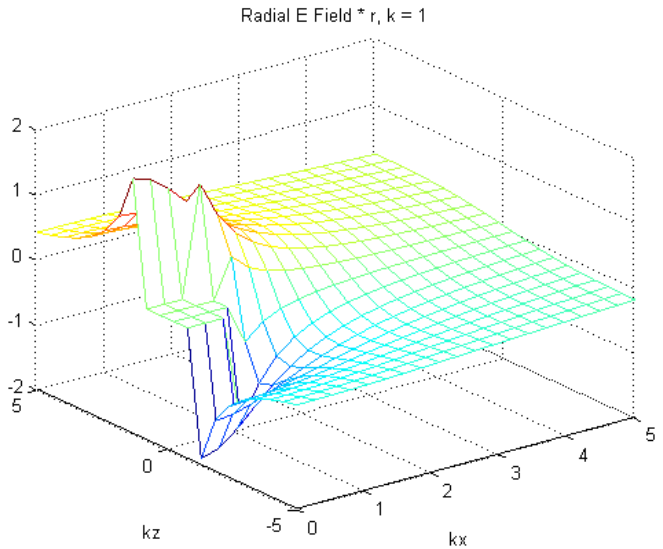


Figure 3.30: Radial electric field of an oscillating dipole. The region $r < 1$ has been set to zero.

light has a transverse electric field which accelerates the electrons and drives their motion with the frequency of the incoming light. The electrons radiate photons with the same frequency and these photons are, in turn, transverse to the acceleration. Therefore, the radiation is preferentially emitted in the direction of the incident light, both forward and backward. Indeed, the angular distribution, shown in Equation (3.12) is enhanced in the direction of the incident light which is here taken to be the z axis. The process is called Thompson scattering.

$$d\sigma/d\Omega \sim (1 + \cos^2\theta) \tag{3.12}$$

This behavior has important practical implications, since when radio waves bounce off obstacles, if they changed frequency, tuning by locking to a specific frequency would be impossible.

Chapter 4

Waves and Optics

"Classifications like 'optics' or 'thermodynamics' are just straitjackets, preventing physicists from seeing countless intersections."

— **Ted Chiang**

"Of Newton with his prism . . . a mind forever voyaging through strange seas of thought, alone."

— **William Wordsworth**

The electromagnetic fields have wave solutions as indicated at the end of the last section. However, there are many other cases of wave phenomena, so that a separate section is now devoted to general wave behavior. There are two basic regimes; one is that of geometric optics which obtains when the diffraction of light rays is small. That regime exists when the size of the objects being illuminated is much larger than the wavelength of the wave that scatters off the object. We daily operate in such a regime because visible light contains wavelengths of a few thousand angstroms which is much smaller than macroscopic, everyday objects. The other regime is that of wave optics where, for example, diffraction is important. This section starts with wave optics and then looks at a few examples from geometric optics where light is assumed to move in a straight line and diffraction can be neglected.

4.1. Adding Waves

A monochromatic wave is characterized by an amplitude, a frequency and a phase. Two waves can exhibit interference when combined. A script called "Osc_Add_Waves" looks at adding two waves. The initial printout defines the parameters of the waves in question, as seen in Figure 4.1. Since k is here used for wave number, it no longer

```
 Harmonic Oscillation Introduction
 T = 1/f,  w = 2*pi*f, Period, Frequency
 d2x/dt2 = kx/m, wo^2 = k/m
 x = Acos(wot)
 Examples: pendulum wo^2 = g/l, circuit = 1/L*C
 Wave Number = k = 2*pi/lambda = w/c, v = f*lambda
 Two Waves phase difference = path difference*2*pi/lambda
 Adding 2 Waves
Input the Ratio of the Amplitudes: 2
 Waves of Equal Amplitude, = 1, and Phase
Input the Fractional Difference of the 2 Frequencies = (w1-w2)/(w1+w2): 2
```

Figure 4.1: Initial printout for the case of adding two plane waves.

represents the spring constant, also called k. The circular frequency of the wave is ω_o.

A specific user defined input is shown in Figure 4.1. First, the intensity of the sum of the amplitudes for a specific amplitude ratio is shown as a function of the phase between the amplitudes, which illustrates constructive and destructive interference. The second part of the script shows the result of adding waves with the same amplitude but with different frequencies. There are 20 examples shown as a movie sequence and the "beat" frequencies build up as the difference in frequencies increases. The last plot, with a frequency ratio of twenty, is shown in Figure 4.2. The beat frequency gives the overall modulating behavior to the rapid oscillations with the sum of the frequencies. It is instructive for the user to see the way the beats build up with frequency difference. The phenomena of beat frequencies is familiar in music, for example.

4.2. Damped and Driven Oscillations

A first look was taken at damped and driven oscillation in the previous section on classical mechanics. The script "Osc_Damped" goes a bit deeper, partly because the phenomena occurs in many areas of physics and engineering. There are two series of plots which are each shown as a function of time in movie frames. In the first, the dialogue is shown in Figure 4.3, where the differential equation is shown and the solution in the under damped and over damped cases

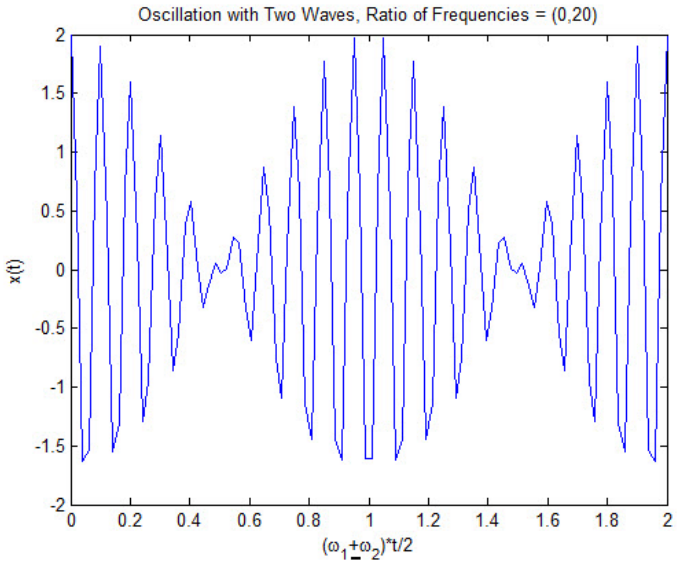


Figure 4.2: Adding two waves with a large difference in their frequencies. There are rapid variations at high frequencies and an envelope of slower modulations of the sum of the waves.

```
 Damped Harmonic Oscillator
 d2x/dt2 = kx/m -b*dxdt, wo^2 = k/m, frictional force - viscosity
 Undamped Harmonic Oscillator, x(o) = A, dx(0) = 0

x =

(A*exp(-(t*(-k*m)^(1/2))/m)*(exp((2*t*(-k*m)^(1/2))/m) + 1))/2

 Damped Harmonic Oscillator
 d = b/(2*m), D = d/wo, x(t) = A*exp(-d*t)*cos(wt) w^2 = wo^2 - d^2, underdamped
 x(t) = A*exp[-d +- sqrt(d^2 - wo^2)]*t, overdamped
 Example: R,L, C circuit, wo^2 = 1/(LC), d = R/(2L)
 Damped, Driven Oscillator - Resonance Behavior
```

Figure 4.3: Printout from the "Osc_Damped" script which shows the differential equation and the two possible solutions.

is defined. The MATLAB script "dsolve" is used to solve the problem symbolically.

The waveform for the lightly damped case of $D = 0.12$ is shown in Figure 4.4. The damping parameters D and d are defined in Figure 4.3. The wave goes through several oscillations without

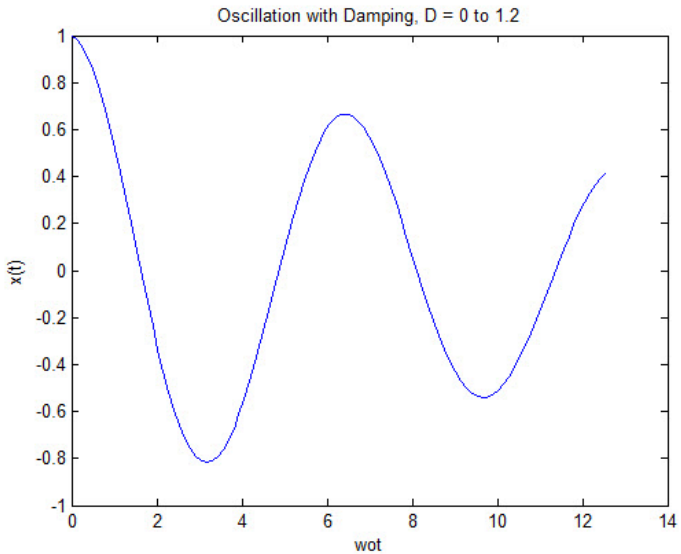


Figure 4.4: Waveform for the damped oscillation case of $D = 0.12$.

losing a significant fraction of its amplitude. For heavier damping the oscillation is washed out.

For a driven and damped oscillator, the exact solution was shown previously in Section 2. In the present case, only the resonant Breit-Wigner shape is plotted. It is the steady state solution after all the transients have died off. In general, for a system with a natural, undamped frequency of ω_o and a damping factor d, the full width of the resonant response at half maximum (FWHM) is approximately $2d$. The amplitude is shown for the steady state solution in Equation (4.1). The FWHM of the intensity I is $2d$. The wave amplitude is A, with $I = |A|^2$.

The resonant shape which obtains at long times is displayed in 20 plots where the response as a function of the external, driving frequency is shown for 20 different damping factors. The specific case of $D = 0.01$ is shown in Figure 4.5. In general, as the damping increases, the amplitude of the driven response drops and the width of the frequency response near the natural frequency increases. The user should find the variations in the response of the damped oscillator amusing. It can be imagined that a poorly damped system might

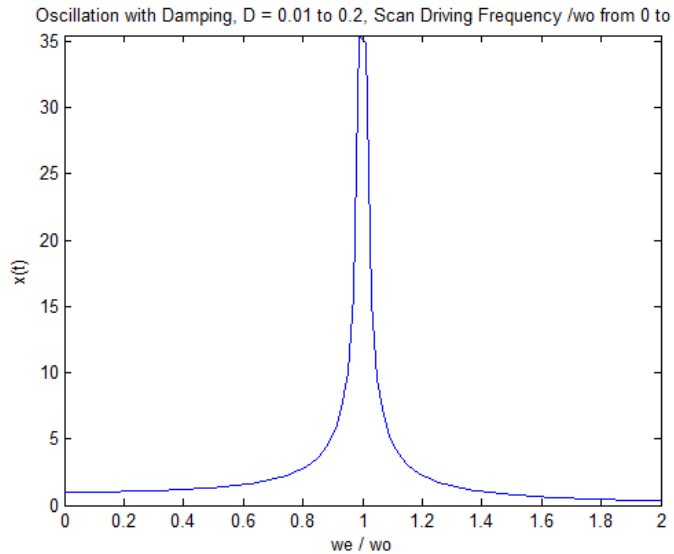


Figure 4.5: Resonant response of a driven oscillator in the damping case $D = 0.01$ as a function of the external driving frequency.

be heavily damaged by a driving force with a frequency near that of the natural frequency of the object. Indeed, the Verrazano bridge collapse is a classical example.

$$\begin{aligned} A &= 1/\sqrt{(\omega_e^2 - \omega_o^2) + (2d\omega_e)^2} \\ A &\sim 1/\sqrt{(\omega_o - \omega_e)^2 + d^2} \\ I &= |A|^2, \quad FWHM \sim 2d \end{aligned} \tag{4.1}$$

4.3. A Plucked String

A wave equation for a string with boundary conditions that it is held at two ends leads to the existence of only discrete frequencies which are allowed for the string motion. Thus, the problem can be easily treated by using the Fourier series. In fact, the square wave and triangular wave Fourier series solutions were already used in the section on symbolic math. In the script, "String_Pluck", the case of a string, initially triangular in shape, is displayed as a movie, where a Fourier series with forty terms is used to display the subsequent

motion of the string. The Fourier expansion is:

$$y = \sum_i b_i \sin(\omega_i t) \sin(i\pi x/L)$$
$$\omega_i = iv\pi/L, \quad v = \sqrt{T/\rho} \tag{4.2}$$

The string length is, L, and the wave velocity, v, is determined by the string tension, T, and string density, ρ. The discrete frequencies are labeled by the integer, i.

One of the string "snapshots" is shown in Figure 4.6. The boundary conditions are imposed by the choice of discrete frequencies. Many frequency components contribute to the motion and the wave begins to propagate to the fixed boundaries where the wave reflects off the end points.

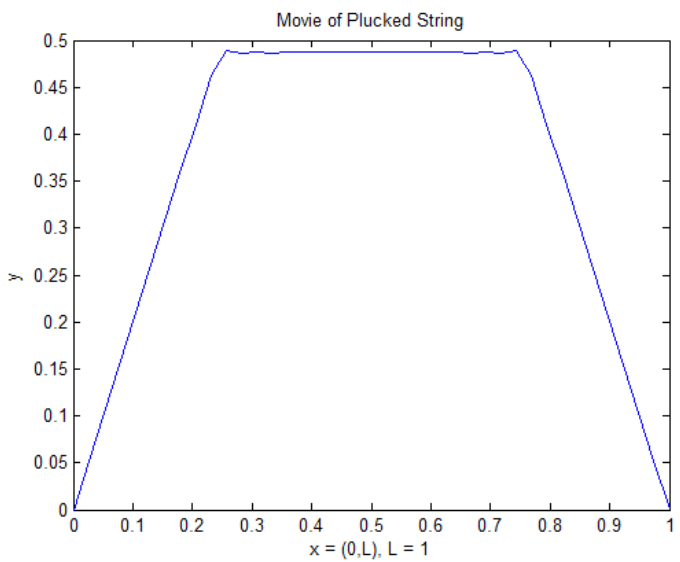


Figure 4.6: A snapshot of the evolution of the plucked string when the string is assumed to be initially triangular. In this case, the string moves both left and right, the waveform widens and the ends of the string are fixed.

The user is then asked to provide the width of a square wave and a movie of the subsequent motion is provided. If a velocity of

the string of one is chosen with a half width of the square wave of 0.2, a snapshot of the subsequent motion appears as an example in Figure 4.7. The initial shape propagates without dissipation (by assumption), is reflected off the fixed end points and evolves as shown in the "movie" which is provided. Clearly, this problem could be extended to two dimensions in Cartesian coordinates. The reader is encouraged to try that.

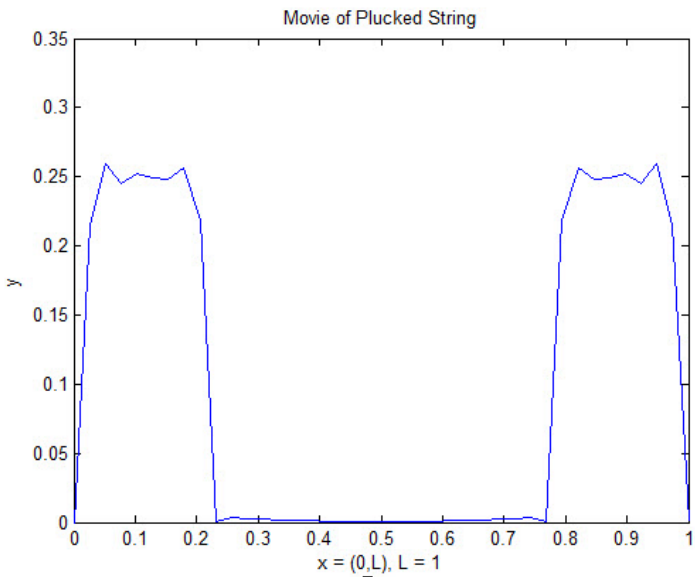


Figure 4.7: Square wave string amplitude evolution at a time "snapshot" prior to reflection off the fixed end points of the string. A full width of 0.2 was chosen initially.

4.4. A Circular Drum

The string in one dimension had solutions which were simple sin functions with wave lengths which were a discrete set because of the boundary conditions. This idea can be extended to solving the wave equation in two dimensions. To make it simple, cylindrical coordinates are employed. The solutions are Bessel functions in radius, sin and cos functions in azimuth and sin and cos functions in time. The

wave velocity is v.

$$\begin{aligned} \partial^2 u/\partial^2 t &= v^2(\partial^2 u/\partial^2 x + \partial^2 u/\partial^2 y) \\ u_{m,n}(r,\phi,t) &= \cos(v\lambda_{m,n}t)J_m(\lambda_{m,n}r)\cos(m\phi) \\ J_m(\lambda_{m,n}a) &= 0 \end{aligned} \tag{4.3}$$

For simplicity the t and azimuthal functions are specialized to cos and the radius is scaled to a equal to one as is the wave velocity v. The boundary conditions are radial, where the Bessel functions vanish at $r = a$. The MATLAB script, "besselj", is used to evaluate the solutions. Any motion of the drum can be expanded in terms of these solutions, as was the case in one dimension with the plucked string as an example. In this case, the solutions given in Equation (4.3) for $m = 0, 1, 2$ and $n = 1, 2, 3$ are available and a movie of the drum head motion is provided to the user, who picks an m and n value. The user dialogue is shown in Figure 4.8.

```
>> Drum_Modes
  Circular Drum - 2d Wave Eq


Circular Drum of Radius r = 1
 Radial Jm(Lmnr), Aximuthal cos(m*phi)
  Temporal, cos(Lnmt), speed = 1, J of type m, with root n
 Enter the Bessel Function Jm, m = 0,1,2: 1
Enter the Order of the Bessel Root n = 1,2,3: 2
```

Figure 4.8: Dialogue for the script "Drum_Modes" with a scpecific choice of Bessel function order and root index.

A frame of the movie for the case $m = 1$ and $n = 2$ is shown in Figure 4.9. This solution is not uniform in azimuth and shows that the choice of this radial Bessel function is one with a zero at $r = 0$. The Bessel function of order zero is non-zero at the origin.

4.5. Diffraction by Slits and Apertures

Diffraction limits the ability to distinguish objects using a wave with a wavelength comparable to the size of that object. This is

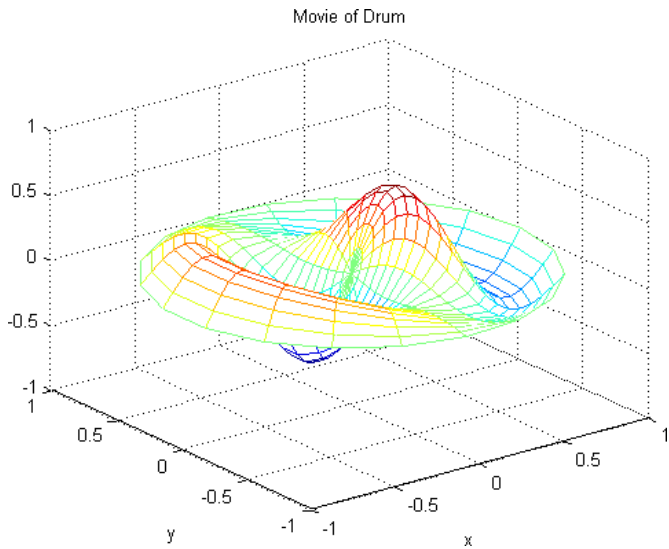


Figure 4.9: Surface of a drum head at one frame of a movie for the mode defined by the dialogue of Figure 4.8.

a fundamental limit, called the diffraction limit. It is the reason why optical microscopes evolved into electron microscopes. It is also the fundamental reason why the energy of particle accelerators continues to increase. In order to observe smaller objects, the wavelength of the "light" used must decrease which means the energy or frequency must increase.

The simplest problem is the single one-dimensional slit. That problem is illustrated in the script "Diffract" which looks at a one-dimensional slit, a circular aperture in two dimensions and a double slit. The intensity of the light as a function of angle after striking a slit in one dimension with light with a wavelength twice the size of the slit width, d, is shown in Figure 4.10. Also shown is the resulting intensity for a circular aperture with a wavelength equal to twice the diameter of the aperture. Note that the MATLAB function "besselj" is used to evaluate the case of the circular aperture. The characteristic diffraction pattern is clear. It is contained in an envelope whose width in angle decreases as the inverse wavelength, or wave number,

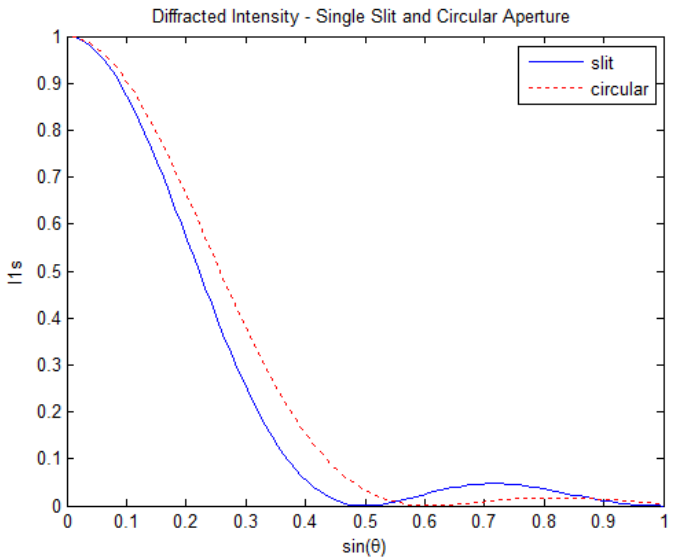


Figure 4.10: Diffraction pattern from a single slit and from a circular aperture for light with wavelength twice the slit width or twice the diameter of the aperture.

$k = 2\pi/\lambda$, increases. In order to have a fixed value of the intensity as the wave number increases, the wavelength decreases, and the slit size d must decrease to keep the intensity I constant.

$$\begin{aligned} \alpha &= \pi \sin\theta d/\lambda = kd \sin\theta/2 \\ I &= (\sin\alpha/\alpha)^2 \end{aligned} \tag{4.4}$$

For the case of two slits, there are interference effects between the waves emitted by the two slits. For the case where the single slit is as defined for Figure 4.10, but with the two slits separated by twenty times the wavelength, the resulting pattern is given in Figure 4.11. The single slit diffraction pattern is modulated by the interference from the second slit, which happens on an angular scale about twenty times smaller than that of the single slit. Finally, the script shows a slide show for a single circular aperture diffraction pattern for a wide span of wavelengths showing how the pattern shrinks in angle as the wave number k increases, or the wavelength decreases.

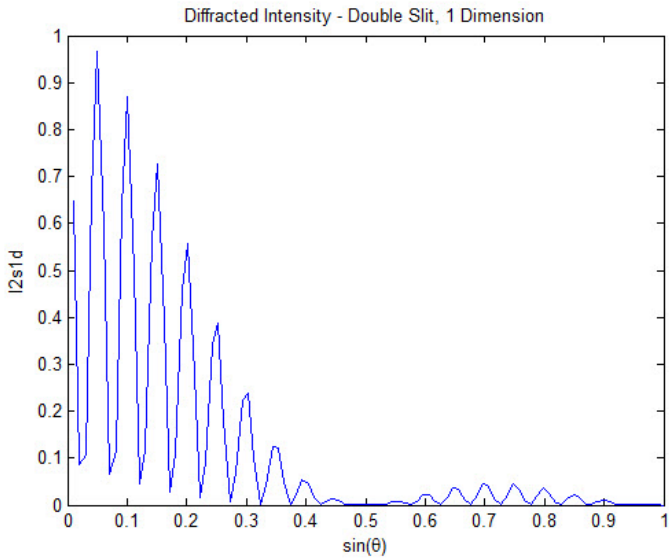


Figure 4.11: Diffraction pattern from two slits for light with a wavelength twice the slit width and twenty times the separation of the two slits. Compared to a single slit, Figure 4.10, there is a rapid angular modulation in the two slit case.

4.6. Edge Diffraction

The previous examples of diffraction were given in the regime where the observation point is far from the diffracting system in terms of the characteristic size of the system. That is the regime of Fraunhofer diffraction. There is another regime where the observation point is at a location with dimensions comparable to the diffracting system and the wavelength. That is called the regime of Fresnel diffraction.

The problem of Fresnel diffraction for an edge barrier or a one dimensional slit or a two dimensional square aperture is covered in the script "Edge_diffract". In order to find the relevant Fresnel integrals, the MATLAB tool "mfun" is used to access the Fresnel "C" and "S" functions. This tool was previously mentioned in the section on symbolic mathematics. These functions depend on the variable:

$$\omega = y\sqrt{2/\lambda z} \tag{4.5}$$

The result for a user menu choice of an observation point, z, behind a barrier, $y < 0$, of 5 times the wavelength, λ is shown in Figure 4.12. The edge diffracts the wave over a region of several wavelengths, making the shadow of the barrier, $y < 0$, not a simple step function.

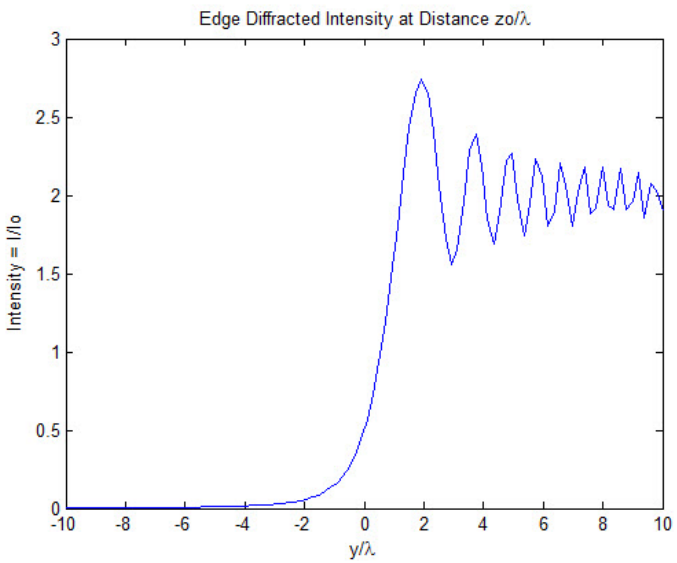


Figure 4.12: Diffraction pattern for a screen in the lower half plane $y < 0$ with the light observed at a z location behind the screen with z equal to five times the wavelength of the light.

The user can choose the observation point and whether to look at a simple edge screen or a slit of width h centered at $y = 0$ or a square aperture centered at $x = y = 0$. The result for a slit of width 10 and an observation point $z = 5$, both in wavelength units is shown in Figure 4.13. The diffraction near the slit boundaries $y = +5$ and $y = -5$ is quite evident. There are also maxima in the pattern at locations with $|y|$ less than 5. A diffraction pattern for a square aperture is shown in Figure 4.14. Diffraction is strong near the aperture edges. The user, after having finished his/her choices, is shown a slide show where the width of the slit is varied from 1 to 20 in wavelength units,

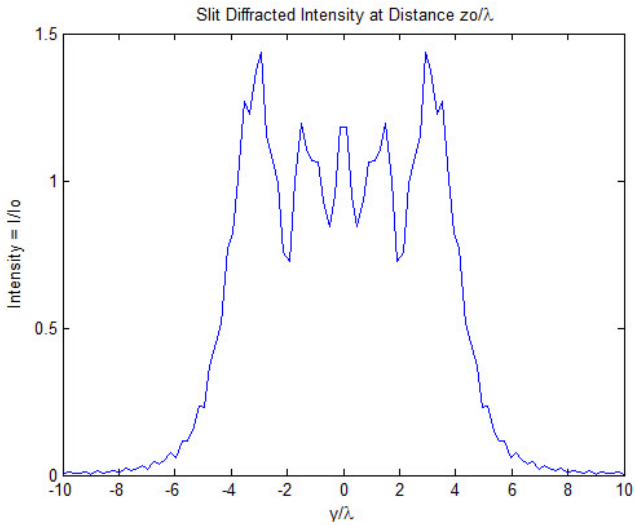


Figure 4.13: Diffraction pattern for light incident on a slit of full width 10 observed at a distance $z = 5$ behind the slit where the width and location are given in wavelength units.

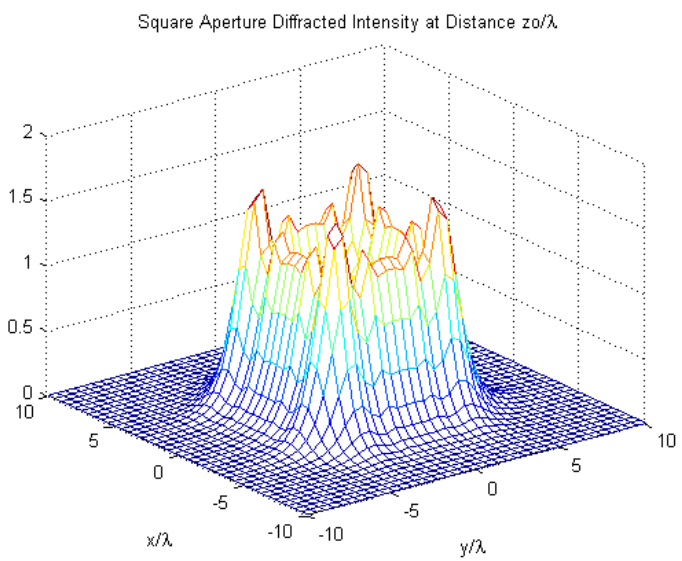


Figure 4.14: Diffraction by a square aperture of full width 10 observed at a distance $z = 5$ behind the aperture with width and location in wavelength units.

showing how the pattern begins with a single maximum at small slit widths, and evolves to a pattern with many local maxima.

4.7. Doppler Shift and Cerenkov Radiation

The phenomena of a Doppler shift of the frequency of a wave depending on the motion of the wave source relative to the observer and the closely related Cerenkov effect is illustrated in the script "Doppler_Cerenkov". There are six discrete emission times, with the emission location depending on the source velocity, which is chosen by the user. The observer is assumed to be at rest. There is a slide show, where the resulting outgoing circular waveforms are sampled and displayed. At the last plot the developing waveform is becoming clear.

The waveforms for a velocity, v, of one half with respect to the wave propagation velocity, v_s , are shown in Figure 4.15. The angle is that of the observer with respect to the z axis which is the direction of uniform motion of the source.

$$\omega/\omega_o = 1 - v\cos\theta/v_s \tag{4.6}$$

In the forward region the wave is blue shifted, while in the backward region, the source is receding and the wave is red shifted. There are many applications of the Doppler effect, such as Doppler radar for traffic control and for weather mapping. The Hubble red shift is due to the recession velocity of galaxies with respect to our own. The red shifted and blue shifted regions are indicated by color.

The case of $v/v_s = 1.5$ is shown in Figure 4.16. In this case, the source outruns the wave and a coherent cone of outgoing waves builds up, called a Mach cone. This is the shock wave that makes a sonic boom when a jet plane exceeds the speed of sound in air. Parenthetically, that speed is the square root of the pressure, divided by the density and is about 0.35 km/sec at STP. It is also a tool in high energy physics where if light is emitted by a particle, the Cerenkov effect, in a medium then it is known that the velocity of that particle is greater than that of light in the medium which has an index of refraction of n, or c/n.

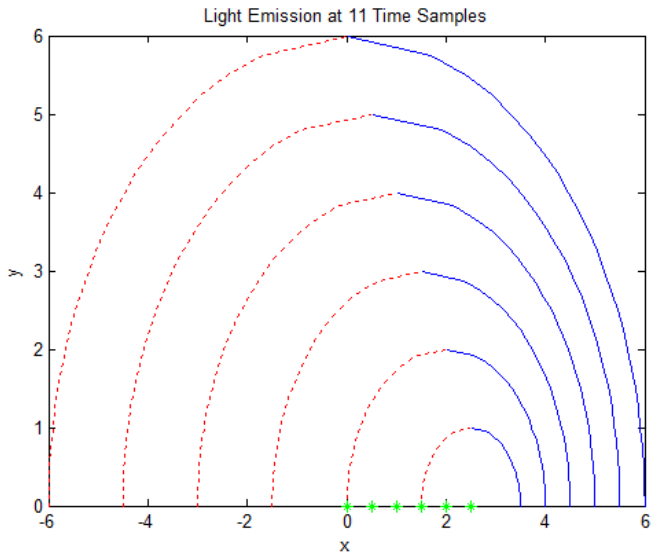


Figure 4.15: Outgoing waves in the case where $v/v_s = 0.5$. The regions of wavelength compression and expansion are seen in the forward and backward positions. The emission points are green *.

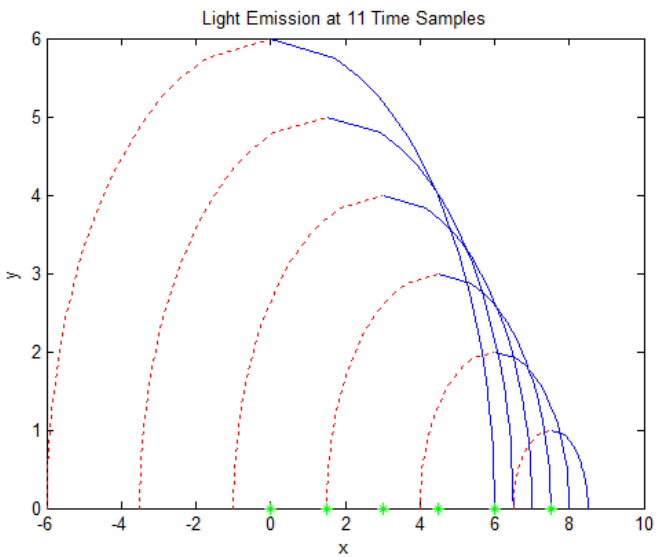


Figure 4.16: Outgoing waves in the case where $v/v_s = 1.5$. The regions of wavelength compression and expansion are seen in the forward and backward positions. The emission points are green *.

4.8. Reflection and Transmission at an Interface

There are, in general, both transmitted and reflected waves when a wave strikes an interface between media with different indices of refraction. It is now assumed that light goes in straight lines and is not diffracted, which is the regime of geometrical optics. This is reasonable since for visible light an object of size 1 cm has a wavelength to size ratio of approximately 0.0001.

The basic relationship between the transmitted, θ_t and incident, θ_i angles of waves with respect to the normal to the surface is called Snell's law.

$$\sin\theta_i / \sin\theta_t = n_t / n_i \tag{4.7}$$

There are two polarization states of the light which respond slightly differently; polarization transverse to the plane of incidence and normal to it. This problem is covered in the script "Reflect_Transmit". The printout from a session where the user has chosen an index ratio of 1.5 and 0.5 is shown in Figures 4.17 and 4.18. In the first case, there is a large angle with no reflection with parallel polarization. Indeed, that is why anti-glare sunglasses are polarized and how the polarization blocking is oriented. There is also a phase change upon reflection, which means a thin coating can cancel out a reflected wave at a given wavelength. That is the principle behind

```
 Reflection and Transmission at an Interface
 Ratio of Indices = ni/nt = 1/n
Input Ratio of n Refracted to n Incident: 1.5
 Snells Law: sini/sinr = nr/ni = n
 Perpendicular Incidence, R = ((n-1)/(n+1))^2, T = (2/(n+1))^2
No Reflection for Parallel Polarization at = 56.3099 (deg)
Input Ratio of n Refracted to n Incident: 0.5
 Snells Law: sini/sinr = nr/ni = n
 Perpendicular Incidence, R = ((n-1)/(n+1))^2, T = (2/(n+1))^2
No Reflection for Parallel Polarization at = 26.5651 (deg)
 Total Internal Reflection at = 30 (deg)
```

Figure 4.17: Printout for the "Reflect_Transmit" script for 2 choices of index ratio, 1.5 and 0.5.

anti-reflection coatings on lenses. For a wave incident from $n = 1$ on a thin film of thickness, t, preceding a medium of index, n, there is no reflection at a given incident wavelength, if the thickness is chosen to be $t = \lambda/4\sqrt{n}$. This topic will be developed later in the quantum mechanical context.

In the user dialogue, the index ratio is chosen. Then a plot of the transmitted angle as a function of incident angle is supplied, along with the transmission and reflection coefficients as a function of incident angle for both transverse and parallel polarization states. An example appears in Figure 4.18.

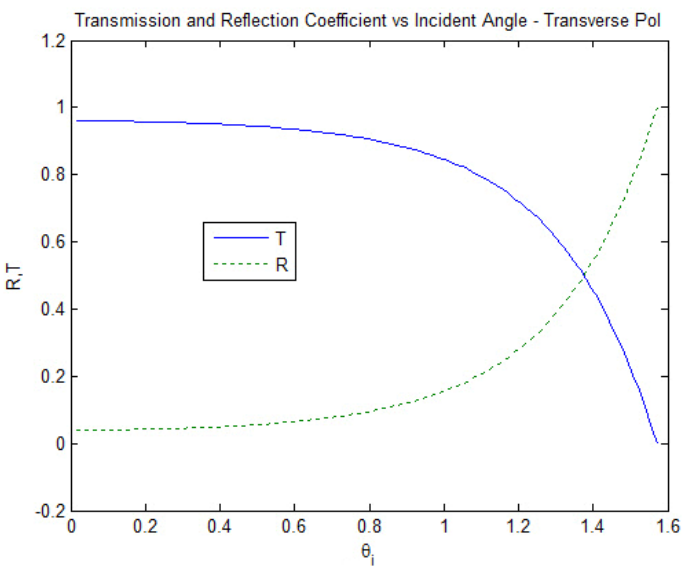


Figure 4.18: Transmission and reflection coefficients as a function of incident angle in the case where the ratio of reflected to incident index of refraction is 1.5.

In case the index ratio is less than one, there will be total internal reflection, as displayed in Figure 4.19 in the case of parallel polarization. In that specific case, the angle at which total internal reflection occurs is thirty degrees. It is this principle which is the basis for transmitting data over fiber optic cables, those cables which are now ubiquitous in technical fields.

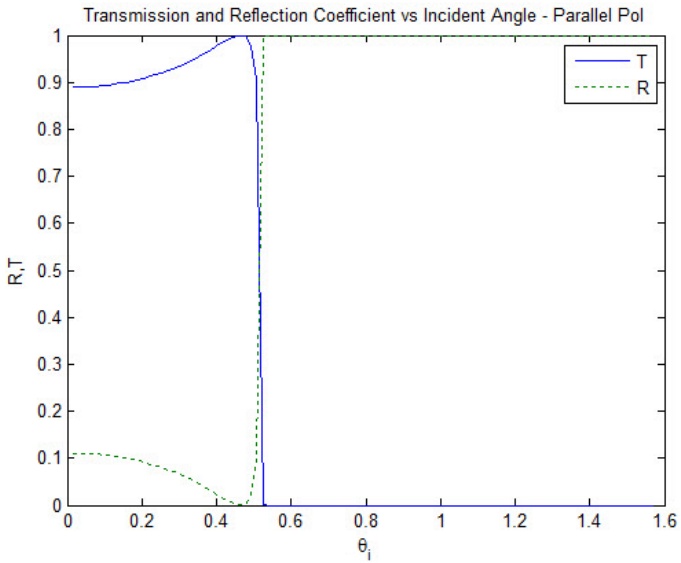


Figure 4.19: Transmission and reflection coefficients as a function of the incident angle in the case where the ratio of reflected to incident index of refraction is 0.5. There is also an angle without reflection at an angle less than the internal reflection angle.

As an aid to memory, a movie is made where five rays are incident from outside and five from inside. The resulting last frame appears for $n = 1.5$ in Figure 4.20.

4.9. A Spherical Mirror

In geometric optics, light goes in straight lines. In fact, ray tracing is a tool used to follow the behavior of a beam of light. An example is shown in the script "Spherical_Mirror". The ray tracing is simple; the incident angle is equal to the reflected angle. The question is how good a focal point exists in the case of the mirror.

The user menu allows for a choice of what fraction of the mirror is filled with incident light. Ray traces in two cases are examined in this example. First rays using the full aperture of a spherical mirror are shown in Figure 4.21. It is clear that the rays' incident at large distances off the axis of the mirror suffer from severe aberration.

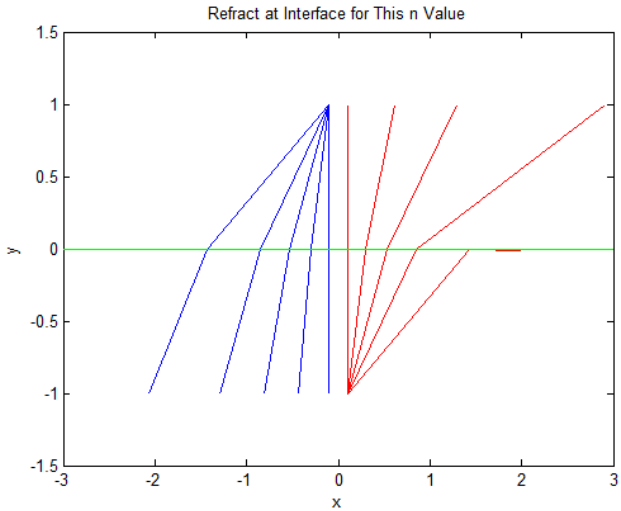


Figure 4.20: Light rays incident from top onto $n = 1.5$ (blue) and incident from the bottom (red). Since $n > 1$, the blue rays are bent toward the normal, the red away, and total internal reflection occurs.

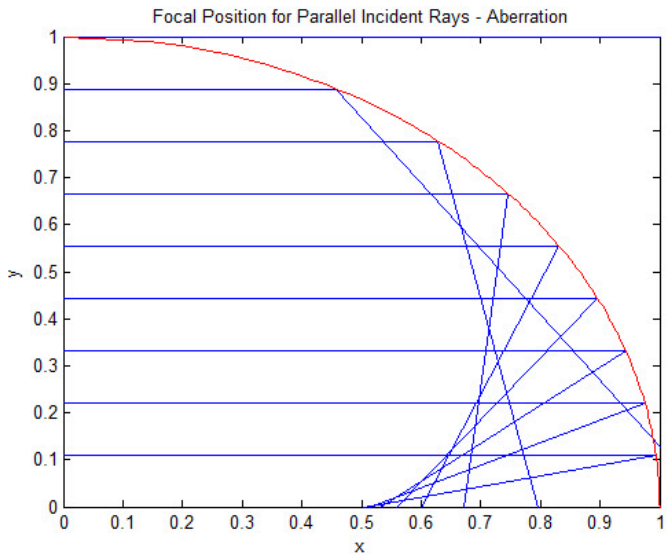


Figure 4.21: Ray tracing for a spherical mirror where the rays are incident on a mirror of radius of curvature one out to off axis rays at 90% of the radius.

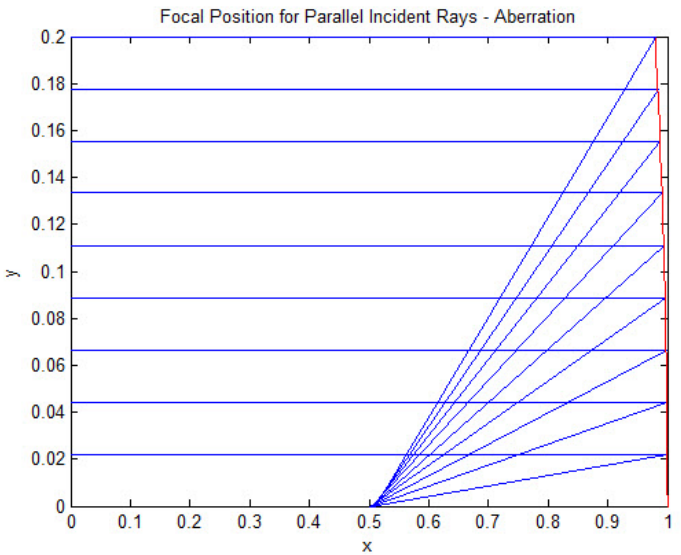


Figure 4.22: Ray tracing for a spherical mirror where the rays are incident on a mirror of radius of curvature one out to off axis rays at 20% of the radius. The focal point is much better defined in this case.

In Figure 4.22, the incident parallel beam only has rays up to 20% of the mirror radius off the mirror axis. Clearly, the off axis rays must be limited if good focal properties are to be maintained. The user can choose the limitation on the incident beam. The focal distance is one half the radius of curvature of the mirror, $f = R/2$.

4.10. A Spherical Lens

A related problem is the focal properties of a spherical lens. The problem is treated in the script, "Spherical_Lens2". The index of refraction of the lens is fixed at $n = 1.5$. The radius of the lens is $R = 10$. Ray tracing is accomplished by the use of Snell's law. The user chooses the angular size of the lens, limited to be less than about 60 degrees. A first example is seen in Figure 4.23, where the chosen angle was 50 degrees. Obviously, this choice has a focal point which is not very well localized. Larger impact rays have a reduced focal length. The user can easily restrict the region of incident rays to see

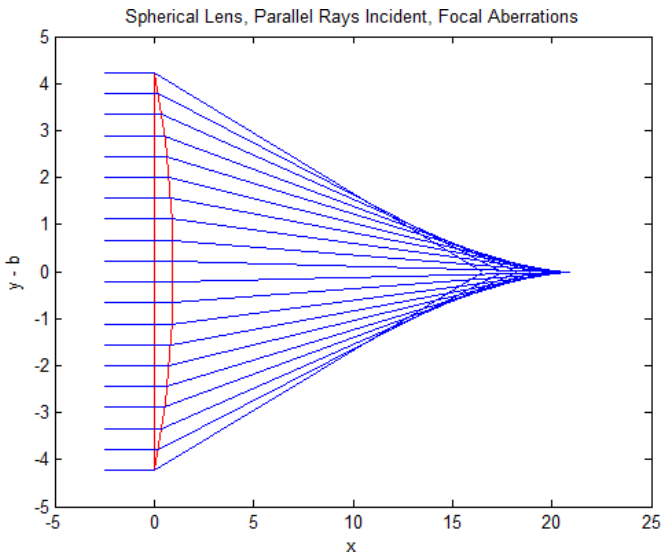


Figure 4.23: Ray tracing for a spherical lens with index of refraction = 1.5, where the incident beam fills the lens up to 50 degrees with respect to the lens axis.

how the focal length is better defined as the extent of the incident light is limited.

The situation when the beam fills only up to 20 degrees appears in Figure 4.24. It is evident that the focal point is much better localized in this case. The active fraction of a lens is limited to the area near the lens axis, as was the case for the spherical mirror. The lens maker's equation for an incident parallel beam of light is, $1/f = (n-1)/R$, where f is the focal length, n is the index of refraction of the lens, and R is the radius of curvature of the lens. In this example, $n = 1.5$ so that $f = 2R$, as is observed in Figure 4.24.

4.11. A Magnetic Quadrupole Lens System

This section finishes with a rather more complex problem, something more akin to the type of question a working physicist might encounter, albeit still simplified here. The exercises refer to a system of magnetic lenses. These lenses are manufactured to create a quadrupole magnetic potential, Φ, where the magnetic field is the

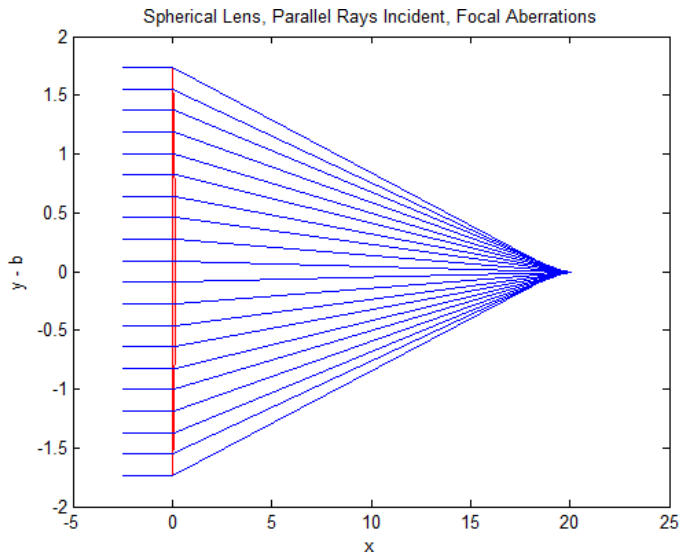


Figure 4.24: Ray tracing for a spherical lens with index = 1.5 where the incident beam fills the lens up to 20 degrees with respect to the lens axis.

gradient of the potential.

$$\begin{aligned} \Phi &= (dB/dr)xy \\ B_x &= -(dB/dr)y \\ B_y &= -(dB/dr)x \end{aligned} \tag{4.8}$$

The magnetic field gradient, dB/dr, causes a charged particle moving almost entirely along the z axis and having an x displacement to be focused by encountering a force toward the x axis, while in the y direction, a y displacement is de-focused away from the y axis. Therefore, the simplest electromagnetic system which provides focusing in both transverse dimensions is a doublet of these quadrupoles. This system provides net focusing of a beam of charged particles.

Solving the equations of motion in a quadrupole magnet for a particle with momentum, P, can best be cast into a matrix form, assuming that the motion is largely along the average beam direction, or z,

axis and that displacements transverse to z are small. In the focusing case:

$$k = a(dB/dr)/P$$

$$\phi = \sqrt{k}L$$

$$\begin{bmatrix} x \\ dx/dz \end{bmatrix} = \begin{bmatrix} \cos\phi & \sin\phi/\sqrt{k} \\ -\sqrt{k}\sin\phi & \cos\phi \end{bmatrix} \begin{bmatrix} x_o \\ (dx/dz)_o \end{bmatrix} \tag{4.9}$$

the transverse motion is characterized by an initial position, x_o, and angle, $(dx/dz)_o$, and the matrix transforms the incoming beam into the outgoing beam at the exit of the quadrupole of length, L, for one transverse dimension. The amount of focusing depends on the field gradient, dB/dr, the inverse of the momentum, P, and a constant, a. The other dimension, which must be de-focusing, has a similar matrix with the trigonometric functions replaced by hyperbolic functions.

In analogy to the optical case, there is the behavior of a thick lens (Figure 4.23) and a thin lens (Figure 4.24) limit where, in the latter case, the position does not change, but the angular change in dx/dz, is equal to kLx or x/f, where f is the thin lens focal length, $1/kL$.

The script for studying the doublet of two quadrupole lenses is spread over several files. The major file is the script "Quad_Doublet". The script "Quadrupole" evaluates the matrix elements, the script "Doublet_Fit" uses the MATLAB function "fminsearch" to find the focal lengths which satisfy certain beam conditions, and "Doublet Plot" makes a plot of the solution to the fit. Because the fit is non-linear, starting estimates for the solution are needed, and they are provided by the script "Thin_Lense" which has explicit solutions for the case of a thin lens. The user can find the thin lens solutions there and verify them if interested. The printout for the user dialogue is shown in Figure 4.25.

The geometry of the doublet is defined and is completed by the user, in this case a 10 m distance from the initial beam to the entrance of the first quadrupole. A "drift" is a space with no magnets, and particles are un-deflected in a "drift space". All three solutions are displayed, point target to parallel captured beam, parallel beam to

```
>> Quad_Doublet
  Program to make quadrupole doublet ray trace, thick lense - MATLAB
  fminsearch for solutions such as point to parallel


Incident Beam at z=0, Variable Drift, 5m QF, 5 m Drift, 5m QD, and 7.5 m Outgoing Beam
 Enter Distance from z = 0 to First Quad Entrance: 10
 Enter Constraint,=1,2,3 = point to parallel, parallel to point, point to point: 1
Thin Lense - Starting Values (m) = 8.33333
Thin Lense - Starting Values (m) = 15
Thick Lense - fminsearch solution (m) = 7.20502
Thick Lense - fminsearch solution (m) = 13.3069
 Enter Distance from z = 0 to First Quad Entrance: 10
 Enter Constraint,=1,2,3 = point to parallel, parallel to point, point to point: 2
Thin Lense - Starting Values (m) = 14.1421
Thin Lense - Starting Values (m) = 7.07107
Thick Lense - fminsearch solution (m) = 12.4818
Thick Lense - fminsearch solution (m) = 6.02112
 Enter Distance from z = 0 to First Quad Entrance: 10
 Enter Constraint,=1,2,3 = point to parallel, parallel to point, point to point: 3
Thin Lense - Starting Values (m) = 6.5372
Thin Lense - Starting Values (m) = 5.88348
Thick Lense - fminsearch solution (m) = 5.47576
Thick Lense - fminsearch solution (m) = 4.87339
```

Figure 4.25: Printout for the "Quad_Doublet" script. The focal lengths in thin lens approximations and after the non-linear fit are printed for the three defined solutions.

point focus, and point target to re-focused point beam. The thin lens starting values and the fit values are also given.

Two of the ray traces are shown in Figures 4.26 and 4.27. The trajectory in the quadrupoles is exact, as long as the motion is largely in the z, or beam, direction. The rays in Figure 4.26 correspond to the case of a point like target where particles are created and captured into a beam. The first quadrupole is y focusing and x defocusing, leading to rather different beam sizes in the two transverse dimensions. The second case fully refocuses the beam. However, the aperture of the quadrupole is again filled very asymmetrically, with a large x value in the second one.

A real target has a finite size and a real quadrupole has magnetic field imperfections. The solutions shown here are for a monochromatic beam and any real beam contains a spread in momentum. In general, there is a central momentum, usually defined by bending the beam in a dipole magnet and then collimating which serves to select particles of like charge and momentum. Nevertheless, the exer-

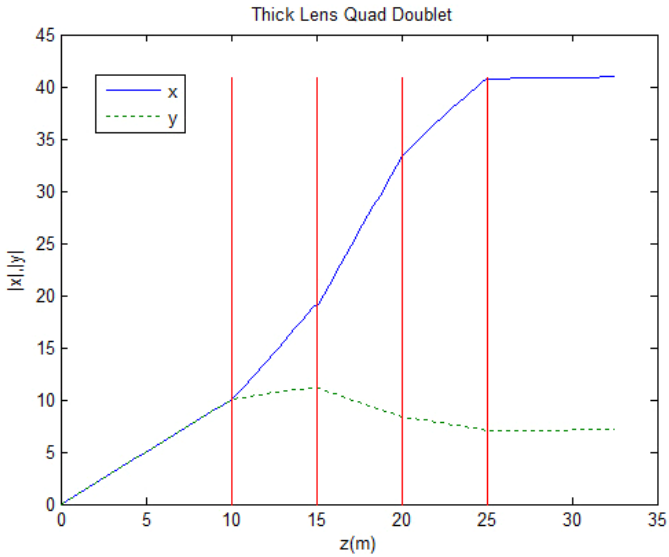


Figure 4.26: Solution in the case where a point target is captured into a parallel ($dx/dz = dy/dz = 0$) beam. The boundaries of the quadrupoles are shown in red.

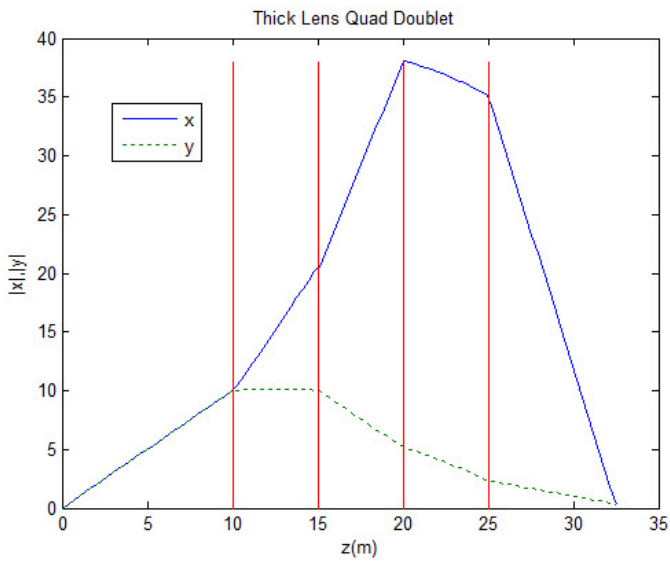


Figure 4.27: Solution in the case where a point target is captured and re-focused into a point like beam.

cise shown here in beam design and ray tracing is one that has use in the practical work of an experimental physicist. This demonstration is one which shows how complex problems can be approached by linking together module scripts which make specific and limited computations.

Chapter 5

Gases and Fluid Flow

"Time flows away like the water in the river."

— **Confucius**

"It is life, I think, to watch the water. A man can learn so many things."

— **Nicholas Sparks**

5.1. The Atmosphere

The atmosphere of the Earth is a mixture of gases. For example, there is essentially no helium in the atmosphere, although hydrogen and helium make up the vast bulk of the masses of the stars. This fact is related to the distribution of the velocities of molecules with different molecular weights. Heavier molecules move more slowly than light ones because all species have the same mean thermal kinetic energy. Basically, for a body to have an atmosphere, it must be cold enough and massive enough that its gravity well defines an escape velocity which is large with respect to the thermal velocity. Typical thermal velocities at STP are a few km/sec.

The script "Atmosphere" begins to explore this topic. The printout made by this script appears in Figure 5.1. Several facts about the ideal gas law and Boltzmann distributions are given in the user dialogue. The parameter, E, is now again the particle kinetic energy by established convention and not to be confused with the electric field or the total energy.

The Maxwell–Boltzmann distribution of energy at different temperatures is shown in Figure 5.2. The distribution and the related distribution of velocities are given in Equation (5.1). The distribution is the probability of observing a kinetic energy, E, for a system of many objects of mass, m, in thermal equilibrium at temperature, T. The constant k in this section is the Boltzmann constant, $k \sim 1/40\,\text{eV}$ at

```
>> Atmosphere
  Ideal Gas Law and Earth's atmosphere


Ideal Gas Law - PV = NkT
Density of Air at STP ~ 1.3 kg/m^3
Pressure: 1 bar = 10^5 Pa, 1 atm = 760 Torr = 101 kPA
Pressure Falloff with Altitude, P/Po ~ exp(-rho*g/P)y, ~ 0.12/km
<E> = 3/2*kT, PV = 2/3<E>
(<v^2>) = 2*<E>/m = 3*kT/m
kT~1/40 eV at STP, beta for a gas of p is 9x10^-5 or v ~ 2.7 km/sec
Speed  of Sound in O2 at STP is ~ 0.32 km/sec
Escape Velocity - Earth (km/sec) = 11.1807
No He in the Atmosphere?, Thermal He Velocity (km/sec) = 1.09515
O2 Thermal Velocity (km/sec) = 0.387195
```

Figure 5.1: Printout which gives some numerical values for the atmosphere of the Earth.

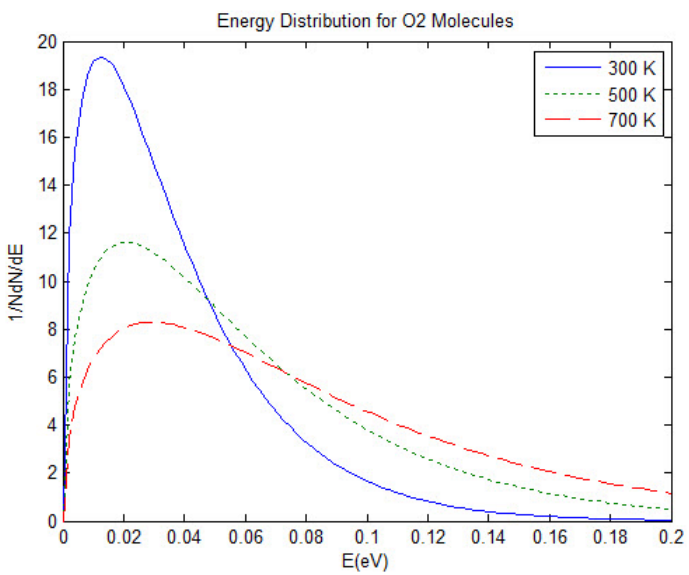


Figure 5.2: Maxwell–Boltzmann energy distribution for oxygen molecules at different temperatures.

300 degrees Kelvin. Normalizing to the total number of objects, N, the probability is $1/N(dN/dE)$.

$$\begin{aligned} dN/dE &\sim \sqrt{E}e^{-E/kT} \\ dN/dv &\sim v^2 e^{-mv^2/2kT} \end{aligned} \tag{5.1}$$

It is easy to convert variables by finding the value of dE/dv, for example, in order to switch from the distribution of E to that for velocity.

The mean velocities at room temperature, 300 K are quoted in Figure 5.1. The mean energy increases proportional to the temperature, so that the mean velocity goes as the square root. The velocity distribution for helium and oxygen is shown in Figure 5.3. The plot shows that the helium is much more likely (note the logarithmic vertical scale) to attain escape velocity than the oxygen, which implies

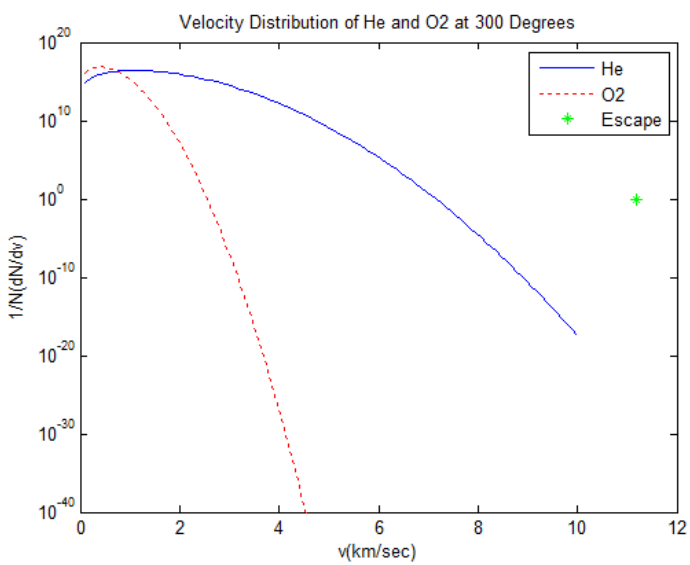


Figure 5.3: Velocity distribution for He (blue) and oxygen (red). The escape velocity for the Earth is the * in green. Escape seems unlikely, but there are many possible thermal fluctuations in the course of eons.

that the Earth at present has an atmosphere composed of heavier diatomic molecules.

5.2. An Ideal Gas Model in Two Dimensions

It is instructive to make a simple model of a two-dimensional non-interacting gas. The MATLAB function "rand" is used to create distributions for the random variable so produced. This method of creating models is called the Monte Carlo method. For example, in the script "Maxwell_Boltz", the distribution of energy in two dimensions is a simple exponential and that behavior can be properly weighted by taking the log of a random number. The user should try this by making, say, 1000 trials and histogramming the results using the MATLAB script "hist". The script "Maxwell_Boltz" adopts a more general case where the energy is defined between maximum and minimum limits. The user may wish to look at that specific code. Similarly, the initial positions of the "gas molecules" in the two-dimensional box are randomly chosen in x and y.

The printout of the script for a specific set of user choices is given in Figure 5.4. The user picks the temperature of the gas and the area of the box which holds the gas. In the example shown, an area of one and two temperatures are picked. The number of collisions of the molecules with the walls is tracked as is the total momentum impulse given to the walls by the reflection of the molecules. In this way, the ideal gas law can be modeled, albeit with only about 100 molecules and not 10^{23}.

With a two-fold increase in temperature, the mean velocity is expected to increase by the square root of two, while an increase of a factor 1.38 is seen. The temperature increase leads to more wall collisions per unit time and a larger momentum transfer impulse per collision, with a "pressure" increase of a factor 1.68. Note that the statistical accuracy here is limited by the small number of "molecules". The user can try several times with a fixed number of "molecules" to see the statistical variations or, although it is slow, increase the number of "molecules" or the number of time steps. A snapshot of the gas area with parameters defined in Figure 5.4 is shown in Figure 5.5.

```
>> Maxwell_Boltz
  Ideal Gas Laws and the Maxwell - Boltzmann distribution


Input the Number of Gas Molecules : 100
Input the Number of Time Steps : 50
Input the Gas Temperature * k, Mass = 1: 1
Mean Velocity 1.25979
Input the Box Height, Length = 1: 1

 Number of Wall Collisions 380

 Momentum Impulse to Walls 929.829
Input the Number of Gas Molecules : 100
Input the Number of Time Steps : 50
Input the Gas Temperature * k, Mass = 1: 2
Mean Velocity 1.69297
Input the Box Height, Length = 1: 1

 Number of Wall Collisions 519

 Momentum Impulse to Walls 1563.43
```

Figure 5.4: Printout for the script "Maxwell_Boltz" for specific user choices, in this case area equal to one and two temperatures, one and two.

5.3. Maxwell–Boltzmann Distributions

A classical gas of non-interacting particles follows the Maxwell–Boltzmann distribution. The distribution follows from the statement that all velocity components are equally probable, subject to an overall weighting factor depending on temperature. The distributions, as yet un-normalized in velocity and kinetic energy for a three-dimensional gas are:

$$\begin{aligned} dN &\sim dv_x dv_y dv_z [e^{-Mv^2/2kT}] \\ dN/dv &\sim v^2 [e^{-Mv^2/2kT}] \\ E &= Mv^2/2, \quad dE/dv = Mv \\ dN/dE &\sim \sqrt{E} [e^{-E/ktkT}] \end{aligned} \tag{5.2}$$

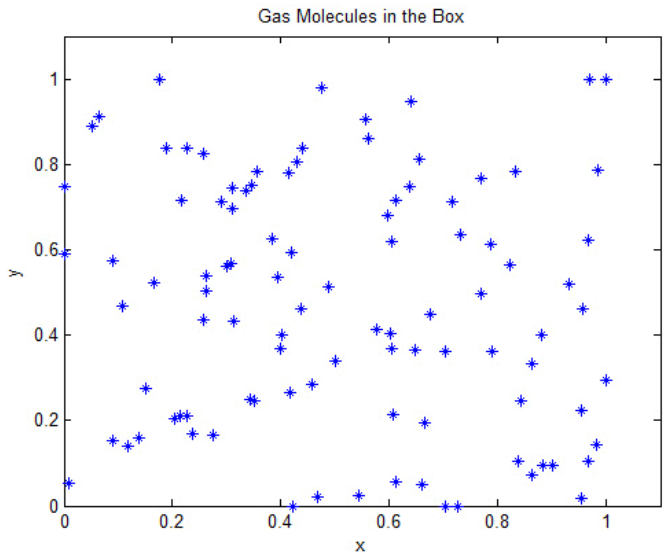


Figure 5.5: Snapshot of the movie of a two-dimensional area containing "molecules" with a distribution of energies and initially random positions and angular directions of the velocities within the area.

The distribution of kinetic energy, dN/dE, follows from the Jacobian connecting energy, E, and velocity, $dN/dE = dN/dv(dv/dE)$, $dv/dE \sim 1/v$. It is easy to see that in two dimensions, the velocity distribution goes as $dN/dv \sim v$ times the exponential factor, while in one dimension $dN/dv \sim 1$ times the exponential. The energy distributions goes as $\sim$1 times the exponential and $1/\sqrt{E}$ times for the two- and one-dimensional cases, respectively. These results were already quoted in Section 5.2 for the two dimensional gas example.

The normalized distributions for velocity and energy are computed symbolically in the script "moments_max_boltz". In addition, expressions for the mean, root mean square and most probable values for energy and velocity are also computed and printed. The results are given in Equation (5.3). The energy is in kT units, while the velocity is also computed in dimensionless units, in this case $\sqrt{kT/M}$. The distributions are shown in Figures 5.6 and 5.7, respectively. The distribution in velocity is more clustered because of the dependence

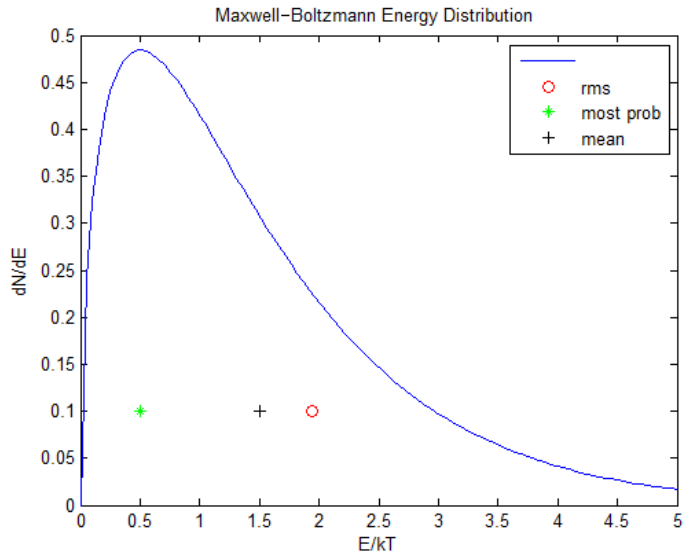


Figure 5.6: Distribution of kinetic energy in kT units for a Maxwell–Boltzmann gas.

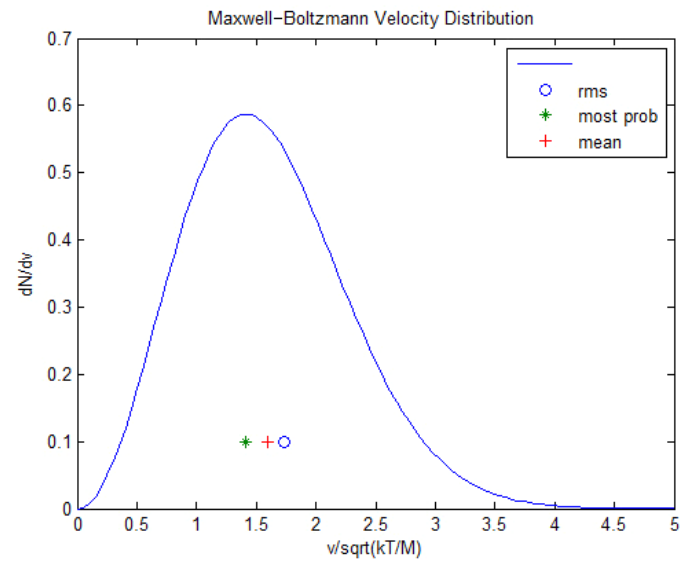


Figure 5.7: Distribution of velocities for a Maxwell–Boltzmann gas.

of the distribution on the square of the velocity compared to the energy distribution where the factor is the square root of the energy.

$$\begin{array}{lll} & \text{Velocity} & \text{Energy} \\ \text{Mean} & \sqrt{8kT/\pi M} & (3/2)kT \\ \text{r.m.s.} & \sqrt{3kT/M} & (\sqrt{15}/2)kT \\ \text{most prob} & \sqrt{2kT/M} & kT/2 \end{array} \tag{5.3}$$

5.4. Fermi-Dirac and Bose-Einstein Distributions

Quantum mechanics implies that non-interacting particles still have effects due to the spin and statistics obeyed by fermions and bosons. For bosons there can be any number of particles in a given quantum state, while for fermions, the Fermi Exclusion Principle requires at most two particles for spin $1/2$, in a particular spatial quantum state. Therefore, at low temperature, bosons will tend to pile up in the lowest quantum state, while for fermions, all states up to some maximum energy will be populated by a pair of fermions. The functions, f, are the mean occupation number of a quantum state of energy, E, approximated as a continuous variable, since the spacing between quantized energy levels is small and un-normalized here in the three cases are shown in Equation (5.4):

$$\begin{array}{ll} \text{Maxwell–Boltzmann} & f_{MB}(E) = e^{-E/kT} \\ \text{Fermi–Dirac} & f_{FD}(E) = 1/[e^{E/kT} + 1] \\ \text{Bose–Einstein} & f_{BE}(E) = 1/[e^{E/kT} - 1] \end{array} \tag{5.4}$$

The normalization will be calculated later. It is defined by replacing the energy, E, by $E - \mu$, where μ is the chemical potential which then normalizes the functions to be the mean occupation number. A plot of these three distributions is shown in Figure 5.8. At low temperatures, the Fermi–Dirac distribution becomes constant because all the states are full, while for the Bose–Einstein case there is no limitation. At high temperatures, the states are only sparsely filled, so that the two quantum distributions and the classical Maxwell–Boltzmann distribution are all similar. Note that these distributions are not yet properly normalized, nor are the power law energy factors

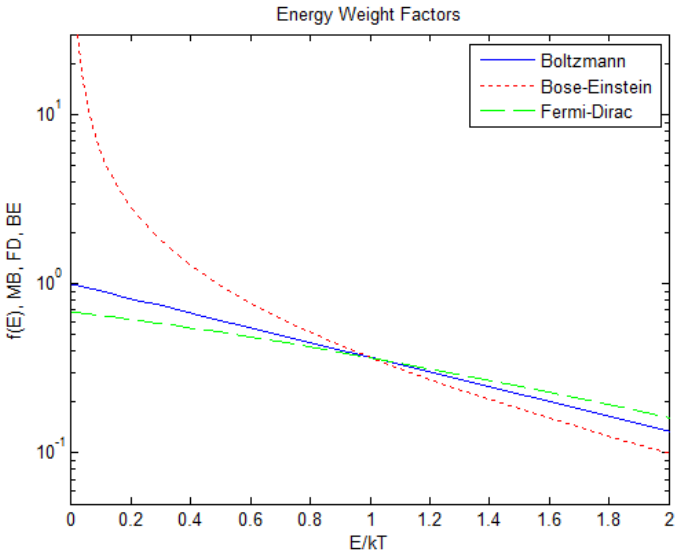


Figure 5.8: Energy distributions for the three cases of a classical gas or a fermionic or bosonic gas.

having to do with the "phase space" of the particles included as they were for the classical gas, Equation (5.2), in Section 5.3. The three distributions are defined to agree at $E = kT$.

In the quantum case, the density of states, dn/dE, the number per unit volume, n, has an energy distribution which is normalized to the possible number of quantum states. The wave numbers are quantized in a box of side L, $k_x = (2\pi/L)n_x$, which leads to a number per unit volume and momentum of, $dn = d\vec{k}/(2\pi)^3, d\vec{p} = \hbar d\vec{k}$. The distribution can then be converted to energy, E. The resulting value for dn/dE is proportional to $1/\hbar^3$, and the volume of available states in position and momentum is found to be proportional to the quantum graininess of the world, as one might expect. This sets the proportionality which was lacking in Equation (5.2).

$$dn/dE \sim \sqrt{2}M^{3/2}\sqrt{E}/\hbar^3\pi^2 \tag{5.5}$$

5.5. Chemical Potential, Bosons

The normalization of the mean occupation numbers of a quantum state of energy E, $f(E)$, will now be calculated for bosons. The normalization is fixed by the total number of particles, N, and the volume in which they are contained, V. The number density is $n = N/V$, while the mass density is $\rho = mn$, where m is the mass of the atom or electron. The energy density is u while the total energy is U. The spectral energy density is $u(E)$, whose integral is u.

The particles effectively have energies reduced by the "chemical potential", $E \to E - \mu$ in Equation (5.4), which is, for example, the work function of electrons bound in a metal. The chemical potential must be evaluated by properly normalizing the probabilities.

In the Bose–Einstein case, for non-relativistic particles, the number of states goes as the square of the momentum, as shown already in the Boltzmann case. The integral for n is then, ignoring numerical factors, $dn \sim \int \sqrt{E}[1/(e^{(E-\mu)/kT} - 1)]dE$. This integral can be done in closed form. Integrating the distribution over the particle energy, E, the number density and total energy, U, are:

$$\begin{aligned} N/V = n &= s(mkT/2\pi\hbar^2)^{3/2}\zeta_{3/2}(e^{\mu/kT}) \\ U &= 3/2kTV(mkT/2\pi\hbar^2)^{3/2}\zeta_{5/2}(e^{\mu/kT}) \\ U &\to 3/2kTN \end{aligned} \tag{5.6}$$

The number of quantum states of spin is s. The Riemann zeta function, ζ, arises from integrating the spectral number density over all energies, and the chemical potential μ is a normalization factor. The zeta function has a radius of convergence from zero to one, or chemical potentials from zero to infinity. The energy density, u, arises from integrating the spectral energy density, $u(E) = En(E)$ over all energies and it also contains a zeta function. MATLAB has utility functions to evaluate the zeta functions as an infinite power series.

If $n = N/V$ is small or the temperature is high, then the Riemann zeta function ratios in U/N are roughly one, and the classical result shown in Equation (5.6) is recovered. However, at low temperatures, the inter-particle spacing becomes comparable to the de Broglie wavelength, h/P, and a critical temperature is reached when

the chemical potential is zero. This corresponds to the behavior of liquid helium spin zero nucleus as the temperature is decreased. At a critical temperature it becomes a "super-fluid".

Taking $s = 1$, $2.612 =$ the zeta function of order $3/2$ and of argument one, or chemical potential equal to zero, the critical temperature when this occurs is estimated to be:

$$\begin{aligned} T_c &= 2\pi\hbar^2/Mk[(n/s\zeta_{3/2}(1))^{2/3} \\ E_c &= kT_c = \hbar^2 k_c^2/2M \\ k_c &= \sqrt{4\pi}[n/s\zeta_{3/2}(1)]^{1/3} \end{aligned} \tag{5.7}$$

The critical temperature for He, mass M, is evaluated in the script "Chemical_Potential". The zeta functions are evaluated as a series of five hundred terms, and the relationship of N/V to the chemical potential at a given T is evaluated using the MATLAB function "fminsearch" in order to solve for the chemical potential explicitly using Equation (5.6) at fixed number density. The resulting solutions are plotted in Figure 5.9. The estimated critical temperature, using

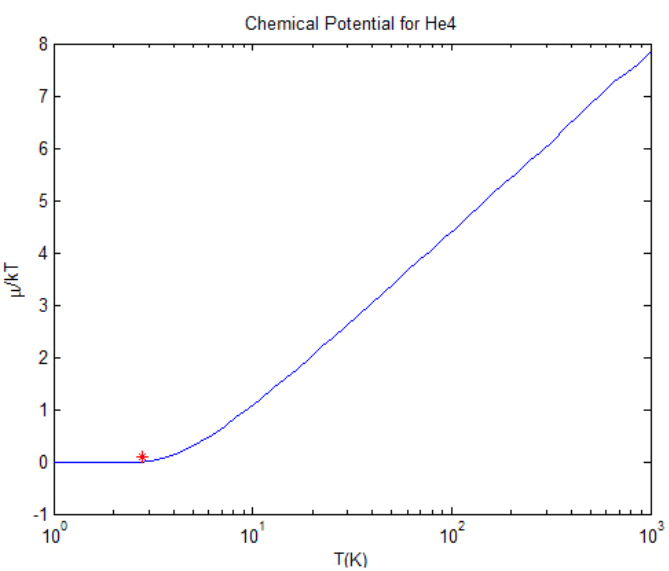


Figure 5.9: Plot of the chemical potential for He as a function of temperature. The approximate result of T_c at zero potential is shown as a *.

Equation (5.7) for zero chemical potential, is $T_c = 2.8\,\mathrm{K}$. The experimental number when helium becomes a super-fluid is 5.2 degrees. The explicit calculation of the chemical potential as a function of temperature is made available using the powerful MATLAB tools which are provided.

Note in Equation (5.7) that the wave number scales as the one-third power of the number density. This behavior is also true for the chemical potential for fermions. Critical behavior occurs when the wave number becomes comparable to the intermolecular spacing. At that temperature, quantum effects can be expected to manifest themselves.

5.6. Chemical Potential, Fermions

The Fermi level is often thought of as the chemical potential at zero degrees, where all states below that energy are filled so that the occupation number is one below the "Fermi Energy" and zero above it. In this case the integration is trivial, $n \sim \int_0^{E_F} \sqrt{E} dE \sim E_F^{3/2}$.

$$\begin{aligned} E_F(0) &= \hbar^2/2m_e(3\pi^2 N/V)^{2/3} \\ &= \hbar^2 k_F^2/2m_e, \quad k_F = (3\pi^2 n)^{1/3} \end{aligned} \tag{5.8}$$

All states above the Fermi energy are empty because no thermal excitation is possible. The example used is for Li, and an approximate expansion for the temperature dependence of the Fermi Energy is plotted in Figure 5.10. The zero temperature result simply again follows from the density and is 4.7 eV, or an equivalent temperature of 56,000 degrees. The very high characteristic temperature is a reflection of the fact that for bosons, the energies are smaller than the classical case due to clustering, while for the fermions the effective energies are pushed higher due to the requirements of the exclusion principle. At room temperature, the approximation of using the zero temperature Fermi energy is often very useful.

The energy density can easily be found in the low temperature limit.

$$u(o) = 3/5(nE_{_F}) \tag{5.9}$$

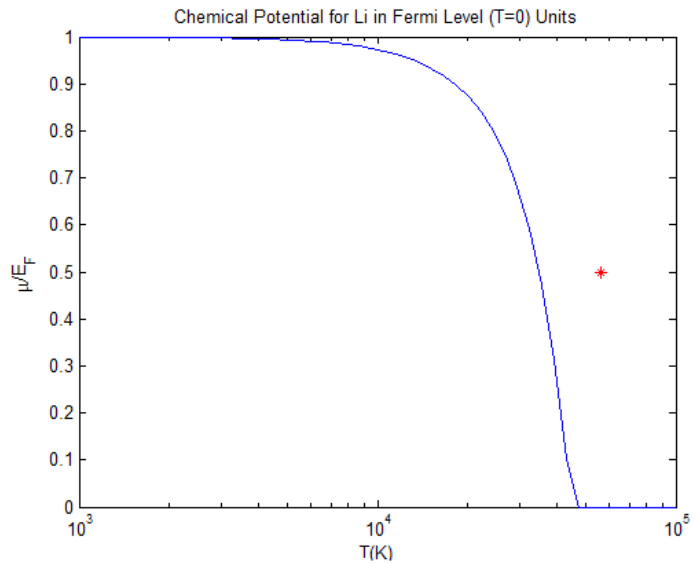


Figure 5.10: Approximate calculation of the temperature dependence of the chemical potential divided by the $T = 0$ Fermi level for Li. The red * is the $T = 0$ Fermi level in temperature units.

5.7. Critical Temperature for He

The temperature dependence of the chemical potential was discussed in a previous section. Further calculations are made available in the script "Crit_Temp_He4". Printout from that script is shown in Figure 5.11. The user chooses the atomic weight of the atom.

```
>> Crit_Temp_He4
  Mean Energy per Molecule vs. Temp for BE Gas - Critical Temp


Critical Temperature for a Bose-Einstein Liquid
k*Tc = (2 * pi *hbar^2)/m[(N/V)/s*zeta3/2(1)]^2/3
Input the Atomic Weight of a Spin 0 Atom (s=1): 4
Critical Energy and Temperature = 0.000241004 eV, 2.79425 oK
Wavelength Scales as sqrt(1/T*A)
Critical Temperature Scales as rho^2/3 / A^5/3
```

Figure 5.11: The estimated critical temperature is a repeat. The appropriate scaling of important parameters is also indicated.

It has already been commented that in a Bose-Einstein gas, the mean energy per atom is less than that for a classical system. That statement is quantified in Figure 5.12. The plot is the mean energy of He as a function of temperature scaled to the classical mean energy. This behavior is very characteristic of a bosonic system. Note that what is plotted are approximations for large and small values of T/T_c. The experimental critical temperature for He is also indicated by a red *.

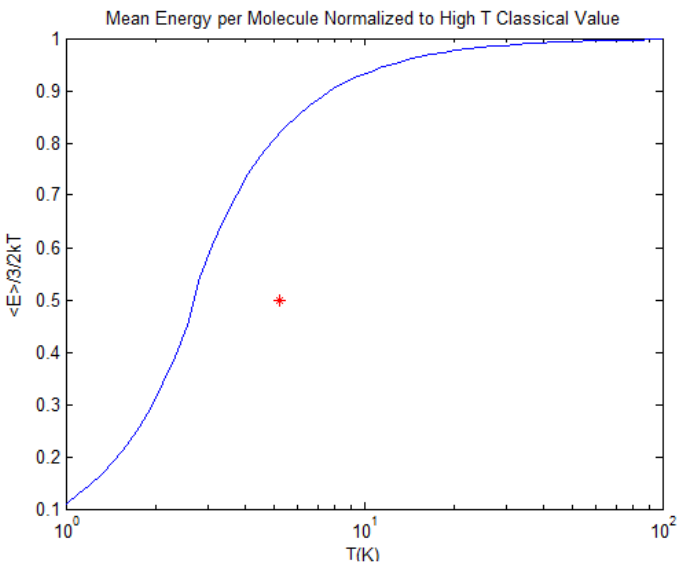


Figure 5.12: Mean thermal energy of He as a function of temperature relative to the classical result. The red * is the experimental critical temperature.

The behavior of the mean energy has two expansions, at low temperatures and at high temperatures one of which follows from Equation (5.6):

$$\begin{aligned} \langle E \rangle &\sim (1 - T_c/T)^{3/2} \\ \langle E \rangle &\sim [\varsigma_{5/2}(e^{\mu/kT})/\varsigma_{3/2}(e^{\mu/kT})](T/T_c)^{3/2} \end{aligned} \tag{5.10}$$

Quantum effects are expected at low temperatures because, as the temperature falls, the momentum decreases and hence the

deBroglie wavelength rises. When that wavelength exceeds the typical spacing between atoms, estimated to be approximately $(1/n)^{1/3}$, quantum effects will become important.

$$\lambda_{dB} \sim 2\pi\hbar/\sqrt{3kTM} \tag{5.11}$$

A plot of the temperature dependence of the de Broglie wavelength using Equation (5.11), and the interatomic spacing, assumed to be constant, is shown in Figure 5.13. The curves cross near the critical point for helium, indicated by a red *, as expected. Note that from Equation (5.6) and (5.7), $n \sim 1/\lambda_{dB}^3$, $k \sim 1/\lambda_{dB}$ as is expected from simple dimensional arguments.

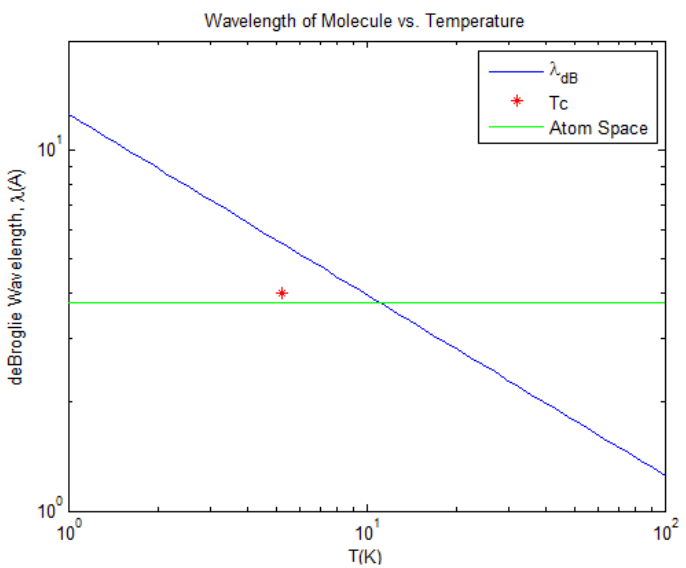


Figure 5.13: Temperature dependence of the de Broglie wavelength for helium in comparison to the interatomic spacing in angstroms. The experimental value of the critical temperature is indicated by a red *.

5.8. Exact Fermion Chemical Potential

The integrals that need to be evaluated such as that for the number density, $n \sim \int_0^\infty \sqrt{E}/[e^{(E-\mu)/kT} + 1]dE$, in order to find the fermion

chemical potential are not analytically tractable in the Fermi–Dirac case. Because of that, the script "Fermi_Dirac_Tne0" uses the MATLAB numerical integration tool "quad" to make numerical evaluations. The specification of the number density gives an implicit relationship to the chemical potential as noted previously. At $T = 0$, the number density should go as the third power of the Fermi energy, as displayed by the script and quoted above.

Operationally, a chemical potential is chosen and the number density is solved for using "quad". This procedure is done as a function of the temperature. The results are plotted as a surface of density in the variables of temperature and chemical potential, called Fermi energy here, E_F, in Figure 5.14. A contour plot of constant density in the temperature — chemical potential plane is given in Figure 5.15. Also plotted there are points from an expansion in powers of T around $T = 0$ for the Fermi energy, used in computing points for Figure 5.10. The approximate behavior is quite good if the temperature is low, however, at higher temperatures it becomes negative and unphysical.

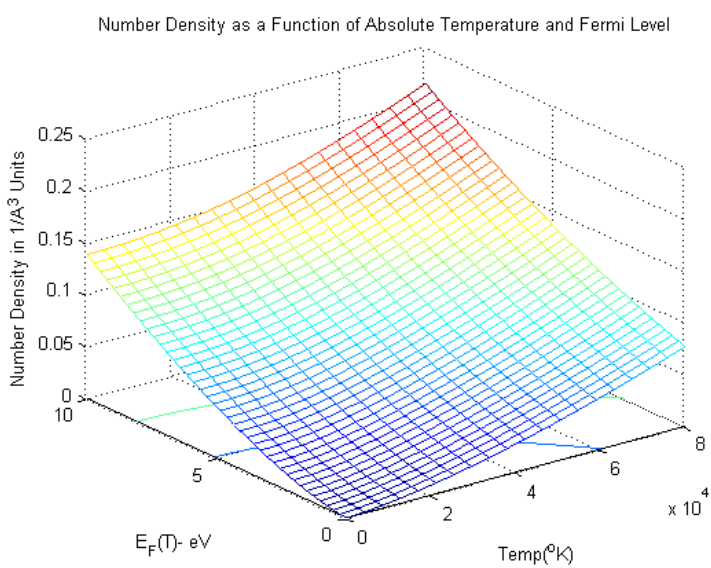


Figure 5.14: Density as a function of temperature and chemical potential.

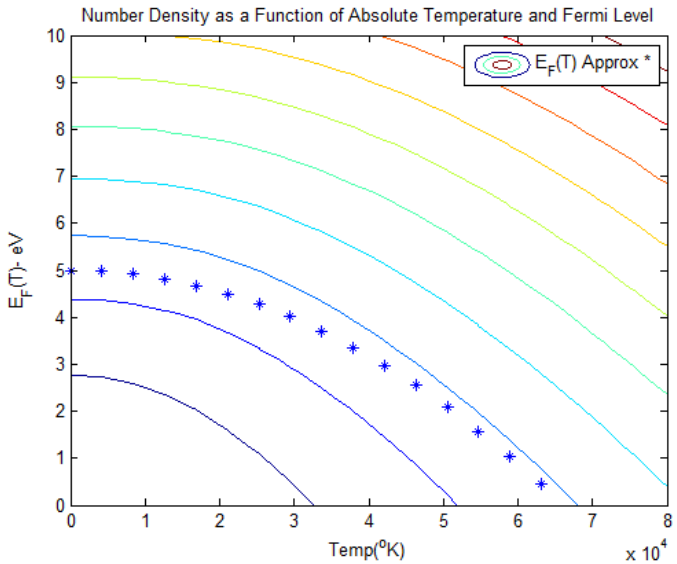


Figure 5.15: Contours, from the numerical exercise of Figure 5.14, of constant density as a function of temperature and chemical potential. The blue * are a power law expansion in temperature used in Figure 5.10, which becomes negative at high T.

Given the ubiquity of solid state electronics in our culture, a nodding acquaintance with the energetics of electrons is a useful thing.

Clearly, as the temperature increases, the Fermi level increases. Higher number densities imply higher Fermi energies, albeit with a weak dependence on n.

5.9. Complex Variables and Flow

The complex variable techniques which were used in two dimensional electrostatics can also be applied to fluid flow, since the underlying mathematics is the same. The potentials of electromagnetism become the streamlines, and the gradient which defined the electric fields becomes the fluid velocity. Otherwise the same differential equation is being solved in the two cases. For a fluid flow potential of Φ, the velocity of the fluid flow is $\nabla\Phi = \vec{v}$. Note that this flow formulation only applies to the ideal case of no friction or viscosity for the fluid.

The flow streamlines for an obstacle placed in a uniform flow along the x axis are explored in the script "Flow_windtun". Plots of the unobstructed flow, scale chosen by the user, with a circular obstructing object and with a linear object are presented as choices. In the linear case, an upright line is shown and then the user is asked to give an angle to the linear obstacle. The streamlines for the vertical line are shown in Figure 5.16, while the same object inclined by thirty degrees appears in Figure 5.17.

5.10. Complex Variables and Airfoils

A mapping exists which gives an obstacle with a shape similar to an airplane wing. These are called Joukowski profiles. The script used is "Flow_Airfoil". The user dialogue allows for a choice of a parameter which defines the airfoil shape and also a choice of the angle of attack of the shape. There is an angle of attack with respect to the external flow of sixty degrees in Figure 5.18.

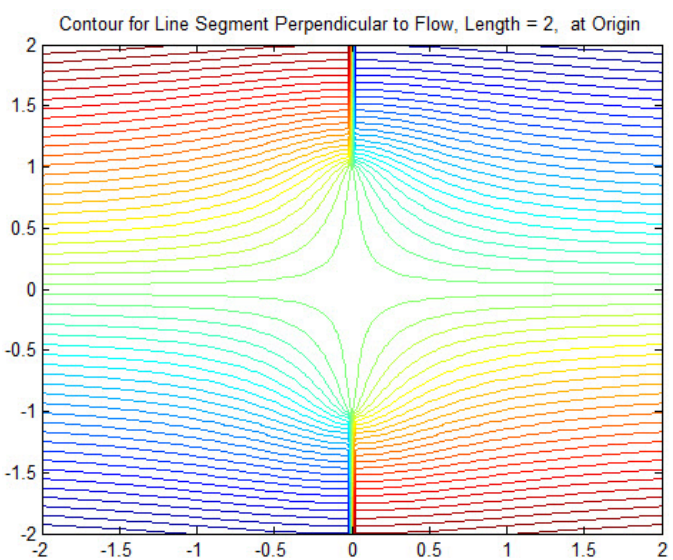


Figure 5.16: Streamlines for a vertical linear obstacle placed in flow along the x direction.

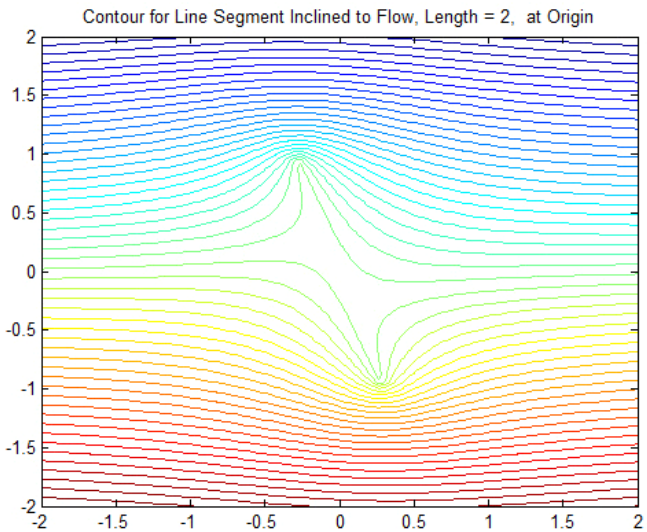


Figure 5.17: Streamlines for a linear obstacle inclined at 30 degrees placed in flow along the x direction.

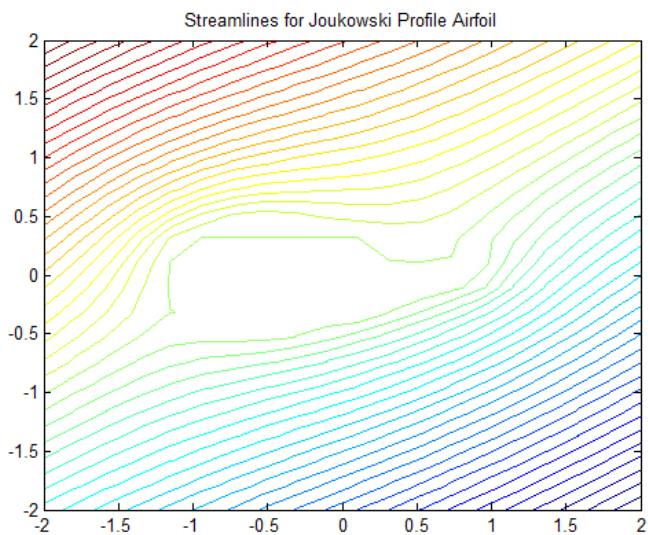


Figure 5.18: Streamlines in the specific case of an airfoil shape defined by $R = 0.5$ and with an angle of attack with respect to the external flow of 60 degrees.

The velocity vectors corresponding to the streamlines of Figure 5.18 appear in Figure 5.19. The view enhances the idea that this airfoil has "lift".

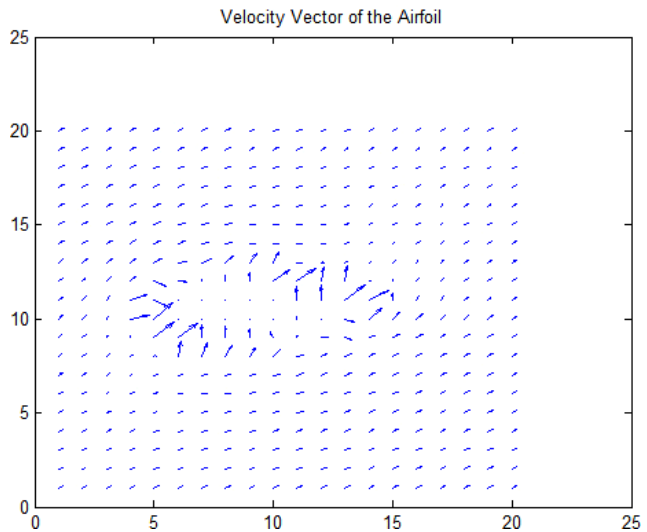


Figure 5.19: Velocity vector map of the fluid flow around the airfoil of Figure 5.18.

5.11. Complex Variables and Sources of Flow

There are several flow sources displayed in the script "Flow_Source", including both sources and sinks, rotational flow and barriers. One example taken from the script is a point source of flow in the presence of a barrier at x equal to zero, which extends over all y, while the source is at positive x on the y axis.

The streamlines for this case appear in Figure 5.20, while the x value of the velocity is displayed in Figure 5.21.

These complex variable techniques give a very nice visual presentation of the potentials/streamlines and fields/velocity vectors. The interested user can modify the provided scripts and then try different geometric configurations since there are many mappings provided in the literature. For example, a GOOGLE search yields http://www.math.umn.edu/∼olver/pd_/cm.pdf.

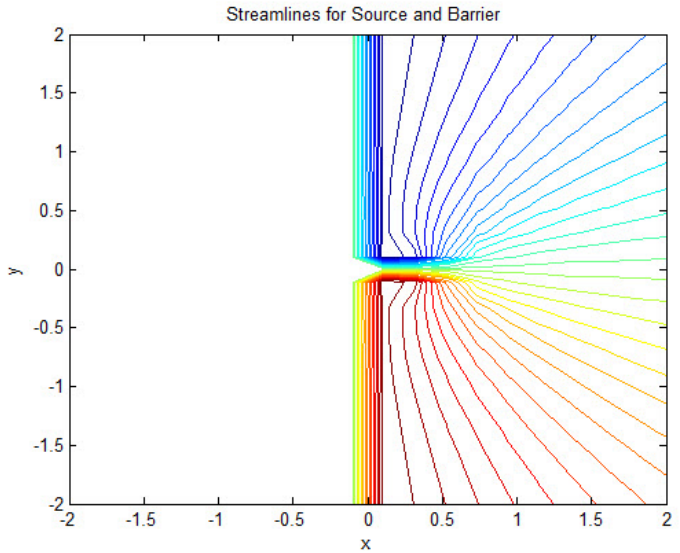


Figure 5.20: Streamlines for the case of a point flow source located on the y axis in the presence of a barrier at x equal zero covering all y locations.

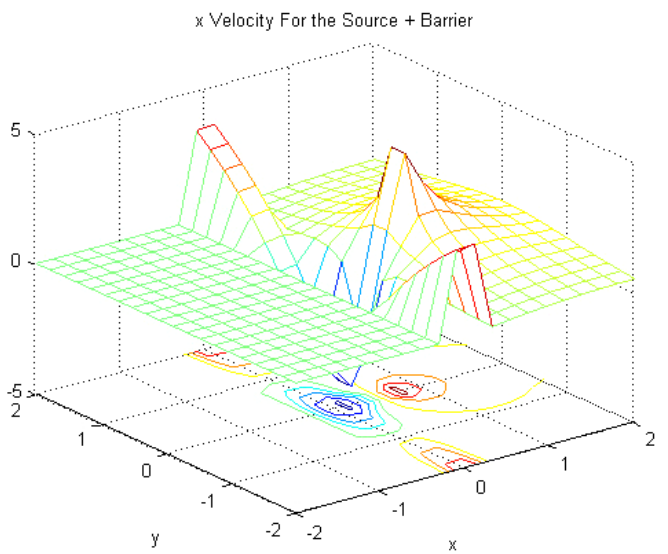


Figure 5.21: Velocity along the x direction for the case of a point flow source located on the y axis in the presence of a barrier at x equal zero covering all y locations.

5.12. Viscosity Model

Ideal gases do not self-interact, which was assumed in the flow examples shown above. As a simple attempt to make a more realistic gas model, the script "Viscosity_Model" allows the molecules of the gas to elastically scatter off one another. The size of the molecules in the numerical, two dimensional simulation, is such that 400 molecules would fill the area of the gas.

The user chooses the number of molecules, the number of time steps to follow the gas volume, the gas temperature, and the acceleration due to an applied field. An ideal gas would simply flow in the field direction without impediment. The acceleration might be due to a heat differential as in thermal conductivity or an electric field in the case of electrical conductivity of electrons or ions in a gas. An example of the dialogue appears in Figure 5.22.

```
>> Viscosity_Model
  Model with gas-gas collisions -> viscosity


Input the Number of Gas Molecules : 100
Input the Number of Time Steps : 100
Input the Gas Temperature * k, Mass = 1: 1
Mean Velocity 1.42661
Input Acceleration due to External Field ~ 1: 2
Number of Wall Collisions 1014
Momentum Impulse to Walls 2601.59
Momentum Impulse to Left and Right x Walls 345.32 , 1090.34
Momentum Impulse to Top and Bottom y Walls 512.138 , 653.796
```

Figure 5.22: User dialogue for evaluating the effect of an external acceleration.

If there were no collisions, the molecules would be swept to the right x wall of the gas area. The wall collisions and the temperature are treated in the same way as in the case of the ideal gas model. The script in this case checks for a collision between molecules, and randomizes the velocity direction in case of a collision.

The user can watch the movie of the evolution of the system. In addition, the momentum transferred to the walls is tracked and

printed out. As can be expected for no acceleration, the left and right momentum transfers are equal — statistically. An acceleration of "2" enhances the right momentum transfer with respect to the left. Increasing the temperature reduces the left-right asymmetry, since the thermal motion washes out the acceleration. Finally, reducing the number of molecules enhances the asymmetry because the number of collisions between gas molecules which goes as the square of the number of molecules, is reduced. The last frame of a movie arising from the dialogue above appears in Figure 5.23 where the asymmetry in space is also evident.

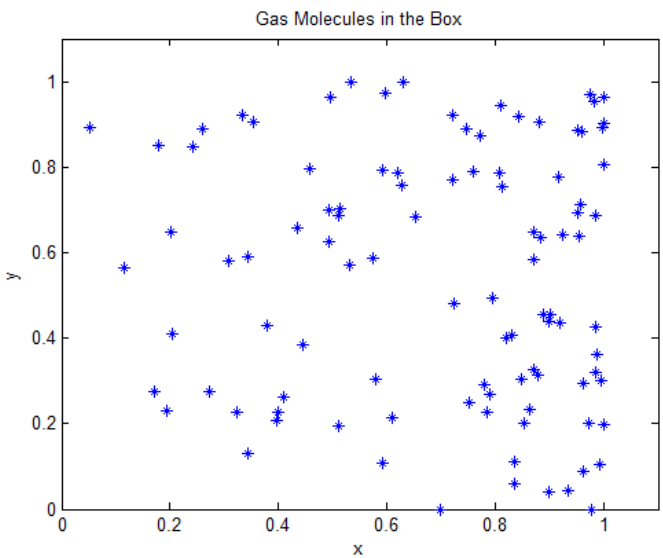


Figure 5.23: Last frame of the movie for the specific choice of 100 molecules, with temperature = 1 and acceleration = 2.

For example, an electron in an applied electric field $\vec{E}$ in a gas has a drift velocity v_d:

$$\begin{aligned} v_d &= (e\vec{E}/m_e)\tau \\ \tau &= 1/\langle v_T \rangle \sigma n \end{aligned} \tag{5.12}$$

The mean time between collisions, τ, depends on the average thermal velocity, $\langle v_T \rangle$ the collision cross section, σ, and the gas

number density, n. As a result, the drift velocity of the medium is proportional to the average thermal velocity. The electrical conductivity, κ_e, or current flow per unit electric field, is $\kappa_e = ne^2\tau/m$, proportional to the drift velocity.

5.13. Transport and Viscosity

Transport phenomena in gases and liquids can be formulated using the Maxwell–Boltzmann velocity distribution. An example is worked out in the script "MB_Transport". The collision cross section, σ, for a hard sphere gas is taken to be π times the diameter of a single molecule squared by geometry. The mean thermal velocity of the medium is $\langle v_T \rangle$, which was already evaluated to be $\sqrt{8kT/\pi m}$. The number of collisions suffered by a given molecule per unit time is:

$$N_c = n\sigma\langle v_T \rangle/\sqrt{2} \tag{5.13}$$

where n is the number of molecules per unit volume. The mean time between collisions is the inverse of N_c. The mean free path, L, between collisions is $\langle v_T \rangle$ times the mean time between collisions, $L = \tau\langle v_T \rangle = \langle v_T \rangle/N_c = \sqrt{2}/n\sigma$, which is independent of temperature. The viscosity, η, can then be evaluated and depends only on the thermal velocity and the cross section:

$$\eta = mnL\langle v_T \rangle/3 = \sqrt{2}m\langle v_T \rangle/3\sigma \tag{5.14}$$

The printout from the script for the user choice of hydrogen gas is shown in Figure 5.24.

```
>> MB_Transport
  Transport and viscosity and the Maxwell - Boltzmann distribution


Cross section for gas scattering = pi*(2*r)^2 = 5.78053e-15 (cm^2)
Mean thermal velocity at 300 oK = sqrt(8*kt/pi*mp*A) = 174761 (cm/sec)
Number of molecules/cm^3 = No*rho/A = 2.5284e+19
Number of collisions/sec = Nc = n*sig*v/sqrt(2) = 1.80611e+10
Mean time between collisions = tau =  1/Nc = 5.53677e-11 (sec)
Mean free path (cm) between collisions) = v*tau = 9.67613e-06
Viscosity = (rho*lmean *v)/3 =  (gm/(cm*sec)) = 4.73483e-05
```

Figure 5.24: Printout for a user defined choice of H gas. Other options are He, N_2, or O_2 either as a gas or a liquid.

The experimental value for the viscosity of H_2 at STP is $\sim 9.5 \times 10^{-5}$ gm/(cm*sec). The treatment here implicitly assumed that the medium is dilute and that the mean free path is much greater than the size of the molecules, which is appropriate for a gas since the mean free path is approximately 1000 times the size of the molecules which are typically angstroms. The temperature dependence of H_2 gas in this model is shown in Figure 5.25, where the square root behavior of the mean thermal velocity is observed. The simple treatment given here is approximate but qualitatively correct.

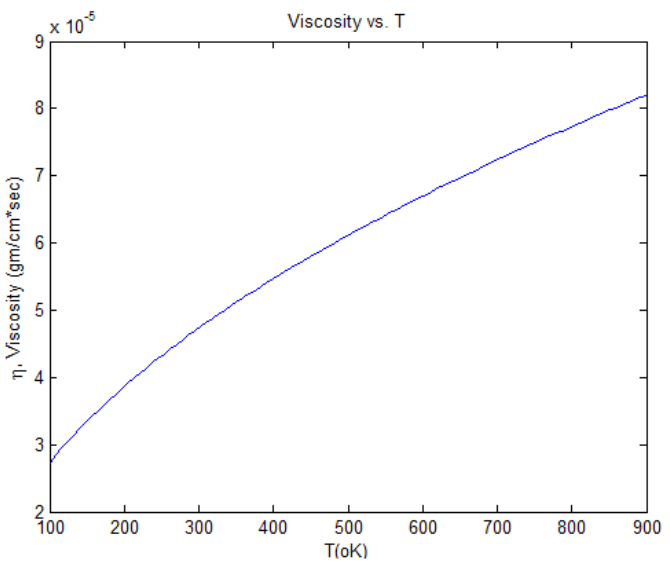


Figure 5.25: Temperature dependence of the viscosity of H_2 gas.

5.14. Fluid Flow in a Pipe

The next level of approximation for flow is to allow for internal friction, or viscosity, but not take turbulence into account. In that case, the flow is laminar and the boundary conditions are that the velocity of the fluid at the pipe interface is zero. For a fluid with viscosity, there are collisions which create friction in the fluid. For example, there is a drag force, $F_d \sim a\eta v$, on an object of size a in a fluid

moving with velocity, v, and having viscosity, η. The equations for fluid velocity, v, as a function of radius, r, from the pipe center and for the volume, V, flow with time are:

$$dv/dr = -pr/(2\eta L) \tag{5.15}$$

$$v = p/(\eta L)[R^2 - r^2] \tag{5.16}$$

$$dV/dt = \pi p R^4/(8\eta L) \tag{5.17}$$

The pressure is p, the pipe length is L and the viscosity is η. The volume flow is proportional to the fourth power of the pipe radius and proportional to the driving pressure divided by the pipe length.

Printout for the script "Flow_Pipe" is displayed in Figure 5.26. The user dialogue covers the choice of driving pressure, the pipe length and transverse size, as well as a choice of two geometries; a circular pipe or flow between two infinite plates. The fluid is assumed to be water at near room temperature. The maximum velocity is computed as is the overall volume flow of the water for the pipe. The shape of the velocity profile follows from the boundary conditions which require vanishing velocity at the pipe boundary and is shown in Figure 5.27 for a specific choice of parameters for the circular pipe option.

```
 Frictional Force in Fluid Balances Pressure Force
 dv/dr = Velocity Gradient = -Pr/(2*eta*L)
 v = velocity, P = pressure, R,L = pipe radius, length, eta = viscosity
Enter Pressure in Atm: 1
Enter Pipe Length in m: 1000
Enter Pipe Radius/ Plate Separation in m: 0.05
Velocity at r = 0 = 62.1272 (m/sec)
Flow for This Setup = 0.243973 (m^3/sec)
```

Figure 5.26: Printout for a specific example of viscous flow in a pipe.

Laminar flow is a reasonable description of viscous flow in the absence of turbulence. Ideal flow, on the other hand, has no frictional forces. This idealization is not as realistic, but it does admit of solutions which are known from electrostatics since the Laplace and Poisson equations apply.

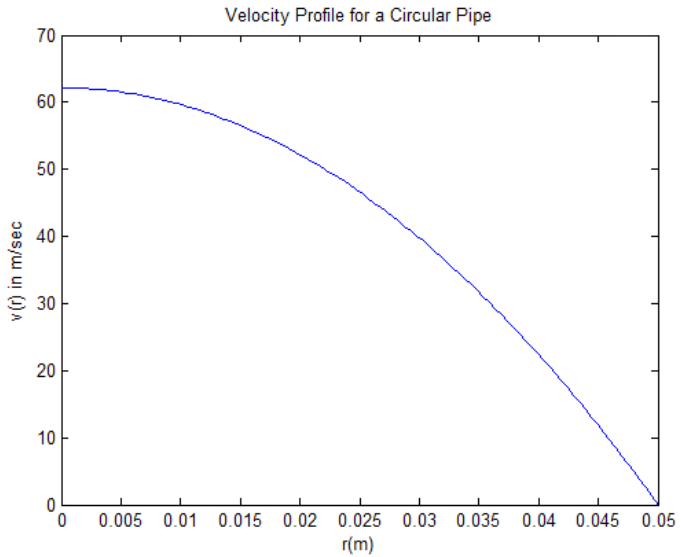


Figure 5.27: Transverse velocity profile for the case of a circular pipe with parameters shown in Figure 5.26.

5.15. Heat and Diffusion

MATLAB has the ability to solve partial differential equations numerically in one spatial dimension. The script is called "pdepe" which can handle one dimension in space and first order partial derivatives in time. The heat diffusion equation is studied numerically, solving:

$$\kappa \partial^2 T / \partial^2 x = \partial T / dt \tag{5.18}$$

The thermal conductivity is κ. The spatial and temporal coordinates are x and t, respectively. The script solves for the temperature, $T(x,t)$, distribution subject to initial conditions and boundary conditions. The boundary conditions are that T vanishes at the extreme values of x. They should be placed far enough from the regions of interest so as not to affect the results.

The script which was written to solve the heat equation is "PDE_Heat". Initial conditions for the temperature distribution $T(x,0)$ are supplied and the value and first derivative of T on the x

boundaries are fixed. The shape of the initial $T(x,0)$ distribution is chosen by the user from four possibilities. A more advanced use would be to have the user rewrite $T(x,0)$ to enable an arbitrary symbolic functional input or alter the conductivity κ.

The results for an initial square distribution at starting and ending times are shown in Figure 5.28. The user sees a movie with all the intermediate time solutions as the temperature evolves in time. The heat distribution diffuses into the initially cool regions as expected. A second option is shown in Figure 5.29.

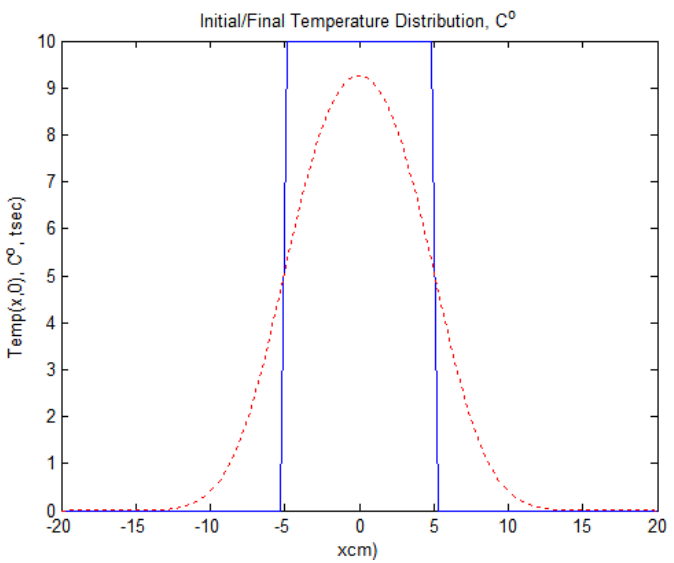


Figure 5.28: Initial, blue, and final, red, temperature distributions for an initially square distribution of temperature.

In addition, the heat equation is close in structure to the Schrodinger equation, except that the appearance of i in the latter allows for diffusion but also to contain oscillatory solutions. This aspect will be taken up in the next section which looks at quantum mechanical demonstrations. As with the heat equation, a numerical study of the solutions of the Schrodinger equation will be made using wave packets to simulate a "particle" localized in position

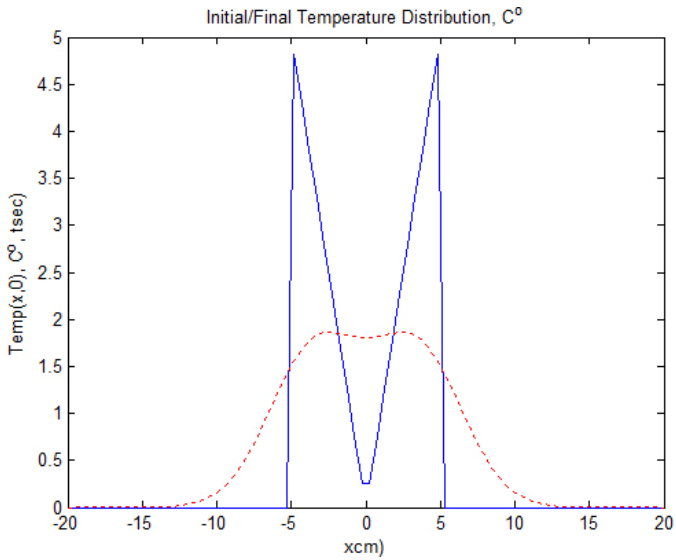


Figure 5.29: Initial, blue, and final, red, temperature distributions for an initial distribution with sharper structure than that in Figure 5.28.

and momentum, but whose probability density spreads spatially in time, rather like the behavior of the temperature distributions shown above. Clearly, many topics in physics are inter-related.

Some intuition about the diffusion of heat can be gained by trying the several examples.

Chapter 6

Quantum Mechanics

"We have sought for firm ground and found none. The deeper we penetrate, the more restless becomes the universe; all is rushing about and vibrating in a wild dance."

— **Max Born**

"Those who are not shocked when they first come across quantum theory cannot possibly have understood it."

— **Niels Bohr**

"Photons have mass? I didn't even know they were Catholic."

— **Woody Allen**

6.1. Preliminaries — Planck Distribution

The advent of quantum mechanics began with the exploration of black-body radiation. This problem is now viewed as the behavior of a photon gas with zero chemical potential. In the last section, the chemical potential was defined for a system with a fixed number of particles. In this case, photons can be emitted and absorbed, so that the chemical potential must be zero. The Planck distribution in photon number is as assumed classically; all momentum, P, components are equally probable. However, since the photon is massless, this leads to a number distribution that goes as the square of the energy, E, and a total energy, U, distribution that goes as the third power, modulated by the Bose–Einstein factor.

$$\begin{aligned} dn/dP &\sim P^2 dP/[e^{E/kT} - 1] \\ E &= cP, du/dE = E dn/dE \\ dn/dE &\sim E^2 dE/[e^{E/kT} - 1] \end{aligned} \tag{6.1}$$

The moments of the distribution are evaluated in the script "Moments_Planck2". The results are plotted in Figure 6.1 below.

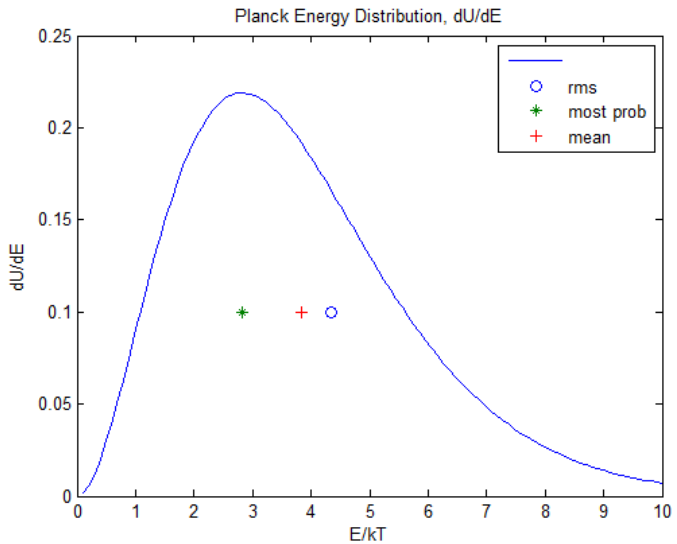


Figure 6.1: Energy density distribution for a photon gas with zero chemical potential.

The script printout gives the symbolic values for the different moments.

The integrals are done symbolically and are printed out. The mean energy is expressed in terms of the Riemann zeta functions of argument one (zero chemical potential), which is, in kT units, numerically equal to 3.83. Since du/dE goes as the cube of E, the energy density goes as kT to the fourth power, which is the Stefan–Boltzmann law. The thermal energy density is u, while the power per unit area, or the luminosity is L.

$$\begin{aligned} \langle E/kT \rangle &= 360\zeta_5(1)/\pi^4 \\ \sqrt{\langle (E/kT)^2 \rangle} &= \pi\sqrt{40/21} \\ u &= 4/c(\sigma T^4), L = \sigma T^4 \\ \sigma &= \pi^2 k^4/(60\hbar^3 c^2) \end{aligned} \tag{6.2}$$

The result for u is simply, $u = 4\pi^2(kT)^4/60(\hbar c)^3$, where kT has dimensions of energy and $\hbar c$ has dimensions of energy times distance. Numerically σ is 5.67×10^{-8} W/m^2 $\times$ k^4.

6.2. Bound States — Oscillating or Damped

Quantum mechanics has a description called the Schrödinger equation which is formally similar to the heat diffusion equation which was already studied. The appearance of the imaginary number i, however, allows solutions to this equation which are either oscillatory or exponentially damped. These are then the two main categories, bound states which are exponentially damped far from the confining potential and scattering states which are not so confined. The Schrödinger equation determines the behavior of the wave function ψ whose square modulus gives the probability to observe the object:

$$\begin{aligned} (T+V)\psi &= E\psi \\ P &\to i\hbar\partial/\partial x \\ [-(\hbar^2 2m)\partial^2/\partial^2 x + V(x)]\psi &= E\psi \\ \psi &\sim e^{ikx}, \hbar k = \sqrt{2m(E-V(x))} \end{aligned} \tag{6.3}$$

The kinetic energy is T, the potential is V, and E is now the total energy of the particle. It determines the wave function $\psi(x)$. The classical momentum, P, becomes a differential operator and the oscillatory solutions have a wave number, k, proportional to the square root of $(E - V)$. If E is less than V, the wave number is complex and the wave function is exponentially damped. There is a more general equation allowing for time dependence where the energy, E, is replaced by the operator, $i\hbar\partial/\partial t$.

The sizes of quantum systems can be estimated by looking at the basic quantum, Planck's reduced constant, $\hbar$. The electron mass is expressed in energy units in Equation (6.4), while Planck's reduced constant is expressed in energy–length units. An appropriate atomic length scale is the angstrom, 10^{-10} m. Subsequently, quantities scaled to c, for example Pc, will be written as P and energy units used. This is customary and simplifies the formulae. One can restore the results by inserting c later. For a system characterized by a size of

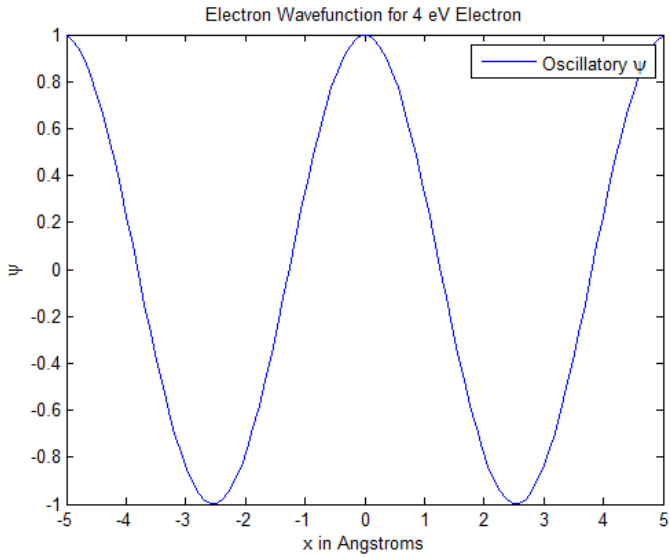


Figure 6.2: Wave function for a 4 eV electron in a constant potential of −2 eV.

one angstrom, the energy scale is a few eV.

$$\begin{aligned} \hbar c &= 2000\,\text{eV}\,\text{Å} \\ m_e c^2 &= 511000\,\text{eV} \\ a = 1\ \text{Å}, \hbar c^2/2m_e c^2 a^2 &= 3.9\,\text{eV} \end{aligned} \tag{6.4}$$

The script "qm_intro" sets up an electron with 4 eV kinetic energy and asks the user for a potential. The resulting plots for a potential $V = -2$ eV and 10 eV appear in Figures 6.2 and 6.3. Note that the wave number is numerically the same in the two cases, 0.81 angstroms, since $|E - V|$ is 6 eV in either case.

6.3. Hydrogen Atom

There are only a few solvable problems in quantum mechanics. One of them is the hydrogen atom, in analogy to the Kepler solution in classical mechanics. The binding energy depends on the principal quantum number, n, as the inverse square. The atomic size, a, rises with n. The overall energy scale is eV and the characteristic size is

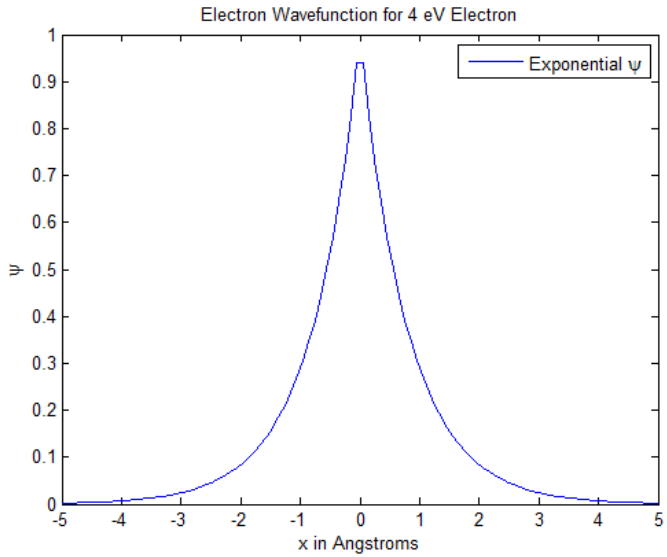


Figure 6.3: Wave function for a 4 eV electron in a constant potential of 10 eV.

the angstrom, Å. The fine structure constant is $\alpha = e^2/\hbar c = 1/137$ and is a measure of the strength of the binding by the Coulomb potential. In the ground state, the energy is $E_o = -13.6\,\text{eV}$ for an electron with mass, m, and charge, e. The speed with respect to light, β, is small, so that non-relativistic mechanics is appropriate.

$$\begin{aligned} E_o &= -mc^2\alpha^2/2 = -13.6\,\text{eV} \\ \alpha_o &= \hbar/mc\alpha = 0.54\,\text{Å} \\ E &= e^2/2a_o, \quad \beta = \alpha \end{aligned} \tag{6.5}$$

The exact solution can be evaluated using results in the literature and the MATLAB mfun Laguerre, mfun('L',n,x). However, some important features can be found more simply. First the Schrödinger equation in three dimensions for a central force has as angular solutions the spherical harmonics, Y_ℓ^m, which will be shown later. These angular solutions are appropriate for all central forces and are the quantum analogue of the classical central force motion in a plane with conserved angular momentum.

There is an effective one-dimensional equation of motion as was the case in classical mechanics. A comparison to the Kepler discussion provides some insight. The equation at small r is dominated by the centrifugal potential which is effectively a repulsive inverse cube force. At large r, the potential V falls with r, so that the energy factor dominates.

$$\psi \sim (u/r)Y_\ell^m$$
$$d^2u/d^2 - [\ell(\ell+1)/r^2 + 2m(E-V)/\hbar^2]u = 0 \tag{6.6}$$

Putting that together, the behavior is a power law at small r defined by the angular momentum quantum number and a falling exponential (bound state with negative energy) at large r.

$$\psi \sim r^\ell e^{-r/a_0 n} \tag{6.7}$$

The atomic wave-functions are plotted using the script "qm_H_Atom" which shows energy levels and mean radii in Figure 6.4 and wave functions in Figure 6.5 for the three lowest energy bound states.

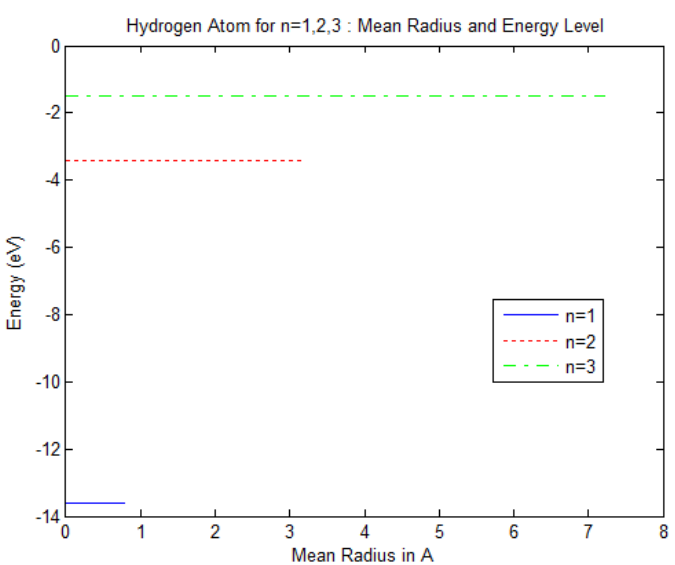


Figure 6.4: Mean radius and energy level for the three most deeply bound hydrogen states.

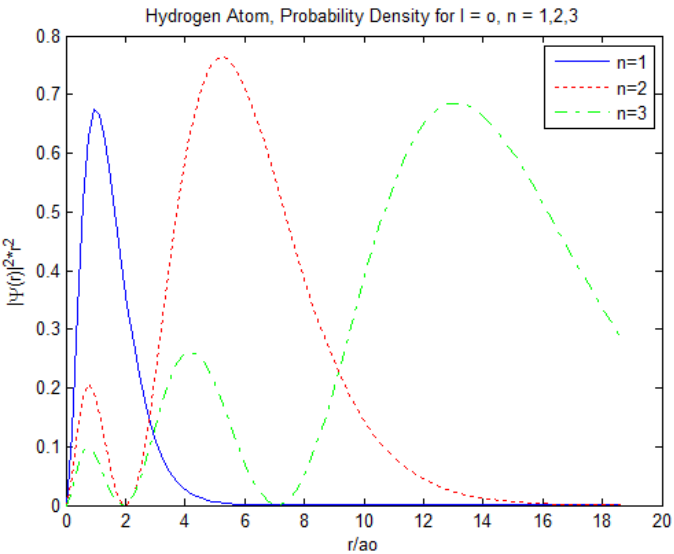


Figure 6.5: Probability density as a function of radius for the three lowest energy states in hydrogen with zero angular momentum.

The mean radius increases as n^2 reflecting that the higher n states are less deeply bound in the potential. Note that, in Figure 6.5, the number of maxima increases with n. Since the oscillation raises the momentum which is proportional to the derivative of the wave function, the higher n states are less deeply bound. Indeed for the zero angular momentum states, the number of oscillations is just the principle quantum number n.

6.4. Periodic Table — Ionization Potential and Atomic Radius

There are atomic data which give good insights into the underlying quantum structure of the elements. All of the complexity of the periodic table is contained in the simplicity of the hydrogen atom solution and the Fermi exclusion principle. It is a wonderful thing that chemistry can be so simply understood, at least in broad brush strokes.

Some of the data are presented by the script "qm_Atom_Periodic_Tab". Plots of the first ionization potential and atomic

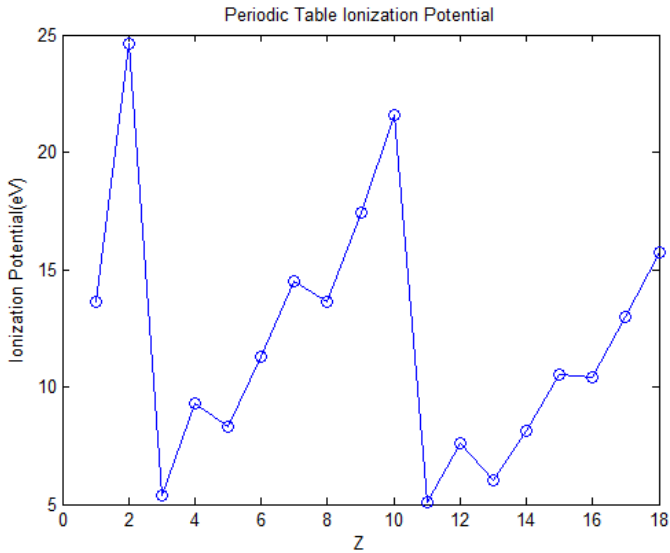


Figure 6.6: Ionization potential (first) as a function of Z.

radius are displayed. The ionization potential has a characteristic structure as a function of atomic number Z and is displayed in Figure 6.6. The mean atomic radius, measured using scattering data and are not numerically the same as the radius mentioned above, is shown in Figure 6.7.

For a hydrogen-like atom with atomic number Z and ignoring atomic screening of the charge Z by intervening electrons:

$$E = E_o Z^2/n^2$$
$$a = a_o/Z, \langle r \rangle = an^2 \tag{6.8}$$

The subscript refers to the hydrogen atom, as in Equation (6.5).

The behavior can be understood as due to the filling of atomic energy levels consistent with the Fermi exclusion principle and with the lowest energy states being filled first. For $l = 0$, only two spin paired electrons are possible. For $l = 1$, there are three possible angular momentum projections specified by $m = -1$, 0 and 1 which allows for up to six electrons, while for $l = 2$ there are ten electron states possible, using the five possible m values -2, -1, 0, 1, 2. As for the

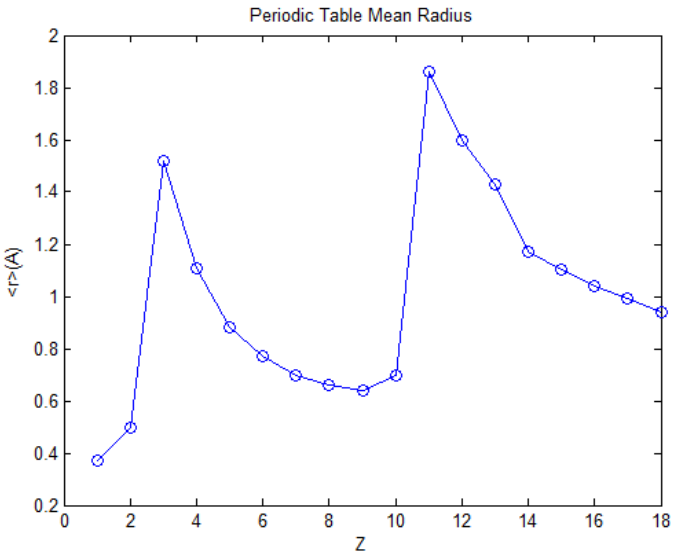


Figure 6.7: Atomic radius as a function of Z.

energetics, the energy should scale approximately as $(Z/n)^2$, while the radius should scale as n^2/Z. The expected state filling sequence is:

$$(1s)^2(2s)^2(2p)^6(3s)^2(3p)^6$$

For historical reasons the s and p notation refers to $l = 0$ and 1, respectively. The exponent refers to the number of electrons in that state and 1, 2 and 3 refer to the quantum number n.

The $1s$ "shell" closes with a noble gas, helium. The $2s$ shell starts with a loosely bound electron — a metal lithium. The $2p$ shell is almost full at fluorine, a base, and closes with neon, another noble gas. The $3s$ shell begins with a metal, sodium. The $3p$ shell is almost closed with chlorine, another strong base, and closes with argon. Clearly, much of the periodic table can be understood on the basis of the energy states of the hydrogen atom, although the details are computationally challenging.

It is important to observe that the properties of complex atoms can be calculated very accurately. Let us explore the lowest order corrections. Take the helium atom as the simplest case. The energy

of a single electron bound to a nucleus with Z protons is given in Equation (6.8). With two non-interacting electrons, the energy of the helium ground state would be eight times that for hydrogen, or $-109\,\text{eV}$. Experimentally it is $-79\,\text{eV}$. There is an energy shift due to the mutual repulsion of the electrons which can be estimated using hydrogenic wave functions and quantum perturbation theory to be $\Delta E \sim 5/8(Z\alpha/a_o)$ or $34\,\text{eV}$ for helium, changing the binding energy to $-74.8\,\text{eV}$ which is close to the observed value. The effect of the electrons screening of the nuclear charge has not been taken into account yet. A variational approach to the problem, letting the effective Z vary, leads to an estimate of the binding energy of $-77.4\,\text{eV}$ which is an improvement and an effective Z value of 1.69 is obtained which shows the screening effect.

Comparing to Figure 6.6 which plots the first ionization potential of 24.6 eV, the 79 eV binding energy should be adjusted for the energy after one electron is removed, $Z = 2$, and the once ionized binding energy is 54.4 eV. The ionization potential is then estimated to be 24.6 eV.

Lithium, assuming perfect screening by the two $1s$ electrons will have an ionization energy of about 3.4 eV in the $n = 2$ hydrogenic state. Lithium, see Figure 6.6, is observed to be more deeply bound, 5.4 eV, which means the effective Z is 1.26 rather than one due to screening by the $1s$ electrons. The size of hydrogen is roughly the Bohr radius, 0.54 angstroms. For lithium, with perfect screening, a radius of 1.55 angstroms is expected which is close to the point in Figure 6.7. Other atoms can be explored and with modern computational tools, models of any needed accuracy are available.

6.5. Simple Harmonic Oscillator

The simple harmonic oscillator is also a solvable problem. It is, in addition, a useful approximation to the general case of the lowest bound state of a quantum system, since the Taylor expansion about the minimum of an arbitrary potential yields an oscillator potential. The exact numerical solution is available using the MATLAB mfun('H',n,x) for evaluating the Hermite polynomials which are the

solutions. However, some general characteristics can again be found by looking at the Schrödinger equation at large displacements. The quadratic potential is:

$$F = -kx, \quad V = kx^2/2 = m\omega^2 x^2/2 \tag{6.9}$$

The potential term in the Schrödinger equation dominates at large x, leading to a Gaussian component of the wave function in one dimension. This behavior ensures that the state is bound by the potential.

$$\begin{aligned} 1/a &= \sqrt{m\omega/\hbar} \\ \psi &\sim e^{-x^2/2a^2} \end{aligned} \tag{6.10}$$

The exact wave functions for the lowest three states are shown in Figure 6.8. As in other cases, the increasing energy with the principle quantum number n is due to the increasing number of oscillations, or the increasing momentum.

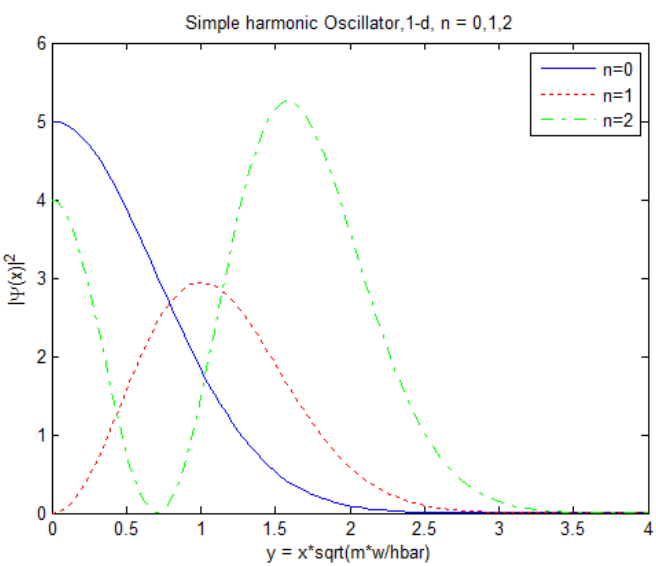


Figure 6.8: Square of the harmonic oscillator wave functions in one dimension for the three lowest energy bound states.

6.6. Other Force Laws

It is of interest to try to connect the observed spectroscopy of energy levels to the underlying force law. The hydrogen atom has a $1/r$ potential which leads to energies which go as $1/n^2$. The harmonic oscillator has a x^2 potential which leads to energy levels to go as n. Other cases are estimated by requiring that the de Broglie wavelength corresponding to a circular orbit of radius r contain an integral number of phase advances in order to set up a stable standing wave. For a power law potential, the energy is then minimized with respect to the radial variable, r, which yields an estimate of the quantized energy levels and the system size. This procedure is not rigorous and it only meant to be indicative.

The results for this calculation are displayed in the script "qm_forcelaw" and a user menu allows for a look at other power law potentials.

$$\begin{aligned} \lambda &= \hbar/p = r/n \\ V &= \alpha/r^b \\ r &= (n^2/m\alpha b)^\gamma, \gamma = 1/(2-b) \\ E &= n^{-2b\gamma} m^{b\gamma} \alpha^{2\gamma} \end{aligned} \tag{6.11}$$

The printout for the Coulomb force and the simple harmonic oscillator is shown in Figure 6.9. The examples of the hydrogen atom and the harmonic oscillator are printed out, as is the general solution. Note that this procedure is not exact, but is heuristic.

The result of this exercise for a Coulomb potential is $a \sim n^2/\alpha m, E \sim m\alpha^2/n^2$ which agrees with the exact solution. The user can try several other force laws in order to see how the energy levels change.

6.7. Deep Square Well

Continuing with the study of bound states, consider the case of a one dimensional well with very high potential sides, or a very deep well. The wave function must vanish at the well boundaries, located at $x = a$ and $-a$, which limits the wavelengths to quantized values.

```
Enter Force Law Coefficient b: 1
Coulomb - Bohr Atom
System Size

   2
  n
  ---
  a m
System Energy

   2
  a  m
  ----
    2
   n
Enter Force Law Coefficient b: -2
Harmonic Oscillator
System Size

  /     2   \1/4
  |    n    |
  | - ----- |
  \   2 a m /
System Energy

   1/2
  a    n
  ------
    1/2
```

Figure 6.9: Printout for the size and energy levels for the general case of an arbitrary force law for two known special cases.

This is formally the same requirement as was seen already in the "plucked string" exercise.

$$ka = n\pi/2$$
$$E = \hbar^2 k^2/2m = \hbar^2/2m(n\pi/2a)^2 \tag{6.12}$$

The energy levels go as n^2. The script "qm_infbox" has a dialogue with the user where the user chooses a well size and the script returns the ground state energy and a plot of the three lowest energy wavefunctions. The results for a well of size four angstroms full width is shown in Figure 6.10.

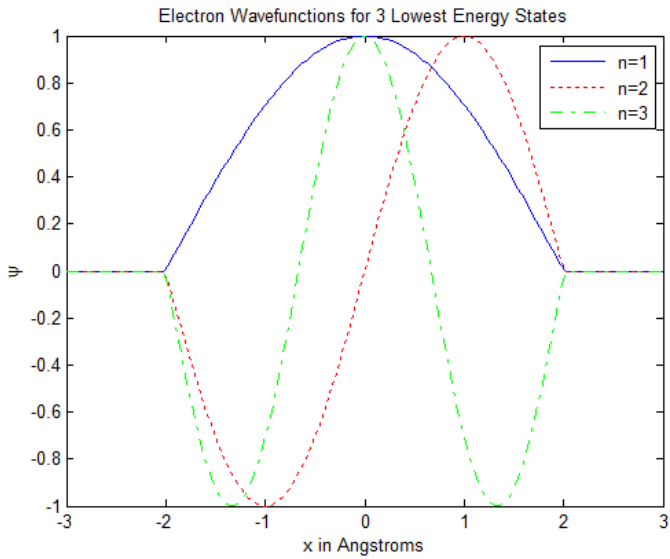


Figure 6.10: Wave functions for a deep potential well of size four angstroms full width. The number of oscillations inside the well increases with n, thus increasing the energy.

6.8. Shallow Square Well

The very deep well is a useful first approximation to the more practical problem of a well of finite depth. In that case, there are not an infinite number of bound states. Indeed, it may be for a localized and shallow well that there are no possible stable bound states.

The bound state solution has exponential behavior outside the well and oscillatory behavior inside. The wave function and its derivative are matched on the well boundaries and that requirement quantizes the energies. For $V = 0$ inside the well and $V = V_o > 0$ outside, the bound energy is, $0 < E < V_o$. The wave numbers are k and K outside and inside the well. The energies are those that satisfy the equations for matching at the well boundaries:

$$\begin{aligned} (\hbar k)^2 &= 2m(V_o - E), |x| > a \\ (\hbar K)^2 &= 2mE, \ |x| < a \\ \tan(Ka) &= k/K, \text{odd} \\ \cot(Ka) &= -k/K, \text{even} \end{aligned} \tag{6.13}$$

```
>> qm_inwell2
  qm_inwell - Particle in a potential well depth Vo, walls at +-a, MATLAB
  tools fminsearch


Particle in a Square Well
Enter Well Width in Angstroms, Walls at +- a : 2
Enter Well Depth in eV, V=0 Inside,V=Vo Outside : 5
Ground State Energy in Inf Box = 2.41429 eV
First Excited State Energy in Inf Box = 9.65715 eV
Ground State and n=1 States Energy = 1.13104 , = 4.02496 eV
Ground State Wave Function plotted from x =  -5 A to x = 5 A
```

Figure 6.11: Printout of the script "qm_inwell2" for the example of a well of full width 4 A and a depth of 5 eV.

The matching is distinct for odd (sine like) and even (cosine like) interior solutions. The problem is explored using the script "qm_inwell2". The script uses the infinite square well energies as a starting value and imposes matching boundary conditions on the interior oscillatory solutions and the external exponentially damped solutions to find the bound state energies using the MATLAB function "fminsearch". The printout for an example is shown in Figure 6.11. The user supplies a well size and a well depth. The script computes the energies of the first two lowest energies, if they exist, plots them as in Figure 6.12 and plots the wave function of the lowest energy state, as seen in Figure 6.13.

The lowest energy state is even while the first excited state is odd. It is instructive for the user to vary the depth and size of the well in order to see when a bound state becomes impossible due to the well being too narrow or too shallow. For example, with a 4-angstrom full width, a first excited state just becomes possible for a well depth >2.95 eV.

6.9. Wave Packets

A wave function with a precise momentum, $\psi = e^{\pm ikx}$, a plane wave, is totally un-localized since the modulus is the same at all spatial points. Conversely, a completely localized wave function contains all frequencies. That is a consequence of the fundamental uncertainty principle in quantum mechanics, $dkdx \sim 1, dEdt \sim \hbar$. A classical

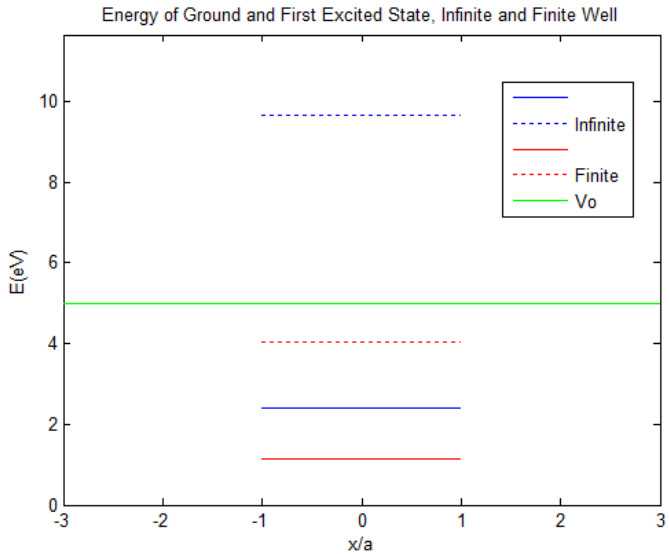


Figure 6.12: Energy levels for the lowest two bound states for an infinite well and the well of full width 4 angstroms and depth 5 eV.

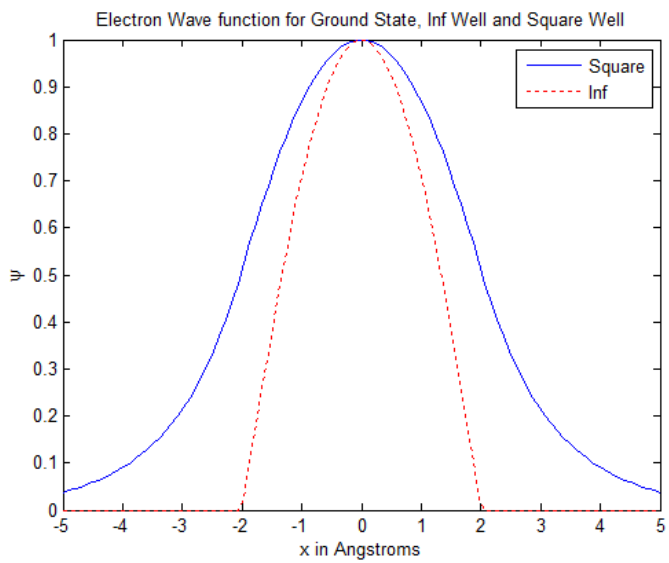


Figure 6.13: Wave functions for the lowest bound state for an infinite well and the well of full width 4 angstroms and depth 5 eV.

particle can be approximated by a superposition of waves localized in both position and momentum with localization consistent with quantum limitations.

A spatially localized wave packet, with a spread dx, which contains a spread of frequencies, $dk \sim 1/dx$, can be partially localized. This packet will spread spatially in time because of the uncertainty relationship. The characteristic time of spreading is $dt \sim \hbar/dE, dE = (\hbar c)^2(dk)^2/(2m_e c^2)$. Nevertheless, the wave packet is a useful approximation to the classical behavior of a particle. The characteristic time of packet spreading is set by $\hbar$ and is equal to 0.66, with energy in eV units of 10^{-15} sec. Light goes 3000 angstroms in this time, so a packet with v/c of 0.01 will go about 30 angstroms. This unit of time is adopted in what follows when the time development of a system is explored using MATLAB to make movies of systems.

The script "qm_WavePak" sets up a wave packet based on user input. An example of the user dialogue is shown in Figure 6.14.

The input parameters are a spatial spread, dx, and a wave packet central wave number value, k. Given that the momentum spread follows from the uncertainty relation the energy spread is also fixed. The energy spread sets the time for the spreading of the packet. The packet is:

$$\psi \sim e^{-[(x-<x>)/2dx]^2} e^{ikx} \tag{6.14}$$

The velocity sets the size of the wave number, $k = 250\ \beta(\text{Å}^{-1}), E(\text{eV}) = 3.9k^2$. These wave packets will then be used to

```
  Schroedinger Eq. and Uncertainty for Wave Packets


Enter Wave Packet Spatial Spread dx(0) (A): 1
tau = 0.170163, Expected Time for Packet Spreading in 10^-15 sec Units:
Enter Wave Packet Initial Mean Location <x> (A): 0
k = 250*beta (A^-1), E = 3.9 k^2 (eV)A^2)
Enter Wave Packet Initial Velocity w.r.t. c, beta = v/c: 0.01
k = 2.555, in 1/A units - wave number
```

Figure 6.14: User dialogue and printout for the script "qm_WavePak". A typical atomic state with size of about 1 angstrom and speed of about 1/100 that of light is used in this example.

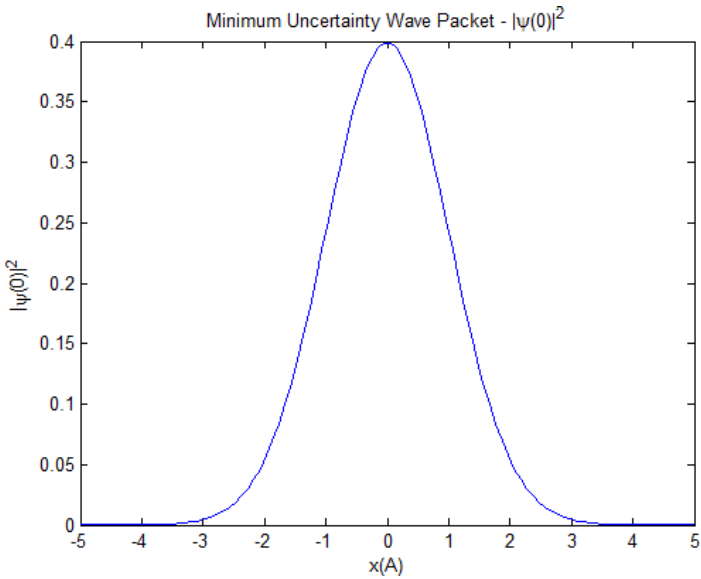


Figure 6.15: Wave packet probability density for the specific case $dx = 1\,\text{Å}$, $\langle x \rangle = 0$ and $v/c = 0.1$ for time, $t = 0$.

study the behavior of bound states or free particles ($V = 0$) and are used as the probes for numerical scattering and bound state exercises. An example packet appears in Figure 6.15 for time equal to zero.

6.10. Numerical Solution for Bound States

Bound states are probed with wave packet "particles" in the script "PDE_Sch_Well_SHO". A free particle can be studied as a special case of a well with zero potential. Generally, a user defines a width and depth. The harmonic oscillator potential can also be set up with the wave packet bound near the origin. The MATLAB partial differential equation function "pdepe" is used in analogy to the use of the package with the heat equation.

The user is supplied with a movie of the subsequent motion of the wave packet which the user has specified in the initial dialogue. The packet has an energy of 5 eV and $\langle x \rangle = 0$, starting at the center of a potential well. The user specifies a packet width and velocity.

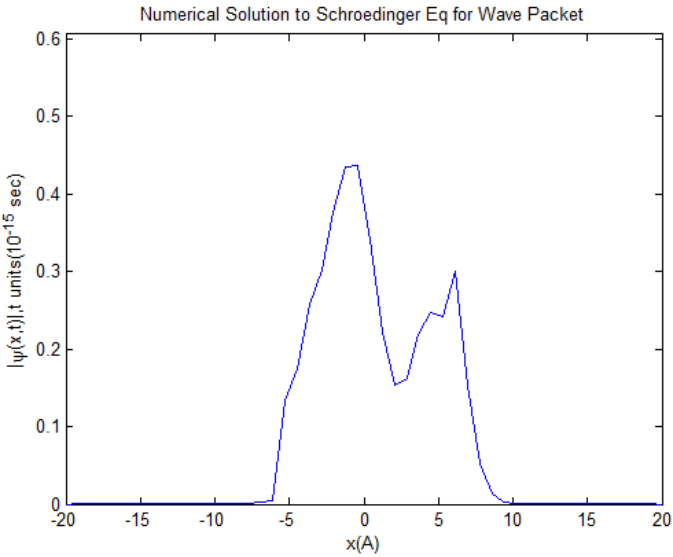


Figure 6.16: Last movie frame for a fairly deeply bound wave packet. The well extends for $|x| < 5$ Å and is 20 eV deep.

The well potential is zero inside $|x| < a$ and Vo outside. The time development over 10^{-15} sec. is supplied in a movie with 100 frames.

In the examples shown, $dx = 1$ angstrom and $v/c = 0.01$ is chosen. It is amusing to start with a choice of potential equal to zero in order to see the free particle spreading of the wave packet.

The last frame of the movie for a well of half width of 5 Å and a V_o of 20 eV is shown in Figure 6.16. Some of the wave function leaks out of the well because there is a spread of energies since the packet is localized.

The case of a harmonic potential is shown using an example shown in Figure 6.17. Again, the high energy components can leak out to fairly large $|x|$ values, but the bulk of the packet is localized near the minimum of the potential.

It is difficult to show on a flat page how the packets evolve in time. The user can and should look at free particle packet spreading, bound state square wells of different widths and depths, and also harmonic oscillator potentials of different strengths. These explorations should aid the user in building up an intuition about the bound states.

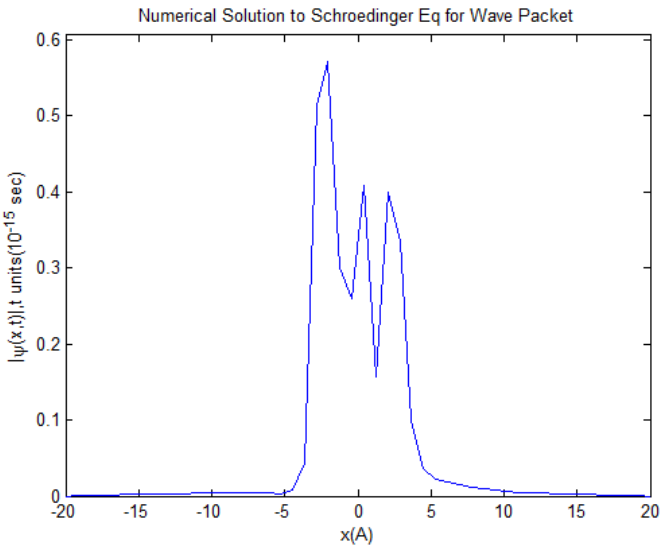


Figure 6.17: Last movie frame for a fairly deeply bound wave packet. The harmonic potential is set with a large k, $k = 2$, which contains the packet in locations near the $x = 0$ origin.

6.11. Scattering Off a Potential Step

Previously the focus has been on bound states, using both analytic and numerical methods. Now there is a shift to scattering states which are not localized in space. These states can be used to probe the forces which act on them. The simplest example is the change in a constant potential, V_o. The algebra is in exact analogy to the change in index of refraction at an interface in optics, at normal incidence. The index of refraction is defined by the potential, $n \sim \sqrt{E - V}$. If n becomes complex, then the analogy is to a metal in optics and the reflection is total. Comparing to Figure 4.17, the results are exactly the same with $n = \sqrt{1 - V_o/E}$. This illustrates the deep connections between wave optics and quantum mechanics. The solutions for incident energy, E, both for E less than the barrier height and above the barrier are:

$$\begin{aligned} k &= \sqrt{2m(E - V_o)}/\hbar \\ K &= \sqrt{2m(V_o - E)}/\hbar \\ 1 + r = t, t &= 2k/(k + K), t = 2k/(k + iK) \end{aligned} \tag{6.15}$$

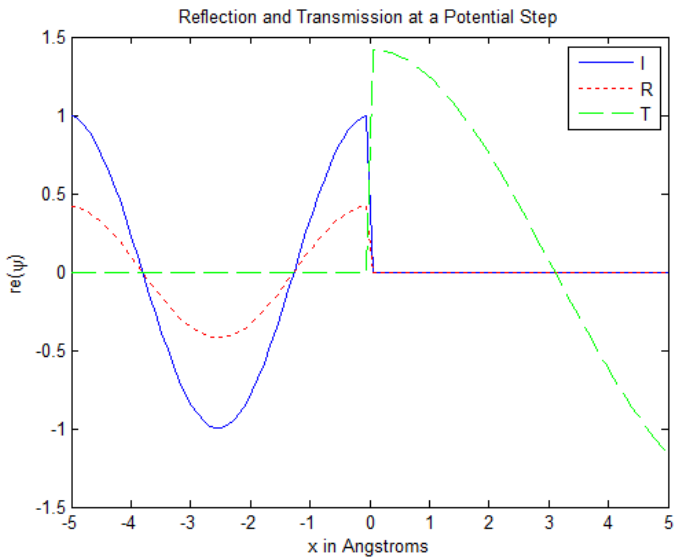


Figure 6.18: Incident, reflected and transmitted waves for a 6 eV electron incident on a step potential of 5 eV.

For E greater than V_o, solutions are oscillatory with wave number k. For E less than V_o, the solutions are damped exponentially with length parameter equal to $1/K$. Matching the wave functions and the derivatives (continuity of probability and momentum) at the boundary, $x = 0$, leads to the solutions for the reflected wave function amplitude r and the transmitted wave amplitude t at the boundary.

The result for a 6 eV electron incident on a 5 eV potential is given in Figure 6.18.

The matching of the solutions at $x = 0$ is clear in Figure 6.18. However, these are complex functions so that the reflection coefficient $R = |r|^2$ is less than one as is the transmission coefficient T.

The situation for a step of 4 eV is shown in Figure 6.19. In this case, the solution for $x > 0$ is exponentially decreasing. This plot illustrates the matching at the boundary. However, it is very misleading. The reflection coefficient is, in fact, equal to one and, although an incident wave would penetrate into the region $x > 0$ for a short period of time, it is ultimately totally reflected. This fact is reflected in Figure 6.20 which displays the R and T values for different energy

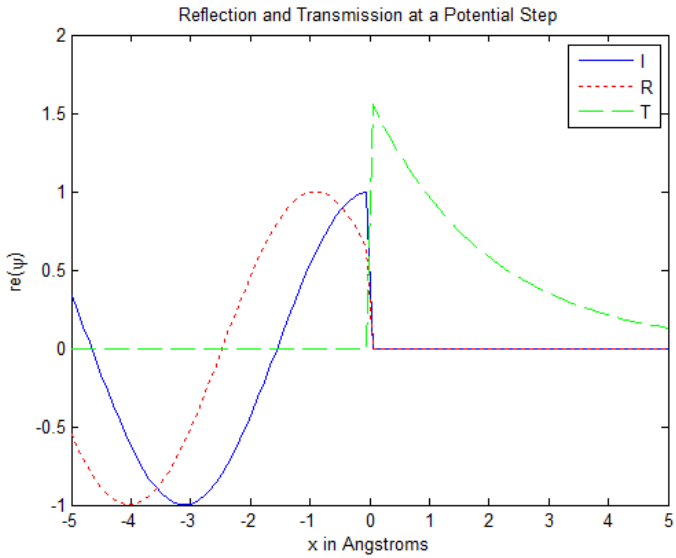


Figure 6.19: Incident, reflected and transmitted waves on a step potential of 5 eV by a 4 eV electron.

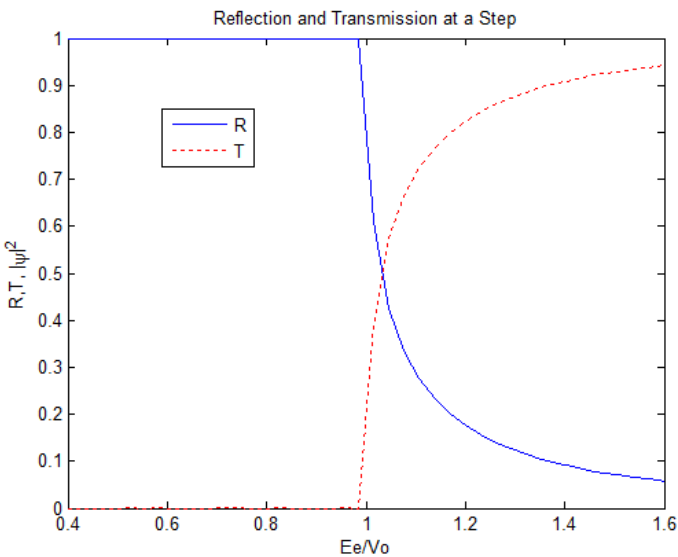


Figure 6.20: Reflection and transmission coefficients for a wave of energy E incident on a step of potential V_o.

waves incident on the step. Note that R is equal to one for E less than V_o.

The script is found in "qm_step". The user has a choice of the energy of the particle incident from $x < 0$. The potential is fixed at 5 eV. Classically, there is perfect transmission for $E > V_o$, but, due to the quantum wave behavior, there is a region of E greater than V_o but near to it where the reflection coefficient is not zero. This behavior is familiar from wave optics.

6.12. Scattering Off a Potential Well or Barrier

The next level of complexity is to scatter off a well with a finite width. This situation is considered in the script "qm_tunn". The most interesting case is probably the one of "tunneling" where classically the barrier is too high to be penetrated but in quantum mechanics the exponentially falling solutions can, with some probability, penetrate the barrier.

The wave is incident from the left. There is a reflected wave, in general. Inside the well/barrier there are waves with wave number K [Equation (6.15)]. Exiting the potential region on the right, there is a transmitted wave. The solutions for these wave amplitudes follow from matching the wave function and its derivative at the two boundaries, in an extension of the technique used for the step potential.

There are two limiting cases of interest. In the case of a barrier with height above the wave function energy, the transmission coefficient depends exponentially on the width of the barrier and the wave number in the barrier region. The most important factor is the exponential decrease of T with tunneling distance a, although other power law behavior is present:

$$T \sim e^{-2Ka} \tag{6.16}$$

This phenomenon is known as "tunneling through" a barrier. It is crucial, for example, in the Coulomb barriers that retard fusion because of the repulsive Coulomb forces between positively charged nuclei. The Sun works only because it operates at an enormous temperature which gives sufficient thermal kinetic energy to the nuclei

to overcome these barriers. On Earth, the reactions which are being attempted for fusion reactors or already achieved in fusion weapons, are the fusion of deuterium and tritium.

$$\begin{aligned} \mathrm{H}_1^2 + \mathrm{He}_1^3 &\rightarrow \mathrm{He}_1^4 + n(17.6\,\mathrm{MeV}) \\ \mathrm{H}_1^2 + \mathrm{H}_1^2 &\rightarrow \mathrm{H}_1^3 + \mathrm{p}(4\,\mathrm{MeV}) \end{aligned} \tag{6.17}$$

There is another interesting limit which occurs for potential wells rather than barrier penetration. It is called the Ramsauer effect, when there is perfect transmission for an incoming wave of fixed wave number k. It occurs because of phase changes for reflected waves which make it possible for the reflected waves to be cancelled out. The Ramsauer relationship for incoming energy E, well potential V_o and full well width a corresponds to an energy for a bound state in an infinite well:

$$\begin{aligned} Ka &= n\pi \\ E + V_o &= \hbar^2(n\pi/a)^2/2m \end{aligned} \tag{6.18}$$

The Ramsauer total transmission is related by direct analogy to anti-reflection thin film coatings in optics. There is a phase change upon reflection. Therefore, there is a destructive interference between the waves reflected at the two interfaces.

The printout for a specific example is shown in Figure 6.21. In this case the well is a thin barrier, 1 eV higher than the electron energy and 1 Å wide.

```
>> qm_tunn
  qm_tunn - Particle Scatters off a Potential Well of depth Vo, walls at (0,a)


Electron of Energy 5 eV, Scatters off a Potential Well
Ramsauer if E =  -Vo +((hbar*k))^2/2*m, ka = n*pi
Enter Well Full Width, a, in Angstroms : 1
Enter Well Depth Vo in eV, if <0 it is Barrier Height -6
k = 1.13027, K = 0.50547, r = -0.387294, t = 0.60559, p = -0.280894, q = 2.02099, for Ee = 5 eV
Wavefunctions Squared, R = 0.333523, T = 0.666477
Wave Functions plotted from x =  -1(A) to x = 2 (A)
 R,T for Vo in (-10 10) eV and a in (0.5 5.5) A
```

Figure 6.21: Printout for the case of a thin barrier, of width 1 Å and for a barrier 1 eV higher than the incident wave energy. The p and q are the amplitudes inside the well with positive and negative x behavior $\pm Kx$.

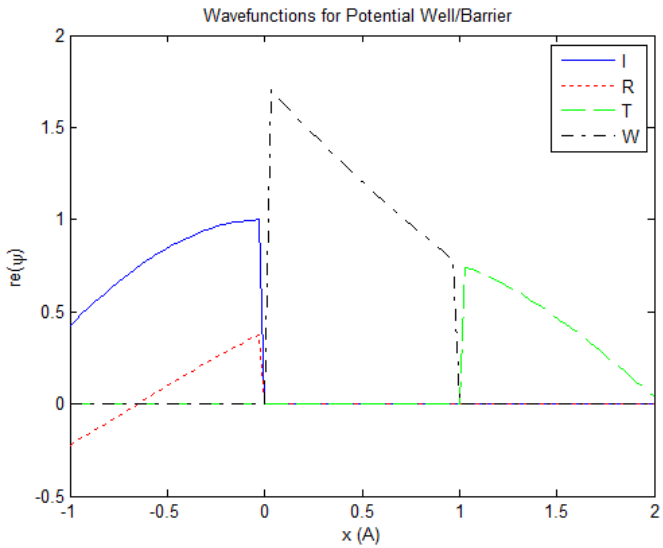


Figure 6.22: Electron of 5 eV incident on a barrier of height 6 eV extending over a width of 1 Å. The solution within the barrier, W, is exponentially falling with x.

The solutions which match the incident, reflected and transmitted waves at the two interfaces are shown in Figure 6.22. Indeed, the transmitted wave inside the barrier is exponentially falling with increasing x. The surviving wave for large x is again oscillatory.

A map of the transmission coefficient for a range of constant potentials covering both wells and barriers and with different widths for the potential is shown in Figure 6.23. There is a steep fall in the transmission coefficient, T, for barriers above the 5 eV energy of the incident wave. This falloff is steeper as the width of the potential increases, as expected.

For potential wells, the Ramsauer points of perfect transmission occur at different well widths and well depths. These points are clearly seen in the figure. This is another connection to optics since this condition is exactly that which applies for a Fabray–Perot interferometer.

6.13. Wave Packet Scattering on a Well or Barrier

The wave packets which were already introduced can, in turn, be scattered off potential wells or barriers. In this case the packet is

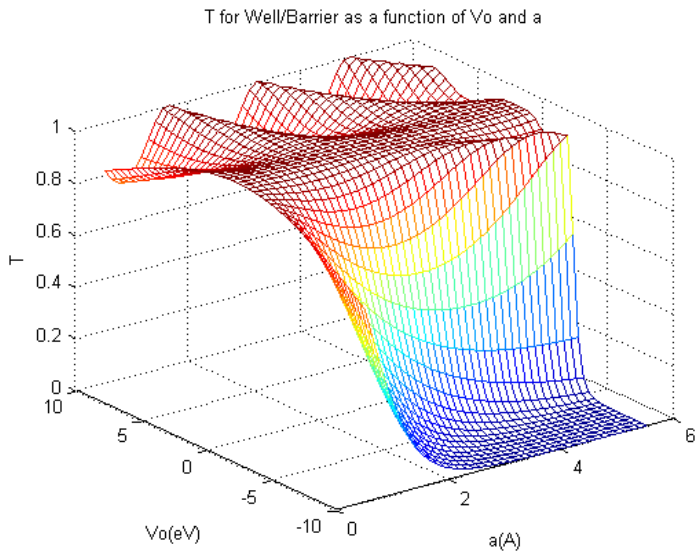


Figure 6.23: Electron of 5 eV incident on a barrier, $V_o < 0$, or well, $V_o > 0$, extending over a width of a (Å). The Ramsauer solutions with $T = 1$ are evident.

not started at $\langle x \rangle = 0$, bound in a potential, but at $\langle x \rangle = -10$ Å. As before, a movie of the evolution of the wave as it encounters the potential is provided. There are several distinct possibilities available to the user. The step can be checked by making the region of the potential very wide. For a barrier, the mean packet energy can be above or below the barrier height. For a well, the Ramsauer effect can be explored, although the wave number spread of the packet makes the effect somewhat diluted.

A specific example is shown using the script, "PDE_Sch". As before, the MATLAB tool "pdepe" is used to numerically solve the Schrödinger equation with the initial conditions being the defined wave packet. The user selects a wave packet, in this example, $\langle x \rangle = -10$ Å, $dx = 1$ Å, $v/c = 0.01$. The scattering is off a barrier of height 20 eV extending over 5 Å for a 5 eV incident electron. A movie is provided, one frame of which is shown in Figure 6.24. It is difficult to display the details available in a movie or to see how varying the parameters of the problem help to build up insight. Indeed, that is why the exercises are open ended.

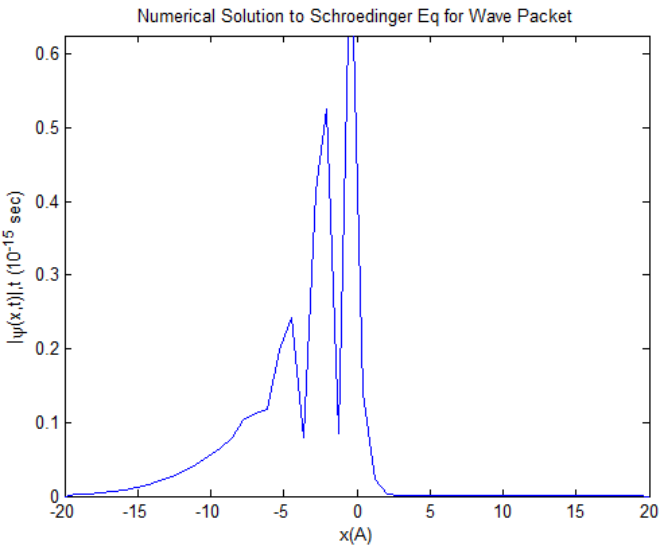


Figure 6.24: Wave packet scattering off a barrier of height 20 eV and width 5 Å. The wave is largely reflected but a portion is extending into the well and will tunnel through.

In the frame shown, the wave has begun to encounter the potential, is largely reflected, but has a portion which is propagating through the well and which will ultimately tunnel through to large x values.

The solutions for a potential well are illustrated by one specific example of a packet scattering off a well of 20 eV depth and 2-angstrom width. One frame is shown in Figure 6.25, when the wave has encountered the potential and the oscillatory behavior inside the deep well becomes evident.

The user should "play" with this script in order to see the effect of changing all the parameters. The time development of the solutions is instructive.

6.14. Born Approximation — Scattering and Force Laws

Continuing the study of scattering states, the Born approximation can be used to compute the scattering amplitude in the

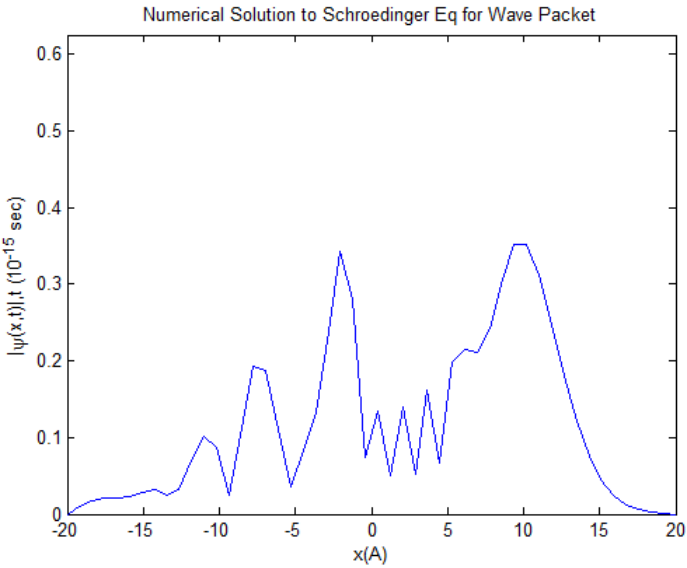


Figure 6.25: Wave packet scattering off a well of depth 20 eV and width 2 Å. The packet has oscillatory behavior evident in the potential well region.

approximation that the initial state is a plane wave as is the final state. In that case, the scattering amplitude is essentially the Fourier transform of the scattering potential, $V(r)$, into the momentum transfer variable q.

$$A_{Born} \sim \int_0^\infty V(r)[\sin(qr)/qr]r^2 dr \tag{6.19}$$

The Born amplitude is set up by the script "qm_BornScatt3". Use is made of the symbolic integration tool in MATLAB "int". Examples are given in the printout; a square well amplitude goes as the inverse third power of q, while a Coulomb potential goes as the inverse square, and a screened Coulomb goes as the inverse of the sum of the squares of q and the inverse of the screening cutoff length a. The square well amplitude is plotted in Figure 6.26. The amplitude falls rapidly for q greater than a, the size of the well. That falloff is true in general for an object of characteristic size a.

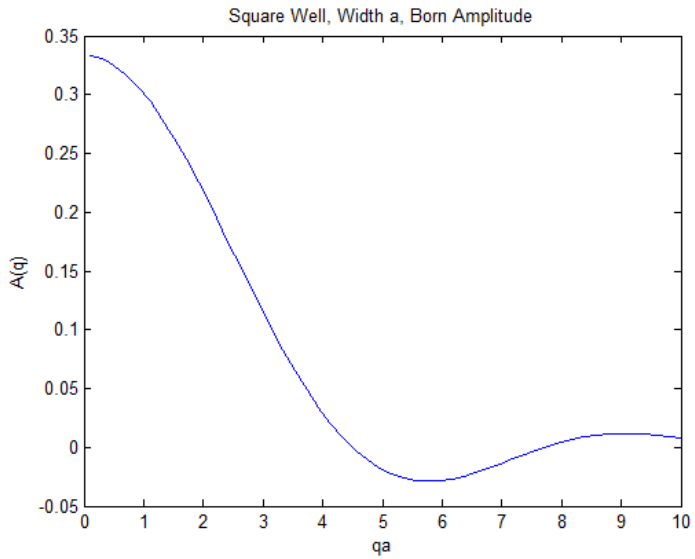


Figure 6.26: Born amplitude for scattering from a square well of depth a in qa units.

The momentum transfer, q, is the vector difference in wave number between the incoming and outgoing state. It assumes elastic scattering and is determined by the incoming momentum, P, and the scattering angle θ.

$$\begin{aligned} q^2 &= 2(p/\hbar)^2(1-\cos\theta) \\ &= (P/\hbar)^2 \sin^2(\theta/2) \end{aligned} \tag{6.20}$$

In the case of a Coulomb potential, the amplitude $A(q)$ goes as the inverse square of the momentum transfer. If there is screening of the nuclear charge by the atomic electrons, there is a cutoff distance, a, which is on the scale of the size of the atom, which is about 100,000 times larger than the size of the nucleus. Ignoring the small screening effect, the Rutherford cross section is obtained.

$$\begin{aligned} V(r) &\sim e^{-r/a}/r \\ A(q) &\sim 1/(q^2+1/a^2) \\ d\sigma/d\Omega &\sim |A(q)|^2 \to 1/\sin^4(\theta/2) \end{aligned} \tag{6.21}$$

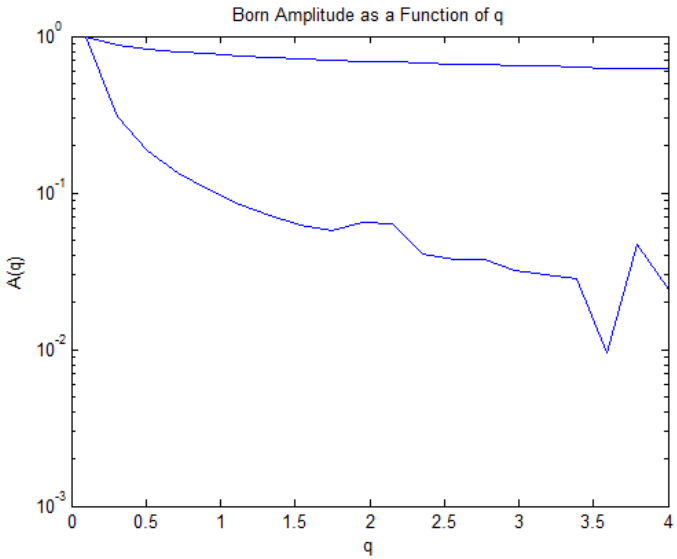


Figure 6.27: Born amplitude for power law potentials — inverse square and inverse cube. The inverse-cubic potential gives the amplitude with the largest value at high q values.

The user is given a choice of several potentials. The Born amplitude is then evaluated numerically using the MATLAB tool for numerical integration, "quad". The user should try different scattering force laws. The results for potentials going as two power laws are shown in Figure 6.27 — for inverse square and inverse cube potentials. The more singular power laws, such as inverse cubic, have an amplitude that extends to higher q values. Generally more singular potentials support higher momentum transfers, allowing one to differentiate between different force laws by studying the scattering angular distributions.

6.15. Spherical Harmonics — 3D

The Schrödinger equation in three dimensions for central forces has a common set of angular solutions. This feature has as its analogue the fact that central forces have a conserved angular momentum in classical mechanics. The solutions are characterized by a quantum number specifying the angular momentum, ℓ, and a quantum number

specifying the projection of the angular momentum onto the z axis, m. A few plots of the low ℓ values of these functions are provided by the script "qm_Ylm2" and are displayed in Figure 6.28. MATLAB does not provide a symbolic function to evaluate the functions. There are $\ell+1$ maxima seen in Figure 6.28 for the $m=0$ state. For $\ell=m$, there is only a single maximum. Surfaces of the spherical harmonics for $\ell=2$ and $m=0$ and 2 are shown in Figure 6.29. The script supplies plots of all the harmonics for $\ell<3$.

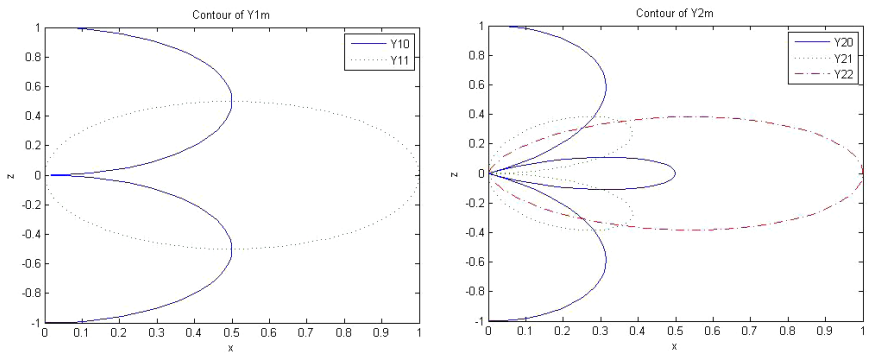


Figure 6.28: Contours of the spherical harmonics for $\ell=1$ states (left) and $\ell=2$ states (right) in the (x, z) plane.

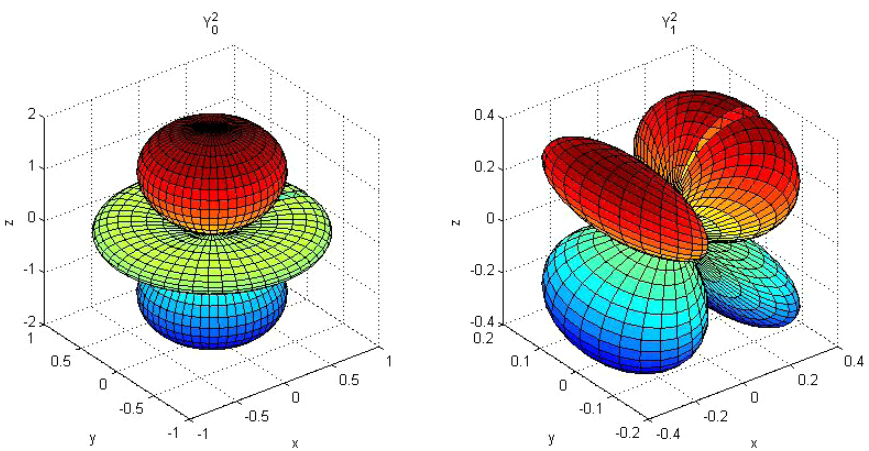


Figure 6.29: Surfaces for the spherical harmonics for Y_2^0 (left) and Y_2^1 (right).

6.16. Free Particle in 3D

A free particle in three dimensions is the simplest extension of the plane wave scattering state in one dimension. In the case of a spherically symmetric situation, the best approach is to exploit the spherical symmetry and use the angular solution, Y_ℓ^m. The radial Schrödinger equation is then:

$$d^2u/d^2r - u\ell(\ell+1)/r^2 + k^2u = 0 \tag{6.22}$$

The terms in the radial equation are the radial kinetic energy, the effective repulsive inverse square centrifugal potential proportional to the square of the angular momentum, and the energy of the state, set by the wave number k. The connections to the classical Kepler problem should be noted.

The solutions are known and are displayed in the script "qm_Schro_3d_J". The solutions can be found symbolically in MATLAB using the function "dsolve" for Equation (6.22). The solutions are Bessel functions, which are symbolically available using the MATLAB function "bessel$j$(n,x)"

$$j_\ell(z) = \sqrt{\pi/2z}J_{\ell+1/2}(z) \tag{6.23}$$

The wave functions are then found as $u(k)$ and the full solution is the product of the spherical harmonics appropriate to a central, or no, force and the Bessel functions for the free particle case. Generally J is appropriate for cylindrical geometries, while j is appropriate for spherical geometries.

$$\begin{aligned} u(r) &= \sqrt{r}J_{(\ell+1/2)}(kr) \\ k &= \sqrt{2mE}/\hbar c \\ \psi &= [u(r)/r]Y_l^m \end{aligned} \tag{6.24}$$

The results of the symbolic printout are shown in Figure 6.30. Explicit solutions are provided for the $\ell = 0$, 1 cases.

These solutions are plotted in Figure 6.31. It is seen that the higher ℓ solutions are pushed away from the origin by the centrifugal potential as expected. The Bessel functions are functions

```
>> qm_Schro_3d_J
  3-d Schroedinger Eq. for free particle, u


|
      1/2                                    1/2
  C2 r    besselj(l + 1/2, q r) + C3 r    bessely(l + 1/2, q r)
J for l = 0, 1

   1/2
  2    sin(q r)
  --------------
    1/2       1/2
  pi    (q r)

     1/2 /             sin(q r) \
    2    | cos(q r) - -------- |
         \               q r    /
  - -----------------------------
              1/2       1/2
           pi    (q r)
```

Figure 6.30: Symbolic printout for the solutions of the free particle Schrödinger equation in three dimensions.

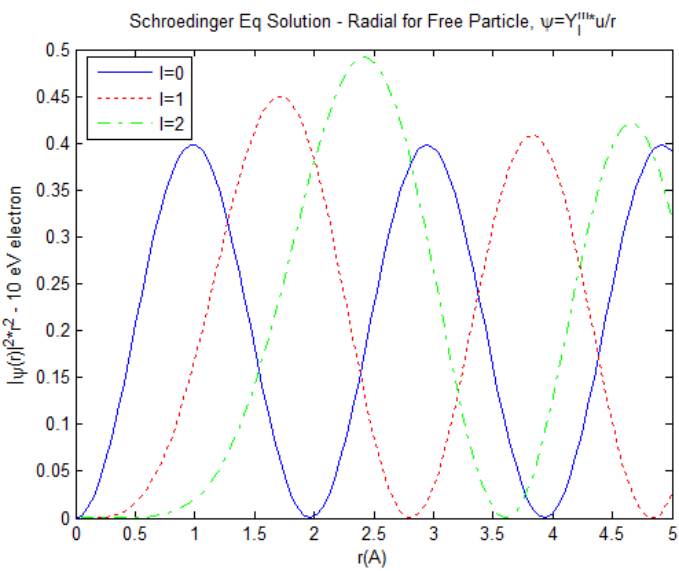


Figure 6.31: Probability density for the lowest three angular momentum values for a free particle moving in three dimensions, with $m = 0$.

of cos and sin and powers of r. Their explicit forms can be found symbolically in MATLAB by declaring x symbolic, syms x, and then typing, for example, besselj (1/2, x) in the Command Window. The result is (2^(1/2)*sin(x))/(pi^(1/2)*x^(1/2)), which appears in Figure 6.30.

6.17. Radioactive Decay — Fitting

With the discovery of radioactive decays, it became clear that elements were unstable and could transmute. Soon after nuclear transmutation was established experimentally using neutron bombardment and by other means.

It is a prediction of quantum mechanics that such decays follow an exponential behavior in time with a lifetime which is characteristic of the particular dynamics.

The script "Radioactive_Decay" uses the MATLAB function "rand" to simulate decays by creating a data model of decay times with a fixed lifetime. Random numbers are a key element in making detailed models of processes, using what is called the "Monte Carlo" method. A simple script to generate decay times t with unit lifetime and plot them is shown below in Equation (6.25). The user can enter this script easily using the Command Window. Other functional dependencies are possible and, if interested, the user can try to generate a few.

```
>> for i = 1 : 1000
t(i) = -log(rand);
end
>> [n, t] = hist(t, 50);
>> semilogy(t, n, 'o')
```
(6.25)

The user is first asked to pick a lifetime for the sample. A histogram of the results of choosing a lifetime of ten years is shown in Figure 6.32. The exponential, within the statistics of the sample of one thousand decays, is a straight line on a semilog plot. The errors shown in Figure 6.32 are the square root of the number of

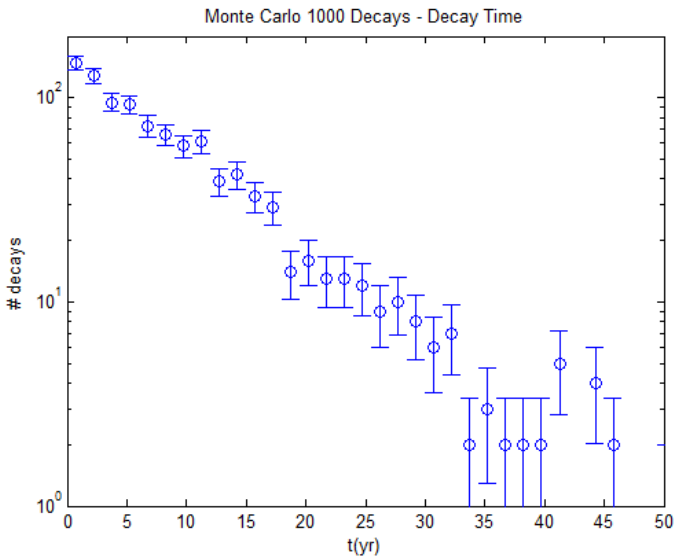


Figure 6.32: Histogram of 1000 simulated decay times for a sample with a lifetime of 10 years.

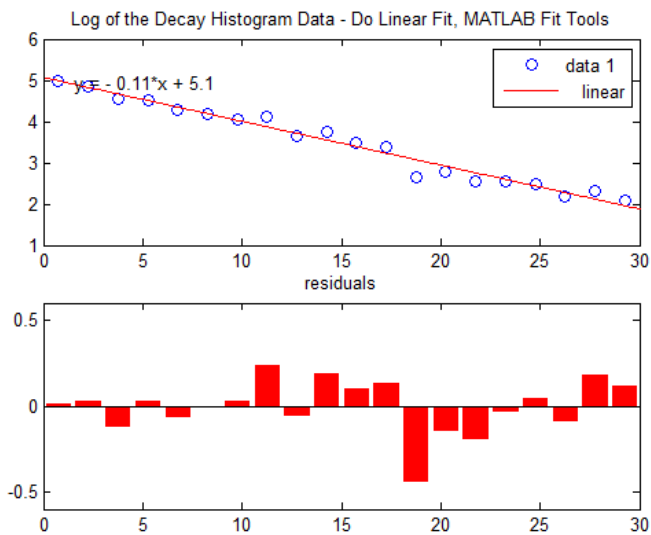


Figure 6.33: Results of using the MATLAB fitting package. Top figure is the histogram data plotted as the log of the bin contents. The fit is to a straight line and the slope is 0.11.

events in the respective histogram bin. The MATLAB simple fitting tool is used again to fit a straight line to the log plot of the simulated decay histogram. The result, shown in Figure 6.33, is 0.11/yr or a lifetime of ∼9 years. If more events were used, or if the statistical errors were more properly used to weight the fit, see Section 6.1, the result would be a bit closer to the expected value of ten years.

Chapter 7

Special and General Relativity

"Time is an illusion."

— **Albert Einstein**

"In some sense, gravity does not exist; what moves the planets and the stars is the distortion of space and time."

— **Michio Kaku**

Relativity covers extreme situations, far removed from 'normal" life. Either the speeds involved are near that of light, special theory, or the gravitational fields are very strong, general theory. In such cases, a comparison to prior classical demonstrations, such as rocket motion or Keplerian orbits can be made which brings out the salient differences.

However, this is not to say that relativity is not important in our everyday lives. To pick one example, our annual background dose of radiation is largely due to cosmic ray muons at the Earth's surface. The dose is due to muons depositing ionization energy. The muons are produced high in the atmosphere. At rest they have a lifetime of $2.2\,\mu s$. If they had a velocity of c, classically the muons would only go a distance of 660 m before decaying into electrons. In fact, we are bombarded by muons and that is so because of relativistic time dilation.

7.1. Time Dilation

The basic postulate of the special theory of relativity is that light has the same speed, c, in all reference frames moving with uniform velocity with respect to one another. This then, immediately leads to time dilation, where a time interval T in a frame where a clock moves with velocity $v/c = \beta$ is larger than the time measured on a

clock at rest, the proper time T_o. The proper time is measured on a single clock, while the time in the frame where the clock moves is measured by different observers located at different spatial points, who use a synchronized array of clocks.

$$\begin{aligned} T &= T_o\gamma \\ \gamma &= 1/\sqrt{1-\beta^2} \\ \beta &= \nu/c \end{aligned} \tag{7.1}$$

A simple "proof" of time dilation is provided in the script "SR_Time_Dilate" which uses a clock consisting of a light flasher and a mirror. In the rest frame of the clock, the time T_o is $2L/c$ where L is the distance between the clock and the mirror. In the frame where the clock moves, the light must travel a longer distance, but still at the speed c, so that the time T in that frame is larger than the proper time, $T^2 = (2L/c)^2 + (\beta T)^2$.

The script makes two movies for the user. In the first, the light goes to the mirror and back in the clock rest frame. In the second, the clock moves through the frame and the time is counted in clock ticks. The user chooses the velocity in that frame. The last frame for a specific case is shown in Figure 7.1. The distance travelled by the light flash is indicated in the figure, which then gives the time dilation factor. The user should try several different velocities. In the specific example, the v/c value is 0.9 and there are 20 clock "ticks" in the rest frame and 45 in the moving frame. The "observed" time dilation factor is 2.25, due to finite tick size, while the calculated value is, Equation (7.1), 2.29.

7.2. Relativistic Travel

The time dilation effect can have large consequences. Consider the case of a vehicle which has a uniform proper acceleration, in this case g so as to make the travelers comfortable. The velocity change seen from an observer at rest, dv, can be related to the velocity change in the frame of the vehicle, dv^*, $dv/[1-(v/c)^2] = dv^*$ using the Lorentz transformation equations for time and position.

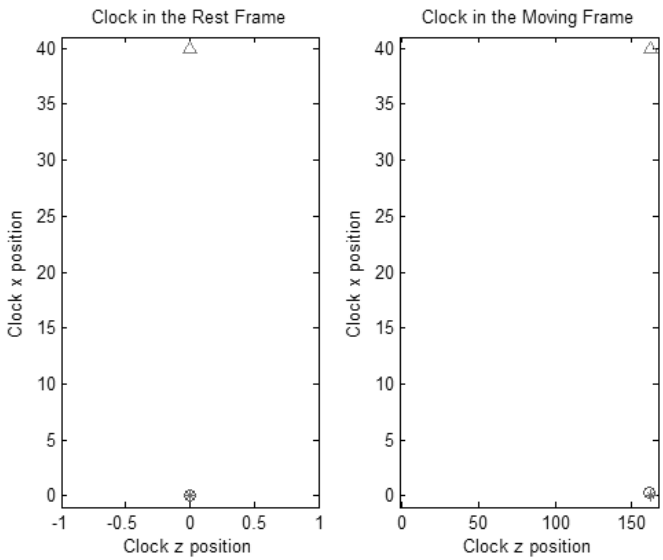


Figure 7.1: Thought experiment for time dilation. On the left is the situation in the rest frame on the last movie frame. On the right the situation in the frame where the clock moves is shown.

The algebra of such a trip is displayed in the script "sr_rel_rock". The user picks a travel time, t^*, for the passengers. Formally, the mathematics is the same as the trajectory of a charged particle in a uniform electric field. There is a constant proper acceleration in the rocket rest frame of g. The velocity and distance traveled and the time elapsed for the stay at home observers are:

$$\begin{aligned} \alpha &= g/c \\ \beta &= \tanh(\alpha t^*) \\ \gamma &= \cosh(\alpha t^*) \\ t &= \sinh(\alpha t^*)/\alpha \\ z &= c[\cosh(\alpha t^*) - 1]/\alpha \end{aligned} \tag{7.2}$$

The non-relativistic (NR) limit is that $v \to gt^*, t \to t^*$ and $z \to gt^2/2$ so the classical result is recovered. Printout from the script is shown in Figure 7.2. For a twenty-year voyage, as experienced by the passengers, 439 million years elapsed at home and the passengers will

```
>> sr_rel_rock
  rocket - Compute motion of a  relativistic rocket, constant acceleration
  = g


Proper Acceleration is g
Enter Maximum Ship Time in Years : 20
Final Speed w.r.t. c = 1
Final gamma factor = 4.52997e+08
Total Time Elapsed at Home (yr) = 4.3928e+08
Total Distance Travelled in LY (Home) = 4.3928e+08
Payload Ratio for Relativistic Rocket = 9.05995e+08 - Light Exhaust
```

Figure 7.2: Printout for a rocket under constant proper acceleration.

travel 439 million light years. Of course, it has not been specified how the acceleration is achieved nor how the passengers are to be shielded against the high energy bombardment by interstellar dust. Still, it is a wonderful fact that interstellar travel is possible in a short time for the passengers.

The trip time for the at home observers as a function of the time for the passengers is shown in Figure 7.3. As the velocity builds up,

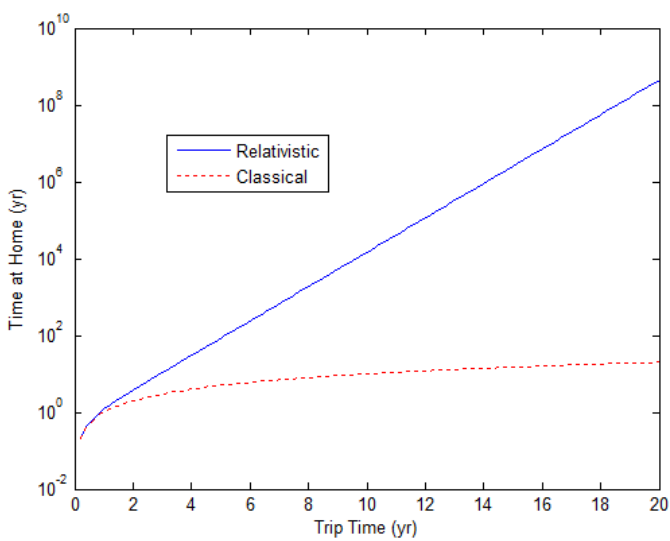


Figure 7.3: Trip time for the passengers vs. time elapsed for the stay at home observers. Note the semilogy scale. Classically time is absolute and the same independent of speed and trip time equals home time.

the time dilation factor continues to increase, leading to the enormous time difference between the two sets of people. Yes, interstellar travel is possible but truly "you can't go home again".

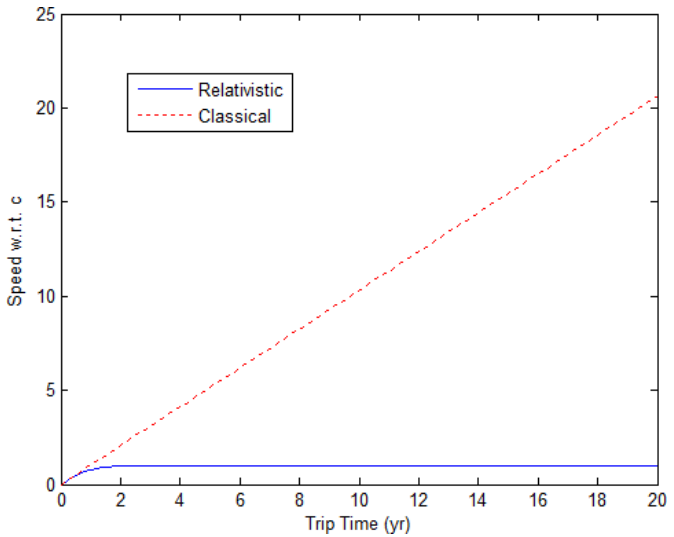


Figure 7.4: Trip time for the passengers vs the speed of the rocket. The classical speed is not limited by light speed.

In Figure 7.4, the trip distance is shown for a classical twenty year trip and for a relativistic trip. There is no classical "speed limit", and $v > c$ is not forbidden. Even though the ship can have a speed greater than c classically, the Lorentz length contraction is a much larger relativistic effect, allowing the travelers to cover immense distances at speeds close to c.

7.3. The Relativistic Rocket

This stellar travel sounds great but how is it accomplished? Previously the classical rocket was demonstrated as well as "solar sailing" using light pressure as possible methods. First, let us look at the relativistic rocket. It can be thought of as an object of mass, m, which "decays" into a mass $(m - dm)$ and an exhaust object moving with velocity $\nu_o/c = \beta_o$ with respect to the rocket. The modified rocket

equation, compare to Equation (2.14) and Equation (2.15) in the classical rocket case, and the first integral of the equation is:

$$md\beta/dm = \beta_o/\gamma^2$$
$$\beta = (1 - m/m_o)^{2\beta_o}/(1 + m/m_o)^{2\beta_o} \tag{7.3}$$

The differential equation approaches the NR equation in the limit of low velocity, $d\beta/dm \rightarrow \beta_o/m$.

Clearly, light is the best propellant. There is a closed form solution which is possible for the velocity but not for the position. The velocity is initially zero and it builds up to a value defined by the exhaust velocity and the payload ratio. With no payload, the velocity becomes c. Printout of the script "sr_rocket3" is given in Figure 7.5. The rocket equation is first solved symbolically using "dsolve" for the velocity and then integrated numerically using "quad" to obtain the position specific to the user input for the payload ratio and exhaust velocity. The user also supplies the mass burn rate so that the mass variable can be translated into time. The script returns the velocity at the end of the "burn".

```
>> sr_rocket3
  rocket - Compute motion of a  relativistic rocket, variable accel


 Relativistic Rocket Velocity, m*db/dm = bo*(1-b^2)

             2 bo
         2 mo
  1 - --------------
       2 bo     2 bo
      m     + mo
Enter Payload Mass Ratio Final Mass/Initial Mass < 1: 10^-6
Enter Exhaust Velocity w.r.t. Light in Rocket Frame: 0.8
Velocity Measured in Frame Where Rocket Was Initially at Rest
Final beta and gamma at the End of the Burn = 1 , 31547.9
Enter Mass Burn Rate (sec-1) as a Fraction of Initial Mass: 10^-5
```

Figure 7.5: Printout for the script describing the relativistic rocket.

As seen from Figure 7.5 with a reasonable payload ratio and a fancifully optimistic exhaust velocity, a respectable time dilation

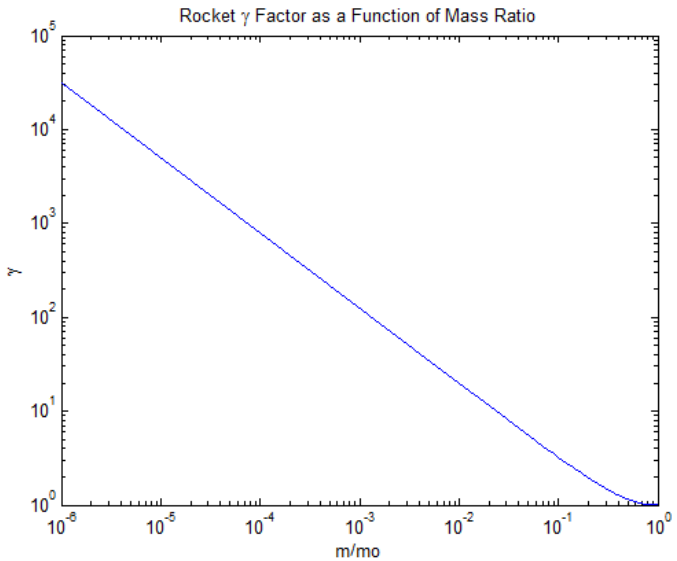


Figure 7.6: Time dilation factor of a rocket as defined in Figure 7.5 as a function of the mass ratio.

factor can be achieved. The plot of the gamma factor as a function of the mass remaining in the rocket is shown in Figure 7.6. With a high enough exhaust velocity, stellar trips are possible.

7.4. Charge in an Electric Field

A specific realization of a situation with uniform proper acceleration occurs for a charged particle immersed in a uniform electric field. Instead of force, in special relativity, the replacement of the force by the time rate change of momentum is often the correct modification to obtain relativistically correct equations. The solution of the equations of motion in one dimension is shown in Equation (7.4).

The momentum is P, the electric field is E, $a = qE/m$, and the particle energy is $\varepsilon = \sqrt{P^2 + m^2}$. The momentum increases linearly with time. If at is much less than one, the classical results are recovered. The script "sr_E_Accel" solves for velocity and position symbolically and creates plots both for the relativistic and the classical cases. The velocity plot is shown in Figure 7.7. The velocity is limited to c

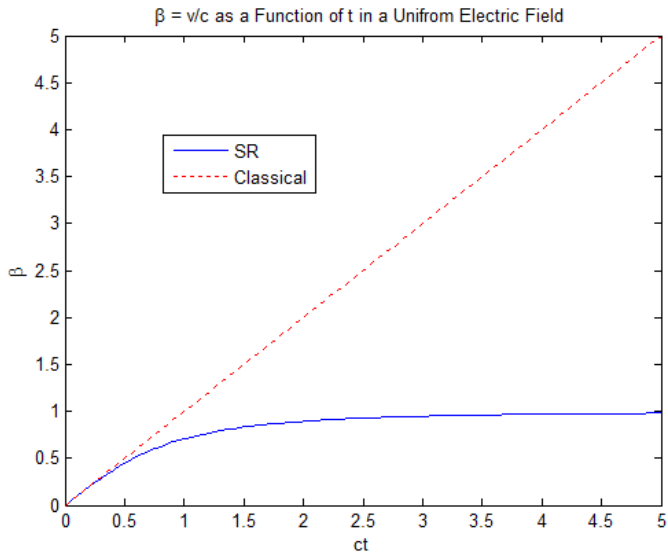


Figure 7.7: Velocity as a function of time for a charged particle in a uniform electric field.

in the case of relativistic mechanics. Note that in Equation (7.2), the time t^* referred to the passenger's rest frame while t here refers to the clock time in the frame where the charge moves and has momentum of zero at $t = 0$.

$$\begin{aligned} dP/dt &= qE \\ P &= qEt \\ \beta &= P/\varepsilon = at/\sqrt{(at)^2+1} \\ a &= qE/m \\ z &= a[\sqrt{(at)^2+1}-1] \end{aligned} \tag{7.4}$$

The Taylor series for the classical z result has a first correction term, $a^3t^4/8$.

7.5. Charge in Electric and Magnetic Fields

The more complex case of both electric and magnetic fields in three dimensions is treated in the script "ExB_ODE_SR". Again, the

replacement is of the force by the time rate change of the relativistic expression for momentum. The notation is the same as was used in Section 7.4. The equations appear in Equation (7.5). This script is the relativistic generalization of that already described in Section 3 of the text.

$$\begin{aligned} d\vec{P}/dt &= q(\vec{E} + \vec{\nu} x \vec{B}) \\ \vec{\beta} &= \vec{P}/\varepsilon = \vec{P}/\sqrt{\vec{P}^2 + M^2} \\ d\vec{P}/dt &= q(\vec{E} + \vec{P} x \vec{B}/\varepsilon) \\ d\vec{x}/dt &= c\vec{P}/\varepsilon \end{aligned} \tag{7.5}$$

When the velocity is small with respect to c, $\vec{P} \to m\vec{\nu}$, $\varepsilon \to M$ and the NR equations are recovered, the time rate change of momentum due to a magnetic field is now limited by the inverse energy factor which replaces the NR mass factor. Classically, the Larmor angular frequency of circular rotation qB/m becomes $qB/\gamma m$, while the radius of circular motion goes from $v_T m/qB$ to P_T/qB where the subscript T means transverse to the B-field direction. There are practical ramifications. An accelerator needs to change the frequency of the r.f. which supplies energy to the beam as the accelerated particles gain energy and the radius continues to increase, requiring larger magnetic field volumes, even though the velocity approaches the limit of c.

The script uses the "ode45" MATLAB tool to solve the six differential equations for the three momentum components and the three positions as was the case in the classical treatment. The units are chosen so that all quantities are of order one. The user chooses the three electric field components, the magnetic field magnitude and the three initial momentum components. It is useful to start by trying special cases with only electric or only magnetic fields. The example which is plotted in Figure 7.8 and Figure 7.9 is for the choices $E = [0.3\ 0\ 0.3]$, $B = 1$, $P = [10\ 0\ 0]$. The momentum components are plotted in Figure 7.8. The P_z starts at zero but is increased by E_z. The P_x starts at 10, increases slightly due to E_x but is also curved by B_z. The (P_x, P_y) contour is shown in Figure 7.9. It is circular

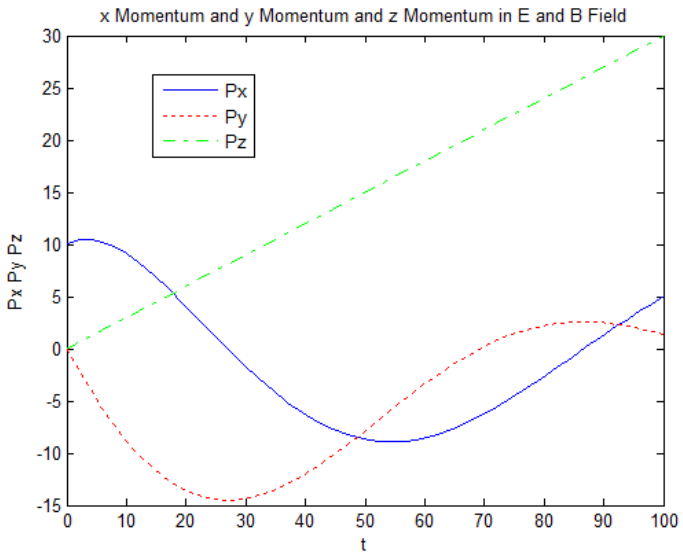


Figure 7.8: Plot of P_x, P_y and P_z as a function of time for a specific, user defined, example.

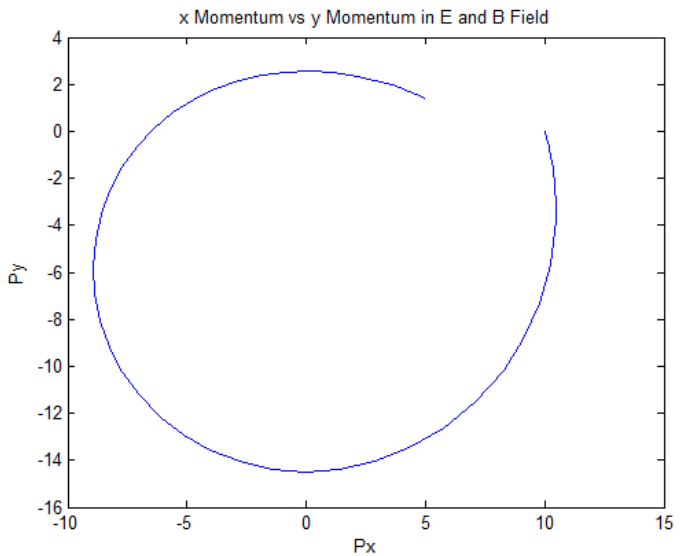


Figure 7.9: Plot of P_x vs P_y for a specific, user defined, example. Note that the radius of curvature in the magnetic field is now dependent on the momentum which increases with time due to acceleration by the electric field.

due to B_z but the radius of curvature depends on the momentum perpendicular to B_z, which increases due to E_x leading to a radius of curvature which is momentum dependent.

7.6. Relativistic Scattering and Decay

The relativistically correct scattering and decay kinematics are provided in the script "sr_dec_scat". The situation in classical mechanics was previously demonstrated in Section 2 of the text. In this case, instead of the conservation of vector velocity and scalar kinetic energy, the vector momentum and the scalar total energy are conserved. These modified conservation laws define the kinematics.

There are two cases explored here. For the decay of a particle of mass M into two particles of mass m, the center of momentum, CM, quantities are simply:

$$\begin{aligned} M_{\mathrm{CM}} = M, \beta_{\mathrm{CM}} = P/\varepsilon, \gamma_{\mathrm{CM}} = \varepsilon/M \\ \varepsilon^* = M/2 \end{aligned} \tag{7.6}$$

The CM energy is simply the mass M particle, and the daughters of mass m share the CM energy equally. The mass M particle moves in the lab frame with momentum P.

In the second case of elastic scattering, $m + M \rightarrow m + M$, the initial state has a target of mass M at rest and a projectile, mass m, moving with momentum P and energy ε. The CM in this case is:

$$\begin{aligned} M_{\mathrm{CM}}^2 = m^2 + M^2 + 2M\varepsilon, \\ \beta_{\mathrm{CM}} = P/(\varepsilon + M), \gamma_{\mathrm{CM}} = (\varepsilon + M)/M_{\mathrm{CM}} \\ P^* = PM/M_{\mathrm{CM}} \end{aligned} \tag{7.7}$$

The CM quantities for final state objects are indicated by a * superscript. In both cases, they are defined by a single "scattering/decay angle", θ^*. Making a Lorentz transformation back to the laboratory:

$$\begin{aligned} P_T = P^* \sin\theta^* \\ P_L = \gamma_{\mathrm{CM}}(P^* \cos\theta^* + \beta_{\mathrm{CM}}\varepsilon^*) \end{aligned} \tag{7.8}$$

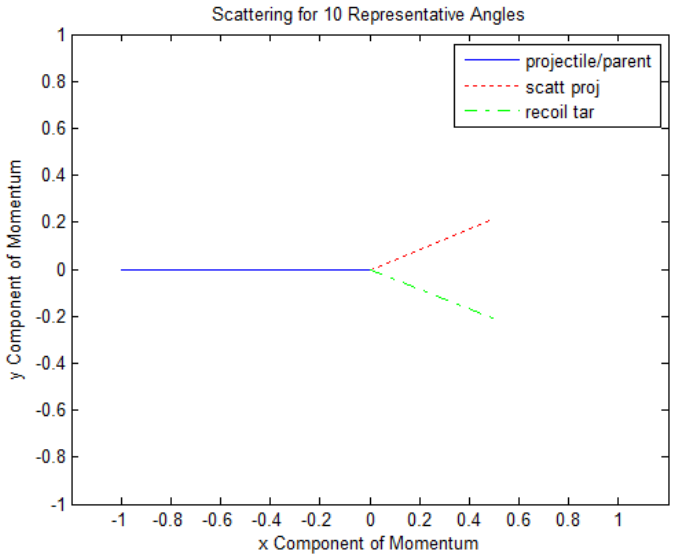


Figure 7.10: Scattering of equal mass particles. The angle between the scattered projectile and the recoiling target is less than ninety degrees for high momentum projectiles.

As an example of a difference from the classical situation, for the scattering of equal mass particles, ($m + M \rightarrow m + M$ in general with target mass M), the angle between the outgoing particles is no longer ninety degrees as it was in the NR case. An example is shown in Figure 7.10. In this script, a "movie" is provided illustrating the situation for several different scattering angles. The user chooses M and the momentum of the projectile, mass m. There is also a choice to explore scattering or decays.

In relativity mass can be converted into energy. Therefore, a heavy object can convert to two or more lighter objects and also impart momentum to them. Such decays are not possible in classical mechanics where mass is independently conserved. An example of a parent with mass, $M = 3$, and momentum $P = 5$, decaying into two daughter particles with mass, $m = 1$ is shown in Figure 7.11. The printout gives the maximum angles. The maximum of one angle corresponds to nearly the minimum angle of the other. A movie is shown for different decay angles in the lab frame.

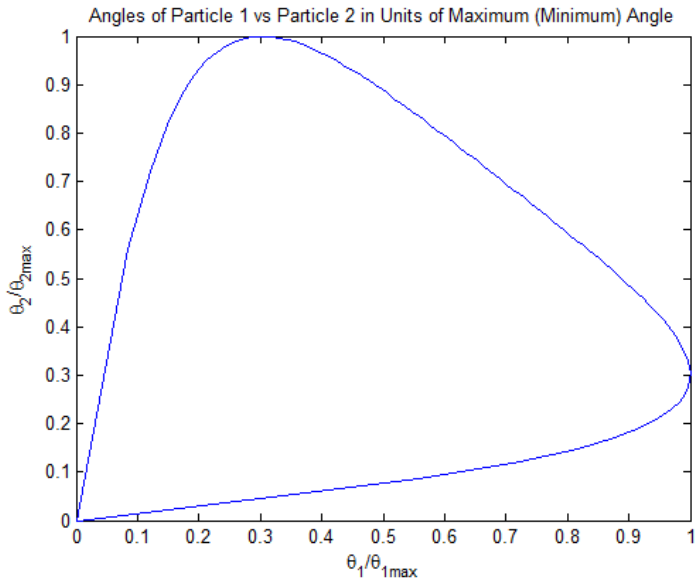


Figure 7.11: Correlation of the decay angles with respect to the parent direction for a parent with $M = 3$ decaying into two daughters with $m = 1$. The angles are scaled to their maximum attained values.

```
>> sr_dec_scat
  cm_dec_scat - Decays and Scattering - Relativistic


Enter Mass of Target (Projectile m =1) or Decay Particle (Daughter m = 1) 3
Enter Momentum of Incoming Particle in m=1 Units 5
CM Energy = 3, beta CM = 0.980581, gamma CM = 1.69967, P in CM = 1.11803
Maximum(Minimum) Angles for Particle 1 = 0.602786 (rad) and 2 = -0.602786
```

Figure 7.12: Printout for the specific case of the decay of a parent of mass, $M = 3$, with laboratory momentum = 5.

The user dialogue for the case of particle decay is shown in Figure 7.12. When the mass M and momentum of the decaying particle are specified, the only remaining variable is the decay angle in the CM. The CM energy, beta and gamma and the momentum of the daughters, $m = 1$, in the CM are already specified by the kinematics. The script runs over all possible decay angles in making the plots.

It is very useful for the user to vary the parameters for both scattering and decay. In this way, some facility with the range of possible effects can be attained.

7.7. Electric Field of a Moving Charge

Previously, the focus was on kinematics in relativity; time dilation, rockets, acceleration and scattering or decay. The focus now shifts to dynamics — the collisions between particles, the interactions between them and the radiation arising from acceleration when particles are in relativistic motion.

The electric field of a charge, q, in uniform motion, velocity $v = c\beta$ depends on the angle between the velocity and the observation point at the present position of the charge, θ, not the position when the light was emitted (the retarded position).

$$\vec{E} \sim [q(1-\beta^2)/(1-\beta^2 \sin^2\theta)^{3/2}]\vec{r}/r^3 \tag{7.9}$$

The transverse field scales as gamma, while the longitudinal field falls as the inverse square of gamma. The non-relativistic isotropic field is recovered at low velocity. The fields are plotted in the script "E_SR", where first the user chooses a velocity. Then, as shown in Figure 7.13, contours for a representative range of velocities are computed and plotted. The longitudinal field shrinks and the transverse field grows with velocity as expected from Equation (7.9). In the limit of highly relativistic motion, the electric field looks like a "pancake" oriented transverse to the direction of motion.

7.8. Minimum Ionizing Particle

Again consider a charged particle moving uniformly along the z axis. The observation point is chosen to a point fixed in space in the lab frame at fixed x with y equal to zero. The longitudinal electric field at the observation point integrates to zero by symmetry. The transverse field, with increasing velocity, is compressed in time but increases in field strength, leading to a momentum impulse, $q\int E_x dt$, which is constant, independent of incident particle velocity at high velocities. This means all fast particles impart the same energy, independent of

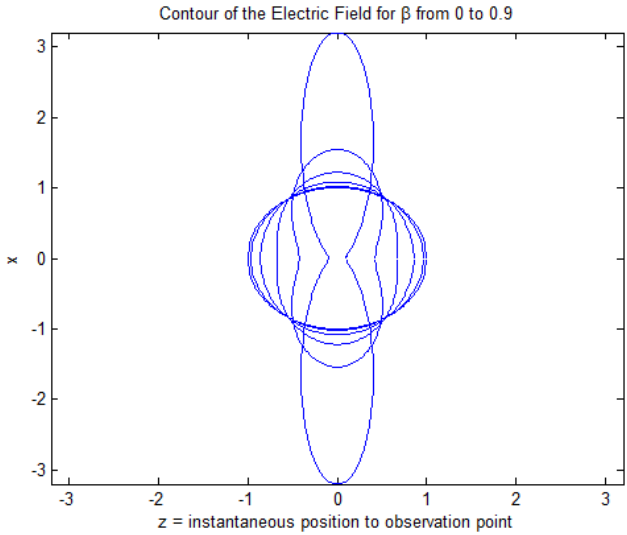


Figure 7.13: Contours of the electric field of a uniformly moving charge for different β values.

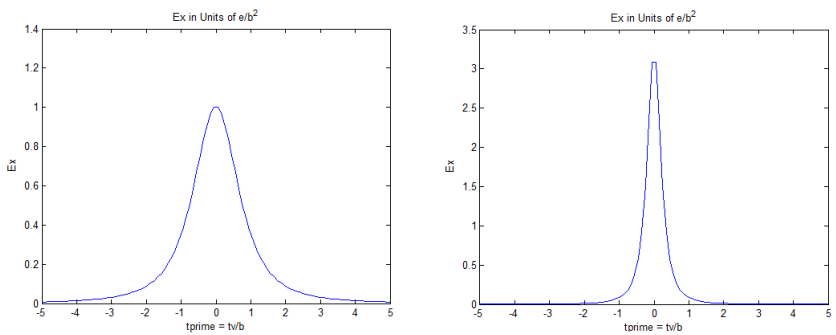


Figure 7.14: Transverse fields of a moving charge observed at the point $x = d$, $y = 0 = z$. At left the charge moves with $\beta = 0.1$, while at right it has velocity, $\beta = 0.95$.

velocity. The transverse field for two velocities is shown in Figure 7.14 as computed in the script "E_Move_Charge_SR". The field is plotted in the natural units of e/d^2, while the time is plotted in units of d/v. In the case of slow velocities, the field should be of magnitude one and persist for a time of order one. Indeed, as seen in the leftmost plot in Figure 7.14, with $\beta = 0.1$, this expectation is borne out.

A movie of the moving charge is provided, where the vector from the charge to the observation point is shown in blue, while the electric field vector at the point is displayed in red. One frame in the case corresponding to the right plot of Figure 7.14 appears in Figure 7.15.

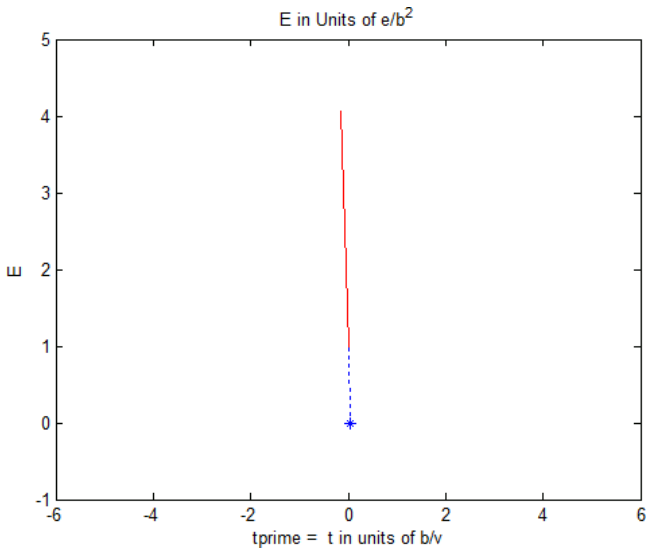


Figure 7.15: A frame of the movie for a charged particle with $\beta = 0.95$ when the charge is near the observation point. The blue line goes from the charge, blue *, to that point, while the red vector shows the size and direction of the electric field.

At high velocity, the transverse field gets stronger (Section 7.7), while the time of activity Δt shrinks as illustrated in the right plot of Figure 7.14. The result is that the momentum impulse ΔP imparted by fast particles is independent of velocity. The transfer of kinetic energy to the observation point ΔT approaches a constant as the velocity approaches c. At low velocities the energy transfer is greater than at large velocity, so that fast particles are "minimum ionizing".

$$\begin{aligned} \Delta t &= b/v \rightarrow b/\gamma v \\ F &= e^2/b^2 \rightarrow \gamma e^2/b^2 \\ \Delta P &= e^2/bv \rightarrow e^2/bc \\ \Delta T &= \Delta P^2/2m = e^4/4Tb^2 \rightarrow e^4/2mcb^2 \end{aligned} \tag{7.10}$$

7.9. Range and Energy Loss

The kinetic energy transfer goes as $1/T$ when $\beta < 1$, as shown in Equation (7.10). For example a 100 MeV proton goes 5.35 cm in water before giving up all its energy or what is called coming to the end of its range. Range, R, goes as square of initial kinetic energy, $dT/dz \sim r/T$, where r is proportional to the minimum energy loss per unit distance.

$$T = \sqrt{T_o^2 - 2rz}$$
$$R = T_o^2/2r \tag{7.11}$$

Range calculations are made for protons in water in the script "Range_Energy". The user supplies an initial energy. Results for a 100 MeV proton are shown in Figures 7.16 and 7.17. The user can make a series of choices, as usual.

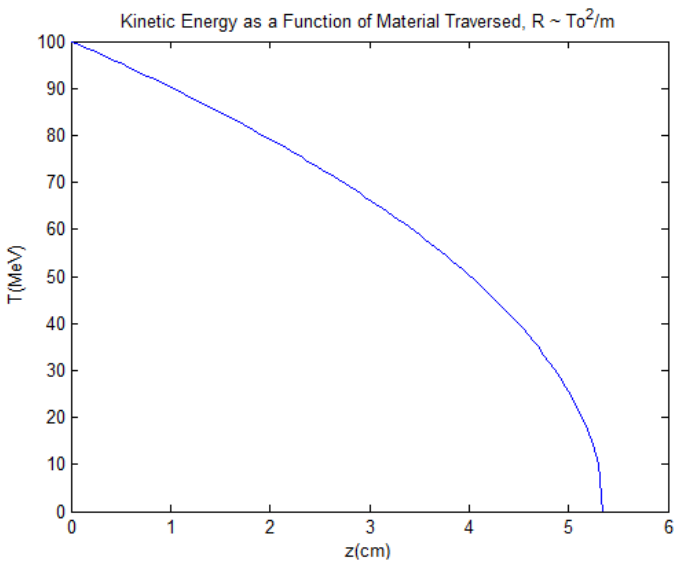


Figure 7.16: Kinetic energy of a 100 MeV proton in water as a function of the distance travelled.

The majority of the proton energy is deposited at the end of the range, since $dT \sim 1/T$. This feature is very useful in some medical applications. In proton therapy, the initial energy can be set

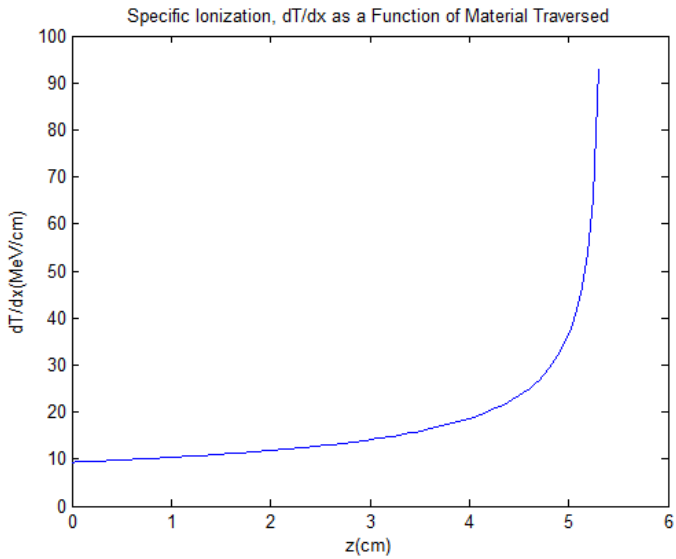


Figure 7.17: Energy deposited by a 100 MeV proton in water at a specific location as a function of the distance travelled.

by a cyclotron in order to target the deposit of ionization energy for radiation therapy at a very specific location. This minimizes the dose given to healthy intervening tissue.

7.10. Relativistic Radiation

In general, the importance of radiation loss by a charged particle increases with the velocity of the particle. Energy loss from ionization or other processes dominates at low velocities, but radiation takes precedence at high velocities, mirroring the growth of the electric and magnetic fields and the effect of relativity in the case of radiation. In comparison, ionization approaches a minimum as the velocity increases and all particles become minimum ionizing.

The radiated power depends on the square of the acceleration. In the NR limit, the radiated power goes as $a^2 \sin^2 \theta$, where a is the acceleration and the angle θ is between the observation point and the acceleration. The radiated power, defined to be p here, for two special cases, acceleration parallel, L, to the velocity and circular motion with acceleration perpendicular, T, to the velocity has the

angular dependence:

$$dp_L/d\Omega \sim \sin^2\theta/(1-\beta\cos\theta)^5$$
$$dp_T/d\Omega \sim 1/(1-\beta\cos\theta)^3[1-\sin^2\theta/\gamma^2(1-\beta\cos\theta)^2] \quad (7.12)$$

In the case where the acceleration and the velocity are parallel, the angle is as defined in the NR case. Now, however, there is an angle, $\theta_{\max} \sim 1/2\gamma$, where the radiated power is at a maximum. In general, the power is directed "forward", in the direction of the velocity vector. The total radiated power is larger than in the NR case by a factor of γ^6 for the parallel case. In the case of circular motion, the velocity is taken to be along the z axis, while the acceleration is along the x axis. The distribution shown in Equation (7.12) applies to the case where the azimuthal angle ϕ is zero so that it applies in the plane defined by the velocity and acceleration vectors. The power in this case is a factor γ^4 larger than the NR power. In the NR limit the dipole pattern is recovered, but $\sim \cos^2\theta$ because the angle is defined with respect to the velocity and not the acceleration.

The power spectra as a function of observation angle are computed in the script "Rel_Radiate". The user picks a velocity to examine. At the end of the script a family of curves is generated for different velocities. The radiation pattern contours are shown for the two special cases in Figures 7.18 and 7.19.

In the parallel case, the low velocity limit is a familiar dipole like $\sin^2\theta$ pattern. That pattern is distorted and tipped forward in the high velocity case. In the perpendicular case, the pattern is already forward — backward symmetric in the low velocity case. The effect of high velocity is to strengthen the forward radiation and shrink its angular extent. The forward going nature of the pattern is a general property and it is called the "searchlight effect" in the literature.

7.11. Compton Scattering

Consider first the simpler case of a source of light at rest in the starred frame, S*, and moving with velocity β in a second inertial frame, S. Using the Lorentz transformation of the energy and longitudinal momentum of a photon, the purely kinematic effect of SR is shown

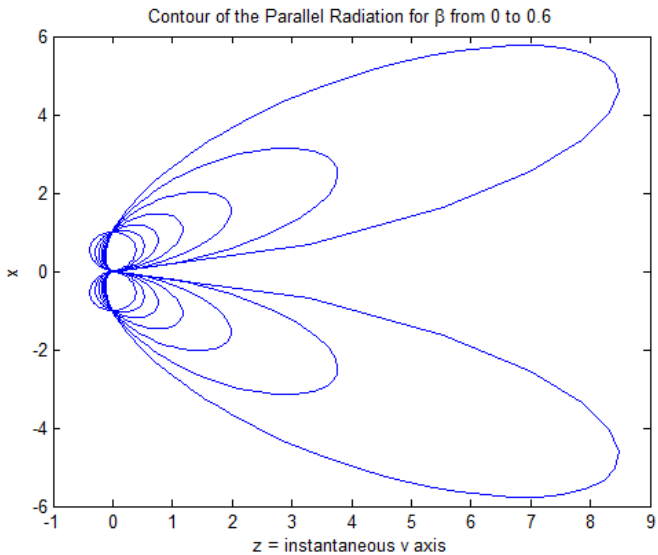


Figure 7.18: Parallel acceleration angular pattern as a function of the source velocity.

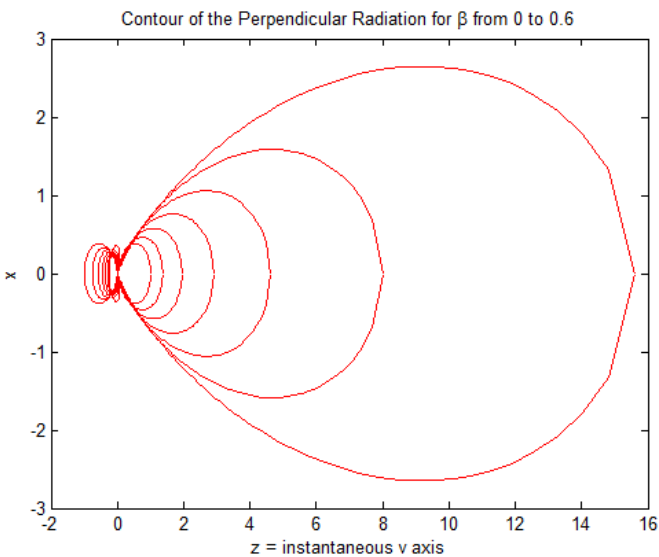


Figure 7.19: Perpendicular acceleration angular pattern as a function of the source velocity.

in Equation (7.13), assuming that the light is isotropic in the S* frame.

$$\begin{aligned} P^* \cos\theta^* &= \gamma P(\cos\theta - \beta) \\ P^* &= \gamma P(1 - \beta\cos\theta) \\ \cos\theta^* &= (\cos\theta - \beta)/(1 - \beta\cos\theta) \\ d\sigma/d\Omega &= 1/[\gamma^2(1 - \beta\cos\theta)^2] \end{aligned} \tag{7.13}$$

This factor was seen already in Equation (7.12) and is purely of kinematic origin.

Compton scattering is the scattering of a photon off an atomic electron. The low velocity angular distribution was mentioned previously and is called Thompson scattering. The angular distribution goes as: $1 + \cos^2\theta$ which is forward-backward symmetric. At higher photon energies, much larger than the electron rest mass, the distribution is thrown forward.

The Compton effect was one of the first experimental indications of the particle nature of light. Since light carries energy and momentum, then when it scatters off an electron, the electron recoil can take off significant energy. The outgoing photon then has lost energy, going from E_o incident to E outgoing. The energy relationship can be determined solely by applying particle kinematics to the photon and solving the equations expressing the conservation of energy and momentum. The angular distribution was first worked out by Klein and Nishina, where the angle θ is the angle of the outgoing photon with respect to the incident photon.

$$\begin{aligned} y = E/E_o &= 1/[1 + (E_o/m)(1 - \cos\theta)] \\ d\sigma/d\Omega &\sim y^2[y + 1/y - (1 - \cos^2\theta)]/2 \end{aligned} \tag{7.14}$$

The Compton angular distribution is evaluated in the script "ComptonScat". The user chooses a velocity and then the angular distribution for a set of velocities is plotted in Figure 7.20. At 10 keV, the distribution is quite isotropic, while at 100 MeV, roughly 200 times the electron mass, the forward peak of the photon is very prominent. The NR limit where the energy is much less than the

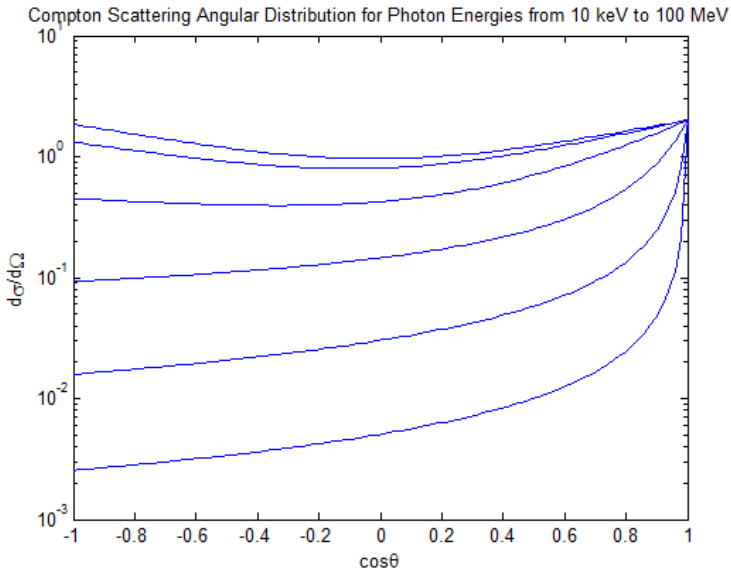


Figure 7.20: Angular distribution of the photon scattering angle for Compton scattering as a function of the photon energy.

electron mass gives $y \to 1$ and $d\sigma/d\Omega \to (1+\cos^2\theta)/2$ which are the NR Thompson results.

7.12. Photoelectric Effect

The photoelectric effect describes the absorption of a photon of energy, $\hbar\omega$, by an electron bound in a material with a work function, V_o, and the subsequent emission of the electron with energy equal to $\hbar\omega - V_o$. This effect was one of the first to display the quantum aspects of light and was cited in the Nobel Prize of Einstein. Energy conservation implies that $\hbar\omega - V_o = P^2/2m$.

The photoelectric effect has an angular distribution which is also quite dependent on the energy of the incident photon. The photon has transverse electric fields which, at low energies, exert forces on the electrons and preferentially eject them at right angles to the photon. With higher energy photons, the angular distribution for

light elements, $Z\alpha << 1$, and in an unscreened approximation is:

$$d\sigma/d\Omega \sim \sin^2\theta/[(Z\alpha)^2 + 2E/mc^2(1-\beta\cos\theta)]^4 \\ \rightarrow \sin^2\theta/(1-\beta\cos\theta)^4 \qquad (7.15)$$

The angle θ is the angle of the ejected electron with respect to the incident photon direction and β refers to the electron velocity.

The angular distribution is evaluated in the script "Photoelectric." As per usual, the user chooses an energy and then the angular distribution for a representative set of photon energies is computed. The resulting plot is shown in Figure 7.21. At high energies, the ejected electrons are thrown forward, while at low energies, the electrons are ejected in the direction of the transverse electric fields of the photon.

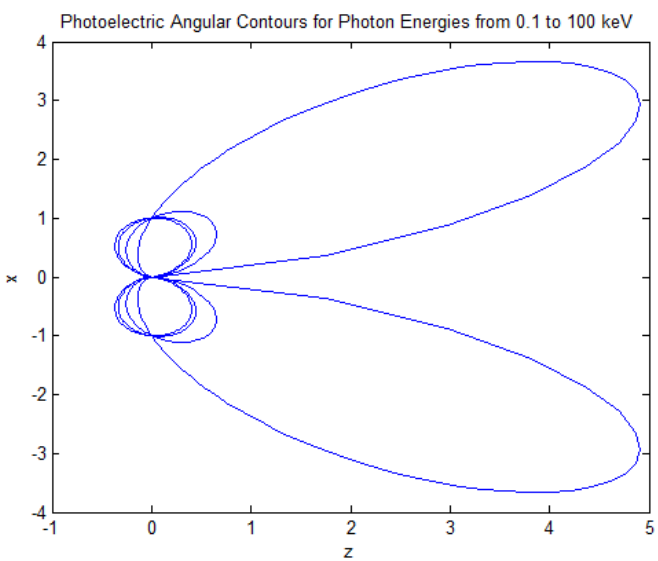


Figure 7.21: Angular distribution for electrons emitted in the photoelectric effect for different photon energies.

7.13. Electrons and Muons in Materials

The passage of electrons and heavy electrons, or muons, through high Z materials has points of interest. In this case, experimental data is

used. The script is "HF_Movie_e_u". The data comes from work done with beams of electrons and muons incident on a block of material which is instrumented to sample the energy deposited as a function of the depth in the block.

The total depth of the block is 5 inches of lead. There are 40 samples of the energy. The script provides a movie of the passage of twenty electrons of 156 GeV (1 GeV $= 10^9$ eV) mean incident energy, then 10 muons of 15 GeV energy, followed by 10 muons of 240 GeV energy. Printout is shown in Figure 7.22.

```
>> HF_Movie_e_u
  movie for HF data, e  and muons incident


Hanging File Test Beam Data
shows depth distribution for electrons and muons
Hanging File stack is 40 plates of 1/8"
horizontal axis is plate number, vertical is energy in that plate
 Mean Total Energy in 40 Plates for 15 and 240 GeV Muons = 0.464465, 0.436163 (GeV)
 Mean Total Energy in 40 Plates for 150 GeV Electrons = 160.253 (GeV)
```

Figure 7.22: Printout for the script to study energy deposits by electrons and muons.

The electrons interact strongly by radiating and deposit essentially all their energy in the block as seen in Figure 7.23. They are very relativistic with a gamma factor of about $156{,}000/0.511 = 305{,}000$. The muons have masses about 205 times larger than the electrons and, therefore, a smaller gamma factor. Muons and electrons have the same charge and the same electromagnetic interactions.

The electrons lose energy by radiation. A cascade of electrons and photons develops, which ultimately deposits all the electron energy in the block. Since energy loss by radiation is proportional to a high power of gamma, the muons do not radiate as much. In fact, they serve as a sample of minimum ionizing particles, depositing much less energy in the block. As seen from the printout, the electrons deposit 150 GeV, while the muons of 15 and 240 GeV energy deposit almost the same energy of less than one GeV. Clearly, the muons represent minimum ionizing particles that deposit the same amount of energy independent of their own energy, as expected.

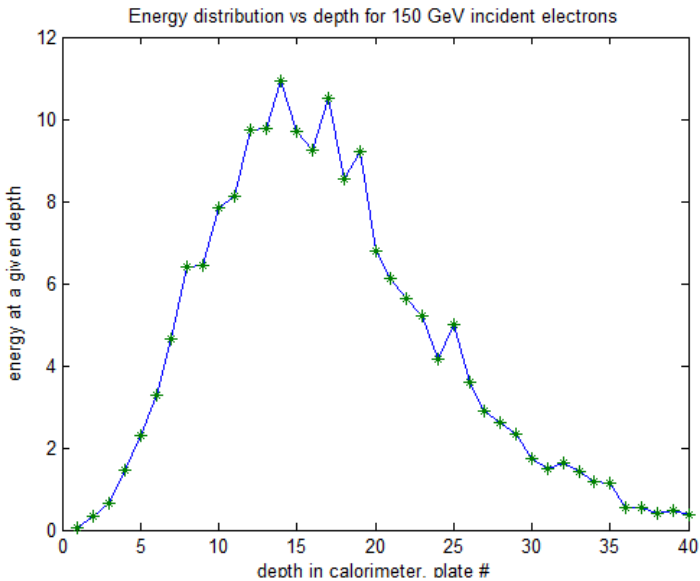


Figure 7.23: Energy deposit for a 156 GeV electron as a function of depth in a block of lead. There are 40 plates, $1/8''$ thick, for a total of 5 inches.

Parenthetically, most cosmic rays at the surface of the earth are muons because they survive as minimally ionizing parts of the cosmic ray showers, while electrons are absorbed in the atmosphere and because their decay times are time dilated, as mentioned previously. The energy deposit for a 15 GeV muon is shown in Figure 7.24. Note that, although there are large statistical fluctuations, the scale of energy deposits is small compared to Figure 7.23. A rough estimate is that the muons will lose 0.16 GeV by ionization in traversing the block. The muons are not totally absorbed in the block, while the electrons are. This observation shows that radiation is indeed paramount for relativistic particles.

7.14. Radial Geodesics

In special relativity (SR) relative motion affected both time and space. There was an invariant interval, the proper time of a clock at rest in a frame, with time, *ds*. In a frame where the clock moved

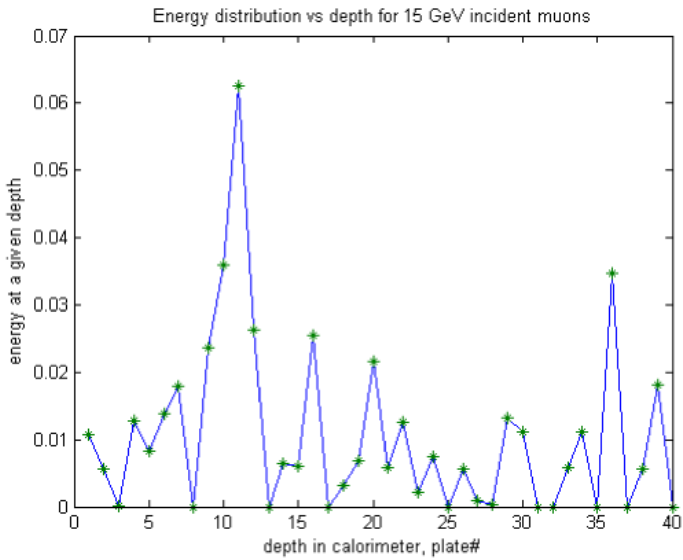


Figure 7.24: Energy deposit for a 15 GeV muon passing through a block of lead.

with a velocity, v, there was a time interval, dt

$$\begin{aligned}(ds)^2 &= (cdt)^2 - (d\vec{x})^2 \\ &= (cdt/\gamma)^2 \end{aligned} \tag{7.16}$$

In Equation (7.16), the basic time dilation effect is very clear. In general relativity (GR) gravity and mass are incorporated into the metric of space and time, where in SR this metric, Equation (7.16), is Euclidian, or "flat".

In general relativity (GR) masses determine the geometry of space-time and then particles move on geodesics in that non-Euclidian space-time. There are only a few solvable solutions of the nonlinear GR field equations which relate the mass to the metric. In addition, GR is nonlinear so that solutions are not additive. In the case of a Schwarzschild solution for a non-rotating point mass, M, the metric is:

$$\begin{aligned} ds^2 &= (cdt)^2(1 - r_s/r) - dr^2/(1 - r_s/r) - r^2 d^2\Omega \\ r_s &= 2GM/c^2 \end{aligned} \tag{7.17}$$

This metric is the interval between two events labeled by coordinate clock time, t, and ruler distance, r. The metric is spherically symmetric but the temporal interval and the radial interval are non-Euclidian, with a characteristic length called the Schwarzschild radius, r_s, determined by the gravitational coupling, G, of the point mass, M. This solution is the GR analogue of the classical point particle solution for a mass, M, expressed as a GM/r potential.

Particles move on geodesics of this metric. The geodesic is the path of maximal metric, like a great circle route on a spherical surface. In a flat space, with no masses, the geodesic is a straight line which is expected for a free particle. We consider only a simple case here, radial motion with an initial location r_o and initial velocity dr/dt equal to zero. The interval at large $r \gg r_s$, is the SR interval with clocks and rulers of observers at large r. In the space ds refers to proper clocks. The solutions to the geodesic equations are:

$$\begin{aligned} s &= \int ds = \sqrt{r_o/r_s} \int \sqrt{r/(r_o - r)} dr \\ ct &= c \int dt \\ &= \sqrt{(r_o/r_s)(1 - r_s/r_o)} \int (r^{3/2}/[\sqrt{r_o - r}(r - r_s)]) dr \end{aligned} \quad (7.18)$$

The solution for the trajectory as a function of proper time, s, turns out to be the classical solution. The solution in terms of coordinate time, t, has additional factors. Solutions for both time markers are constructed in the script "gr_schwarz". The user chooses an initial radius and a mass, M, and the trajectories are computed and plotted. The printout is shown in Figure 7.25 for the specific choice of four solar masses and dropping from rest at a radius of five Schwarzschild radii. The equations are integrated using a trivial numerical approach, although symbolic solutions are possible. For example, the total proper time to reach the origin is $\pi r_o^{3/2}/2r_s$. Indeed, dr/ds is clearly well behaved for all r, while dr/dct approaches zero as $r \to r_s$.

```
>> gr_schwarz
  radial geodesics in a Schwarzchild space


Enter Mass in Solar Masses: 4
Schwarzchild Radius (km) = 11.8578
Enter Radius in rs units From Which You Are Released At Rest: 5
Velocities, dr/ds and dr/dct, drds is Classical Result

  / rs \1/2 /   r - ro \1/2
  | -- |    | - ------ |
  \ ro /    \     r    /

          1/2          /   rs    \1/2
  (ro - r)    (r - rs) | ------- |
                       \ ro - rs /
  ----------------------------------
                  3/2
                 r
Proper Time to Reach r = 0

       3/2 / 1  \1/2
  pi ro    | -- |
           \ rs /
  ------------------ |
           2
```

Figure 7.25: Printout for the script "gr_schwarz". The velocities computed using proper and coordinate times are plotted.

The two velocity measurements are shown in Figure 7.26, while the two time markers are shown as functions of r in Figure 7.27. The velocity dr/ds, smoothly increases as the object falls into the black hole at the origin. In contrast, the velocity dr/dct vanishes at the Schwarzschild radius. The proper time, or the time for an observer at rest with respect to the particle, is well behaved and follows the classical solution. On the other hand, the coordinate time, or the time recorded by an observer at large radius approaches infinity as the radius approaches the Schwarzschild radius.

This difference is due to the fact that a gravity field influences the flow of time. Science fiction writers often use this fact. From the viewpoint of an observer far away, the object never reaches the

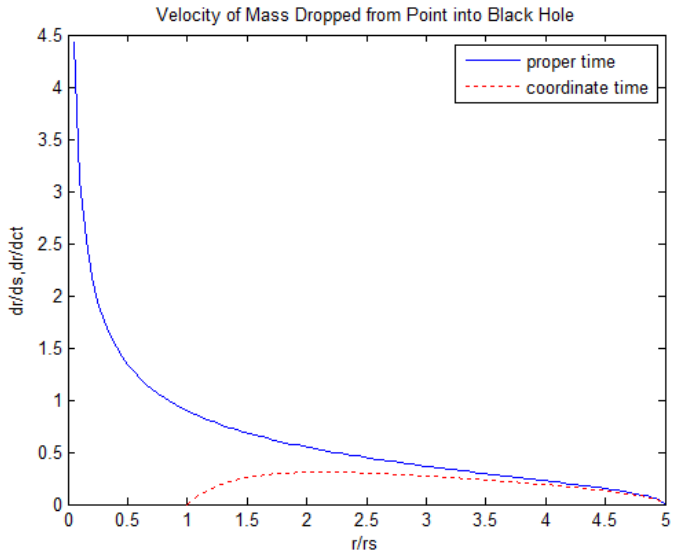


Figure 7.26: dr/ds and dr/dct velocities as a function of radius.

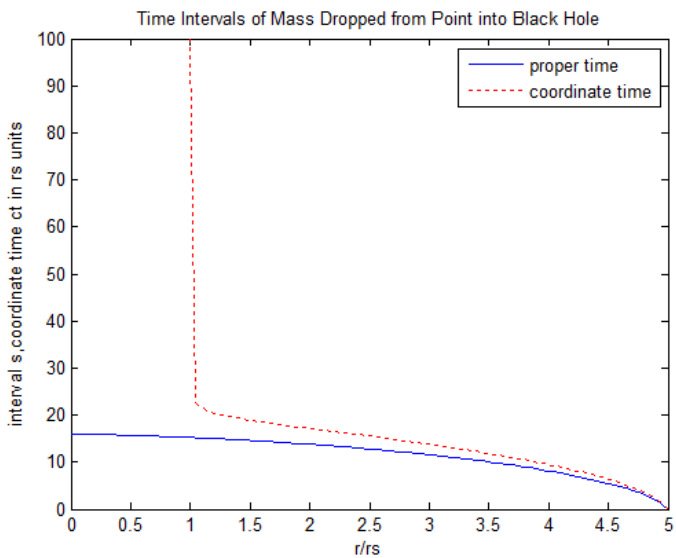


Figure 7.27: Times s and ct as a function of radius for motion on a Schwarzschild radial geodesic for a particle released at rest at 5 Schwarzschild radii.

Schwarzschild radius; while for a person riding the object, all seems finite and classical. Time appears to run slowly in a gravity field to an outside observer. However, as will be seen later, tidal forces will destroy any object which falls into the "black hole".

7.15. Inspiraling Binary Stars

As was the case in electromagnetism, accelerated gravitationally coupled systems can generate gravitational waves. In most cases, these are below the threshold of present detection methods because space-time is "stiff" and difficult to deform. However, in extreme cases, detection might be possible. One such case concerns binary star systems. These are very numerous in our galaxy. It is important that the stars be compact so that the accelerations will be large because the distances are small. We will see later that heavy stars will either form a black hole or will become very compact neutron stars and, therefore, the assumption of a compact star is not absurd.

The total luminosity of gravitational radiation for a binary is:

$$L \sim (32G^4/5c^5)2M^4/R^5 \tag{7.19}$$

Consider a binary star system of stellar mass M orbiting about the common CM with radius R. The system radiates gravitational energy and this causes it to be more tightly bound which means that the radius decreases. This is the "death spiral". The period, T, radius, R, and orbital frequency, ω, as a function of time, relative to the collapse time, t_c are:

$$\begin{aligned} dT/dt &= -96/5(4\pi^2)^{2/3}(GM)^{5/3}/[c^5 2^{1/3} T^{5/3}] \\ R(t) &= [(32(GM)^3/5c^5)(t_c - t)]^{1/4} \\ \omega^2(t) &= GM/4R^3(t) \end{aligned} \tag{7.20}$$

The radiation is quadrupole because there is no negative mass and a dipole is, therefore, impossible. The radius goes to zero and the frequency rises rapidly, hence the name "chirp" as a characteristic signature for a binary system collapse due to gravitational radiation.

The gravitational radiation frequency increases as t as $R(t)$ decreases, leading to a space-time deformation going as the

$-1/4$ power of t. The binary motion is demonstrated in a script "GR_Chirp". The user chooses a binary mass, M, and an initial radius R. A "movie" is then shown as the binary spirals in toward a collision. The masses are assumed to have no radial extent. The typical frequencies are a few kHz, so that gravity wave detectors using the binaries as "standard candles" need to be sensitive in this frequency range.

Printout for the script "GR_Chirp" is shown in Figure 7.28 while $R(t)$ is shown in Figure 7.29. As the radius approaches zero the frequency increases very rapidly, as shown in Figure 7.30.

```
>> GR_Chirp
  Inspiraling Binary due to Grav Radiation


Enter Star Mass in Solar Masses: 1
Enter Initial Binary Radius in km (< 20): 10
Initial Orbital Frequency (Hz) = 5774.95
Inspiral Time (sec) = 0.00159941
```

Figure 7.28: Printout for a specific choice of binary mass and radius.

The movie which is produced can be very instructive. The system changes radius and angular velocity in a quite characteristic way as the binary collapses under the action of energy loss due to gravitational radiation. Indeed, existing detectors use the characteristic pattern as a way to reject random noise and improve the sensitivity of their searches using the expected "chirp" signature. The user should try several masses and initial radii to explore how the binary system collapses.

7.16. Gravity Wave Detector

Assuming that gravitational radiation exists, as predicted in GR, detectors need to be designed to discover the existence and then location of the sources of that radiation. What exactly defines gravity? It is not a simple acceleration, because that can be removed by going

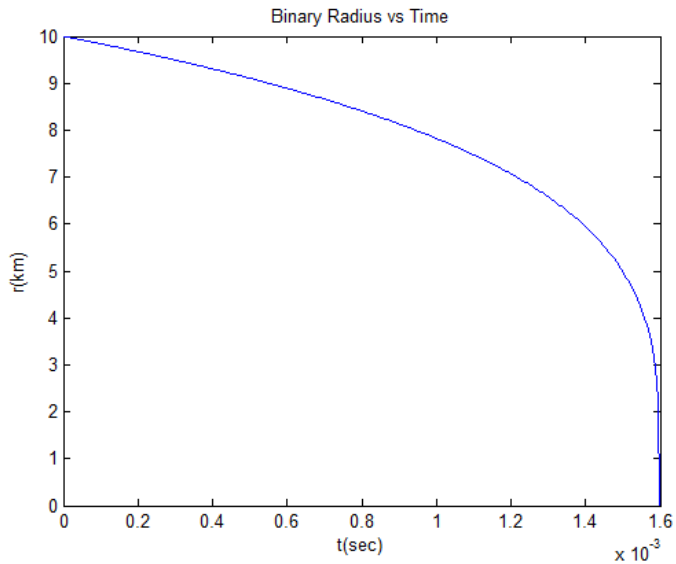


Figure 7.29: Plot of the radius of the binary system as a function of time.

into a freely falling reference system as in the thought experiments of the Equivalence Principle. What are definitive are the tidal forces. For a mass M interacting with a test mass m, these forces tend to elongate an extended object in the direction of M, z, and compress it in a direction perpendicular to M, x. Expanding the force F about the center of an object located at a radius r from the mass M:

$$\begin{aligned} F_z &= 2z(GMm/r^3) \\ F_x &= -x(GMm/r^3) \end{aligned} \tag{7.21}$$

The tidal forces are divergence-less, so that a tidal potential, can be defined:

$$\Phi_{tide} = -(z^2 - x^2/2)/r^{3/2} \tag{7.22}$$

A contour plot of that potential is displayed in Figure 7.31. The script "Grav_Rad_Tidal" also produces plots for the x and z forces and a movie of the response of a gravity wave "antenna" to a passing wave, shown in Figure 7.32. The compression and elongation of the

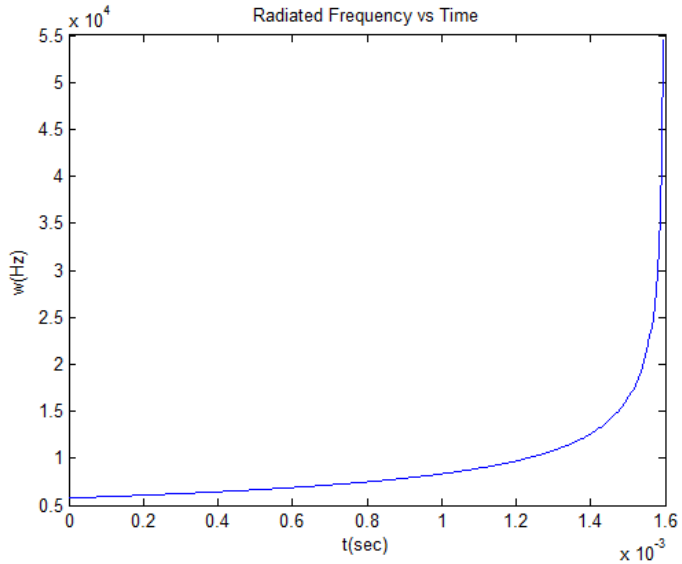


Figure 7.30: Orbital frequency as a function of time for a typical binary system.

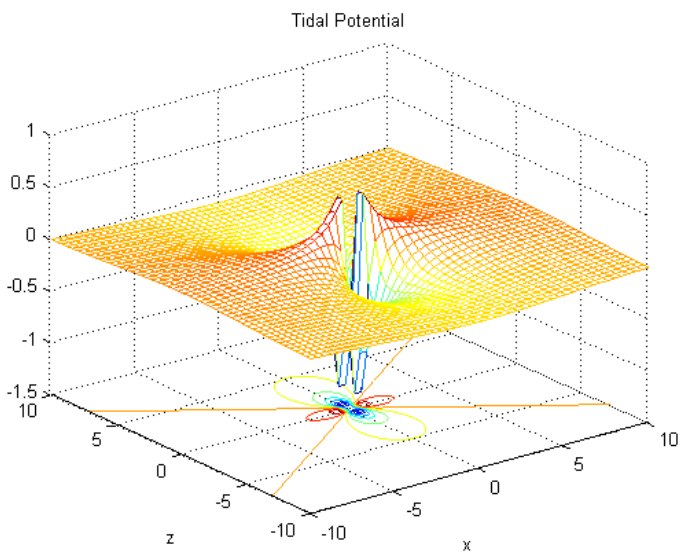


Figure 7.31: Tidal potential as a function of x and z with contours suggestive of elongation and compression.

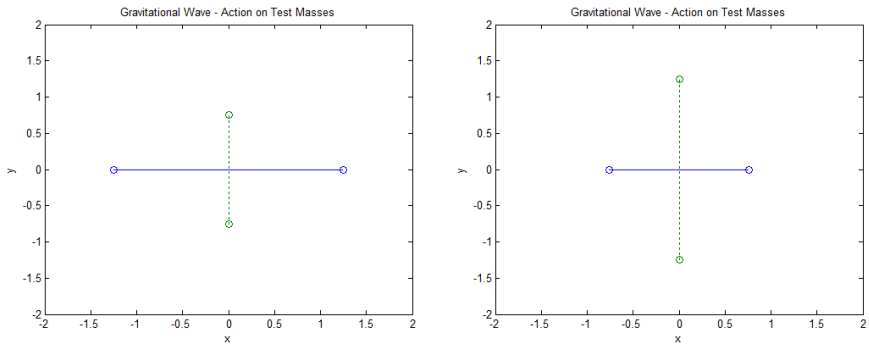


Figure 7.32: Two frames of the "movie" of the response of an "antenna" to a gravitational wave showing the wave of compression and elongation.

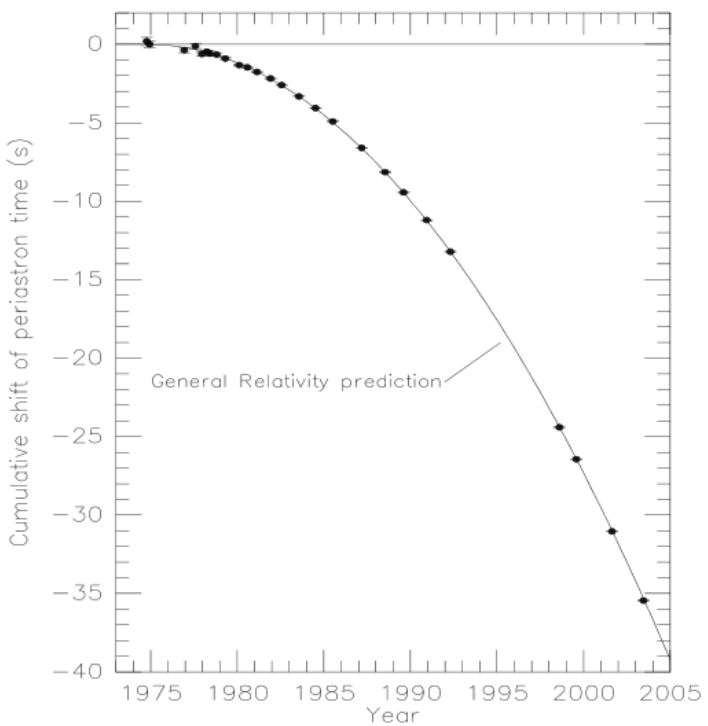


Figure 7.33: Change in the binary pulsar orbital period compared to that expected due to losses of energy from gravitational radiation.

"antenna" is due to the intrinsic gravitational tidal forces carried by the wave.

The "antenna" consists of four small test masses. Gravitational radiation is quadrupole not dipole as in electromagnetism, so that more complex arrays are needed, whereas a dipole antenna can be used for radio waves. The effect of the passing wave on the antenna is vastly exaggerated. Fractional dimensional changes of order one part in 10^{24} are more realistic and must be contemplated for the design of a successful detector.

At present gravitational radiation has not been directly detected. If compact binary chirps are to be detected, with circular frequencies of about 10 kHz, then antennae of about 200 km are needed. That size antenna array is more appropriate to a space based location and, indeed, such a facility, called LISA, has been proposed.

Gravitational radiation has been observed by watching the reduction in the period of binary pulsars with the example given showing the Nobel prize data shown in Figure 7.33. The observed reductions with time are consistent with Equation (7.19). However, direct detection would be more compelling.

Chapter 8

Astrophysics and Cosmology

"In the beginning there was nothing, which exploded."

— **Terry Pratchett**

"Learn from yesterday, live for today, hope for tomorrow. The important thing is to not stop questioning."

— **Albert Einstein**

8.1. Gravity and Clustering

Gravity is always attractive, so that ultimately gravity wins. A script "Grav_Clump" illustrates this fact by simulating a two dimensional system of particles self-interacting under gravity. There is no temperature and the particles start at rest at random locations in a box. If they encounter one another they are clustered, cease to move and become inert. This is only a very crude model, but it gives some idea of how gravitational clustering comes about. Obviously, any thermal velocity slows down clustering.

The user first chooses the number of particles in the box. Starting with two or three is instructive. Larger systems also tend to clump, and the root mean square (r.m.s.) value of the initial and final particle separation is printed. A movie of the time evolution of the system is provided, and the last frame for ten particles is shown in Figure 8.1. Eight of the particles have clumped together under mutual gravitational attraction.

Gravity always wins. Random thermal motion resists the clumping, but systems typically cool as they evolve. The time, t_G, for a low temperature gravitational system of particles to clump together starting with a mass density, ρ_o, is:

$$t_G = \sqrt{3\pi/32G\rho_o} \tag{8.1}$$

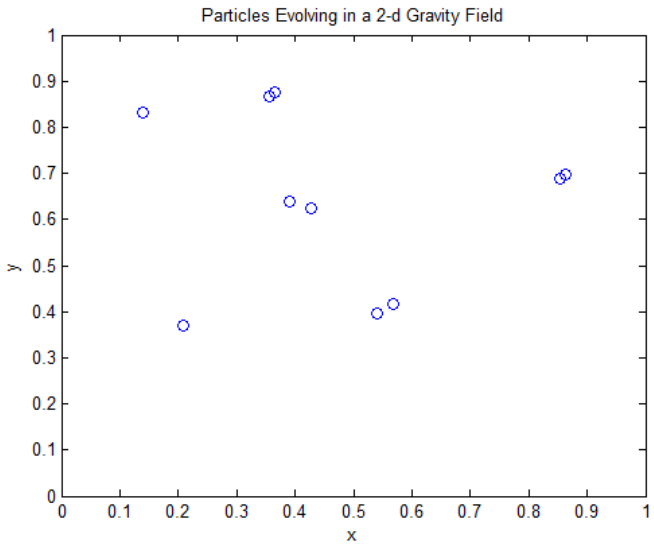


Figure 8.1: Final frame of the movie for an example of ten particles interacting gravitationally at zero temperature.

Less dense systems remain unclumped longer. That is a reflection of the fact that at higher density, more particles feel stronger gravitational forces due to the inverse square nature of the force. This fact can be approximately observed using the model with different numbers of total particles. Numerically, for a density near the present average density of the Universe, $2 \times 10^{-26}\,\mathrm{kg/m^3}$, the cluster time is about 15 billion years.

8.2. Fermi Pressure and Stars

Stars start out as clusters of protons, electrons and neutrons at high temperatures due to the gravitational binding energy rising as clusters form. The temperatures are needed to overcome the Coulomb barriers in order that the exothermic fusion reactions can occur. The first cycle of reactions with the basic protons creates helium.

$$\begin{aligned} \mathrm{p} + \mathrm{p} &\rightarrow \mathrm{H}_1^2 + \mathrm{e}^+ + \nu_\mathrm{e}, \quad 1.44\,\mathrm{MED} \\ \mathrm{H}_1^2 + \mathrm{p} &\rightarrow \mathrm{He}_2^3 + \gamma, \quad 5.49\,\mathrm{MeV} \\ \mathrm{He}_2^3 + \mathrm{He}_2^3 &\rightarrow \mathrm{He}_2^4 + \mathrm{p} + \mathrm{p}, \quad 12.85\,\mathrm{MeV} \end{aligned} \tag{8.2}$$

The initial state nuclei Coulomb repel and therefore will react rapidly only at high temperatures. As the star evolves more complex nuclei are created because the nuclear binding energy is, very roughly, 8 MeV per added neutron or proton. However, when iron is reached, heavier nuclei are no longer more deeply bound. Adding protons reduces binding because of their mutual Coulomb repulsion. To alleviate this repulsion, higher Z nuclei have more neutrons than protons. The heavier nuclei beyond iron arise from reactions in supernova explosions not stellar fusion reactions. As the saying goes, "we are stardust".

At some point, therefore, a star uses up its fuel, cools and begins to contract. Gravity is always attractive and the star gains binding energy by contracting. The star compresses, with mass density ρ, under gravity due to self-interactions which dimensionally go as the inverse radius of the star, R and the square of the mass, M, of the star. The gravitational biding energy is:

$$\begin{aligned} U_G &= -(4\pi)^2 G\rho^2 R^5 \\ &\sim GM^2/R \end{aligned} \tag{8.3}$$

The compression leads to an increase in pressure, p_G, which dimensionally goes as the inverse fourth power of the radius of the star. The notation is that U_G is the gravitational potential, and V is the volume of the star. The gravitational pressure therefore goes as the inverse 4/3 power of the volume and as the square of the number of nuclei, N.

$$p_G = \partial U_G/\partial V \sim GM^2/R^4 \sim N^2/V^{4/3} \tag{8.4}$$

Assuming that the star is burned out, the only resistance to gravitational contraction arises from the exclusion principle pressure of the fermions, either the electrons or the nucleons. The Fermi energy is proportional to the 2/3 power of the number density, as explored in Section 5.6 above. Therefore, the Fermi pressure rises more rapidly than the gravitational pressure, and an equilibrium can be achieved.

$$\begin{aligned} k_F &\sim n^{1/3} = (3\pi^2 n)^{1/3} \\ E_F &\sim k_F^2 \sim n^{2/3} \end{aligned}$$

$$\begin{aligned} U_F &\sim NE_F \sim N^{5/3}/V^{2/3} \\ p_F &= \partial U_F/\partial V \sim N^{5/3}/V^{5/3} \\ &= 2U_F/3 \end{aligned} \tag{8.5}$$

Parenthetically, why does the world appear to be solid when we know it is mostly empty space composed of atoms of angstrom size, with almost all the mass localized in a nucleus about 100,000 times smaller? The answer is that the stiffness of matter exists as a result of the Fermi Exclusion Principle. Indeed, the Fermi pressure implies a bulk modulus, $B = 5p/3$, where B is the inverse of the fractional change in volume with pressure, $1/B = -1/V(\partial V/\partial p)_T$. Numerically, B can be estimated using Equation (8.5), to be $10^{10}\,\mathrm{nt/m^2}$ for $Z = 10$, which is comparable to the measured value. It is the Fermi pressure which makes matter stiff and provides us with the illusion of solidity.

The stars burn by fusion and the radiation pressure stabilizes the radius. When the fuel is exhausted the star contracts and the contraction is resisted first by the electrons. As they become relativistic for massive stars, they are pushed into the protons and a neutron star is formed. For some total number of nucleons, N, a stable radius, R_n, exists in the balance of gravity, $1/V^{4/3}$ and the Fermi pressure of the neutrons, $1/V^{5/3}$.

$$R_n = (81\pi^2/16)^{1/3}\hbar^2 N^{-1/3}/Gm_n^3 \tag{8.6}$$

The velocity of particles near the top of the Fermi sea is $\sim m\beta c \sim \hbar k_F$:

$$\beta \sim (\pi\hbar/mc)(3^{1/6}/2^{1/2})n^{1/3} \tag{8.7}$$

U_G scales as N^2 while U_F has a weaker dependence. Thus for massive stars, under contraction n rises, the Fermi momentum rises, the particles become relativistic, and matter becomes less stiff because the energy is then proportional to momentum as in SR and not momentum squared as in the NR case.

$$\begin{aligned} p_F &= \hbar^2\pi^3(3n/\pi)^{5/3}/15\,m \\ p_F &= \hbar c n^{4/3} \end{aligned} \tag{8.8}$$

```
>> Fermi_Pressure
  Program to look at Stellar Masses and e and n Pressure


 Fermion Pressure ~ n^5/3 (NR), ~ n^4/3 (UR)
Enter the Stellar Mass in Units of the Mass of the Sun: 2
 Schwartzchild Radius = 5928.89(m)
 Hawking Temperature of Black Hole = 3.23878e-08(oK)
 If e Pressure Exceeds Gravitational = White Dwarf
 If n (e + p -> n + v) Pressure Exceeds Gravitational = Neutron Star
 If n (e + p -> n + v) Pressure Fails as n Become relativistic = Black Hole
 For M ~ a Few Solar Masses, R of Dwarf ~ 10^7 m, R of n Star ~ 10^4 m and R of a Black Hole ~ 10^3 m
```

Figure 8.2: Printout for the demonstration of Fermi pressure.

When β approaches one, the resisting matter fails and gravity wins. In this case, which occurs for stars with masses of a few solar masses, resistance crumbles and a black hole is formed. A more precise mass value is the Chandrasekhar limit which is $\sim$1.44 solar masses. The sun has a radius 7×10^8 km. An electron stabilized star will have a size about 10,000 km, while a neutron stabilized star has a size about 10 km.

The situation is explored in a script "Fermi_Pressure". The user supplies a mass in units of the solar mass. Printout for that choice appears in Figure 8.2. NR and UR refer to non-relativistic and ultra-relativistic.

The velocity for the electrons and nucleons as a function of the contracting stellar radius is shown in Figure 8.3. The light electrons, Equation (8.8), become relativistic at large radii, about 10,000 km. The neutrons, about 2000 times heavier, become relativistic at radii around 10 km.

The pressure due to gravity, resisted by the Fermi pressure, as a function of stellar radius is shown in Figure 8.4. The contributions due to electrons are sharply reduced at radii around 10,000 km, while the neutrons become ineffective at about 10 km. The * labels the Schwarzschild radius which indicates the formation of a black hole when Fermi pressure fails for stellar masses of a few solar masses.

Since this fate seems inevitable for massive stars, and since the theory of general relativity, GR, predicts the existence of gravitational waves, there is a worldwide attempt to search for these waves which should be emitted in the process of stellar collapse.

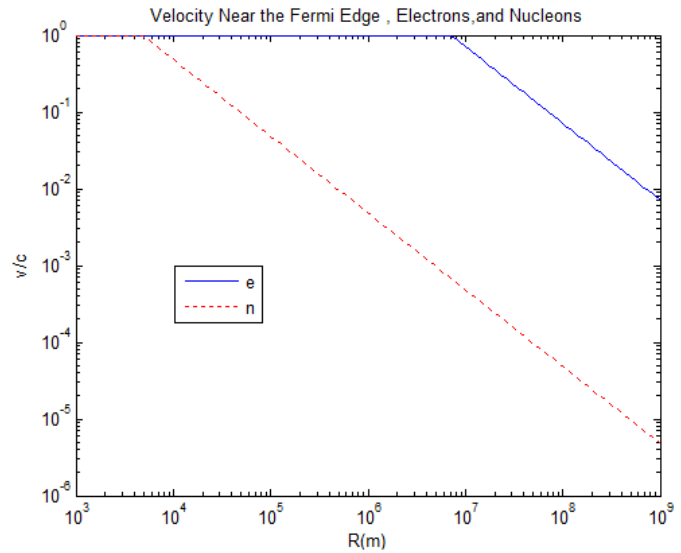


Figure 8.3: Velocity of electrons and neutrons for stars as a function of the stellar radius.

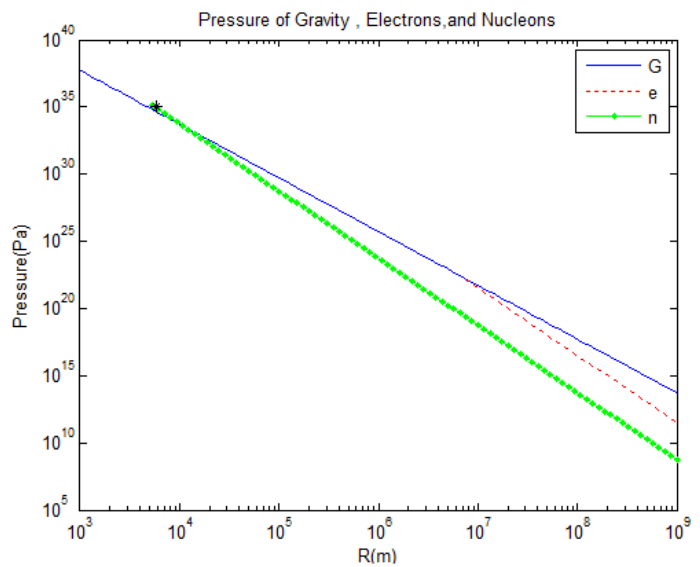


Figure 8.4: Pressure due to gravity and the opposing Fermi pressure as a function of stellar radius. Electrons can stabilize large radii, while neutrons stabilize smaller radii.

8.3. Uniform Density Star

A reasonable model of a star is a complex undertaking. The discussion of Fermi pressure above assumed a uniform density star as a first approximation, which is a crude starting point. A simple model with uniform density is calculated in the script "Star_Constant_Density". The user supplies the stellar mass. The gravitational pressure at the center of the star is then:

$$p(0) = GM/Rc^2 \tag{8.9}$$

The resulting printout is given in Figure 8.5. The solar density in this model is about that of water, which is assumed for the star in finding the radius. A star with ten stellar masses has a Schwarzschild radius of about 30 km and a radius, assuming a solar density, of 1.5 billion m. It is not relativistic as expected from the prior discussion of Fermi pressure. Assuming a core temperature to 30 million degrees confirms the fact that such a star is a NR object. The dimensionless ratio of pressure to density, $p/\rho c^2$, is small for NR matter since the rest mass then exceeds the kinetic mass due to motion.

```
>> Star_Constant_Density
  Program to look at Star with Constant Density


 Solar Mean Density = 1416.16(kg/m^3)
Enter the Stellar Mass in Units of the Mass of the Sun: 10
 Schwartzchild Radius = 29866.7(m)
 Hawking Temperature of Black Hole = 6.08184e-09(oK)
 Relativistic Star? - Radius = 1.49632e+09 m, Ratio P(o)/rho*c*c = 9.98003e-06
 Relativistic Star? - Nucleon Gas - P(o)/rho*c*c = kT/Mn*c*c = 2.67094e-06
Critical Density- when mean particle seperation~ deBroglie lambda ~ 2 x 10^9 kg/m^3
```

Figure 8.5: Printout in the case of a star of constant uniform density as a first approximation.

8.4. Stellar Differential Equations

A more complex model of a star can be attempted and compared to solar data. The next level of approximation accounts for the radial dependence of the density, mass, pressure luminosity and temperature.

The differential equations which describe such a simplified star model are:

$$\begin{aligned} dM/dr &= 4\pi r^2 \rho(r) \\ dp/dr &= -\rho(r) GM(r)/r^2 \\ dL/dr &= \varepsilon(r) dM/dr \\ dT/dr &= -3\kappa(r)\rho(r)L(r)/[16\pi r^2 acT^3(r)], \quad \text{radiative} \\ dT/dr &= (1 - 1/\gamma)T/P(dp/dr), \quad \text{convective} \end{aligned} \tag{8.10}$$

The increment in mass, $M(r)$, in a shell of radius dr is dM/dr. The increment in pressure, $p(r)$, due to the mass at that r is dp/dr. The luminosity, $L(r)$, change due to energy production, $\varepsilon(r)$, at that r is dL/dr. The energy production scales with temperature as, $\varepsilon \sim T^5$, reflecting the need for high temperatures to overcome the Coulomb barriers in fusion processes. The temperature change with r is controlled by the opacity κ for radiative temperature distribution, but by the pressure gradient for convective mixing. Convection will be assumed to dominate in this model. The pressure due to the hydrogen gas and the photon gas gives the star equation of state. The mean atomic number of the gas is μ.

$$p = \rho kT/\mu m_p + aT^4/3 \tag{8.11}$$

The user is asked for the input mass and radius of a star which are only used as integration limits. The central density and temperature are then requested as the defining inputs. The printout corresponding to using values for the sun is shown in Figure 8.6.

The mass and luminosity at $r = 0$ are taken to be zero as a boundary condition. The core temperature and density are used to derive the core pressure. The solution starts at $r = 0$ and integrates out to $r = R$ using the MATLAB script "ode45". The variables are mass, pressure, luminosity and temperature. The density is a dependent variable and is derived from the temperature and pressure using the equation of state, Equation (8.11).

The model yields a core pressure which is quite close to the solar value. The radial dependence of all five quantities are computed and compared to solar data. The agreement is not too bad considering the

```
>> Star_ODE3
  Star Models - ODE solutions, numerical integration from r = 0 out


Enter Mass in Solar Mass Units: 1
Enter Radius in Solar Radius Units: 1

 Now Solve ODE System for a Convective Hydrogen Star
Enter Central Density (kgm/m^3), Sun is 150,000 - 150000
Enter Core Temperature (10^7 K), Sun is 1.5 - 1.5
Core Pressure (kgm*m/sec^2) Due to Gas, 1.85928e+12 and Radiation 1.27575e+09
dM/dr = 4*pi*r^2*rho - M = mass < r
dP/dr = -rho*G*M/r^2 - Pressure change due to gravity
dL/dr  4*pi*r^2*rho*eps - Luminoisty change due to energy production
dT/dr = -3*kap*rho*L/(16*pi*r^2*a*c*T^3) - Temp change due to opacity kap - radiation
dT/dr = (1-1/g)T/P(dP/dr) - convective temperature gradient
P = (rho*k*T)/mu*mp + (a/3)*T^4 - eq of state, pressure due to gas and radiation
Opacity Kappa at Core (m^2/kgm) 0.0849187
Energy Production Epsilon at Core (W/kgm*sec) 0.00227813
Start at r=0, M(0)=L(0)=0, P(0)=Pc, T(0)=Tc, rho derived from P and T
```

Figure 8.6: Printout from "Star_ODE3" for a star corresponding to the sun.

approximations which were made. Much more detailed models have been made, but this simple model does reasonably well at bringing out the physics of a hydrogen star somewhat like the sun.

The five variables are plotted by the script. The computed results are compared to solar data, shown as *o*, in Figures 8.7 and 8.8. The mass distribution and the temperature distribution are approximately correct. The pressure and density shapes are not as good, while the luminosity distribution reproduces the solar distribution quite well. Although not accurate in the details, nevertheless, it is a useful exercise to attempt to simply model the stars using well understood physics concepts.

8.5. Radiation and Matter in the Universe

Moving from stars, their structure and their evolution, to still larger objects, we end by considering the entire Universe. It is perhaps astounding that the large scale properties of the Universe can be understood by applying GR to a model and using just a few experimental facts. The strangest thing about the Universe is that it is explicable!

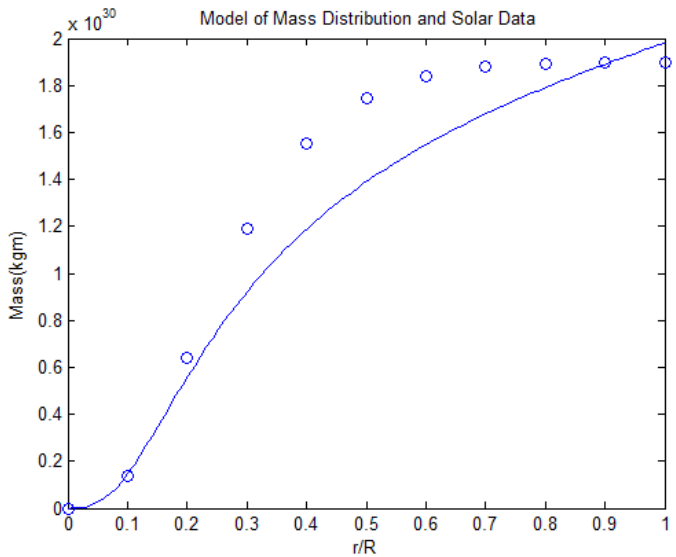


Figure 8.7: Shape of the stellar mass distribution as computed, −, and with solar data, *o*.

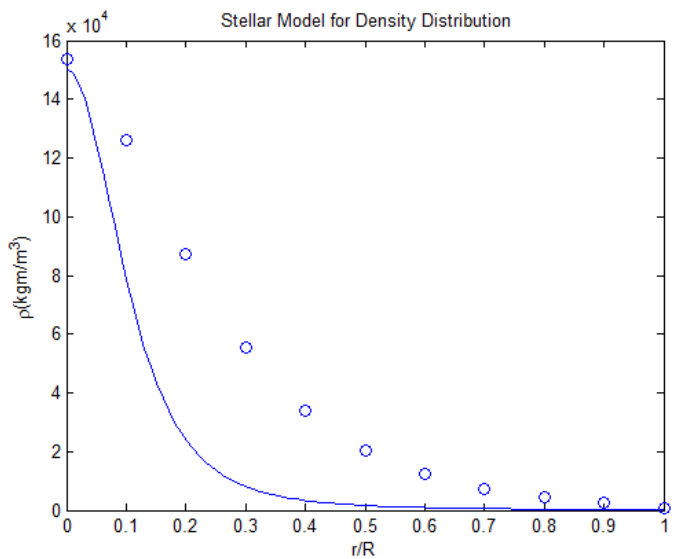


Figure 8.8: Shape of the stellar density distribution as computed, −, and with solar data, *o*.

The universe appears to have begun in a "Big Bang" at enormous energy and temperature but at zero "size". The matter and radiation expanded and cooled. At early times the radiation dominated, but it cooled as the inverse fourth power of the distance scale R, three powers for volume expansion and one for energy redshift. At present, it has a temperature of about 2.72 degrees Kelvin and the photon number density is about 410 microwave photons per cm^3. The number of photons is taken to be a constant and the adiabatic expansion leads to the redshift.

The matter density is not now relativistic and is, therefore, dominated by the rest masses of the particles. Therefore, it also falls with expansion, but only as the inverse third power of the distance scale R, due to the volume expansion of space-time as the Universe expands. Therefore, at long times, ignoring dark energy, the Universe will become matter dominated.

The Universe appears to be "flat" with a total mass density equal to the critical closure density, ρ_c. The three dimensional spatial geometry is assumed to be flat in a homogeneous and isotropic Universe. The full four dimensional geometry is curved by the matter-energy of the Universe. Under those assumptions the metric of the Universe is the flat Robertson–Walker metric, defined by the scale parameter, $R(t)$, or $ds^2 = (cdt)^2 - R^2(t)[dr^2 - (rd\Omega)^2]$.

The density is related to the current Hubble constant, H_o, where $H = (dR/dt)/R$ measures the expansion of the metrical scale factor, R. The critical density arises from considerations of the GR dynamics which relates the energy density of a model Universe, homogeneous and isotropic, to the expansion of the metrical scale R due to gravitational coupling G.

$$\rho_c = 3H_o^2/8\pi G \tag{8.12}$$

Using galactic redshifts to determine the Hubble parameter, this average density is presently about 5.6 GeV per cubic meter or about six protons. The present cosmic microwave background of red-shifted photons, CMB, is about 41,000 times less than the present critical density.

The expansion of the Universe is tracked by the change of scale factor R with time. It is measured by observing the redshift of light emitted at scale, R, and wavelength, λ. As the space-time of the Universe expands, all stars appear to be Doppler shifted with respect to any observation point. The stars are embedded in space-time and do not move with respect to it. Rather space-time expands, driven by the energy density of the Universe.

$$\begin{aligned} 1 + z &= (\lambda_o - \lambda)/\lambda \\ &= R_o/R \end{aligned} \tag{8.13}$$

The GR dynamics relating $R(t)$ to the energy density ρ, ignoring any cosmological sources, follows from Equation (8.12) and the definition of H.

$$\begin{aligned} dR/dt &= R\sqrt{8\pi G\rho/3} \\ &\rho \sim 1/R^3, R \sim t^{2/3}, \text{matter} \\ &\rho \sim 1/R^4, R \sim t^{1/2}, \text{radiation} \end{aligned} \tag{8.14}$$

The time dependence of the scale factor is different depending on whether the Universe is in a radiation or matter dominated phase. This GR prediction is the needed dynamical input. The power law solutions arise when the GR field equations relating the space-time metric to the matter in the Universe are used. The $R(t)$ solutions follow from Equation (8.14). The behavior of the density with R could already be justified heuristically as was done above.

The time dependence of the scale factor R, the mass density ρ, and the temperature T are given in Table 8.1 for the two situations which are encountered; early times when radiation dominated and later times when NR matter dominated.

The script "Cosmos_Power_Law" produces plots of the behavior of matter and radiation over the life of the Universe. The user chooses a current Hubble parameter from which that behavior is derived. Printout of the user dialogue appears in Figure 8.9.

The user makes an input of the present Hubble constant. The present density is assumed to be the critical density ("flat" Universe). The cosmic microwave background density and temperature are the

Table 8.1: Power law behavior in time of matter and radiation.

Power Law	Radiation Epoch	Matter Epoch
$R \sim t^n$	$1/2$	$2/3$
Matter Density	$-3/2$	-2
Radiation Density	-2	$-8/3$
Temperature	$-1/2$	$-2/3$

```
>> Cosmos_Power_Law2
  Cosmos - find flat space quantities in matter and radiation dominated eras


Enter present Ho in km/sec*million lyr (~30): 30
 Enter estimate for h (~0.73): 0.73
Hubble Time in byr = 13.6986
Present time in byr = 9.13242
Density at Equal Radiation and Matter in gm/cm^3 = 1.32565e-16
Temperature at Equal Density - degree K = 63189.4
Time at Equal Density - sec = 1.12092e+12
Time (in radiation dominated) at sec = 0.1
Matter Density at That Time = 4974.96
Radiation Density at That Time = 1.66562e+10
Radiation Temperature at That Time = 2.11559e+11
```

Figure 8.9: Printout of the behavior of matter and radiation over the history of the Universe.

other parameters used. The scaling with time displayed in Table 8.1 is then used.

The Hubble time for the user choice shown is about 13.7 billion years. Radiation dominated until about 35,600 years at a temperature of about 63,000 degrees Kelvin. A plot of the temperature as a function of time is shown in Figure 8.10.

The cosmic microwave background is measured today and it can be extrapolated backwards in time because the Universe is transparent to these photons up until the point where hydrogen can be ionized by the photons and then the Universe becomes opaque. Scaling the present CMB kT value of 0.23 meV to 13.6 eV, the temperature would be 163,000 degrees Kelvin. Assuming matter domination during the

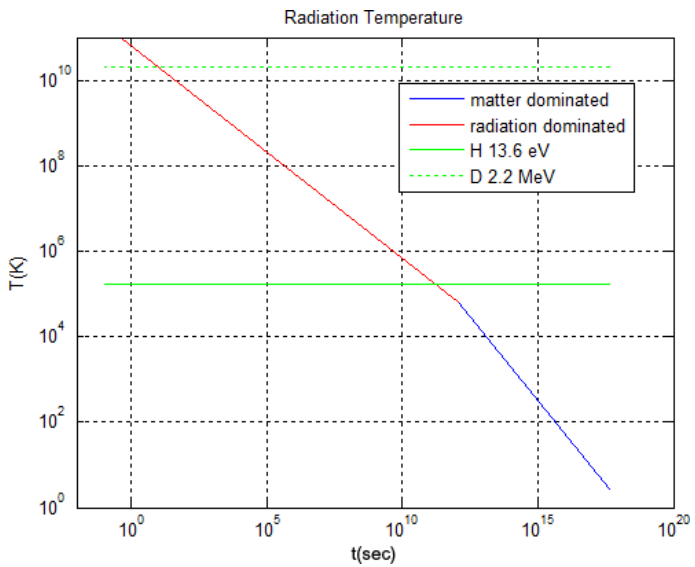


Figure 8.10: Plot of the temperature as a function of time for matter domination (blue) and radiation dominated (red) epochs.

extrapolation, the opacity sets in at approximately the point where matter and radiation have roughly equal effects on the evolution.

At earlier times the Universe is opaque to light. That is why the cosmic microwave background maps are made for about 380,000 years after the "Big Bang". At the energy scale for nuclear binding, for example, 2.2 MeV for deuterium binding, the temperature would be about 2×10^{10} degrees which would be relevant in the first few seconds. However, knowing the nuclear physics and the thermodynamics, the behavior of the Universe can be extrapolated to times which cannot be directly observed but which yield testable predictions about the elements.

The behavior of the energy density contained in the matter and radiation as a function of time is shown in Figure 8.11. As expected, matter dominates at present, assuming that there is no "dark energy" component to the energy density for now. It is amazing that a measurement of the present Hubble constant, the critical present density and the cosmic microwave background enables an extrapolation over such an enormous range of time.

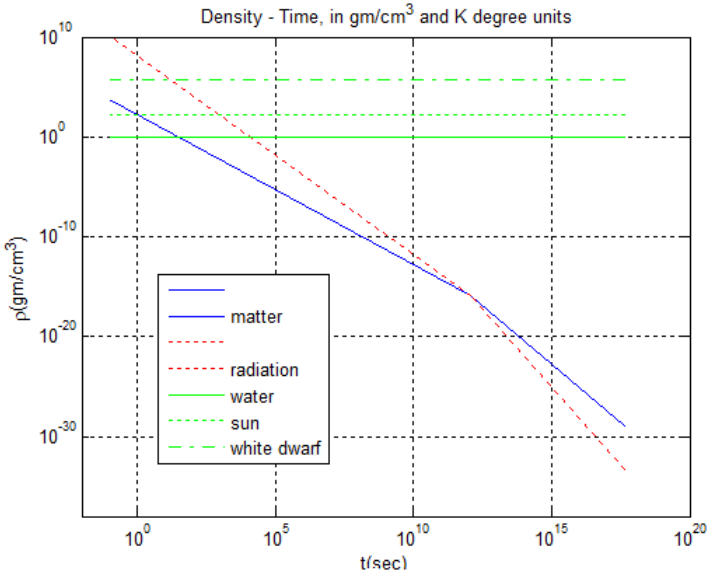


Figure 8.11: Energy density of the Universe as a function of time showing the matter and radiation components of the total energy density.

8.6. Element Abundance and Entropy

Although the Universe is opaque at short times, knowing the physics means that predictions about the abundance of the elements which were formed in fusion reactions at high temperatures can be made and compared to measurements of the primordial abundance of the nuclei. Indeed, the agreement is quite good, indicating that events with quite short times, a few seconds to minutes, after the Big Bang, can be understood and the subsequent evolution of the Universe can be explained.

With a number density of about 400 photons per cubic centimeter, radiation dominates the entropy of the Universe compared to the present six protons per cubic meter. As the Universe cools, protons and neutrons become stable particles and they, in turn, bind into nuclei. The Boltzmann distribution for the number density of non-relativistic nuclei of atomic weight A is:

$$n_A \sim T^{3/2} e^{(\mu_A - m_A)/kT} \tag{8.15}$$

The chemical potential is μ_A. The neutron to proton ratio is then fixed by the mass difference, $Q = 1.29\,\text{MeV}$, where neutrons are more massive than protons and thus less plentiful. At a "freeze out" temperature, the rate of the reactions $p \to n$ and $n \to p$ becomes less than the expansion rate, H, and the neutron and proton mixing falls out of equilibrium. Then the relic abundance, n/p, is approximately fixed.

$$n/p = e^{-Q/kT} \tag{8.16}$$

This ratio is roughly 0.17 for a 0.7 MeV freezout temperature which depends on H which sets the scale for expansion. An approximate plot of the n/p ratio as a function of temperature is shown in Figure 8.12 using the script "Cosmos_Elements". Printout made by that script appears in Figure 8.13. At high temperatures, the n/p ratio is close to one. At freezeout it is about 0.17. Note that, at lower

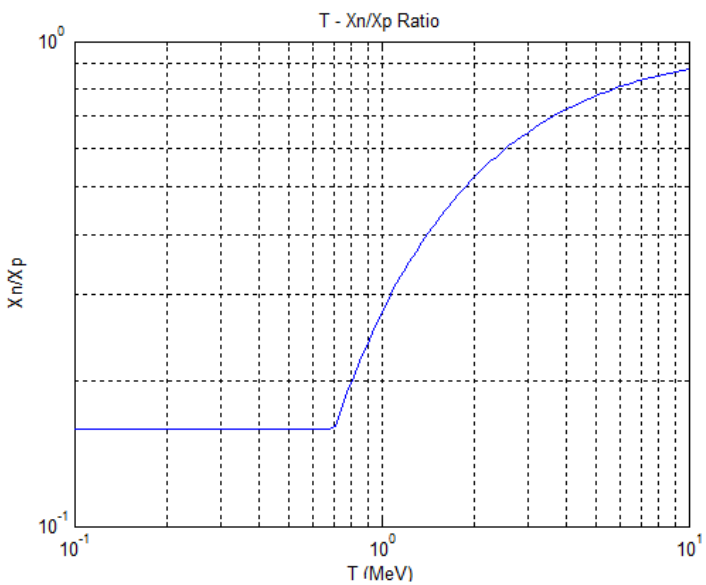


Figure 8.12: Ratio of the abundances of n to p as a function of temperature for a 0.7 MeV freeze out temperature which is the temperature below which the Hubble expansion is such that transitions between n and p cannot remain in equilibrium.

```
>> Cosmos_Elements
  nuc synth - look at big bang nucleosynthesis - only very approximately
  Kolb, Turner"The Early Universe"


Enter present Ho in km/sec*million lyr (~30): 30
 Enter estimate for h (~0.7): 0.7
Enter present baryon fraction Omegab (~0.04): 0.04
Freezeout for n occurs when expansion H is > p-> n rate
Enter neutron freezeout temp in MeV (~ 0.8 MeV): 0.7
Hot Universe and element abundance depends on ratio of B to photons ~ 5.292e-10
Elemental abundance driven by entropy and binding energy
Helium Mass Abundance ~ 0.273427
```

Figure 8.13: Printout for "Cosmos_Elements". The Helium abundance depends on H and the abundance of baryons.

temperatures, the ratio does not stay at 0.17 because the neutrons quickly go into the creation of helium.

Note that the prediction of detailed solar models is that the sun has only consumed about 0.3% of its mass over the 4.5 billion years of its existence. Therefore, the extrapolation in time to primordial abundance is not a large one and is not a source of large errors.

The light nuclei all have a binding energy B. Helium is very deeply bound, for reasons in nuclear physics similar to the "noble gas" behavior in atoms, a closed shell. Deuterium, in comparison, is very weakly bound. Since the nuclei are bound, as the temperature falls with expansion, at some point the nuclei are no longer broken apart thermally and they also "freeze out". Therefore, the abundance of light nuclei, X, $X_A = n_A A/N_n$ represents a balance between the hot photon bath, or the ratio of photons to baryons, η, and the binding energy.

$$X_A \sim T^{3/2}\eta^{A-1}e^{B/kT}X_n^{A-Z}X_p^Z \tag{8.17}$$

The fractional baryon abundance, or the X values of a nucleus A depends on the temperature, the entropy η, the binding energy, B, and the availability of the constituents, the Z protons and the $(A-Z)$ neutrons. Because of the very deep binding of ^{4}He, ($B = 28.3\,\mathrm{MeV}$) almost all the neutrons that are available after neutron freeze out go into making that element. Very approximately, $X_4 \sim 2(n/p)/[1 + (n/p)]$. The behavior of the abundance of p, deuterons ($B = 2.2\,\mathrm{MeV}$)

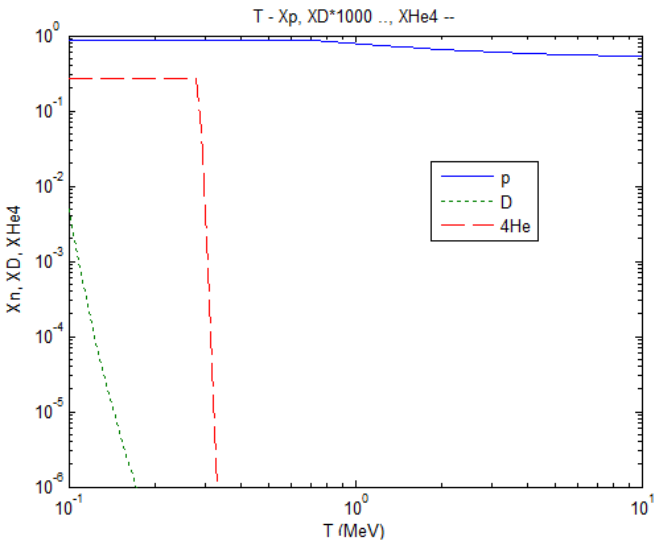


Figure 8.14: Abundances of p, deuterons, ^{2}H, and helium, ^{4}He, nuclei as a function of temperature. The n which are available after freeze out almost all go into the formation of helium nuclei because of the large helium binding energy.

and ^{4}He is shown in Figure 8.14. The deuterium abundance, with a small B, is less than that of the Helium with more constituents but very deep binding.

It appears that the fractions of light nuclei can be well understood by combining cosmology with nuclear physics. Surely, it is a great achievement to understand the Universe over a range of billions of years from first principles.

8.7. Dark Matter

Although the Universe appears to be "flat" or to have a density equal to the critical density, the origin of this matter is not known. Indeed, the observed matter of suns and other visible matter is only about 4.2% of the critical density. The photons are a very small fraction at present. It seems that there is matter with gravitational interactions that accounts for about 23% of the critical density. This is called "dark matter" and it is not known what it is composed of. The remaining 73% is called "dark energy" and no one knows

```
>> DM_Evidence
  Program to look at evidence for dark mattr


Dark Matter is About 1/4 of the Universe by Mass
Dark Matter is Inferred from Galactic Rotation Curves
Dark Matter is Inferred from Einstein Ring Sizes
Dark Matter is Searched for in Scattering Experiments
The Galaxy Has a radius of ~ 30,000 LY
The Galaxy Rotates every 200 Million Years
Velocity of Galaxy w.r.t. Dark Matter (km/sec) = 283.043
Maximum Kinetic Energy of 200 GeV DM Particle (MeV) = 0.0890145
```

Figure 8.15: Printout for dark matter evidence.

what that is. It appears to be a "cosmological term" which will be explored in the next section. It is humbling to realize that we have so far explored and understood only about 4% of the Universe.

Some of the evidence for dark matter is given in the script "DM_Evidence". The printout for that script appears in Figure 8.15.

There is a plot provided that shows a schematic of the rotation curves of a galaxy. Simple kinematics indicates that the velocity of a star within the galaxy goes linearly as the distance from the center, while outside the galaxy, the velocity should fall as the inverse of the square root of the radius. The velocities are measured by looking at the Doppler shifts of the spectral lines of stars. Equating the gravitational force to the centrifugal force:

$$F = GMm/r^2 = mv^2/r$$
$$M \sim r^3 \tag{8.18}$$

Outside the galaxy, M is a constant and the velocities decrease. What is observed is that the velocities continue to increase outside the core of visible stars. Thus, masses which are dark are inferred to extend well beyond the limits of the visible galaxy.

A second piece of evidence is the size of the Einstein rings that are a consequence of the general relativistic deflection of light by a gravity field. A schematic of the light rays is shown in Figure 8.16. Gravity bends the light through an angle $\theta_G = 4GM/bc^2$ when it

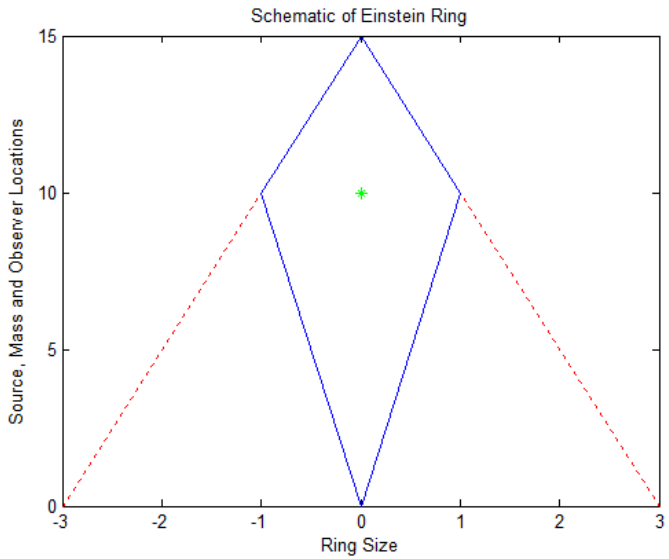


Figure 8.16: Schematic of a source which emits light, blue lines. A mass, green *, is passed which deflects the light which subsequently is observed at (0,15). The light appears to have originated at a ring source, red dashed lines. The behavior is exactly like that of an optical lens.

passes by a mass, M, with impact parameter, b. A galaxy at distance $d_s - d_l$ away from the mass has an observer at distance d_l beyond that mass. The source emits with angle θ_s which has impact parameter b when it passes the mass M. The geometry implies:

$$\begin{aligned} \theta_s &= b/(d_s - d_\ell) \\ \theta_s d_s - \theta_G d_\ell &= 0 \\ b = \theta_E d_\ell &= \sqrt{4GM/c^2[d_\ell(d_s - d_\ell)/d_s} \end{aligned} \tag{8.19}$$

The equation for the ring size should be familiar from the previous work on the focal length of lenses, in this case with a finite object and image distance.

The angular radius of the Einstein ring is θ_E which depends on the mass, M. Again, data implies that the visible mass is woefully insufficient to explain the observed ring sizes.

Given this information, many physicists are looking for dark matter on Earth. As seen in the printout, Figure 8.15, our galaxy

rotates with a velocity about 283 km/sec. Assuming that this is the velocity of dark matter with respect to a laboratory on Earth, and that dark matter consists of particles with 200,000 MeV mass, then the maximum energy that could be transferred to a detection apparatus would be about 0.09 MeV. Since minimum ionizing particles like cosmic rays deposit about 1.5 MeV/cm in light materials, it is clear that these searches are hard. In addition, the collision cross section is very small, so the collisions occur only very rarely. Nevertheless, the importance of understanding the Universe makes such an observation of fundamental importance.

8.8. Dark Energy

There is the possibility in the GR field equations of both ordinary matter as a source and a vacuum energy because there are two possible tensor sources — the energy — momentum tensor which contains matter and radiation, and the metric tensor itself. In fact, Einstein originally included a "cosmological term" in his cosmological model but later removed it. Nevertheless, the existence of "dark energy", a repulsive force, has been observed which is consistent with a vacuum energy density. Apparently the vacuum may contain cosmological energy density.

The matter density falls as the inverse cube of the scale R, while the vacuum density, proportional to the metric itself, is a constant. Therefore, ultimately the vacuum energy will dominate as the Universe cools and expands and it will drive the Universe to an exponential decrease in density.

In the presence of both matter and vacuum, or dark, energy the scale R has a time dependence:

$$\begin{aligned}(dR/dt)^2 &= 8\pi G\rho R^2/3 + \Lambda R^2/3 \\ \rho_m &= \rho_m^o (R/R_o)^3, \Omega = \rho/\rho_c = 1 \\ H(t) &= H_o[(R_o/R)^3(1-\Omega_v) + \Omega_v]^{1/2}\end{aligned} \tag{8.20}$$

Equation (8.20) is the Friedmann equation for the special case of a flat Universe, where a term kc^2 would appear if the curvature k were non-zero. The contribution of matter/energy to the expansion

rate is controlled by ρ. The Universe is "flat", with a fraction of the critical density due to matter which is just $1 - \Omega_v$.

The limit where the vacuum energy is small has t going as the 3/2 power of R in a matter dominated Universe. If the vacuum energy dominates, the scale R increases exponentially with an exponent proportional the square root of the vacuum density.

In the general case, the solution of Equation (8.20) for the coordinate time t is:

$$\begin{aligned} H_o t &= 2/3\sqrt{\Omega_v}\log[\sqrt{1+(R/R_o)^3/a}+\sqrt{(R/R_o)^3/a}] \\ a &= \Omega_v/(1-\Omega_v) = \Omega_v/\Omega_m \end{aligned} \tag{8.21}$$

This result is evaluated with the script "Cosmos_Vacuum2". The user supplies a present Hubble constant and a value for the vacuum energy ratio to the critical energy density. The $H(t)$ and $R(t)$ functions are then computed and plotted. The printout is shown in Figure 8.17.

The plot of Figure 8.18 shows the time dependence of the Hubble "constant". In the absence of dark energy H will fall with R as seen in Equation (8.20), $H = (dR/dt)/R \sim \sqrt{8\pi G\rho} \sim R^{-3/2}$. Dark energy, in contrast, has a Hubble constant, $H = \sqrt{\Lambda/3}$, which is a constant.

The plot of Figure 8.19 shows the scale R as a function of time in an initially matter dominated Universe. At short times the scale evolves as the 2/3 power of t. At later times, where the crossover is shown as a *, the scale begins to increase exponentially since the vacuum energy dominates. In Equation (8.20), the vacuum dominated Hubble parameter would goes as $dR/R = \sqrt{\Lambda/3}dt$, $R \sim e^{\sqrt{\Lambda/3}t}$. If

```
>> Cosmos_Vacuum2
  now look at flat space with a mixture of matter and vacuum energy
  supernova + CMB wiggles indicate omegav ~ 0.71, omegam ~ 0.04, dm = 0.25


Enter present Ho in km/sec*million lyr (~30): 30
 Enter estimate for h (~0.73): 0.73
 Enter Value for Vacuum Omega = (0,1): 0.7
 Hubble Time, in byr = 13.6986
```

Figure 8.17: Printout for an example of the evaluation of the effect of vacuum energy.

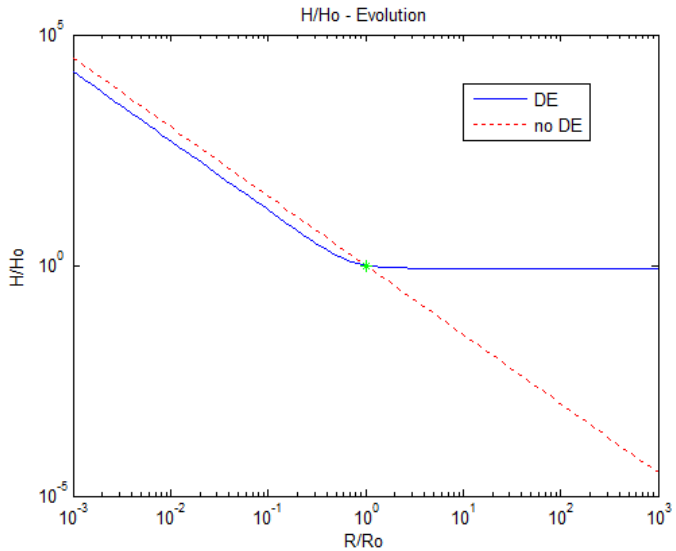


Figure 8.18: Hubble parameter as a function of the scale factor R with dark energy and without.

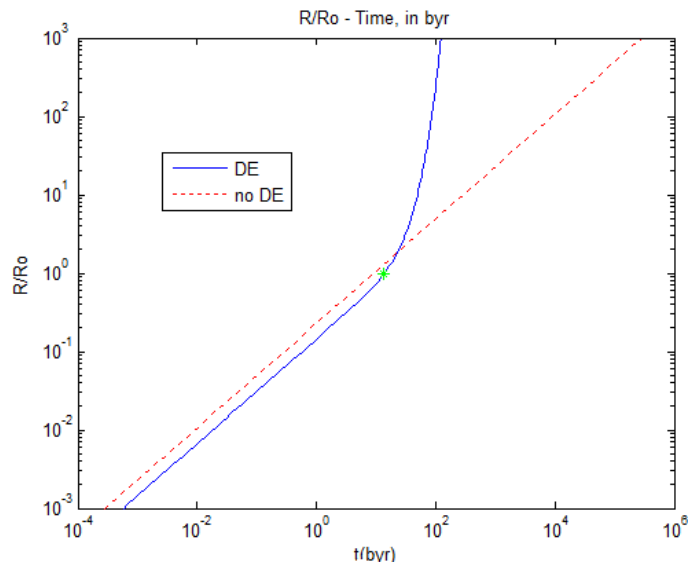


Figure 8.19: The scale R as a function of time in an initially matter dominated Universe. The power law behavior at short times is due to matter domination, while the exponential behavior at long times is driven by the vacuum energy.

the observed dark energy is indeed a cosmological constant, then the fate of the Universe is to suffer a "Big Stretch" after starting in a Big Bang.

In a related topic, the Universe is now conjectured to have undergone a rapid era of "inflation" with an exponential increase in the scale R which smoothed out the distributions at early times. The physical agent of that mechanism remains unknown, in contrast to our present knowledge of the nuclear physics and particle physics which explains somewhat later times. Nevertheless, the time "frontier" continues to be pushed both earlier and later.

Appendix — Script for Classical Mechanics

All the scripts are available to the user using the enclosed media. However, it is useful to be able to quickly jump to a written version in order to see what MATLAB commands are used. To that end, the script text for the section on Classical Mechanics is enclosed below.

2.1. Simple Harmonic Oscillator

```
%
% Program to compute oscillations - single spring, damped and driven
%
clear; help cm_osc; % Clear memory and print header
%
% spring (k,m) mass and spring constant
%
syms k m x t y yd B w
%
y = dsolve('D2x=-k*x/m','x(0)=A','Dx(0)=0'); % free oscillation
%
fprintf(' SHM, Initial Position = A, No Initial Velocity \n')
pretty(y)
%
% now a damped oscillation
%
yd = dsolve('D2x=-k*x/m-b*Dx/m','x(0)=A','Dx(0)=0');
%
fprintf(' SHM - Damped with Amplitude b, Initial Position = A, No Initial Velocity \n')
%pretty(yd);
%
```

```
ydr = dsolve('D2x=-k*x/m-b*Dx/m+B*cos(w*t)','x(0)=A','
Dx(0)=0');
%
fprintf(' SHM - Damped with Amplitude b, Driven with Amplitude
B, Frequency w \n')
%pretty(ydr);
%
% now numerical evaluations
%
irun = 1;
iloop = 0;
%
while irun > 0
   kk = menu('Pick Another Driven, Damped Spring?','Yes','No');
   if kk == 2
      irun = -1;
      break
   end
   if kk == 1
      %
      %kkk = input('Enter Spring Constant k: ');
      %mm = input('Enter Mass on Spring m: ');
      %wo = sqrt(kkk ./mm);
      wo = 1;
      fprintf('Spring Natural Frequency = %g \n',wo);
      AA = input('Enter Initial Displacement A: ');
      bb = input('Enter Damping Coefficient b/m: ');
      %
      gam = bb ./2.0;
      wnatsq = (wo .^2 - gam .^2);
      if wnatsq < 0
         fprintf('Overdamped \n')
      else
         wnat = sqrt(wnatsq);
         fprintf(' Underdamped Oscillation Frequency = %g \n',wnat);
      end
```

```
%
tt = linspace(0,10,100);
k = 1;
m = 1;
A = AA;
b = bb;
for i = 1:100
  t = tt(i);
  yy(i) = eval(y);
  yyd(i) = eval(yd);
end
%
iloop = iloop + 1;
figure(iloop)
for j = 1:length(tt)
  plot(tt(j),real(yy(j)),'bo',tt(j),real(yyd(j)),'r*:')
  title(' Spring Motion, Undamped and Damped')
  xlabel('t')
  ylabel('x')
  legend('undamped','damped')
  axis([0 10 -1 1])
  pause(0.1);
end
plot(tt,real(yy),'b-',tt,real(yyd),'r:')
title(' Spring Motion, Undamped and Damped')
xlabel('t')
ylabel('x')
legend('undamped','damped')
%
BB = input('Enter Driving Amplitude B: ');
ww = input('Enter Driving Frequency \omega: ');
wressq = wo .^2 - (bb .^2) ./2.0;
wres = sqrt(wressq);
fprintf('Driven Resonant Frequency = %g \n',wres);
w = ww;
B = BB;
```

```
for i = 1:100
  t = tt(i);
  yydr(i) = eval(ydr);
end
%
iloop = iloop + 1;
figure(iloop)
plot(tt,real(yyd),tt,real(yydr),':')
title('Damped Spring Motion, Undriven and Driven')
xlabel('t')
ylabel('x')
legend('undriven','driven')
%
www = linspace(0,2 .*wo,25);
%
B = BB;
for j = 1:25
  w = www(j);
  for i = 1:100
    t = tt(i);
    yyydr(i) = eval(ydr);
  end
  ydrmx(j) = max(yyydr);
end
iloop = iloop + 1;
figure(iloop)
plot(www,abs(ydrmx))
title(' Damped Spring Motion, Max Amplitude vs Driving
Frequency')
xlabel('\omega')
ylabel('x')
hold on
plot(wo,AA,'ro',abs(wnat),AA,'b*',abs(wres),AA,'g+',abs(wo-
gam),AA+0.5,'r+',abs(wo+gam),AA+0.5,'r+')
ydrvmx = max(abs(ydrmx));
ymaxplt = AA+1;
```

```
    if ydrvmx > AA+1;
      ymaxplt = ydrvmx;
    end
    axis([min(www),max(www),0.,ymaxplt])
    %
    hold off
    legend('max x','\omegao','\omegadamped','\omegares',
    'ResWidth')
    %
  end
end
%
```

2.2. Coupled Pendula

```
%
% Program to compute coupled simple harmonic motion
%
clear;
help cm_2sho;
%
% Clear memory and print header
%
% for 2 pendula both (k,m) coupled by a spring (k12)
%
syms xx x1 x2 k m k12 A1 A2 X1 X2
%
fprintf('2 pendula with (k,m) and Coupling k12, Solution with Initial
Amplitude But No Velocity \n')
fprintf('D2x1=(-k*x1-k12*(x1-x2))/m,
D2x2=(-k*x2+k12*(x1-x2))/m \n')
%
[X1, X2] = dsolve('D2x1=(-k*x1-k12*(x1-x2))/m','D2x2=(-k*x2+
k12*(x1-x2))/m','x1(0)=A1','Dx1(0)=0','x2(0)=A2','Dx2(0)=0');
%
% symbolic solution for the 2 displacements
```

```
%
%pretty(X1)
%pretty(X2)
%
iloop = 0;
irun = 1;
while irun > 0
  %
  krun = menu('Another Set of Parameters to Solve?','Yes','No');
  if krun == 2
    irun = -1;
    break
  end
  %
  if krun == 1
    iloop = iloop + 1
    mm = input('Enter Equal Masses: ');
    kk = input('Enter Equal Spring Constants: ');
    kk12 = input('Enter 1-2 Spring coupling: ');
    AA = input('Enter Initial Displacements of the 2 Springs - [A(1),
    A(2)]: ');
    m = mm;
    k = kk;
    k12 = kk12;
    A1 = AA(1);
    A2 = AA(2);
    tt = linspace(0,10,100);
    for i=1:100
      t = tt(i);
      xxx1(i) = real(eval(X1));
      xxx2(i) = real(eval(X2));
    end
    %
    figure(iloop)
    plot(tt,xxx1,'b-',tt,xxx2,'r-')
    title('Spring Coupled Motion of 2 Pendula')
```

```
xlabel('t')
ylabel('x_1, x_2')
legend('x_1','x_2')
%
iloop = iloop + 1;
figure(iloop)
plot(tt,xxx1-xxx2,'b-',tt,xxx1+xxx2,'r-')
title(' \omega for x_1+x_2=sqrt(k/m), \omega for x_1-x_2 =
sqrt(k+2k12)/m)')
xlabel('t')
ylabel('x_1+- x_2')
legend('x_1-x_2','x_1+x_2')
%
iloop = iloop + 1;
figure(iloop)
x1max = max(xxx1);
x2max= max(xxx2);
x1min = min(xxx1);
x2min = min(xxx2);
xmin = x1min;
if x2min < x1min
  xmin = x2min;
end
xmax = x1max;
if x2max > x1max
  xmax = x2max;
end
xp1(1) = 0;
yp1(1) = 1;
yp1(2) = 0;
xp2(1) = 0;
yp2(1) = 1;
yp2(2) = 0;
xc = 0;
yc = 1;
%
```

```
    for i=1:100
      xp1(2) = xxx1(i);
      xp2(2) = xxx2(i) ; %+ 5;
      plot(xxx1(i),0,'o',5 + xxx2(i),0,'o')
      hold on
      plot(xp1,yp1,'r-',xp2+5,yp2,'g-',xc,yc,'-')
      xcou(2) = xxx2(i)+5; ycou(1) = 0;
      xcou(1) = xxx1(i); ycou(2) = 0;
      plot(xcou,ycou,'–')
      title('x_1 and x_2 Movie in Time')
      xlabel('x_1 x_2')
      axis([xmin xmax+5 -0.5 1]);
      pause(0.1)
      hold off
    end
%
  end
end
%
```

2.3. Triatomic Molecule

```
%
% Program to symbolically solve ODE for linear molecule - 3 masses,
2 springs
% k/m = 1, (m/M) ratio = b, outer atoms have mass m, central
atom has M
%
clear all;
help cm_triatomic % Clear the memory and print header
%
syms x1 x2 x3 b t w Aw y bb x xy
%
% now exactly solve the eqs of motion, with arbitrary initial posi-
tions, 0 initial velocities
%
```

```
% Initialize - get Differential Eq to solve, x are w.r.t. equilibrium
% positions
%
[x1, x2, x3] = dsolve('D2x1 = (x2-x1),D2x2 = b*(-2*x2+x1+x3),
D2x3=(x2-x3)', .....
        'x1(0)=x(1),x2(0)=x(2),x3(0)=x(3)',....
        'Dx1(0)=0,Dx2(0)=0,Dx3(0)=0');
%
fprintf('Symbolic Solution Initial Velocities = 0, Initial Positions x(i)
\n')
%
x1=simple(x1);
%pretty(x1)
x2=simple(x2);
%pretty(x2)
x3 = simple(x3);
%pretty(x3)
%
% use MATLAB tools to find eigenvalues , y = w^2
%
fprintf(' Aw is the Oscillation Matrix for the 3 Atoms, y = w^2 \n')
Aw = [1-y, -1, 0; -1, 2-y/bb, -1;0, -1, 1-y]
xy = det(Aw);
%
% use eigen tools on MATLAB
%
fprintf('Eigenfrequencies = 0, 1, and sqrt(1+2b) \n')
fprintf(' The Determinant of Aw has Roots y = w^2 of the Eigen-
frequencies in sqrt(k/m) Units \n')
factor(xy)
%
iloop = 0;
irun = 1;
while irun > 0
   %
   krun = menu('Another Molecule?','Yes','No');
```

```
if krun == 2
  irun = -1;
  break
end
%
if krun == 1;
  iloop = iloop + 1
  x = input('Enter initial displacements [x(1) x(2) x(3)]: ')
  fprintf('Initial Velocities are Zero \n')
  mm = input('Enter Ratio of Small Outer Masses,m, to Inner
  Mass,M: ');
  b = mm;
  %
  tt = linspace(0,10);
  %
  for i=1:100
    t = tt(i);
    X1(i) = eval(x1);
    X2(i) = eval(x2);
    X3(i) = eval(x3);
  end
  %
  figure(iloop)
  plot(tt,real(X1),tt,real(X2),':',tt,real(X3),'-.')
  title('Motion of the Three Masses, time in 1/\omega Units of
  Outer Masses')
  xlabel('time')
  ylabel('displacement')
  legend('x1','x2','x3')
  %
  iloop = iloop + 1;
  figure(iloop)
  for i=1:100
      plot(real(X1(i))-10,0.,'o',real(X2(i)),0.,'*',
      real(X3(i))+10,0.,'o')
      axis([-15 15 -1 1])
```

```
        title(' Movie of Motion of the Three Masses')
        xlabel('displacement')
        hold on
        xcou(1) = real(X1(i))-10; ycou(1) = 0;
        xcou(2) = real(X2(i)); ycou(2) = 0;
        xcoup(2) = real(X3(i))+10; ycoup(2) = 0;
        xcoup(1) = real(X2(i)); ycoup(2) = 0;
        plot(xcou,ycou,'-',xcoup,ycoup,'-')
        hold off
        pause(0.1)
    end
  end
end
%
```

2.4. Scattering Angle and Force Laws

```
%
% Program to compute the trajectory for scattering of diffferent
Force Laws
% use MATLAB ode
%
function Scatt_Force_Law
%
clear;
help Scatt_Force_Law; % Clear memory and print header
%
global iforce qq
%
% menu
%
fprintf(' Energy and Mass Defined = 1 \n')
%
irun = 1;
iloop = 0;
%
```

```
while irun > 0
  kk = menu('Pick Another Force Law?','Yes','No');
  if kk == 2
    irun = -1;
    break
  end
  if kk == 1
    %
    iforce = menu('F(r) = 1/r^n','n=1','n=2','n=3','n=4');
    qq = menu(' Repel/Attract?','Attractive ', 'Repulsive');
    if qq == 1
      qq = -1; % attractive
    end
    if qq == 2
      qq = 1; % repulsive
    end
    %
    E = 1;
    m = 1;
    vo = sqrt((2.0 .*E) ./m); % units so initial velocity = sqrt(2)
    b = linspace(0.4, 4, 10); % impact parameter
    iloop = iloop + 1;
    jloop = iloop;
    tspan = linspace(0,20,50);
    N = length(tspan);
    xc(1) = 0;
    yc(1) = 0;
    %
    % protect for attractive and central forces
    if qq == -1 && iforce > 2
      b = linspace(1.0, 6.6, 10);
    end
    for ii = 1:10
      [t,y] = ode45(@impact,tspan,[vo -10 0 b(ii)]); % initial vx =
      vo, vy = 0, x = -10, y = b
      %
```

```
        xx = y(:,2);
        yy = y(:,4);
        for jj = 1:N
          figure(jloop)
          plot(xx(jj),yy(jj),'o',xc,yc,'*')
          title('Trajectory of Scattering for This Force Law')
          xlabel('x')
          ylabel('y')
          if qq == -1
          axis([-15 20 -20 5])
          end
          if qq == 1
          axis([-15 20 0 20])
          end
          pause(0.05)
        end
        costheta = y(N,2) ./sqrt(y(N,2) .^2 + y(N,4) .^2); % scattered
        angle
        theta(ii) = acos(costheta);
        figure(jloop + 1)
        plot(xx,yy,'-',xc,yc,'*')
        title('Trajectory of Scattering for This Force Law')
        xlabel('x')
        ylabel('y')
        hold on
      end
      hold off
      iloop = iloop + 2;
      figure(iloop)
      plot(b, theta,'-')
      title('Scattering Angle vs. Impact parameter')
      xlabel('b')
      ylabel('\theta')
      %
    end
end
```

```
%
%- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
function dy = impact(t,y)
global iforce qq
%
dy = zeros(4,1);
r = sqrt(y(2) .^2 + y(4) .^2);
if iforce == 1
     fr = qq .*1.0 ./r .^1.0;
end
if iforce == 2
     fr = qq .*1.0 ./r .^2.0;
end
if iforce == 3
     fr = qq .*1.0 ./r .^3.0;
end
if iforce == 4
     fr = qq .* 1.0 ./r .^4.0;
end
dy(1) = (y(2) ./r) .*fr;
dy(3) = (y(4) ./r) .*fr;
dy(2) = y(1);
dy(4) = y(3);
%
```

2.5. Classical Hard-Sphere Scattering

```
%
% Program to solve 2 body NR collsions. Target at rest. No decays
%
clear all;
help cm_NR_scatt % Clear the memory and print header
%
% Initialize - Setup Momentum and Energy conservation
% 0 + T -> 1 + 2 but non-relativistic so mo = 1 = m1, elastic only
% assume o velocity is in +x, T is at rest
```

```
%
fprintf('Non_Relativistic Elastic Scattering, Incident Mass = 1,
Target Mass Variable \n')
%
% now some numerical plots
%
irun = 1;
iloop = 0;
while irun > 0
   kk = menu('Pick Another Target Mass?','Yes','No');
   if kk == 2
      irun = -1;
      break
   end
   if kk == 1
      %
      u = input('Enter Target Mass: ');
      %
      % loop on scattering angle of recoiling target
      % find recoil velocity and scattered projectile angle and velocity
      %
      for i = 1:100
         cph(i) = i ./101; % recoil angle
         sph(i) = sin(acos(cph(i)));
         %
         % graphics for in line collision
         %
         v2(i) = (2.0 .*cph(i)) ./(1.0 + u); % recoil velocity
         fact1 = 1.0 + v2(i) .^2 - 2.0 .*v2(i) .*cph(i);
         v1(i) = sqrt(fact1); % scattered projectile velocity
         st(i) = (v2(i) .*sin(acos(cph(i)))) ./v1(i); % scattered projec-
         tile angle
         ct(i) = cos(asin(st(i)));
         v1y(i) = v1(i) .*st(i);
         v1x(i) = v1(i) .*ct(i);
         v2y(i) = -v2(i) .*sph(i);
```

```
  v2x(i) = v2(i) .*cph(i);
end
%
iloop = iloop + 1;
figure(iloop)
plot(cph,v2,'-');
title('Velocity Of Outgoing Target w.r.t. Incoming Velocity vs.
Outgoing Angle')
xlabel('cos\phi')
ylabel('velocity')
%
iloop = iloop + 1;
figure(iloop)
plot(ct,v1,'-');
title('Velocity Of Outgoing Projectile w.r.t. Incoming Velocity
vs. Scattering Angle')
xlabel('cos\theta')
ylabel('velocity')
%
iloop = iloop + 1;
figure(iloop)
plot(v1,v2,'-')
title('Velocity of Projectile vs. Velocity of Target')
xlabel('v projectile')
ylabel('v target')
%
iloop = iloop + 1;
for i = 1:10
  figure(iloop);
  %
  % incident projectile
  %
  xp(1) = -1;
  yp(1) = 0;
  xp(2) = 0;
  yp(2) = 0;
```

```
        xsp(1) = xp(2);
        ysp(1) = yp(2);
        j = i .*10;
        xsp(2) = v1x(j);
        ysp(2) = v1y(j);
        plot(xsp,ysp,'r:')
        xst(1) = xp(2);
        yst(1) = yp(2);
        xst(2) = v2x(j);
        yst(2) = v2y(j);
        plot(xp,yp,'b-',xsp,ysp,'r:',xst,yst,'g-.')
        title('Scattering for 10 Representative Angles')
        axis([-1.2 1.2 -1 1]);
        xlabel('x Component of Velocity')
        ylabel('y Component of Velocity')
        legend('projectile','scatt proj','recoil tar')
        pause(1);
      end
    end
    %
end
%
```

2.6. Ballistics and Air Resistance

```
%
% Program to compute the trajectory of a projectile with air
resistance
%
clear;
help cm_ballis_sym; % Clear memory and print header
%
syms g k x y t vo alf ax ay p q pp qq ttt
%
% eqs of motion of projectile falling under gravity
%
```

```
fprintf('Projectile Motion, Air Resistance - Acceleration = k*dy/dt
\n');
fprintf(' Air resistance k (sec^-1), Initial Angle/Velocity alf, vo - x(t)
and y(t) \n');
%
p = dsolve('D2x+k*Dx=0','Dx(0)=vo*cos(alf)','x(0)=0');
pp = dsolve('D2x=0','Dx(0)=vo*cos(alf)','x(0)=0');
pretty(p)
fprintf('x(t) With No Resistance \n')
pretty(pp)
%
q = dsolve('D2y+k*Dy+g=0','Dy(0)=vo*sin(alf)','y(0)=0');
qq = dsolve('D2y+g=0','Dy(0)=vo*sin(alf)','y(0)=0');
pretty(q)
fprintf('y(t) With No Resistance \n')
pretty(qq)
%
% terminal velocity
%
fprintf('y Velocity With Air Resistance \n');%
ttt = diff(q,t);
pretty(ttt)
%
gg = 9.8; % MKS units m/sec^2
kkk = 0.1; % has 1/T units
%
irun = 1;
iloop = 0;
%
while irun > 0
   kk = menu('Pick Another Initial Velocity and Angle?','Yes','No');
   if kk == 2
      irun = -1;
      break
   end
   if kk == 1
```

```
%
syms g k x y t vo alf ax ay p q pp qq ttt
fprintf('Projectile Motion, Air Resistance - Initial Velocity vo
\n');
%
voo = input('Enter Initial Projectile Velocity (m/sec): ');
%
aa = input('Enter Initial Projectile Angle (deg): ');
%
aa = (aa .*2 .*pi) ./360.0;
%
% pick max time from no resistance case
%
tt = linspace(0, (2.0 .*voo .*sin(aa)) ./gg);
%
p = dsolve('D2x+k*Dx=0','Dx(0)=vo*cos(alf)','x(0)=0');
pp = dsolve('D2x=0','Dx(0)=vo*cos(alf)','x(0)=0');
q = dsolve('D2y+k*Dy+g=0','Dy(0)=vo*sin(alf)','y(0)=0');
qq = dsolve('D2y+g=0','Dy(0)=vo*sin(alf)','y(0)=0');
alf = aa;
%
g = gg;
k = kkk;
vo = voo;
%
for i=1:100
   t = tt(i);
   xxx(i) = eval(p); % resistance dxdt
   yyy(i) = eval(q); % resistance dydt
   if yyy(i) < 0;
      yyy(i) = 0;
   end
end
%
for i=1:100
   t = tt(i);
```

```
  Xxx(i) = eval(pp); % free fall
  Yyy(i) = eval(qq);
  if Yyy(i) < 0
    Yyy(i) = 0;
  end
end
%
iloop = iloop + 1;
%
figure(iloop)
plot(tt,xxx,tt,Xxx,':')
title('x as a function of t, with and without air resistance')
xlabel('t(sec)')
ylabel('x(m)')
%
iloop = iloop + 1;
figure(iloop)
plot(tt,yyy,tt,Yyy,':')
title('y as a function of t, with and without air resistance')
xlabel('t(sec)')
ylabel('y(m)')
%
iloop = iloop + 1;
figure(iloop)
jj = length(xxx);
xmax = max(Xxx);
ymax = max(Yyy);
for i = 1:jj
  plot(xxx(i),yyy(i),'o',Xxx(i),Yyy(i),'*')
  title('x as a function of y, with and without air resistance')
  xlabel('x(m)')
  ylabel('y(m)')
  pause(0.1)
  axis([0, xmax, 0, ymax])
  hold on
end
```

```
    hold off
    plot(xxx,yyy,'-',Xxx,Yyy,':')
    title('x as a function of y, with and without air resistance')
    xlabel('x(m)')
    ylabel('y(m)')
    legend('Air Resist','No Resist')
    %
  end
end
```

2.7. Rocket Motion — Symbolic

```
%
% Solve non-relativistic rocket, symbolically - no friction or forces
%
clear all;
help cm_rocket_sym % Clear the memory and print header
%
% solve the rocket equation - free of forces
%
fprintf('Solve d2y/dt2 = vo /(T-t), vo = exhaust velocity w.r.t.
rocket, T is Burn Time = mo/dmdt \n')
%
vs = dsolve('Dy -vo/(T-t)','y(0)=0');
ys = dsolve('D2y -vo/(T-t)','y(0)=0','Dy(0)=0');
v =simple(vs);
v
y = simple(ys);
y
%
fprintf('Final Velocity = vo*ln(mo/mp), mp = Payload Mass - Works
with Multi-Stage Analysis \n')
%
fprintf('Solve With Rocket in a Uniform Gravity Field - g \n')
%
vg = dsolve('Dy -vo/(T-t) + g','y(0)=0');
```

```
yg = dsolve('D2y -vo/(T-t) + g','y(0)=0','Dy(0)=0');
v =simple(vg);
v
y = simple(yg);
y
%
% go back to the simple rocket with no forces and make plots
% total possible burn time T is mo/(dm/dt) = 10000
% payload ratio mp/mo = 1-tp/T, tp = burn time for this
payload
%
fprintf('Numerical Results: total possible burn time = T \n')
fprintf('Payload ratio mp/mo => Payload burn time tp = T(1-
mp/mo) \n')
fprintf('vo = exhaust velocity, acceleration in vo/T units, velocity in
vo units \n')
fprintf('Distance at the end of payload burn in voT units \n')
%
%
irun = 1;
iloop = 0;
%
while irun > 0
  kk = menu('Pick Another Payload Ratio?','Yes','No');
  if kk == 2
    irun = -1;
    break
  end
  if kk == 1
    %
    mpmo = input('Input the Payload Ratio: ');
    tpT = 1.0 - mpmo; % burn time for this payload
    tt = linspace(0,tpT);
    acel = 1.0 ./(1.0-tt); % acceleration in vo/T units
    vel = log(1.0 ./(1.0-tt)); % velocity in vo units
    dis = (1.0-tt) .*(1.0 ./(1.0-tt) -1.0 - log(1.0 ./(1.0-tt))); %
    distance in voT units
```

```
    %
    iloop = iloop + 1;
    figure(iloop)
    semilogy(tt,acel,'-')
    title('Rocket - Acceleration in vo/T units')
    xlabel('Time in Total Possible Burn Time Units from 0 to
    Payload Burn Time')
    ylabel('Acceleration in vo/T units')
    %
    iloop = iloop + 1;
    figure(iloop)
    plot(tt,vel,'-')
    title('Rocket - Velocity in vo units')
    xlabel('Time in Total Possible Burn Time Units from 0 to
    Payload Burn Time')
    ylabel('Velocity in vo units')
    %
    iloop = iloop + 1;
    figure(iloop)
    for j = 1:length(tt)
      plot(tt(j),dis(j),'*')
      title('Rocket - Distance in vo*T units')
      xlabel('Time in Total Possible Burn Time Units from 0 to
      Payload Burn Time')
      ylabel('Distnace in vo*T units')
      axis([0,max(tt),0,1])
      pause(0.1)
    end
    plot(tt,dis,'-')
    title('Rocket - Distance in vo*T units')
    xlabel('Time in Total Possible Burn Time Units from 0 to
    Payload Burn Time')
    ylabel('Distnace in vo*T units')
  %
  end
end
%
```

2.8. Rocket Motion — Numerical

```
%
% Solve non-relativistic rocket, numerically using Saturn V as an
example
%
clear all;
help cm_rocket_num2 % Clear the memory and print header
%
% solve the rocket equation - done in cm_rocket_sym
% now do some numerical evaluations
%
gg = 9.8; % accel at earth surface m/sec^2
re = 6.378 .*10 .^6; % earth radius - m
veq = (2.0 .*pi .*re) ./(24 .*3.6 .*10^3); % equatorial launch velocity
km/sec
rs = 1.5 .*10 .^11; % distance to sun - m
me = 6.0 .*10 .^24; % earth mass - kg
ms = 2.0 .*10 .^30; % sun mass, - kg
%
vorb = sqrt(gg .*re); % orbital velocity - circular, low orbit
ve =sqrt(2.0 .*gg .*re); % escape velocity for Earth ~ 11.2 km/sec
vs = ve .*sqrt(ms .*re ./(me .*rs)); % escape velocity to leave solar
system ~ 42 km/sec
%
fprintf('Velocity, Satellite Low Circular Orbit (m/sec) = %g
\n',vorb);
fprintf('Escape Velocity - Earth (m/sec) = %g \n',ve);
fprintf('Escape Velocity - Solar System (m/sec) = %g \n',vs);
fprintf('Equatorial Launch Velocity (m/sec) = %g \n',veq);
%
irun = 1;
iloop = 0;
%
while irun > 0
   kk = menu('Pick Another Rocket?','Yes','No');
   if kk == 2
```

```
    irun = -1;
    break
  end
  if kk == 1
    %
    % total possible burn time T is mo/(dm/dt)
    % payload ratio mp/mo = 1-tp/T, tp = burn time for this
    payload
    %
    mo = input('Input the Rocket Mass (in 10^6 kg units) - Saturn
    = 4x10^6 kg: ');
    mo = mo .*10 .^6;
    mp = input('Input the Payload Mass (in kg) - Saturn Escape
    Module = 24610 kg: ');
    vo = input('Input the Exhaust Velocity (in m/sec) - Saturn =
    2200 m/sec: ');
    dmdt = input('Input Burn Rate (in kg/sec) - Saturn = 15000
    kg/sec: ');
    %
    T = mo ./dmdt ; % max possible burn rate, with no payload
    tp = T .*(1 - mp ./mo); % burn time for this payload
    mpf = mo .*exp(-ve ./vo); % estimated payload for free rocket
    to attain escape velocity
    mps = mo .*exp(-vs ./vo); % escape velocity from the solar
    system
    %
    fprintf('Maximum Burn Time (sec) = %g \n',T);
    fprintf('Burn Time for This Payload (sec) = %g \n',tp);
    fprintf('Payload Mass for Free Rocket to Attain Earth Escape
    Velocity = %g \n',mpf);
    fprintf('Payload Mass for Free Rocket to Attain Solar Escape
    Velocity = %g \n',mps);
    %
    tt = linspace(0,tp);
    tt = tt ./T;
    %
```

```
% the free rocket
%
for i = 1:length(tt)
   x = 1.0 ./(1.0 - tt(i));
   AF(i) = (vo ./T) .*x;
   VF(i) = vo .*log(x);
   YF(i) = vo .*T .*(1.0 - 1 ./x - log(x) ./x);
end
tt = tt .*T;
%
% the rocket in a uniform field = g
%
tl = -vo ./gg + T; % t=0 is ignition, t = tl is lift time, when
acceleration > 0
ul = T - tl;
fprintf('Time After Ignition for Acceleration to be > 0, Liftoff
= %g \n',tl);
%
for i = 1:length(tt)
   if tt(i) < tl
      AG(i) = 0.0;
      VG(i) = 0.0;
      YG(i) = 0.0;
   else
   %
      u = T - tt(i);
      AG(i) = vo ./u - gg;
      VG(i) = -vo .*log(u ./ul) + gg .*(u-ul);
      YG(i) = vo .*(u .*log(u ./ul) - (u-ul)) - gg .*(u-ul) .*(u-ul)
./2.0;
   end
end
iloop = iloop + 1;
figure(iloop)
semilogy(tt,AF,'-',tt, AG)
title(' Rocket - Acceleration in m/sec^2')
```

```
xlabel('Burn Time - sec')
ylabel('Acceleration')
legend('Free rocket','Rocket in g')
%
iloop = iloop + 1;
figure(iloop)
semilogy(tt,VF,'-',tt,VG)
hold on
semilogy(tt,vorb,'r-',tt,ve,'r:',tt,vs,'r–')
title(' Rocket - Velocity in m/sec')
xlabel('Burn Time - sec')
ylabel('Velocity')
legend('Free rocket','Rocket in g','Orbital Velocity', 'Earth
Escape Velocity', 'Sun Escape Velocity')
hold off
%
iloop = iloop + 1;
figure(iloop)
%
jj = length(YF);
xmax = max(tt);
ymax = max(YF);
for i = 1:jj
  semilogy(tt(i),YF(i),'o',tt(i),YG(i),'*')
  title(' Rocket - Distance m')
  xlabel('Burn Time - sec')
  ylabel('Distance - m')
  pause(0.1)
  axis([0, xmax, 0, ymax])
  hold on
end
hold off
semilogy(tt,YF,'-',tt,YG)
hold on
semilogy(tt,re,'r-')
title(' Rocket - Distance m')
```

```
    xlabel('Burn Time - sec')
    ylabel('Distance - m')
    legend('Free rocket','Rocket in g', 'Earth Radius')
    hold off
  %
  end
end
%
```

2.9. Taking the Free Subway

```
%
% earth subway - compute free fall through chord of earth
%
clear all; % Clear memory
help cm_subway2; % Print header
%
% Initialize variables, subway is defined by chord
%
g = 9.81; % Gravitational acceleration (m/s^2)
re = 6.38 .*10 .^6; % Earth radius (m)
%
fprintf("Free" Subway - Earth Radius = %g (m) \n',re);
%
irun = 1;
iloop = 0;
%
while irun > 0
  kk = menu('Pick Another Subway Distance?','Yes','No');
  if kk == 2
    irun = -1;
    break
  end
  if kk == 1
  %
  % pick chord for "free" subway
```

```
%
  dist = input('Enter Subway Distance in km : ');
  dist = dist .*1000;
  theta = asin(dist ./(2.0 .*re));
  depth = re .*(1-cos(theta));
  sdist = (re .*2.0 .*theta);
  %
  fprintf('Free Subway Max Depth = %g (m) \n',depth);
  fprintf('Free Subway Distance Along Earth = %g (m) \n',sdist);
  %
  % Gauss law - uniform Earth density ==> force due to distance
  to earth
  % center, mass inside scales at r^3
  % eq of motion is; accel = gx/re, due to |a|= GM/r^2~ r and
  dir
  % cosine= x/r
  % start with no velocity, supply no energy = "free subway" -
  "drop" to destination
  % simple harmonic motion
  %
  omega = sqrt(g ./re ); % SHM frequency
  T = (2.0 .*pi) ./omega; % period
  T = T ./2.0; % trip is one way = 1/2 period
  fprintf('Circular Frequency = %g Trip Time = %g (sec)
  \n',omega,T)
  %
  t = linspace(0,T);
  x = - re .*sin(theta) .*cos(omega .*t);
  %
  N = length(t);
  for jj=1:N
    plot(t(jj),real(x(jj)-x(1)),'o')
    title('Movie of Subway Trip')
    xlabel('time(sec)')
    ylabel('Distance Traversed by Subway (m)')
    axis([0 t(N) 0 max(x-x(1))])
```

```
      pause(0.1)
    end
    iloop = iloop + 1;
    figure(iloop)
    plot(t,real(x-x(1)))
    title('x(m) as a function of t along the subway')
    xlabel('t(sec)')
    ylabel('x(m)')
    %
  end
end
%
```

2.10. Large-Angle Oscillations — Pendulum

```
%
% Program to compute the motion of a simple pendulum
% using MATLAB tools
%
function cm_pendul
%
clear all;
help cm_pendul % Clear the memory and print header
%
global gL vo tho
%
fprintf('Pendulum - Large Oscillations \n');
%
irun = 1;
iloop = 0;
%
while irun > 0
  kk = menu('Pick Another Pendulum?','Yes','No');
  if kk == 2
    irun = -1;
    break
```

```
end
if kk == 1
  %
  % Set initial position and velocity of pendulum
  %
  tho = input('Enter Initial Angle (degrees): ');
  tho = (tho .*pi) ./180.0; % Convert angle to radians
  vo = input('Enter Initial Angular Velocity (degrees/sec): ');
  vo = (vo .*pi) ./180.0;
  gL = input('Enter g/L in MKS units: ');
  %
  % small angle period, omeg = sqrt(gL);
  %
  omega = sqrt(gL);
  T = (2.0 .*pi) ./omega;
  tt = linspace(0, 2.0 .*T);
  %
  fprintf('Small Angle Circular Frequency = %g 1/sec \n',omega);
  fprintf('Small Angle Period (sec) = %g \n',T);
  fprintf('Period = 2 * pi *sqrt(L/g) Increased by Factor 1 +
  theta o^2/16 \n');
  %
  % numerical solution using ODE tools
  %
  [t,y] = ode45(@pend,tt,[vo tho]);
  %
  % small angle SHM for comparison
  %
  yyy = tho .*cos(omega .*tt) + (vo .* sin(omega .*tt)) ./omega;
  yyyy = -tho .*omega .*sin(omega .*tt) + vo .*cos(omega .*tt);
  %
  iloop = iloop + 1;
  figure(iloop)
  yy = y(:,1);
  plot(t,yy,'-',tt,yyyy,':')
  title('Angular Velocity')
```

```
    xlabel('t(sec)')
    ylabel('d\theta/dt')
    legend('Full Solution','Small Oscillation')
    %
    iloop= iloop + 1;
    figure(iloop)
    zz = y(:,2);
    N = length(t);
    for j = 1:N
      %
      plot(t(j),zz(j),'o',tt(j),yyy(j),'*')
      title('Angular Position')
      xlabel('t(sec)')
      ylabel('\theta(rad)')
      legend('Full Solution','Small Oscillation')
      axis([0 max(t),min(zz),max(zz)])
      pause(0.1)
    end
    plot(t,zz,'b-',tt,yyy,'r:')
    title('Angular Position')
    xlabel('t(sec)')
    ylabel('\theta(rad)')
    legend('Full Solution','Small Oscillation')
  end
end
%
function dy = pend(t,y)
%
global gL vo tho
%
dy = zeros(2,1);
dy(1) = -gL .*sin(y(2));
dy(2) = y(1);
%
```

2.11. Double Pendulum

```
%
% Program to compute the motion of 2 coupled pendula
% using MATLAB tools, chaotic large angle motion
%
function cm_chaotic
%
clear all;
help cm_chaotic % Clear the memory and print header
%
global L vo tho
%
fprintf('Two Coupled Pendulum - Large Oscillations \n');
%
irun = 1;
iloop = 0;
%
while irun > 0
  kk = menu('Pick Another Two Initial Angles?','Yes','No');
  if kk == 2
    irun = -1;
    break
  end
  if kk == 1
    %
    % Set initial position of pendula
    %
    tho = input('Enter Initial Angles (degrees), Velocities = 0,[th1,
    th2]: ');
    tho = (tho .*pi) ./180.0; % Convert angle to radians
    % L = input('Enter L in MKS units, m = 1 and g/L = 1: ');
    L = 1;
    vo = [0 0];
    %
```

```
% numerical solution using ODE tools
%
tspan = linspace(0,50,100);
[t2,y2] = ode45(@pend2,tspan,[vo(1) vo(2) tho(1) tho(2)]);
%
iloop = iloop + 1;
figure(iloop)
yy1 = y2(:,1);
yy2 = y2(:,2);
plot(t2,yy1,'-b',t2,yy2,'r:')
title('Angular Velocity of Pendula')
xlabel('t(sec)')
ylabel('d\theta/dt')
legend('First Pendulum','Second Pendulum')
%
iloop = iloop + 1;
figure(iloop)
yy3 = y2(:,3);
yy4 = y2(:,4);
plot(t2,yy3,'-b',t2,yy4,'r:')
title('Angular Position of Pendula')
xlabel('t(sec)')
ylabel('\theta(t) - rad')
legend('First Pendulum','Second Pendulum')
%
iloop= iloop + 1;
figure(iloop)
zz1 = y2(:,3);
zz2 = y2(:,4);
N = length(t2);
for j = 1:N
  %
  xxx1(1) = 0;
  yyy1(1) = 0;
  xxx1(2) = L .*sin(zz1(j));
  yyy1(2) = -L .*cos(zz1(j));
```

```
        xxx2(1) = xxx1(2);
        yyy2(1) = yyy1(2);
        xxx2(2) = xxx1(2) +L .*sin(zz2(j));
        yyy2(2) = yyy1(2) -L .*cos(zz2(j));
        plot(xxx1,yyy1,'-b',xxx2,yyy2,'-r',xxx1(1),yyy1(1),'*g',
xxx1(2),yyy1(2),'bo',xxx2(2),yyy2(2),'ro')
        title('Two Pendula')
        xlabel('x')
        ylabel('y')
        axis([-1.5 1.5 -2.5 0.5])
        pause(0.1)
      end
   end
end
%
function dy = pend2(t,y)
%
global L vo tho
%
dy = zeros(4,1);
fact = dy(3) .*dy(4) .*sin(y(3)-y(4)) + 3 .*sin(y(3));
dy(1) = - (L .^2 .*fact) ./2.0;
fact = -dy(3) .*dy(4) .*sin(y(3)-y(4)) + sin(y(4));
dy(2) = - (L .^2 .*fact) ./2.0;
fact = 16.0 - 9.0 .*(cos(y(3)-y(4)) .^2);
dy(3) = 6.0 ./(L .*L .*fact);
dy(3) = dy(3) .*(2.0 .*y(1)-3.0 .*cos(y(3)-y(4)) .*y(2));
dy(4) = 6.0 ./(L .*L .*fact);
dy(4) = dy(4) .*(8.0 .*y(2)-3.0 .*cos(y(3)-y(4)) .*y(1));
%
```

2.12. Coriolis Force

```
%
% Program to look at Coriolis force, symbolic solution plus numerical
% Free fall on Surface of the Earth
```

```
%
clear all;
help cm_coriolis; % Clear memory and print header
%
fprintf('Coriolis Force, Nothern Hemisphere, w is w*cos, Latitude
\n');
%
% look at symbolic ODE solution, z vertical, x south, y east
%
syms y z w g yy zz
%
[yy,zz] = dsolve('D2y=-2*Dz*w','Dy(0)=0','y(0)=0','D2z=g','Dz(0)
=0','z(0)=0');
%
fprintf(' z is Vertical, x is South and y is East \n')
%
zz
yy
%
w = ( 2.0 .*pi) ./(24 .*60 .*60); % Earth rotation, rad/sec
g = 9.8; % acceleration in m/sec^2
%
% now numerical evaluations
%
irun = 1;
iloop = 0;
%
while irun > 0
  kk = menu('Pick Another Free Fall Height and Latitude?','Yes',
  'No');
  if kk == 2
    irun = -1;
    break
  end
  if kk == 1
    %
```

```
    zh = input('Enter Initial Free Fall Height (m): ');
    th = input('Enter Latitude (deg): ');
    th = (2.0 .*pi .*th) ./360.0;
    %
    z = linspace(0,zh);
    zz = z(100)-z;
    %
    % z = gt^2/2 - remove t to find y(z)
    %
    y = - ((g .*w .*cos(th)) .*(((2.0 .*z) ./g) .^1.5)) ./3.0;
    yy = y(1) - y;
    %
    fprintf('Total Eastward Deflection (m) = %g \n',yy(100));
    %
    iloop = iloop + 1;
    figure(iloop)
    plot(yy,zz)
    xlabel('y(m)')
    ylabel('z(m)')
    title('Free Fall Coriolis Deflection')
    %
  end
  %
end
%
```

2.13. Kepler Orbits — Numerical

```
%
% Kepler — Program to compute solar system orbits - simple
numerical integration
%
function cm_kepl3
%
clear all;    % Clear memory
help cm_kepl3;    % Print header
```

```
%
global G Mo
%
% Initialize variables
%
G = 6.67 .*10 .^-11; % MKS units
Mo = 2.0 .*10 .^30; % solar mass
au = 1.49 .*10 .^11; % AU = earth-Sun distance, m
yr = 60.0 .*60.0 .*24.0 .*365.0; % sec
%
irun = 1;
iloop = 0;
%
while irun > 0
  kk = menu('Pick Another Solar Orbit?','Yes','No');
  if kk == 2
    irun = -1;
    break
  end
  if kk == 1
    %
    ro = input('Enter Initial Distance ro(AU): ');
    roa = ro .*au;
    %
    ve = sqrt((2.0 .*G .*Mo) ./roa); % escape velocity
    fprintf('Escape Velocity, v = %g m/sec\n',ve);
    %
    vc = sqrt(G .*Mo ./roa);
    fprintf('Velocity of circular orbit, v = %g m/sec\n',vc);
    T = (2.0 .*pi .*roa) ./vc;
    fprintf('For circular orbit, period = %g sec\n',T);
    fprintf('For Earth Orbit, 1 au = %g m, period = %g sec
    \n',au,yr);
    %
    voy = input('Enter initial tangential velocity (AU/yr), 2\pi for
    Circle: ');
```

```
vox = input('Enter initial radial velocity (AU/yr): ');
%
% Set up for plotting the orbit
%
% convert to m, sec
%
vox = (vox .*au) ./yr ;
voy = (voy .*au) ./yr ;
%
tspan = linspace(0,5.0 .*T,200);
[t,y] = ode45(@kepler,tspan,[vox roa voy 0]);
%
iloop = iloop + 1;
figure(iloop)
%
N = length(tspan);
Nloop = 0;
for j = 1:N
  xx(j) = y(j,2) ./roa;
  yy(j) = y(j,4) ./roa;
end
for j = 1:N-1
  if (yy(j+1) .* yy(j) + 0.001 > 0) ||(yy(j+1) < 0)
    Nloop = Nloop + 1;
    xxl(Nloop) = xx(j);
    yyl(Nloop) = yy(j);
  else
    break
  end
end
xmax = max(xxl);
xmin = min(xxl);
ymax = max(yyl);
ymin = min(yyl);
%
% the movie first, to understand orbital velocity
```

```
      %
      for i = 1:Nloop
        plot(xxl(i),yyl(i),'o')
        hold on
        title('Trajectory of Orbit, 5 Circular Periods or Stop on
        Repeat')
        xlabel('x/ro')
        ylabel('y/ro')
        plot(0.0,0.0,'r*')
        axis([xmin xmax ymin ymax])
        pause(0.1)
      end
      hold off
      %
      plot(xxl,yyl,'-',0.0,0.0,'r*')
      title('Trajectory of Orbit, 5 Circular Periods or Stop on Repeat')
      xlabel('x/ro')
      ylabel('y/ro')
      axis([xmin xmax ymin ymax])
   end
%
end
%
%- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
%
function dy = kepler(t,y)
global G Mo
%
dy = zeros(4,1);
r = sqrt(y(2) .^2 + y(4) .^2);
fr = -(G .*Mo) ./(r .^2.0);
dy(1) = (y(2) .*fr) ./r;
dy(3) = (y(4) .*fr) ./r;
dy(2) = y(1);
dy(4) = y(3);
%
```

2.14. Analytic Kepler Orbits — Energy Considerations

```
%
% Program to compute solar system orbits, closed and open
%
clear all;    % Clear memory
help cm_kepl;    % Print header
%
G = 6.67 .*10 .^-11; % MKS units
Mo = 2.0 .*10 .^30; % solar mass, kg
au = 1.49 .*10 .^11; % AU = earth-Sun distance, m
yr = 60.0 .*60.0 .*24.0 .*365.0; % year in sec
%
irun = 1;
iloop = 0;
%
while irun > 0
   kk = menu('Pick Another Solar Orbit?','Yes','No');
   if kk == 2
      irun = -1;
      break
   end
   if kk == 1
      %
      ro = input('Enter Initial Distance ro(AU): ');
      %
      fprintf('L^2 is GMro/m^2 for a Circular Orbit at Radius ro \n');
      %
      % Find Effective 1-d Potential - Use centrifugal Potential,
V~1/r^2
      %
      xxx = linspace(0.25,10.0); % r variation in ro units
      Veff = 1 ./(2.0 .*xxx .*xxx) - 1.0 ./xxx;
      xmin = 1 ; % min of Veff
      Veffmin = -1.0 ./2.0; % Veff at min
      %
```

```
iloop = iloop + 1;
figure(iloop)
plot(xxx,Veff,xmin,Veffmin,'*')
title(' Effective Potential for L^2, Ec = -1/2,0 > Eellipse >
Ecircle')
xlabel('r/ro')
ylabel('Veff/(GMm/ro)')
axis([0 10 -0.6 0.5])
%
% circular radius for this L is ro, controlling variable is energy
% E
%
ac = ro .*au; % in m
%
ve = sqrt((2.0 .*G .*Mo) ./ac); % escape velocity, circular orbit
vee = ve ./1000.;
vo = sqrt((G .*Mo) ./ac); % circular velocity
voo = vo ./1000.;
To = (2.0 .*pi .*ac) ./vo; % circular orbit period
Too = To ./yr;
%
fprintf('Escape Velocity (km/sec) = %g at ac (au) =
%g\n',vee,ac ./au);
fprintf('For circular orbit, v(km/sec) = %g, Period (yr) = %g
\n',voo,Too);
%
q = input('Enter Total Energy in Units of Circular Energy -
G*M*m/2*ro, = q > -1: ');
ecc = sqrt(1.0 + q); % eccentricity
%
if q < -1
  fprintf('No Solution\n');
end
if q > 0
  fprintf('Hyperbolic Orbits\n');
  % turning points of potential in terms of ro - i.e. elliptical axes
```

```
        x1 = (1.0 ./q) .*(-1.0 + sqrt(1.0 +q));
        x2 = (1.0 ./q) .*(-1.0 - sqrt(1.0 +q)); % eccentricity is e =
sqrt(1+q)
        fprintf('For Hyperbolic Orbit, Turning Point in ro units = %g
        \n',x1);
        hold on
        plot(x1,-q .*Veffmin,'o')
        hold off
        %
      end
      if q == 0
        fprintf('Parabolic Orbits\n');
      end
      if q < 0 & q >= -1
        fprintf('Elliptical Orbits\n');
        %
        % find the turning points, axes and period
        %
        ac = ac ./au ; % circular orbit radius in ro units of au
        ae = ac ./abs(q); % major axis
        be = ae .*sqrt(1.0 - ecc .^2); % minor axis
        % period
        TT = (2.0 .*pi .*(ae .*au) .^1.5)./sqrt(G .*Mo);
        TT = TT ./yr;
        % turning points of potential in terms of ro - i.e. elliptical axes
        x1 = (1.0 ./q) .*(-1.0 + sqrt(1.0 +q));
        x2 = (1.0 ./q) .*( -1.0 - sqrt(1.0 +q)); % eccentricity is e =
        sqrt(1+q)
        %
        fprintf('For Elliptical Orbit, Major/Minor Axes (au) = %g,
        %g \n',ae,be);
        fprintf('For Elliptical Orbit, Turning Points in ro units = %g,
        %g \n',x1,x2);
        fprintf('For Elliptical Orbit, Orbital Period (yr) = %g \n',TT);
        fprintf('For Elliptical Orbit, Eccentricity = %g \n',ecc);
        %
```

```
  hold on
  plot(x1,-q .*Veffmin,'o', x2,-q .*Veffmin,'o')
  hold off
  %
end
%
% populate cos theta and find the radius r, in ro units - x as
above
%
iloop = iloop + 1;
figure(iloop)
theta = linspace(0,2 .*pi);
ct = cos(theta);
st = sin(theta);
Xr = 1.0 ./(1.0 + ct .*ecc);
dth = theta(2)-theta(1);
%
% numerical integration to get elapsed time on orbit points, t in
units of To
%
t(1) = 0;
for i =2:100
  t(i) = t(i-1) + dth ./(2.0 .*pi .*((1.0 + ct(i) .*ecc) .^2));
end;
%
% velocity in units of circular velocity at radius ac
%
vel = sqrt(q + 2.0 .*(1.0 + ct .*ecc));
%
xx = ct .*Xr;
yy = st .*Xr;
%
plot(xx,yy,'b-',0,0,'r*');
if q < 0.0 & q >= -1
  hold on
```

```
        minx(1) = (x1-x2) ./2;
        minx(2) = (x1-x2) ./2;
        miny(1) = -be;
        miny(2) = be;
        majx(1) = x1;
        majx(2) = -x2;
        majy(1) = 0;
        majy(2) = 0;
        plot(minx,miny,'-r',majx,majy,'g-')
        axis square
        axis equal
        hold off
      end
      title('Orbit for this Choice of ro and E')
      xlabel('x/ro')
      ylabel('y/ro')
      %
      iloop = iloop + 1;
      figure(iloop)
      plot(theta ./(2.0 .*pi),t .*To ./yr)
      xlabel('orbit angle/2\pi')
      ylabel('Time Elapsed (yr)')
      title('Orbital Time as a Function of Orbital Angle')
      %
      iloop = iloop + 1;
      figure(iloop)
      plot(theta ./(2.0 .*pi),vel)
      xlabel('orbit angle/2\pi')
      ylabel('Orbital Velocity')
      title('Orbital Velocity in Units of Circular Velocity at Radius =
      ro')
    end
%
end
%
```

2.15. Stable Orbits and Perihelion Advance

```
%
% Program to look at Perturbed Circular Orbits - Stability, and
Perihelion Advance
%
function cm_circl_orbit
%
clear all;
help cm_circl_orbit; % Clear memory and print header
%
global Itype n b
%
fprintf('Circular Orbits - Perturbed \n');
%
% now numerical evaluations
%
irun = 1;
iloop = 0;
%
while irun > 0
   kk = menu('Pick Another Power Law?','Yes','No');
   if kk == 2
      irun = -1;
      break
   end
   if kk == 1
      %
      n = input('Enter Power Law Force n for f(r) ~ 1/r^n : ');
      fprintf('Circular Orbits - Perturbed, Stable only for n < 3 \n');
      %
      if n < 3
         fprintf('Stable Perturbations: ')
      else
         fprintf('Unstable Perturbations: ')
      end
      %
```

```
    fprintf('For a Given n, Period T^2 ~ radius^(n+1), Kepler is
T^2 = r^3 \n')
    %
    % now time interval for 3 circular orbits
    % pick a = radius = 1, f(a) = c/a^n and c = 1% pick m = 1
==> v = 1
    %
    Itype = 1;
    tspan = linspace(0,2.0 .*pi);
    [t,y] = ode45(@Perihel,tspan,[0.0 0.1]);
    %
    fprintf('Initial position is displaced by x(0) = 0.10 \n ')
    fprintf('Initial velocity is v(o) = 1, Dimensionless Units \n ')
    %
    iloop = iloop + 1;
    figure(iloop)
    plot(t ./(2.0 .*pi),y(:,2))
    xlabel('t(periods)')
    ylabel('x/a')
    title('Deviation from Circular Orbit Over One Unperturbed
    Period')
    %
  end
  %
end
% now the perihelion advance
%
fprintf('Perihelion Advance - Inverse Square Law Plus Small Inverse
Fourth Power \n ')
Itype = 2;
b = input('Input the Coefficient of the Fourth Power: ');
tspan = linspace(0,6.0 .*pi);
[t,y] = ode45(@Perihel,tspan,[0.0 0.1]);
fprintf('Initial position is displaced by x(0) = 0.10 \n ')
fprintf('Initial velocity is v(o) = 1, Dimensionless Units \n ')
%
```

```
iloop = iloop + 1;
figure(iloop)
plot(t ./(2.0 .*pi),y(:,2))
xlabel('t(periods)')
ylabel('x/a')
title('Deviation from Closed Orbit Over Three Periods')
%
iloop = iloop + 1;
figure(iloop)
plot(cos(t) + y(:,2), sin(t))
xlabel('t(periods)')
ylabel('x/a')
title('Orbit Over Three Periods')
%
%- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
%
function dy = Perihel(t,y)
%
global Itype n b
%
dy = zeros(2,1);
if Itype == 1; % perturbed circ orbits - different force laws - n
   dy(1) = -1.0 ./(1 + y(2)) .^n + 1.0 ./(1.0 + y(2)) .^3 ;
   dy(2) = y(1); % y(1) = vx, y(2) = x
end
%
if Itype == 2; % central inverse sq law with small (b) inverse fourth
   power
   dy(1) = -1.0 ./(1 + y(2)) .^2 - b ./(1 + y(2)) .^4 + (1.0 + b)
./(1.0 + y(2)) .^3;
   dy(2) = y(1); % y(1) = vx, y(2) = x
end
```

This page intentionally left blank

References

1. *Numerical Methods for Physics*, 2nd Ed., Alejandro L. Garcia, Prentice Hall, 2000, ISBN: 0-13-906744-2. This text has companion script which I have found very useful as a starting point for some exercises.
2. *Numerical Recipes in Fortran 77*: *The Art of Scientific Computing*, 2nd Ed., W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, Cambridge University Press, 1992, ISBN 978 0 521 43064 7. This book is a veritable encyclopedia of numerical methods.

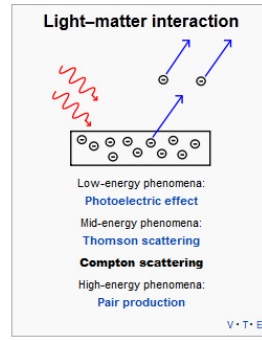
There are many excellent textbooks available for use as references. However, as technology has evolved, the increasing use of online resources is at hand. Therefore, a complete set of paper textbooks is not quoted here. Rather the search engines and compiled online knowledge bases are invoked.

3. Google is an enormously useful search engine and many specific searches will yield a great variety of information.
4. Wikipedia has a large store of interesting Physics topics, and a search through them will very often start the user on a good path. Indeed, a particular article often has many links that can be followed deeper into the topic. An example of the first page of a search for "Compton Scattering" is shown below. There are additional links and references provided that give the user a very good reference experience. Indeed, while looking at the MATLAB scripts for this text, the user can easily dip into the online resources and gain further knowledge.

Compton scattering

From Wikipedia, the free encyclopedia

Compton scattering is an inelastic scattering of a photon by a free charged particle, usually an electron. It results in a decrease in energy (increase in wavelength) of the photon (which may be an X-ray or gamma ray photon), called the **Compton effect**. Part of the energy of the photon is transferred to the scattering electron. **Inverse Compton scattering** also exists, in which a charged particle transfers part of its energy to a photon.



Contents [hide]

Introduction [edit]

Index