SECOND EDITION

MATHEMATICA *for* THEORETICAL PHYSICS

Electrodynamics, Quantum Mechanics, General Relativity, and Fractals

GERD BAUMANN

Springer

CD-ROM Included

Mathematica® for Theoretical Physics

Mathematica®

for Theoretical Physics

Electrodynamics,
Quantum Mechanics,
General Relativity,
and Fractals

Second Edition

Gerd Baumann

CD-ROM Included

Springer

Gerd Baumann
Department of Mathematics
German University in Cairo GUC
New Cairo City
Main Entrance of Al Tagamoa Al Khames
Egypt
Gerd.Baumann@GUC.edu.eg

This is a translated, expanded, and updated version of the original German version of the work "*Mathematica*® in der Theoretischen Physik," published by Springer-Verlag Heidelberg, 1993 ©.

Library of Congress Cataloging-in-Publication Data
Baumann, Gerd.
[Mathematica in der theoretischen Physik. English]
Mathematica for theoretical physics / by Gerd Baumann.—2nd ed.
p. cm.
Includes bibliographical references and index.
Contents: 1. Classical mechanics and nonlinear dynamics — 2. Electrodynamics, quantum mechanics, general relativity, and fractals.
ISBN 0-387-21933-1
1. Mathematical physics—Data processing. 2. Mathematica (Computer file) I. Title.

QC20.7.E4B3813 2004
530′.285′53—dc22 2004046861

ISBN-10: 0-387-21933-1 e-ISBN 0-387-25113-8 Printed on acid-free paper.
ISBN-13: 978-0387-21933-2

Printed in the United States of America. (HAM)

9 8 7 6 5 4 3 2 1

springeronline.com

To Carin,
for her love, support, and encuragement.

Preface

As physicists, mathematicians or engineers, we are all involved with mathematical calculations in our everyday work. Most of the laborious, complicated, and time-consuming calculations have to be done over and over again if we want to check the validity of our assumptions and derive new phenomena from changing models. Even in the age of computers, we often use paper and pencil to do our calculations. However, computer programs like *Mathematica* have revolutionized our working methods. *Mathematica* not only supports popular numerical calculations but also enables us to do exact analytical calculations by computer. Once we know the analytical representations of physical phenomena, we are able to use *Mathematica* to create graphical representations of these relations. Days of calculations by hand have shrunk to minutes by using *Mathematica*. Results can be verified within a few seconds, a task that took hours if not days in the past.

The present text uses *Mathematica* as a tool to discuss and to solve examples from physics. The intention of this book is to demonstrate the usefulness of *Mathematica* in everyday applications. We will not give a complete description of its syntax but demonstrate by examples the use of its language. In particular, we show how this modern tool is used to solve classical problems.

This second edition of *Mathematica in Theoretical Physics* seeks to prevent the objectives and emphasis of the previous edition. It is extended to include a full course in classical mechanics, new examples in quantum mechanics, and measurement methods for fractals. In addition, there is an extension of the fractal's chapter by a fractional calculus. The additional material and examples enlarged the text so much that we decided to divide the book in two volumes. The first volume covers classical mechanics and nonlinear dynamics. The second volume starts with electrodynamics, adds quantum mechanics and general relativity, and ends with fractals. Because of the inclusion of new materials, it was necessary to restructure the text. The main differences are concerned with the chapter on nonlinear dynamics. This chapter discusses mainly classical field theory and, thus, it was appropriate to locate it in line with the classical mechanics chapter.

The text contains a large number of examples that are solvable using *Mathematica*. The defined functions and packages are available on CD accompanying each of the two volumes. The names of the files on the CD carry the names of their respective chapters. Chapter 1 comments on the basic properties of *Mathematica* using examples from different fields of physics. Chapter 2 demonstrates the use of *Mathematica* in a step-by-step procedure applied to mechanical problems. Chapter 2 contains a one-term lecture in mechanics. It starts with the basic definitions, goes on with Newton's mechanics, discusses the Lagrange and Hamilton representation of mechanics, and ends with the rigid body motion. We show how *Mathematica* is used to simplify our work and to support and derive solutions for specific problems. In Chapter 3, we examine nonlinear phenomena of the Korteweg–de Vries equation. We demonstrate that *Mathematica* is an appropriate tool to derive numerical and analytical solutions even for nonlinear equations of motion. The second volume starts with Chapter 4, discussing problems of electrostatics and the motion of ions in an electromagnetic field. We further introduce *Mathematica* functions that are closely related to the theoretical considerations of the selected problems. In Chapter 5, we discuss problems of quantum mechanics. We examine the dynamics of a free particle by the example of the time-dependent Schrödinger equation and study one-dimensional eigenvalue problems using the analytic and

numeric capabilities of *Mathematica*. Problems of general relativity are discussed in Chapter 6. Most standard books on Einstein's theory discuss the phenomena of general relativity by using approximations. With *Mathematica*, general relativity effects like the shift of the perihelion can be tracked with precision. Finally, the last chapter, Chapter 7, uses computer algebra to represent fractals and gives an introduction to the spatial renormalization theory. In addition, we present the basics of fractional calculus approaching fractals from the analytic side. This approach is supported by a package, FractionalCalculus, which is not included in this project. The package is available by request from the author. Exercises with which *Mathematica* can be used for modified applications. Chapters 2–7 include at the end some exercises allowing the reader to carry out his own experiments with the book.

Acknowledgments Since the first printing of this text, many people made valuable contributions and gave excellent input. Because the number of responses are so numerous, I give my thanks to all who contributed by remarks and enhancements to the text. Concerning the historical pictures used in the text, I acknowledge the support of the http://www-gapdcs.st-and.ac.uk/~history/ webserver of the University of St Andrews, Scotland. My special thanks go to Norbert Südland, who made the package FractionalCalculus available for this text. I'm also indebted to Hans Kölsch and Virginia Lipscy, Springer-Verlag New York Physics editorial. Finally, the author deeply appreciates the understanding and support of his wife, Carin, and daughter, Andrea, during the preparation of the book.

Cairo, Spring 2005

Gerd Baumann

Contents

Volume I

Volume II

4
Electrodynamics

4.1 Introduction

This chapter is concerned with electric fields and charges encountered in different systems. Electricity is an ancient phenomenon already known by the Greeks. The experimental and theoretical basis of the current understanding of electrodynamical phenomena was established by two men: Michael Farady, the self-trained experimenter, and James Clerk Maxwell, the theoretician. The work of both were based on extensive material and knowledge by Coulomb. Farady, originally, a bookbinder, was most interested in electricity. His inquisitiveness in gaining knowledge on electrical phenomena made it possible to obtain an assistantship in Davy's lab. Farady (see Figure 4.1.1) was one of the greatest experimenters ever. In the course of his experiments, he discovered that a suspended magnet would revolve around a current bearing-wire. This observation led him to propose that magnetism is a circular force. He invented the dynamo in 1821, with which a large amount of our current electricity is generated. In 1831, he discovered electromagnetic induction. One of his most important contributions to

physics in 1845 was his development of the concept of a field to describe magnetic and electric forces.

Figure 4.1.1. Michael Faraday: born September 22, 1791; died August 25, 1867.

Maxwell (see Figure 4.1.2) started out by writing a paper entitled "On Faraday's Lines of Force" (1856), in which he translated Faraday's theories into mathematical form. This description of Faraday's findings by means of mathematics presented the lines of force as imaginary tubes containing an incompressible fluid. In 1861, he published the paper "On Physical Lines of Force" in which he treated the lines of force as real entities. Finally, in 1865, he published a purely mathematical theory known as "On a Dynamical Theory of the Electromagnetic Field". The equations derived by Maxwell and published in "A Treaties on Electricity and Magnetism" (1873) are still valid and a source of basic laws for engineering as well as physics.

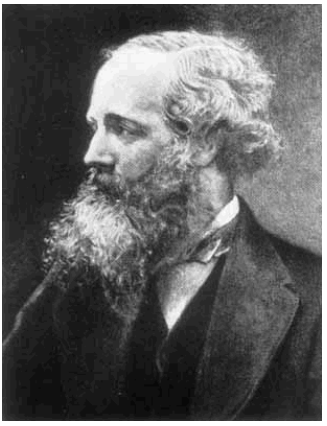


Figure 4.1.2. James Clerk Maxwell: born June 13, 1831; died November 5, 1879.

The aim of this chapter is to introduce basic phenomena and basic solution procedures for electric fields. The material discussed is a collection of examples. It is far from being complete by considering the huge diversity of electromagnetic phenomena. However, the examples discussed demonstrate how symbolic computations can be used to derive solutions for electromagnetic problems.

This chapter is organized as follows: Section 4.2 contains material on point charges. The exampl discuss the electric field of an assembly of discrete charges distributed in space. In Section 4.3, a standard boundary problem from electrostatics is examined to solve Poisson's equation for an angular segment. The dynamical interaction of electric fields and charged particles in a Penning trap is discussed in Section 4.4.

4.2 Potential and Electric Fields of Discrete Charge Distributions

In electrostatic problems, we often need to determine the potential and the electric fields for a certain charge distribution. The basic equation of electrostatics is Gauss' law. From this fundamental relation connecting the charge density with the electric field, the potential of the field can be derived. We can state Gauss' law in differential form by

$$\text{div}\,\vec{E} = 4\pi\rho(\vec{r}). \tag{4.2.1}$$

If we introduce the potential Φ by $\vec{E} = -\text{grad}\,\Phi$, we can rewrite Eq. (4.2.1) for a given charge distribution ρ in the form of a Poisson equation

$$\Delta\Phi = -4\pi\rho \tag{4.2.2}$$

where ρ denotes the charge distribution. To obtain solutions of Eq. (4.2..2), we can use the Green's function formalism to derive a particular solution. The Green's function $G(\vec{r}, \vec{r}')$ itself has to satisfy a Poisson equation where the continuous charge density is replaced by Dirac's delta function $\Delta_r G(\vec{r}, \vec{r}') = -4\pi\delta(\vec{r} - \vec{r}')$. The potential Φ is then given by

$$\Phi(\vec{r}) = \int_V G(\vec{r}, \vec{r}')\,\rho(\vec{r}')\,d^3r'. \tag{4.2.3}$$

In addition, we assume that the boundary condition $G|_V = 0$ is satisfied on the surface of volume V. If the space in which our charges are located is infinitely extended, the Green's function is given by

$$G\left(\vec{r}, \vec{r}'\right) = \frac{1}{|\vec{r}-\vec{r}'|} \tag{4.2.4}$$

The solution of the Poisson equation (4.2.3) becomes

$$\Phi(\vec{r}) = \int \frac{\rho(\vec{r}')}{|\vec{r}-\vec{r}'|}\,d^3r'. \tag{4.2.5}$$

Our aim is to examine the potential and the electric fields of a discrete charge distribution. The charges are characterized by a strength q_i and are located at certain positions $\vec{r}_i$. The charge density of such a distribution is given by

$$\rho(\vec{r}) = \sum_{i=1}^{N} q_i\,\delta(\vec{r}_i). \tag{4.2.6}$$

The potential of such a discrete distribution of charges is in accordance with Eq. (4.2.5):

$$\Phi(\vec{r}) = \sum_{i=1}^{N} \frac{q_i}{|\vec{r}-\vec{r}_i|}, \tag{4.2.7}$$

where $\vec{r}_i$ denotes the location of the point charge. The corresponding electrical field is given by

$$\vec{E}(\vec{r}) = -\sum_{i=1}^{N} q_i \vec{r} - \frac{\vec{r}_i}{|\vec{r}-\vec{r}_i|^3} \tag{4.2.8}$$

and the energy density of the electric field of such a charge distribution is given by

$$w = \frac{1}{8\pi} \left| \vec{E} \right|^2. \tag{4.2.9}$$

Three fundamental properties of a discrete charge distribution are defined by Eqs. (4.2.7), (4.2.8), and (4.2.9). In the following, we write a *Mathematica* package which computes the potential, the electric field, and the energy density for a given charge distribution. With this package, we are able to create pictures of the potential, the electric field, and the energy density.

In order to design a graphical representation of the three quantities, we need to create contour plots of a three-dimensional space. To simplify the handling of the functions, we enter the cartesian coordinates of the locations and the strength of the charges as input variables in a list. Sublists of this list contain the information for specific charges. The structure of the input list is given by $\{\{x_1, y_1, z_1, \rho_1\}, \{x_2, y_2, z_2, \rho_2\}, \ldots\}$. To make things simple in our examples, we choose the $y = 0$ section of the three-dimensional space. The package **PointCharge`**, located in the section on packages and programs, contains the equations discussed above. The package generates contour plots of the potential, the electric field, and the energy density.

In order to test the functions of this package, let us consider some ensembles of charges frequently discussed in literature. Our first example describes two particles carrying the opposite charge, known as a dipole. Let us first define the charges and their coordinates by

```
charges = {{1,0,0,1},{-1,0,0,-1}}
```

```
{{1, 0, 0, 1}, {-1, 0, 0, -1}}
```

The charges are located in space at $x = 1$, $y = 0$, $z = 0$ and at $x = -1$, $y = 0$, $z = 0$. The fourth element in the sublists specifies the strength of the charges. The picture of the contour lines of the potential is created by calling

```
FieldPlot[charges,"Potential"];
```

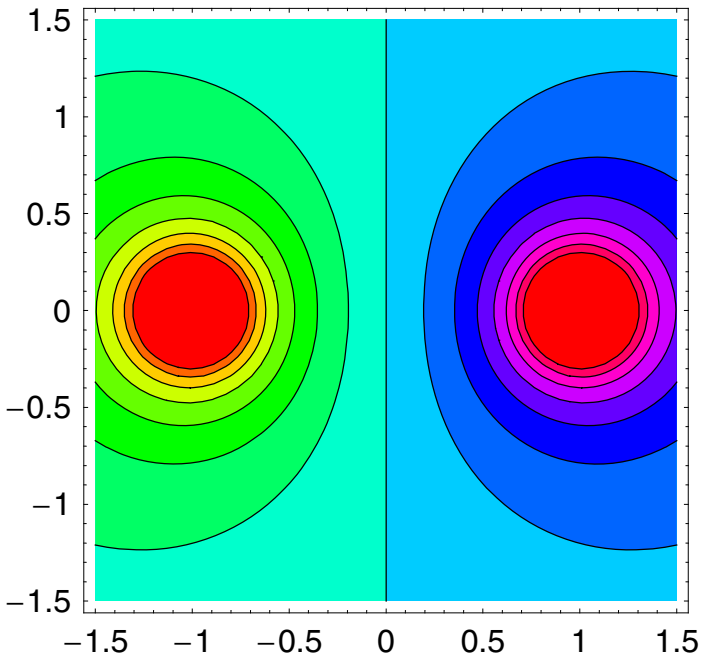


Figure 4.2.3. Contour plot of the potential for two charges in the (x, z)-plane. The particles carry opposite charges.

The second argument of **FieldPlot[]** is given as a string specifying the type of the contour plot. Possible values are *Potential*, *Field*, and *EnergyDensity*.

A graphical representation of the energy density follows by

```
FieldPlot[charges,"EnergyDensity"];
```

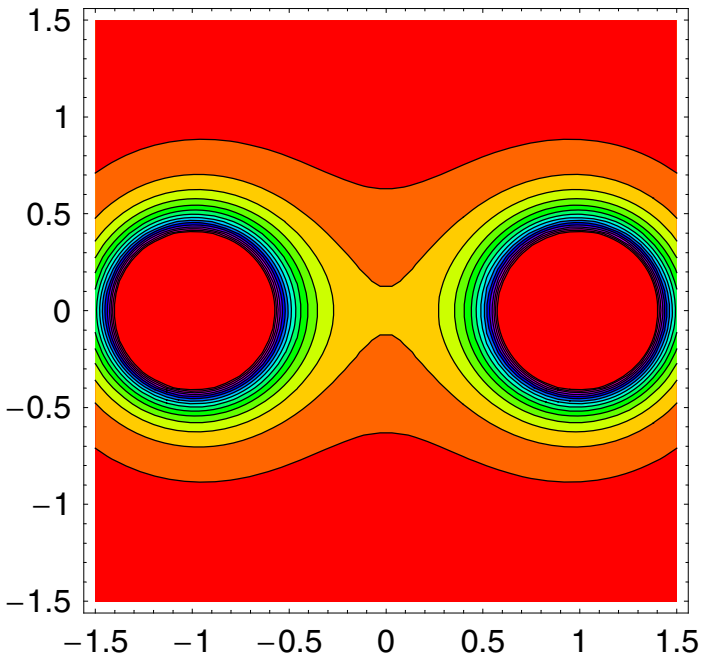
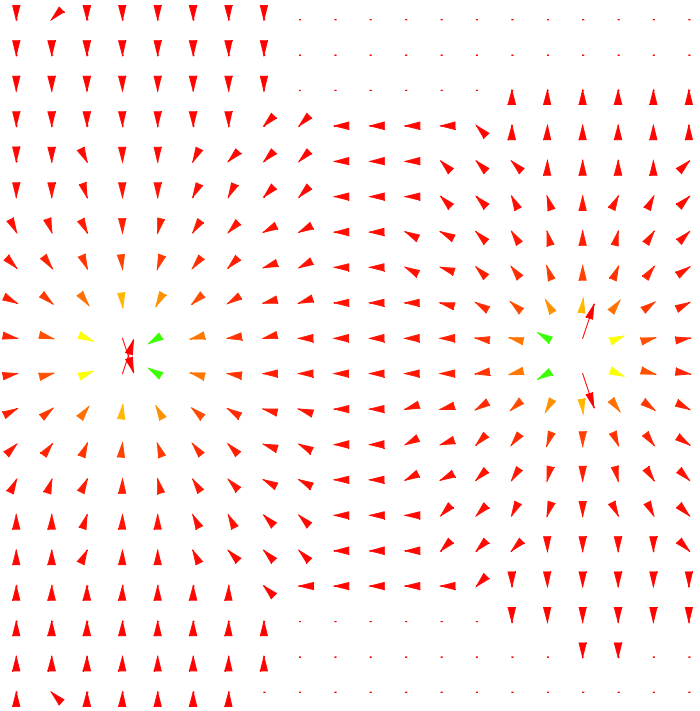


Figure 4.2.4. Contour plot of the energy density of two charges in the (x, z)-plane.

The electrical field of the two charges are generated by

```
FieldPlot[charges, "Field"];
```



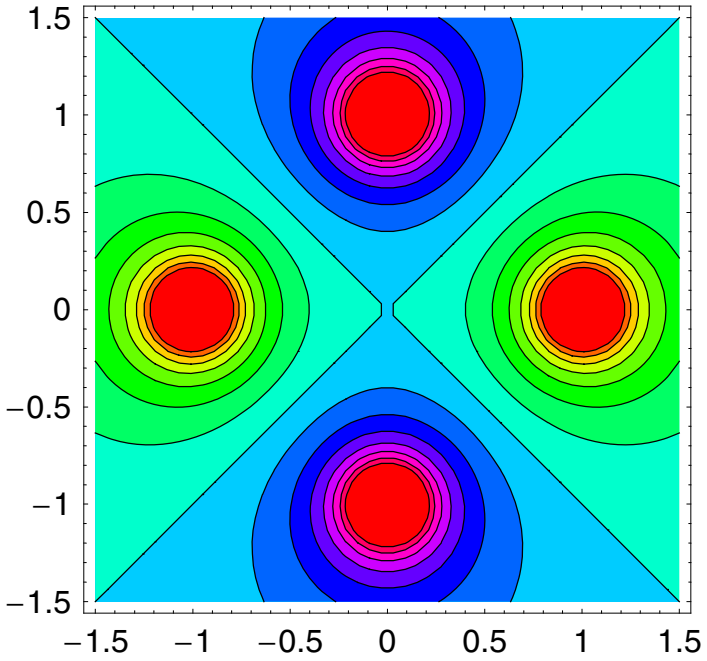
Since the generation of field plots is very flexible, we are able to examine any configuration of charges in space. A second example is given by a quadruple consisting of four charges arranged in a spatial configuration. The locations and strength of the charges are defined by

```
quadrupole = {{-1, 0, 0, -1},
  {1, 0, 0, -1}, {0, 0, 1, 1}, {0, 0, -1, 1}}
```

```
{{-1, 0, 0, -1}, {1, 0, 0, -1},
 {0, 0, 1, 1}, {0, 0, -1, 1}}
```

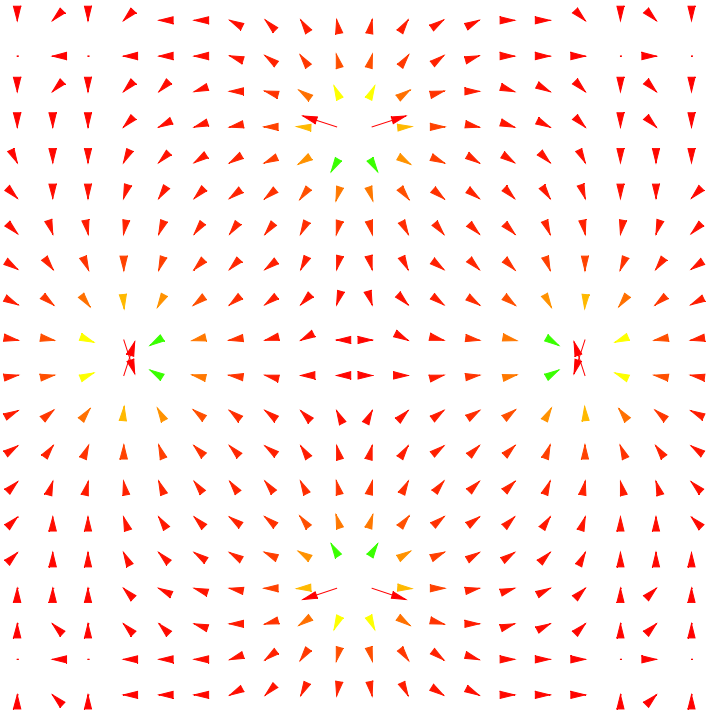
The potential is

```
FieldPlot[quadrupole, "Potential"];
```



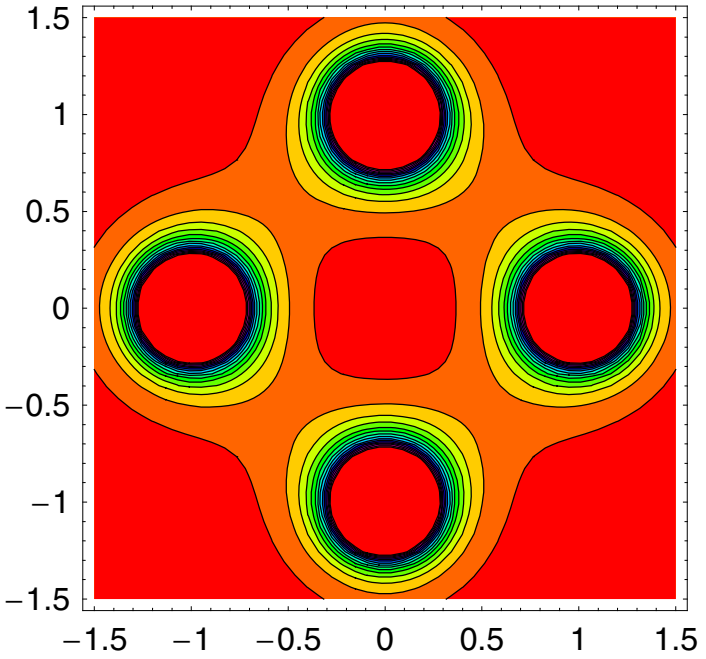
The field lines in the (x, z)-plane with $y = 0$ are

```
FieldPlot[quadrupole, "Field"];
```



The energy density looks like

```
FieldPlot[quadrupole, "EnergyDensity"];
```



4.3 Boundary Problem of Electrostatics

In the previous section, we discussed the arrangement of discrete charges. The problem was solved by means of the Poisson equation for the general case. We derived the solution for the potential using

$$\Delta\phi = 4\pi\rho. \tag{4.3.10}$$

Equation (4.3.10) is reduced to the Laplace equation if no charges are present in the space:

$$\Delta\phi = 0. \tag{4.3.11}$$

The Laplace equation is a general type of equation applicable to many different theories in physics, such as continuum theory, gravitation, hydrodynamics, thermodynamics, and statistical physics. In this section, we use both the Poisson and the Laplace equations (4.3.10) and (4.3.11) to

describe electrostatic phenomena. We show that Eqs. (4.3.10) and (4.3.11) are solvable by use of Green's function. If we know the Green's function of the equation, we are able to consider general boundary problems. A boundary problem is defined as follows: For a certain volume V, the surface of this volume, ∂V, possesses a specific electric potential. The problem is to determine the electric potential inside the volume given the value on the surface. This type of electrostatic boundary problem is called a Dirichlet boundary value problem. According to Eq. (4.3.10), there are charges inside volume V. The distribution or density of these charges is denoted by $\rho(\vec{x})$. The mathematical problem is to find solutions for Eq. (4.3.10) or (4.3.11) once we know the distribution of charges and the electric potential on the surface of the domain.

The Green's function allows us to simplify the solution of the problem. In our problem, we have to solve the Poisson equation (4.3.10) under certain restrictions. The Green's function related to the Poisson problem is defined by

$$\Delta G(\vec{x}, \vec{x}') = -4\pi\delta(\vec{x} - \vec{x}') \tag{4.3.12}$$

under the specific boundary condition

$$G(\vec{x}, \vec{x}')\Big|_{\partial V} = 0 \quad \text{with} \quad \vec{x}' \in \partial V \tag{4.3.13}$$

on the surface ∂V of volume V.

In the previous section, we discussed the Green's function for an infinitely extended space and found that the Green's function is represented by $G(\vec{x}, \vec{x}') = 1/|\vec{x} - \vec{x}'|$. The present problem is more complicated than the one previously discussed. We need to satisfy boundary conditions for a finite domain in space.

For our discussion, we assume that the Green's function exists and that we can use it to solve the boundary problem. The proof of this assumption is given by Arfken [4.1]. The connection between the Green's function and the solution of the boundary problem is derived using Gauss's theorem. The first formula by Green

$$\int_V \operatorname{div} \vec{A}\, d^3x = \int_V \vec{A}\, d^2\vec{f}, \tag{4.3.14}$$

along with an appropriate representation of the vector field $\vec{A} = \Phi \cdot \nabla G - \nabla\Phi \cdot G$ yields the second formula by Green:

$$\operatorname{div} \vec{A} = \Phi \cdot \Delta\, G - \Delta\Phi \cdot G. \tag{4.3.15}$$

Using the integral theorem of Gauss in the form of Eq. (4.3.14), we find

$$\int_V (\Phi \cdot \Delta\, G - \Delta\Phi \cdot G)\, d^3 x = \int_{\partial V} \left(\Phi \frac{\partial G}{\partial n} - G \frac{\partial \Phi}{\partial n}\right) d^2 f, \tag{4.3.16}$$

where $\partial / \partial n = \vec{n} \cdot \nabla$ is the normal gradient. If we use relations (4.3.10), (4.3.12), and (4.3.13) in Eq. (4.3.16), we can derive the potential by the two integrals

$$\Phi(\vec{x}) = \int_V G(\vec{x}, \vec{x}')\, \rho(\vec{x}')\, d^3 x' - \frac{1}{4\pi} \int_{\partial V} \Phi(\vec{x}') \frac{\partial G(\vec{x}, \vec{x}')}{\partial n'} d^2 f'. \tag{4.3.17}$$

A comparison between Eqs. (4.3.17) and (4.2.3) reveals that the total potential in the Dirichlet problem depends on a volume part (consistent with Eq. (4.2.3) and on a surface part as well. The potential Φ at location $\vec{x}$ consists of a volume term containing the charges and of a surface term determined by the electric potential $\Phi(\vec{x})$. The potential $\Phi(\vec{x}')$ used in the surface term is known as a boundary condition. If there are no charges in the present volume, solution (4.3.17) reduces to

$$\Phi(\vec{x}) = -\frac{1}{4\pi} \int_{\partial V} \Phi(\vec{x}') \frac{\partial G(\vec{x}, \vec{x}')}{\partial n'} d^2 f'. \tag{4.3.18}$$

For the charge-free case, the electric potential at a location $\vec{x}$ inside the volume V is completely determined by the potential on the surface $\Phi(\vec{x}')$. We are able to derive Eqs. (4.3.17) and (4.3.18) provided that the Green's function $G(\vec{x}, \vec{x}')$ vanishes on the surface of V. In other words, we assume the surface potential to be a boundary condition. This type of boundary condition is called a Dirichlet boundary condition. A second type is the so-called von Neumann boundary condition, which specifies the normal derivative of the electrostatic potential $\partial\Phi / \partial n$ on the surface. A third type used in potential theory is a mixture of Dirichlet and von Neumann boundary conditions. In the following, we will restrict ourselves to Dirichlet boundary conditions only.

If we take a closer look at solutions (4.3.17) and (4.3.18) of our boundary value problem, we observe that the Green's function as an unknown

determines the solution of our problem. In other words, we solved the boundary problem in a form which contains an unknown function as defined by relation (4.3.12) and the boundary condition (4.3.13). The central problem is to find an explicit representation of the Green's function. One way to tackle this is by introducing an eigenfunction expansion [4.2]. This procedure always applies if the coordinates are separable. The eigenfunction expansion of the Green's function is based on the analogy between an eigenvalue problem and equations (4.3.10) and (4.3.11) for the potential.

The eigenvalue problem related to equation (4.3.10) is given by

$$\Delta\psi+(4\pi\rho+\lambda)\psi = 0. \tag{4.3.19}$$

For a detailed discussion of the connection, see [4.2]. We assume that solutions ψ of Eq. (4.3.19) satisfy the Dirichlet boundary conditions. In this case, the regular solutions of Eq. (4.3.12) only occur if parameter $\lambda = \lambda_n$ assumes certain discrete values. The λ_n's are the eigenvalues of Eq. (4.3.19). Their corresponding functions ψ_n are eigenfunctions. The eigenfunctions ψ_n are orthogonal and satisfy

$$\int_V \psi_m^*(\vec{x})\,\psi_n(\vec{x})\,d^3x = \delta_{mn}. \tag{4.3.20}$$

The eigenvalues of Eq. (4.3.19) can be discrete or continuous. In analogy to Eq. (4.3.12), the Green's function has to satisfy the equation

$$\Delta_x G(\vec{x},\ \vec{x}') + (4\pi\rho + \lambda)\,G(\vec{x},\ \vec{x}') = -4\pi\delta(\vec{x}-\vec{x}'), \tag{4.3.21}$$

where λ is different to the eigenvalues λ_n. An expansion of the Green's function with respect to the eigenfunctions of the related eigenvalue problem is possible if the Green's function satisfies the same boundary conditions. Substituting an expansion of the Green's function

$$G(\vec{x},\ \vec{x}') = \sum_n a_n(\vec{x}')\,\psi_n(\vec{x}) \tag{4.3.22}$$

into Eq. (4.3.21), we get

$$\sum_m a_m(\vec{x}')(\lambda-\lambda_m)\,\psi_m(\vec{x}) = -4\pi\delta(\vec{x}-\vec{x}'). \tag{4.3.23}$$

Multiplying both sides of Eq. (4.3.23) by $\psi_n^*(\vec{x})$ and integrating the result over the entire volume, we obtain the expansion coefficients $a_m(\vec{x}')$. Using the orthogonal relation (4.3.20) simplifies the sum. The expansion coefficients are defined by

$$a_n(\vec{x}\,') = 4\pi \frac{\psi_n^*(\vec{x}\,')}{\lambda_n - \lambda}. \tag{4.3.24}$$

With relation (4.3.24) we get the representation of the Green's function

$$G(\vec{x}, \vec{x}\,') = 4\pi \sum\nolimits_n \frac{\psi_n^*(\vec{x}\,')\,\psi_n(\vec{x})}{\lambda_n - \lambda}. \tag{4.3.25}$$

So far, our considerations have assumed a discrete spectrum of eigenvalues. For a continuous distribution of eigenvalues λ_n, we need to replace the sum in Eq. (4.3.25) with an integral over the eigenvalues.

By using the representation of the Green's function (4.3.25), we can rewrite the solution of the potential (4.3.17) and (4.3.18) in the form

$$\begin{aligned}\Phi(\vec{x}) = &\int_V 4\pi \sum\nolimits_n \frac{\psi_n^*(\vec{x}\,')\,\psi_n(\vec{x})}{\lambda_n - \lambda}\,\rho(\vec{x}\,')\,d^3x' - \\ &\int_{\partial V} \Phi(\vec{x}\,') \sum\nolimits_n \frac{\psi_n(\vec{x})}{\lambda_n - \lambda}\,\frac{\partial \psi_n^*(\vec{x}\,')}{\partial n'}\,d^2f' \\ = &\,4\pi \sum\nolimits_n \frac{\psi_n(\vec{x})}{\lambda_n - \lambda} \int_V \psi_n^*(\vec{x}\,')\,\rho(\vec{x}\,')\,d^3x' - \\ &\sum\nolimits_n \frac{\psi_n(\vec{x})}{\lambda_n - \lambda} \int_{\partial V} \Phi(\vec{x}\,')\,\frac{\partial \psi_n^*(\vec{x}\,')}{\partial n'}\,d^2f'.\end{aligned} \tag{4.3.26}$$

If we know the eigenfunctions and eigenvalues of the problem, we can represent the potential by

$$\Phi(\vec{x}) = \sum\nolimits_n (c_n - d_n)\,\psi_n(\vec{x}), \tag{4.3.27}$$

where the c_n's and the d_n's are expansion coefficients defined by

$$c_n = \frac{4\pi}{\lambda_n - \lambda} \int_V \psi_n^*(\vec{x}\,')\,\rho(\vec{x}\,')\,d^3x' \tag{4.3.28}$$

and

$$d_n = \frac{1}{\lambda_n - \lambda} \int_{\partial V} \Phi(\vec{x}\,')\,\frac{\partial \psi_n^*(\vec{x}\,')}{\partial n'}\,d^2f'. \tag{4.3.29}$$

For the charge-free case $\rho = 0$, we find

$$\Phi(\vec{x}) = -\sum\nolimits_n \frac{\psi_n(\vec{x})}{\lambda_n - \lambda} \int_{\partial V} \Phi(\vec{x}\,')\,\frac{\partial \psi_n^*(\vec{x}\,')}{\partial n'}\,d^2f'. \tag{4.3.30}$$

which reduces to

$$\Phi(\vec{x}) = -\sum\nolimits_n d_n\,\psi_n(\vec{x}). \tag{4.3.31}$$

The unknown quantities of this representation are the eigenfunctions ψ_n and the expansion coefficients c_n and d_n. By examining a specific planar

problem, we show how these unknowns are calculated. To make things simple, we assume that no charges are distributed on the plane.

The problem under consideration examines in a section of a disk in which boundaries have fixed potential values $\Phi(r, \varphi = 0) = 0$, $\Phi(r, \varphi = \alpha) = 0$, and $\Phi(r = R, \varphi) = \Phi_0(\varphi)$. The specific form of the domain and the boundary values are given in Figure 4.3.5.

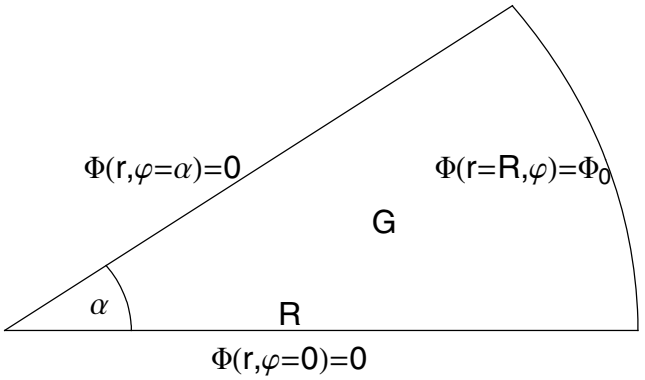


Figure 4.3.5. Boundary conditions on a disk segment. The domain G is free of charges.

The domain G is free of any charges and the potential $\Phi(r, \varphi)$ is regular and finite for $r \to 0$. To solve the problem efficiently, we choose coordinates which reflect the geometry of our problem. In this case, they are plane cylindrical coordinates. Since G is free of any charges, Laplace's equation in plane cylindrical coordinates takes the form

$$\frac{1}{r}\,\frac{\partial}{\partial r}\left(r\,\frac{\partial \Phi}{\partial r}\right) + \frac{1}{r^2}\,\frac{\partial^2 \Phi}{\partial \varphi^2} = 0. \qquad (4.3.32)$$

When deriving the solution, we assume that the coordinates are separated. If we use the assumption of separating the coordinates, we are able to express the electric potential as $\Phi(r, \varphi) = g(r)\,h(\varphi)$. Substituting this expression into Eq. (4.3.32), we get

$$\frac{r}{g(r)}\,\frac{d}{d\,r}\left(r\,\frac{d\,g}{d\,r}\right) = -\frac{1}{h(\varphi)}\,\frac{d^2\,h(\varphi)}{d\,\varphi^2} = \nu^2, \qquad (4.3.33)$$

where ν is a constant. Separating both equations, we get two ordinary differential equations determining g and h. g and h represent the eigenfunctions of the Green's function

$$\frac{r}{g(r)} \frac{d}{dr}\left(r \frac{dg}{dr}\right) = \nu^2, \tag{4.3.34}$$

$$\frac{1}{h(\varphi)} \frac{d^2 h(\varphi)}{d\varphi^2} = -\nu^2, \tag{4.3.35}$$

The eigenfunctions of the radial part of the potential are

$$g_\nu(r) = a_\nu r^\nu + b_\nu r^{-\nu}. \tag{4.3.36}$$

The angular part of the eigenfunctions defined in Eq. (4.3.35) is given by

$$h_\nu(\varphi) = A_\nu \sin(\nu\varphi) + B_\nu \cos(\nu\varphi). \tag{4.3.37}$$

The solutions (4.3.36) and (4.3.37) contain four constants a_ν, b_ν, A_ν, and B_ν for each eigenvalue ν. These constants have to satisfy the boundary conditions and the condition of regularity at $r = 0$.

Let us first examine the radial part of the solution in the domain G. We find that for $\varphi = 0$, the relation

$$\Phi(r, \varphi = 0) = g(r)\, h(\varphi = 0) = 0 \tag{4.3.38}$$

needs to be satisfied. From condition (4.3.38), it follows that $h(\varphi = 0) = B_\nu = 0$. From the boundary condition at $\varphi = \alpha$ we get the condition

$$\Phi(r, \varphi = \alpha) = g(r)\, h(\varphi = \alpha) = 0, \tag{4.3.39}$$

which results in $h(\alpha) = A_\nu \sin(\nu\alpha) = 0$. As a consequence, we get $\nu = n\pi/\alpha$ with $n = 0, 1, 2, 3, \ldots$. The angular part of the solution thus reduces to

$$h_n(\varphi) = A_n \sin\left(\frac{n\pi}{\alpha}\varphi\right). \tag{4.3.40}$$

From the condition of regularity $\Phi(r \to 0, \varphi) < \infty$, it follows from

$$\Phi(r, \varphi) = h_\nu(\varphi)\,(a_\nu r^\nu + b_\nu r^{-\nu}) \tag{4.3.41}$$

that $b_\nu = 0$. The solution of the potential is thus represented by

$$\Phi(r, \varphi) = \sum_{n=0}^{\infty} d_n r^{n\pi/a} \sin\left(\frac{n\pi}{a}\varphi\right), \tag{4.3.42}$$

where $d_n = a_n A_n$. Expression (4.3.42) contains the unknown coefficients d_n, which we need to determine in order to find their explicit

representations. Values for d_n are determined by applying the boundary condition on the circle $\Phi(r = R, \varphi) = \Phi_0(\varphi)$. If we take into account the orthogonality relation for the trigonometric functions

$$\frac{2}{\alpha} \int_0^{\alpha} \sin(\frac{n\pi}{\alpha}\ \varphi) \sin(\frac{m\pi}{\alpha}\ \varphi)\, d\,\varphi = \delta_{mn}, \tag{4.3.43}$$

we are able to derive from the boundary condition of the circle a representation of d_n by

$$\begin{aligned} &\int_0^{\alpha} \Phi_0(\varphi) \sin(\frac{n\pi}{\alpha}\ \varphi)\, d\,\varphi = \\ &\quad \sum\nolimits_{m=0}^{\infty} d_m R^{m\pi/\alpha} \int_0^{\alpha} \sin(\frac{n\pi}{\alpha}\ \varphi) \sin(\frac{m\pi}{\alpha}\ \varphi)\, d\,\varphi \\ &\qquad\qquad = \sum\nolimits_{m=0}^{\infty} d_m R^{m\pi/\alpha}\ \frac{\alpha}{2}\ \delta_{nm} \\ &\qquad\qquad = \frac{\alpha}{2}\ R^{n\pi/\alpha}\ d_n, \end{aligned} \tag{4.3.44}$$

or in explicit form,

$$d_n\ = R^{-n\pi/\alpha}\ \frac{2}{\alpha}\ \int_0^{\alpha} \Phi_0(\varphi) \sin(\frac{n\pi}{\alpha}\ \varphi)\, d\,\varphi. \tag{4.3.45}$$

The representation of d_n by the integral (4.3.45) includes the boundary condition and only contains known parameters. Thus, we can determine d_n's numerical value if we know the boundary condition and if we specify the index m of the expansion in Eq. (4.3.42). The values of d_n are, however, only defined if the integral in Eq. (4.2.45) converges. The specific form of the Green's function is derivable if we compare the representation of the solution (4.3.42) with the definition of the Green's function.

With the above theoretical considerations, an explicit representation of the solution is now necessary. By specifying the geometrical parameters of the problem, the radius R of the segment, the angle α, the potential value along the rim of the disk and Eq. (4.3.42), we can calculate the potential in the domain G. The central quantities of the expansion (4.3.42) are the coefficients d_n. In order to make these factors available, we define the sum (4.3.42) and the integral (4.3.45) in the **Potential[]** function of the package **BoundaryProblem`** (see Section 4.6.2 for details). We define relations (4.3.42) and (4.3.45) to control the accuracy of the calculation using an upper summation index n (see also the definition of the function **Potential[]** in Section 4.6.2). An example of the potential for the parameters R=1, $\alpha = \pi/4$ and $\Phi_0(\varphi)$=1 is given in Figure 4.3.6. The calling sequence of **Potential[]** takes the form **Potential[*f*[*x*], *R*, α, *n*]**.

```
Potential[1, 1, π/4, 10];
```

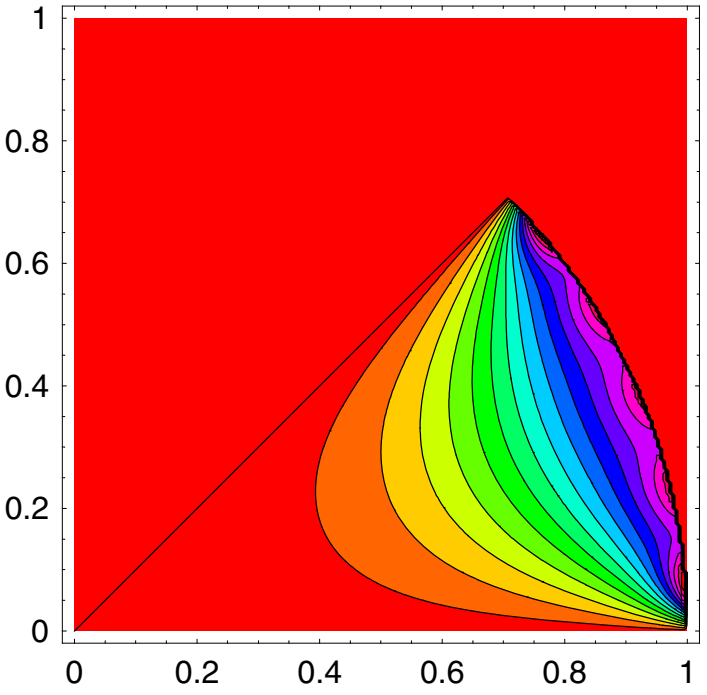


Figure 4.3.6. Contour plot of the potential in the domain *G*. Boundary conditions and geometric parameters are $\Phi_0(\varphi)= 1$, $R=1$, $\alpha = \pi/4$ and $n=10$.

The result shows an approximation of the potential up to order 10. The contour lines show that the approximation shows some wiggles at the rim of the domain. The quality of the approximation can be checked by increasing the approximation order. The increase in quality is shown in the following sequence of plots (Figure 4.3.7):

```
pl = Table[Potential[1, 1, π/4, i], {i, 1, 20, 2}];
```

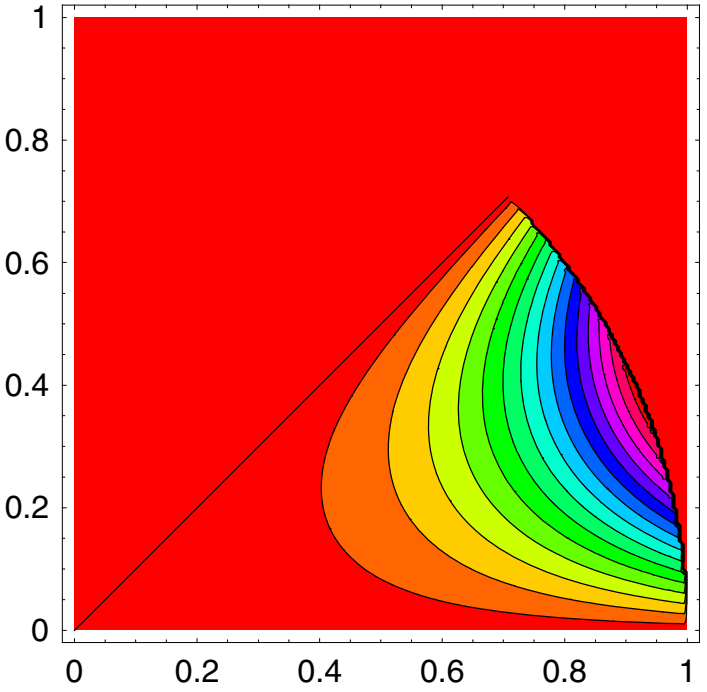


Figure 4.3.7. Sequence of contour plot of the potential in the domain *G*. Boundary conditions and geometric parameters are $\Phi_0(\varphi)= 1$, $R=1$, $\alpha = \pi/4$ and $n\epsilon[1,20,2]$.

At this place, a word of caution should be mentioned. The approximation of the potential shows that the procedure is sensitive in the approximation order. The kind of calculation is also sensitive on the boundary conditions, which is given as first argument in the function **Potential[]**. Although the calculated potential shows the expected behavior, it is not always possible

to calculate the potential for a reasonable approximation order for arbitrary boundary conditions. This shortcoming is due to the calculation of integrals in the procedure. However, the reader should experiment with the function and test the limitations of the method to gain a feeling for the applicability. An example with a spatially varying boundary condition on the rim is presented in Figure 4.3.8.

```
Potential[2 + Sin[7 ϕ], 1, π/4, 20];
```

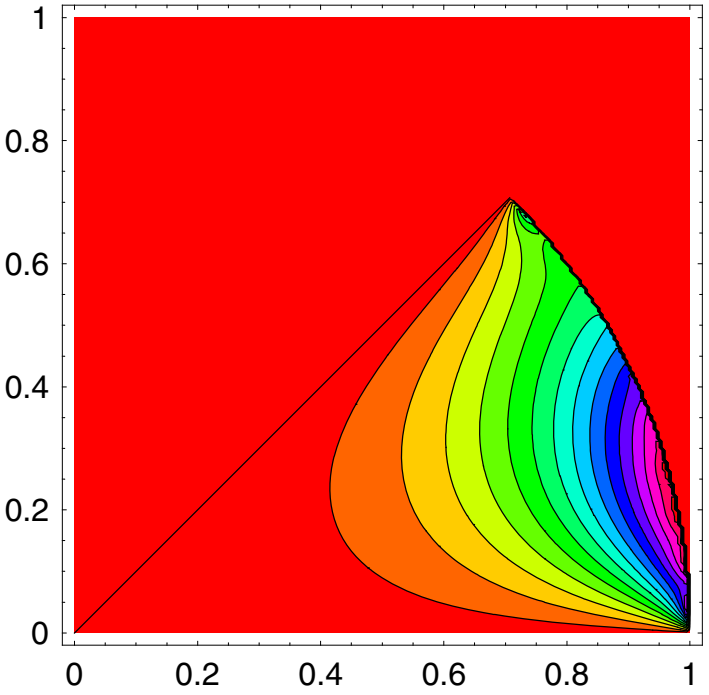


Figure 4.3.8. Contour plot of the potential in the domain G. Boundary conditions and geometric parameters are $\Phi_0(\varphi)= 2+\sin(7\varphi)$, R=1, $\alpha = \pi/4$ and n=10.

4.4 Two Ions in the Penning Trap

The study of spectroscopic properties of single ions requires that one or two ions are trapped in a cavity. Nowadays, ions can be successfully separated and stored by means of ion traps. Two techniques are used for trapping ions. The first method uses a dynamic electric field, while the second method uses static electric and magnetic fields. The dynamic trap was originally invented by Paul [4.3]. The static trap is based on the work of Penning [4.4]. Both traps use a combination of electric and magnetic fields to confine ions in a certain volume in space. Two paraboloids connected to a *dc*-source determine the kind of electric field in which the ions are trapped. The form of the paraboloids in turn determines the field of the trap's interior. Since the motion of the ions in Paul's trap is very complicated, we restrict our study to the Penning trap.

In our discussion of the Penning trap, the form of the quadrupole fields determined by the shapes of the paraboloids is assumed to be

$$\Phi = \frac{U_0}{r_0^2 + 2\, z_0^2}\,(x^2 + y^2 - 2\, z^2), \tag{4.4.46}$$

where U_0 is the strength of the source and r_0 and z_0 are the radial and axial extensions of the trap (see Figure 4.4.9). The shape of the potential is a consequence of the Laplace equation $\Delta\Phi=0$. The given functional shape of the potential is experimentally created by conducting walls which are connected to a dc-battery. The force acting on an ion carrying charge q in the trap is given by

$$\vec{F} = q\,\vec{E} = -q\,\nabla\Phi. \tag{4.4.47}$$

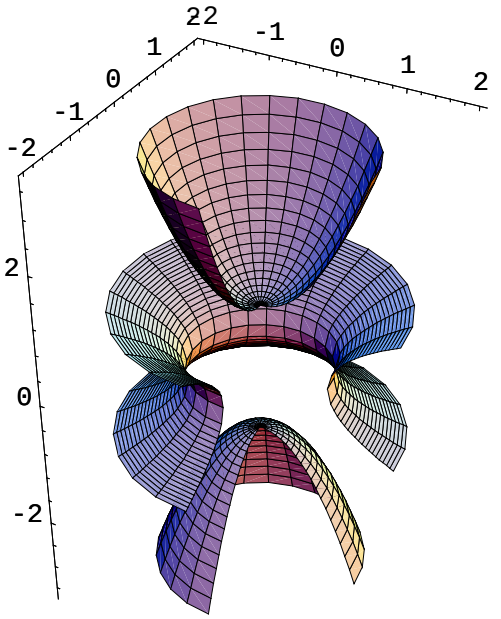


Figure 4.4.9. Cross-section of the Penning trap. The paraboloids are positioned on dc-potentials. A constant magnetic field is superimposed in the z vertical direction (not shown). The ions move in the center of the trap.

From the functional form of the electric field $\vec{E}$ of the trap

$$\vec{E} = -\nabla\Phi = -\frac{2\,U_0}{r_0^2+2\,z_0^2}\begin{pmatrix} x \\ y \\ -2\,z \end{pmatrix} = -\frac{2\,U_0}{r_0^2+2\,z_0^2}\,(\vec{x} - 3\,\vec{e}_z), \tag{4.4.48}$$

we detect a change of sign in the coordinates. This instability allows the ions to escape the trap. To prevent escape from the trap in the z-direction, Paul and co-workers used a high-frequency ac-field and Penning and co-workers used a permanent magnetic field $\vec{B} = B_0\vec{e}_z$.

In a static trap the forces acting on each of the two ions are determined by the electromagnetic force of the external fields and the repulsive force of the Coulomb interaction of the charges. The external fields consist of the static magnetic field along the z-axis and the electric quadrupole field of the trap. The Coulomb interaction of the two particles is mainly governed

by the charges which are carried by the particles. The total force on each particle is a combination of trap and Coulomb forces. Since we have a system containing only a few particles, we can use Newton's theory (see section 2.4) to write down the equations of motion in the form

$$m\,\vec{x}'' = (\vec{F})_i^{\,T} + (\vec{F})_i^{\,\mathrm{Coul}} \qquad i=1,2. \tag{4.4.49}$$

In equation (4.4.49) the trap force $(\vec{F})_i^{\,T}$ denotes the Lorentz force of a particle in the electromagnetic field given by

$$(\vec{F})_i^{\,T} = q\,(\vec{E})_i + q\,(\vec{v}_i \times \vec{B}). \tag{4.4.50}$$

Since the magnetic field $\vec{B}$ is a constant field along the z-direction

$$\vec{B} = B_0 \vec{e}_z, \tag{4.4.51}$$

the total trap force on the ith ion is given by

$$(\vec{F})_i^{\,T} = -\frac{2\,U_0}{r_0^2 + 2\,z_0^2}\,(\vec{x} - 3\,z_i\,\vec{e}_z) + q\,(\vec{x}'_i \times \vec{B})\,. \tag{4.4.52}$$

The Coulomb forces between the first and the second ion are

$$(\vec{F})_{12}^{\,\mathrm{Coul}} = \frac{q^2}{4\pi\varepsilon_0}\,\frac{\vec{x}_1 - \vec{x}_2}{|\vec{x}_1 - \vec{x}_2|^3}\,, \tag{4.4.53}$$

$$(\vec{F})_{21}^{\,\mathrm{Coul}} = \frac{q^2}{4\pi\varepsilon_0}\,\frac{\vec{x}_2 - \vec{x}_1}{|\vec{x}_1 - \vec{x}_2|^3}\,. \tag{4.4.54}$$

The explicit forms of the equations of motion are thus

$$m\,\vec{x}''_1 = -\frac{2\,U_0}{r_0^2 + 2\,z_0^2}\,(\vec{x}_1 - 3\,z_1\,\vec{e}_z) + q\,(\vec{x}'_1 \times \vec{B}) + \frac{q^2}{4\pi\varepsilon_0}\,\frac{\vec{x}_1 - \vec{x}_2}{|\vec{x}_1 - \vec{x}_2|^3}\,, \tag{4.4.55}$$

$$m\,\vec{x}''_2 = -\frac{2\,U_0}{r_0^2 + 2\,z_0^2}\,(\vec{x}_2 - 3\,z_2\,\vec{e}_z) + q\,(\vec{x}'_2 \times \vec{B}) + \frac{q^2}{4\pi\varepsilon_0}\,\frac{\vec{x}_2 - \vec{x}_1}{|\vec{x}_1 - \vec{x}_2|^3}\,. \tag{4.4.56}$$

The two equations of motion (4.4.55) and (4.4.56) are coupled ordinary differential equations of the second order. They can be decoupled by introducing relative and center of mass coordinates:

$$\begin{aligned} \vec{r} &= \vec{x}_1 - \vec{x}_2, \\ \vec{R} &= \tfrac{1}{2}\,(\vec{x}_1 + \vec{x}_2). \end{aligned} \tag{4.4.57}$$

Using Eqs. (4.4.57) in (4.4.55) and (4.4.56), we can describe the motion of the two ions in the center of mass and in relative coordinates. The two transformed equations read

$$\vec{R}'' = -\frac{2\,U_0}{m(r_0^2+2\,z_0^2)}\left(\vec{R}-3\,Z\,\vec{e}_z\right)+\frac{q\,B_0}{m}\left(\vec{R}'\times\vec{e}_z\right), \tag{4.4.58}$$

$$\vec{r}'' = -\frac{2\,U_0}{m(r_0^2+2\,z_0^2)}\,(\vec{r}-3\,z\,\vec{e}_z)+ \frac{q\,B_0}{m}\,(\vec{r}'\times\vec{e}_z)+\frac{q^2}{2\,\pi\,m\,\varepsilon_0}\,\frac{\vec{r}}{|\,\vec{r}\,|^3}. \tag{4.4.59}$$

If we assume that the two ions carry a negative charge $q<0$ and that the dc-potential U_0 on the paraboloids is positive ($U_0>0$), then we can introduce two characteristic frequencies and a scaled charge by

$$\omega_0^2 = \frac{2\,U_0}{m(r_0^2+2\,z_0^2)}, \tag{4.4.60}$$

$$\omega_c = \frac{|q|\,B_0}{m}, \tag{4.4.61}$$

$$Q^2 = \frac{q^2}{2\,\pi\,m\,\varepsilon_0}. \tag{4.4.62}$$

Constant ω_0 denotes the frequency of the oscillations along the z-direction. ω_c is the cyclotron frequency (i.e., the frequency with which the ions spin around the magnetic field). Q represents the scaled charge. Using these constants in the equations of motion (4.4.58) and (4.4.59), we get a simplified system of equations containing only three constants:

$$\vec{R}'' = \omega_0^2\left(\vec{R}-3\,Z\,\vec{e}_z\right)-\omega_c\left(\vec{R}'\times\vec{e}_z\right), \tag{4.4.63}$$

$$\vec{r}'' = \omega_0^2\,(\vec{r}-3\,z\,\vec{e}_z)-\omega_c\,(\vec{r}'\times\vec{e}_z)+Q^2\,\frac{\vec{r}}{|\,\vec{r}\,|^3}. \tag{4.4.64}$$

In the following subsections, we discuss the two different types of motion resulting from these equations.

4.4.1 The Center of Mass Motion

The center of mass motion is determined by Eq. (4.4.63). Writing down the equations of motion in cartesian coordinates X, Y, and Z, we get a coupled system of equations:

$$X'' - \omega_0^2\,X + \omega_c\,Y' = 0, \tag{4.4.65}$$

$$Y'' - \omega_0^2\,Y - \omega_c\,X' = 0, \tag{4.4.66}$$

$$Z'' + 2\,\omega_0^2\,Z = 0. \tag{4.4.67}$$

The equations of motion for the X- and Y- components are coupled through the cross-product. The Z- component of the motion is completely decoupled from the X and Y coordinates. The last of these three equations is equivalent to a harmonic oscillator with frequency $\sqrt{2}\,\omega_0$. Thus, we immediately know the solution of the Z- coordinate given by

$$Z(t) = A\cos\left(\sqrt{2}\,\omega_0 t + B\right). \tag{4.4.68}$$

The arbitrary constants A and B are related to the initial conditions of the motion by $Z(t=0) = Z_0$ and $Z'(t=0)=Z_0'$. Therefore, $A = Z_0^2 + Z'^2_0/2\,\omega_0^2$ and $\tan B = Z_0' \big/ \sqrt{2}\,\omega_0 Z_0$.

A representation of the solution of the remaining two equations (4.4.65) and (4.4.66) follows if we combine the two coordinates X and Y by a complex transformation of the form $\mathcal{Y} = X + i\,Y$. Applying this transformation to the two equations delivers the simple representation

$$\ddot{\mathcal{Y}} - \omega_0^2\,\mathcal{Y} - i\,\omega_c\,\dot{\mathcal{Y}} = 0. \tag{4.4.69}$$

If we assume that the solutions of Eq. (4.4.69) are harmonic functions of the type $\mathcal{Y} = e^{i\,\omega\,t}$, we get the corresponding characteristic polynomial

$$\omega(\omega_c - \omega) - \omega_0^2 = 0. \tag{4.4.70}$$

The two solutions of this quadratic equation are given by the frequencies ω_1 and ω_2:

$$\omega_1 = \tfrac{\omega_c}{2} + \sqrt{\left(\tfrac{\omega_c}{2}\right)^2 - \omega_0^2}\,, \tag{4.4.71}$$

$$\omega_2 = \tfrac{\omega_c}{2} - \sqrt{\left(\tfrac{\omega_c}{2}\right)^2 - \omega_0^2}\,. \tag{4.4.72}$$

The two frequencies are combinations of the cyclotron frequency ω_c and the axial frequency ω_0. The general solution of Eqs. (4.4.65) and (4.4.66) is thus given by

$$X(t) = B_r\cos(\omega_1 t) + B_i\sin(\omega_1 t) + A_r\cos(\omega_2 t) + A_i\sin(\omega_2 t), \tag{4.4.73}$$

$$Y(t) = A_r\sin(\omega_2 t) - A_i\cos(\omega_2 t) + B_r\sin(\omega_1 t) - B\cos(\omega_1 t). \tag{4.4.74}$$

The constants of integration A_r, A_i, B_r, and B_i are related to the initial conditions X_0, Y_0, $X_0^{'}$, and $Y_0^{'}$ by the relations

$$A_r = \frac{Y_0^{'} - \omega_1 X_0}{\omega_2 - \omega_1}, \tag{4.4.75}$$

$$A_i = \frac{X_0^{'} + \omega_1 Y_0}{\omega_2 - \omega_1}, \tag{4.4.76}$$

$$B_r = \frac{Y_0^{'} - \omega_2 X_0}{\omega_1 - \omega_2}, \tag{4.4.77}$$

$$B_i = \frac{X_0^{'} + \omega_2 Y_0}{\omega_1 - \omega_2}. \tag{4.4.78}$$

A special case of solutions (4.4.73) and (4.4.74) is obtained if we assume that the center of mass is initially located in the origin of the coordinate system $X_0 = Y_0 = 0$. We get from (4.4.75) $A_r = -B_r$, and $A_i = -B_i$. The solution then takes the form

$$\begin{aligned} X(t) \;=\; & A_r \sin\left(\frac{\omega_c}{2}\, t\right) \sin\left(\sqrt{\left(\frac{\omega_c}{2}\right)^2 - \omega_0^2}\; t\right) - \\ & A_i \cos\left(\frac{\omega_c}{2}\, t\right) \sin\left(\sqrt{\left(\frac{\omega_c}{2}\right)^2 - \omega_0^2}\; t\right), \end{aligned} \tag{4.4.79}$$

$$\begin{aligned} Y(t) \;=\; & A_i \sin\left(\frac{\omega_c}{2}\, t\right) \sin\left(\sqrt{\left(\frac{\omega_c}{2}\right)^2 - \omega_0^2}\; t\right) - \\ & A_r \cos\left(\frac{\omega_c}{2}\, t\right) \sin\left(\sqrt{\left(\frac{\omega_c}{2}\right)^2 - \omega_0^2}\; t\right). \end{aligned} \tag{4.4.80}$$

The above solutions show that the motion of the center of mass in the (X, Y)-plane is governed by two frequencies. The first frequency is one-half of the cyclotron frequency ω_c and the second frequency is a combination of the axial frequency and the cyclotron frequency given by $\sqrt{(\omega_c/2)^2 - \omega_0^2}$. A plot of the motion in center of mass coordinates is given in Figure 4.4.10. The three-dimensional motion of the center of mass is governed by three frequencies. The axial frequency $\sqrt{2}\,\omega_0$ determines the oscillation rate of the center of mass along the z-axis. The halved cyclotron frequency $\omega_c/2$ governs the spinning of the particles around the magnetic lines.

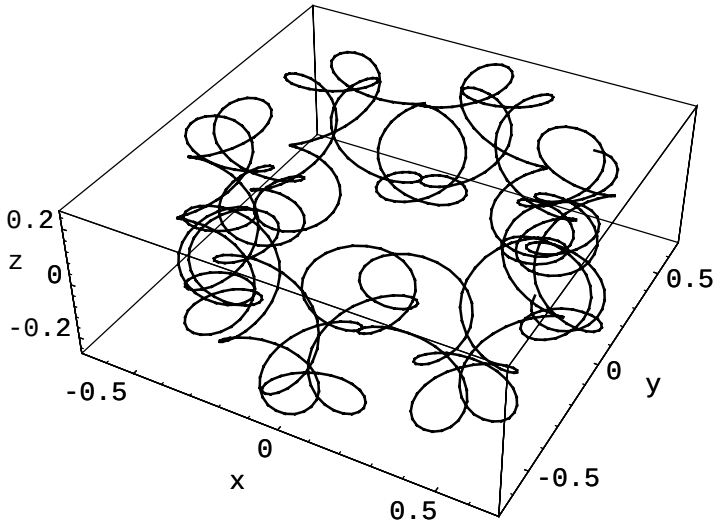


Figure 4.4.10. Motion of the center of mass in space for $t \in [0, 100]$. The initial conditions are $X_0 = 0.5 = Y_0$, $\dot{X}_0 = 0.1 = \dot{Y}_0$. The cyclotron frequency is fixed at ω_c=5.

4.4.2 Relative Motion of the Ions

The relative motion of the two ions is governed by Eq. (4.4.64)

$$\vec{r}'' = \omega_0^2 (\vec{r} - 3\,\overline{z}\vec{e}_z) - \omega_c (\vec{r}' \times \vec{e}_z) + Q^2 \frac{\vec{r}}{|\vec{r}|^3}. \tag{4.4.81}$$

Cylindrical coordinates are the appropriate coordinate system giving an efficient description of the relative motion of the particles. Location $\vec{r}$ of the relative particle is given in cylindrical coordinates by the representation

$$\vec{r} = \rho\, \vec{e}_\rho + \zeta\, \vec{e}_z, \tag{4.4.82}$$

where $\vec{e}_\rho$ and $\vec{e}_z$ represent the unit vectors in the radial and axial directions, respectively.

Using these coordinates in the equation of motion (4.4.81) gives the following representation:

$$\begin{aligned}(\rho'' - \rho\,\varphi'^2)\,\vec{e}_\rho + (2\,\rho'\,\varphi' + \rho\varphi'')\,\vec{e}_\varphi + \zeta''\,\vec{e}_z - \\ \omega_0^2(\rho\,\vec{e}_\rho - 2\,\zeta\,\vec{e}_z) + \omega_c(-\rho'\,\vec{e}_\varphi + \rho\varphi'\,\vec{e}_\rho) = \frac{Q^2(\rho\,\vec{e}_\rho + \zeta\,\vec{e}_z)}{\left(\sqrt{\rho^2+\zeta^2}\right)^3}.\end{aligned} \tag{4.4.83}$$

Separating each coordinate direction, we can split Eq. (4.4.83) into a system of equations for the coordinates ρ, φ, and ζ:

$$\rho'' - \rho\,\varphi'^2 - \omega_0^2\,\rho + \omega_c\,\rho\,\varphi' = \frac{Q^2\,\rho}{\left(\sqrt{\rho^2+\zeta^2}\right)^3}, \tag{4.4.84}$$

$$2\,\rho'\,\varphi' + \rho\,\varphi'' - \omega_c\,\rho = 0, \tag{4.4.85}$$

$$\zeta'' + 2\,\omega_0^2\,\zeta = \frac{Q^2\,\zeta}{\left(\sqrt{\rho^2+\zeta^2}\right)^3}. \tag{4.4.86}$$

By multiplying Eq. (4.4.85) by the radial coordinate ρ and integrating the result, we are able to derive an integral of motion. This integral of motion is given by an extended angular momentum containing the cyclotron frequency and is thus connected with the magnetic field. The conserved quantity is given by

$$\ell_B = \rho^2\,\varphi' - \frac{\omega_c}{2}\,\rho^2. \tag{4.4.87}$$

The integral of motion (4.4.87) eliminates the φ dependence in Eq. (4.4.84). The elimination of φ reduces the system of equations (4.4.84) and (4.4.86) to

$$\rho'' + \left(\left(\frac{\omega_c}{2}\right)^2 - \omega_0^2\right)\rho - \frac{\ell_B^2}{\rho^3} = \frac{Q^2\,\rho}{\left(\sqrt{\rho^2+\zeta^2}\right)^3}, \tag{4.4.88}$$

$$\zeta'' + 2\,\omega_0^2\,\zeta = \frac{Q^2\,\zeta}{\left(\sqrt{\rho^2+\zeta^2}\right)^3}. \tag{4.4.89}$$

This system of equations contains a multitude of parameters. Our aim is to reduce these parameters by appropriately scaling the temporal and spatial coordinates. If we consider the expression $\beta = (\omega_c/2)^2 - \omega_0^2 > 0$ to be positive, time is scaled by $\tau = \beta\,t$. The radial and axial coordinates ρ and ζ are scaled by the factor $d = (Q/\beta)^{\frac{2}{3}}$. Introducing the abbreviations $\nu^2 = (\ell_B/\beta)^2$ and $\lambda^2 = \left(\sqrt{2}\;\omega_0^2/\beta\right)^2$ simplifies the system of equations (4.4.88) and (4.4.89) to

$$\rho'' + \rho - \frac{\nu^2}{\rho^3} = \frac{\rho}{\left(\sqrt{\rho^2+\zeta^2}\right)^3}, \tag{4.4.90}$$

$$\zeta'' + \lambda^2\,\zeta = \frac{\zeta}{\left(\sqrt{\rho^2+\zeta^2}\right)^3}, \tag{4.4.91}$$

containing only two parameters ν and λ. The handling of Eqs. (4.4.90) and (4.4.91) is easier than the four parameter representation in equations (4.4.88) and (4.4.89). Note that Eqs. (4.4.90) and (4.4.91) are equivalent

to the secular equations of the Paul trap. Both systems of equations are derived from a Lagrangian given by

$$\mathcal{L} = \frac{1}{2}\,(\rho'^2 + \zeta'^2) - \left(\frac{1}{2}\,(\rho^2 + 2\,\lambda^2\,\zeta^2) + \frac{1}{\sqrt{\rho^2+\zeta^2}} + \frac{\nu^2}{2\,\rho^2}\right). \qquad (4.4.92)$$

Equations (4.4.90) and (4.4.91) form a highly nonlinear coupled system of equations which can only be solved analytically given a special choice of parameters λ and ν [4.5]. If we wish to choose parameters, we need to integrate the equations numerically. *Mathematica* supports numerical integrations and we use this property to find numerical solutions for Eqs. (4.4.90) and (4.4.91). The package **Penning`,** a listing is given in Section 4.6.3, contains the necessary function **PenningI[]** to integrate Eqs. (4.4.90) and (4.4.91). Function **PenningI[]** also provides a graphical representations of the potential and the path of the relative particle. An example of a typical path in the potential is given in Figure 4.4.11. Parameters λ and ν of this figure have been chosen so that the motion of the relative particle is regular. Figure 4.4.12 shows a path for parameters λ and ν where chaotic motion is present.

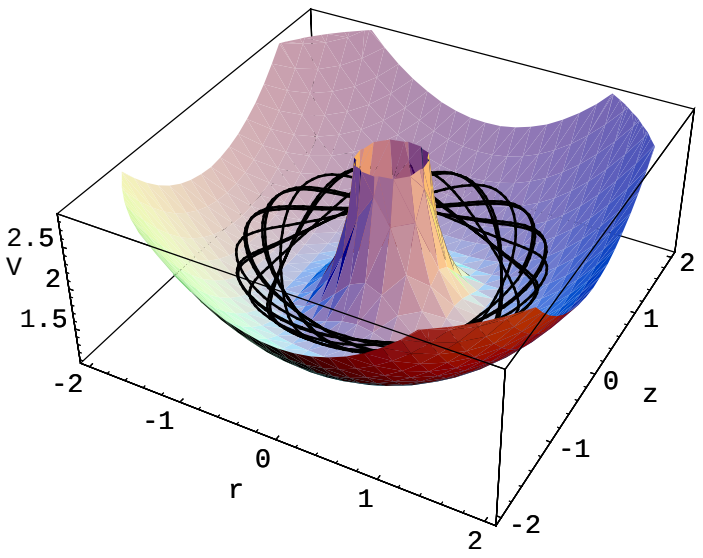


Figure 4.4.11. Relative motion in a Penning trap for $\lambda = 1$ and $\nu = 0$. The plot of the particle is superimposed on the effective potential. The numerical integration extends over $t \in [0, 100]$. The initial conditions are $\rho_0 = 1.1$, $\zeta_0 = 0.5$, $\dot{\rho}_0 = 0.0$, and $E = 2.0$.

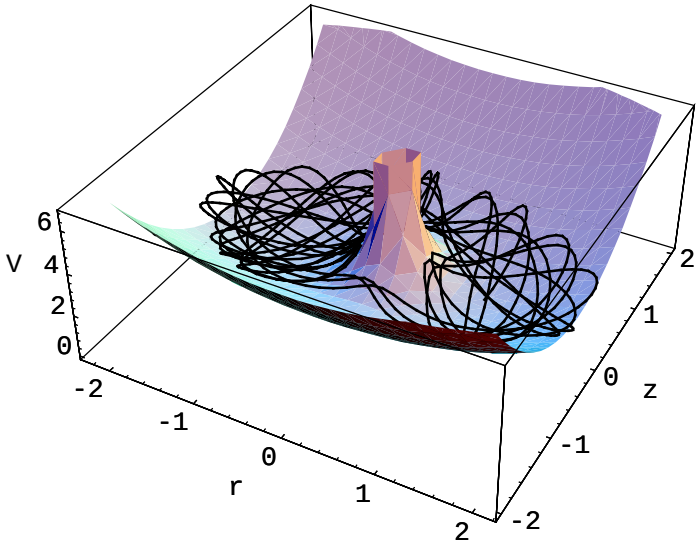
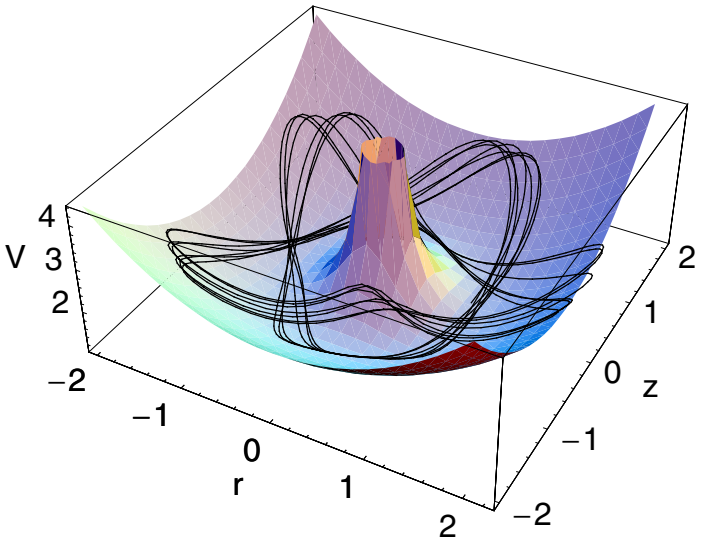


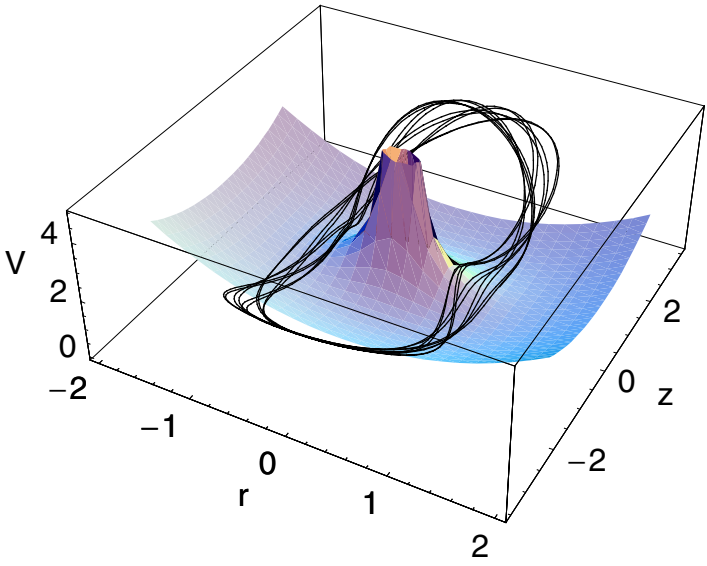
Figure 4.4.12. Relative motion in a Penning trap for $\lambda = 1.75$ and $\nu = 0$. The plot of the particle is superimposed on the effective potential. The numerical integration extends over $t \in [0, 100]$. Initial conditions are $\rho_0 = 1.0$, $\zeta_0 = 0.0$, $\dot{\rho}_0 = 0.0$, and $E = 3.0$.

Figures for different initial conditions and parameters can be generated for example by

```
PenningI[1.0, 0, 3, 0, 1.1, 100];
```

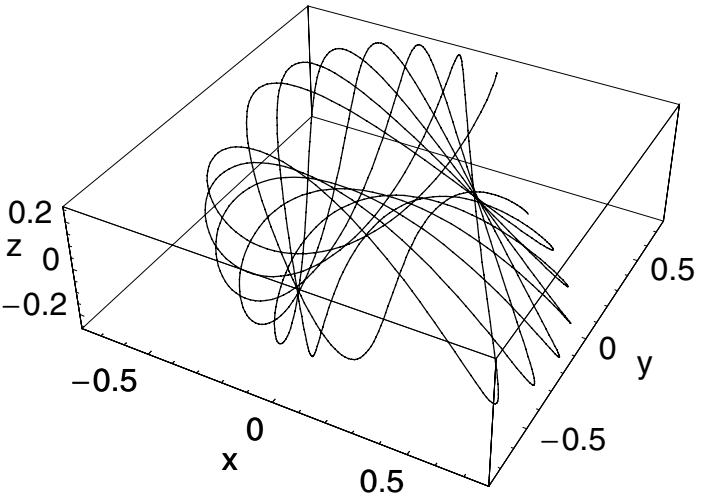



```
PenningI[1.0, 0.1, 3.6, 0, 1/Sqrt[2], 100];
```



The center of mass motion is accessible by the function **PenningCMPlot[]:**

```
PenningCMPlot[0.1, 0.2, 0.01, 0.01, 2.1];
```



4.5 Exercises

1. Create some pictures for a quadrupole arrangement of charges using the package **PointCharge'**. Choose the location of the charges in the representation plane of the potential section. What changes are required if your choice of coordinates for the charges is outside the representation plane? Perform some experiments with a larger number of charges.

2. Examine the electric potential of a disk segment under several boundary conditions using the package **BoundaryProblem'** (e.g., $\Phi_0 = \sin(\varphi)$ or $\Phi_0 = \varphi$). What changes occur in the potential if we change the angle α? Examine the influence of the upper summation index N on the accuracy of the solution.

3. Study the dynamic properties of two ions in a Penning trap for the following:

a) A vanishing angular momentum (ν=0) and different frequency ratios λ. Which λ values result in chaotic motion and in a regular motion of the particles?

b) Find solutions for $\nu \neq 0$, $\lambda = 1$ and $\lambda = 2$.

c) Examine the parameter combination $\nu = 0$ and $\lambda = \frac{1}{2}$.

4. Develop a *Mathematica* function to combine the relative and center of mass coordinates for a representation of motion in real space for the two-ion problem of a Penning trap.

5. Reexamine the Green's function formalism and discuss the problem of a rectangular boundary with one side carrying a constant charge distribution. The three other sides are fixed to the ground potential.

6. Examine a collection of three particles in a Penning trap.

7. Discuss the motion of two particles in a Penning trap for $\nu \neq 0$ and λ arbitrary.

4.6 Packages and Programs

4.6.1 Point Charges

Package for the generation of fields, potentials and energy densities.

```
BeginPackage["PointCharge`"];

(* --- load additional standard packages --- *)

Needs["Graphics`PlotField`"];

Clear[Potential,Field,EnergyDensity,FieldPlot];

(* --- export functions --- *)

Potential::usage = "Potential[coordinates_List]
creates the potential of
an assembly of point charges. The cartesian
coordinates of the locations of
the charges are given in the form of
```

```
{{x,y,z,charge},{x,y,z,charge},...}.";

Field::usage = "Field[coordinates_List] calculates
the electric field for
an ensemble of point charges. The cartesian
coordinates are
lists in the form of {{x,y,z,charge},{...},...}.";

EnergyDensity::usage =
"EnergyDensity[coordinates_List] calculates the
density of the energy for an ensemble of point
charges. The cartesian
coordinates are lists in the form of
{{x,y,z,charge},{...},...}.";

FieldPlot::usage =
"FieldPlot[coordinates_List,typ_,options___] creates
a
ContourPlot for an ensemble of point charges. The
plot type (Potential,
Field, or Density) is specified as string in the
second input variable. The
third argument allows a change of the Options of
ContourPlot and
PlotGradientField.";

(* --- define the global variables x,y,z  --- *)

x::usage;
y::usage;
z::usage;

Begin["`Private`"];

(* --- determine the potential --- *)

Potential[coordinates_List]:=
      Block[{x,y,z},
      Fold[Plus,0,Map[(#[[4]]/Sqrt[(x-#[[1]])^2 +
                  (y-#[[2]])^2 +
                  (z-#[[3]])^2])&, coordinates]]];

(* --- calculate the field ---*)

Field[coordinates_List]:=
      Block[{field,x,y,z},
      field = -
Fold[Plus,0,Map[(#[[4]]*({x,y,z}-Take[#,3])/
```

```
                    (Sqrt[(x-#[[1]])^2 +
                     (y-#[[2]])^2 +
                     (z-#[[3]])^2
])^3)&,coordinates]];
      Simplify[field]
      ];

(* --- calculate the energy --- *)

EnergyDensity[coordinates_List]:=
      Block[{density,x,y,z,field},
      field = Field[coordinates];
      density = field.field/(8*Pi)
      ];

(* --- create plots  --- *)

FieldPlot[coordinates_List,typ_,options___]:=
    Block[
     {pot, ncharges, xmin, xmax, zmin, zmax, xcoord
= {}, zcoord = {},
      pl1, pl2},
       ncharges = Length[coordinates];
(* --- determine limits for the plot --- *)
     Do[
     AppendTo[xcoord,coordinates[[i,1]]];
     AppendTo[zcoord,coordinates[[i,3]]],
     {i,1,ncharges}];
      xmax = Max[xcoord]*1.5;
      zmax = Max[zcoord]*1.5;
      xmax = Max[{xmax,zmax}];
      zmax = xmax;
      xmin = -xmax;
      zmin = xmin;
      Clear[xcoord,zcoord];

(* --- fix the type of the plot ---*)
      If[typ == "Potential",pot =
Potential[coordinates] /. y -> 0,
      If[typ == "Field",pot =
-Potential[coordinates] /. y -> 0,
      If[typ == "EnergyDensity",pot =
EnergyDensity[coordinates] /. y -> 0,
      Print[" "];
      Print["  wrong key word! Choose "];
      Print["  Potential, Field or EnergyDensity  "];
      Print["  to create a plot "];
      Return[]
```

```
      ]]];

(* --- plot the pictures --- *)
       If[typ == "Field",
     pl1 =
PlotGradientField[pot,{x,xmin,xmax},{z,zmin,zmax},
           options,
           PlotPoints->20,
           ColorFunction->Hue
        ],
     pl1=
ContourPlot[pot,{x,xmin,xmax},{z,zmin,zmax},
          options,
          PlotPoints->50,
          ColorFunction->Hue,
          Contours->15]
      ]
       ];
End[];
EndPackage[];
```

4.6.2 Boundary Problem

The following package contains the main calculation steps for determining the expansion coefficients in the harmonic series representation of the potential.

```
BeginPackage["BoundaryProblem`",
  {"Calculus`Integration`"}];

Clear[Potential];

Potential::usage =
  "Potential[boundary_,R_,alpha_,n_] calculates the
potential in a circular segment. Input
    parameters are the potential on the
circle, the radius R of the circle and the
    angle of the segment of the circle.
The last argument n determines the
```

```
    number of expansion terms used to
represent the solution.";

Begin["`Private`"];

Potential[boundary_, R_, alpha_, n_] :=
  Block[{listed = {}, int, boundaryh},
   (*---replace the independent variable
      in the input by Phi---*)boundaryh =
    boundary /. f_[x2_. * x1_] → f[x2 * phi];
   (*---calculate the coefficients
      of the expansion d_n---*)
   int = Integrate[boundaryh * Sin[m * Pi * phi / alpha],
      {phi, 0, alpha}] * R^(m * Pi / alpha) * 2 / alpha;
   Do[AppendTo[listed, If[m == 0, 0, int]], {m, 0, n}];
   (*---calculate the
      potential by using the sum---*)
   pot = Sum[listed[[n1 + 1]] * r^(n1 * Pi / alpha) *
      Sin[n1 * Pi * phi / alpha], {n1, 0, n}];
   (*---transform the potential to
      cartesian coordinates---*)
   pot1 = pot /. {r → Sqrt[x^2 + y^2],
      phi → ArcTan[x, y]};
   (*---graphical representation of the
      potential by ContourPlot---*)
   ContourPlot[ pot1 Boole[x^2 + y^2 ≤ R^2 && y > 0 &&
       y ≤ Tan[alpha] x], {x, 0.0001, R}, {y, 0, R} ,
    PlotPoints -> 200, ColorFunction -> Hue,
    Contours → 15, PlotRange -> All, Epilog ->
     {Line[{{0, 0}, {R Cos[alpha], R Sin[alpha]}}]}]
  ];
End[];
EndPackage[];
```

4.6.3 Penning Trap

This package integrates the equations of motion for the Penning trap.

```
BeginPackage["Penning`"];

Clear[V,PenningI,PenningCMPlot];
```

```
PenningI::usage = "PenningI[r0_,z0_,e0_,n_,l_,te_]
determines the numerical
solution of the equation of motion for the relative
components. To integrate
the equations of motion, the initial conditions r0 =
r(t=0), z0 = z(t=0) and
the total energy e0 are needed as input parameters.
The momentum with respect
to the r direction is set to pr0=0. Parameters l and
n determine the
shape of the potential. The last argument te
specifies the end point of
the integration.";

PenningCMPlot::usage =
"PenningCMPlot[x0_,y0_,x0d_,y0d_,w_] gives a
graphical
representation of the center of mass motion for two
ions in the Penning trap.
The plot is created for a fixed cyclotron frequency
w in cartesian
coordinates (x,y,z). x0, y0, x0d, and y0d are the
initial conditions for
integration.";

Begin["`Private`"];

(* --- potential --- *)
V[x_, y_, l_, n_] := (x^2 + l^2*y^2)/2 + n^2/(2*x^2)
+
              1/(x^2 + y^2)^(1/2);

(*--- numerical integration of the relative motion
---*)

PenningI[r0_,z0_,e0_,n_,l_,te_]:=Block[{intk,pz0},
(* --- initial value of the momentum in z direction
--- *)
    pz0 = Sqrt[2*(e0-V[r0,z0,l,n])];
(* --- numerical solution of the initial value
problem --- *)
intk = NDSolve[{pr'[t] == n^2/r[t]^3 - r[t] +
              r[t]/(r[t]^2+z[t]^2)^(3/2),
        pz'[t] == -l^2*z[t] +
z[t]/(r[t]^2+z[t]^2)^(3/2),
        r'[t]  == pr[t],
        z'[t]  == pz[t],
```

```
(* --- initial values --- *)
         r[0] == r0, z[0] == z0, pr[0] == 0, pz[0]
== pz0},
     {r,z,pr,pz},{t,0,te}, MaxSteps->6000];
(* --- graphical representation --- *)

(* --- plot the potential --- *)
Show[
    Block[{$DisplayFunction=Identity},

{Plot3D[V[x,y,l,n]-0.4,{x,-2,2},{y,-2,2},Mesh->False,
         PlotPoints->25],
(* --- plot the tracks by ParametricPlot3D --- *)

ParametricPlot3D[Evaluate[{r[t],z[t],V[r[t],z[t],l,n]
} /. intk],
             {t,0,te},PlotPoints->1000,
                  AxesLabel->{"r","z","V"}]}
      ],
                  AxesLabel->{"r","z","V"},
                  Prolog->Thickness[0.001],
                  ViewPoint->{1.3,-2.4,2}
    ]

                  ];


(* --- center of mass motion in the Penning trap ---
*)

PenningCMPlot[x0_,y0_,x0d_,y0d_,w_]:= Block[{w0, a1,
b1},
(* --- fix parameters Omega_0 = 1.0 --- *)
      w0 = 1.0;
      a1 = 0.25;
      b1 = 0.0;
      If[w <= 2*w0,Print[" "];
        Print[" cyclotron frequency too small"];
        Print[" choose w > 2"],
(* --- determine the amplitudes from the initial
conditions --- *)
      gl1 = 2*ar + 2*br - x0 == 0;
      gl2 = -2*ai - 2*bi - y0 == 0;
      gl3 = 2*bi*w1 + 2*ai*w2 - x0d == 0;
      gl4 = 2*br*w1 + 2*ar*w2 - y0d == 0;
      result =
Flatten[N[Solve[{gl1,gl2,gl3,gl4},{ar,ai,br,bi}]]];
(* --- solutions for the center of mass motion --- *)
```

```
x = 2*br*Cos[w1*t] + 2*bi*Sin[w1*t] + 2*ar*Cos[w2*t]
+ 2*ai*Sin[w2*t];
y = 2*ar*Sin[w2*t] - 2*ai*Cos[w2*t] + 2*br*Sin[w1*t]
+ 2*bi*Cos[w1*t];
z = a1*Cos[Sqrt[2 w0]*t + b1];
(* --- define frequencies --- *)
w1 = wc/2 + Sqrt[(wc/2)^2 - w0];
w2 = wc/2 - Sqrt[(wc/2)^2 - w0];
(* --- substitute the results result into the
variables x, y, and z --- *)
x = Simplify[x /. result];
y = Simplify[y /. result];
x1 = x /. wc -> w;
x2 = y /. wc -> w;
x3 = z /. wc -> w;
(* --- plot the solution --- *)
ParametricPlot3D[{x1,x2,x3},{t,0,60},AxesLabel->{"x",
"y","z"},
         PlotPoints->1000,
         Prolog->Thickness[0.001]]
      ]];
End[];
EndPackage[];
```

5
Quantum Mechanics

5.1 Introduction

Quantum mechanics compared with mechanics is a very young theory. The theory emerged at 1900 when Max Planck (see Figure 5.1.1) examined the blackbody radiation in thermodynamics. The discovery by Planck was that the blackbody radiation can be described by a unified relation interpolating between the high-frequency limit proposed by Wien and the low-frequency limit favored by Rayleigh. The major assumption by Planck was that the energy in this relation is linear in frequency and discrete ($E = \hbar\,\omega$). Planck believed that this quantization applied only to the absorption and emission of energy by matter, not to electromagnetic waves themselves. However, it turned out to be much more general than he could have imagined.

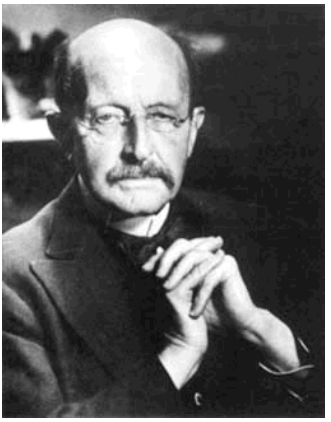


Figure 5.1.1. Max Planck: born April 23, 1858; died October 4, 1947.

Another anchorman in quantum mechanics was Erwin Schrödinger (see Figure 5.1.2) who invented wave mechanics in 1926. Reading the thesis of Louis de Broglie, he was inspired to write down a wave equation which established a second approach to mathematically describe quantum mechanics.

Figure 5.1.2. Erwin Schrödinger: born August 12, 1887; died January 4, 1961.

It was Werner Heisenberg (see Figure 5.1.3) who first gave a sound description of quantum mechanics with his matrix mechanics in 1925. Heisenberg was studying a set of quantized probability amplitudes when he used a matrix algebra. These amplitudes formed a noncommutative algebra. It was Max Born and Jordan in Göttingen who recognized this noncommutative algebra to be a matrix algebra. Another fundamental achievement by Heisenberg in 1927 was the uncertainty principle which governs all quantum mechanical systems.

Figure 5.1.3. Werner Heisenberg: born December 5, 1901; died February 1, 1976.

Today, quantum mechanics is a central theory in physics to describe micro and nano phenomena in atomic systems or semiconductors, for example. Quantum mechanics in its field-theoretic extensions is important in discussions of the unification of fundamental forces. The application of quantum mechanics ranges from nano systems up to large-scale systems such as black holes. Quantum mechanics is, in terms of its application, by no means a self-contained theory. The major open question in quantum theory is the unification with the theory of gravitation.

The current chapter introduces basic concepts of wave functions and demonstrates the application of the Schrödinger equation to different examples. In Section 5.2 the Schrödinger equation is introduced. Section 5.3 is concerned with the one-dimensional quantum dot model. Section 5.4 discusses the harmonic oscillator as a basic system to carry out quantum

mechanical calculations. The harmonic oscillator is extended to an anharmonic oscillator, which is important in the solution of nonlinear field equations. Section 5.6 discusses the motion of a particle in a central force field. The last section is concerned with the calculation of the second virial coefficient and its quantum mechanical correction.

5.2 The Schrödinger Equation

The development of quantum mechanics as a field of study required an equation that would adequately describe experimentally observed quantum mechanical properties, such as the spectroscopic properties of atoms and molecules. In 1926, Schrödinger wrote down the equation of motion for a complex field in close analogy to the eikonal equation of optics [5.1]. Today, it is known as the Schrödinger equation. The Schrödinger equation for a single particle reads

$$i\hbar\,\psi_t = -\frac{\hbar^2}{2m}\,\Delta\psi(\vec{x},\,t) + V(\vec{x})\,\psi(\vec{x},\,t), \tag{5.2.1}$$

where $\psi(\vec{x},\,t)$ denotes the wave function, $V(\vec{x})$ is an external potential representing the source of forces in the quantum system, $\hbar$ is Planck's constant, and m the mass of the particle under consideration.

The Schrödinger equation is a linear equation. It is well known that linear partial differential equations allow a superposition of their solutions to construct general solutions. Using this information with the two solutions ψ_1 and ψ_2 of the Schrödinger equation (5.2.1) allows us to construct the solution $\psi = c_1\,\psi_1 + c_2\,\psi_2$. We can identify Schrödinger's equation as a diffusion equation if we define an imaginary diffusion constant. To solve Schrödinger's equation, we can use, in principle, the same solution procedure as for the diffusion equation. For certain initial values and known boundary values, we find the evolution of the wave function ψ by Eq. (5.2.1).

The main problem at the outset of quantum mechanics was the interpretation of the wave function ψ. Although Schrödinger's linear equation of motion (5.2.1) is completely deterministic, its solution $\psi(\vec{x},\,t)$ is not a measurable quantity. In fact, the only observable quantities in

quantum mechanics are the probability $\psi^*\psi$ and any mean value based on the distribution function ψ denoted by $\langle\psi \mid \Theta \mid \psi\rangle$.

Another consequence of the linearity of the Schrödinger equation is the property of dispersion. It is well known that linear equations of motion have dispersive waves as solutions. Since Schrödinger's equation (5.2.1) contains an imaginary factor i, we can expect the solutions for a free particle to undergo oscillations in the time domain. Plane waves are the simplest solutions to ψ. A particular solution of Eq. (5.2.1) with $V(\vec{x}) = 0$ is given by

$$\psi_k(\vec{x}, t) = \frac{1}{\left(\sqrt{2\pi}\right)^3} e^{i\left(\vec{k}\,\vec{x} - \omega(k)\,t\right)}. \tag{5.2.2}$$

The superposition of this particular solution delivers the general solution by

$$\psi(\vec{x}, t) = \frac{1}{\left(\sqrt{2\pi}\right)^3} \int_{R^3} A(k)\, e^{i\left(\vec{k}\,\vec{x} - \omega(k)\,t\right)}\, d^3k. \tag{5.2.3}$$

For simplicity's sake, we limit our consideration to one spatial dimension. The solution (5.2.3) of the Schrödinger equation (5.2.1) is known as a wave packet. The spectral density $A(k)$ of the packet is completely determined by the initial condition $\psi(x, t = 0) = \psi_0(x)$. The representation (5.2.3) follows from the Fourier transform of the initial condition

$$A(k) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \psi_0(x)\, e^{-ikx}\, dx. \tag{5.2.4}$$

Inserting the spectral density into the general solution (5.2.3), we get the representation

$$\begin{aligned} \psi(x, t) &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \psi_0(x')\, e^{i\left(k(x-x') - \omega(k)\,t\right)}\, dk\, dx' \\ &= \int_{-\infty}^{\infty} \psi_0(x')\, G(x, x', t)\, dx', \end{aligned} \tag{5.2.5}$$

where the Green's function G is defined by

$$G(x, x', t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{i\left(k(x-x') - \omega(k)\,t\right)}\, dk. \tag{5.2.6}$$

The dispersion relation $\omega(k)$ of a dispersive wave is given by the defining equation of motion. For the Schrödinger equation with vanishing external potential $V(x) = 0$, the dispersion relation is $\omega(k) = \hbar k^2/(2m)$. Assuming a localized distribution $\psi_0(x) = \delta(x)$ for the initial condition of the wave function, we can write the related solution as follows:

$$\psi(x, t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{ik(x - \alpha k t)}\, dk. \tag{5.2.7}$$

This initial condition (assumed to derive the wave function ψ) cannot be normalized. Although this assertion contradicts the quantum mechanical interpretation, our only interest here is to show the dispersive behavior of the wave function. The constant $\alpha = \hbar/(2m)$ is purely numerical. The relation (5.2.7) represents a solution of the Schrödinger equation (5.2.1) for the case of a free particle located at $x = 0$ with $t = 0$. Since the Schrödinger equation describes dispersive phenomena, we can observe a broadening of the wave packet diminishing for $t \to \infty$. Its shape is studied in the following. Replacing k by $k = \kappa / \sqrt{\alpha t}$ in Eq. (5.2.7), we obtain

$$\psi(x, t) = \frac{1}{\sqrt{\alpha t}} \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{i\left(\kappa/\sqrt{\alpha t}\,(x-\kappa^2)\right)}\, d\kappa. \tag{5.2.8}$$

Computing the square in the exponent, we get

$$\psi(x, t) = \frac{1}{\sqrt{\alpha t}} \frac{1}{2\pi} e^{i x^2/(4\alpha t)} \int_{-\infty}^{\infty} e^{-i\left(x/\sqrt{4\alpha t} - \kappa\right)^2}\, d\kappa. \tag{5.2.9}$$

Substituting $\Gamma = x / \left(2\sqrt{\alpha t}\right) - \kappa$ gives us

$$\begin{aligned} \psi(x, t) &= \frac{1}{2\pi\sqrt{\alpha t}} e^{i x^2/(4\alpha t)} \int_{-\infty}^{\infty} e^{i\Gamma^2}\, d\Gamma \\ &= \frac{1}{2\sqrt{\alpha \pi t}} e^{i(x^2/(4\alpha t) + \pi/4)}. \end{aligned} \tag{5.2.10}$$

This representation of the wave function for a free particle can be used to determine the probability of locating the particle at a certain time. As discussed earlier, ψ is not a function directly observable by experiment. To locate a particle at a certain location at a certain time, we have to study the probability distribution $|\psi|^2$ of the particle. The probability distribution of solution (5.2.10) is given by the expression

$$|\psi(x, t)|^2 = \frac{1}{4\alpha\pi t}. \tag{5.2.11}$$

This result shows that the probability of finding a free particle as described by Schrödinger's equation vanishes as time goes on. The probability of finding a particle at any location decreases with time and vanishes as $t \to \infty$. The dispersion process of the particle can be represented using *Mathematica* in a sequence of pictures. To animate the dispersion process, we first define the wave function ψ of the free particle:

```
Psi[x_, t_, hbar_: 1, mass_: 1] :=
 Block[{alpha}, alpha = hbar / (2 mass);
  Exp[I (x^2 / (4 alpha t) + Pi / 4)] / (2 Sqrt[alpha t Pi])]
```

where mass m and $\hbar$ are set to unity. By an appropriate scaling of the coordinates, we can eliminate these constants in the equation of motion. The probability distribution $|\psi|^2$ in relation (5.2.11) is only a function of time and does not show any spatial dependence. However, if we examine the wave function itself, we observe the spatial dispersion of the wave.

In Figure 5.2.4 a time sequence of the real part of the wave function is plotted. The pictures are created by

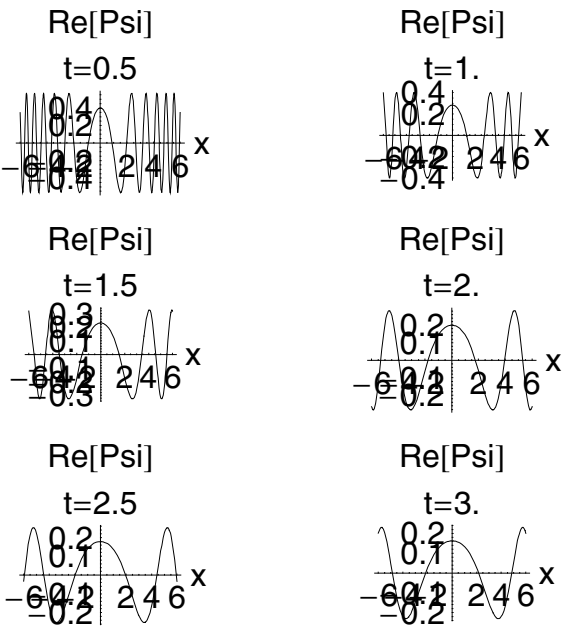


Figure 5.2.4. Time evolution of a wave packet for the Schrödinger equation. Initial conditions are $\psi_0(x) = \delta(x)$.

The plots show that the amplitude of the wave function decreases from about 0.5 to about 0.1 in a time range of 0.5 to 3.0. The dispersion of the wave packet is observable in the wave function. The wave function exhibits a reduced amplitude and a broadening of the initial packet.

The Schrödinger equation (5.2.1) not only describes time-dependent properties of quantum mechanical systems but also stationary properties of these systems. Contrary to our observations about free particles, we now find that Schrödinger's equation describes stable particles. One central question for such a system is how to uncover its intrinsic characteristics such as the spectral properties. In the following, we examine one of the fundamental models of quantum mechanics—the harmonic oscillator.

Before discussing the spectral properties of the harmonic oscillator, we first summarize the solution steps for the time dependent Schrödinger equation by a short graphical representation given in Figure 5.2.5.

1. Starting point of the solution procedure is the partial differential equation (PDE) (5.2.1) and the initial solution of the wave function $\psi(x, 0)$.

2. The use of the Fourier transform allows us to derive the spectral density $A(k)$ from the initial conditions.

3. A complete representation in Fourier space is attained when considering the time evolution, which is given by the dispersion relation $\omega(k)$.

4. The inversion of the representation in Fourier space delivers the solution of the Schrödinger equation.

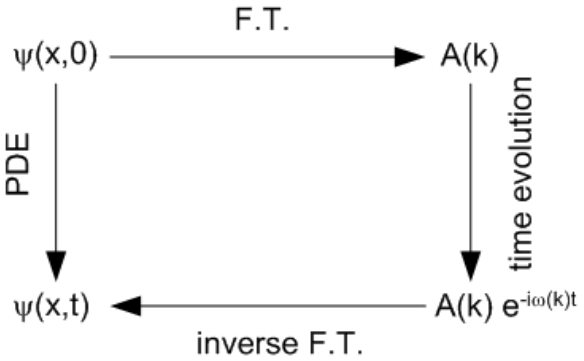


Figure 5.2.5. Solution steps for a linear PDE by using the Fourier transform.

A similar solution procedure for nonlinear PDEs is discussed in Chapter 3 on nonlinear dynamics.

5.3 One-Dimensional Potential

In quantum mechanics, the measurement of a physical quantity A can result only in one of the eigenvalues of the corresponding operator $\hat{A}$. The eigenvalues of $\hat{A}$ forming the spectrum of the operator might be discrete, continuous, or both. The eigenfunctions of $\hat{A}$ form a complete basis that can be used to expand an arbitrary wave function. The expansion coefficients can be used to determine the probability of finding the system in an eigenstate of the operator $\hat{A}$ with eigenvalue a. Central to quantum mechanics is the determination of these eigenvalues and their related eigenfunctions.

One of the fundamental quantities of a quantum dynamical system is its energy. The operator corresponding to energy is the Hamiltonian operator of the system. The Hamiltonian for a particle with mass m located in a potential V is represented by $\hat{H} = -\hbar^2/(2m)\Delta + V(x)$. The determination of eigenvalues and eigenfunctions is demonstrated with a one-dimensional model, the potential well. The potential well of depth $V = -V_0$ discussed in the following extends between $-a \le x \le a$ where a is the maximum

extension. Beyond the maximum extension, the potential vanishes. A graphical representation of the potential is given in Figure 5.3.6.

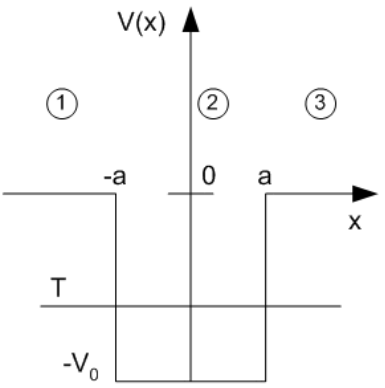


Figure 5.3.6. The potential well of depth V.

We study the case for which the kinetic energy of the particle is smaller than the minimal potential value V_0 (i.e., $T < V_0$). The total energy E of the system is $E = T - V_0 < 0$. The particle has a negative total energy in the domains 1 and 3 depicted in Figure 5.3.6. In classical mechanics, the particle cannot be found in these regions. Contrary to classical mechanics, however, quantum mechanics allows the existence of particles in regions where they are classically forbidden. The domains 1 and 3 are governed by the eigenvalue equations $\hat{H}\,\psi = E\,\psi$, which are given in a differential representation by

$$\psi'' - \kappa^2\,\psi = 0, \tag{5.3.12}$$

where $\kappa^2 = -2\,m\,E/\hbar^2 > 0$ is a positive constant containing the total energy. Primes denote differentiation with respect to the spatial coordinate. The solution of Eq. (5.3.12) represents the domains 1 and 3 by

$$\psi_1 = A_1\,e^{\kappa x} + B_1\,e^{-\kappa x} \qquad \text{for} \qquad -\infty < x \le -a, \tag{5.3.13}$$

$$\psi_3 = A_3\,e^{\kappa x} + B_3\,e^{-\kappa x} \qquad \text{for} \qquad a \le x < \infty. \tag{5.3.14}$$

The related *Mathematica* result reads

```
s13 = DSolve[∂x,x ψ[x] - κ^2 ψ[x] == 0, ψ, x] // Flatten
```

$\{\psi \to \text{Function}[\{x\}, e^{x\kappa}\, c_1 + e^{-x\kappa}\, c_2]\}$

In domain 2 the eigenvalue equation takes the form

$$\psi'' + k^2\,\psi = 0, \tag{5.3.15}$$

where $k^2 = 2\,m(V_0 + E)/\hbar^2 > 0$. The complete solution of (5.3.15) is given by

$$\psi_2 = A_2 \cos k\,x + B_2 \sin k\,x \qquad \text{for} \quad -a \le x \le a. \tag{5.3.16}$$

The computer algebra result is

```
s2 = DSolve[∂x,x ψ[x] + k^2 ψ[x] == 0, ψ, x] // Flatten
```

$\{\psi \to \text{Function}[\{x\}, c_1 \cos(k\,x) + c_2 \sin(k\,x)]\}$

From the normalization condition, it follows that the eigenfunctions given by relations (5.3.13) and (5.3.14) require that the coefficients B_1 and A_3 vanish (i.e., $B_1 = A_3 = 0$). The remaining parameters A_1, B_2, A_2 and B_3 are determined by applying the continuity condition of the wave function and its first derivative at the end points of the potential well ($x = -a$ and $x = a$).

The normalization condition requires

```
ps1 = ψ[x] /. s13 /. {C[1] → A1, C[2] → B1} /. B1 -> 0
```

$\text{A1}\, e^{x\kappa}$

and

```
ps3 = ψ[x] /. s13 /. {C[1] → A3, C[2] → B3} /. A3 -> 0
```

$\text{B3}\, e^{-x\kappa}$

The conditions on the domain boundaries read

$$\psi_1 = \psi_2 \quad \text{and} \quad \psi_1' = \psi_2' \quad \text{for} \quad x = -a, \tag{5.3.17}$$
$$\psi_2 = \psi_3 \quad \text{and} \quad \psi_2' = \psi_3' \quad \text{for} \quad x = a \tag{5.3.18}$$

which can be given as

```
eq1 = ps1 == (ψ[x] /. s2 /. {C[1] → A2, C[2] → B2}) /. x → -a
```

$$\text{A1}\, e^{-a\kappa} == \text{A2}\cos(a\,k) - \text{B2}\sin(a\,k)$$

```
eq2 =
 ∂x ps1 == (∂x ψ[x] /. s2 /. {C[1] → A2, C[2] → B2}) /. x → -a
```

$$\text{A1}\, e^{-a\kappa}\,\kappa == \text{B2}\,k\cos(a\,k) + \text{A2}\,k\sin(a\,k)$$

```
eq3 = ps3 == (ψ[x] /. s2 /. {C[1] → A2, C[2] → B2}) /. x → a
```

$$\text{B3}\, e^{-a\kappa} == \text{A2}\cos(a\,k) + \text{B2}\sin(a\,k)$$

and

```
eq4 =
 ∂x ps3 == (∂x ψ[x] /. s2 /. {C[1] → A2, C[2] → B2}) /. x → a
```

$$-\text{B3}\, e^{-a\kappa}\,\kappa == \text{B2}\,k\cos(a\,k) - \text{A2}\,k\sin(a\,k)$$

The four equations form a homogeneous system of equations for the unknowns A_1, B_3, A_2, and B_2. In a matrix representation, we get

$$\begin{pmatrix} e^{-\kappa a} & -\cos(k\,a) & \sin(k\,a) & 0 \\ \kappa\, e^{-\kappa a} & -k\sin(k\,a) & -k\cos(k\,a) & 0 \\ 0 & -\cos(k\,a) & -\sin(k\,a) & e^{-\kappa a} \\ 0 & k\sin(k\,a) & -k\cos(k\,a) & -\kappa\, e^{-\kappa a} \end{pmatrix} \begin{pmatrix} A_1 \\ A_2 \\ B_1 \\ B_2 \end{pmatrix} = 0. \tag{5.3.19}$$

A nontrivial solution of Eq. (5.3.19) exists if the determinant of the matrix vanishes. This condition delivers the relation

$$\kappa^2 - k^2 + 2\,\kappa\,k\cot(2\,k\,a) = 0 \tag{5.3.20}$$

```
det1 =
 Map[Coefficient[({eq1, eq2, eq3, eq4} /. Equal[a_,
           b_] :> a - b), #] &,
     {A1, A2, B2, B3}] // Transpose // Det // Simplify
```

$$e^{-2\,a\,\kappa}\,(2\,k\,\kappa\cos(2\,a\,k) + (\kappa^2 - k^2)\sin(2\,a\,k))$$

with solutions

$$\kappa = \;\; k\tan(k\,a), \tag{5.3.21}$$
$$\kappa = -k\cot(k\,a). \tag{5.3.22}$$

```
spectral = MapAll[PowerExpand[#] &, Simplify[
    Flatten[Solve[det1 == 0, κ]]]] // FullSimplify
```

$$\{\kappa \to -k\cot(a\,k),\ \kappa \to k\tan(a\,k)\}$$

If we consider the first of these relations (5.3.21), we find that $B_2 = 0$, $B_3 = A_1$, and $A_2 \cos k\,a = A_1\,e^{-\kappa\,a}$. The second relation, (5.3.22), results in the conditions $A_2 = 0$, $B_3 = -A_1$, and $B_2 \sin \mathrm{ka} = -A_1\,e^{-\kappa\,a}$.

```
sol1 = Solve[{eq1, eq2, eq3, eq4} /. spectral[[1]],
    {A1, B2, A2, B3}] // Simplify // Flatten
```

Solve::svars : Equations may not give solutions for all "solve" variables. More…

$$\{\mathrm{A1} \to -\mathrm{B3},\ \mathrm{B2} \to \mathrm{B3}\,e^{a\,k\cot(a\,k)}\csc(a\,k),\ \mathrm{A2} \to 0\}$$

```
sol2 = Solve[{eq1, eq2, eq3, eq4} /. spectral[[2]],
    {A1, A2, B2, B3}] // Simplify // Flatten
```

Solve::svars : Equations may not give solutions for all "solve" variables. More…

$$\{A1 \to B3,\ A2 \to B3\, e^{-a\,k\,\tan(a\,k)}\,\sec(a\,k),\ B2 \to 0\}$$

We can thus distinguish between two systems of eigenfunctions: a symmetric one and an antisymmetric one. The symmetry of the eigenfunctions is obvious if we exchange the coordinates by $x \to -x$. The symmetrical case is represented by

$$\kappa = k\,\tan(k\,a), \tag{5.3.23}$$
$$\psi_1 = A_1\, e^{\kappa\,x}, \tag{5.3.24}$$
$$\psi_2 = A_1\, e^{-\kappa\,a}\ \frac{\cos(k\,x)}{\cos(k\,a)}, \tag{5.3.25}$$
$$\psi_3 = A_1\, e^{-\kappa\,x} \tag{5.3.26}$$

```
ψ1s = ps1 /. sol2 /. spectral[[2]]
```

$$B3\, e^{k\,x\,\tan(a\,k)}$$

```
ψ2s = ψ[x] /. s2 /. {C[1] → A2, C[2] → B2} /. sol2 /.
  spectral[[2]]
```

$$B3\, e^{-a\,k\,\tan(a\,k)}\,\cos(k\,x)\,\sec(a\,k)$$

```
ψ3s = ps3 /. sol2 /. spectral[[2]]
```

$$B3\, e^{-k\,x\,\tan(a\,k)}$$

The antisymmetric case follows from the relations

$$\kappa = -k\,\cot(k\,a), \tag{5.3.27}$$
$$\psi_1 = -A_1\, e^{\kappa\,x}, \tag{5.3.28}$$
$$\psi_2 = A_1\, e^{-\kappa\,a}\ \frac{\sin(k\,x)}{\sin(k\,a)}, \tag{5.3.29}$$
$$\psi_3 = A_1\, e^{-\kappa\,x} \tag{5.3.30}$$

```
ψ1a = ps1 /. sol1 /. spectral[[1]]
```

$$-\text{B3}\, e^{-k\, x \cot(a\, k)}$$

```
ψ2a = ψ[x] /. s2 /. {C[1] → A2, C[2] → B2} /. sol1 /.
  spectral[[1]]
```

$$\text{B3}\, e^{a\, k \cot(a\, k)} \csc(a\, k) \sin(k\, x)$$

```
ψ3a = ps3 /. sol1 /. spectral[[1]]
```

$$\text{B3}\, e^{k\, x \cot(a\, k)}$$

From the normalization condition

$$\int_{-\infty}^{\infty} \psi^2\, dx = \int_{-\infty}^{-a} \psi_1^2\, dx + \int_{-a}^{a} \psi_2^2\, dx + \int_{a}^{\infty} \psi_3^2\, dx, \tag{5.3.31}$$

we get a relation for the undetermined amplitude A_1

$$\frac{1}{A_1^2} = a\, e^{-2\,\kappa\, a} \left(1 + \frac{1}{\kappa\, a} + \frac{\kappa}{k^2\, a} + \frac{\kappa^2}{k^2}\right). \tag{5.3.32}$$

Relation (5.3.32) is satisfied for both the symmetric and antisymmetric eigenfunctions. To calculate the eigenvalues, note that $\kappa^2 + k^2 = 2\, m\, V_0 / \hbar^2 > 0$ is independent of the total energy E. If we introduce the parameter

$$C^2 = a^2\, 2\, m\, \frac{V_0}{\hbar^2} = (\kappa^2 + k^2)\, a^2, \tag{5.3.33}$$

we can eliminate κ from the eigenvalue equations. The equations determining the eigenvalues are now

$$\frac{\sqrt{C^2 - (k\, a)^2}}{k\, a} = \tan(k\, a), \tag{5.3.34}$$

$$-\frac{k\, a}{\sqrt{C^2 - (k\, a)^2}} = \tan(k\, a). \tag{5.3.35}$$

Using relation (5.3.34) or (5.3.35), we can calculate $k\, a$ and $E = \hbar^2\, k^2 - 2\, m\, V_0$.

The problem with the potential well is not the derivation of its solution but the calculation of the eigenvalues determined by Eqs. (5.3.34) and (5.3.35). In the package **QuantumWell`** (see Section 5.8.1), we solve the problem numerically for varying well depths V_0 and well widths a. Because the two determining equations of the eigenvalues are transcendent equations, we have to switch to numeric calculations. The left-hand and right-hand sides of Eqs. (5.3.34) and (5.3.35) are graphically represented in Figure 5.3.7 for $V_0 = 12$ and $a = 1$.

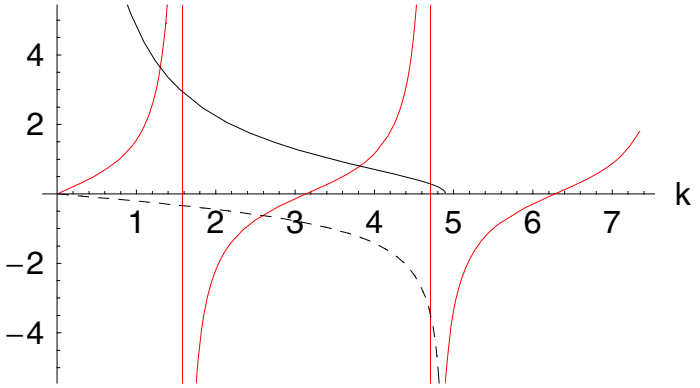


Figure 5.3.7. Graphical representation of the eigenvalue equation for $V_0 = 12$ and $a = 1$. The solid curves represent the symmetrical case and the dashed curves represent the antisymmetric case. The right-hand side of the eigenvalue equation reads $\tan k\,a$.

Figure 5.3.7 is created by means of the function **Spectrum[12,1]** defined in the package **QuantumWell`**. Also defined in the package **QuantumWell`** are the eigenfunctions **PsiSym[]** and **PsiASym[]**. The function **Spectrum[]** provides us with a graphical representation of the eigenfunctions and prints out the related eigenvalues in a list. Some examples of these eigenfunctions are given in Figures 5.3.8 and 5.3.9. Function **Spectrum[]** creates a sequence of eigenfunction pictures starting with the symmetric ones followed by the antisymmetric ones. Figures 5.3.8 and 5.3.9 contain the superposition of these sequences into one picture.

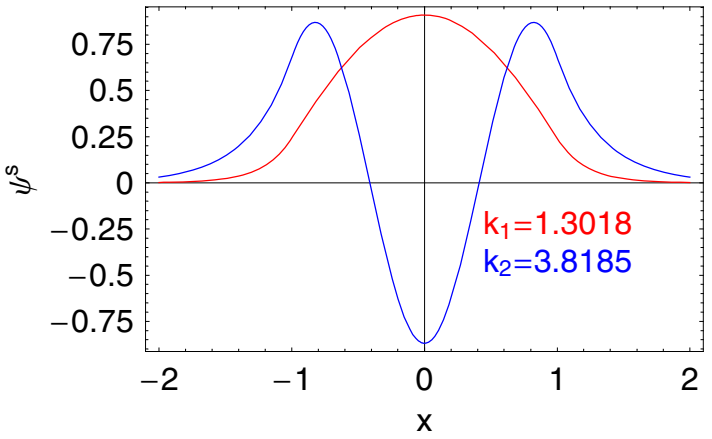


Figure 5.3.8. The symmetric eigenfunctions for a potential well with depth $V_0 = 12$ and width $a = 1$. For the given potential depth, there are a total of four eigenvalues, two of which are shown in this figure and the other two are shown in the next figure. The solid eigenfunction with a broad single maximum and no nodes is related to the lowest eigenvalue k =1.30183 of the symmetric case. The second symmetric eigenvalue is k =3.81858. The corresponding eigenfunction is dashed.

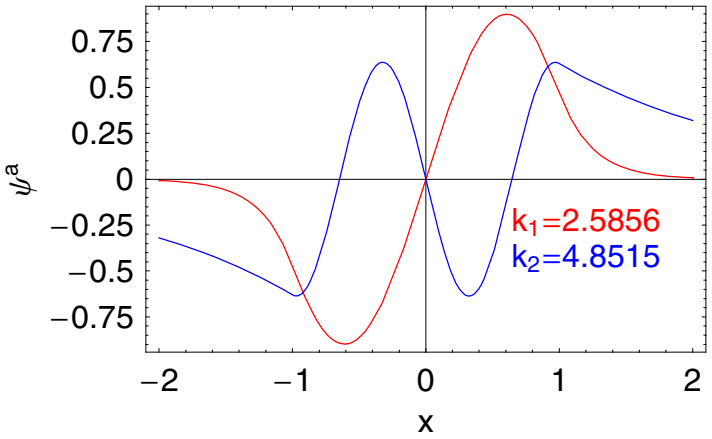
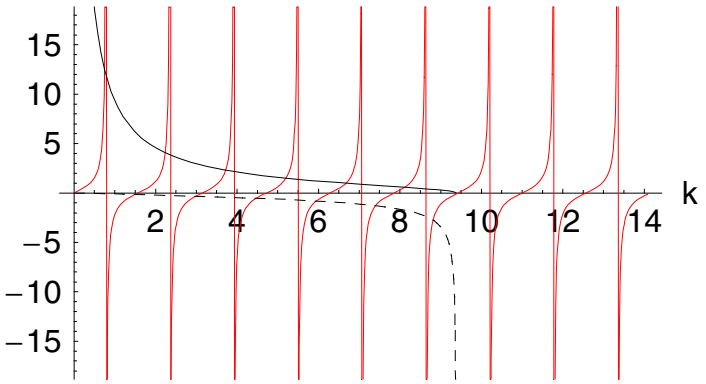
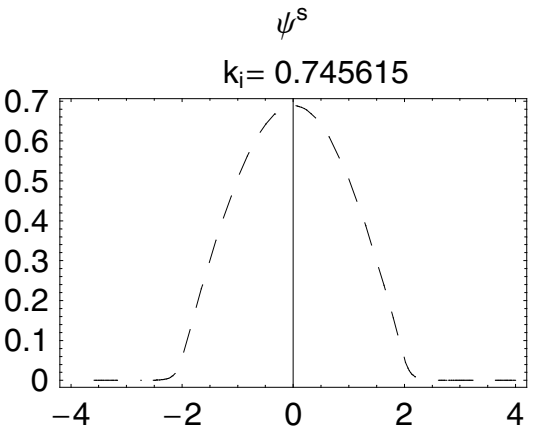


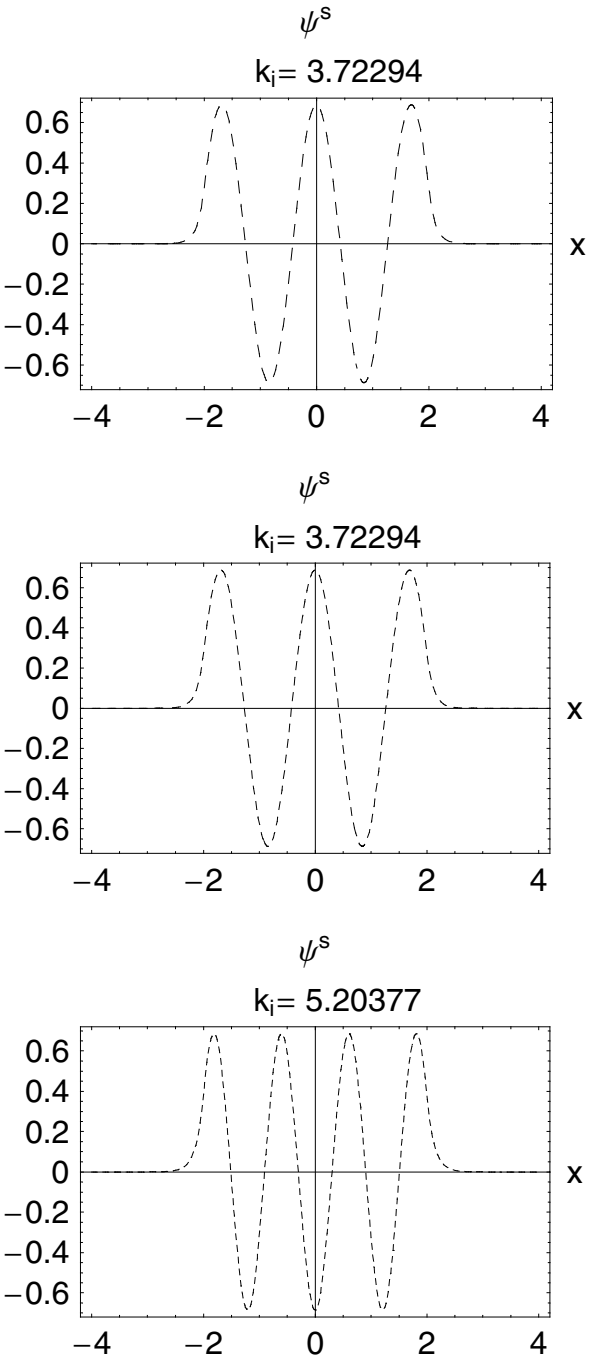
Figure 5.3.9. The antisymmetric eigenfunction for the potential with $V_0 = 12$ and $a = 1$. The two antisymmetric eigenfunctions are correlated with the eigenvalues k =2.5856 and k =4.85759. The first eigenfunction is represented by a solid curve and the second is dashed.

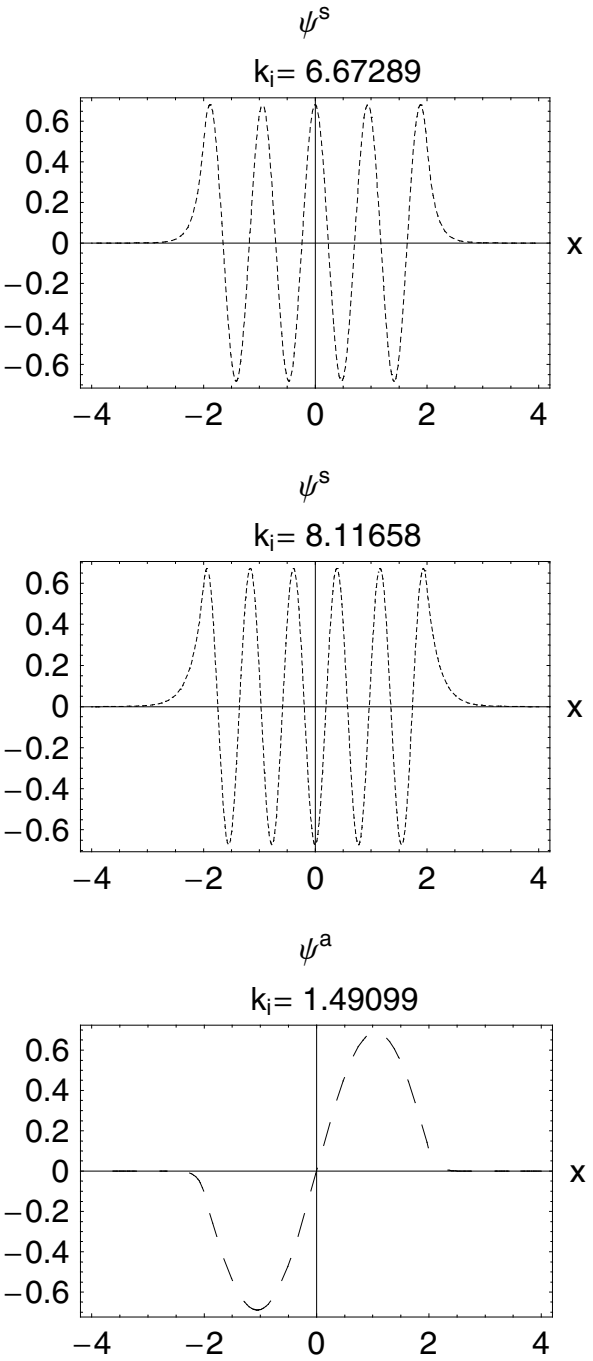
The sequence of eigenfunctions and eignvalues for different potential depths V_0 are generated with the function **Spectrum[].** For a potential depth of $V_0 = 44$ with a potential with $a = 2$ we find

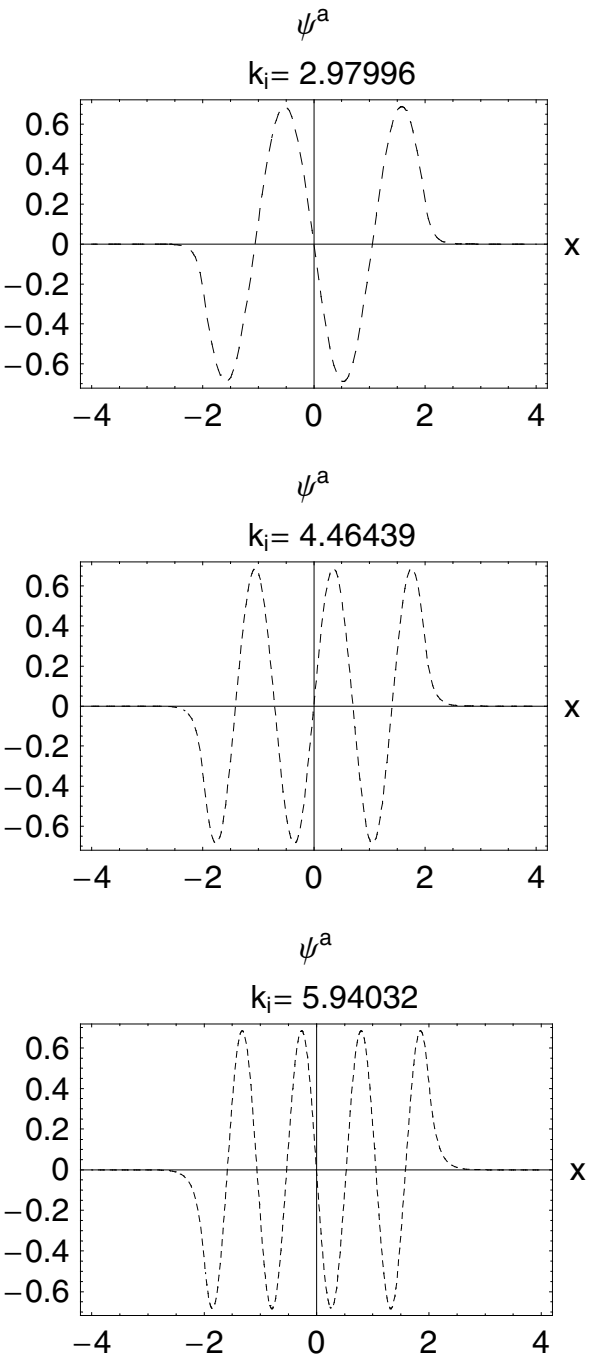
```
Spectrum[44, 2]
```

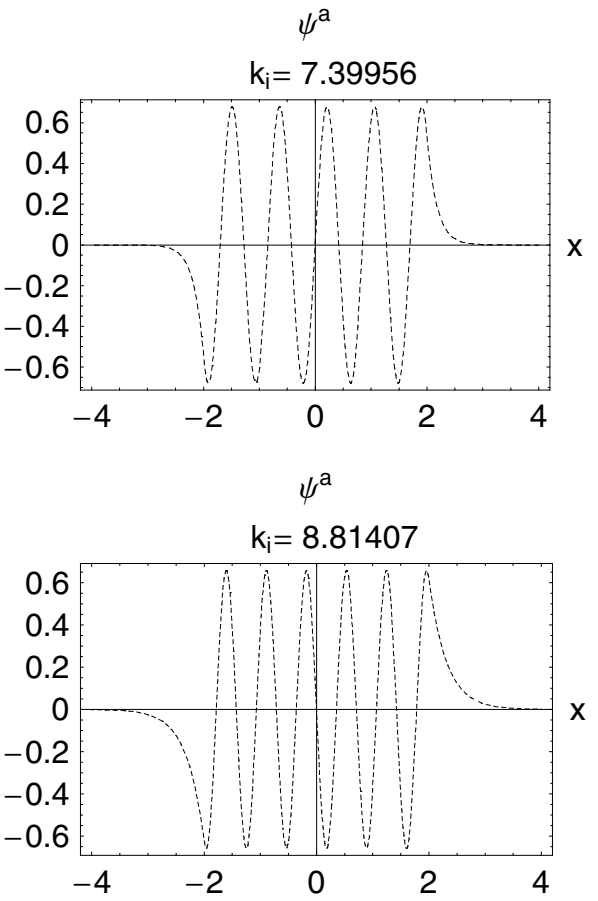





ψ^s
k_i= 3.72294
0.6
0.4
0.2
0
−0.2
−0.4
−0.6
−4
−2
0
2
4
x
ψ^s
k_i= 3.72294
0.6
0.4
0.2
0
−0.2
−0.4
−0.6
−4
−2
0
2
4
x
ψ^s
k_i= 5.20377
0.6
0.4
0.2
0
−0.2
−0.4
−0.6
−4
−2
0
2
4
x


ψ^s
k_i= 6.67289
0.6
0.4
0.2
0
−0.2
−0.4
−0.6
−4
−2
0
2
4
x
ψ^s
k_i= 8.11658
0.6
0.4
0.2
0
−0.2
−0.4
−0.6
−4
−2
0
2
4
x
ψ^a
k_i= 1.49099
0.6
0.4
0.2
0
−0.2
−0.4
−0.6
−4
−2
0
2
4
x


ψ^a
k_i= 2.97996
0.6
0.4
0.2
0
−0.2
−0.4
−0.6
−4
−2
0
2
4
x
ψ^a
k_i= 4.46439
0.6
0.4
0.2
0
−0.2
−0.4
−0.6
−4
−2
0
2
4
x
ψ^a
k_i= 5.94032
0.6
0.4
0.2
0
−0.2
−0.4
−0.6
−4
−2
0
2
4
x




```
----  eigenvalues ----


sym eigenvalue k1 = 0.745615 asym eigenvalue k1 = 1.49099

sym eigenvalue k2 = 3.72294 asym eigenvalue k2 = 2.97996

sym eigenvalue k3 = 3.72294 asym eigenvalue k3 = 4.46439

sym eigenvalue k4 = 5.20377 asym eigenvalue k4 = 5.94032

sym eigenvalue k5 = 6.67289 asym eigenvalue k5 = 7.39956

sym eigenvalue k6 = 8.11658 asym eigenvalue k6 = 8.81407
```

The result is a system allowing 12 eigenvalues corresponding to 6 symmetric and 6 antisymmetric eigenfunctions.

5.4 The Harmonic Oscillator

The potential energy for a stable system exhibits a local minimum. One of the standard methods of physics is to expand the potential energy around the point of a local minimum in a Taylor series,

$$V = V_0 + \frac{1}{2}\left(\frac{\partial^2 V}{\partial x^2}\right)\Big|_{x=0} x^2 + ..., \tag{5.4.36}$$

where x denotes the displacement from the equilibrium point. The potential satisfies $\partial V / \partial x = 0$ at the stable equilibrium point. If the particle of mass m only undergoes small oscillations around the equilibrium point, the first two terms of relation (5.4.36) are sufficient to describe the potential energy. Choosing the origin of the energy to be identical with $V(0)$ of the expansion, we can express the Hamiltonian of the harmonic oscillator

$$H_{\mathrm{cl}} = \frac{p^2}{2m} + \frac{k}{2} x^2, \tag{5.4.37}$$

where $k = \partial^2 V / \partial x^2 \Big|_{x=0}$ is the spring constant of the oscillator. We already know that the classical solution for the harmonic oscillator is given by a periodic function

$$x(t) = A\cos(\omega t + \beta) \text{ where } \omega = \sqrt{\frac{k}{m}} \tag{5.4.38}$$

and the system undergoes harmonic oscillations around the equilibrium point. The time average of the total energy follows from relations (5.4.37) and (5.4.38)

```
⟨E⟩_T =
  ω/(2 π) (∫_0^(2 π/ω) ((m (∂_t x[t])^2)/2 + k/2 x[t]^2) /. x -> Function[t,
      A Cos[ω t + β]]) dt // Simplify) /. k -> ω^2 m
```

$$\frac{1}{2} A^2 m \omega^2$$

$$\langle E \rangle_T = \frac{1}{2} m A^2 \omega^2 = m \omega^2 \overline{x}^2, \tag{5.4.39}$$

where T denotes the period of the oscillation; that is, the time-averaged energy depends quadratically on the amplitude A of the oscillations.

In this section, our aim is to examine the quantum mechanical properties of the harmonic oscillator and compare them with the classical situation. The transition from classical to quantum mechanics is formally achieved by replacing the classical coordinates with quantum mechanical operators: $x \to \hat{x}$ and $p \to \hat{p} = \hbar / i \, \partial_x$. Using the transformations in the Hamiltonian yields the timeless Schrödinger equation in the form of an eigenvalue problem given by

$$\left(\frac{d^2}{d x^2} - \frac{\omega^2 m^2}{\hbar^2} x^2 + \frac{2 m E}{\hbar^2} \right) \psi(x) = 0, \tag{5.4.40}$$

where ψ denotes the set of eigenfunctions of the Hamiltonian. By an appropriate scaling of the spatial coordinate $\xi = \sqrt{m \omega / \hbar}\, x$ and of the eigenvalue $\varepsilon = 2 E / (\hbar \omega)$, we get the eigenvalue problem in a standard form

$$\left(\frac{d^2}{d \xi^2} - \xi^2 + \varepsilon \right) \psi(\xi) = 0. \tag{5.4.41}$$

```
eigenValueEquation = ∂_{ξ,ξ} ψ[ξ] - ξ^2 ψ[ξ] + ε ψ[ξ] == 0
```

$$-\psi(\xi)\, \xi^2 + \varepsilon\, \psi(\xi) + \psi''(\xi) = 0$$

The question here is what type of function $\psi(\xi)$ satisfies Eq. (5.4.41). As a solution, we try the expression

$$\psi(\xi) = v(\xi)\, e^{-\xi^2/2}. \tag{5.4.42}$$

```
ansatz = ψ -> Function[ξ, v[ξ] e^(-ξ^2/2)]
```

$$\psi \to \text{Function}\left[\xi,\ v(\xi)\, e^{-\frac{\xi^2}{2}}\right]$$

From Eq. (5.4.41), it follows that the amplitude v has to satisfy the ODE

$$v'' - 2\,\xi\, v' + (\varepsilon - 1)\, v(\xi) = 0, \tag{5.4.43}$$

```
transformedEVeq =
 eigenValueEquation /. ansatz // Simplify
```

$$e^{-\frac{\xi^2}{2}}\,((\varepsilon - 1)\, v(\xi) - 2\,\xi\, v'(\xi) + v''(\xi)) == 0$$

where primes denote differentiation with respect to ξ. To be physically acceptable, the wave function $\psi(\xi)$ must be continuous and finite. The amplitude $v(\xi)$ defined by Eq. (5.4.43) is a finite function if v is a polynomial of finite order.

```
solution = DSolve[transformedEVeq, v, ξ] // Flatten
```

$$\left\{v \to \text{Function}\left[\{\xi\},\ c_1\, H_{\frac{\varepsilon-1}{2}}(\xi) + c_2\, {}_1F_1\left(\frac{1-\varepsilon}{4};\ \frac{1}{2};\ \xi^2\right)\right]\right\}$$

This type of solutions exists if

$$\varepsilon = 2\,n + 1, \text{ where } n = 0,\ 1,\ 2,\ \ldots. \tag{5.4.44}$$

For each value n there exists a polynomial of order n which satisfies Eq. (5.4.43). These polynomials are known as Hermite polynomials, defined by

$$H_n(\xi) = (-1)^n\, e^{\xi^2}\, \frac{d^n}{d\,x^n}\, e^{-\xi^2}. \tag{5.4.45}$$

In *Mathematica*, the Hermite polynomials are identified by the function **HermiteH[]**. The solutions of the eigenvalue problem become with the eigenvalues

```
eigenValues = ε → 2 n + 1
```

$$\varepsilon \to 2\,n + 1$$

a two-component solution determined by c_1 and c_2, the integration constants

```
ve = v[ξ] /. solution /. eigenValues
```

$$c_1\, H_n(\xi) + c_2\, {}_1F_1\Big(-\frac{n}{2};\, \frac{1}{2};\, \xi^2\Big)$$

it is known that the hypergeometric function ${}_1F_1$ is divergent for $\xi \to \pm\infty$. Thus, we can chose $c_2 = 0$. The eigenfunctions thus are determined by

```
ve = v -> Function[ξ, $v] /. $v -> (ve /. C[2] -> 0)
```

$$v \to \text{Function}[\xi,\, c_1\, H_n(\xi)]$$

The eigenfunctions thus can be written

```
ps = ψ[x] /. ansatz /. ve
```

$$e^{-\frac{x^2}{2}}\, c_1\, H_n(x)$$

where c_1 is a constant determined by the normalization. The wave function ψ of the harmonic oscillator is represented in scaled coordinates by

$$\psi_n(\xi) = \frac{1}{\sqrt{n!\, 2^n\, \sqrt{\pi}}}\, H_n(\xi)\, e^{-\xi^2/2}. \tag{5.4.46}$$

The corresponding eigenvalues of the harmonic oscillator are

$$E_n = \hbar\,\omega\,\Big(n + \frac{1}{2}\Big). \tag{5.4.47}$$

Each eigenvalue has its own eigenfunction which is either even or odd with respect to coordinate reflections in ξ. Note that the eigenvalues and

eigenfunctions have a one-to-one correspondence (i.e. the spectrum is non-degenerate). The first four even and odd eigenfunctions of the harmonic oscillator are depicted in figures 5.4.10 and 5.4.11.

The probability distribution $|\psi|^2$ of finding the harmonic oscillator in a certain state n in the range $\xi \pm d\,\xi$ is given by

$$|\psi|^2\, d\,\xi = \frac{1}{n!\, 2^n \sqrt{\pi}}\, H_n^2(\xi)\, e^{-\xi^2}\, d\,\xi = w_{\mathrm{qm}}(\xi)\, d\,\xi. \tag{5.4.48}$$

The classical probability of finding a particle in the range $x \pm d\,x$ is determined by the period T of the oscillator.

$$w_{\mathrm{cl}}(x) = \frac{d\,t}{T} = \frac{\omega}{2\,\pi}\, d\, \frac{x}{|v|}, \tag{5.4.49}$$

where $x(t)$ is represented by the classical solution (5.4.38). The corresponding velocity v follows from the time derivative of x:

$$v = -A\,\omega \sqrt{1 - \left(\frac{x}{A}\right)^2}. \tag{5.4.50}$$

In scaled variables ξ we find for the classical probability the relation

$$w_{\mathrm{cl}}(\xi) = \frac{1}{2\,\pi \sqrt{2\,n+1}}\, \frac{1}{\sqrt{1-\xi^2/(2\,n+1)}}. \tag{5.4.51}$$

Specifying either the energy or the eigenvalue of the harmonic oscillator enables us to compare the classical probability with the quantum mechanical result. A graphical representation of these two quantities is given in Figures 5.4.12 and 5.4.13. Figure 5.4.12 shows the ground state and Figure 5.4.13 shows the eigenvalue with $n = 5$. It can be clearly seen that the quantum mechanical behavior of the probability density is different from its classical behavior. In the classical case, the particle spends most of its time near the two turning points, where the density $|\psi|^2$ is large. Quantum mechanically, there is a high probability that the particle is located near the center of the potential (ground state). In an excited state, we observe regions where the particle cannot be found (see Figure 5.4.13). This is due to the fact that the quantum mechanical probability density oscillates for $n > 0$, which, in turn, is a consequence of the oscillations of the wave function.

At the classical turning points, a completely different behavior of the quantum particle is apparent. Where the classical particle cannot be found

in quantum mechanics, there is a finite probability for locating a particle outside the potential well. This tunneling of the particle into the potential barrier is unusual and cannot be explained by classical mechanics.

The eigenfunctions and the harmonic potential $V(\xi)$ are superimposed on each other in Figures 5.4.10 and 5.4.11. The related classical and quantum mechanical probabilities are shown in Figures 5.4.12 and 5.4.13. The functions to create these figures for certain eigenvalues are contained in the package **HarmonicOscillator`** (see Section 5.8.2).

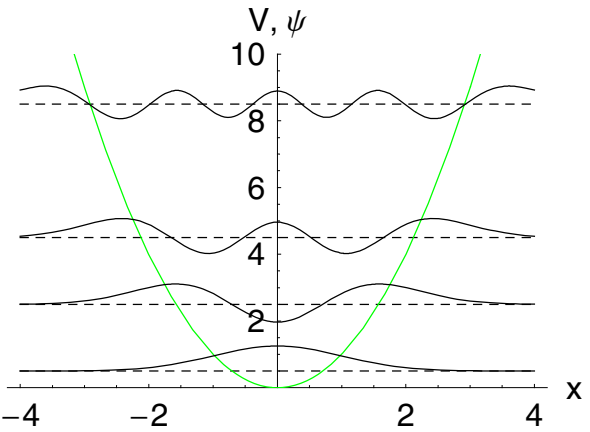


Figure 5.4.10. Symmetric eigenfunctions of the harmonic oscillator $V(x) = x^2$ for eigenvalues $n = 0, 2, 4, 8$. The eigenfunctions are centered around the energetic levels $E = \hbar\omega(n + 1/2)$ corresponding to the eigenvalues n.

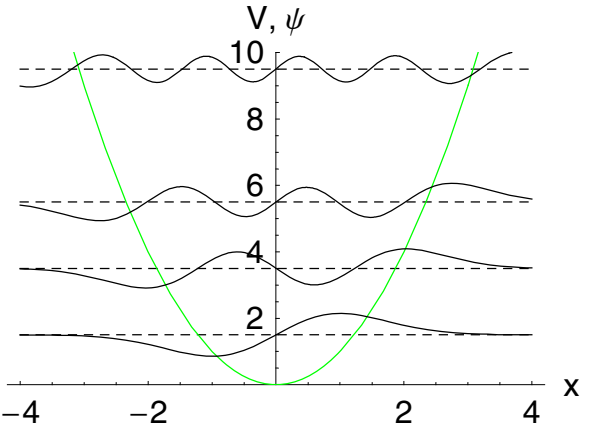


Figure 5.4.11. Antisymmetric eigenfunctions of the harmonic oscillator $V(x) = x^2$ for eigenvalues $n = 1, 3, 5, 9$. The eigenfunctions are centered around the energy levels $E = \hbar\omega(n + 1/2)$ corresponding to the eigenvalues n.

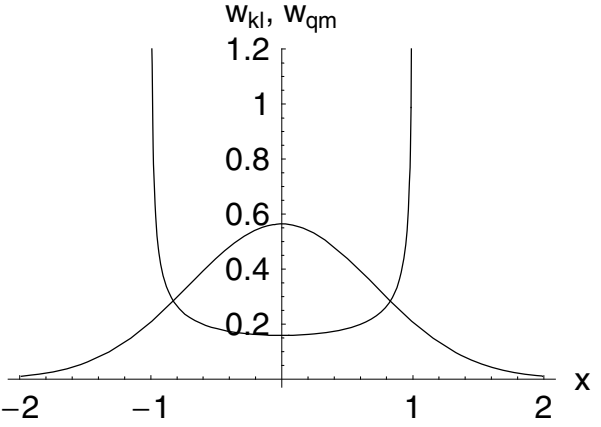


Figure 5.4.12. Classical and quantum mechanical probability density for the harmonic oscillator in the ground state. The classical probability shows a singular behavior at the turning points of the motion.

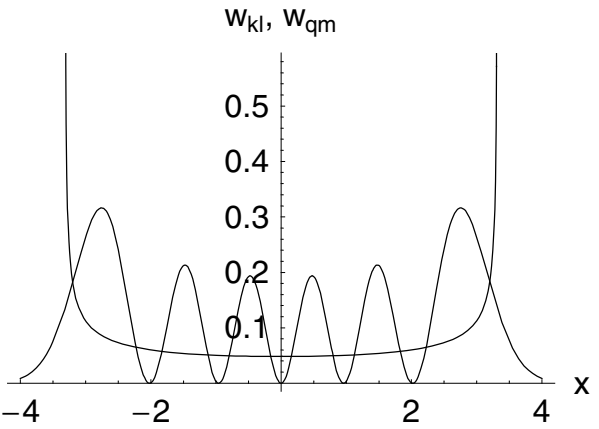


Figure 5.4.13. Comparison between the classical and quantum mechanical probability density for the eigenvalue $n = 5$. The singular points of the classical probability w_{cl} are located at $x = \pm 3.316$.

The given derivation of the wave function is based on the defining equation of the Hermite polynomials (5.4.41). The solution of the scaled equation (5.4.41) delivers the complete set of eigenfunctions in one step. In the following, we show how the set of eigenfunctions can be derived by an iterative procedure involving creation and annihilation operators a^+ and a^-. All of the eigenfunctions are created out of the ground state of the harmonic oscillator,

$$\psi_0(x) = \frac{1}{\sqrt[4]{\pi}}\, e^{-\xi^2/2}. \tag{5.4.52}$$

The whole set of eigenfunctions can be created using the following creation and annihilation operators a^+ and a^-, which act in the spatial and momentum space:

$$a^+ = \frac{1}{\sqrt{2}}\left(\xi - \frac{\partial}{\partial \xi}\right) = \frac{1}{\sqrt{2}}\left(\hat{\xi} - i\,\hat{p}\right), \tag{5.4.53}$$

$$a^- = \frac{1}{\sqrt{2}}\left(\xi + \frac{\partial}{\partial \xi}\right) = \frac{1}{\sqrt{2}}\left(\hat{\xi} + i\,\hat{p}\right). \tag{5.4.54}$$

The name of the operators stems from the action of the wave functions respectively creating and annihilating a quantum mechanical state. The actions of operators a^+ and a^- can be demonstrated by introducing two functions **aminus[]** and **across[]**. The definitions are given below and use the representations of Eqs. (5.4.53) and (5.4.54).

```
aminus[ψ_, ξ_: ξ] := 1/√2 (ξ ψ + ∂ξ ψ)
```

```
across[ψ_, ξ_: ξ] := 1/√2 (ξ ψ - ∂ξ ψ)
```

If we apply the defined functions to the ground state, we get the first excited state or, simply, zero. The definition of the ground state is contained in the function ψ_n.

```
ψn_[ξ_] := 1/√(n! 2^n √π) HermiteH[n, ξ] E^(-ξ^2/2)
```

We get from the application of the generating operator

```
across[ψ0[ξ]]
```

$$\frac{\sqrt{2}\, e^{-\frac{\xi^2}{2}}\, \xi}{\sqrt[4]{\pi}}$$

The anhilation operator appled to the ground state gives

```
aminus[ψ0[ξ]]
```

0

Comparing the *Mathematica* result with the first excited state ψ_1, we find that they are equivalent.

```
across[ψ0[ξ]] == ψ1[ξ] // Simplify
```

True

This is also true if we incorporate the factor $\sqrt{n!}$ on the right-hand side for higher n. The higher eigenfunctions are derived from the ground state by the relation

$$\psi_n(\xi) = \frac{1}{\sqrt{n!}} (a^+)^n \psi_0(x). \tag{5.4.55}$$

Repeatedly applying an operator is achieved by using the function **Nest[]**.

```
Nest[across, ψ0[ξ], 5] // Simplify
```

$$\frac{\sqrt{2}\, e^{-\frac{\xi^2}{2}}\, \xi\, (4\,\xi^4 - 20\,\xi^2 + 15)}{\sqrt[4]{\pi}}$$

We assume that ψ_n is a function of ξ. When using **Nest[]**, we can repeatedly apply the function **across[]** to the wave function **Psi[]**. The number of applications of **across[]** to ψ_n is controlled by the second argument of **Nest[]**. In the above example, we applied **across[]** five times to ψ_n. The result is the representation of ψ_5. If we are interested in the functions preceding ψ_5, we can use **NestList[]** instead.

```
ψList = NestList[across, ψ0[ξ], 5] // Simplify
```

$$\left\{\frac{e^{-\frac{\xi^2}{2}}}{\sqrt[4]{\pi}}, \frac{\sqrt{2}\, e^{-\frac{\xi^2}{2}}\, \xi}{\sqrt[4]{\pi}}, \frac{e^{-\frac{\xi^2}{2}}\, (2\,\xi^2 - 1)}{\sqrt[4]{\pi}}, \frac{\sqrt{2}\, e^{-\frac{\xi^2}{2}}\, \xi\, (2\,\xi^2 - 3)}{\sqrt[4]{\pi}}, \right.$$
$$\left. \frac{e^{-\frac{\xi^2}{2}}\, (4\,\xi^4 - 12\,\xi^2 + 3)}{\sqrt[4]{\pi}}, \frac{\sqrt{2}\, e^{-\frac{\xi^2}{2}}\, \xi\, (4\,\xi^4 - 20\,\xi^2 + 15)}{\sqrt[4]{\pi}}\right\}$$

The unnormalized wave functions contained in the list ψList are eigenfunctions of the harmonic oscillator. To determine the normalization factors, we integrate ψList over the total space:

```
norm = 1 / Sqrt[Map[(Integrate[# d ξ, {ξ, -∞, ∞}]) &, Expand[ψList^2]]]
```

$$\left\{1,\ 1,\ \frac{1}{\sqrt{2}},\ \frac{1}{\sqrt{6}},\ \frac{1}{2\sqrt{6}},\ \frac{1}{2\sqrt{30}}\right\}$$

The normalized eigenfunctions are now given by

```
ψList = ψList norm
```

$$\left\{\frac{e^{-\frac{\xi^2}{2}}}{\sqrt[4]{\pi}},\ \frac{\sqrt{2}\,e^{-\frac{\xi^2}{2}}\,\xi}{\sqrt[4]{\pi}},\ \frac{e^{-\frac{\xi^2}{2}}\,(2\xi^2-1)}{\sqrt{2}\,\sqrt[4]{\pi}},\ \frac{e^{-\frac{\xi^2}{2}}\,\xi\,(2\xi^2-3)}{\sqrt{3}\,\sqrt[4]{\pi}},\right.$$
$$\left.\frac{e^{-\frac{\xi^2}{2}}\,(4\xi^4-12\xi^2+3)}{2\sqrt{6}\,\sqrt[4]{\pi}},\ \frac{e^{-\frac{\xi^2}{2}}\,\xi\,(4\xi^4-20\xi^2+15)}{2\sqrt{15}\,\sqrt[4]{\pi}}\right\}$$

The preceding functions are collected in the package **Harmonic-Oscillator`**. A complete listing is contained in Section 5.8.2.

5.5 Anharmonic Oscillator

So far, we have discussed problems which assume harmonic particle motion. In real systems, harmonic motion is the exception rather than the rule. In general, forces are not proportional to linear displacements. From the example of the pendulum in classical mechanics (see Section 2.4.8.6), we recall that the restoring force is not proportional to linear displacements. Another example is that of large molecules in quantum chemistry: In contrast to the binding potential of a diatomic molecule [5.2], the forces between atoms in a large molecule are anharmonic.

The classical work on anharmonic forces in quantum mechanics was initiated by Pöschel and Teller [5.3], who examined the single anharmonic oscillator. Lotmar [5.4] in 1935 studied an ensemble of anharmonic oscillators and established their connection with large molecules. We examine here an altered Pöschel–Teller potential, which today is used in the inverse scattering method of solving nonlinear evolution equations (see

Chapter 3). The interaction potential for a quantum mechanical system was given by Flügge [5.5] in the form

$$V(x) = -V_0 \operatorname{sech}^2 x, \tag{5.5.56}$$

where V_0 is a constant determining the depth of the potential well. The related stationary Schrödinger equation in scaled variables reads

$$\left(\frac{d^2}{d\,x^2} + \lambda + V_0 \operatorname{sech}^2 x \right) \psi(x) = 0. \tag{5.5.57}$$

```
PTEVproblem = ∂x,x ψ[x] + (λ + V0 Sech[x]^2) ψ[x] == 0
```

$$(V_0 \operatorname{sech}^2(x) + \lambda)\,\psi(x) + \psi''(x) == 0$$

In our examination, we determine the eigenvalues $\lambda = 2\,m\,E/\hbar^2$, which depend on the potential depth V_0. Another point of our study is the form of the wave functions in the asymptotic range $|x| \to \infty$. We first introduce some changes in the notation of Eq. (5.5.57). Substituting for the independent variable x using the relation $\xi = \tanh(x)$ in Eq. (5.5.57), we can carry out the transformation by

```
t1 = PTEVproblem /. ψ → Function[y, ψ[ξ[y]]]
```

$$\psi''(\xi(x))\,\xi'(x)^2 + (V_0 \operatorname{sech}^2(x) + \lambda)\,\psi(\xi(x)) + \psi'(\xi(x))\,\xi''(x) == 0$$

then we replace the new dependent variable ξ by

```
t2 = t1 /. ξ -> Function[x, Tanh[x]]
```

$$\psi''(\tanh(x))\operatorname{sech}^4(x) - 2\tanh(x)\,\psi'(\tanh(x))\operatorname{sech}^2(x) + (V_0 \operatorname{sech}^2(x) + \lambda)\,\psi(\tanh(x)) == 0$$

Using the inverse of the hyperbolic tan, we get

```
t3 = t2 /. x -> ArcTanh[ξ] /. λ -> -λ
```

$$\psi''(\xi)\,(1-\xi^2)^2 - 2\,\xi\,\psi'(\xi)\,(1-\xi^2) + ((1-\xi^2)\,V_0 - \lambda)\,\psi(\xi) == 0$$

which in traditional representation is

$$(1-\xi^2)\,\frac{d}{d\xi}\left((1-\xi^2)\,\frac{d\psi}{d\xi}\right) + (\lambda + V_0(1-\xi^2))\,\psi = 0 \quad \text{where} \quad -1 < \xi < 1, \tag{5.5.58}$$

or the equivalent standard representation of Eq. (5.5.58)

$$\frac{d}{d\xi}\left((1-\xi^2)\,\frac{d\psi}{d\xi}\right) + \left(V_0 + \frac{\lambda}{1-\xi^2}\right)\psi = 0. \tag{5.5.59}$$

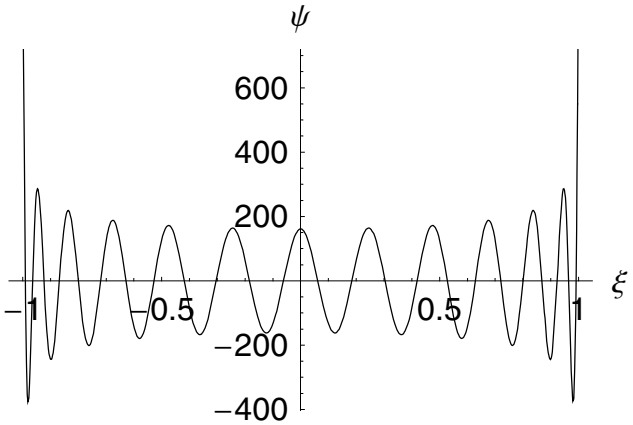
Equation (5.5.59) is the defining equation for the associated Legendre polynomials, which is checked by the line

```
solution = DSolve[t3, ψ, ξ] // Flatten
```

$$\left\{\psi \to \text{Function}\left[\{\xi\},\, c_1\, P^{\sqrt{\lambda}}_{\frac{1}{2}\left(\sqrt{4\,V_0+1}-1\right)}(\xi) + c_2\, Q^{\sqrt{\lambda}}_{\frac{1}{2}\left(\sqrt{4\,V_0+1}-1\right)}(\xi)\right]\right\}$$

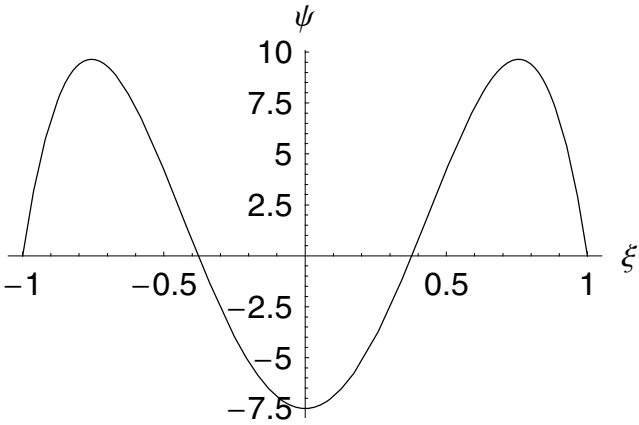
A graphical check of the two Legendre polynomials shows that Legendre Q^n_m is divergent at the boundaries,

```
Plot[Evaluate[
    (ψ[ξ] /. solution /. {V0 → N (N + 1), λ -> n^2}) /.
     {N -> 25, n -> 2, C[1] -> 0, C[2] -> 1}],
  {ξ, -1, 1}, AxesLabel -> {"ξ", "ψ"}];
```



whereas the Legendre P_m^n is finite at the boundaries,

```
Plot[Evaluate[
    (ψ[ξ] /. solution /. {V0 → N (N + 1), λ -> n^2}) /.
     {N -> 4, n -> 2, C[1] -> 1, C[2] -> 0}],
  {ξ, -1, 1}, AxesLabel -> {"ξ", "ψ"}];
```



Thus, for a finite solution of the Pöschel–Teller problem we have to assume that $c_2 = 0$. The solution then becomes

```
solutionPT = solution /. C[2] -> 0
```

$$\left\{\psi \to \text{Function}\left[\{\xi\},\, c_1\, P^{\sqrt{\lambda}}_{\frac{1}{2}\left(\sqrt{4\,V_0+1}-1\right)}(\xi) + 0\, Q^{\sqrt{\lambda}}_{\frac{1}{2}\left(\sqrt{4\,V_0+1}-1\right)}(\xi)\right]\right\}$$

For the solution of Eq. (5.5.59), we assume, in addition, that the potential depth is given by positive integer $V_0 = N\,(N+1)$, where N is a positive number. Equation (5.5.59) possesses discrete bound solutions in the range $\xi \in [-1,\, 1]$ if and only if $\lambda = -n^2 < 0$ with $n = 1,\, 2,\, ...,\, N$. The eigenfunctions of the Schrödinger equation (5.5.59) are proportional to the associated Legendre functions $P_N^n(\xi)$ defined mathematically by

$$P_N^n(\xi) = (-1)^n\,(1-\xi^2)^{n/2}\,\frac{d^n}{d\,\xi^n}\,P_N(\xi), \tag{5.5.60}$$

where $P_N(\xi)$ are the Legendre polynomials of degree N:

$$P_N(\xi) = \frac{1}{N!\,2^N}\,\frac{d^N}{d\,\xi^N}\,(\xi^2-1)^N. \tag{5.5.61}$$

The constant connecting the Legendre functions with the eigenfunctions of the Pöschel–Teller problem is a product of the normalization condition and the eigenfunctions. The following function represents the eigenfunctions of the Pöschel–Teller system. The associated Legendre polynomials are given by the function **LegendreP[]**.

```
PoeschelTeller[x_, n_Integer, N_Integer] :=
  Block[{norm, integrand, xi},
     If[n <= N && n > 0,
(* --- the associated Legendre polynomial specify
the eigenfunction --- *)
     integrand = LegendreP[N, n, xi];
(* --- determine the normalization constant --- *)
     norm = Integrate[integrandˆ2/(1-xiˆ2), {xi, -1,
1}];
(* --- normalize the eigenfunctions --- *)
     integrand = integrand/Sqrt[norm] /. xi ->
Tanh[x];
     Simplify[integrand],
(* --- check errors in the input parameters --- *)
     If[N<n,
     Print["---  wrong argument n > N"]];
     If[n<0,
     Print["---  wrong argument n < 0"]]]
     ]
```

The eigenfunctions for $N = 4$ are

```
Table[PoeschelTeller[x,i,4],{i,1,4}]
```

$$\left\{\frac{1}{4}\sqrt{5}\,\sqrt{\cosh(2x)+\sinh(2x)}\,(\tanh(x)-1)\tanh(x)\,(7\tanh^2(x)-3),\right.$$
$$\frac{1}{4}\sqrt{5}\,(3\cosh(2x)-4)\,\mathrm{sech}^4(x),\ -\frac{1}{4}\sqrt{105}\,\sqrt{\cosh(2x)+\sinh(2x)}$$
$$\left.(\tanh(x)-1)^2\tanh(x)\,(\tanh(x)+1),\ \frac{1}{4}\sqrt{\frac{35}{2}}\,\mathrm{sech}^4(x)\right\}$$

The results for $n = 1$ and $n = 3$ are graphically represented in Figure 5.5.14:

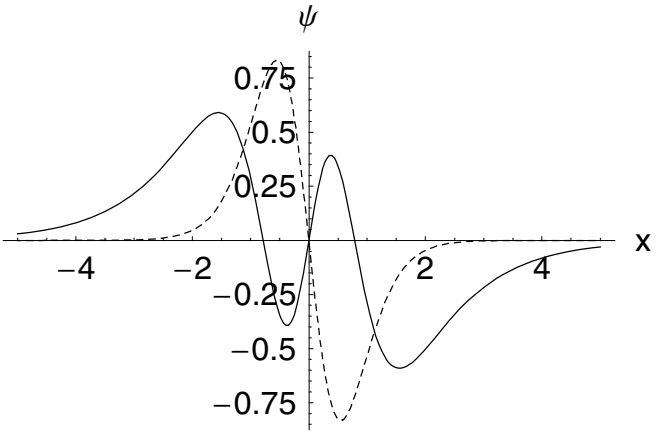


Figure 5.5.14. Eigenfunctions of the modified Pöschel–Teller potential for discrete eigenvalues $n = 1$ (solid) and $n = 3$ (dashed) at $N = 4$.

So far we derived the discrete spectrum of the modified Pöschel–Teller problem. In the following we consider the continuous eigenvalues $\lambda = k^2 > 0$ of the stationary Schrödinger Eq. (5.5.59). The eigenfunctions thus read

$$\psi(x;k) = a(k)\left(\frac{1-\xi^2}{4}\right)^{-ik/2} {}_2F_1\left(\tilde{a},\tilde{b};\tilde{c};\frac{1+\xi}{2}\right), \tag{5.5.62}$$

where $\tilde{a} = 1/2 - ik + \sqrt{V_0 + 1/4}$, $\tilde{b} = 1/2 - ik - \sqrt{V_0 + 1/4}$ and $\tilde{c} = 1 - ik$ are constants depending on the model parameters and the eigenvalues. The label ${}_2F_1$ denotes the Gaussian hypergeometric function. In the limit $x \to \infty$ $\operatorname{sech}(x) = \sqrt{1-\xi^2} = 2e^x/(1+e^{2x}) \sim 2e^{-x}$ and the solution reduces to the form $\psi \sim a(k)\, e^{-ikx}$. The explicit representation in the limit $\xi \to -1$ of the solution (5.5.62) is given by

$$\psi(x;k) = a(k)\, e^{-ikx}\left(1 + \frac{\tilde{a}\tilde{b}}{2\tilde{c}}(1+\xi) + O(\xi^2)\right). \tag{5.5.63}$$

The asymptotic expansion of the hypergeometric function ${}_2F_1$ is carried out by first replacing the argument $\frac{1}{2}(1+\xi)$ with z and then by expanding ${}_2F_1$ up to first order around $z = 0$

```
Series[Hypergeometric2F1[a,b,c,z],{z,0,1}]
```

$$1 + \frac{a\,b\,z}{c} + O(z^2)$$

Hence, the leading term in the asymptotic representation of the eigenfunction ψ for $x \to -\infty$ is

$$\psi \sim a(k)\, e^{-i\,k\,x}. \tag{5.5.64}$$

In the other limit $x \to \infty$, we first transform the hypergeometric function using the linear transformation ${}_2F_1(a, b, c, z) = d\ {}_2F_1(a, b, c, 1 - z)$, yielding

$$\begin{aligned}
&{}_2F_1\left(\frac{1}{2} - i\,k + \sqrt{V_0 + \frac{1}{4}}\,,\right.\\
&\left.\frac{1}{2} - i\,k - \sqrt{V_0 + \frac{1}{4}}\,;\, 1 - i\,k;\, \frac{1+\xi}{2}\right) = \left(\frac{1-\xi}{2}\right)^{i\,k}\\
&{}_2F_1\left(\frac{1}{2} - \sqrt{V_0 + \frac{1}{4}}\,,\ \frac{1}{2} + \sqrt{V_0 + \frac{1}{4}}\,;\, 1 - i\,k;\, \frac{1+\xi}{2}\right)\\
&\qquad = \left(\frac{1-\xi}{2}\right)^{i\,k}\\
&\left({}_2F_1\left(\frac{1}{2} - \sqrt{V_0 + \frac{1}{4}}\,,\ \frac{1}{2} + \sqrt{V_0 + \frac{1}{4}}\,;\, 1 + i\,k;\, \frac{1-\xi}{2}\right)\cdot\right.\\
&\qquad \frac{\Gamma(1+i\,k)\,\Gamma(-i\,k)}{\Gamma\left(\frac{1}{2} - i\,k + \sqrt{V_0+\frac{1}{4}}\right)\Gamma\left(\frac{1}{2} - i\,k - \sqrt{V_0+\frac{1}{4}}\right)} +\\
&\qquad \left(\frac{1-\xi}{2}\right)^{i\,k} {}_2F_1\left(\frac{1}{2} - \sqrt{V_0 + \frac{1}{4}}\,,\ \frac{1}{2} + \sqrt{V_0 + \frac{1}{4}}\,;\right.\\
&\qquad \left.\left.1 - i\,k;\, \frac{1+\xi}{2}\right)\cdot \frac{\Gamma(1-i\,k)\,\Gamma(i\,k)}{\Gamma\left(\frac{1}{2} + \sqrt{V_0+\frac{1}{4}}\right)\Gamma\left(\frac{1}{2} - \sqrt{V_0+\frac{1}{4}}\right)}\right)
\end{aligned} \tag{5.5.65}$$

.

If the potential depth is of the form $V_0 = N\,(N+1)$, we observe that $1/2 - \sqrt{V_0 + 1/4}$ is always a negative integer. Since the function Γ is

singular for these points, the second term on the right hand side always vanishes. Taking this into account (5.5.65) reduces to

$$
\begin{aligned}
&{}_2F_1\left(\frac{1}{2}-i\,k+\sqrt{V_0+\frac{1}{4}}\,,\right. \qquad\qquad .\\
&\left.\frac{1}{2}-i\,k-\sqrt{V_0+\frac{1}{4}}\,;1-i\,k;\,\frac{1+\xi}{2}\right)=\left(\frac{1-\xi}{2}\right)^{i\,k}\\
&{}_2F_1\left(\frac{1}{2}-\sqrt{V_0+\frac{1}{4}}\,,\;\frac{1}{2}+\sqrt{V_0+\frac{1}{4}}\,;1+i\,k;\,\frac{1-\xi}{2}\right)\cdot\\
&\frac{\Gamma(1+i\,k)\,\Gamma(-i\,k)}{\Gamma\left(\frac{1}{2}-i\,k+\sqrt{V_0+\frac{1}{4}}\right)\Gamma\left(\frac{1}{2}-i\,k-\sqrt{V_0+\frac{1}{4}}\right)}
\end{aligned}
\tag{5.5.66}
$$

In the limit $x\to\infty$, the wave function ψ has the representation

$$\psi\sim\ e^{-i\,k\,x}+b(k)\,e^{i\,k\,x},\tag{5.5.67}$$

where $b(k)$ is the reflection coefficient of the wave. Relation (5.5.67) means that an incoming wave of amplitude 1 is reflected by a part determined by $b(k)$.

An asymptotic expansion of the hypergeometric function for $\xi\to 1$ consequently gives us the representation in the form

$$\psi\sim a(k)\,\frac{\Gamma(1+i\,k)\,\Gamma(-i\,k)}{\Gamma\left(\frac{1}{2}-i\,k+\sqrt{V_0+\frac{1}{4}}\right)\Gamma\left(\frac{1}{2}-i\,k-\sqrt{V_0+\frac{1}{4}}\right)}\;e^{-i\,k\,x}.\tag{5.5.68}$$

Comparing relation (5.5.68) with (5.5.67), we observe that the reflection coefficient of the wave vanishes. The transmission coefficient $a(k)$ in the case $V_0=N\,(N+1)$ takes the form

$$a(k)\;=\;\frac{\Gamma\left(\frac{1}{2}-i\,k+\sqrt{V_0+\frac{1}{4}}\right)\Gamma\left(\frac{1}{2}-i\,k-\sqrt{V_0+\frac{1}{4}}\right)}{\Gamma(1+i\,k)\,\Gamma(-i\,k)}.\tag{5.5.69}$$

A wave is free of reflection if the potential takes the form $V=V_0\,\mathrm{sech}(x)$ and the depth of the potential is an integer number $V_0=N\,(N+1)$.

For $V_0=N\,(N+1)$, the entire calculation procedure can be activated by **AsymptoticPT[]** which is part of the package **Anharmonic-Oscillator`**(see Section 5.8.3). By calling **AsymptoticPT[]** we get the asymptotic representation of the eigenfunction in the limits $x\to\pm\infty$. The

results of the expansion are contained in the global variables **w1a** and **w2a**. Function **AsymptoticPT[]** can also handle the case in which N is an integer. In addition to the eigenfunction, function **AsymptoticPT[]** delivers information about the reflection and transmission coefficients $|b|^2$ and $|a|^2$. These two characteristic properties of the scattering problem satisfy $|a|^2 + |b|^2 = 1$. **PlotPT[]**, which is also part of the package **AnharmonicOscillator`**, gives a graphical representation of the reflection and transmission coefficients. This function plots five curves for different k values. The range of the k values is specified as first and second arguments in the function **PlotPT[]**. The third argument of **PlotPT[]** determines the coefficient. We can choose between two types of coefficient. Whereas **"t"** will create a plot for the transition coefficient, the **"r"** string will create the reflection plot. Two examples for $k_{\text{ini}} = 0.05$ and $k_{\text{end}} = 0.5$ are given in Figures 5.5.15 and 5.5.16. The pictures are created by

```
PlotPT[0.05, .5, "r"];
```

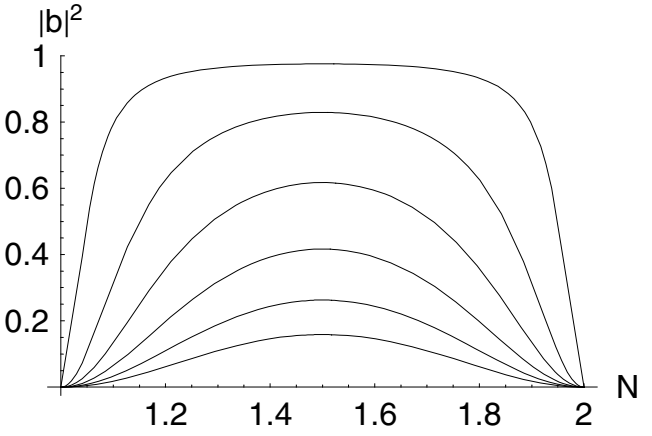


Figure 5.5.15. The reflection coefficient $|b|^2$ is plotted as a function of N. The ensemble of curves represent the reflection coefficient for energy values k in the interval $k \in [0.05, 0.5]$ for $N \in [1, 2]$. The top curve represents the value $k = 0.05$. The other k values > 0.05 follow below the top curve.

and

```
PlotPT[0.05, .5, "t"];
```

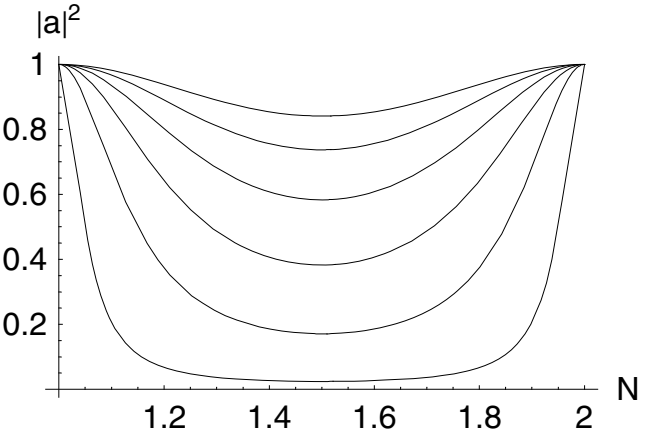


Figure 5.5.16. The transmission coefficient $|a|^2$ of the Pöschel–Teller potential is plotted across the depth parameter N of the potential. The energy values k are taken from the interval $k \in [0.05, 0.5]$ for $N \in [1, 2]$. The lowest curve corresponds to k =0.05.

The structure represented in Figures 5.5.15 and 5.5.16 is repeated in each of the intervals $\{N, N+1 \,|\, N \geq 1\}$. Two neighboring intervals for a potential depth ranging between $V_0 = 2$ and $V_0 = 6$ ($N = 1$ and $N = 2$) are represented in Figure 5.5.17. In this figure, the reflection coefficient is shown for a range of k values by means of a surface plot. The pictures are created by the sequence

```
th = AsymptoticPT[NN, kk];
```

```
Plot3D[Evaluate[th[[2]]], {NN, 1, 3},
 {kk, 0.05, 0.75}, AxesLabel → {"N", "k", "|b|²    "},
 PlotPoints -> 30, Mesh -> False];
Plot3D[Evaluate[th[[1]]], {NN, 1, 3}, {kk, 0.05, 0.75},
 AxesLabel → {"N", "k", "|a|²    "},
 PlotPoints -> 30, Mesh -> False];
```

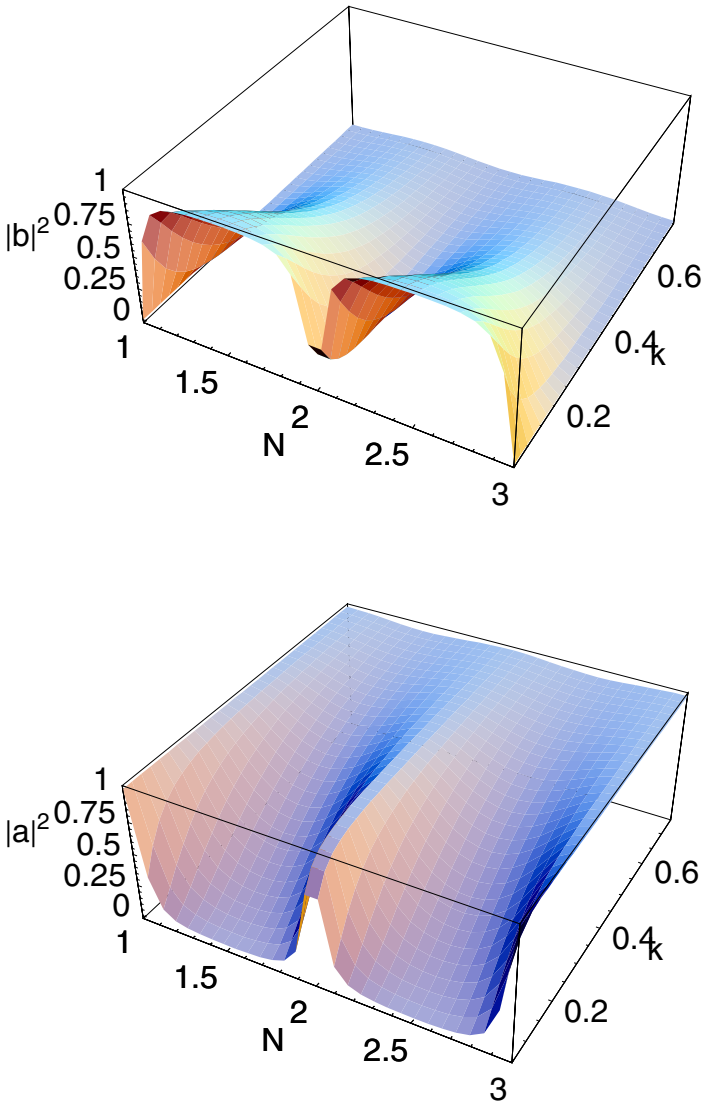


Figure 5.5.17. The reflection and transmission coefficient is plotted as a function of N and k. The values for the potential depth are taken from $N \in [1, 3]$ and the energy interval is $k \in [0.05, 0.75]$. We observe that the reflection coefficient decreases as the energy increases. On the other hand, the transmission coefficient increases with the increase in energy.

A collection of functions examining the anharmonic Pöschel-Teller potential is contained in the package **AnharmonicOscillator`**. Useful

functions in examining the anharmonic model are **PoeschelTeller[]**, **AsymptoticPT[]** and **PlotPT[]** (compare the complete listing in Section 5.8.3).

5.6 Motion in the Central Force Field

The stationary states of a particle in a spherically symmetric potential are determined by the Schrödinger equation with the Hamiltonian operator

$$\hat{H} = -\frac{\hbar^2}{2m}\nabla^2 + V(r), \tag{5.6.70}$$

where $r = \sqrt{x^2 + y^2 + z^2}$ measures the distance of the particle from the origin of the potential. Using the spherical symmetry of the problem, we can rewrite the Schrödinger equation in spherical coordinates

$$\begin{aligned}\Big[-\frac{\hbar^2}{2m}\,\frac{1}{r}\,\frac{\partial^2}{\partial r^2}\,r - \\ \frac{\hbar^2}{2mr^2}\left(\frac{1}{\sin\vartheta}\,\frac{\partial}{\partial\vartheta}\sin\vartheta\,\frac{\partial}{\partial\vartheta} + \frac{1}{\sin^2\vartheta}\,\frac{\partial^2}{\partial\phi^2}\right) + \\ V(r) - E\Big]\psi(r,\vartheta,\phi) = 0,\end{aligned} \tag{5.6.71}$$

or, in a more compact form,

$$\left(-\frac{\hbar^2}{2m}\,\frac{1}{r}\,\frac{\partial^2}{\partial r^2}\,r + \frac{\hbar^2}{2mr^2}\,\hat{L}^2 + V(r) - E\right)\psi = 0, \tag{5.6.72}$$

where $\hat{L}^2$ is the square of the angular momentum operator. Problems which can be identified by such a Hamiltonian operator are very common in physics such as follows:

1. The H-atom
2. An ion with one electron
3. The three-dimensional harmonic oscillator
4. The three-dimensional potential well, quantum dot
5. The Yukawa particle (a shielded Coulomb potential)
6. The free particle.

In close analogy to classical motion in a central field, we find in quantum mechanics that the angular momentum is conserved. The angular momentum is defined by

$$\vec{L} = \vec{x} \times \vec{p}. \tag{5.6.73}$$

Other constants of motion are the Hamiltonian, the square of the angular momentum, and the z-component of the angular momentum. The related operators $\hat{H}$, $\hat{L}^2$, and $\hat{L}_z$ create a complete system of commuting operators. The solutions of the related eigenvalue problems completely determine the properties of the system. As in classical mechanics, we can take advantage of the conservation of angular momentum to reduce a three-dimensional problem to a one-dimensional one. Similarly, we can use the conservation of the angular momentum to separate the coordinates r, ϑ, and ϕ in the Schrödinger equation (5.6.72).

The dependence of the wave function ψ on the angles ϑ and ϕ is determined by the operators $\hat{L}^2$ and $\hat{L}_z$. In spherical coordinates, we can express the z component of the angular momentum by $\hat{L}_z = -i\hbar\partial_\phi$. The eigenvalues of $\hat{L}_z$, are found by solving the equation

$$\frac{\hbar}{i} \frac{\partial\psi(\phi)}{\partial\phi} = L_z \psi(\phi), \tag{5.6.74}$$

where $0 \le \phi \le 2\pi$. The solutions of Eq. (5.6.74) are

$$\psi(\phi) = A\, e^{\frac{i}{\hbar} L_z \phi}. \tag{5.6.75}$$

Since the solution (5.6.75) must be uniquely defined, it has to satisfy the condition

$$\psi(\phi) = \psi(\phi + 2\pi). \tag{5.6.76}$$

The eigenvalues $L_z/\hbar = m$ where $m = 0, \pm 1, \pm 2, \ldots$ satisfy condition (5.6.76). The eigenvalues of the operator $\hat{L}_z$ are thus discrete and represented by

$$L_z = \hbar m, \text{ where } m = 0, \pm 1, \pm 2, \ldots. \tag{5.6.77}$$

Since we require normalized eigenfunctions (i.e., $\int_0^{2\pi} \psi_m^* \psi_m \, d\phi = 1$), the normalized solutions are

$$\psi_m(\phi) = \frac{1}{\sqrt{2\pi}} e^{i m \phi} \tag{5.6.78}$$

A similar treatment yields the eigenvalues and eigenfunctions of the square of the angular momentum $\hat{L}^2$ from the differential equation

$$\hat{L}^2 \psi = L^2 \psi. \tag{5.6.79}$$

In spherical coordinates, the operator $\hat{L}^2$ is represented by

$$\hat{L}^2 = -\hbar^2 \left(\frac{1}{\sin\vartheta} \frac{\partial}{\partial\vartheta} \sin\vartheta \frac{\partial}{\partial\vartheta} + \frac{1}{\sin^2\vartheta} \frac{\partial^2}{\partial\phi^2} \right). \tag{5.6.80}$$

Inserting expression (5.6.80) into Eq. (5.6.79), we get

$$\left(\frac{1}{\sin\vartheta} \frac{\partial}{\partial\vartheta} \sin\vartheta \frac{\partial}{\partial\vartheta} + \frac{1}{\sin^2\vartheta} \frac{\partial^2}{\partial\phi^2} + \frac{L^2}{\hbar^2} \right) \psi(\vartheta, \phi) = 0. \tag{5.6.81}$$

Equation (5.6.81) is the defining equation of the spherical harmonics $Y_{l,m}$ if the eigenvalues satisfy $L^2 = \hbar^2 l(l+1)$ with $l = 0, 1, 2, \ldots$:

$$\left(\frac{1}{\sin\vartheta} \frac{\partial}{\partial\vartheta} \sin\vartheta \frac{\partial}{\partial\vartheta} + \frac{1}{\sin^2\vartheta} \frac{\partial^2}{\partial\phi^2} + l(l+1) \right) Y_{l,m}(\vartheta, \phi) = 0. \tag{5.6.82}$$

The eigenvalues of $\hat{L}^2$ are determined by the quantum numbers $l = 0, 1, 2, \ldots$. Their related eigenfunctions are the spherical harmonics $Y_{l,m}$ of order l. Comparing the structure of the eigenfunctions of the harmonic oscillator to that of the eigenfunctions of the angular momentum $\hat{L}^2$, we observe that in the case of $\hat{L}^2$ with eigenvalues $L^2 = \hbar^2 l(l+1)$, there are $2l+1$ eigenfunctions $Y_{l,m}$. The eigen- functions $Y_{l,m}$, however, are different in the second quantum number m, which is known as the magnetic quantum number. For a fixed value of L^2, m counts the different projections on the z-axis. If we determine l, we find different values for m

$$m = 0, \pm 1, \pm 2, \ldots, \pm l \tag{5.6.83}$$

and limited to the range $-l \le m \le l$. For the proof of the above relations, we refer the reader to the book by Cohen-Tannoudji et al. [5.6].

The complete representation of the spherical harmonics for positive m is

$$Y_{l,m}(\vartheta, \phi) = \frac{(-1)^m}{\sqrt{2\pi}} e^{i m \phi} \sqrt{\frac{(2l+1)(l-m)!}{2(l+m)!}} \sin^m\vartheta P_l^m(\cos\vartheta). \tag{5.6.84}$$

$P_l^m(x)$ denotes the mth associated Legendre function of order l. In case of negative quantum numbers m, we use the relation

$$Y_{l,-m}(\vartheta, \phi) = (-1)^m Y_{l,m}^*(\vartheta, \phi). \tag{5.6.85}$$

If we use the representation of the spherical harmonics given by relation (5.6.84), it is easy to show that the $Y_{l,m}$ are also eigenfunctions of the operator $\hat{L}_z$. By a simple calculation, we find

$$\frac{\hbar}{i} \frac{\partial}{\partial\phi} Y_{l,m}(\vartheta, \phi) = \hbar m Y_{l,m}(\vartheta, \phi). \tag{5.6.86}$$

We now can state that the spherical harmonics are eigenfunctions of both the z-component of the angular momentum operator and the square of the angular momentum operator. The corresponding eigenvalues are

$$L^2 = \hbar^2 l\,(l+1) \qquad \text{and} \qquad L_z = \hbar m. \tag{5.6.87}$$

The spherical harmonics are accessed in *Mathematica* by the function **SphericalHarmonicY[]** available in the package CentralField` in Section 5.8.4. The Legendre polynomials are available using **LegendreP[]**.

So far, we have determined the eigenfunctions depending on ϑ and ϕ. Separating the angular terms from the radial part of the wave function, we get the representation

$$\psi(r, \vartheta, \phi) = h(r)\, Y_{l,m}(\vartheta, \phi). \tag{5.6.88}$$

Relation (5.6.88) used with Eq. (5.6.72) allows the derivation of a determining equation for the radial part $h(r)$ of the wave function ψ. The wave function separates because the coordinate system of our problem is separable. The radial function $h(r)$ is dependent on the energy E, the quantum number l, and the potential energy $V(r)$. Consequently, the radial part of the wave function is independent of m: In a spherical potential, there are no distinguishing directions breaking the symmetry.

Inserting relation (5.6.88) into the Schrödinger equation (5.6.72) and using our above results for the angular momentum, we get, after substituting $u(r) = r\,h(r)$, the eigenvalue problem for the radial part of the wave function

$$\left(-\frac{\hbar^2}{2m}\,\frac{d^2}{d\,r^2} + V(r) + \frac{\hbar^2\, l\,(l+1)}{2\,m\,r^2}\right) u(r) = E\,u(r). \tag{5.6.89}$$

$u(r) = r\,h(r)$ is substituted since for $r \to 0$, the function $h(r)$ has to be finite (i.e., $u(r) \to 0$ for $r \to 0$). Note that in Eq. (5.6.89), all parameters are known except for potential $V(r)$. For the following discussion, we assume that the potential $V(r)$ represents a Coulomb interaction of the two particles,

$$V(r) = -\frac{Z\,e^2}{r}. \tag{5.6.90}$$

This type of potential applies to the hydrogen and hydrogenlike atoms where $Z = 1$ as well as to ionized atoms like He^+, Li^{2+}, and so forth.

The stationary states of an electron in a Coulomb potential result from the eigenvalue equation

$$\left(\frac{d^2}{d\,r^2} + \frac{2\,m\,E}{\hbar^2} + \frac{2\,m\,Z\,e^2}{\hbar^2\,r} - \frac{l\,(l+1)}{r^2}\right) u(r) = 0. \tag{5.6.91}$$

To carry out our calculation, it is convenient to introduce scaled variables

$$\rho = \frac{r}{a} \qquad \text{and} \qquad \varepsilon = \frac{E}{E_0}, \tag{5.6.92}$$

where $a = \hbar^2/(m\,e^2) \approx 5.29 \times 10^{-11}\,\text{m}$ is Bohr's radius and $E_0 = e^2/(2\,a) = m\,e^4/\hbar^2 \approx 13.5\,\text{eV}$, the ionization energy of the hydrogen atom. The Schrödinger equation (5.6.91) is thus represented by

$$\left(\frac{d^2}{d\,\rho^2} + \varepsilon + \frac{2\,Z}{\rho} - \frac{l\,(l+1)}{\rho^2}\right) u(\rho) = 0, \tag{5.6.93}$$

which allows a representation as

```
radialEVProblem = ∂ρ,ρ u[ρ] + (ϵ + 2 Z/ρ - l (l + 1)/ρ^2) u[ρ] == 0
```

$$\left(-\frac{l\,(l+1)}{\rho^2} + \epsilon + \frac{2\,Z}{\rho}\right) u(\rho) + u''(\rho) == 0$$

We restrict our calculations to the case of bound states characterized by negative energy values. To find appropriate representations of a solution ansatz for $u(r)$, we examine the limits $r \to 0$ and $r \to \infty$. The function $u(r)$ is either given by a polynomial in ρ $u_l(\rho) = r^{\alpha}(1 + a_1\,\rho + a_2\,\rho^2 + ...)$ or by an exponential relation $u_l = A\,e^{-\gamma\,\rho} + B\,e^{\gamma\,\rho}$, where $\gamma^2 = -\varepsilon$. The results of these expressions are conditions for the parameters α and B which satisfy $\alpha = l + 1$, $B = 0$. Using these results both expressions are reducible to

$$u_l(\rho) = \rho^{l+1}\,e^{-\gamma\,\rho}\,f(\rho) \tag{5.6.94}$$

or, in a manageable form,

```
tr1 = u -> Function[ρ, ρ^(l+1) ℯ^(-γ ρ) f[ρ]]
```

$u \to \text{Function}[\rho,\ \rho^{l+1}\,e^{-\gamma\,\rho}\,f(\rho)]$

Substituting expressions (5.6.94) into Eq. (5.6.93) and using $x = 2\gamma\rho$, we get the standard form of Kummer's differential equation:

$$x f'' + (2(l+1) - x) f' - \left(l + 1 - \frac{Z}{\gamma}\right) f = 0, \tag{5.6.95}$$

where primes denote differentiation with respect to x. The *Mathematica* version of this transformation using original variables is gained by

```
g1 = radialEVProblem /. tr1 // Simplify
```

$$e^{-\gamma\rho}\,\rho^{l}\left((\rho\gamma^2 - 2(l+1)\gamma + 2Z + \epsilon\rho) f(\rho) + 2(l - \gamma\rho + 1) f'(\rho) + \rho f''(\rho)\right) == 0$$

The solution can be directly derived from

```
solution = DSolve[g1, f, ρ] // Flatten
```

$$\left\{f \to \text{Function}\left[\{\rho\}, e^{(\gamma - i\sqrt{\epsilon})\rho} c_1 U\left(-\frac{-\sqrt{\epsilon}\,l - iZ - \sqrt{\epsilon}}{\sqrt{\epsilon}}, 2l+2, 2i\sqrt{\epsilon}\,\rho\right) + e^{(\gamma - i\sqrt{\epsilon})\rho} c_2 L^{2l+1}_{\frac{-\sqrt{\epsilon}\,l - iZ - \sqrt{\epsilon}}{\sqrt{\epsilon}}}\left(2i\sqrt{\epsilon}\,\rho\right)\right]\right\}$$

which simplifies if we assume that the energies ϵ are negative:

```
f[ρ] /. solution /. ϵ -> -ϵ // PowerExpand // Simplify
```

$$e^{(\gamma + \sqrt{\epsilon})\rho}\left(c_1 U\left(l + \frac{Z}{\sqrt{\epsilon}} + 1, 2l+2, -2\sqrt{\epsilon}\,\rho\right) + c_2 L^{2l+1}_{-l - \frac{Z}{\sqrt{\epsilon}} - 1}\left(-2\sqrt{\epsilon}\,\rho\right)\right)$$

The solutions of Eq. (5.6.95) are, in general, confluent hypergeometric functions (${}_1F_1$)

$$f_l(\rho) = c\ {}_1F_1\left(l + 1 - \frac{Z}{\gamma}, 2l+2; 2\gamma\rho\right) \tag{5.6.96}$$

reducing to Laguerre's and Kummer's function. To satisfy the normalization condition, series (5.6.96) must terminate at a finite order.

This restriction excludes Kummer's function ($C_1 = 0$) and induces the quantization of the energy values by

$$l + 1 - \frac{Z}{\gamma} = -n_r, \qquad \text{with} \quad n_r = 0,\ 1,\ 2,\ .... \tag{5.6.97}$$

The solution of Eq. (5.6.97) with respect to γ delivers

$$\gamma = \frac{Z}{n_r + l + 1}, \tag{5.6.98}$$

or, by replacing $\gamma^2 = -\varepsilon$, yields energy values $\varepsilon = -Z^2/(n_r + l + 1)^2$ to be

$$E = -\frac{Z^2}{(n_r + l + 1)^2} E_0 = -\frac{Z^2}{n^2} E_0. \tag{5.6.99}$$

The quantum number n is the principal quantum number determined by the radial quantum number n_r ($n_r = 0, 1, 2, ...$) and the angular quantum number l ($l = 0, 1, 2, \ldots$). The wave function of the electron in the Coulomb potential is given by

$$\psi_{n,l,m}(\rho, \vartheta, \phi) = N_{n,l}\, \rho^{Z\rho/n}\ {}_1F_1\left(l + 1 - n,\ 2l + 2;\ \frac{2Z}{n}\rho\right) Y_{l,m}(\vartheta, \phi), \tag{5.6.100}$$

where $N_{n,l}$ is the normalization constant

$$N_{n,l} = \frac{1}{(2l+1)!}\sqrt{\frac{(n+l)!}{2n(n-l-1)!}}\left(\frac{2Z}{n}\right)^{l+3/2}. \tag{5.6.101}$$

The radial part of the wave function $h(\rho)$ consists of

$$h_{n,l}(\rho) = N_{n,l}\, \rho^l\, e^{-Z\rho/n}\ {}_1F_1\left(l + 1 - n,\ 2l + 2;\ \frac{2Z}{n}\rho\right). \tag{5.6.102}$$

Since the first argument in the hypergeometric function is a negative integer, the function ${}_1F_1$ in the radial part reduces to a polynomial known as a Laguerre polynomial. In *Mathematica,* the Laguerre polynomials are denoted by **LaguerreL[]**. One useful parameter of the radial wave function is $n_r = n - l - 1$. This parameter counts the nodes of the eigenfunction along the horizontal axis. This behavior is shown in Figure 5.6.18 for $n = 3$ and $l = 0, 1, 2$. Figure 5.6.18 is created by

```
Plot[{Radial[r, 3, 0, 1], Radial[r, 3, 1, 1],
   Radial[r, 3, 2, 1]}, {r, 0, 25},
  AxesLabel → {"r", "h"}, Prolog → Thickness[0.001]];
```

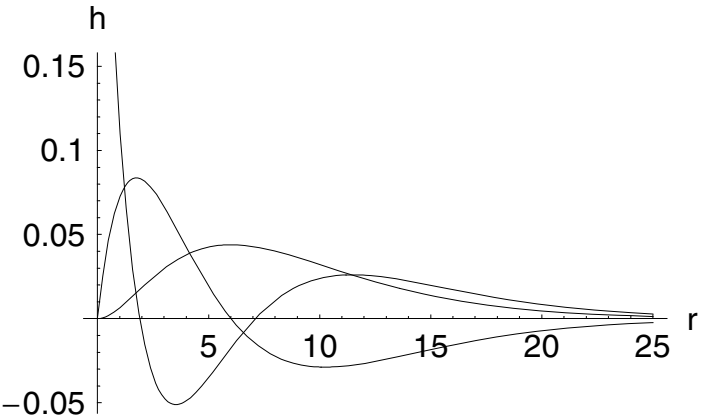


Figure 5.6.18. Radial part h of the wave function for $n = 3$ and $l = 0, 1, 2$.

The function **Radial[]** used in the **Plot[]** function is part of the package **CentralField`**. This package also contains **Angle[]** for the angular part of the wave function. The definition of **Angle[]** is, in some ways, redundant since *Mathematica* accounts for the angular part of the wave function under the name **SphericalHarmonicY[]**. However, we separately define the angular part of the wave function to show how relations (5.6.84) and (5.6.85) are expressed in terms of *Mathematica*.

The above wave function is applied to representations of orbitals of an atom or a molecule. Chemists, for example, work with molecular orbital theory to describe the binding of atoms. This theory makes extensive use of the angular wave functions $Y_{l,m}$. In order to describe the binding of a molecule, it is necessary to use a linear combination of the angular parts of the wave function. We create such a superposition of the $Y_{l,m}$'s by the function **Orbital[]**, which is part of the package **CentralField`**. **Orbital[]** creates sums and differences of the spherical harmonics in the form

$$w(\vartheta, \phi) = (\,|\,Y)_{l,m} \pm Y_{l,-m}\,|^2. \tag{5.6.103}$$

Relation (5.6.103) represents the probability of finding an electron within a certain domain of the angular part of the space. In Figures 5.6.19-22, we have plotted some particular examples for orbitals.

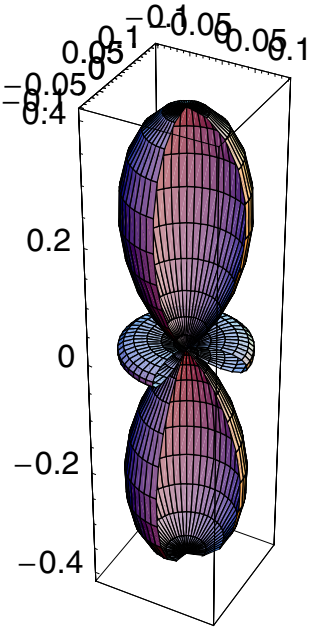


Figure 5.6.19. Angular part of the wave function for $l = 2$ and $m = 0$.

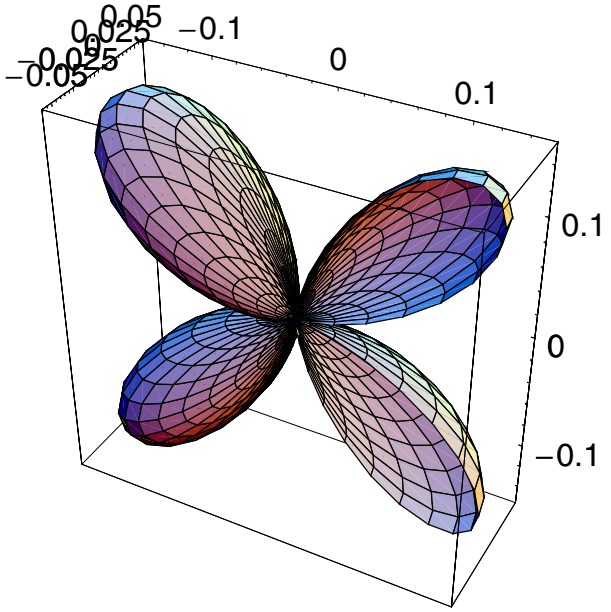


Figure 5.6.20. Orbital for the quantum numbers $l=2$ and $m=\pm 1$ formed from the difference $|Y_{2,1}-Y_{2,-1}|^2$.

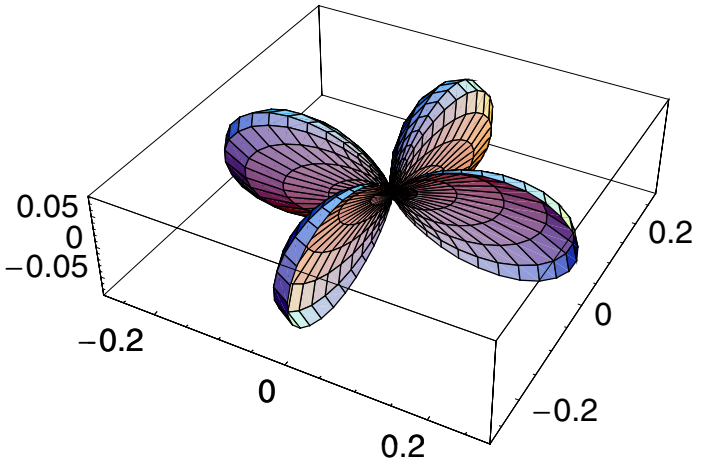


Figure 5.6.21. A plot of the sum of the wave functions with quantum numbers $l=2$ and $m=\pm 2$.

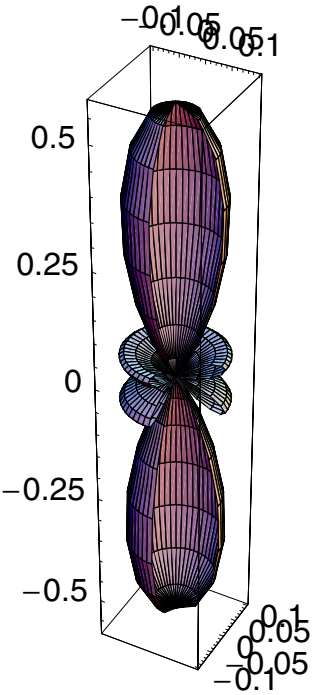
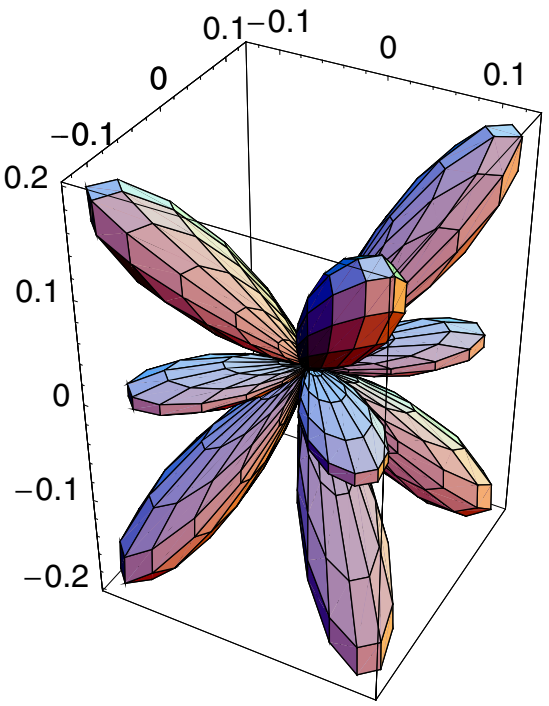


Figure 5.6.22. Representation of the orbital $|Y|^2$ for quantum numbers $l=3$ and $m=0$.

Figures 5.6.19-22 show an inner view of the orbitals for a certain range of ϕ. Similar pictures for other quantum numbers are created by the superposition of the angular wave functions $Y_{l,m}$ with the help of **Angle[]**. The figures of the orbitals are created by the function sequence

AnglePlot[Orbital[l,m,ϑ,-ϕ,"plus"],ϑ,ϕ]. An example of the appli- cation of this function is given below.

```
AnglePlot[Orbital[θ, ϕ, 4, 2, "minus"], θ, ϕ];
```



5.7 Second Virial Coefficient and Its Quantum Corrections

Nearly 100 years ago, Kannerligh Onnes discribed the thermodynamic behavior of a gas in form of an equation which should become as virial equation of state one of the most successful theories for the link between the microscopic physics of molecular interactions and macroscopic thermodynamic properties:

$$\frac{P\,V}{R\,T} = 1 + \frac{B(T)}{V} + \frac{C(T)}{V^2} + \frac{D(T)}{V^3} + \ldots, \tag{5.7.104}$$

where $B(T)$, $C(T)$, and $D(T)$ are the second, third, and fourth viral coefficients of increasing complexity, R is the gas constant, V is the volume, and T is the absolute temprature in the virial equation.

Immediately after the introduction of the virial equation, Ornstein calculated the second virial coefficient (SVC) using Gibb's statistical calculation techniques

$$B(T) = -2\pi N_A \int_0^\infty \left(\frac{e^{-U(r)/r}}{k_B T} - 1 \right) r^2 \, dr \tag{5.7.105}$$

where N_A is Avogadro's constant, $U(r)$ is the intermolecular potential, and k_B is the Boltzmann constant. The exciting history of the virial equation and its relation to the phenomenological van der Waals equation as well as the history of the calculation of $B(T)$ for various molecular potentials is covered in an excellent article by Rowlinson [5.7]. He discusses the van der Waals equation and its implications to the development of the real gas and the liquid. In spite of the strong influence of the van der Waals equation on the study of molecular interactions, it could not describe accurately the behavior of any substance. Rowlinson points out how an empirical proposal of Onnes was combined with the theoretical development of Gibbs and Ornstein to produce the viral equation of state, one of the most useful theories of any state of matter.

Before the theory was worked out completely and before the quantum theory of the intermolecular potential was developed, the second virial coefficient (SVC) was investigated by interaction potentials of the kind

$$U(r) = \left(\frac{A}{r^m} - \frac{B}{r^n} \right) \tag{5.7.106}$$

mostly associated with Lennard-Jones [5.8]. After the derivation of the dispersion forces proportional to r^{-6} by London [5.9], the $(12-6)$-potential has become very popular. Theory and numerical results of this and related potential are discussed in detail in the classical monographs by Hirschfelder et al. [5.10] on the molecular theory of gases and liquids and by Mason and Spurling [5.11] on the viral equation of state. As will be pointed out subsequently, the SVC is an integral over a function of $U(r)$. In teaching statistical thermodynamics, however, one wants to give a final result not as an integral but as an explicit function of the temperature and

molecular parameters. Especially for the $(m-n)$-Lennard-Jones potential (abbreviated by (m,n)-LJ) analytical results in terms of series expansions with the Γ function have been given in [5.12]. It was pointed out, however, by several authors, also in recent textbooks that for the potential, especially the $12-6$, no closed solution exists. That this statement is not correct will be shown subsequently in the sketch on analytical approaches to the SVC. What is lacking, however, is a consistent derivation of the SVC, its quantum corrections, and its temperature derivatives from one integral. The present section aims at such a unified derivation. Also, other results in the literature will be reduced to these results.

5.7.1 The SVC and Its Relation to Thermodynamic Properties

The necessary formulas for the SVC and its quantum corrections are collected to show the importance for thermodynamic functions. The virial equation of state was given in Eq. (5.7.104). A knowledge of the virial coefficients and their temperature dependence describes the pVT behavior of the gas completely, if one assumes the convergence of the series. For the classical part B_c of the $B(T)$, one derives

$$\begin{aligned} B_c(T) &= 2\pi N_A \int_0^\infty (e^{-\beta U(r)} - 1)\, r^2\, dr \\ &= -\frac{2\pi N_A \beta}{3} \int_0^\infty e^{-\beta U(r)} \left(\frac{dU}{dr}\right) r^3\, dr \end{aligned} \tag{5.7.107}$$

after partial integration. N_A is Avogadro's number, $\beta = (k_B T)^{-1}$, k_B is the Boltzmann constant, and $U(r)$ is the interatomic or intermolecular potential. The index c on B denotes the purely classical part of our considerations. For low temperatures and light atoms and molecules like He, Ne, and H_2, one has to take quantum mechanics into account. It was shown with the $(12-6)$ potential for He that at very low temperatures, the full quantum mechanical calculation has to be performed, but for temperatures above 5K, the semiclassical expansion without the symmetry term is sufficient:

$$B = B_c + \frac{\hbar^2}{m} B_{q_1} + \left(\frac{\hbar^2}{m}\right)^2 B_{q_2} + \ldots \tag{5.7.108}$$

with

$$B_{q_1} = \frac{\pi N_A \beta^3}{6} \int_0^\infty e^{-\beta U} (U')^2\, r^2\, dr \tag{5.7.109}$$

and

$$B_{q_2} = -\frac{\pi N_A \beta^4}{6} \int_0^\infty e^{-\beta U}\left[\frac{(U'')^2}{10} + \frac{(U')^2}{5r^2} + \frac{\beta(U')^3}{9r} - \frac{\beta^2(U')^4}{72}\right] r^2 dr. \tag{5.7.110}$$

The SVC is important for the correct calculation of thermodynamic functions at high temperatures, as it includes not only the bound states usually only taken into account in the calculation of partition functions but also meta-stable and continuum states. This was shown explicitly for a Rydberg diatomic potential by Sinanoglu and Pitzer [5.13]; a more recent discussion on the splitting of the phase space of the SVC was given by Friend [5.14].

The thermodynamic functions related to the SVC, B, and its temperature derivatives $B_n = T^n (d^n B / dT^n)$ are given by the internal energy

$$\frac{\tilde{U} - \tilde{U}^0}{RT} = -\left(\frac{B_1}{\tilde{V}} + \ldots\right), \tag{5.7.111}$$

the enthalpy

$$\frac{\tilde{H} - \tilde{H}^0}{RT} = \frac{B - B_1}{\tilde{V}} + \ldots, \tag{5.7.112}$$

the entropy

$$\frac{\tilde{S} - \tilde{S}^0}{R} = -\left\{\ln p + \frac{B_1}{\tilde{V}} + \frac{B^2}{2\tilde{V}^2} \ldots\right\}, \tag{5.7.113}$$

and the specific heat

$$\frac{\tilde{C}_p - \tilde{C}_p^0}{RT} = -\left\{\frac{B_2}{\tilde{V}} - \frac{(B - B_1)^2}{\tilde{V}} + \ldots\right\} \tag{5.7.114}$$

$$\mu_{JT} C_p^0 = -[B - T B'] + \frac{1}{\tilde{V}}\left[2B^2 - 2T B B' - \frac{RT^2}{C_p^0}(B - T B') B'' + \ldots\right] + \ldots. \tag{5.7.115}$$

Thermodynamic functions give the extent of the values from the value of a perfect gas in its normal state denoted by a superscript (°); the tilde (~) represents molar quantities. From these formulas follows that for a complete analytical theory of the SVC and for thermodynamic functions with two-body interactions, one has to calculate

$$B(T) = B_c(T) + B_{q_1}(T) + B_{q_2}(T) + \ldots \tag{5.7.116}$$

5.7.2 Calculation of the Classical SVC $B_c(T)$ for the $(2n - n)$-Potential

A useful method of evaluating the thermodynamic properties of gases at high temperatures is to treat the entire gas as a monoatomic assembly with gas imperfections given by

$$\frac{PV}{RT} = 1 + \frac{B(T)}{V} + \frac{C(T)}{V^2} + \frac{D(T)}{V^3} + \ldots, \tag{5.7.117}$$

where $B(T)$, $C(T)$, and $D(T)$ denote the second, third, and fourth virial coefficients, respectively. Our interest here is the second virial coefficient $B(T)$ (SVC) and its quantum mechanical corrections up to second order. All of the thermodynamic properties of the gas are then obtained directly from the equation of state as represented by Eq. (5.7.117).

In the following calculations, we will examine the two-parameter Lennard-Jones potential (LJ):

$$\mathbf{LJ} = 2\left(\frac{n\epsilon}{n-m}\right)^{\frac{m}{n-m}}\left(\left(\frac{\sigma}{r}\right)^n - \left(\frac{\sigma}{r}\right)^m\right)$$

$$2\left(\frac{n\in}{-m+n}\right)^{\frac{m}{-m+n}}\left(-\left(\frac{\sigma}{r}\right)^m + \left(\frac{\sigma}{r}\right)^n\right)$$

where ϵ is the well depth and σ is the internuclear distance. Our interest is mainly concerned with the case when $m = n$ and n is replaced by an even number of m. As a two-parameter potential with ϵ, σ, the $(m - n)$-potential, is simple but not very flexible. An additional parameter is introduced in the spherical Kihara hard-core potential [5.15]:

$$\mathbf{Kihara} = \frac{2n\left(\frac{n}{m}\right)^{\frac{m}{n-m}}\epsilon\left(\left(\frac{\sigma-2a}{r-2a}\right)^n - \left(\frac{\sigma-2a}{r-2a}\right)^m\right)}{n-m}$$

$$\frac{2n\left(\frac{n}{m}\right)^{\frac{m}{-m+n}}\in\left(-\left(\frac{-2a+\sigma}{-2a+r}\right)^m + \left(\frac{-2a+\sigma}{-2a+r}\right)^n\right)}{-m+n}$$

The Kihara potential is ∞ for $r < 2a$ and is connected with the LJ potential if we replace the radial coordinate r, the potential depth ϵ, and intermolecular distance by

```
transforms = {r → 2 a + r, ϵ → ((n - m) (m ϵ/(n - m))^(m/(n - m)))/n, σ → 2 a + σ};
```

Applying these transformations to the Kihara potential, we find

```
tK = Simplify[Kihara /. transforms]
```

$$2\left(\frac{n}{m}\right)^{\frac{m}{n-m}}\left(\frac{m\epsilon}{n-m}\right)^{\frac{m}{n-m}}\left(\left(\frac{\sigma}{r}\right)^{n}-\left(\frac{\sigma}{r}\right)^{m}\right)$$

Comparing the LJ potential with the transformed Kihara potential, we observe their equivalence:

```
PowerExpand[tK == LJ]
```

True

meaning that both potentials are identical. Thus, we can unify the calculations for one type of potential. We therefore restrict our considerations to the LJ potential. We note that the following results are also valid in case of the Kihara potential. Our main interest is concerned with a subclass of LJ potentials where the exponents (n, m) are given by an even integer and the integer itself. For such a combination, the LJ potential reduces to a $(2n - n)$-potential, which is given by

```
U(r) = LJ /. {n → 2 n, m → n}
```

$$4\epsilon\left(\left(\frac{\sigma}{r}\right)^{2n}-\left(\frac{\sigma}{r}\right)^{n}\right)$$

The first derivative, the intermolecular force, needed to evaluate (5.7.107) follows from the potential by differentiating $U(r)$ with respect to r:

```
Force = Simplify[-∂U(r)/∂r]
```

$$\frac{4\,n\,\epsilon\,(\frac{\sigma}{r})^n\,(2\,(\frac{\sigma}{r})^n-1)}{r}$$

Inserting the potential $U(r)$ and the force into Eq. (5.7.107), we find

```
B_c = 1/3 (2 π N_A β) ∫_0^∞ e^(-β U(r)) Force r^3 dr
```

Integrate::gener : Unable to check convergence. More…

$$\frac{2}{3}\,\pi\,\beta\left(\int_0^\infty 4\,e^{-4\,\beta\,\epsilon\,((\frac{\sigma}{r})^{2n}-(\frac{\sigma}{r})^n)}\,n\,r^2\,\epsilon\left(\frac{\sigma}{r}\right)^n\left(2\left(\frac{\sigma}{r}\right)^n-1\right)dr\right)N_A$$

At first glance, the result is disappointing because *Mathematica* does not evaluate the integral. However, it returns the integral containing the explicit expressions for the potential U and its first derivative. A second examination of the integral reveals that we found a Laplace transform of the first derivative of U, the negative force. To recognize that the above integral represents a Laplace transform, let us introduce the following substitutions:

```
substitution = {r → σ t^(-1/n), DifferentialD(r) → ∂(σ t^(-1/n))/∂t DifferentialD(t)};
```

Applying this substitution to the integrand B_c, we are able to reduce (5.7.107) to a Laplace integral. The integrand of this integral is calculated by the transformation

```
integrand = PowerExpand[
    1/3 ((-2 π N_A β)) e^(-β U(r)) Force r^3 DifferentialD(r) /. substitution /.
      DifferentialD(t) → 1]
```

$$\frac{8}{3}\, e^{-4\,(t^2-t)\,\beta\,\epsilon}\,\pi\, t^{-3/n}\,(2\,t-1)\,\beta\,\epsilon\,\sigma^3\,N_A$$

Inserting the new integrand into the classical part of the SVC, we find

```
Bc = ∫_0^∞ integrand dt
```

Integrate::gener : Unable to check convergence. More…

$$\text{If}\Big[\text{Re}(\beta\,\epsilon) > 0 \bigwedge \text{Re}\Big(\frac{1}{n}\Big) < \frac{1}{3},\ \frac{1}{3}\,2^{\frac{n+3}{n}}\,\pi\,(\beta\,\epsilon)^{\frac{3}{2n}}\,\sigma^3$$
$$\Big(\Gamma\Big(1-\frac{3}{2\,n}\Big)\Big({}_1F_1\Big(1-\frac{3}{2\,n};\ \frac{1}{2};\ \beta\,\epsilon\Big) - 2\,\beta\,\epsilon\,{}_1F_1\Big(1-\frac{3}{2\,n};\ \frac{3}{2};\ \beta\,\epsilon\Big)\Big) +$$
$$\sqrt{\beta\,\epsilon}\,\Big(2\,\Gamma\Big(\frac{3\,(n-1)}{2\,n}\Big)\,{}_1F_1\Big(\frac{3\,(n-1)}{2\,n};\ \frac{3}{2};\ \beta\,\epsilon\Big) -$$
$$\Gamma\Big(\frac{n-3}{2\,n}\Big)\,{}_1F_1\Big(\frac{n-3}{2\,n};\ \frac{1}{2};\ \beta\,\epsilon\Big)\Big)\Big)\,N_A,$$
$$\text{Integrate}\Big[\frac{16}{3}\,e^{-4\,(t-1)\,t\,\beta\,\epsilon}\,\pi\,t^{1-\frac{3}{n}}\,\beta\,\epsilon\,\sigma^3\,N_A - \frac{8}{3}\,e^{-4\,(t-1)\,t\,\beta\,\epsilon}\,\pi\,t^{-3/n}\,\beta\,\epsilon\,\sigma^3\,N_A,$$
$$\{t,\,0,\,\infty\},\ \text{Assumptions} \to \text{Re}\Big(\frac{1}{n}\Big) \geq \frac{1}{3} \bigvee \text{Re}(\beta\,\epsilon) \leq 0\Big]\Big]$$

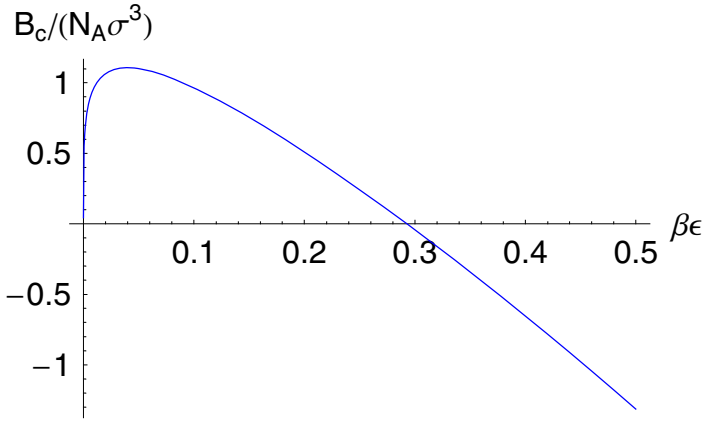
The result shows that under the conditions $\text{Re}(\frac{1}{n}) < \frac{1}{3}$ and $\text{Re}(\beta\,\epsilon) > 0$, the integral exists and the SVC is represented by hypergeometric functions ${}_1F_1$ depending on the potential parameter n, the inverse temperature β, and the potential depth ϵ. If the conditions on n and $\beta\,\epsilon$ are not satisfied, we observe that the integral cannot be evaluated. A more usable representation of the result for our further calculations is generated if we supress the conditions under which the integral is solvable. We select

```
Bc = Bc /. a_. If[b_, c_, d___] -> a c
```

$$\frac{1}{3}\,2^{\frac{n+3}{n}}\,\pi\,(\beta\epsilon)^{\frac{3}{2n}}\,\sigma^3$$
$$\left(\Gamma\left(1-\frac{3}{2n}\right)\left({}_1F_1\left(1-\frac{3}{2n};\frac{1}{2};\beta\epsilon\right)-2\,\beta\epsilon\,{}_1F_1\left(1-\frac{3}{2n};\frac{3}{2};\beta\epsilon\right)\right)+\right.$$
$$\sqrt{\beta\epsilon}\left(2\,\Gamma\left(\frac{3\,(n-1)}{2n}\right){}_1F_1\left(\frac{3\,(n-1)}{2n};\frac{3}{2};\beta\epsilon\right)-\right.$$
$$\left.\left.\Gamma\left(\frac{n-3}{2n}\right){}_1F_1\left(\frac{n-3}{2n};\frac{1}{2};\beta\epsilon\right)\right)\right)N_A$$

The result is that the classical SVC for $(2n-n)$ potentials can be represented by hypergeometric functions. A graphical representation of the SVC in a scaled representation follows:

```
Plot[Evaluate[Bc /. {ϵ → 1, σ → 1, n → 6, N_A → 1}],
  {β, 0., 1 / 2}, AxesLabel → {"βϵ", "B_c / (N_A σ^3)"},
  PlotStyle → RGBColor[0, 0, 0.996109]];
```



The plot shows that the classical SVC possesses a single maximum in the variable $\beta\epsilon$. In addition to the graphical representation of the SVC, the analytical result allows us to apply the result to thermodynamic quantities as given in Eqs. (5.7.111-5.7.115). This opens the way to access thermodynamic quantities like the internal erenrgy. The internal energy for example is defined in terms of the SVC by

$$\frac{\tilde{U}-\tilde{U}^{0}}{RT} = -\left(\frac{B_1}{\tilde{V}} + \ldots\right) \qquad B_n = T^n \frac{d^n B}{\mathrm{dT}^n}, \tag{5.7.118}$$

which becomes

$$\text{InternalEnergy} = -\frac{T\,\frac{\partial\left(\text{Bc}/.\,\beta\to\frac{1}{k_B T}\right)}{\partial T}}{V}$$

$$-\frac{1}{V}\left(T\left(\frac{1}{3}\,2^{\frac{n+3}{n}}\,\pi\,\sigma^3\right.\right.$$
$$\left(-\left(\epsilon\left(2\,\Gamma\left(\frac{3\,(n-1)}{2\,n}\right)\,{}_1F_1\left(\frac{3\,(n-1)}{2\,n};\,\frac{3}{2};\,\frac{\epsilon}{T\,k_B}\right)-\Gamma\left(\frac{n-3}{2\,n}\right)\right.\right.\right.$$
$${}_1F_1\left(\frac{n-3}{2\,n};\,\frac{1}{2};\,\frac{\epsilon}{T\,k_B}\right)\Big)\Big)\Big/\left(2\,T^2\,\sqrt{\frac{\epsilon}{T\,k_B}}\;k_B\right)+$$
$$\Gamma\left(1-\frac{3}{2\,n}\right)\left(\frac{4\,(1-\frac{3}{2\,n})\,{}_1F_1\left(2-\frac{3}{2\,n};\,\frac{5}{2};\,\frac{\epsilon}{T\,k_B}\right)\epsilon^2}{3\,T^3\,k_B^2}+\right.$$
$$\frac{2\,{}_1F_1\left(1-\frac{3}{2\,n};\,\frac{3}{2};\,\frac{\epsilon}{T\,k_B}\right)\epsilon}{T^2\,k_B}-$$
$$\left.\frac{2\,(1-\frac{3}{2\,n})\,{}_1F_1\left(2-\frac{3}{2\,n};\,\frac{3}{2};\,\frac{\epsilon}{T\,k_B}\right)\epsilon}{T^2\,k_B}\right)+$$
$$\left(\frac{(n-3)\,\epsilon\,\Gamma(\frac{n-3}{2\,n})\,{}_1F_1\left(\frac{n-3}{2\,n}+1;\,\frac{3}{2};\,\frac{\epsilon}{T\,k_B}\right)}{n\,T^2\,k_B}-\right.$$
$$\left.\frac{2\,(n-1)\,\epsilon\,\Gamma(\frac{3\,(n-1)}{2\,n})\,{}_1F_1\left(\frac{3\,(n-1)}{2\,n}+1;\,\frac{5}{2};\,\frac{\epsilon}{T\,k_B}\right)}{n\,T^2\,k_B}\right)$$
$$\left.\sqrt{\frac{\epsilon}{T\,k_B}}\right)\left(\frac{\epsilon}{T\,k_B}\right)^{\frac{3}{2n}}N_A-\frac{1}{n\,T^2\,k_B}$$
$$\left(2^{\frac{n+3}{n}-1}\,\pi\,\epsilon\,\sigma^3\left(\sqrt{\frac{\epsilon}{T\,k_B}}\,\left(2\,\Gamma\left(\frac{3\,(n-1)}{2\,n}\right)\,{}_1F_1\left(\frac{3\,(n-1)}{2\,n};\,\frac{3}{2};\right.\right.\right.\right.$$
$$\left.\left.\left.\left.\frac{\epsilon}{T\,k_B}\right)-\Gamma\left(\frac{n-3}{2\,n}\right)\,{}_1F_1\left(\frac{n-3}{2\,n};\,\frac{1}{2};\,\frac{\epsilon}{T\,k_B}\right)\right)+\right.\right.$$
$$\Gamma\left(1-\frac{3}{2\,n}\right)\left(\,{}_1F_1\left(1-\frac{3}{2\,n};\,\frac{1}{2};\,\frac{\epsilon}{T\,k_B}\right)-\right.$$
$$\left.\left.\left.\left.\left.\frac{2\,\epsilon\,{}_1F_1\left(1-\frac{3}{2\,n};\,\frac{3}{2};\,\frac{\epsilon}{T\,k_B}\right)}{T\,k_B}\right)\right)\left(\frac{\epsilon}{T\,k_B}\right)^{\frac{3}{2n}-1}N_A\right)\right)\right)$$

In the above line, we used relation (5.7.111) to represent the internal energy. Since the SVC in our calculations does not depend explicitly on

the temperature T, we replaced the reduced temperature β by $1/(k_B T)$. After this replacement in B_c, we differentiate the resulting expression with respect to T. A multiplication of the result by T and a normalization with the volume V delivers the final result. All of these steps are contained in the above input line. The result is a general analytic expression for the internal energy allowing the choice of the temperature T, the potential depth ϵ, the radius σ, and the exponent of the potential n. To describe a specific gas, we have to insert numeric values for the parameters into the result. For example, we find for $\epsilon = 1$, $n = 6$, $\sigma = 1.$, $N_A = 1$, $k_B = 1$, $T = 200$, and $V = 1$ an internal energy of

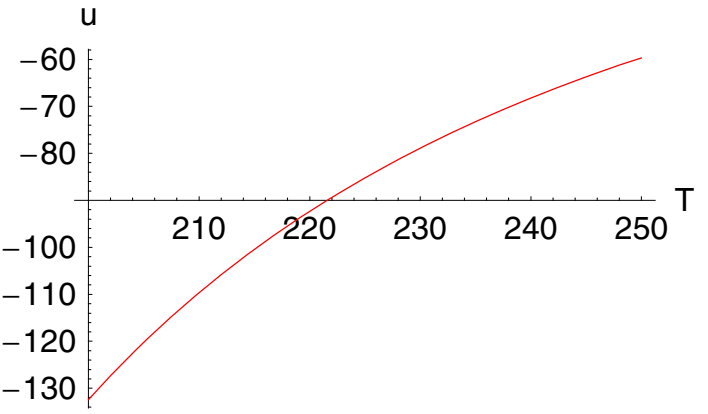
```
<<Miscellaneous`PhysicalConstants`
```

```
InternalEnergy /.
  {ϵ → 10^-20 Joule, n → 6, σ → 10^-8, N_A → AvogadroConstant,
    k_B → BoltzmannConstant, T → 200 Kelvin, V → 1}
```

$$-\frac{132.423}{\text{Mole}}$$

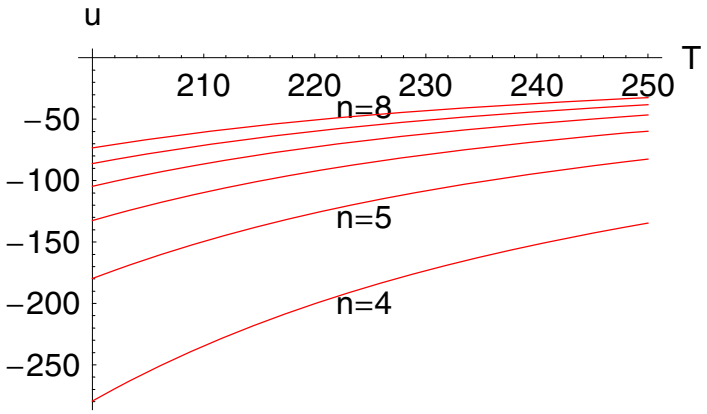
By inserting the model parameters ϵ, n, σ, and the other thermodynamic parameters N_A, k_B, and V, we have access to the numerical values of the internal energy as well. If we vary the temperature T, these values show the dependence of the internal energy on T. If we are interested in the temperature dependence of the internal energy, we can generate a plot by

```
Plot[Evaluate[InternalEnergy Mole /. {ϵ → 10^-20 Joule,
    n → 6, σ → 10^-8, N_A → AvogadroConstant,
    k_B → BoltzmannConstant, T → t Kelvin, V → 1}],
 {t, 200, 250}, AxesLabel → {"T", "u"},
 PlotStyle → RGBColor[0.996109, 0, 0]];
```



If we change, in addition to T, the exponent n in the potential, we get the following figure.

```
Plot[Evaluate[Table[
   InternalEnergy Mole /. {ϵ → 10^-20 Joule, n → ν, σ → 10^-8,
     N_A → AvogadroConstant, k_B → BoltzmannConstant,
     T → t Kelvin, V → 1}, {ν, 4, 9, 1}]],
 {t, 200, 250}, AxesLabel → {"T", "u"},
 PlotStyle → RGBColor[0.996109, 0, 0],
 Prolog → {Text["n=4", {224.474, -199.729}],
   Text["n=8", {224.474, -40}],
   Text["n=5", {224.474, -130}]}];
```



The reader can determine other thermodynamic quantities of his interest, such as enthalpy, entropy, heat capacity at constant pressure, or the Joul–Thomson coefficient.

5.7.3 Quantum Mechanical Corrections $B_{q_1}(T)$ and $B_{q_2}(T)$ of the SVC

Up to the present considerations, we only know the classical behavior of the gas for high temperatures. The following discussion includes two quantum mechanical corrections allowing us to discuss all thermodynamic quantities in cases where quantum corrections are necessary.

The quantum mechanical corrections B_{q1} and B_{q2} in Eq. (5.7.109) and (5.7.110) are realized by the same substitution as demonstrated in the

classical calculation. The integrand of the first quantum correction is transformed by

integrandQc1 =

$$\text{Simplify}\Big[\frac{2\,\pi\,N_A\,\beta^3\,e^{-\beta\,U(r)}\,r^2\,\left(\frac{\partial U(r)}{\partial r}\right)^2\,\text{DifferentialD}(r)}{48\,\pi^2}\ /.\ \text{substitution}\ /.\ \text{DifferentialD}(t)\to 1\Big]$$

$$-\frac{2\,e^{-4\left(t^{\frac{1}{n}}\right)^n\left(\left(t^{\frac{1}{n}}\right)^n-1\right)\beta\epsilon}\,n\,t^{-\frac{n+1}{n}}\left(t^{\frac{1}{n}}\right)^{2n}\left(1-2\left(t^{\frac{1}{n}}\right)^n\right)^2\beta^3\,\epsilon^2\,\sigma\,N_A}{3\,\pi}$$

The related integral follows by inserting the integrand into the integral:

$$\text{Bq1} = \int_0^\infty \text{integrandQc1}\,dt$$

$$\text{If}\Big[\text{Re}(\beta\epsilon)>0,$$

$$-\frac{1}{3\,\pi}\Big(2^{\frac{1}{n}-2}\,n\,\beta^3\,\epsilon^2\,(\beta\epsilon)^{\frac{1}{2}\left(\frac{1}{n}-5\right)}\,\sigma\Big(\Gamma\Big(1-\frac{1}{2\,n}\Big)\,{}_1F_1\Big(1-\frac{1}{2\,n};\frac{1}{2};\beta\epsilon\Big)(\beta\epsilon)^{3/2}+$$

$$\Gamma\Big(2-\frac{1}{2\,n}\Big)\Big({}_1F_1\Big(2-\frac{1}{2\,n};\frac{1}{2};\beta\epsilon\Big)-4\,\beta\epsilon\,{}_1F_1\Big(2-\frac{1}{2\,n};\frac{3}{2};\beta\epsilon\Big)\Big)$$

$$\sqrt{\beta\epsilon}+2\,\beta\epsilon\Big(\Gamma\Big(\frac{3}{2}-\frac{1}{2\,n}\Big)$$

$$\Big(\beta\epsilon\,{}_1F_1\Big(\frac{3}{2}-\frac{1}{2\,n};\frac{3}{2};\beta\epsilon\Big)-{}_1F_1\Big(\frac{3}{2}-\frac{1}{2\,n};\frac{1}{2};\beta\epsilon\Big)\Big)+$$

$$\Gamma\Big(\frac{5}{2}-\frac{1}{2\,n}\Big)\,{}_1F_1\Big(\frac{5}{2}-\frac{1}{2\,n};\frac{3}{2};\beta\epsilon\Big)\Big)\Big)\,N_A\Big),$$

$$\text{Integrate}\Big[-\frac{2\,e^{-4\left(t^{\frac{1}{n}}\right)^n\left(\left(t^{\frac{1}{n}}\right)^n-1\right)\beta\epsilon}\,n\,t^{-\frac{n+1}{n}}\,\beta^3\,\epsilon^2\,\sigma\,N_A\left(t^{\frac{1}{n}}\right)^{2n}}{3\,\pi}+$$

$$\frac{8\,e^{-4\left(t^{\frac{1}{n}}\right)^n\left(\left(t^{\frac{1}{n}}\right)^n-1\right)\beta\epsilon}\,n\,t^{-\frac{n+1}{n}}\,\beta^3\,\epsilon^2\,\sigma\,N_A\left(t^{\frac{1}{n}}\right)^{3n}}{3\,\pi}-$$

$$\frac{8\,e^{-4\left(t^{\frac{1}{n}}\right)^n\left(\left(t^{\frac{1}{n}}\right)^n-1\right)\beta\epsilon}\,n\,t^{-\frac{n+1}{n}}\,\beta^3\,\epsilon^2\,\sigma\,N_A\left(t^{\frac{1}{n}}\right)^{4n}}{3\,\pi},$$

$$\{t,\,0,\,\infty\},\ \text{Assumptions}\to\text{Re}(\beta\epsilon)\le 0\Big]\Big]$$

```
Bq1 = Bq1 /. a_. If[b_, c_, d___] -> a c
```

$$-\frac{1}{3\pi}\left(2^{\frac{1}{n}-2}\, n\, \beta^3\, \epsilon^2\, (\beta\epsilon)^{\frac{1}{2}\left(\frac{1}{n}-5\right)}\, \sigma \right.$$
$$\left(\Gamma\left(1-\frac{1}{2n}\right){}_1F_1\left(1-\frac{1}{2n};\frac{1}{2};\beta\epsilon\right)(\beta\epsilon)^{3/2}+\Gamma\left(2-\frac{1}{2n}\right)\right.$$
$$\left.\left({}_1F_1\left(2-\frac{1}{2n};\frac{1}{2};\beta\epsilon\right)-4\,\beta\epsilon\,{}_1F_1\left(2-\frac{1}{2n};\frac{3}{2};\beta\epsilon\right)\right)\sqrt{\beta\epsilon}+\right.$$
$$2\,\beta\epsilon\left(\Gamma\left(\frac{3}{2}-\frac{1}{2n}\right)\left(\beta\epsilon\,{}_1F_1\left(\frac{3}{2}-\frac{1}{2n};\frac{3}{2};\beta\epsilon\right)-{}_1F_1\left(\frac{3}{2}-\frac{1}{2n};\right.\right.\right.$$
$$\left.\left.\left.\left.\frac{1}{2};\beta\epsilon\right)\right)+\Gamma\left(\frac{5}{2}-\frac{1}{2n}\right){}_1F_1\left(\frac{5}{2}-\frac{1}{2n};\frac{3}{2};\beta\epsilon\right)\right)\right)N_A\Big)$$

Again, we find an analytic representation of the first quantum mechanical correction of the SVC by means of hypergeometric functions ${}_1F_1$. The integrand for the second quantum correction B_{q_2} follows by

```
integrandQc2 = Simplify[
  PowerExpand[-1/(1920 π^4) (2 π N_A β^4 E^(-β U(r)) (-1/36 5 β^2 (∂U(r)/∂r)^4 +
    (10 β (∂U(r)/∂r)^3)/(9 r) + (2 (∂U(r)/∂r)^2)/r + (∂^2 U(r)/∂r ∂r)^2) r^2
    DifferentialD(r)) /. substitution /. DifferentialD(t) → 1]]
```

$$-\frac{1}{540\,\pi^3\,\sigma}\left(e^{-4\,(t-1)\,t\,\beta\epsilon}\, n\, t\, \beta^4\, \epsilon^2 \left(4\,(18\,n^2+(27-10\,\beta\epsilon)\,n+9)\,t^{1+\frac{1}{n}}+\right.\right.$$
$$4\,((5\,\beta^2\,\epsilon^2-36)\,n^2+12\,(5\,\beta\epsilon-3)\,n-9)\,t^{2+\frac{1}{n}}-160\,n\,\beta\epsilon$$
$$(n\,\beta\epsilon+3)\,t^{3+\frac{1}{n}}+160\,n\,\beta\epsilon\,(3\,n\,\beta\epsilon+2)\,t^{4+\frac{1}{n}}-640\,n^2\,\beta^2\,\epsilon^2\,t^{5+\frac{1}{n}}+$$
$$\left.320\,n^2\,\beta^2\,\epsilon^2\,t^{6+\frac{1}{n}}-9\,(n+1)^2\,t^{\frac{1}{n}}-72\,\sigma\,t^2+72\,\sigma\,t-18\,\sigma\right)N_A\Big)$$

and the explicit integration provides

$$\mathbf{Bq2} = \int_0^\infty \mathbf{integrandQc2}\, dt$$

Integrate::gener : Unable to check convergence. More…

$$-\frac{1}{540\,\pi^3\,\sigma}\left(n\,\beta^4\,\epsilon^2\,\text{If}\left[\text{Re}\left(\frac{1}{n}\right) > -2 \bigwedge \text{Re}(\beta\,\epsilon) > 0,\right.\right.$$

$$-\frac{5\,2^{-1-\frac{1}{n}}\,n\,\Gamma\left(\frac{1}{2}\left(3+\frac{1}{n}\right)\right)\,{}_1F_1\left(\frac{3}{2}+\frac{1}{2n};\,\frac{1}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{\frac{1}{2}-\frac{1}{2n}}}{\beta\,\epsilon} +$$

$$\frac{9\,2^{-2-\frac{1}{n}}\,\Gamma\left(\frac{1}{2}\left(3+\frac{1}{n}\right)\right)\,{}_1F_1\left(\frac{3}{2}+\frac{1}{2n};\,\frac{1}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{\frac{1}{2}-\frac{1}{2n}}}{\beta^2\,\epsilon^2} +$$

$$\frac{9\,2^{-1-\frac{1}{n}}\,n^2\,\Gamma\left(\frac{1}{2}\left(3+\frac{1}{n}\right)\right)\,{}_1F_1\left(\frac{3}{2}+\frac{1}{2n};\,\frac{1}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{\frac{1}{2}-\frac{1}{2n}}}{\beta^2\,\epsilon^2} +$$

$$\frac{27\,2^{-2-\frac{1}{n}}\,n\,\Gamma\left(\frac{1}{2}\left(3+\frac{1}{n}\right)\right)\,{}_1F_1\left(\frac{3}{2}+\frac{1}{2n};\,\frac{1}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{\frac{1}{2}-\frac{1}{2n}}}{\beta^2\,\epsilon^2} -$$

$$\frac{9\,2^{-2-\frac{1}{n}}\,\Gamma\left(\frac{1}{2}\left(3+\frac{1}{n}\right)\right)\,{}_1F_1\left(\frac{3}{2}+\frac{1}{2n};\,\frac{3}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{\frac{1}{2}-\frac{1}{2n}}}{\beta\,\epsilon} -$$

$$\frac{9\,2^{-2-\frac{1}{n}}\,n^2\,\Gamma\left(\frac{1}{2}\left(3+\frac{1}{n}\right)\right)\,{}_1F_1\left(\frac{3}{2}+\frac{1}{2n};\,\frac{3}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{\frac{1}{2}-\frac{1}{2n}}}{\beta\,\epsilon} -$$

$$\frac{9\,2^{-1-\frac{1}{n}}\,n\,\Gamma\left(\frac{1}{2}\left(3+\frac{1}{n}\right)\right)\,{}_1F_1\left(\frac{3}{2}+\frac{1}{2n};\,\frac{3}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{\frac{1}{2}-\frac{1}{2n}}}{\beta\,\epsilon} +$$

$$5\,2^{-2-\frac{1}{n}}\,n^2\,\Gamma\left(\frac{1}{2}\left(5+\frac{1}{n}\right)\right)\,{}_1F_1\left(\frac{5}{2}+\frac{1}{2n};\,\frac{3}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{\frac{1}{2}-\frac{1}{2n}} -$$

$$\frac{5\,2^{-1-\frac{1}{n}}\,n^2\,\Gamma\left(\frac{1}{2}\left(5+\frac{1}{n}\right)\right)\,{}_1F_1\left(\frac{5}{2}+\frac{1}{2n};\,\frac{1}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{\frac{3}{2}-\frac{1}{2n}}}{\beta^2\,\epsilon^2} -$$

$$\frac{15\,2^{-1-\frac{1}{n}}\,n\,\Gamma\left(\frac{1}{2}\left(5+\frac{1}{n}\right)\right)\,{}_1F_1\left(\frac{5}{2}+\frac{1}{2n};\,\frac{1}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{\frac{3}{2}-\frac{1}{2n}}}{\beta^3\,\epsilon^3} +$$

$$\frac{15\,2^{-1/n}\,n\,\Gamma\left(\frac{1}{2}\left(5+\frac{1}{n}\right)\right)\,{}_1F_1\left(\frac{5}{2}+\frac{1}{2n};\,\frac{3}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{\frac{3}{2}-\frac{1}{2n}}}{\beta^2\,\epsilon^2} -$$

$$\frac{9\,2^{-2-\frac{1}{n}}\,\Gamma\left(\frac{1}{2}\left(5+\frac{1}{n}\right)\right)\,{}_1F_1\left(\frac{5}{2}+\frac{1}{2n};\,\frac{3}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{\frac{3}{2}-\frac{1}{2n}}}{\beta^3\,\epsilon^3} -$$

$$\frac{9\,2^{-1/n}\,n^2\,\Gamma\left(\frac{1}{2}\left(5+\frac{1}{n}\right)\right)\,{}_1F_1\left(\frac{5}{2}+\frac{1}{2n};\,\frac{3}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{\frac{3}{2}-\frac{1}{2n}}}{\beta^3\,\epsilon^3} -$$

$$\frac{9\,2^{-1/n}\,n\,\Gamma(\frac{1}{2}\,(5+\frac{1}{n}))\,{}_1F_1(\frac{5}{2}+\frac{1}{2n};\,\frac{3}{2};\,\beta\,\epsilon)\,(\beta\,\epsilon)^{\frac{3}{2}-\frac{1}{2n}}}{\beta^3\,\epsilon^3}-$$

$$\frac{5\,2^{-1-\frac{1}{n}}\,n^2\,\Gamma(\frac{1}{2}\,(7+\frac{1}{n}))\,{}_1F_1(\frac{7}{2}+\frac{1}{2n};\,\frac{1}{2};\,\beta\,\epsilon)\,(\beta\,\epsilon)^{\frac{5}{2}-\frac{1}{2n}}}{\beta^4\,\epsilon^4}+$$

$$\frac{15\,2^{-1-\frac{1}{n}}\,n^2\,\Gamma(\frac{1}{2}\,(7+\frac{1}{n}))\,{}_1F_1(\frac{7}{2}+\frac{1}{2n};\,\frac{3}{2};\,\beta\,\epsilon)\,(\beta\,\epsilon)^{\frac{5}{2}-\frac{1}{2n}}}{\beta^3\,\epsilon^3}+$$

$$\frac{5\,2^{-1/n}\,n\,\Gamma(\frac{1}{2}\,(7+\frac{1}{n}))\,{}_1F_1(\frac{7}{2}+\frac{1}{2n};\,\frac{3}{2};\,\beta\,\epsilon)\,(\beta\,\epsilon)^{\frac{5}{2}-\frac{1}{2n}}}{\beta^4\,\epsilon^4}+$$

$$\frac{5\,2^{-2-\frac{1}{n}}\,n^2\,\Gamma(\frac{1}{2}\,(9+\frac{1}{n}))\,{}_1F_1(\frac{9}{2}+\frac{1}{2n};\,\frac{3}{2};\,\beta\,\epsilon)\,(\beta\,\epsilon)^{\frac{7}{2}-\frac{1}{2n}}}{\beta^5\,\epsilon^5}-$$

$$\frac{9\,2^{-3-\frac{1}{n}}\,\Gamma(1+\frac{1}{2n})\,{}_1F_1(1+\frac{1}{2n};\,\frac{1}{2};\,\beta\,\epsilon)\,(\beta\,\epsilon)^{-\frac{1}{2n}}}{\beta\,\epsilon}-$$

$$\frac{9\,2^{-3-\frac{1}{n}}\,\Gamma(\frac{1}{2n})\,{}_1F_1(1+\frac{1}{2n};\,\frac{1}{2};\,\beta\,\epsilon)\,(\beta\,\epsilon)^{-\frac{1}{2n}}}{\beta\,\epsilon}-$$

$$\frac{9\,2^{-4-\frac{1}{n}}\,n\,\Gamma(\frac{1}{2n})\,{}_1F_1(1+\frac{1}{2n};\,\frac{1}{2};\,\beta\,\epsilon)\,(\beta\,\epsilon)^{-\frac{1}{2n}}}{\beta\,\epsilon}+$$

$$5\,2^{-3-\frac{1}{n}}\,n^2\,\Gamma\Big(2+\frac{1}{2n}\Big)\,{}_1F_1\Big(2+\frac{1}{2n};\,\frac{1}{2};\,\beta\,\epsilon\Big)\,(\beta\,\epsilon)^{-\frac{1}{2n}}+$$

$$\frac{15\,2^{-1-\frac{1}{n}}\,n\,\Gamma(2+\frac{1}{2n})\,{}_1F_1(2+\frac{1}{2n};\,\frac{1}{2};\,\beta\,\epsilon)\,(\beta\,\epsilon)^{-\frac{1}{2n}}}{\beta\,\epsilon}-$$

$$\frac{9\,2^{-3-\frac{1}{n}}\,\Gamma(2+\frac{1}{2n})\,{}_1F_1(2+\frac{1}{2n};\,\frac{1}{2};\,\beta\,\epsilon)\,(\beta\,\epsilon)^{-\frac{1}{2n}}}{\beta^2\,\epsilon^2}-$$

$$\frac{9\,2^{-1-\frac{1}{n}}\,n^2\,\Gamma(2+\frac{1}{2n})\,{}_1F_1(2+\frac{1}{2n};\,\frac{1}{2};\,\beta\,\epsilon)\,(\beta\,\epsilon)^{-\frac{1}{2n}}}{\beta^2\,\epsilon^2}-$$

$$\frac{9\,2^{-1-\frac{1}{n}}\,n\,\Gamma(2+\frac{1}{2n})\,{}_1F_1(2+\frac{1}{2n};\,\frac{1}{2};\,\beta\,\epsilon)\,(\beta\,\epsilon)^{-\frac{1}{2n}}}{\beta^2\,\epsilon^2}-$$

$$5\,2^{-1/n}\,n\,\Gamma\Big(2+\frac{1}{2n}\Big)\,{}_1F_1\Big(2+\frac{1}{2n};\,\frac{3}{2};\,\beta\,\epsilon\Big)\,(\beta\,\epsilon)^{-\frac{1}{2n}}+$$

$$\frac{9\,2^{-1-\frac{1}{n}}\,\Gamma(2+\frac{1}{2n})\,{}_1F_1(2+\frac{1}{2n};\,\frac{3}{2};\,\beta\,\epsilon)\,(\beta\,\epsilon)^{-\frac{1}{2n}}}{\beta\,\epsilon}+$$

$$\frac{9\,2^{-1/n}\,n^2\,\Gamma(2+\frac{1}{2n})\,{}_1F_1(2+\frac{1}{2n};\,\frac{3}{2};\,\beta\,\epsilon)\,(\beta\,\epsilon)^{-\frac{1}{2n}}}{\beta\,\epsilon}+$$

$$\frac{27\,2^{-1-\frac{1}{n}}\,n\,\Gamma(2+\frac{1}{2n})\,{}_1F_1(2+\frac{1}{2n};\,\frac{3}{2};\,\beta\,\epsilon)\,(\beta\,\epsilon)^{-\frac{1}{2n}}}{\beta\,\epsilon}+$$

$$\frac{15\,2^{-2-\frac{1}{n}}\,n^2\,\Gamma(3+\frac{1}{2n})\,{}_1F_1(3+\frac{1}{2n};\,\frac{1}{2};\,\beta\epsilon)\,(\beta\epsilon)^{-\frac{1}{2n}}}{\beta\epsilon}+$$

$$\frac{5\,2^{-1-\frac{1}{n}}\,n\,\Gamma(3+\frac{1}{2n})\,{}_1F_1(3+\frac{1}{2n};\,\frac{1}{2};\,\beta\epsilon)\,(\beta\epsilon)^{-\frac{1}{2n}}}{\beta^2\,\epsilon^2}-$$

$$5\,2^{-1/n}\,n^2\,\Gamma\Big(3+\frac{1}{2n}\Big)\,{}_1F_1\Big(3+\frac{1}{2n};\,\frac{3}{2};\,\beta\epsilon\Big)\,(\beta\epsilon)^{-\frac{1}{2n}}-$$

$$\frac{15\,2^{-1/n}\,n\,\Gamma(3+\frac{1}{2n})\,{}_1F_1(3+\frac{1}{2n};\,\frac{3}{2};\,\beta\epsilon)\,(\beta\epsilon)^{-\frac{1}{2n}}}{\beta\epsilon}+$$

$$\frac{5\,2^{-3-\frac{1}{n}}\,n^2\,\Gamma(4+\frac{1}{2n})\,{}_1F_1(4+\frac{1}{2n};\,\frac{1}{2};\,\beta\epsilon)\,(\beta\epsilon)^{-\frac{1}{2n}}}{\beta^2\,\epsilon^2}-$$

$$\frac{5\,2^{-1/n}\,n^2\,\Gamma(4+\frac{1}{2n})\,{}_1F_1(4+\frac{1}{2n};\,\frac{3}{2};\,\beta\epsilon)\,(\beta\epsilon)^{-\frac{1}{2n}}}{\beta\epsilon}-$$

$$\frac{27\,e^{\beta\epsilon}\,\sqrt{\pi}\,\left(\frac{2\,\beta\epsilon}{3}+1\right)\sigma\,(\beta\epsilon)^{5/2}}{8\,\beta^4\,\epsilon^4}+$$

$$\frac{9\,e^{\beta\epsilon}\,\sqrt{\pi}\,(2\,\beta\epsilon+1)\,\sigma\,(\beta\epsilon)^{3/2}}{4\,\beta^3\,\epsilon^3}-\frac{9\,e^{\beta\epsilon}\,\sqrt{\pi}\,\sigma\,(\beta\epsilon)^{3/2}}{4\,\beta^2\,\epsilon^2}+$$

$$\frac{9\,e^{\beta\epsilon}\,\sigma\left(2\,\sqrt{\pi}\,\left(1-\text{erf}\left(\sqrt{\beta\epsilon}\right)\right)-\frac{2\,e^{-\beta\epsilon}}{\sqrt{\beta\epsilon}}-2\,\sqrt{\pi}\right)\sqrt{\beta\epsilon}}{8\,\beta\epsilon}+$$

$$\frac{1}{2\,\beta\epsilon}\left(9\,\sigma\left(e^{\beta\epsilon}\,\sqrt{\beta\epsilon}\,\left(\sqrt{\pi}-\sqrt{\pi}\,\left(1-\text{erf}\left(\sqrt{\beta\epsilon}\right)\right)\right)+\right.\right.$$

$$\left.\left.\frac{e^{\beta\epsilon}\left(\sqrt{\pi}-\sqrt{\pi}\,\left(1-\text{erf}\left(\sqrt{\beta\epsilon}\right)\right)\right)}{2\,\sqrt{\beta\epsilon}}+1\right)\right)-$$

$$\frac{1}{8\,\beta^2\,\epsilon^2}\left(9\,\sigma\left(-e^{\beta\epsilon}\left(2\,\sqrt{\pi}\,\left(1-\text{erf}\left(\sqrt{\beta\epsilon}\right)\right)-\frac{2\,e^{-\beta\epsilon}}{\sqrt{\beta\epsilon}}-2\,\sqrt{\pi}\right)\right.\right.$$

$$(\beta\epsilon)^{3/2}-\frac{3}{2}\,e^{\beta\epsilon}$$

$$\left.\left.\left(2\,\sqrt{\pi}\,\left(1-\text{erf}\left(\sqrt{\beta\epsilon}\right)\right)-\frac{2\,e^{-\beta\epsilon}}{\sqrt{\beta\epsilon}}-2\,\sqrt{\pi}\right)\sqrt{\beta\epsilon}-1\right)\right),$$

$$\text{Integrate}\Big[e^{-4\,(t-1)\,t\,\beta\epsilon}\,t\,\big(4\,(18\,n^2+(27-10\,\beta\epsilon)\,n+9)\,t^{1+\frac{1}{n}}+$$

$$4\,((5\,\beta^2\,\epsilon^2-36)\,n^2+12\,(5\,\beta\epsilon-3)\,n-9)\,t^{2+\frac{1}{n}}-$$

$$160\,n\,\beta\epsilon\,(n\,\beta\epsilon+3)\,t^{3+\frac{1}{n}}+160\,n\,\beta\epsilon\,(3\,n\,\beta\epsilon+2)\,t^{4+\frac{1}{n}}-$$

$$640\,n^2\,\beta^2\,\epsilon^2\,t^{5+\frac{1}{n}}+320\,n^2\,\beta^2\,\epsilon^2\,t^{6+\frac{1}{n}}-$$

$$9\,(n+1)^2\,t^{\frac{1}{n}}-72\,\sigma\,t^2+72\,\sigma\,t-18\,\sigma\big),\,\{t,\,0,\,\infty\},$$

$$\text{Assumptions}\to\neg\left(\text{Re}\left(\frac{1}{n}\right)>-2\bigwedge\text{Re}(\beta\epsilon)>0\right)\Big]\Big]\,N_A\Bigg)$$

Bq2 = Bq2 /. a_. If[b_, c_, d___] → *a c*

$$-\frac{1}{540\,\pi^3\,\sigma}$$

$$\left(n\,\beta^4\,\epsilon^2\left(-\frac{5\,2^{-1-\frac{1}{n}}\,n\,\Gamma\left(\frac{1}{2}\left(3+\frac{1}{n}\right)\right)\,{}_1F_1\left(\frac{3}{2}+\frac{1}{2n};\,\frac{1}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{\frac{1}{2}-\frac{1}{2n}}}{\beta\,\epsilon}+\right.\right.$$

$$\frac{9\,2^{-2-\frac{1}{n}}\,\Gamma\left(\frac{1}{2}\left(3+\frac{1}{n}\right)\right)\,{}_1F_1\left(\frac{3}{2}+\frac{1}{2n};\,\frac{1}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{\frac{1}{2}-\frac{1}{2n}}}{\beta^2\,\epsilon^2}+$$

$$\frac{9\,2^{-1-\frac{1}{n}}\,n^2\,\Gamma\left(\frac{1}{2}\left(3+\frac{1}{n}\right)\right)\,{}_1F_1\left(\frac{3}{2}+\frac{1}{2n};\,\frac{1}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{\frac{1}{2}-\frac{1}{2n}}}{\beta^2\,\epsilon^2}+$$

$$\frac{27\,2^{-2-\frac{1}{n}}\,n\,\Gamma\left(\frac{1}{2}\left(3+\frac{1}{n}\right)\right)\,{}_1F_1\left(\frac{3}{2}+\frac{1}{2n};\,\frac{1}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{\frac{1}{2}-\frac{1}{2n}}}{\beta^2\,\epsilon^2}-$$

$$\frac{9\,2^{-2-\frac{1}{n}}\,\Gamma\left(\frac{1}{2}\left(3+\frac{1}{n}\right)\right)\,{}_1F_1\left(\frac{3}{2}+\frac{1}{2n};\,\frac{3}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{\frac{1}{2}-\frac{1}{2n}}}{\beta\,\epsilon}-$$

$$\frac{9\,2^{-2-\frac{1}{n}}\,n^2\,\Gamma\left(\frac{1}{2}\left(3+\frac{1}{n}\right)\right)\,{}_1F_1\left(\frac{3}{2}+\frac{1}{2n};\,\frac{3}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{\frac{1}{2}-\frac{1}{2n}}}{\beta\,\epsilon}-$$

$$\frac{9\,2^{-1-\frac{1}{n}}\,n\,\Gamma\left(\frac{1}{2}\left(3+\frac{1}{n}\right)\right)\,{}_1F_1\left(\frac{3}{2}+\frac{1}{2n};\,\frac{3}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{\frac{1}{2}-\frac{1}{2n}}}{\beta\,\epsilon}+$$

$$5\,2^{-2-\frac{1}{n}}\,n^2\,\Gamma\left(\frac{1}{2}\left(5+\frac{1}{n}\right)\right)\,{}_1F_1\left(\frac{5}{2}+\frac{1}{2n};\,\frac{3}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{\frac{1}{2}-\frac{1}{2n}}-$$

$$\frac{5\,2^{-1-\frac{1}{n}}\,n^2\,\Gamma\left(\frac{1}{2}\left(5+\frac{1}{n}\right)\right)\,{}_1F_1\left(\frac{5}{2}+\frac{1}{2n};\,\frac{1}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{\frac{3}{2}-\frac{1}{2n}}}{\beta^2\,\epsilon^2}-$$

$$\frac{15\,2^{-1-\frac{1}{n}}\,n\,\Gamma\left(\frac{1}{2}\left(5+\frac{1}{n}\right)\right)\,{}_1F_1\left(\frac{5}{2}+\frac{1}{2n};\,\frac{1}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{\frac{3}{2}-\frac{1}{2n}}}{\beta^3\,\epsilon^3}+$$

$$\frac{15\,2^{-1/n}\,n\,\Gamma\left(\frac{1}{2}\left(5+\frac{1}{n}\right)\right)\,{}_1F_1\left(\frac{5}{2}+\frac{1}{2n};\,\frac{3}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{\frac{3}{2}-\frac{1}{2n}}}{\beta^2\,\epsilon^2}-$$

$$\frac{9\,2^{-2-\frac{1}{n}}\,\Gamma\left(\frac{1}{2}\left(5+\frac{1}{n}\right)\right)\,{}_1F_1\left(\frac{5}{2}+\frac{1}{2n};\,\frac{3}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{\frac{3}{2}-\frac{1}{2n}}}{\beta^3\,\epsilon^3}-$$

$$\frac{9\,2^{-1/n}\,n^2\,\Gamma\left(\frac{1}{2}\left(5+\frac{1}{n}\right)\right)\,{}_1F_1\left(\frac{5}{2}+\frac{1}{2n};\,\frac{3}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{\frac{3}{2}-\frac{1}{2n}}}{\beta^3\,\epsilon^3}-$$

$$\left.\left.\frac{9\,2^{-1/n}\,n\,\Gamma\left(\frac{1}{2}\left(5+\frac{1}{n}\right)\right)\,{}_1F_1\left(\frac{5}{2}+\frac{1}{2n};\,\frac{3}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{\frac{3}{2}-\frac{1}{2n}}}{\beta^3\,\epsilon^3}-\right.\right.$$

$$\frac{5\,2^{-1-\frac{1}{n}}\,n^2\,\Gamma\left(\frac{1}{2}\,(7+\frac{1}{n})\right)\,{}_1F_1\left(\frac{7}{2}+\frac{1}{2n};\,\frac{1}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{\frac{5}{2}-\frac{1}{2n}}}{\beta^4\,\epsilon^4}+$$

$$\frac{15\,2^{-1-\frac{1}{n}}\,n^2\,\Gamma\left(\frac{1}{2}\,(7+\frac{1}{n})\right)\,{}_1F_1\left(\frac{7}{2}+\frac{1}{2n};\,\frac{3}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{\frac{5}{2}-\frac{1}{2n}}}{\beta^3\,\epsilon^3}+$$

$$\frac{5\,2^{-1/n}\,n\,\Gamma\left(\frac{1}{2}\,(7+\frac{1}{n})\right)\,{}_1F_1\left(\frac{7}{2}+\frac{1}{2n};\,\frac{3}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{\frac{5}{2}-\frac{1}{2n}}}{\beta^4\,\epsilon^4}+$$

$$\frac{5\,2^{-2-\frac{1}{n}}\,n^2\,\Gamma\left(\frac{1}{2}\,(9+\frac{1}{n})\right)\,{}_1F_1\left(\frac{9}{2}+\frac{1}{2n};\,\frac{3}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{\frac{7}{2}-\frac{1}{2n}}}{\beta^5\,\epsilon^5}-$$

$$\frac{9\,2^{-3-\frac{1}{n}}\,\Gamma\left(1+\frac{1}{2n}\right)\,{}_1F_1\left(1+\frac{1}{2n};\,\frac{1}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{-\frac{1}{2n}}}{\beta\,\epsilon}-$$

$$\frac{9\,2^{-3-\frac{1}{n}}\,\Gamma\left(\frac{1}{2n}\right)\,{}_1F_1\left(1+\frac{1}{2n};\,\frac{1}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{-\frac{1}{2n}}}{\beta\,\epsilon}-$$

$$\frac{9\,2^{-4-\frac{1}{n}}\,n\,\Gamma\left(\frac{1}{2n}\right)\,{}_1F_1\left(1+\frac{1}{2n};\,\frac{1}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{-\frac{1}{2n}}}{\beta\,\epsilon}+$$

$$5\,2^{-3-\frac{1}{n}}\,n^2\,\Gamma\left(2+\frac{1}{2n}\right)\,{}_1F_1\left(2+\frac{1}{2n};\,\frac{1}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{-\frac{1}{2n}}+$$

$$\frac{15\,2^{-1-\frac{1}{n}}\,n\,\Gamma\left(2+\frac{1}{2n}\right)\,{}_1F_1\left(2+\frac{1}{2n};\,\frac{1}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{-\frac{1}{2n}}}{\beta\,\epsilon}-$$

$$\frac{9\,2^{-3-\frac{1}{n}}\,\Gamma\left(2+\frac{1}{2n}\right)\,{}_1F_1\left(2+\frac{1}{2n};\,\frac{1}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{-\frac{1}{2n}}}{\beta^2\,\epsilon^2}-$$

$$\frac{9\,2^{-1-\frac{1}{n}}\,n^2\,\Gamma\left(2+\frac{1}{2n}\right)\,{}_1F_1\left(2+\frac{1}{2n};\,\frac{1}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{-\frac{1}{2n}}}{\beta^2\,\epsilon^2}-$$

$$\frac{9\,2^{-1-\frac{1}{n}}\,n\,\Gamma\left(2+\frac{1}{2n}\right)\,{}_1F_1\left(2+\frac{1}{2n};\,\frac{1}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{-\frac{1}{2n}}}{\beta^2\,\epsilon^2}-$$

$$5\,2^{-1/n}\,n\,\Gamma\left(2+\frac{1}{2n}\right)\,{}_1F_1\left(2+\frac{1}{2n};\,\frac{3}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{-\frac{1}{2n}}+$$

$$\frac{9\,2^{-1-\frac{1}{n}}\,\Gamma\left(2+\frac{1}{2n}\right)\,{}_1F_1\left(2+\frac{1}{2n};\,\frac{3}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{-\frac{1}{2n}}}{\beta\,\epsilon}+$$

$$\frac{9\,2^{-1/n}\,n^2\,\Gamma\left(2+\frac{1}{2n}\right)\,{}_1F_1\left(2+\frac{1}{2n};\,\frac{3}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{-\frac{1}{2n}}}{\beta\,\epsilon}+$$

$$\frac{27\,2^{-1-\frac{1}{n}}\,n\,\Gamma\left(2+\frac{1}{2n}\right)\,{}_1F_1\left(2+\frac{1}{2n};\,\frac{3}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{-\frac{1}{2n}}}{\beta\,\epsilon}+$$

$$\frac{15\,2^{-2-\frac{1}{n}}\,n^2\,\Gamma\left(3+\frac{1}{2n}\right)\,{}_1F_1\left(3+\frac{1}{2n};\,\frac{1}{2};\,\beta\,\epsilon\right)(\beta\,\epsilon)^{-\frac{1}{2n}}}{\beta\,\epsilon}+$$

$$\frac{5\,2^{-1-\frac{1}{n}}\,n\,\Gamma(3+\frac{1}{2n})\,{}_1F_1(3+\frac{1}{2n};\,\frac{1}{2};\,\beta\epsilon)\,(\beta\epsilon)^{-\frac{1}{2n}}}{\beta^2\,\epsilon^2} -$$

$$5\,2^{-1/n}\,n^2\,\Gamma\Big(3+\frac{1}{2n}\Big)\,{}_1F_1\Big(3+\frac{1}{2n};\,\frac{3}{2};\,\beta\epsilon\Big)\,(\beta\epsilon)^{-\frac{1}{2n}} -$$

$$\frac{15\,2^{-1/n}\,n\,\Gamma(3+\frac{1}{2n})\,{}_1F_1(3+\frac{1}{2n};\,\frac{3}{2};\,\beta\epsilon)\,(\beta\epsilon)^{-\frac{1}{2n}}}{\beta\epsilon} +$$

$$\frac{5\,2^{-3-\frac{1}{n}}\,n^2\,\Gamma(4+\frac{1}{2n})\,{}_1F_1(4+\frac{1}{2n};\,\frac{1}{2};\,\beta\epsilon)\,(\beta\epsilon)^{-\frac{1}{2n}}}{\beta^2\,\epsilon^2} -$$

$$\frac{5\,2^{-1/n}\,n^2\,\Gamma(4+\frac{1}{2n})\,{}_1F_1(4+\frac{1}{2n};\,\frac{3}{2};\,\beta\epsilon)\,(\beta\epsilon)^{-\frac{1}{2n}}}{\beta\epsilon} -$$

$$\frac{27\,e^{\beta\epsilon}\,\sqrt{\pi}\,\left(\frac{2\beta\epsilon}{3}+1\right)\sigma\,(\beta\epsilon)^{5/2}}{8\,\beta^4\,\epsilon^4} +$$

$$\frac{9\,e^{\beta\epsilon}\,\sqrt{\pi}\,(2\,\beta\epsilon+1)\,\sigma\,(\beta\epsilon)^{3/2}}{4\,\beta^3\,\epsilon^3} - \frac{9\,e^{\beta\epsilon}\,\sqrt{\pi}\,\sigma\,(\beta\epsilon)^{3/2}}{4\,\beta^2\,\epsilon^2} +$$

$$\frac{9\,e^{\beta\epsilon}\,\sigma\left(2\,\sqrt{\pi}\,\left(1-\text{erf}\left(\sqrt{\beta\epsilon}\right)\right)-\frac{2\,e^{-\beta\epsilon}}{\sqrt{\beta\epsilon}}-2\,\sqrt{\pi}\right)\sqrt{\beta\epsilon}}{8\,\beta\epsilon} +$$

$$\frac{1}{2\,\beta\epsilon}\left(9\,\sigma\left(e^{\beta\epsilon}\,\sqrt{\beta\epsilon}\,\left(\sqrt{\pi}-\sqrt{\pi}\,\left(1-\text{erf}\left(\sqrt{\beta\epsilon}\right)\right)\right)+\right.\right.$$

$$\left.\frac{e^{\beta\epsilon}\left(\sqrt{\pi}-\sqrt{\pi}\,\left(1-\text{erf}\left(\sqrt{\beta\epsilon}\right)\right)\right)}{2\,\sqrt{\beta\epsilon}}+1\right)\Big)-\frac{1}{8\,\beta^2\,\epsilon^2}$$

$$\left(9\,\sigma\left(-e^{\beta\epsilon}\left(2\,\sqrt{\pi}\,\left(1-\text{erf}\left(\sqrt{\beta\epsilon}\right)\right)-\frac{2\,e^{-\beta\epsilon}}{\sqrt{\beta\epsilon}}-2\,\sqrt{\pi}\right)(\beta\epsilon)^{3/2}-\right.\right.$$

$$\frac{3}{2}\,e^{\beta\epsilon}\left(2\,\sqrt{\pi}\,\left(1-\text{erf}\left(\sqrt{\beta\epsilon}\right)\right)-\frac{2\,e^{-\beta\epsilon}}{\sqrt{\beta\epsilon}}-2\,\sqrt{\pi}\right)\sqrt{\beta\epsilon}-$$

$$\left.\left.1\right)\right)\Big)N_A\Big)$$

Using ϵ/k_B and $\frac{2}{3}\,\pi\,N_A\,\sigma^3$ in a scaling transformation for the temperature in the SVC, we get $B^*(T)$ tbulated in books like Hirschfelder et al [5.10]. These authors introduce a scaled representation of the SVC by $B_c^* = B_c\,/\,(\frac{2}{3}\,\pi\,N_A\,\sigma^3)$ with a reduced temperature of $T^* = k_B\,T\,/\,\epsilon$.

$$\text{BStar} = \text{Simplify}\left[\frac{\text{Bq2}\left(\frac{\hbar^2}{m}\right)^2 + \frac{\text{Bq1}\,\hbar^2}{m} + \text{Bc}}{\frac{2}{3}\pi N_A \sigma^3} \text{ //. } \left\{\epsilon \to \frac{1}{\beta T}, \hbar \to \Lambda\sigma\sqrt{m\epsilon}\right\}\right]$$

$$\frac{1}{45\pi^4}\left(2^{-7-\frac{1}{n}}\left(\frac{1}{T}\right)^{4-\frac{1}{2n}}\right.$$

$$\left(9\,2^{2+\frac{1}{n}}\,n\,\Lambda^4\,\sigma\left(\frac{1}{T}\right)^{\frac{1}{2n}-2} + 9\,2^{1+\frac{1}{n}}\,e^{\frac{1}{T}}\,n\sqrt{\pi}\,\Lambda^4\,\sigma\left(\frac{1}{T}\right)^{\frac{1}{2}\left(\frac{1}{n}-3\right)} +\right.$$

$$9\,2^{1+\frac{1}{n}}\,e^{\frac{1}{T}}\,n\sqrt{\pi}\,\Lambda^4\,\sigma\,\text{erf}\left(\sqrt{\frac{1}{T}}\right)\left(\frac{1}{T}\right)^{\frac{1}{2}\left(\frac{1}{n}-3\right)} -$$

$$45\,2^{4+\frac{2}{n}}\,n\pi^2\,\Lambda^2\,\Gamma\left(2-\frac{1}{2n}\right){}_1F_1\left(2-\frac{1}{2n};\frac{1}{2};\frac{1}{T}\right)\left(\frac{1}{T}\right)^{\frac{1}{n}-3} +$$

$$45\,2^{5+\frac{2}{n}}\,n\pi^2\,\Lambda^2\,\Gamma\left(\frac{3}{2}-\frac{1}{2n}\right){}_1F_1\left(\frac{3}{2}-\frac{1}{2n};\frac{1}{2};\frac{1}{T}\right)\left(\frac{1}{T}\right)^{\frac{1}{n}-\frac{5}{2}} -$$

$$45\,2^{5+\frac{2}{n}}\,n\pi^2\,\Lambda^2\,\Gamma\left(\frac{5}{2}-\frac{1}{2n}\right){}_1F_1\left(\frac{5}{2}-\frac{1}{2n};\frac{3}{2};\frac{1}{T}\right)\left(\frac{1}{T}\right)^{\frac{1}{n}-\frac{5}{2}} -$$

$$45\,2^{4+\frac{2}{n}}\,n\pi^2\,\Lambda^2\,\Gamma\left(1-\frac{1}{2n}\right){}_1F_1\left(1-\frac{1}{2n};\frac{1}{2};\frac{1}{T}\right)\left(\frac{1}{T}\right)^{\frac{1}{n}-2} +$$

$$45\,2^{6+\frac{2}{n}}\,n\pi^2\,\Lambda^2\,\Gamma\left(2-\frac{1}{2n}\right){}_1F_1\left(2-\frac{1}{2n};\frac{3}{2};\frac{1}{T}\right)\left(\frac{1}{T}\right)^{\frac{1}{n}-2} -$$

$$45\,2^{5+\frac{2}{n}}\,n\pi^2\,\Lambda^2\,\Gamma\left(\frac{3}{2}-\frac{1}{2n}\right){}_1F_1\left(\frac{3}{2}-\frac{1}{2n};\frac{3}{2};\frac{1}{T}\right)\left(\frac{1}{T}\right)^{\frac{1}{n}-\frac{3}{2}} -$$

$$45\,2^{7+\frac{4}{n}}\,\pi^4\,\Gamma\left(\frac{n-3}{2n}\right){}_1F_1\left(\frac{n-3}{2n};\frac{1}{2};\frac{1}{T}\right)\left(\frac{1}{T}\right)^{\frac{2}{n}-\frac{7}{2}} +$$

$$45\,2^{8+\frac{4}{n}}\,\pi^4\,\Gamma\left(\frac{3(n-1)}{2n}\right){}_1F_1\left(\frac{3(n-1)}{2n};\frac{3}{2};\frac{1}{T}\right)\left(\frac{1}{T}\right)^{\frac{2}{n}-\frac{7}{2}} + 45\,2^{7+\frac{4}{n}}$$

$$\pi^4\,\Gamma\left(1-\frac{3}{2n}\right)\left(T\,{}_1F_1\left(1-\frac{3}{2n};\frac{1}{2};\frac{1}{T}\right) - 2\,{}_1F_1\left(1-\frac{3}{2n};\frac{3}{2};\frac{1}{T}\right)\right)$$

$$\left(\frac{1}{T}\right)^{\frac{2}{n}-3} - 20\,n^3\,\Lambda^4\,\Gamma\left(\frac{1}{2}\left(5+\frac{1}{n}\right)\right){}_1F_1\left(\frac{1}{2}\left(5+\frac{1}{n}\right);\frac{3}{2};\frac{1}{T}\right)\sqrt{\frac{1}{T}} +$$

$$18\,n\,T\,\Lambda^4\,\Gamma\left(1+\frac{1}{2n}\right){}_1F_1\left(1+\frac{1}{2n};\frac{1}{2};\frac{1}{T}\right) +$$

$$9\,n^2\,T\,\Lambda^4\,\Gamma\left(\frac{1}{2n}\right){}_1F_1\left(1+\frac{1}{2n};\frac{1}{2};\frac{1}{T}\right) +$$

$$18\,n\,T\,\Lambda^4\,\Gamma\left(\frac{1}{2n}\right){}_1F_1\left(1+\frac{1}{2n};\frac{1}{2};\frac{1}{T}\right) -$$

$$
\begin{aligned}
&10\,n^3\,\Lambda^4\,\Gamma\Big(2+\frac{1}{2\,n}\Big)\,{}_1F_1\Big(2+\frac{1}{2\,n};\,\frac{1}{2};\,\frac{1}{T}\Big)+\\
&72\,n^3\,T^2\,\Lambda^4\,\Gamma\Big(2+\frac{1}{2\,n}\Big)\,{}_1F_1\Big(2+\frac{1}{2\,n};\,\frac{1}{2};\,\frac{1}{T}\Big)+\\
&72\,n^2\,T^2\,\Lambda^4\,\Gamma\Big(2+\frac{1}{2\,n}\Big)\,{}_1F_1\Big(2+\frac{1}{2\,n};\,\frac{1}{2};\,\frac{1}{T}\Big)+\\
&18\,n\,T^2\,\Lambda^4\,\Gamma\Big(2+\frac{1}{2\,n}\Big)\,{}_1F_1\Big(2+\frac{1}{2\,n};\,\frac{1}{2};\,\frac{1}{T}\Big)-\\
&120\,n^2\,T\,\Lambda^4\,\Gamma\Big(2+\frac{1}{2\,n}\Big)\,{}_1F_1\Big(2+\frac{1}{2\,n};\,\frac{1}{2};\,\frac{1}{T}\Big)+\\
&80\,n^2\,\Lambda^4\,\Gamma\Big(2+\frac{1}{2\,n}\Big)\,{}_1F_1\Big(2+\frac{1}{2\,n};\,\frac{3}{2};\,\frac{1}{T}\Big)-\\
&144\,n^3\,T\,\Lambda^4\,\Gamma\Big(2+\frac{1}{2\,n}\Big)\,{}_1F_1\Big(2+\frac{1}{2\,n};\,\frac{3}{2};\,\frac{1}{T}\Big)-\\
&216\,n^2\,T\,\Lambda^4\,\Gamma\Big(2+\frac{1}{2\,n}\Big)\,{}_1F_1\Big(2+\frac{1}{2\,n};\,\frac{3}{2};\,\frac{1}{T}\Big)-\\
&72\,n\,T\,\Lambda^4\,\Gamma\Big(2+\frac{1}{2\,n}\Big)\,{}_1F_1\Big(2+\frac{1}{2\,n};\,\frac{3}{2};\,\frac{1}{T}\Big)-\\
&40\,n^2\,T^2\,\Lambda^4\,\Gamma\Big(3+\frac{1}{2\,n}\Big)\,{}_1F_1\Big(3+\frac{1}{2\,n};\,\frac{1}{2};\,\frac{1}{T}\Big)-\\
&60\,n^3\,T\,\Lambda^4\,\Gamma\Big(3+\frac{1}{2\,n}\Big)\,{}_1F_1\Big(3+\frac{1}{2\,n};\,\frac{1}{2};\,\frac{1}{T}\Big)+\\
&80\,n^3\,\Lambda^4\,\Gamma\Big(3+\frac{1}{2\,n}\Big)\,{}_1F_1\Big(3+\frac{1}{2\,n};\,\frac{3}{2};\,\frac{1}{T}\Big)+\\
&240\,n^2\,T\,\Lambda^4\,\Gamma\Big(3+\frac{1}{2\,n}\Big)\,{}_1F_1\Big(3+\frac{1}{2\,n};\,\frac{3}{2};\,\frac{1}{T}\Big)-\\
&10\,n^3\,T^2\,\Lambda^4\,\Gamma\Big(4+\frac{1}{2\,n}\Big)\,{}_1F_1\Big(4+\frac{1}{2\,n};\,\frac{1}{2};\,\frac{1}{T}\Big)+\\
&80\,n^3\,T\,\Lambda^4\,\Gamma\Big(4+\frac{1}{2\,n}\Big)\,{}_1F_1\Big(4+\frac{1}{2\,n};\,\frac{3}{2};\,\frac{1}{T}\Big)+\\
&\frac{40\,n^2\,\Lambda^4\,\Gamma(\frac{1}{2}\,(3+\frac{1}{n}))\,{}_1F_1(\frac{1}{2}\,(3+\frac{1}{n});\,\frac{1}{2};\,\frac{1}{T})}{\sqrt{\frac{1}{T}}}+\\
&\frac{36\,n^3\,\Lambda^4\,\Gamma(\frac{1}{2}\,(3+\frac{1}{n}))\,{}_1F_1(\frac{1}{2}\,(3+\frac{1}{n});\,\frac{3}{2};\,\frac{1}{T})}{\sqrt{\frac{1}{T}}}+\\
&\frac{72\,n^2\,\Lambda^4\,\Gamma(\frac{1}{2}\,(3+\frac{1}{n}))\,{}_1F_1(\frac{1}{2}\,(3+\frac{1}{n});\,\frac{3}{2};\,\frac{1}{T})}{\sqrt{\frac{1}{T}}}+\\
&\frac{36\,n\,\Lambda^4\,\Gamma(\frac{1}{2}\,(3+\frac{1}{n}))\,{}_1F_1(\frac{1}{2}\,(3+\frac{1}{n});\,\frac{3}{2};\,\frac{1}{T})}{\sqrt{\frac{1}{T}}}+\\
&\frac{40\,n^3\,\Lambda^4\,\Gamma(\frac{1}{2}\,(5+\frac{1}{n}))\,{}_1F_1(\frac{1}{2}\,(5+\frac{1}{n});\,\frac{1}{2};\,\frac{1}{T})}{\sqrt{\frac{1}{T}}}-
\end{aligned}
$$

$$\frac{240\,n^2\,\Lambda^4\,\Gamma(\frac{1}{2}\,(5+\frac{1}{n}))\,{}_1F_1(\frac{1}{2}\,(5+\frac{1}{n});\,\frac{3}{2};\,\frac{1}{T})}{\sqrt{\frac{1}{T}}} -$$

$$\frac{120\,n^3\,\Lambda^4\,\Gamma(\frac{1}{2}\,(7+\frac{1}{n}))\,{}_1F_1(\frac{1}{2}\,(7+\frac{1}{n});\,\frac{3}{2};\,\frac{1}{T})}{\sqrt{\frac{1}{T}}} -$$

$$\frac{72\,n^3\,\Lambda^4\,\Gamma(\frac{1}{2}\,(3+\frac{1}{n}))\,{}_1F_1(\frac{1}{2}\,(3+\frac{1}{n});\,\frac{1}{2};\,\frac{1}{T})}{(\frac{1}{T})^{3/2}} -$$

$$\frac{108\,n^2\,\Lambda^4\,\Gamma(\frac{1}{2}\,(3+\frac{1}{n}))\,{}_1F_1(\frac{1}{2}\,(3+\frac{1}{n});\,\frac{1}{2};\,\frac{1}{T})}{(\frac{1}{T})^{3/2}} -$$

$$\frac{36\,n\,\Lambda^4\,\Gamma(\frac{1}{2}\,(3+\frac{1}{n}))\,{}_1F_1(\frac{1}{2}\,(3+\frac{1}{n});\,\frac{1}{2};\,\frac{1}{T})}{(\frac{1}{T})^{3/2}} +$$

$$\frac{120\,n^2\,\Lambda^4\,\Gamma(\frac{1}{2}\,(5+\frac{1}{n}))\,{}_1F_1(\frac{1}{2}\,(5+\frac{1}{n});\,\frac{1}{2};\,\frac{1}{T})}{(\frac{1}{T})^{3/2}} +$$

$$\frac{144\,n^3\,\Lambda^4\,\Gamma(\frac{1}{2}\,(5+\frac{1}{n}))\,{}_1F_1(\frac{1}{2}\,(5+\frac{1}{n});\,\frac{3}{2};\,\frac{1}{T})}{(\frac{1}{T})^{3/2}} +$$

$$\frac{144\,n^2\,\Lambda^4\,\Gamma(\frac{1}{2}\,(5+\frac{1}{n}))\,{}_1F_1(\frac{1}{2}\,(5+\frac{1}{n});\,\frac{3}{2};\,\frac{1}{T})}{(\frac{1}{T})^{3/2}} +$$

$$\frac{36\,n\,\Lambda^4\,\Gamma(\frac{1}{2}\,(5+\frac{1}{n}))\,{}_1F_1(\frac{1}{2}\,(5+\frac{1}{n});\,\frac{3}{2};\,\frac{1}{T})}{(\frac{1}{T})^{3/2}} +$$

$$\frac{40\,n^3\,\Lambda^4\,\Gamma(\frac{1}{2}\,(7+\frac{1}{n}))\,{}_1F_1(\frac{1}{2}\,(7+\frac{1}{n});\,\frac{1}{2};\,\frac{1}{T})}{(\frac{1}{T})^{3/2}} -$$

$$\frac{80\,n^2\,\Lambda^4\,\Gamma(\frac{1}{2}\,(7+\frac{1}{n}))\,{}_1F_1(\frac{1}{2}\,(7+\frac{1}{n});\,\frac{3}{2};\,\frac{1}{T})}{(\frac{1}{T})^{3/2}} -$$

$$\frac{20\,n^3\,\Lambda^4\,\Gamma(\frac{1}{2}\,(9+\frac{1}{n}))\,{}_1F_1(\frac{1}{2}\,(9+\frac{1}{n});\,\frac{3}{2};\,\frac{1}{T})}{(\frac{1}{T})^{3/2}}\Bigg)\Bigg)$$

where $\Lambda = \hbar/(\sigma(m\,\epsilon)^{1/2})$ is the reduced de Broglie wavelength of relative motion. B^* is an even polynomial of fourth order in Λ. It contains the classical, first, and second quantum corrections as zeroth-, second-, and fourth- order coefficients, respectively. We extract the reduced representation of the second quantum correction by

bq2 = Coefficient[BStar, Λ, 4]

General::spell1 : Possible spelling error: new symbol name "bq2" is similar to existing symbol "Bq2". More…

$$\frac{1}{45\,\pi^4}\left(2^{-7-\frac{1}{n}}\left(\frac{1}{T}\right)^{4-\frac{1}{2n}}\left(9\,2^{2+\frac{1}{n}}\,n\,\sigma\left(\frac{1}{T}\right)^{\frac{1}{2n}-2}+9\,2^{1+\frac{1}{n}}\,e^{\frac{1}{T}}\,n\,\sqrt{\pi}\,\sigma\left(\frac{1}{T}\right)^{\frac{1}{2}\left(\frac{1}{n}-3\right)}+\right.\right.$$

$$9\,2^{1+\frac{1}{n}}\,e^{\frac{1}{T}}\,n\,\sqrt{\pi}\,\sigma\,\mathrm{erf}\left(\sqrt{\frac{1}{T}}\right)\left(\frac{1}{T}\right)^{\frac{1}{2}\left(\frac{1}{n}-3\right)}-$$

$$20\,n^3\,\Gamma\left(\frac{1}{2}\left(5+\frac{1}{n}\right)\right){}_1F_1\left(\frac{1}{2}\left(5+\frac{1}{n}\right);\frac{3}{2};\frac{1}{T}\right)\sqrt{\frac{1}{T}}+$$

$$18\,n\,T\,\Gamma\left(1+\frac{1}{2n}\right){}_1F_1\left(1+\frac{1}{2n};\frac{1}{2};\frac{1}{T}\right)+9\,n^2\,T\,\Gamma\left(\frac{1}{2n}\right)$$

$${}_1F_1\left(1+\frac{1}{2n};\frac{1}{2};\frac{1}{T}\right)+18\,n\,T\,\Gamma\left(\frac{1}{2n}\right){}_1F_1\left(1+\frac{1}{2n};\frac{1}{2};\frac{1}{T}\right)-$$

$$10\,n^3\,\Gamma\left(2+\frac{1}{2n}\right){}_1F_1\left(2+\frac{1}{2n};\frac{1}{2};\frac{1}{T}\right)+72\,n^3\,T^2\,\Gamma\left(2+\frac{1}{2n}\right)$$

$${}_1F_1\left(2+\frac{1}{2n};\frac{1}{2};\frac{1}{T}\right)+72\,n^2\,T^2\,\Gamma\left(2+\frac{1}{2n}\right){}_1F_1\left(2+\frac{1}{2n};\frac{1}{2};\frac{1}{T}\right)+$$

$$18\,n\,T^2\,\Gamma\left(2+\frac{1}{2n}\right){}_1F_1\left(2+\frac{1}{2n};\frac{1}{2};\frac{1}{T}\right)-120\,n^2\,T\,\Gamma\left(2+\frac{1}{2n}\right)$$

$${}_1F_1\left(2+\frac{1}{2n};\frac{1}{2};\frac{1}{T}\right)+80\,n^2\,\Gamma\left(2+\frac{1}{2n}\right){}_1F_1\left(2+\frac{1}{2n};\frac{3}{2};\frac{1}{T}\right)-$$

$$144\,n^3\,T\,\Gamma\left(2+\frac{1}{2n}\right){}_1F_1\left(2+\frac{1}{2n};\frac{3}{2};\frac{1}{T}\right)-$$

$$216\,n^2\,T\,\Gamma\left(2+\frac{1}{2n}\right){}_1F_1\left(2+\frac{1}{2n};\frac{3}{2};\frac{1}{T}\right)-$$

$$72\,n\,T\,\Gamma\left(2+\frac{1}{2n}\right){}_1F_1\left(2+\frac{1}{2n};\frac{3}{2};\frac{1}{T}\right)-$$

$$40\,n^2\,T^2\,\Gamma\left(3+\frac{1}{2n}\right){}_1F_1\left(3+\frac{1}{2n};\frac{1}{2};\frac{1}{T}\right)-$$

$$60\,n^3\,T\,\Gamma\left(3+\frac{1}{2n}\right){}_1F_1\left(3+\frac{1}{2n};\frac{1}{2};\frac{1}{T}\right)+$$

$$80\,n^3\,\Gamma\left(3+\frac{1}{2n}\right){}_1F_1\left(3+\frac{1}{2n};\frac{3}{2};\frac{1}{T}\right)+$$

$$240\,n^2\,T\,\Gamma\left(3+\frac{1}{2n}\right){}_1F_1\left(3+\frac{1}{2n};\frac{3}{2};\frac{1}{T}\right)-$$

$$10\,n^3\,T^2\,\Gamma\left(4+\frac{1}{2n}\right){}_1F_1\left(4+\frac{1}{2n};\frac{1}{2};\frac{1}{T}\right)+$$

$$80\,n^3\,T\,\Gamma\left(4+\frac{1}{2\,n}\right){}_1F_1\left(4+\frac{1}{2\,n};\,\frac{3}{2};\,\frac{1}{T}\right)+$$
$$\frac{40\,n^2\,\Gamma(\frac{1}{2}\,(3+\frac{1}{n}))\,{}_1F_1(\frac{1}{2}\,(3+\frac{1}{n});\,\frac{1}{2};\,\frac{1}{T})}{\sqrt{\frac{1}{T}}}+$$
$$\frac{36\,n^3\,\Gamma(\frac{1}{2}\,(3+\frac{1}{n}))\,{}_1F_1(\frac{1}{2}\,(3+\frac{1}{n});\,\frac{3}{2};\,\frac{1}{T})}{\sqrt{\frac{1}{T}}}+$$
$$\frac{72\,n^2\,\Gamma(\frac{1}{2}\,(3+\frac{1}{n}))\,{}_1F_1(\frac{1}{2}\,(3+\frac{1}{n});\,\frac{3}{2};\,\frac{1}{T})}{\sqrt{\frac{1}{T}}}+$$
$$\frac{36\,n\,\Gamma(\frac{1}{2}\,(3+\frac{1}{n}))\,{}_1F_1(\frac{1}{2}\,(3+\frac{1}{n});\,\frac{3}{2};\,\frac{1}{T})}{\sqrt{\frac{1}{T}}}+$$
$$\frac{40\,n^3\,\Gamma(\frac{1}{2}\,(5+\frac{1}{n}))\,{}_1F_1(\frac{1}{2}\,(5+\frac{1}{n});\,\frac{1}{2};\,\frac{1}{T})}{\sqrt{\frac{1}{T}}}-$$
$$\frac{240\,n^2\,\Gamma(\frac{1}{2}\,(5+\frac{1}{n}))\,{}_1F_1(\frac{1}{2}\,(5+\frac{1}{n});\,\frac{3}{2};\,\frac{1}{T})}{\sqrt{\frac{1}{T}}}-$$
$$\frac{120\,n^3\,\Gamma(\frac{1}{2}\,(7+\frac{1}{n}))\,{}_1F_1(\frac{1}{2}\,(7+\frac{1}{n});\,\frac{3}{2};\,\frac{1}{T})}{\sqrt{\frac{1}{T}}}-$$
$$\frac{72\,n^3\,\Gamma(\frac{1}{2}\,(3+\frac{1}{n}))\,{}_1F_1(\frac{1}{2}\,(3+\frac{1}{n});\,\frac{1}{2};\,\frac{1}{T})}{(\frac{1}{T})^{3/2}}-$$
$$\frac{108\,n^2\,\Gamma(\frac{1}{2}\,(3+\frac{1}{n}))\,{}_1F_1(\frac{1}{2}\,(3+\frac{1}{n});\,\frac{1}{2};\,\frac{1}{T})}{(\frac{1}{T})^{3/2}}-$$
$$\frac{36\,n\,\Gamma(\frac{1}{2}\,(3+\frac{1}{n}))\,{}_1F_1(\frac{1}{2}\,(3+\frac{1}{n});\,\frac{1}{2};\,\frac{1}{T})}{(\frac{1}{T})^{3/2}}+$$
$$\frac{120\,n^2\,\Gamma(\frac{1}{2}\,(5+\frac{1}{n}))\,{}_1F_1(\frac{1}{2}\,(5+\frac{1}{n});\,\frac{1}{2};\,\frac{1}{T})}{(\frac{1}{T})^{3/2}}+$$
$$\frac{144\,n^3\,\Gamma(\frac{1}{2}\,(5+\frac{1}{n}))\,{}_1F_1(\frac{1}{2}\,(5+\frac{1}{n});\,\frac{3}{2};\,\frac{1}{T})}{(\frac{1}{T})^{3/2}}+$$
$$\frac{144\,n^2\,\Gamma(\frac{1}{2}\,(5+\frac{1}{n}))\,{}_1F_1(\frac{1}{2}\,(5+\frac{1}{n});\,\frac{3}{2};\,\frac{1}{T})}{(\frac{1}{T})^{3/2}}+$$
$$\frac{36\,n\,\Gamma(\frac{1}{2}\,(5+\frac{1}{n}))\,{}_1F_1(\frac{1}{2}\,(5+\frac{1}{n});\,\frac{3}{2};\,\frac{1}{T})}{(\frac{1}{T})^{3/2}}+$$

$$\frac{40\,n^3\,\Gamma(\frac{1}{2}\,(7+\frac{1}{n}))\,{}_1F_1(\frac{1}{2}\,(7+\frac{1}{n});\,\frac{1}{2};\,\frac{1}{T})}{(\frac{1}{T})^{3/2}} -$$

$$\frac{80\,n^2\,\Gamma(\frac{1}{2}\,(7+\frac{1}{n}))\,{}_1F_1(\frac{1}{2}\,(7+\frac{1}{n});\,\frac{3}{2};\,\frac{1}{T})}{(\frac{1}{T})^{3/2}} -$$

$$\left.\frac{20\,n^3\,\Gamma(\frac{1}{2}\,(9+\frac{1}{n}))\,{}_1F_1(\frac{1}{2}\,(9+\frac{1}{n});\,\frac{3}{2};\,\frac{1}{T})}{(\frac{1}{T})^{3/2}}\right)\Bigg)$$

The first quantum mechanical correction is extracted by

```
bq1 = Coefficient[BStar, Λ, 2]
```

General::spell1 : Possible spelling error: new symbol name "bq1" is similar to existing symbol "Bq1". More…

$$\frac{1}{45\,\pi^4}\left(2^{-7-\frac{1}{n}}\left(\frac{1}{T}\right)^{4-\frac{1}{2n}}\left(-45\,2^{4+\frac{2}{n}}\,n\,\pi^2\,\Gamma\left(2-\frac{1}{2n}\right){}_1F_1\left(2-\frac{1}{2n};\,\frac{1}{2};\,\frac{1}{T}\right)\left(\frac{1}{T}\right)^{\frac{1}{n}-3}+\right.\right.$$

$$45\,2^{5+\frac{2}{n}}\,n\,\pi^2\,\Gamma\left(\frac{3}{2}-\frac{1}{2n}\right){}_1F_1\left(\frac{3}{2}-\frac{1}{2n};\,\frac{1}{2};\,\frac{1}{T}\right)\left(\frac{1}{T}\right)^{\frac{1}{n}-\frac{5}{2}}-$$

$$45\,2^{5+\frac{2}{n}}\,n\,\pi^2\,\Gamma\left(\frac{5}{2}-\frac{1}{2n}\right){}_1F_1\left(\frac{5}{2}-\frac{1}{2n};\,\frac{3}{2};\,\frac{1}{T}\right)\left(\frac{1}{T}\right)^{\frac{1}{n}-\frac{5}{2}}-$$

$$45\,2^{4+\frac{2}{n}}\,n\,\pi^2\,\Gamma\left(1-\frac{1}{2n}\right){}_1F_1\left(1-\frac{1}{2n};\,\frac{1}{2};\,\frac{1}{T}\right)\left(\frac{1}{T}\right)^{\frac{1}{n}-2}+$$

$$45\,2^{6+\frac{2}{n}}\,n\,\pi^2\,\Gamma\left(2-\frac{1}{2n}\right){}_1F_1\left(2-\frac{1}{2n};\,\frac{3}{2};\,\frac{1}{T}\right)\left(\frac{1}{T}\right)^{\frac{1}{n}-2}-$$

$$\left.\left.45\,2^{5+\frac{2}{n}}\,n\,\pi^2\,\Gamma\left(\frac{3}{2}-\frac{1}{2n}\right){}_1F_1\left(\frac{3}{2}-\frac{1}{2n};\,\frac{3}{2};\,\frac{1}{T}\right)\left(\frac{1}{T}\right)^{\frac{1}{n}-\frac{3}{2}}\right)\right)$$

And the classical SVC in reduced variables is

```
bc = Coefficient[BStar, Λ, 0]
```

$$\frac{1}{45\pi^4}\left(2^{-7-\frac{1}{n}}\left(\frac{1}{T}\right)^{4-\frac{1}{2n}}\left(-45\,2^{7+\frac{4}{n}}\,\pi^4\,\Gamma\left(\frac{n-3}{2n}\right){}_1F_1\left(\frac{n-3}{2n};\frac{1}{2};\frac{1}{T}\right)\left(\frac{1}{T}\right)^{\frac{2}{n}-\frac{7}{2}}+\right.\right.$$
$$45\,2^{8+\frac{4}{n}}\,\pi^4\,\Gamma\left(\frac{3(n-1)}{2n}\right){}_1F_1\left(\frac{3(n-1)}{2n};\frac{3}{2};\frac{1}{T}\right)\left(\frac{1}{T}\right)^{\frac{2}{n}-\frac{7}{2}}+$$
$$45\,2^{7+\frac{4}{n}}\,\pi^4\,\Gamma\left(1-\frac{3}{2n}\right)$$
$$\left.\left.\left(T\,{}_1F_1\left(1-\frac{3}{2n};\frac{1}{2};\frac{1}{T}\right)-2\,{}_1F_1\left(1-\frac{3}{2n};\frac{3}{2};\frac{1}{T}\right)\right)\left(\frac{1}{T}\right)^{\frac{2}{n}-3}\right)\right)$$

The derived results are analytic expressions in terms of hypergeometric functions ${}_1F_1$ allowing a graphical and analytical treatment of the SVC, including quantum corrections. The representation of the second virial coefficient up to second-order quantum corrections is thus given by

```
bstar = bc + Λ^2 bq1 + Λ^4 bq2;
```

To demonstrate the influence of the quantum mechanical corrections, let us graphically examine the classical SVC, the two quantum corrections, and the total representation of the SVC. We plot the reduced quantities depending on the variable $T^* = T$. Figure 5.7.23 shows the influence of the first and second quantum correction on the SVC:

```
Plot[Evaluate[
   {bc, bq1, bq2, bstar} /. {Λ → 1, σ → 1, n → 6}],
  {T, 0.2, 10}, AxesLabel → {"T*", "B*"},
  PlotStyle → {RGBColor[0, 0, 1], RGBColor[0, 1, 0],
    RGBColor[1, 0, 0], RGBColor[0, 0, 0]}, PlotRange →
   {-2, 2}, Prolog → {Text["B*c", {1.6607, -0.650973}],
    Text["B*q2", {0.660171, -0.452358}],
    Text["B*q1", {0.861535, 0.704103}],
    Text["B*", {2.08858, -0.916639}]}];
```

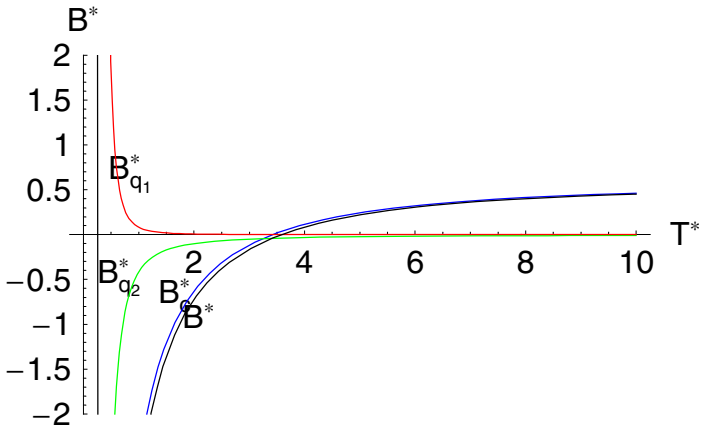


Figure 5.7.23. The figure contains the classical SVC (blue), the first quantum mechanical correction (red), the second quantum correction of SVC (green), and the sum of the three parts (black). We note that the second quantum corrections contains terms linear in σ. Therefore, in addition to Λ and n we have to specify the value of σ.

For practical applications, it is sometimes necessary to have the numerical values of the SVC and its first and second temperature derivatives available. The numerical values of these quantities are tabulated in the book by Hirschfelder et al. for the (12-6)-LJ potential. The first and second derivative of B_c^* with respect to T^* then follows by

b1 = $T \dfrac{\partial \mathbf{bc}}{\partial T}$

$$
T\left(\frac{1}{45\pi^4}\left(2^{-7-\frac{1}{n}}\left(\frac{1}{T}\right)^{4-\frac{1}{2n}}\left(45\,2^{7+\frac{4}{n}}\,\pi^4\,\Gamma\left(1-\frac{3}{2n}\right)\right.\right.\right.
$$

$$
\left(\left.{}_1F_1\left(1-\frac{3}{2n};\frac{1}{2};\frac{1}{T}\right)-\frac{2\left(1-\frac{3}{2n}\right){}_1F_1\left(2-\frac{3}{2n};\frac{3}{2};\frac{1}{T}\right)}{T}+\right.\right.
$$

$$
\left.\left.\frac{4\left(1-\frac{3}{2n}\right){}_1F_1\left(2-\frac{3}{2n};\frac{5}{2};\frac{1}{T}\right)}{3\,T^2}\right)\left(\frac{1}{T}\right)^{\frac{2}{n}-3}+
$$

$$
45\,2^{7+\frac{4}{n}}\left(\frac{2}{n}-\frac{7}{2}\right)\pi^4\,\Gamma\left(\frac{n-3}{2n}\right){}_1F_1\left(\frac{n-3}{2n};\frac{1}{2};\frac{1}{T}\right)\left(\frac{1}{T}\right)^{\frac{2}{n}-\frac{5}{2}}-
$$

$$
45\,2^{8+\frac{4}{n}}\left(\frac{2}{n}-\frac{7}{2}\right)\pi^4\,\Gamma\left(\frac{3\,(n-1)}{2n}\right){}_1F_1\left(\frac{3\,(n-1)}{2n};\frac{3}{2};\frac{1}{T}\right)
$$

$$
\left(\frac{1}{T}\right)^{\frac{2}{n}-\frac{5}{2}}-45\,2^{7+\frac{4}{n}}\left(\frac{2}{n}-3\right)\pi^4\,\Gamma\left(1-\frac{3}{2n}\right)
$$

$$
\left(T\,{}_1F_1\left(1-\frac{3}{2n};\frac{1}{2};\frac{1}{T}\right)-2\,{}_1F_1\left(1-\frac{3}{2n};\frac{3}{2};\frac{1}{T}\right)\right)\left(\frac{1}{T}\right)^{\frac{2}{n}-2}+
$$

$$
\frac{45\,2^{7+\frac{4}{n}}\,(n-3)\,\pi^4\,\Gamma\left(\frac{n-3}{2n}\right){}_1F_1\left(\frac{n-3}{2n}+1;\frac{3}{2};\frac{1}{T}\right)\left(\frac{1}{T}\right)^{\frac{2}{n}-\frac{3}{2}}}{n}-
$$

$$
\frac{1}{n}\left(45\,2^{8+\frac{4}{n}}\,(n-1)\,\pi^4\,\Gamma\left(\frac{3\,(n-1)}{2n}\right)\right.
$$

$$
\left.\left.\left.{}_1F_1\left(\frac{3\,(n-1)}{2n}+1;\frac{5}{2};\frac{1}{T}\right)\left(\frac{1}{T}\right)^{\frac{2}{n}-\frac{3}{2}}\right)\right)\right)-
$$

$$
\frac{1}{45\pi^4}\left(2^{-7-\frac{1}{n}}\left(4-\frac{1}{2n}\right)\left(\frac{1}{T}\right)^{5-\frac{1}{2n}}\left(-45\,2^{7+\frac{4}{n}}\,\pi^4\,\Gamma\left(\frac{n-3}{2n}\right)\right.\right.
$$

$$
{}_1F_1\left(\frac{n-3}{2n};\frac{1}{2};\frac{1}{T}\right)\left(\frac{1}{T}\right)^{\frac{2}{n}-\frac{7}{2}}+45\,2^{8+\frac{4}{n}}\,\pi^4\,\Gamma\left(\frac{3\,(n-1)}{2n}\right)
$$

$$
{}_1F_1\left(\frac{3\,(n-1)}{2n};\frac{3}{2};\frac{1}{T}\right)\left(\frac{1}{T}\right)^{\frac{2}{n}-\frac{7}{2}}+45\,2^{7+\frac{4}{n}}\,\pi^4\,\Gamma\left(1-\frac{3}{2n}\right)
$$

$$
\left.\left.\left(T\,{}_1F_1\left(1-\frac{3}{2n};\frac{1}{2};\frac{1}{T}\right)-2\,{}_1F_1\left(1-\frac{3}{2n};\frac{3}{2};\frac{1}{T}\right)\right)\left(\frac{1}{T}\right)^{\frac{2}{n}-3}\right)\right)\right)
$$

b2 = $T^2 \dfrac{\partial^2 \mathrm{bc}}{\partial T\, \partial T}$

$$
\begin{aligned}
&T^2\Bigg(\frac{1}{45\pi^4}\Bigg(2^{-7-\frac{1}{n}}\Bigg(45\,2^{7+\frac{4}{n}}\pi^4\,\Gamma\Big(1-\frac{3}{2n}\Big)\Bigg(-\frac{8\,(1-\frac{3}{2n})\,{}_1F_1(2-\frac{3}{2n};\frac{5}{2};\frac{1}{T})}{3\,T^3}+\\
&\qquad\frac{4\,(1-\frac{3}{2n})\,(2-\frac{3}{2n})\,{}_1F_1(3-\frac{3}{2n};\frac{5}{2};\frac{1}{T})}{3\,T^3}-\\
&\qquad\frac{8\,(1-\frac{3}{2n})\,(2-\frac{3}{2n})\,{}_1F_1(3-\frac{3}{2n};\frac{7}{2};\frac{1}{T})}{15\,T^4}\Bigg)\\
&\quad\Big(\frac{1}{T}\Big)^{\frac{2}{n}-3}-90\,2^{7+\frac{4}{n}}\Big(\frac{2}{n}-3\Big)\pi^4\,\Gamma\Big(1-\frac{3}{2n}\Big)\\
&\quad\Bigg({}_1F_1\Big(1-\frac{3}{2n};\frac{1}{2};\frac{1}{T}\Big)-\frac{2\,(1-\frac{3}{2n})\,{}_1F_1(2-\frac{3}{2n};\frac{3}{2};\frac{1}{T})}{T}+\\
&\qquad\frac{4\,(1-\frac{3}{2n})\,{}_1F_1(2-\frac{3}{2n};\frac{5}{2};\frac{1}{T})}{3\,T^2}\Bigg)\Big(\frac{1}{T}\Big)^{\frac{2}{n}-2}-\\
&\quad45\,2^{7+\frac{4}{n}}\Big(\frac{2}{n}-\frac{7}{2}\Big)\Big(\frac{2}{n}-\frac{5}{2}\Big)\pi^4\,\Gamma\Big(\frac{n-3}{2n}\Big)\,{}_1F_1\Big(\frac{n-3}{2n};\frac{1}{2};\frac{1}{T}\Big)\\
&\quad\Big(\frac{1}{T}\Big)^{\frac{2}{n}-\frac{3}{2}}+45\,2^{8+\frac{4}{n}}\Big(\frac{2}{n}-\frac{7}{2}\Big)\Big(\frac{2}{n}-\frac{5}{2}\Big)\pi^4\\
&\quad\Gamma\Big(\frac{3\,(n-1)}{2n}\Big)\,{}_1F_1\Big(\frac{3\,(n-1)}{2n};\frac{3}{2};\frac{1}{T}\Big)\Big(\frac{1}{T}\Big)^{\frac{2}{n}-\frac{3}{2}}+\\
&\quad45\,2^{7+\frac{4}{n}}\Big(\frac{2}{n}-3\Big)\Big(\frac{2}{n}-2\Big)\pi^4\,\Gamma\Big(1-\frac{3}{2n}\Big)\\
&\quad\Big(T\,{}_1F_1\Big(1-\frac{3}{2n};\frac{1}{2};\frac{1}{T}\Big)-2\,{}_1F_1\Big(1-\frac{3}{2n};\frac{3}{2};\frac{1}{T}\Big)\Big)\Big(\frac{1}{T}\Big)^{\frac{2}{n}-1}-\\
&\quad\frac{1}{n}\Bigg(45\,2^{7+\frac{4}{n}}\Big(\frac{2}{n}-\frac{7}{2}\Big)(n-3)\,\pi^4\,\Gamma\Big(\frac{n-3}{2n}\Big)\\
&\qquad{}_1F_1\Big(\frac{n-3}{2n}+1;\frac{3}{2};\frac{1}{T}\Big)\Big(\frac{1}{T}\Big)^{\frac{2}{n}-\frac{1}{2}}\Bigg)-\\
&\quad\frac{1}{n}\Bigg(45\,2^{7+\frac{4}{n}}\Big(\frac{2}{n}-\frac{3}{2}\Big)(n-3)\,\pi^4\,\Gamma\Big(\frac{n-3}{2n}\Big)\\
&\qquad{}_1F_1\Big(\frac{n-3}{2n}+1;\frac{3}{2};\frac{1}{T}\Big)\Big(\frac{1}{T}\Big)^{\frac{2}{n}-\frac{1}{2}}\Bigg)+
\end{aligned}
$$

$$\frac{1}{n}\left(45\,2^{8+\frac{4}{n}}\left(\frac{2}{n}-\frac{7}{2}\right)(n-1)\,\pi^4\,\Gamma\left(\frac{3\,(n-1)}{2\,n}\right)\right.$$

$$\left.{}_1F_1\left(\frac{3\,(n-1)}{2\,n}+1;\,\frac{5}{2};\,\frac{1}{T}\right)\left(\frac{1}{T}\right)^{\frac{2}{n}-\frac{1}{2}}\right)+$$

$$\frac{1}{n}\left(45\,2^{8+\frac{4}{n}}\left(\frac{2}{n}-\frac{3}{2}\right)(n-1)\,\pi^4\,\Gamma\left(\frac{3\,(n-1)}{2\,n}\right)\right.$$

$$\left.{}_1F_1\left(\frac{3\,(n-1)}{2\,n}+1;\,\frac{5}{2};\,\frac{1}{T}\right)\left(\frac{1}{T}\right)^{\frac{2}{n}-\frac{1}{2}}\right)-$$

$$\frac{1}{n}\left(15\,2^{8+\frac{4}{n}}\left(\frac{n-3}{2\,n}+1\right)(n-3)\,\pi^4\,\Gamma\left(\frac{n-3}{2\,n}\right)\right.$$

$$\left.{}_1F_1\left(\frac{n-3}{2\,n}+2;\,\frac{5}{2};\,\frac{1}{T}\right)\left(\frac{1}{T}\right)^{\frac{1}{2}+\frac{2}{n}}\right)+$$

$$\frac{1}{n}\left(9\,2^{9+\frac{4}{n}}\left(\frac{3\,(n-1)}{2\,n}+1\right)(n-1)\,\pi^4\,\Gamma\left(\frac{3\,(n-1)}{2\,n}\right)\right.$$

$$\left.\left.{}_1F_1\left(\frac{3\,(n-1)}{2\,n}+2;\,\frac{7}{2};\,\frac{1}{T}\right)\left(\frac{1}{T}\right)^{\frac{1}{2}+\frac{2}{n}}\right)\right)\left(\frac{1}{T}\right)^{4-\frac{1}{2n}}\Bigg)-$$

$$\frac{1}{45\,\pi^4}\left(2\,2^{-7-\frac{1}{n}}\left(4-\frac{1}{2\,n}\right)\left(45\,2^{7+\frac{4}{n}}\,\pi^4\,\Gamma\left(1-\frac{3}{2\,n}\right)\right.\right.$$

$$\left(\left.{}_1F_1\left(1-\frac{3}{2\,n};\,\frac{1}{2};\,\frac{1}{T}\right)-\frac{2\,(1-\frac{3}{2\,n})\,{}_1F_1(2-\frac{3}{2\,n};\,\frac{3}{2};\,\frac{1}{T})}{T}+\right.\right.$$

$$\left.\left.\frac{4\,(1-\frac{3}{2\,n})\,{}_1F_1(2-\frac{3}{2\,n};\,\frac{5}{2};\,\frac{1}{T})}{3\,T^2}\right)\left(\frac{1}{T}\right)^{\frac{2}{n}-3}+\right.$$

$$45\,2^{7+\frac{4}{n}}\left(\frac{2}{n}-\frac{7}{2}\right)\pi^4\,\Gamma\left(\frac{n-3}{2\,n}\right){}_1F_1\left(\frac{n-3}{2\,n};\,\frac{1}{2};\,\frac{1}{T}\right)\left(\frac{1}{T}\right)^{\frac{2}{n}-\frac{5}{2}}-$$

$$45\,2^{8+\frac{4}{n}}\left(\frac{2}{n}-\frac{7}{2}\right)\pi^4\,\Gamma\left(\frac{3\,(n-1)}{2\,n}\right){}_1F_1\left(\frac{3\,(n-1)}{2\,n};\,\frac{3}{2};\,\frac{1}{T}\right)$$

$$\left(\frac{1}{T}\right)^{\frac{2}{n}-\frac{5}{2}}-45\,2^{7+\frac{4}{n}}\left(\frac{2}{n}-3\right)\pi^4\,\Gamma\left(1-\frac{3}{2\,n}\right)$$

$$\left.\left(T\,{}_1F_1\left(1-\frac{3}{2\,n};\,\frac{1}{2};\,\frac{1}{T}\right)-2\,{}_1F_1\left(1-\frac{3}{2\,n};\,\frac{3}{2};\,\frac{1}{T}\right)\right)\left(\frac{1}{T}\right)^{\frac{2}{n}-2}+\right.$$

$$\left.\frac{45\,2^{7+\frac{4}{n}}\,(n-3)\,\pi^4\,\Gamma(\frac{n-3}{2\,n})\,{}_1F_1(\frac{n-3}{2\,n}+1;\,\frac{3}{2};\,\frac{1}{T})\,(\frac{1}{T})^{\frac{2}{n}-\frac{3}{2}}}{n}-\right.$$

$$\frac{1}{n}\left(45\,2^{8+\frac{4}{n}}\,(n-1)\,\pi^4\,\Gamma\Big(\frac{3\,(n-1)}{2\,n}\Big)\right.$$
$$\left.{}_1F_1\Big(\frac{3\,(n-1)}{2\,n}+1;\,\frac{5}{2};\,\frac{1}{T}\Big)\Big(\frac{1}{T}\Big)^{\frac{2}{n}-\frac{3}{2}}\right)$$
$$\Big(\frac{1}{T}\Big)^{5-\frac{1}{2n}}\Big)+\frac{1}{45\,\pi^4}\left(2^{-7-\frac{1}{n}}\Big(4-\frac{1}{2\,n}\Big)\Big(5-\frac{1}{2\,n}\Big)\right.$$
$$\left(-45\,2^{7+\frac{4}{n}}\,\pi^4\,\Gamma\Big(\frac{n-3}{2\,n}\Big)\,{}_1F_1\Big(\frac{n-3}{2\,n};\,\frac{1}{2};\,\frac{1}{T}\Big)\Big(\frac{1}{T}\Big)^{\frac{2}{n}-\frac{7}{2}}+\right.$$
$$45\,2^{8+\frac{4}{n}}\,\pi^4\,\Gamma\Big(\frac{3\,(n-1)}{2\,n}\Big)\,{}_1F_1\Big(\frac{3\,(n-1)}{2\,n};\,\frac{3}{2};\,\frac{1}{T}\Big)\Big(\frac{1}{T}\Big)^{\frac{2}{n}-\frac{7}{2}}+$$
$$45\,2^{7+\frac{4}{n}}\,\pi^4\,\Gamma\Big(1-\frac{3}{2\,n}\Big)\Big(T\,{}_1F_1\Big(1-\frac{3}{2\,n};\,\frac{1}{2};\,\frac{1}{T}\Big)-$$
$$\left.\left.2\,{}_1F_1\Big(1-\frac{3}{2\,n};\,\frac{3}{2};\,\frac{1}{T}\Big)\Big)\Big(\frac{1}{T}\Big)^{\frac{2}{n}-3}\right)\Big(\frac{1}{T}\Big)^{6-\frac{1}{2n}}\right)\right)$$

The first few lines of table I-B contained in the appendix of Hirschfelder et al. then follows by

```
t1 = Table[N[{T, bc, b1, b2, b1 − bc} /. n → 6, 9], {T, .3, 1, .05}];
PrependTo[t1, {"T", "B*c", "b1", "b2", "b1−B*c"}];
TableForm[Map[Map[PaddedForm[#, {5, 2}] &, #] &, t1]]
```

T	B_c^*	b1	b2	b1−B_c^*
0.30	−27.88	76.61	−356.88	104.49
0.35	−18.75	45.25	−189.47	64.00
0.40	−13.80	30.27	−116.37	44.07
0.45	−10.75	21.99	−78.88	32.74
0.50	−8.72	16.92	−57.34	25.64
0.55	−7.27	13.58	−43.88	20.86
0.60	−6.20	11.25	−34.92	17.45
0.65	−5.37	9.55	−28.64	14.91
0.70	−4.71	8.26	−24.06	12.97
0.75	−4.18	7.25	−20.61	11.43
0.80	−3.73	6.45	−17.94	10.19
0.85	−3.36	5.80	−15.83	9.17
0.90	−3.05	5.26	−14.12	8.31
0.95	−2.77	4.81	−12.71	7.59
1.00	−2.54	4.43	−11.54	6.97

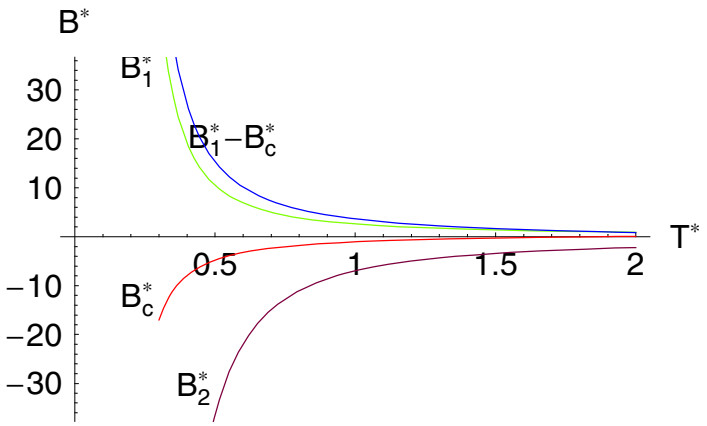
Comparing the calculated figures with Hirschfelder's result demonstrates excellent agreement. The analytical results derived in the calculations above are not restricted to the (12-6)-LJ potential but allow any choice of the exponent $n > 3$. For example, we can determine the SVC for a (16-8)-potential by

```
t2 = Table[N[{T, bc, b1, b2, b1 − bc} /. n → 8, 9], {T, 1, 2, .05}];
PrependTo[t2,
  {"T", "\!\(\(B\_c\%∗\)\)", "b1", "b2", "b1−\!\(\(B\_c\%∗\)\)"}];
TableForm[((PaddedForm[#1, {5, 4}] &) /@ #1 &) /@ t2]
```

T	B_c^*	b1	b2	b1−B_c^*
1.0000	−1.0453	2.6042	−6.9604	3.6495
1.0500	−0.9233	2.4020	−6.3427	3.3253
1.1000	−0.8157	2.2270	−5.8170	3.0427
1.1500	−0.7201	2.0742	−5.3651	2.7943
1.2000	−0.6348	1.9397	−4.9732	2.5745
1.2500	−0.5580	1.8205	−4.6305	2.3786
1.3000	−0.4887	1.7142	−4.3288	2.2029
1.3500	−0.4259	1.6188	−4.0612	2.0447
1.4000	−0.3686	1.5328	−3.8226	1.9014
1.4500	−0.3162	1.4549	−3.6087	1.7710
1.5000	−0.2681	1.3839	−3.4160	1.6520
1.5500	−0.2238	1.3191	−3.2415	1.5429
1.6000	−0.1828	1.2597	−3.0828	1.4425
1.6500	−0.1449	1.2050	−2.9380	1.3499
1.7000	−0.1097	1.1545	−2.8055	1.2642
1.7500	−0.0769	1.1077	−2.6836	1.1847
1.8000	−0.0463	1.0643	−2.5713	1.1107
1.8500	−0.0177	1.0239	−2.4674	1.0416
1.9000	0.0091	0.9862	−2.3711	0.9771
1.9500	0.0342	0.9509	−2.2817	0.9167
2.0000	0.0579	0.9179	−2.1983	0.8600

We also can represent the data graphically:

```
Plot[Evaluate[{bc, b1, b2, b1 − bc} /. n → 8],
    {T, .3, 2}, AxesLabel → {"T*", "B*"}, PlotStyle →
      {RGBColor[0.996109, 0, 0], RGBColor[0.500008, 0.996109, 0],
        RGBColor[0.500008, 0, 0.250004], RGBColor[0, 0, 0.996109]},
    Prolog → {Text["Bc*", {0.224533, −12.4014}], Text["B1*",
          {0.224533, 34.4014}], Text["B2*", {0.425166, −30.6213}],
        Text["B1*−Bc*", {0.564147, 20.2026}]}];
```



Knowing the analytical expressions of the SVC, we are able to calculate either numerical values of of the classical SVC and its derivatives or represent the data graphically. We are not only restricted to classical values but can incorporate the quantum mechanical corrections. The first and second temperature derivatives for B_q^* are

```
bqq1 = T ∂T bstar;
```

```
bqq2 = T^2 ∂T,T bstar;
```

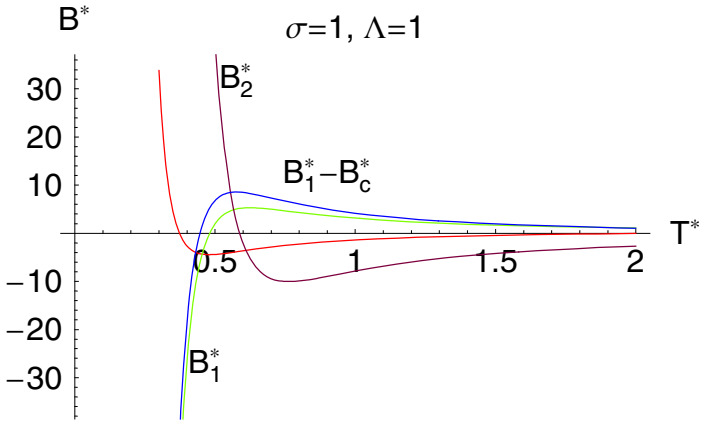
A table containing the SVC with quantum corrections and the two derivatives is generated by

```
t3 = Table[
    N[{T, bstar, bqq1, bqq2, bqq1 – bstar} /. {n → 8, σ → 1, Λ → 1}, 9],
    {T, 1, 2, .05}]; PrependTo[t3, {"T", "B*", "B1*", "B2*", "B1*–B*"}];
TableForm[Map[Map[PaddedForm[#, {5, 4}] &, #] &, t3]]
```

T	B^*	B_1^*	B_2^*	$B_1^*-B^*$
1.0000	−1.3947	3.1190	−7.8531	4.5137
1.0500	−1.2481	2.8954	−7.3252	4.1435
1.1000	−1.1181	2.6961	−6.8361	3.8141
1.1500	−1.0022	2.5181	−6.3879	3.5203
1.2000	−0.8985	2.3587	−5.9797	3.2572
1.2500	−0.8051	2.2156	−5.6087	3.0207
1.3000	−0.7208	2.0866	−5.2718	2.8074
1.3500	−0.6443	1.9700	−4.9658	2.6143
1.4000	−0.5746	1.8643	−4.6874	2.4388
1.4500	−0.5108	1.7680	−4.4338	2.2788
1.5000	−0.4524	1.6801	−4.2022	2.1325
1.5500	−0.3987	1.5995	−3.9903	1.9982
1.6000	−0.3491	1.5256	−3.7959	1.8746
1.6500	−0.3032	1.4574	−3.6173	1.7606
1.7000	−0.2606	1.3945	−3.4527	1.6551
1.7500	−0.2210	1.3362	−3.3007	1.5572
1.8000	−0.1842	1.2821	−3.1600	1.4662
1.8500	−0.1497	1.2317	−3.0296	1.3814
1.9000	−0.1175	1.1848	−2.9084	1.3023
1.9500	−0.0873	1.1409	−2.7955	1.2282
2.0000	−0.0590	1.0998	−2.6902	1.1588

These values are graphically represented by

```
Plot[Evaluate[{bstar, bqq1, bqq2, bqq1 − bc} /. {n → 8, σ → 1, Λ → 1}],
    {T, .3, 2}, AxesLabel → {"T*", "B*"}, PlotStyle →
      {RGBColor[0.996109, 0, 0], RGBColor[0.500008, 0.996109, 0],
        RGBColor[0.500008, 0, 0.250004], RGBColor[0, 0, 0.996109]},
    Prolog → {Text["B*", {0.304995, 40.1747}], Text["B1*",
          {0.466965, −26.9214}], Text["B2*", {0.578254, 32.1998}],
        Text["B1*−Bc*", {0.901671, 12.5162}]},
    PlotLabel → "σ=1, Λ=1"];
```



The first and second derivatives of B^* with respect to T^* are of practical importance.

5.7.4 Shape Dependence of the Boyle Temperature

Stogryn and Hirschfelder [5.16] showed that the SVC can be separated into a bound state, a meta-stable state, and a continuum state contribution. For the (12-6)-LJ potential, they gave the temperature dependence of these contributions in tabular form.

At low temperatures, the average energies of the colliding molecules are of the order of the energy of the well depth. The molecule spends much time in the bound region of the molecular potential. Mutual attraction of the molecules results in a decrease of pressure, and the SVC is negative.

At high temperatures, corresponding to high energies compared to the well depth, the main contribution comes from the repulsive branch of the potential. Repulsion increases the pressure and SVS becomes negative.

From the above-mentioned investigation of the SVC for the (12-6)-LJ potential follows that the SVC for the bound states and the meta-stable states remains positive, whereas the contribution by the continuum states becomes negative and equals the bound state and metastable state contribution at the Boyle temperature leading to $B(T) = 0$. The shape dependence of the SVC on the exponent n is shown in Figure 5.7.24.

```
ns = {4.0, 4.5, 5, 5.5, 6, 6.5, 7, 7.5};
Plot[Evaluate[Map[bc /. n → # &, ns]],
  {T, 1, 300}, AxesLabel → {"T*", "B*q0"},
  PlotStyle → RGBColor[0, 0, 0.996109],
  TextStyle → {FontFamily → "Arial",
    FontSize → 15, FontWeight → "Bold"},
  AxesStyle → {Thickness[0.005]},
  Prolog → {Text["n=4.0", {161.363, 0.171682}],
    Text["n=7.5", {161.363, 0.6}]}];
```

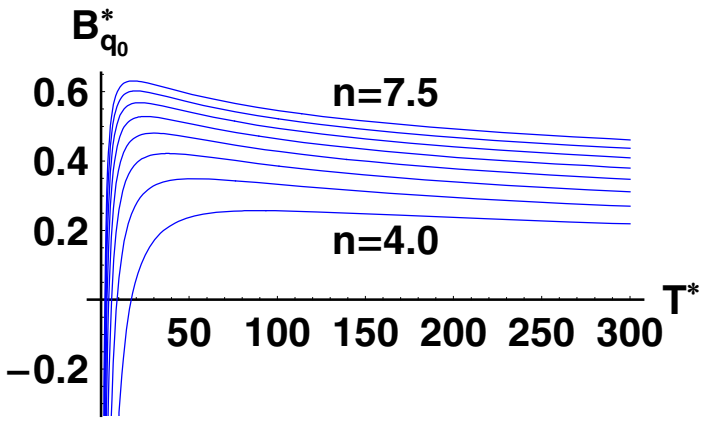


Figure 5.7.24. The scaled SVC for different potential orders n.

The Boyle temperatures are calculated by solving the defining equation $B(T_B) = 0$. The solution is carried out by the function FindRoot[]:

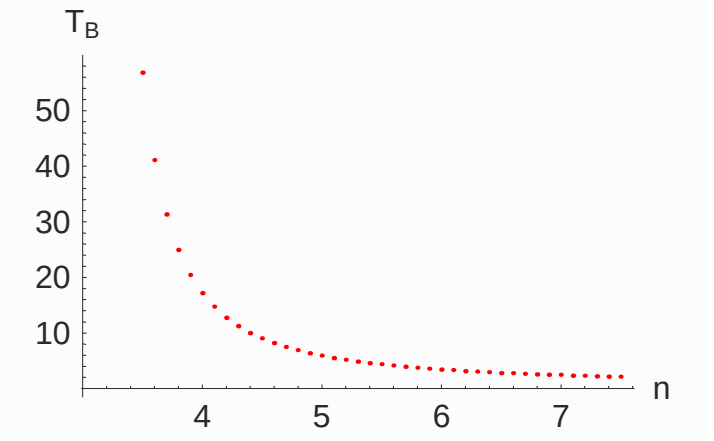
```
BoyleT = (FindRoot[#1 == 0, {T, 10}] &) /@
      (bc /. n → #1 &) /@ Table[i, {i, 3.1, 7.5, .1}];
```

The following table collects the Boyle-temperatures for different values of n:

```
tabBoyle = {Table[i, {i, 3.1, 7.5, .1}], T /. BoyleT}^T;
tb  = Prepend[tabBoyle, {"n", "T_B"}];
```

A graphical representation of these numerical values is given in the following plot:

```
ListPlot[tabBoyle, AxesLabel → {"n", "T_B"},
  PlotStyle → RGBColor[0.996109, 0, 0]];
```



The result is that the Boyle temperaure is a decreasing function which has a singularity at $n = 3$. For n values much larger than 3, the Boyle temperature approaches zero.

At often unphysically high temperatures, the molecules collide with such high energies that they interpenetrate each other. They behave as if they had a smaller volume so that B(T) goes through a maximum. This is shown in Figure 5.7.24. To determine the change of this maximum by altering the potential order, we determine the coresponding temperature values by

```
Tmax = (FindRoot[#1 == 0, {T, 20}] &) /@
    (∂T bc /. n → #1 &) /@ Table[i, {i, 3.2, 7.5, .1}];
```

FindRoot::lstol :
The line search decreased the step size to within tolerance specified by AccuracyGoal and PrecisionGoal but was unable to find a sufficient decrease in the merit function. You may need more than MachinePrecision digits of working precision to meet these tolerances. More…

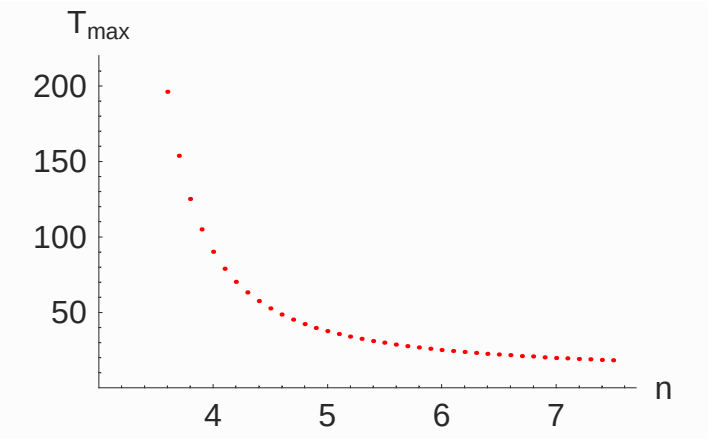
FindRoot::lstol :
The line search decreased the step size to within tolerance specified by AccuracyGoal and PrecisionGoal but was unable to find a sufficient decrease in the merit function. You may need more than MachinePrecision digits of working precision to meet these tolerances. More…

These maximum temperatures are collected in the following table. For He, this maximum was observed experimentally near 200 K.

```
tabTmax =
 Transpose[{Table[i, {i, 3.2, 7.5, .1}], T /. Tmax}];
tT = Prepend[tabTmax, {"n", "TB"}];
```

The graphical representation of these data is given in the following plot:

```
ListPlot[tabTmax, AxesLabel → {"n", "Tmax"},
  PlotStyle → RGBColor[0.996109, 0, 0],
  PlotRange → {{3, 7.7}, {0, 220}}];
```



5.7.5 The High-Temperature Partition Function for Diatomic Molecules

The partition function of a diatomic molecule is important for many applications from astrophysics to reaction kinetics. In courses on physical chemistry, it is treated in the harmonic oscillator approximation – rigid rotator approximation, and anharmonicity and rotation – vibration interactions are included in the spirit of the JANAF tables.

It is known from high-temperature chemistry that for accurate thermodynamic functions, bound states from the solution of the rotation–vibration Schrödinger equation of the molecule

$$\frac{\partial^2 \psi_{v\mathrm{J}}}{\partial r^2} - \left(U(r) + \frac{J(J+1)\,\hbar^2}{2\,\mu\,r^2} - E_{v\mathrm{J}}\right)\psi_{v\mathrm{J}} = 0 \qquad (5.7.119)$$

where $\psi_{v\mathrm{J}}$ are rotation–vibration eigenfunctions, v and J are vibrational and rotational quantum numbers, respectively, μ represents the reduced mass, and $E_{v\mathrm{J}}$, the rotation–vibration eigenvalues, must be calculated. The meta-stable states behind the rotational barrier must be included.

Mies and Julienne [5.18] investigated the statistical thermodynamic of the diatomic molecule using numerical techniques for the exact scattering theory of the SVC.

For the equilibrium reaction

$$X_2 \rightleftarrows 2\,X$$

as an example, they showed that the concentration equilibrium constant K_c can be expressed by the SVC

$$B(T) = -K_c \qquad (5.7.120)$$

As for real molecules and atoms excited, often degenerate electronic states must be included, they defined a generalized SVC by

$$\langle B\rangle = \left(\sum_i B_i(t)\, g_i(x_2)\, e^{-E_{\mathrm{ij}}/(k_B\,T)}\right)\left(\sum_j g_j(x)\, e^{-E_{\mathrm{ij}}/(k_B\,T)}\right),$$

where B_i is the SVC for a molecular state i, g is the electronic degeneracy E_{ij} is the excitation energy, g_j is the electronic degeneracy of the atomic

state j and E_j is its excitation energy. Phair, Birlsi and Holland [5.17] derived the partition function from

$$K_p = \frac{K_c}{RT} = \frac{1}{k_B T} \frac{\frac{q(X_2)}{V}}{\left(\frac{q(X)}{V}\right)^2},$$

with K_p the pressure equilibrium constant, V the volume of the system, $q(X)$ the monomer partition function, and $q(X_2)$ the dimer parition function.

As $q(X)$ depends only on mass, temperature, volume, and electronic degeneracy g, the diatomic partition function for the bound state can be written

$$q(X_2) = -B(T)\left(\frac{2\pi m_x k_B T}{h^2}\right)^3 \frac{V}{N_A} g_0^2(X)\, e^{-D_0/(k_B T)}, \tag{5.7.121}$$

where D_0 is the spectroscopic dissociation energy of the ground electronic state of the molecule X_2, if the energy zero is taken as the lowest vibrational state (one can take as well D_e as energy zero). If we insert the analytical results for $B(T)$ for the $(2n-n)$-potential derived above a closed-form representation of a realistic partition function including rotation–vibration coupling, anharmonicity up the disociation limit, meta-stable states behind the rotational barrier, and the continuum or scattering states.

From the diatomic partition function, the molecular thermodynamic functions can be calculated by standard methods. Phais et al. [5.17] gave explicite formulas for

$$H_T^0 - H_0^0 = RT\left(4 + \frac{B_1^*}{B^*} + \ldots\right) \tag{5.7.122}$$

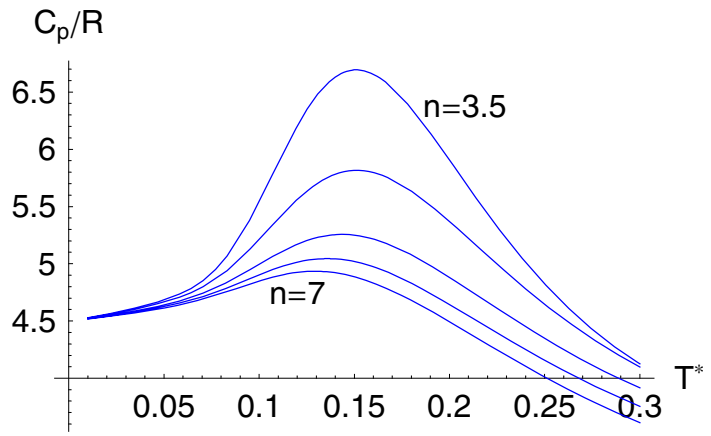
$$C_p^0 = R\left(4 + 2\frac{B_1^*}{B^*} + \frac{B_2^*}{B^*} - \left(\frac{B_1^*}{B^*}\right)^2\right). \tag{5.7.123}$$

Equation (5.7.122) scaled by R reads

```
Cp = -(b1/bc)^2 + 2 b1/bc + b2/bc + 4;
```

A graphical representation of this function for differet values of n is given next.

```
pl1 =
  Plot[Evaluate[Map[Cp /. n → # &, {3.5, 4, 5, 6, 7}]],
   {T, .01, .3}, AxesLabel → {"T*", "Cp/R"},
   PlotStyle → RGBColor[0, 0, 0.996109],
   Prolog → {Text["n=3.5", {0.208115, 6.37933}],
     Text["n=7", {0.120138, 4.74692}]}];
```



Phair et al. used a five-parameter Hulburt–Hirschfelder potential in their numerical calculations for B_{q_0}. The following set of data is taken from their article representing the scaled C_p values [5.17].

```
data = {{0.0174, 4.5}, {0.0384, 4.57}, {0.0522, 4.69},
   {0.0696, 4.75}, {0.0869, 5.035}, {0.1043, 5.52},
   {0.1217, 5.99}, {0.1304, 6.14}, {0.139, 6.20},
   {0.147, 6.19}, {0.156, 6.09}, {0.174, 5.75},
   {0.191, 5.28}, {0.208, 4.83}, {0.217, 4.62}};
```

A combination of our symbolic calculations and their numerical results demonstrates a qualitative agreement. The results are shown in the Figure 5.7.25.

```
Show[
  {pl1, ListPlot[data, DisplayFunction -> Identity]},
  DisplayFunction -> $DisplayFunction];
```

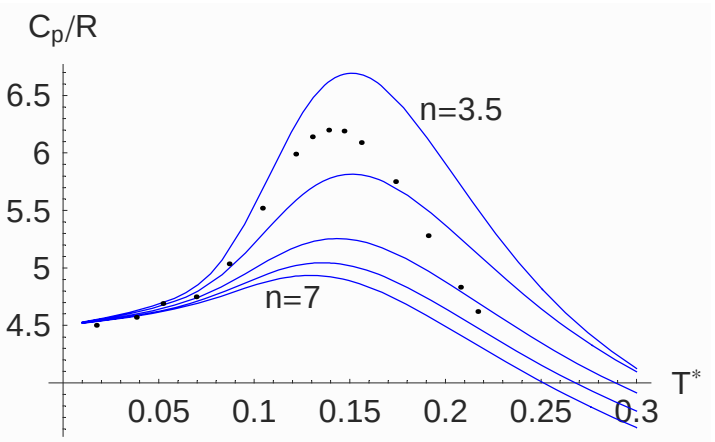


Figure 5.7.25. Shape dependence of the ``dissociation" maximum of the heat capacity C_p. The points denoted by dots are for N_2 teken from Phair et al, using the five parameter Hulburt–Hirschfelder potential in the numerical calculation of B_{q_0} and its temperature derivatives.

5.8 Exercises

1. Examine the spectrum of the eigenvalues for a potential well with different depths. Give a graphical representation of the eigenvalues depending on different depths.

2. Determine the wave functions for different eigenvalues for the potential well by using the methods discussed in Section 5.3.

3. Check the relation $|a|^2 + |b|^2 = 1$ for the anharmonic oscillator.

4. Reexamine the Pöschel–Teller problem and study the expectation values $\langle x^n \rangle$ given by

$$\langle x^n \rangle = \int \psi^* x^n \psi \, dx$$

for different values of n.

5. Plot the radial part of the wave function of the hydrogen atom for different quantum numbers n and l. Examine the influence of the charge Z.

6. Create a graphical representation of the f orbital for the europium atom.

5.9 Packages and Programs

5.9.1 Package QuantumWell

This package serves to examine a one-dimensional quantum dot.

```
BeginPackage["QuantumWell`"];

Clear[PsiSym, PsiASym, Spectrum];

PsiSym::usage =
  "PsiSym[x_,k_,a_] determines the symmetric
    eigenfunction for a potential well of depth -
    V0. The input parameter k fixes the energy and
    2a the width of the well. PsiSym is useful for
    a numerical representation of eigenfunctions.";

PsiASym::usage =
  "PsiASym[x_,k_,a_] determines the antisymmetric
    eigenfunction for a potential well of depth -
    V0. The input parameter k fixes the energy and 2
    a the width of the well. PsiASym is useful for
    a numerical representation of eigenfunctions.";

Spectrum::usage =
  "Spectrum[V0_,a_] calculates the negative
    eigenvalues in a potential well. V0 is the
    potential depth and 2a the width of the
    well. The eigenvalues are returend as a list
    and are available in the variables lsym and
    lasym as replacement rules. The corresponding
    plots of eigenfunctions are stored in the
    variables Plsym and Plasym. The determining
    equation for the eigenvalues is plotted.";
```

```
(*---define global variables---*)

Plsym::usage = "Variables containing the
    symmetric plots of the eigenfunctions.";
Plasym::usage = "Variables containing the
    antisymmetric plots of the eigenfunctions.";
lsym::usage = "List of symmetric eigenvalues.";
lasym::usage = "List of antisymmetric eigenvalues.";
k : usage = "Eigenvalue.";

Begin["`Private`"];

(*---symmetric eigenfunctions---*)
PsiSym[x_, k_, a_] := Block[{},
   (*---define the three domains of solution---*)
   Which[-Infinity < x && x < -a,
    1 / Sqrt[a Exp[-2 a k Tan[k a]] (1 + 1 / (k Tan[k a] a) +
          k Tan[k a] / (k^2 a) + (k Tan[k a])^2 / k^2)]
     Exp[k Tan[k a] x], -a ≤ x && x ≤ a,
    1 / Sqrt[a Exp[-2 a k Tan[k a]] (1 + 1 / (k Tan[k a] a) +
          k Tan[k a] / (k^2 a) + (k Tan[k a])^2 / k^2)]
     Exp[-k Tan[k a] a] Cos[k x] / Cos[k a],
    a < x && x < Infinity, 1 / Sqrt[a Exp[-2 a k Tan[k a]]
        (1 + 1 / (k Tan[k a] a) + k Tan[k a] / (k^2 a) +
          (k Tan[k a])^2 / k^2)] Exp[-k Tan[k a] x]]];

(*---antisymmetric eigenfunctions---*)
PsiASym[x_, k_, a_] := Block[{},
   (*---define the three domains of solution---*)
   Which[-Infinity < x && x < -a,
    -1 / Sqrt[a Exp[-2 a (-k Cot[k a])]
        (1 + 1 / (-k Cot[k a] a) + (-k Cot[k a]) / (k^2 a) +
          (-k Cot[k a])^2 / k^2)] Exp[(-k Cot[k a]) x],
    -a ≤ x && x ≤ a, 1 / Sqrt[a Exp[-2 a (-k Cot[k a])]
        (1 + 1 / (-k Cot[k a] a) + (-k Cot[k a]) / (k^2 a) +
          (-k Cot[k a])^2 / k^2)]
     Exp[-(-k Cot[k a]) a] Sin[k x] / Sin[k a],
    a < x && x < Infinity,
    1 / Sqrt[a Exp[-2 a (-k Cot[k a])]
        (1 + 1 / (-k Cot[k a] a) + (-k Cot[k a]) / (k^2 a) +
          (-k Cot[k a])^2 / k^2)]
```

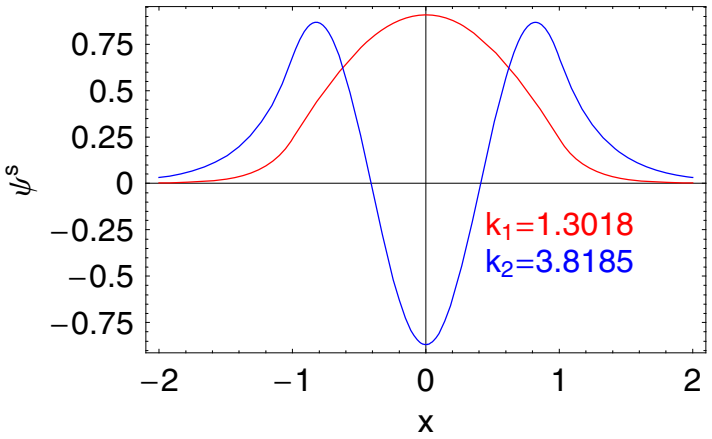
```
     Exp[-(-k Cot[k a]) x]]];

(*---determination of the eigenvalues;
 plot of the eigenfunctions---*)
Spectrum[V0_, a_] :=
  Block[{hbar = 1, m = 1, ymax, C2, rhs, lhssym,
    lhsasym, equatsym, equatasym, kmax, nsym,
    nasym, resultsym, resultasym}, (*---define
      constants and the eigenvalue equation---*)
   C2 = 2 m V0 a^2 / (hbar^2);
   rhs = Tan[k a];
   lhssym = Sqrt[C2 - (k a)^2] / (k a);
   lhsasym = -k a / Sqrt[C2 - (k a)^2];
   equatsym = Sqrt[C2 - (k a)^2] / (k a) - Tan[k a];
   equatasym = -k a / Sqrt[C2 - (k a)^2] - Tan[k a];
   (*---location of the singularity in k---*)
   kmax = Sqrt[C2 / a^2];
   (*---number of symmetric eigenvalues---*)
   nsym = Floor[N[kmax / (Pi / a)]] + 1;
   (*---number of antisymmetric eigenvalues---*)
   nasym = Floor[N[(kmax - Pi / (2 a)) / (Pi / a)]] + 1;
   (*---initialize the
      lists for the eigenvalues---*)
   lsym = {};
   lasym = {};
   (*---calculate the symetric eigenvalues---*)
   Do[resultsym = Chop[FindRoot[
       equatsym == 0, {k, 0.1 + (Pi / a) (i - 1)}]];
    AppendTo[lsym, resultsym], {i, 1, nsym}];
   (*---Chop[] replaces
      small numbers (<10^(-10)) by 0---*)
    (*---calculate the antisymmetric eigenvalues---*)
   Do[resultasym = Chop[FindRoot[equatasym == 0,
       {k, Pi / (2 a) + 0.1 + (Pi / a) (i - 1)}]];
    AppendTo[lasym, resultasym], {i, 1, nasym}];
   (*---plot the eigenvalue equation---*)
   ymax = lhssym 1.5 /. lsym[[1]];
   Off[Plot::plnr];
   Plot[{rhs, lhssym, lhsasym},
    {k, 0.01, 3 kmax / 2}, PlotRange → {-ymax, ymax},
    Prolog → Thickness[0.001],
    PlotStyle → {RGBColor[1, 0, 0], Dashing[{}],
```

```
      Dashing[{1 / 60}]}, AxesLabel → {"k", " "}];
   On[Plot::plnr];
   (*---plot the symmetric eigenfunctions---*)
   Do[k1 = k /. lsym[[i]];
    Plsym[i] = Plot[PsiSym[x, k1, a], {x, -2 a, 2 a},
      AxesLabel → {"x", "\!\(ψ\^s\)\n"}, PlotLabel ->
       "     \!\(k\_i\)= " <> ToString[k1],
      Frame -> True, PlotRange → All,
      Prolog → Thickness[0.001], PlotStyle →
       {Dashing[{1 / (i 20)}]}], {i, 1, nsym}];
   (*---plot the antisymmetric eigenfunctions---*)
   Do[k1 = k /. lasym[[i]];
    Plasym[i] = Plot[PsiASym[x, k1, a], {x, -2 a, 2 a},
      AxesLabel → {"x", "\!\(ψ\^a\)\n"},
      PlotLabel -> "    \!\(k\_i\)= " <> ToString[k1],
      Frame -> True, PlotRange → All,
      Prolog → Thickness[0.001], PlotStyle →
       {Dashing[{1 / (i 20)}]}], {i, 1, nasym}];
   (*---print the eigenvalues---*)
   Print["  "];
   Print[" ----  eigenvalues ---- "];
   Print["  "];
   Do[k1 = k /. lsym[[i]];
    If[i ≤ nasym, k2 = k /. lasym[[i]], k2 = "---"];
    Print[" sym eigenvalue k",
     i, " = ", k1, " asym eigenvalue k",
     i, " = ", k2], {i, 1, nsym}]];

End[];
EndPackage[];
```

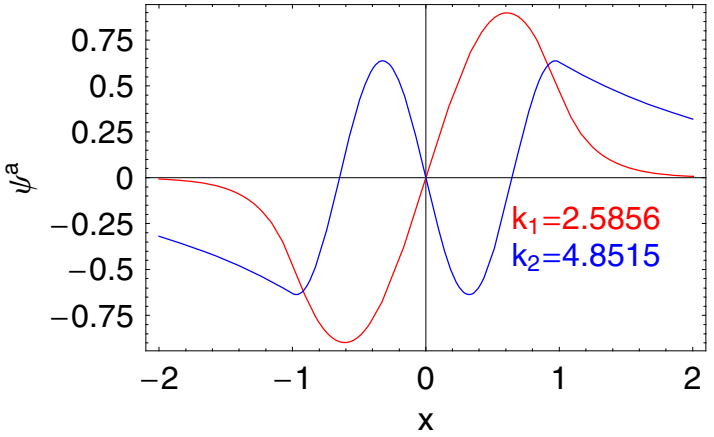
```
Set::patset : Warning: k : usage in assignment
   k : usage = Eigenvalue. represents a named pattern;
   use symbol::tag to represent a message name. More…
```

Here are some tests of the symmetric and antisymmetric wave function.

```
Plot[{PsiSym[x, 1.30183, 1],
   PsiSym[x, 3.818578969739773`, 1]}, {x, -2., 2},
  Frame -> True, FrameLabel -> {"x", "ψˢ"},
  PlotStyle -> {RGBColor[1, 0, 0], RGBColor[0, 0, 1]},
  Prolog -> {{RGBColor[1, 0, 0], Text["k1=1.3018",
      {1., -0.220252}]}, {RGBColor[0, 0, 1],
    Text["k2=3.8185", {1., -0.420252}]}}];
```



```
Plot[{PsiASym[x, 2.5856031391976373`, 1],
   PsiASym[x, 4.851591489119471`, 1]}, {x, -2., 2},
  Frame -> True, FrameLabel -> {"x", "ψ^a"},
  PlotStyle -> {RGBColor[1, 0, 0], RGBColor[0, 0, 1]},
  Prolog -> {{RGBColor[1, 0, 0], Text["k1=2.5856",
      {1.2, -0.220252}]}, {RGBColor[0, 0, 1],
    Text["k2=4.8515", {1.2, -0.420252}]}}];
```



5.9.2 Package HarmonicOscillator

The package HarmonicOscillator provides functions to represent eigenfunctions of the harmonic oscillator.

```
BeginPackage["HarmonicOscillator`"];

Clear[a, across, Psi, wcl, wqm];

Psi::usage =
  "Psi[xi_,n_] represents the eigenfunction
    of the harmonic oscillator. The first
    argument xi is the spatial coordinate. The
    second argument n fixes the eigenstate.";

wcl::usage =
  "wcl[xi_,n_] calculates the classical probability
```

```
    of locating the particle in the harmonic
    potential. The first argument xi is the
    spatial coordinate while n determines
    the energy given as eigenvalue.";

wqm::usage =
  "wqm[xi_,n_] calculates the quantum mechanical
    probability for an eigenvalue state n. The first
    argument represents the spatial coordinate.";

a::usage = "a[psi_, xi_:x] annihilation operator for
    eigenfunction psi. The second argument specifies
    the independent variable of the function psi.";

across::usage =
  "across[psi_, xi_:x] creation operator for
    eigenfunction psi. The second argument
    specifies the independent variable of psi.";

x::usage;

Begin["`Private`"];

(*---eigenfunctions of the harmonic oscillator---*)
Psi[xi_, n_] :=
  HermiteH[n, xi] Exp[-xi^2 / 2] / Sqrt[n! 2^n Sqrt[Pi]];

\!\(\(ψ\_n_[ξ_] := Psi[ξ, n];\)\)

(*---classical probability distribution
   of the harmonic oscillator---*)
wcl[xi_, n_] := 1 / (Sqrt[2 n + 1]
     Sqrt[1 - (xi / Sqrt[2 n + 1])^2] 2 Pi);

(*---quantummechanical probability
   distribution of the harmonic oscillator---*)
wqm[xi_, n_] := Psi[xi, n]^2;

(*---annihilation operator---*)
a[psi_, xi_: x] := (xi psi + D[psi, xi]) / Sqrt[2];

(*---creation operator---*)
```

```
across[psi_, xi_: x] := (xi psi - D[psi, xi]) / Sqrt[2];

End[];
EndPackage[];
```

5.9.3 Package AnharmonicOscillator

The package AnharmonicOscillator serves to determine the properties of the Pöschel–Teller problem.

```
BeginPackage["AnharmonicOscillator`"];
Clear[AsymptoticPT, PlotPT, PoeschelTeller];
PoeschelTeller::usage =
  "PoeschelTeller[x_, n_, indexN_] calculates the
    eigenfunction of the Poeschel Teller potential
    for discrete eigenvalues.N determines the
    depth of the potential V0 Sech[x] by V0=N(
    N+1).n fixes the state where 0 < n <= N.";

w1a::usage = "The variable contains the
    analytic expression for the asymptotic
    approximation for x -> -Infinity.";

w2a::usage = "The variable contains the
    analytic expression for the asymptotic
    approximation for x -> Infinity.";

Transmission::usage =
  "Variable containing the expression
    for the transmission coefficient. The
    independent variables are N and k.";

Reflection::usage =
  "Variable containing the reflection coefficient.
    The independent variables are N and k.";

AsymptoticPT::usage =
  "AsymptoticPT[indexN_,kin_] determines the
    asymptotic approximation for |x|->Infinity
    for the continuous case of eigenvalues in
```

```
    a Poeschel Teller potential. The function
    yields an analytic expression for|b(k)|^2.
    The variables Transmission and Reflection
    contain the expressionsfor the transmission
    and the reflection coefficients. w1a and
    w2a contain the approximations for x->
    -Infinity and x->Infinity, respectively.";

PlotPT::usage =
  "PlotPT[kini_,kend_,type_] gives a graphical
    representation of the reflection or transmission
    coefficient depending on the value of
    thevariable type. If type is set to
    the string r the reflection coefficient
    isplotted. If type is set to the
    transmission coefficient is represented.
    This function creates 5 different curves.";

Begin["`Private`"];

(*---define the eigenfunctions---*)
PoeschelTeller[x_, n_Integer, indexN_Integer] :=
  Block[{norm, integrand, xi},
   If[n ≤ indexN && n > 0, (*---eigenfunctions are
       the associated Legendre polynomials---*)
     integrand = LegendreP[indexN, n, xi];
    (*---calculate the normalization constant---*)
    norm =
     Integrate[integrand^2 / (1 - xi^2), {xi, -1, 1}];
    (*---normalize and simplify the functions---*)
    integrand = integrand / Sqrt[norm] /. xi → Tanh[x];
    Simplify[integrand],
    (*---error conditions---*)
    If[indexN < n,
     Print["---  wrong argument! use n > N"]];
    If[n < 0, Print["---  wrong
        argument! use n < 0"]]]];

(*---asymptotic expansion---*)
AsymptoticPT[indexN_, kin_] :=
  Block[{k, rule1, rule2, wavefkt1, wavefkt2,
    asympt1, w1, asymt2, w2, akh, bkh, ak},
```

```
    (*---replacement rules for the parameters---*)
    rule1 = {a → 1 / 2 - I k + (1 / 4 + V0) ^ (1 / 2),
      b → 1 / 2 - I k - (1 / 4 + V0) ^ (1 / 2), c → 1 - I k};
    rule2 = {V0 → indexN (1 + indexN)};
    wavefkt1 = ak ((1 - xi ^ 2) / 4) ^ (-I k / 2);
    wavefkt2 = Hypergeometric2F1[a, b, c, (1 + xi) / 2];
    (*---asymptotic expansion for x→-Infinity,
     equation 5.5 .63---*)
    asymt1 = Series[wavefkt2, {xi, -1, 0}];
    w1 = wavefkt1 Normal[asymt1] /. rule1;
    w1 = w1 /. rule2;
    w1 = w1 /. xi → Tanh[x];
    w1 = Simplify[w1];
    w1 = w1 /. Sech[x] → 2 Exp[-x];
    w1a = PowerExpand[w1];
    (*---asymptotic expansion for x→
      Infinity by equation 5.5 .65-5.5 .68---*)
    asymt2 = Series[wavefkt2, {xi, 1, 1}];
    (*---invert substitution---*)
    w2 = wavefkt1 Normal[asymt2] /. xi → Tanh[x];
    (*---eliminate higher terms---*)
    w2 = Expand[Simplify[w2 /. -1 + Tanh[x] → 0]];
    (*---asymptotic
       behavior for Sech[] and Tanh[]---*)
    w2 = w2 /. {Sech[x] → Exp[-x],
       1 - Tanh[x] → Exp[-2 x]};
    w2 = w2 /. rule1;
    w2 = w2 /. rule2;
    w2a = PowerExpand[w2];
    (*---determine the
       coefficients a[k] and b[k]---*)
    akh = Coefficient[w2a, Exp[-I k x]] /. ak → 1;
    bkh = Coefficient[w2a, Exp[I k x]] /. ak → 1 / akh;
    (*---calculate the transmission and
       reflection coefficient---*)Transmission =
     1 / (akh Conjugate[akh]) /. k → kin;
    Reflection = bkh Conjugate[bkh] /. k → kin;
    {Transmission, Reflection}];

 (*---graphical representation of the
    reflection and transmission coefficient---*)
 PlotPT[kini_, kend_, type_] :=
```

```
  Block[{k0 = kini, ke = kend, p, t1, label},
   t1 = Transpose[Table[AsymptoticPT[indexxN, kk],
      {kk, k0, ke, (ke - k0) / 5}]];
   If[type == "r", p = t1[[2]];
    label = "|b|^2",
       p = t1[[1]];
       label = "|a|^2"];
   Plot[Chop[p], {indexxN, 1, 2}, AxesLabel →
     {"N", label}, Prolog → Thickness[0.001]]];
End[];
EndPackage[];
```

5.9.4 Package CentralField

CentralField is a package allowing you to represent the eigenfunctions for problems with a central field.

```
BeginPackage["CentralField`"];
Clear[Radial, Angle, AnglePlot, Orbital];

Radial::usage = "Radial[ro_, n_, l_,
    Z_] calculates the radial representation
of the eigenfunctions for an electron in
    the Coulomb potential. The numbers
n and l are the quantum numbers for the
    energy and the angular momentum
operator. Z specifies the number of
    charges in the nucleus. The radial
distance between the center and the
    electron is given by ro.";

Angle::usage = "Angle[theta_, phi_,
    l_, m_] calculates the angular part of
the wave function for an electron in the
    Coulomb potential. The numbers L
and m denote the quantum numbers for the
    angular momentum operator. Theta
and phi are the angles in the spherical
    coordinate system.";
```

```
Orbital::usage = "Orbital[theta_,
    phi_,l_,m_,type_String]  calculates the
superposition of two wave functions for
    the quantum numbers m_l = +m and
m_l = -m. The variable type allows the
    creation of the sum or the difference
of the wave functions. The string values
    of type are either plus or minus.";

AnglePlot::usage =
  "AnglePlot[pl_,theta_,phi_] gives a graphical
representation of the function contained
    in pl. The range of representation
is Pi <= phi < 5 Pi/2 and 0 < theta <
    Pi. Theta is measured with
respect to the vertical axis. This function
    is useful for ploting the orbitals
and the angular part of the eigenfunction.";

(*---define global variables---*)

theta::usage;
phi::usage;
ro::usage;
n::usage;
l::usage;
m::usage;

Begin["`Private`"];

(*---radial part of the eigenfunctions
   in the Coulomb potential---*)

Radial[ro_, n_, l_, Z_] := Block[{norm, hnl},
   (*---normalization---*)
   norm = (Sqrt[(n + l) ! / (2 n (n - l - 1) !)]
       ((2 Z) / n) ^ (l + 3 / 2)) / (2 l + 1) !;
   (*---definition of the wave function---*)
   hnl = norm ro ^ l Exp[- ((Z ro) / n)]
     Hypergeometric1F1[l + 1 - n, 2 l + 2, (2 Z ro) / n]];
```

```
(*---angular part of the
   eigenfunctions in the Coulomb field---*)

Angle[theta_, phi_, l_, m_] := Block[
   {norm, legendre, x, angle, m1, result}, m1 = Abs[m];
   (*---normalization---*)
   norm = (-1)^m1 Sqrt[
       ((2 l + 1) (l - m1) !) / (2 (l + m1) !)] / Sqrt[2 Pi];
   (*---eigenfunctions---*)
   legendre =
    Sin[theta]^m1 D[LegendreP[l, x], {x, m1}];
   legendre = legendre /. x → Cos[theta];
   (*---consider the cases m>0 and m<0---*)
   If[m ≥ 0, angle = Exp[I m phi],
    angle = (-1)^m1 Exp[-(I m1 phi)]];
   (*---normalized eigenfunction---*)
   result = norm legendre angle];

(*--create orbitals---*)

Orbital[theta_, phi_, l_, m_, type_String] :=
  Block[{norm, m1, rule, wave, wave2},
   m1 = Abs[m];
   (*---replacement rule
      for the exponential function---*)
   rule = {E^(Complex[0, a_] (x_.)) →
      Cos[x Abs[a]] + I Sign[a] Sin[x Abs[a]]};
   (*---distinguish different cases---*)
   If[m1 ≥ 1,
    If[type == "plus",
     (*---sum of the
        wave functions for a fixed m---*)
     wave = Expand[Angle[theta, phi, l, m1] +
         Angle[theta, phi, l, -m1] /. rule],
     (*---difference of the wave function
        for a fixed m---*)
     wave = Expand[Angle[theta, phi, l, m1] -
         Angle[theta, phi, l, -m1] /. rule]];
    wave2 = wave^2;
    (*---normalization of the superposition---*)
    norm =
     Integrate[wave2, {phi, 0, 2 Pi}, {theta, 0, Pi}];
```

```
    wave2 = Expand[wave2 / Abs[norm]],
    wave = Angle[theta, phi, l, m1]^2]];

(*---graphical representation
   of the angular part---*)

AnglePlot[pl_, theta_, phi_] := Block[{},
   (*---theta is measured with respect to
      the vertical axis---*)ParametricPlot3D[
    {-pl Sin[theta] Cos[phi], -pl Sin[theta] Sin[phi],
     pl Cos[theta]}, {phi, Pi, 5 Pi / 2}, {theta, 0, Pi},
    PlotRange → All, PlotPoints → {40, 40}]];
End[];
EndPackage[];
```

6
General Relativity

6.1 Introduction

This chapter collects a few examples discussed in connection with general relativity. The examples are the bending of a light beam in a gravitational field, Einstein's field equations, the Schwarzschild solution, and the Reissner–Nordstrom solution for a charged mass point. The given examples are prominent examples to exemplify the use and techniques of symbolic computing in the field of general relativity.

General relativity is a widespread theory which today incorporates different disciplines such as experimental test, exact solutions, formalism of general relativity, gravitational radiation, gravitational collapses, initial value problem, alternative theories, unified field theories, quantum gravity, and many others. In our discussions, we will restrict ourselves to exact solutions and modeling of gravitational effects. These branches were originally created by different people. The core contributions were made by Einstein (see Figure 6.1.1) who based his theory on Riemann's theory on curved space. The specific contributions of original and successful

solutions for different problems originating from Einstein's input were given by Friedman, Schwarzschild, and others. The derivation of solutions and applications to specific problems continuous until the present.

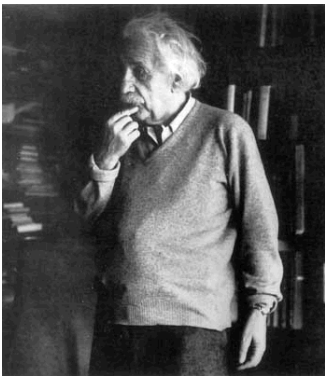


Figure 6.1.1. Albert Einstein: born March 14, 1879; died April 18, 1955.

Riemann (see Figure 6.1.2) by himself was never involved in the creation of general relativity but contributed a theory that supports efficiently and successfully to describe the phenomenon of gravitation in a contemporary way. When Riemann established his theory on curved space, the traditional theory by Newton was used to describe gravitation phenomena. Newtonian theory provides an outstanding example for a theory which governed many centuries of science. At the end of the 19th century, it was becoming increasingly clear that something was fundamentally wrong with the current theories, but there was considerable reluctance to make any fundamental changes to them. Instead, a number of artificial assumptions required the genius of Einstein to overthrow the prejudices of centuries and demonstrate in a number of simple thought experiments that some of the most cherished assumptions of Newtonian theory were untenable. This was the beginning of relativity. Relativity developed in different stages. First, with Einstein's brilliant papers in 1905, the special theory of relativity was introduced. Later, on in the 1920s, Einstein developed general relativity.

Figure 6.1.2. Georg Friedrich Bernhard Riemann: born September 17, 1826; died June 20, 1866.

Out of the general relativity theory a number of old and new questions arose. One of these questions was the movement of the perihelion of Mercury. It was an outstanding question of how these movement could be described in a consistent way. However, Newton's theory allows a way of explaining how the movement can be motivated, but it remained an open problem until Einstein's general relativity theory was established. Since then, many old questions could be attacked. However, there also occurred new ones due to the mathematics by Riemann. A famous solution of the spherical Einstein equations was given by Schwarzschild (see Figure 6.1.3). He and others realized that the nonlinear Einstein equations are very complicated and allow a wealth of new solutions. This will be one of the subjects in this chapter.

Figure 6.1.3. Karl Schwarzschild: born October 09, 1873; died May 11, 1916.

In Section 6.2 we introduce some notions from general relativity theory. Light bending is discussed in Section 6.3. Einstein's field equations are presented in Section 6.4. The Scharzschild solution and the Reissner Nordstrom solutions are discussed in Sections 6.5 and 6.6.

6.2 The Orbits in General Relativity

From the classical theory of orbital motion we know that a planet in a central force field moves in an ellipse around the center of the planetary system. The orbit of the planet is confined to a plane with fixed orientation. This behavior is in contradiction to the observations made at the turn of the century. From observations of the orbital motion of planets, especially of Mercury, astronomers have discovered that the perihelion of the orbit is rotating. This movement of the perihelion is called perihelion shift. The classical theories of Kepler and Newton do not accurately describe the perihelion shift. The second law of Kepler states that a planet moves in an ellipse around the center of the planetary system. In classical theory, the orbital motion is governed by the conservation of energy and angular momentum. The conservation of angular momentum confines the planet to a plane. Another conserved quantity of Newton's theory is the Lenz vector. The Lenz vector is a vector from the focus to the perihelion that is constant (i.e., in classical theory, the perihelion is at a fixed point in space). In Einstein's general theory of relativity (GR), these assumptions are altered. In GR, the orbits are not closed paths and there exists a perihelion rotation. The actual planetary orbits are rosettes. For these types of orbit, the perihelion rotates slowly around the Sun. The rotation of the orbit results from two effects [6.1]:

1. To calculate the orbit using special relativity, we have to take into account an increase of the mass by

$$m = \frac{m_0}{\sqrt{1-v^2/c^2}}, \tag{6.2.1}$$

where m_0 is the rest mass of the planet, c is the velocity of light, and v is the velocity of the planet in the orbit.

2. The central star produces a gravitational field. According to Einstein's theory, this gravitational field is related to an energy density which, in turn, is directly connected with a mass density. The additional mass density of the field adds a certain amount of field strength to the strength of the Sun.

Both effects are relevant in explaining the perihelion shift of a planet. In the following, we consider the second effect in more detail [6.1]. The Sun of our solar system possesses a much larger mass than the accompanying

planets, which means that we can locate the origin of the coordinate system in the Sun. Since the orbit is confined to a plane in space (conservation of angular momentum), we can use plane polar coordinates (r, ϕ) to describe the motion of the planets. In GR, the distance between two points is not simply given by the radial distance r but is also a function of the radial coordinate. If we denote time by t, we can express the line element ds^2 in space-time in the Schwarzschild metric by

$$ds^2 = c^2 \left(1 - \frac{R^s}{r}\right) dt^2 - \frac{dr^2}{1 - R^s/r} - r^2 \, d\phi^2, \tag{6.2.2}$$

```
ds2 = c^2 (1 - Rs/r) Dt[t]^2 - 1/(1 - Rs/r) Dt[r]^2 - r^2 Dt[ϕ]^2
```

$$-\frac{(dr)^2}{1 - \frac{\mathrm{Rs}}{r}} + c^2 \left(1 - \frac{\mathrm{Rs}}{r}\right)(dt)^2 - r^2 \, (d\phi)^2$$

[6.2], where c denotes the speed of light and $R^s = 2\,G\,m/c^2$ is the Schwarzschild radius of the gravitational field. G is the gravitational constant and m is the mass of the Sun. The Lagrangian of the motion in this metric is given by

$$L = c^2 \left(1 - \frac{R^s}{r}\right) t'^2 - \frac{r'^2}{1 - R^s/r} - \frac{1}{2} \, r^2 \, \phi'^2, \tag{6.2.3}$$

```
schwarzschildLagrangian =
 c^2 (1 - Rs/r[s]) (∂_s t[s])^2 - (∂_s r[s])^2/(1 - Rs/r[s]) - 1/2 r[s]^2 (∂_s ϕ[s])^2
```

$$-\frac{r'(s)^2}{1 - \frac{\mathrm{Rs}}{r(s)}} + c^2 \left(1 - \frac{\mathrm{Rs}}{r(s)}\right) t'(s)^2 - \frac{1}{2} \, r(s)^2 \, \phi'(s)^2$$

where the primes denote differentiation with respect to the line element s. Since GR is a geometrically based theory, the orbits of the theory are derivable by a variational principle. Fermat's principle, which governs the path of a light beam, is an example from optics. In GR, the orbits follow from the extremum of the action as determined by the Lagrangian. In close analogy to our considerations in Section 2.6, the equations of motion of GR follow from the Euler–Lagrange equations in the form

$$\frac{d}{ds}\left(\frac{\partial L}{\partial r'}\right) - \frac{\partial L}{\partial r} = 0, \tag{6.2.4}$$

$$\frac{d}{ds}\left(\frac{\partial L}{\partial \phi'}\right) - \frac{\partial L}{\partial \phi} = 0, \tag{6.2.5}$$

$$\frac{d}{ds}\left(\frac{\partial L}{\partial t'}\right) - \frac{\partial L}{\partial t} = 0. \tag{6.2.6}$$

```
swEquations =
 EulerLagrange[schwarzschildLagrangian, {r, ϕ, t}, s];
swEquations // TableForm
```

$$-\frac{\text{Rs}\, r'(s)^2}{\left(1-\frac{\text{Rs}}{r(s)}\right)^2 r(s)^2} + \frac{c^2\, \text{Rs}\, t'(s)^2}{r(s)^2} - r(s)\,\phi'(s)^2 + \frac{2\, r''(s)}{1-\frac{\text{Rs}}{r(s)}} == 0$$

$$\phi''(s)\, r(s)^2 + 2\, r'(s)\,\phi'(s)\, r(s) == 0$$

$$-\frac{2\,\text{Rs}\, r'(s)\, t'(s)\, c^2}{r(s)^2} + \frac{2\,\text{Rs}\, t''(s)\, c^2}{r(s)} - 2\, t''(s)\, c^2 == 0$$

Unlike the classical theory of variation, here we consider time t as a dependent variable. Using Eq. (6.2.3), Eqs. (6.2.5) and (6.2.6) lead to angular momentum l and energy conservation:

$$\frac{\partial L}{\partial \phi'} = \text{const.} = l, \tag{6.2.7}$$

$$\frac{\partial L}{\partial t'} = \text{const.} = E_0 \tag{6.2.8}$$

or

$$r^2\,\phi' = l = \frac{1}{\sqrt{\beta}}, \tag{6.2.9}$$

$$c^2\left(1 - \frac{R^s}{r}\right) t' = E_0 = -\frac{k^2}{c^2\sqrt{\beta}}, \tag{6.2.10}$$

```
angularMomentum =
 Map[Integrate[#, s] &, {swEquations[[2, 1]]}][[1]] == 1/Sqrt[β]
```

$$r(s)^2\,\phi'(s) == \frac{1}{\sqrt{\beta}}$$

```
energy =
 MapAt[Integrate[#, s] &, {swEquations[[3, 1]]}, 1][[1]] ==
  -k^2/(c^2 Sqrt[β])
```

$$-\frac{2\,c^2\,(r(s)-\mathrm{Rs})\,t'(s)}{r(s)} == -\frac{k^2}{c^2\sqrt{\beta}}$$

where k and β are appropriate constants for the following considerations. Using the conserved quantities in the expression of the line element (6.2.2), we get

$$\frac{dr^2}{1-\frac{R^s}{r}} = \left(-\beta\,r^4 + \frac{k^2\,r^4}{c^2(1-R^s/r)} - r^2\right) d\phi^2. \tag{6.2.11}$$

Substituting $u = 1/r$ simplifies the equation of the orbit to

$$\left(\frac{d\,u}{d\,\phi}\right)^2 = \frac{k^2}{c^2} - (1 - R^s\,u)\,(\beta + u^2). \tag{6.2.12}$$

This exact equation is usually solved by using the perturbation theory, which approximates the solution for a certain range [6.3, 6.4]. In Section 6.8.2, the code is given using the solution steps to solve Eq. (6.2.12). The package implements the essential steps. Since the equation consists of a polynomial of third order in u, the solution of Eq. (6.2.12) is expressible by elliptic functions. To see how this occurs, we carry out the necessary transformation

$$u = \frac{4\,U}{R^s} + \frac{1}{3\,R^s} \tag{6.2.13}$$

and substitute it into Eq. (6.2.12). The resulting differential equation is the defining equation for the Weierstrass function $\mathcal{P}(z)$:

$$\left(\frac{d\,U}{d\,\phi}\right)^2 = 4\,U^3 - g_2\,U - g_3. \tag{6.2.14}$$

However, *Mathematica* can deliver a preliminary version of this solution by

```
DSolve[(∂ϕ U[ϕ])^2 == 4 U[ϕ]^3 - g2 U[ϕ] - g3, U, ϕ]
```

Solve::tdep : The equations appear to involve the variables to be solved for in an essentially non−algebraic way. More…

Solve::tdep : The equations appear to involve the variables to be solved for in an essentially non−algebraic way. More…

$$\left\{\text{Solve}\left[\left(2\,F\left(\sin^{-1}\left(\sqrt{\left(\text{Root}[4\,\#1^3 - g2\,\#1 - g3\,\&,\,3] - U(\phi)\right)\Big/\left(\text{Root}[4\,\#1^3 - g2\,\#1 - g3\,\&,\,3] - \text{Root}[4\,\#1^3 - g2\,\#1 - g3\,\&,\,2]\right)}\right)\middle|\,\frac{\text{Root}[4\,\#1^3 - g2\,\#1 - g3\,\&,\,2] - \text{Root}[4\,\#1^3 - g2\,\#1 - g3\,\&,\,3]}{\text{Root}[4\,\#1^3 - g2\,\#1 - g3\,\&,\,1] - \text{Root}[4\,\#1^3 - g2\,\#1 - g3\,\&,\,3]}\right)\right.\right.$$

$$\sqrt{\frac{U(\phi) - \text{Root}[4\,\#1^3 - g2\,\#1 - g3\,\&,\,1]}{\text{Root}[4\,\#1^3 - g2\,\#1 - g3\,\&,\,3] - \text{Root}[4\,\#1^3 - g2\,\#1 - g3\,\&,\,1]}}$$

$$\sqrt{\frac{U(\phi) - \text{Root}[4\,\#1^3 - g2\,\#1 - g3\,\&,\,2]}{\text{Root}[4\,\#1^3 - g2\,\#1 - g3\,\&,\,3] - \text{Root}[4\,\#1^3 - g2\,\#1 - g3\,\&,\,2]}}$$

$$\left.\left.(U(\phi) - \text{Root}[4\,\#1^3 - g2\,\#1 - g3\,\&,\,3])\right)\Big/\left(\sqrt{\frac{U(\phi) - \text{Root}[4\,\#1^3 - g2\,\#1 - g3\,\&,\,3]}{\text{Root}[4\,\#1^3 - g2\,\#1 - g3\,\&,\,2] - \text{Root}[4\,\#1^3 - g2\,\#1 - g3\,\&,\,3]}}\,\sqrt{4\,U(\phi)^3 - g2\,U(\phi) - g3}\right) == c_1 - \phi,\,U(\phi)\right],$$

$$\text{Solve}\left[\left(2\,F\left(\sin^{-1}\left(\sqrt{\left(\text{Root}[4\,\#1^3 - g2\,\#1 - g3\,\&,\,3] - U(\phi)\right)\Big/\left(\text{Root}[4\,\#1^3 - g2\,\#1 - g3\,\&,\,3] - \right.\right.\right.\right.$$

$$\text{Root}[4\,\#1^3 - \text{g2}\,\#1 - \text{g3}\,\&,\, 2]))\Big|$$

$$\frac{\text{Root}[4\,\#1^3 - \text{g2}\,\#1 - \text{g3}\,\&,\, 2] - \text{Root}[4\,\#1^3 - \text{g2}\,\#1 - \text{g3}\,\&,\, 3]}{\text{Root}[4\,\#1^3 - \text{g2}\,\#1 - \text{g3}\,\&,\, 1] - \text{Root}[4\,\#1^3 - \text{g2}\,\#1 - \text{g3}\,\&,\, 3]}$$

$$\Bigg)$$

$$\sqrt{\frac{U(\phi) - \text{Root}[4\,\#1^3 - \text{g2}\,\#1 - \text{g3}\,\&,\, 1]}{\text{Root}[4\,\#1^3 - \text{g2}\,\#1 - \text{g3}\,\&,\, 3] - \text{Root}[4\,\#1^3 - \text{g2}\,\#1 - \text{g3}\,\&,\, 1]}}$$

$$\sqrt{\frac{U(\phi) - \text{Root}[4\,\#1^3 - \text{g2}\,\#1 - \text{g3}\,\&,\, 2]}{\text{Root}[4\,\#1^3 - \text{g2}\,\#1 - \text{g3}\,\&,\, 3] - \text{Root}[4\,\#1^3 - \text{g2}\,\#1 - \text{g3}\,\&,\, 2]}}$$

$$(U(\phi) - \text{Root}[4\,\#1^3 - \text{g2}\,\#1 - \text{g3}\,\&,\, 3])\Bigg)\Bigg/$$

$$\left(\sqrt{\frac{U(\phi) - \text{Root}[4\,\#1^3 - \text{g2}\,\#1 - \text{g3}\,\&,\, 3]}{\text{Root}[4\,\#1^3 - \text{g2}\,\#1 - \text{g3}\,\&,\, 2] - \text{Root}[4\,\#1^3 - \text{g2}\,\#1 - \text{g3}\,\&,\, 3]}}\right.$$

$$\left.\sqrt{4\,U(\phi)^3 - \text{g2}\,U(\phi) - \text{g3}}\right) == \phi + c_1,\, U(\phi)\Big]\Big\}$$

where

$$g_2 = \frac{1}{12} - \frac{(R^s)^2\,\beta}{4}, \tag{6.2.15}$$

$$g_3 = \frac{1}{216} - \frac{(R^s)^2\,\beta}{24} - \frac{(R^s)^2\,k^2}{16\,c^2}. \tag{6.2.16}$$

The solution of U is thus

$$U = \mathcal{P}(\phi + C;\, g_2,\, g_3), \tag{6.2.17}$$

where C denotes the integration constant. The orbits are now represented by the coordinates r and ϕ as:

$$r(\phi) = \frac{3\,R^s}{1+12\,\mathcal{P}(\phi+C;g_2,g_3)}. \tag{6.2.18}$$

6.2.1 Quasielliptic Orbits

If g_2 and g_3 are real and the discriminant $\Delta = g_2^3 - 27\,g_3^2 > 0$ we find three real roots of the characteristic polynomial $4\,x^3 - g_2\,x - g_3 = 0$ which we call e_1, e_2 and e_3. The roots of the characteristic polynomial can be arranged in the order $e_2 < e_3 < e_1$. Using the roots and the expressions g_1 and g_2, we can express the periods ω_1 and ω_2 of the Weierstrass function by

$$\omega_1 = \int_{e_1}^{\infty} \frac{dx}{\sqrt{4\,x^3 - g_2\,x - g_3}} \tag{6.2.19}$$

and

$$\omega_2 = i \int_{-\infty}^{e^2} \frac{dx}{\sqrt{4\,x^3 - g_2\,x - g_3}}. \tag{6.2.20}$$

The first period ω_1 is a real and the second period ω_2 is an imaginary number. ω_2 is the period of the angle ϕ. If we introduce a third frequency ω_3, the equation of the orbit (6.2.18) is expressible in the form

$$r(\phi) = \frac{3\,R^s}{1 + 12\,\mathcal{P}(\phi + \omega_3; g_2, g_3)}. \tag{6.2.21}$$

By introducing ω_3, we are able to suppress the singularity of the Weierstrass function at $z = 0$. The correct specification of the orbit is made by the choice of the locations of the perihelion and the aphelion. Choosing the coordinate system so that the perihelion is reached at $\phi = 0$, we get from Eq. (6.2.21)

$$r(0) = \frac{3\,R^s}{1 + 12\,\mathcal{P}(-\omega_3)} = \frac{3\,R^s}{1 + 12\,e_3}, \tag{6.2.22}$$

$$\frac{d\,r^{-1}}{d\,\phi} = 0 \tag{6.2.23}$$

and

$$\frac{d^2\,r^{-1}}{d\,\phi^2} < 0. \tag{6.2.24}$$

Once the planet has approached the aphelion, it has traced one-half of the total orbit. This point of the orbit is characterized by the angle $\phi = \omega_1$. The radial coordinate at this point is expressed by

$$r(\omega_1) = \frac{3\,R^s}{1 + 12\,\mathcal{P}(\omega_1 - \omega_3)} = \frac{3\,R^s}{1 + 12\,\mathcal{P}(\omega_2)} = \frac{3\,R^s}{1 + 12\,e_2}, \tag{6.2.25}$$

$$\frac{d\, r^{-1}}{d\, \phi} = 0, \tag{6.2.26}$$

and

$$\frac{d^2\, r^{-1}}{d\, \phi^2} > 0. \tag{6.2.27}$$

The relations (6.2.25) and (6.2.27) are correct if the condition $\frac{1}{12} + e_2 > 0$ is satisfied. This condition is equivalent to the relation $c^2\, \beta > k^2$, relating the parameters of the Weierstrass function to the physical parameters of the path. The radial coordinate of the orbit varies between the limits of the perihelion and the aphelion measured from the origin of the coordinate system. The two extremal values of the orbit are

$$r_P = \frac{3\, R^s}{1+12\, e_3}, \tag{6.2.28}$$

$$r_A = \frac{3\, R^s}{1+12\, e_2}. \tag{6.2.29}$$

The planet is thus confined between two circles with radii r_P and r_A. The path itself is an open orbit in the form of a rosette (see Figure 6.2.4, where only the path is shown). The orbit in Figure 6.2.4 is similar to the classical orbit of Kepler's theory. Unlike the classical orbit, the GR shows shifts of the perihelion and the aphelion. From the classical theory of planet motion, we know that the difference of phase between two complete rotations is given by $\phi = 2\, \pi$. Within GR the difference in the angle is exactly $2\, \omega_1$. The shift in the perihelion is thus determined by

$$\Delta \phi^P = 2\, (\pi - \omega_1). \tag{6.2.30}$$

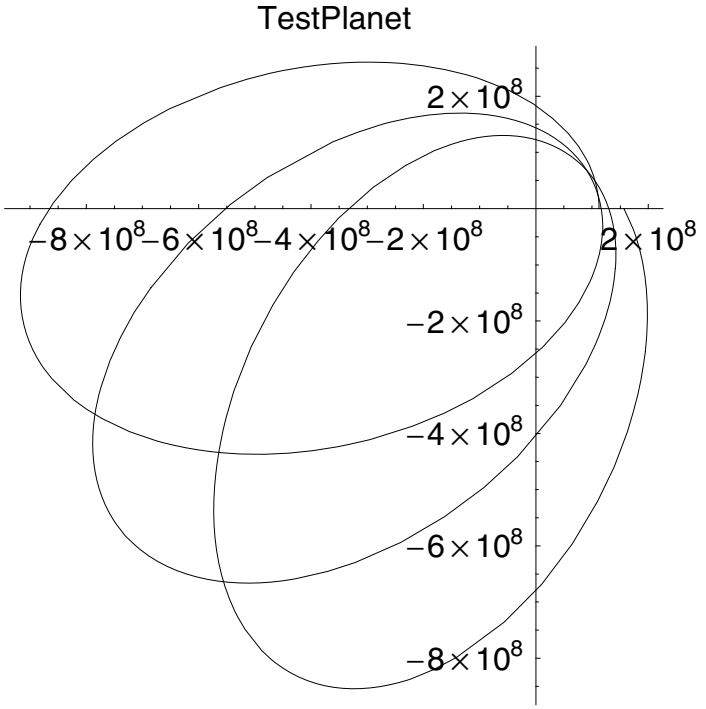


Figure 6.2.4. Perihelion shift for a system of planets with $m = 5.6369 \times 10^{33}$ kg, $a = 5.2325 \times 10^8$ m and eccentricity e=0.61713. The numeric value of the perihelion shift is calculated to be $\Delta\Phi^P = 90122.8"$.

The perihelion shift in the solar system is very small and its experimental observation is very difficult. However, the calculation of Eq. (6.2.30) needs to be precise in order to determine the exact numerical value of the perihelion shift. To calculate the shift using the Weierstrass function, we need an absolute accuracy of 10^{-8} in the values for $\mathcal{P}(z)$. In a graphical representation of the Mercury orbit for example, the shift is invisible. The observed and calculated shift for Mercury is 43.1" for 415 cycles (approximately one century).

The perihelion and the aphelion are determined by relation (6.2.28). The locations of the perihelion and the aphelion are usually given by the classical parameters: the major semiaxis a and eccentricity e. If we combine both parameters of GR and classical theory, we get the relations for r_P and r_A:

$$r_P = \frac{p}{1+e}, \tag{6.2.31}$$

$$r_P = \frac{p}{1-e}, \tag{6.2.32}$$

where $p = b^2/a$ and $e = \sqrt{a^2 - b^2}\,/a$. Having determined the extreme points of the orbit, we know the roots of the Weierstrass function $\mathcal{P}$: e_2 and e_3 from relation (6.2.28). The roots are given by

$$e_2 = -\frac{1}{12}\left(1 - \frac{3R^s}{r_A}\right), \tag{6.2.33}$$

$$e_3 = -\frac{1}{12}\left(1 - \frac{3R^s}{r_P}\right). \tag{6.2.34}$$

In terms of the orbit parameters, we find

$$e_2 = -\frac{1}{12}\left(1 - \frac{3R^s a}{b^2}\left(1 - \frac{\sqrt{a^2-b^2}}{a}\right)\right), \tag{6.2.35}$$

$$e_3 = -\frac{1}{12}\left(1 - \frac{3R^s a}{b^2}\left(1 + \frac{\sqrt{a^2-b^2}}{a}\right)\right). \tag{6.2.36}$$

The roots of the $\mathcal{P}$ function have to satisfy the relations

$$e_1 + e_2 + e_3 = 0, \tag{6.2.37}$$

$$2\,(e_1^2 + e_2^2 + e_3^2) = g_2, \tag{6.2.38}$$

$$4\,e_1\,e_2\,e_3 = g_3. \tag{6.2.39}$$

Here, the root e_1 becomes

$$e_1 = \frac{1}{6}\left(1 - \frac{3\,\mathrm{aR}^s}{b^2}\right). \tag{6.2.40}$$

The quantities g_2 and g_3 are determined by expressions (6.2.15) and (6.2.16) and satisfy relations (6.2.38) and (6.2.39). We are now able to determine the energy E_0 and the angular momentum l from the orbital parameters from Eq. (6.2.9) and (6.2.10). The angular momentum and the energy can be represented by

$$l = \frac{R^s}{2\sqrt{\frac{1}{12} - g_2}}, \tag{6.2.41}$$

$$E_0 = -\frac{2}{c}\sqrt{\frac{\frac{1}{54} - \frac{1}{6}\,g_2 - g_3}{\frac{1}{12} - g_2}}\,. \tag{6.2.42}$$

One problem with using the exact solution theory is the determination of the angles ω_1 and ω_2 when calculating the perihelion shift with *Mathematica*. As mentioned earlier, we need a high degree of accuracy in our calculation to find the right value for $\Delta\phi$. If we do the calculations by simply integrating Eqs. (6.2.19) and (6.2.20), we have a singularity at one of the endpoints of the integration interval. Since we have no convergent representation of the integral, the results are very crude. However, we know from the theory of the Weierstrass functions that the periods are

expressible by complete elliptic integrals of the first kind. Using the properties of the elliptic integrals, we can overcome the inaccurate numerical integration of *Mathematica*:

$$\omega_1 = \frac{K(m)}{\sqrt{e_1 - e_2}}, \tag{6.2.43}$$

$$\omega_2 = i\,\frac{K'(m)}{\sqrt{e_1 - e_2}} = i\,\frac{K(1-m)}{\sqrt{e_1 - e_2}}, \tag{6.2.44}$$

where the module m is given by $m = (e_3 - e_2)/(e_1 - e_2)$, the roots of the Weierstrass function.

The above considerations are collected in the *Mathematica* package PerihelionShift. An example of the application of **PerihelionShift`** is given next. Let us first check the contents of the database for the planets

```
Planets["List"]
```

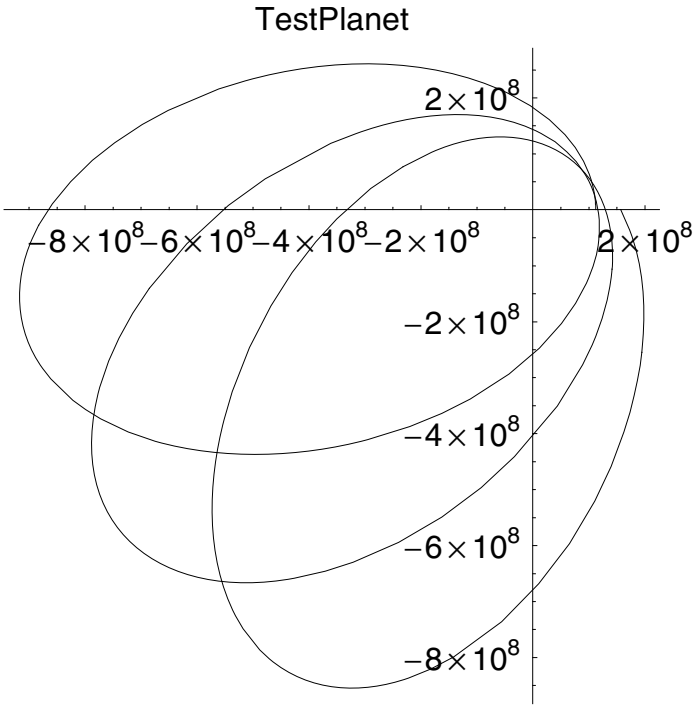
planet	mean radius	eccentricity	mass
Mercury	5.791×10^{10}	0.206	1.993×10^{30}
Venus	1.082×10^{11}	0.007	1.993×10^{30}
Earth	1.497×10^{11}	0.017	1.993×10^{30}
Icarus	1.610×10^{11}	0.827	1.993×10^{30}
Mars	2.228×10^{11}	0.093	1.993×10^{30}
Ceres	4.136×10^{11}	0.076	1.993×10^{30}
Jupiter	7.780×10^{11}	0.048	1.993×10^{30}
Saturn	1.427×10^{12}	0.056	1.993×10^{30}
Uranus	2.870×10^{12}	0.047	1.993×10^{30}
Neptune	4.496×10^{12}	0.009	1.993×10^{30}
Pluto	5.910×10^{12}	0.250	1.993×10^{30}
PSR1916	7.020×10^{8}	0.617	5.637×10^{30}
TestPlanet	5.233×10^{8}	0.617	5.637×10^{33}

As result, we get a table containing 13 objects. The last planet is incorporated to visualize the perihelion shift in a plot. This shift can be calculated and visualized by

```
Orbit["TestPlanet"];
```

TestPlanet	
mass	.56369414099999999e34
minor axes	323780558.91557515
major axes	523270000.00000006
eccentricity	.61713130000000005

Perihelion shift = 90122.8 arcs



6.2.2 Asymptotic Circles

In this subsection, we discuss a limiting case of GR orbits that is closely related to the classical orbits of the Kepler theory. We assume that the constants k and β are such that the discriminant Δ vanishes. For this case, two of the roots e_1, e_2, and e_3 are equal. If we denote the common root by e, the remaining root takes the value $-2\,e$. For $e > 0$, the solution of the orbit equation (6.2.18) is

$$r(\phi) = \frac{3\,R^s \cosh(n\,\phi)}{1-8\,n^2}, \tag{6.2.45}$$

where $n^2 = 3\,e$. This solution results in an apogee with $\phi = 0$, provided that $8\,n^2 < 1$. This restriction is equivalent to the condition $(R^s)^2\,\beta > \frac{1}{4}$.

If ϕ increases, the orbit of the planet spirals down to a circle of asymptotic radius

$$r = \frac{3\,R^s}{1+4\,n^2}. \tag{6.2.46}$$

This radius is smaller than the initial distance of the planet from the center of the planetary system (see Figure 6.2.5). If we choose n so that the relation $0 < n^2 < \frac{1}{8}$ is satisfied, the radius of the asymptotic circle lies between the limits $3\,R^s$ and $2\,R^s$. The orbit for such cases is obtained by function **D0Orbit[]** defined in the package **PerihelionShift`**. An example for the application of this function to the test planet shows the following line:

```
D0Orbit["TestPlanet", 3 π];
```

TestPlanet	
mass	.56369414099999999e34
minor axes	323780558.91557515
major axes	523270000.00000006
eccentricity	.61713130000000005

Perihelion shift = 90122.8 arcs

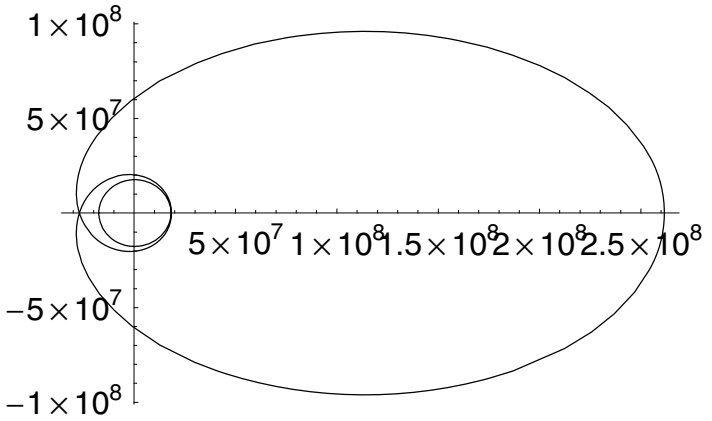


Figure 6.2.5. Orbit for a test planet with Δ=0.

6.3 Light Bending in the Gravitational Field

Einstein's general theory of relativity predicts that a light ray is bent in a gravitational field. The corresponding equation of motion follows from the null geodesic condition $ds^2 = 0$ [6.2]. We discuss the bending of a light ray in the Schwarzschild metric. The equation of motion is given by

$$u'' + u - \tfrac{3}{2} R^s u^2 = 0, \tag{6.3.47}$$

where $u = 1/r$ and $R^s = 2\,G\,m/c^2$ is the Schwarzschild radius of the mass m. G denotes the gravitational constant and c is the speed of light. Multiplying Eq. (6.3.47) by $u' = d\,u/d\,\phi$ and integrating it with respect to parameter s we get

$$\frac{1}{2}\,u'^2 + \frac{1}{2}\,u^2 - \frac{R^s}{2}\,u^3 = E = \frac{k^2}{c^2}, \tag{6.3.48}$$

where E and k, the energy and the scaled energy, are appropriately chosen constants. The substitution $u = 4\,U/R^s + 1/(3\,R^s)$ transforms equation (6.3.48) to a standard form of differential equations defining the Weierstrass function:

$$\left(\frac{d\,U}{d\,\phi}\right)^2 = 4\,U^3 - g_2\,U - g_3 \tag{6.3.49}$$

with

$$g_2 = \frac{1}{12}, \tag{6.3.50}$$

$$g_3 = \frac{1}{216} - \frac{(R^s)^2\,k^2}{16\,c^2}. \tag{6.3.51}$$

The solution for the variable U is given by

$$U = \mathcal{P}(\phi + C;\, g_2,\, g_3). \tag{6.3.52}$$

The path of the light ray $r(\phi)$ is

$$r(\phi) = \frac{3\,R^s}{1+12\,\mathcal{P}(\phi+C;g_2,g_3)}. \tag{6.3.53}$$

The geometrical locations of the planet and the light rays are given in Figure 6.3.6. Figure 6.3.6 shows that the light ray has a distance R from the planet if the angle $\phi = 0$.

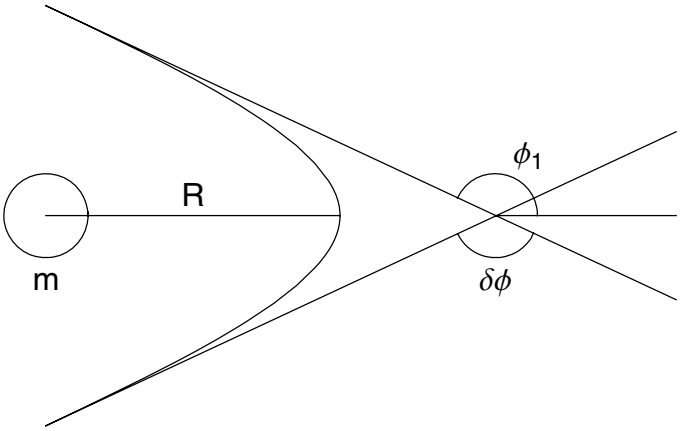


Figure 6.3.6. Geometry of light bending in the neighborhood of a mass m. The deviation angle ϕ_1 follows from the relations $\phi_2 = \pi - \phi_1$ and $\delta\phi = \pi - 2\,\phi_2 = 2\,\phi_1 - \pi$.

When $\phi = \phi_1$, the radius (6.3.53) is infinite. The deviation or bending of the light ray $\delta\phi$ is determined by the relation

$$\delta\phi = 2\,\phi_1 - \pi \tag{6.3.54}$$

(see Figure 6.3.6). Since the Schwarzschild radius R^s and the constant k^2/c^2 are greater than zero, it follows that the discriminant $\Delta = g_2^3 - 27\,g_3^2 > 0$.

The equation $r(\phi = 0) = R$ gives us the first expression for the determination of the roots e_1, e_2, and e_3 of the characteristic polynomial $4\,t^3 - g_2\,t - g_3 = 0$. If we set $\phi = 0$, it follows from Eq. (6.3.53) that

$$r(\phi = 0) = R = \frac{3\,R^s}{1+12\,\mathcal{P}(C;g_2,g_3)}\,. \tag{6.3.55}$$

If we choose the integration constant as the imaginary period of the Weierstrass function $C = -\omega_2$, we get from the condition $\mathcal{P}(-\omega_2) = e_2$ the relation

$$R = \frac{3\,R^s}{1+12\,e_2} \tag{6.3.56}$$

and thus $e_2 = -\frac{1}{12}\,(1 - 3\,R^s/R)$. Since g_2 is fixed to $1/12$ in the light bending problem, the remaining two roots e_1 and e_3 satisfy

$$g_2 = 2\,(e_1^2 + e_2^2 + e_2^3) = \frac{1}{12}\,, \tag{6.3.57}$$

$$e_1 + e_2 + e_3 = 0. \tag{6.3.58}$$

We find, by eliminating $e_3 = -(e_1 + e_2)$, in Eq. (6.3.57), the relation

$$e_1^2 + e_1\, e_2 + e_2^2 - \frac{1}{4}\, g_2 = 0, \tag{6.3.59}$$

which has the solution

$$e_1 = -\frac{1}{2}\, e_2 \pm \frac{\sqrt{3}}{12}\, \sqrt{1 - 36\, e_2^2}\,. \tag{6.3.60}$$

From Eq. (6.3.58), we can derive the solution for e_1 to be

$$e_3 = -(e_1 + e_2) = -\Big(\frac{1}{2}\, e_2 \pm \frac{\sqrt{3}}{12}\, \sqrt{1 - 36\, e_2^2}\Big). \tag{6.3.61}$$

The remaining problem is to find the angle of inclination (i.e., the angle ϕ_1 for which the radius tends to infinity). We can express this condition by

$$r(\phi = \phi_1) = \infty = \frac{3\, R^s}{1 + 12\, \mathcal{P}(\phi_1 - \omega_2; g_2, g_3)}\,. \tag{6.3.62}$$

Equation (6.3.62) is satisfied if

$$\mathcal{P}(\phi_1 - \omega_2;\, g_2,\, g_3) + \frac{1}{12} = 0. \tag{6.3.63}$$

The frequency ω_2 is derived from the roots e_1, e_2, and e_3 and satisfies the relations

$$\omega_2 = \omega + \omega', \tag{6.3.64}$$
$$\omega_1 = \omega, \qquad \text{real}, \tag{6.3.65}$$
$$\omega_3 = \omega', \qquad \text{imaginary}. \tag{6.3.66}$$

In addition, there are two relations for the frequencies ω and ω':

$$\omega = \frac{K(m)}{\sqrt{e_1 - e_3}} \qquad \text{and} \qquad \omega' = i\, \frac{K(1-m)}{\sqrt{e_1 - e_3}}, \tag{6.3.67}$$

where the modulus $m = (e_2 - e_3)/(e_1 - e_3)$. Equation (6.3.63) is only solvable numerically and provides us with the limiting angle ϕ_1. The angle determines the asymptotic direction of the light ray. An example of the bending of a light ray near the surface of the Sun is shown in Figure 6.3.7. The graphical representation of the light bending is created using **Orbit[]**, a function of the package **LightBending`** which is available in Section 6.8.3. The function **Deviation[]**, which is also contained in this package, allows the numerical calculation of the bending angle. The arguments of **Deviation[]** are the mass of the planet and the closest approach of the light ray.

```
Orbit[RadiusOfTheSun, MassOfTheSun];
```

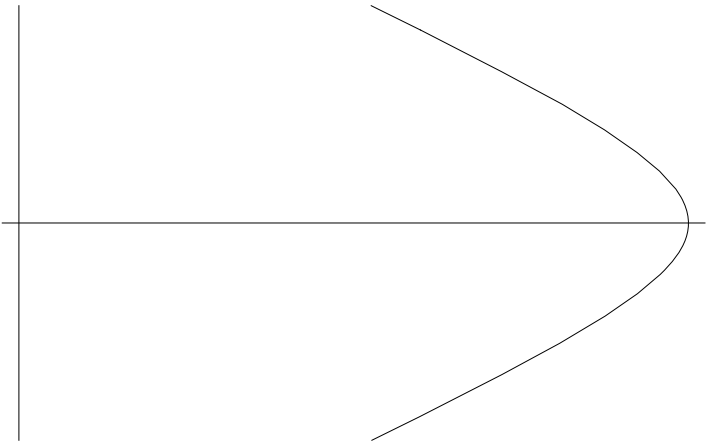


Figure 6.3.7. Path of a light ray in the neighborhood of the sun.

The deviation of a light beam passing the Sun can be determined by

```
Deviation[RadiusOfTheSun, MassOfTheSun]
```

FindRoot::lstol :
The line search decreased the step size to within tolerance specified by AccuracyGoal and PrecisionGoal but was unable to find a sufficient decrease in the merit function. You may need more than 34. digits of working precision to meet these tolerances. More…

Deviation = 1.74416 arcs

$$8.455905338175976 \times 10^{-6}$$

6.4 Einstein's Field Equations (Vacuum Case)

Einstein's theory of gravitation can be described by Riemannian geometry. In Riemannian geometry, space is characterized by its metric. The metric is normally represented by its line element ds^2 or equivalently by the metric tensor which can be read from the line element. The metric tensor allows the calculation of the scalar product of two vectors as well as the equations of motion. Einstein's field equations are the central equations of GR and describe the motion of a particle in space time. Since GR is primarily based on geometry, we have to consider the related metric of the space in addition to the physical problem. For our considerations, we assume that the independent variables in the space are given by

```
IndepVar={t,x,y,z}
```

$\{t, x, y, z\}$

The coordinates are used in the determination of the metric tensor. The function **metric[]** calculates the coefficients of the metric tensor from a given line element. **metric[]** takes the line element ds^2 and a list of coordinates as input variables. The result is the symmetric metric tensor of the underlying space. The following lines determine the metric tensor the comments in the function give a short description of the step performed:

```
metric[lineelement_,independentvars_List]:=Block[
{lenindependent,differentials,diffmatrix,
metricform,varmetric,gh,sum,equation,rule,
varhelp,zeros,zerorule},

(* --- determine the number of independent variables
---*)
    lenindependent = Length[independentvars];

(* --- create the differentials corresponding to
dx,dt .... --- *)
    differentials = Map[Dt,independentvars];

(* --- a matrix of differential products --- *)
```

```
    diffmatrix = Outer[Times,differentials,
                       differentials];

(* --- the general metric form --- *)
    metricform = Array[gh,{lenindependent,
                               lenindependent}];
    varmetric = Variables[metricform];

(* --- built a system of equations to determine
       the elements of the metric              ---*)
    If[Length[metricform] == Length[diffmatrix],
       sum = 0;
       Do[
          Do[
             sum = sum +
                 metricform[[i,j]] diffmatrix[[i,j]],
          {j,1,lenindependent}],
       {i,1,lenindependent}],
     sum = 0
     ];

(* --- construct the metric tensor --- *)
     If[ sum === 0,
         Return[sum],
         sum = sum - lineelement;
         equation = CoefficientList[sum,
                                differentials]==0;
         rule = Solve[equation,varmetric];
         metricform = metricform /. rule;
         varmetric = Variables[metricform];

(* --- replace the nonzero elements --- *)
           varhelp = {};
           Do[
              If[Not[FreeQ[varmetric[[i]],gh]],
                 AppendTo[varhelp,varmetric[[i]] ]
              ],
           {i,1,Length[varmetric]}];
         zeros = Table[0,{Length[varhelp]}];
         SubstRule[x_,y_]:=x->y;
         zerorule = Thread[SubstRule[varhelp,zeros]];
         metricform = Flatten[metricform /.
                                      zerorule,1];

(* --- make the metricform symmetric --- *)
         metricform = Expand[(metricform +
                          Transpose[metricform])/2]
     ];
```

```
    metricform
];
Off[Solve::svars];
Off1[Solve::svars];
```

The application of this function to different examples is demonstrated next.

6.4.1 Examples for Metric Tensors

As a first example, we consider a simple metric of a hypothetical two-dimensional space in x and t coordinates. The *Mathematica* symbol **Dt[x]** expresses the differential dx in line elements.

```
MatrixForm[metric[t x Dt[t]^2 + x Dt[x]^2, {x, t}]]
```

$$\begin{pmatrix} x & 0 \\ 0 & t\,x \end{pmatrix}$$

The result is a (2×2) matrix containing the coefficients of the line element. A simple three-dimensional example is the Euclidean space with the well-known cartesian metric. The corresponding line element is $ds^2 = dx^2 + dy^2 + dz^2$.

In traditional form, we get the metric by

```
metric((dx)^2 + (dy)^2 + (dz)^2, {x, y, z})
```

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

which is the expected result for the metric tensor. We see that **metric[]** extracts the metric tensor from the line element. The information contained in the metric tensor is of some importance in the derivation of the field equations.

The line element or the metric tensor for Euclidean space changes its form if we use a different coordinate system (e.g., the transformation from cartesian coordinates to spherical coordinates). In spherical coordinates, the metric tensor is given by

MatrixForm[metric($(dr)^2 + r^2 (d\theta)^2 + r^2 (d\phi)^2 \sin(\theta)$, {$r$, θ, ϕ})]

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & r^2 & 0 \\ 0 & 0 & r^2 \sin(\theta) \end{pmatrix}$$

where r is the radius and ϕ and θ are the spherical polar angles.

A nontrivial example in three dimensions characterizing a curved space is given by the line element $ds^2 = dr^2 + r^2\, d\theta^2 + dz^2$ in cylindrical coordinates r, ϕ, and z. The corresponding metric tensor is

MatrixForm[metric($(dr)^2 + (dz)^2 + r^2 (d\phi)^2$, {$r$, ϕ, z})]

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & r^2 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

In four dimensions – three space dimensions and one time coordinate – the space corresponding to Euclidean space in three dimensions is the Minkowski space. Euclidean space with cartesian coordinates x, y, and z is extended by an additional time dimension t. Note the sign difference when distinguishing between the time coordinate and the space-time dimensions. The line element in x, y, z, and t is given by $ds^2 = dt^2 - dx^2 - dy^2 - dz^2$ (speed of light equals unity, $c = 1$). The corresponding metric tensor of Minkowski space reads

```
MatrixForm[metric((dt)^2 - (dx)^2 - (dy)^2 - (dz)^2, {t, x, y, z})]
```

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}$$

The Minkowski space is a trivial solution of Einstein's field equations for the vacuum case. A time-independent solution of the field equations with spherical symmetry is the famous Schwarzschild solution. The line element ds^2 in the coordinates t, r, θ, and ϕ is

```
ds2 = -((dθ)^2 + (dϕ)^2 sin^2(θ)) r^2 - (dr)^2/(1 - 2m/r) + (1 - 2m/r) (dt)^2
```

$$(-(d\theta)^2 - (d\phi)^2 \sin^2(\theta))\, r^2 - \frac{(dr)^2}{1 - \frac{2m}{r}} + \left(1 - \frac{2m}{r}\right)(dt)^2$$

The corresponding metric is

```
erg = metric(ds2, {t, r, θ, ϕ}); MatrixForm[erg]
```

$$\begin{pmatrix} 1 - \frac{2m}{r} & 0 & 0 & 0 \\ 0 & \frac{r}{2m-r} & 0 & 0 \\ 0 & 0 & -r^2 & 0 \\ 0 & 0 & 0 & -r^2 \sin^2(\theta) \end{pmatrix}$$

This representation of the line element is a spherically symmetric solution of the vacuum field equations. The timelike coordinate t can be interpreted as the world time. The coordinates θ and ϕ can be identified as the usual angles in spherical coordinates.

The above line element ds^2 resembles the line element in Euclidean space. In the following example, the radial coordinate r is transformed so that we can write the line element in the isotropic form

$ds^2 = G(\rho)\,dt^2 - F(\rho)\,(d\rho^2 + \rho^2\,d\theta^2 + \rho^2\sin^2(\theta)\,d\phi^2)$. The transformation reads $r = \rho(1 + m/(2\,\rho))^2$. The corresponding line element of the metric reads

ds3 = $\dfrac{\left(1 - \frac{m}{2\rho}\right)^2 (dt)^2}{\left(\frac{m}{2\rho} + 1\right)^2} - \left(\dfrac{m}{2\rho} + 1\right)^4 (((d\theta)^2 + (d\phi)^2 \sin^2(\theta))\,\rho^2 + (d\rho)^2)$

$$\frac{\left(1 - \frac{m}{2\rho}\right)^2 (dt)^2}{\left(\frac{m}{2\rho} + 1\right)^2} - \left(\frac{m}{2\rho} + 1\right)^4 (((d\theta)^2 + (d\phi)^2 \sin^2(\theta))\,\rho^2 + (d\rho)^2)$$

and the corresponding metric tensor is

g = metric(ds3, {t, ρ, θ, ϕ})

$$\left\{\left\{\frac{m^2}{(m + 2\,\rho)^2} - \frac{4\,\rho\,m}{(m + 2\,\rho)^2} + \frac{4\,\rho^2}{(m + 2\,\rho)^2}, 0, 0, 0\right\},\right.$$
$$\left\{0, -\frac{m^4}{16\,\rho^4} - \frac{m^3}{2\,\rho^3} - \frac{3\,m^2}{2\,\rho^2} - \frac{2\,m}{\rho} - 1, 0, 0\right\},$$
$$\left\{0, 0, -\frac{m^4}{16\,\rho^2} - \frac{m^3}{2\,\rho} - \frac{3\,m^2}{2} - 2\,\rho\,m - \rho^2, 0\right\},$$
$$\left\{0, 0, 0, -\frac{\text{Sin}[\theta]^2\,m^4}{16\,\rho^2} - \frac{\text{Sin}[\theta]^2\,m^3}{2\,\rho} -\right.$$
$$\left.\left.\frac{3}{2}\,\text{Sin}[\theta]^2\,m^2 - 2\,\rho\,\text{Sin}[\theta]^2\,m - \rho^2\,\text{Sin}[\theta]^2\right\}\right\}$$

Up to now, we have only discussed the line element of the metric and its related metric tensor. To derive the field equations for the vacuum case in GR, we have to introduce other tensors. One of the essential quantities determining the field equations are the Christoffel symbols. These symbols are related to the metric tensor in a straightforward way.

6.4.2 The Christoffel Symbols

Every important relation or equation in a Riemannian space can be expressed in terms of the metric tensor or its partial derivatives. These expressions are often very complex. The Christoffel symbols are important expressions for formulating Einstein's field equations and for expressing the geometric properties of space. The Christoffel symbols contain the inverse of the metric tensor **ginv** and partial derivatives of first order with respect to the coordinates. The Christoffel symbols can be defined by

```
Christoffel[m_, a_, b_, g_, ginv_] := Block[{n},
    Expand[
     Sum[ginv[[m, n]] (D[g[[a, n]], IndepVar[[b]] ] +
            D[g[[b, n]], IndepVar[[a]] ] -
            D[g[[a, b]], IndepVar[[n]] ]),
          {n, 1, Length[g]}] / 2]
     ]
```

In mathematical notation, the function **Christoffel[]** is given by

$$\Gamma^{m}_{a,b} = g^{m\,n}(\partial_b g_{a\,n} + \partial_a g_{b\,n} - \partial_n g_{a\,b}). \tag{6.4.68}$$

Other important tensors needed to formulate the field equations are usually expressed in Christoffel symbols. The Christoffel symbols also appear in equations for metric geodesics (i.e., the equations defining the parameterized curve of a particle moving in space). In the following, we define tensors such as the Riemann tensor, the Ricci tensor, and so forth.

6.4.3 The Riemann Tensor

The curvature tensor, also called the Riemann tensor, is defined in terms of Christoffel symbols by

```
Riemann[a_, b_, c_, d_, g_, ing_] := Block[{},
    Expand[
    D[Christoffel[a, b, d, g, ing], IndepVar[[c]]] -
    D[Christoffel[a, b, c, g, ing], IndepVar[[d]]] +
    Sum[Christoffel[e, b, d, g, ing]
      Christoffel[a, e, c, g, ing],
      {e, 1, Length[g]}] -
    Sum[Christoffel[e, b, c, g, ing]
      Christoffel[a, e, d, g, ing],
      {e, 1, Length[g]}]
            ]
  ]
```

The Riemann tensor describes the geometric properties of the underlying space. A flat space contains a Riemann tensor with zero coefficients.

A contraction of the Riemann tensor delivers the Ricci tensor. The Ricci tensor is a symmetric tensor in the form

```
Ricci[m_, q_, g_, ing_] := Block[{a},
  Expand[
     Sum[Riemann[a, m, a, q, g, ing],
     {a, 1, Length[g]}]]]
```

Another contraction of the Ricci tensor defines the curvature scalar or Ricci scalar:

```
RicciScalar[g_, ing_] := Block[{},
   Expand[Sum[ing[[a, b]] Ricci[a, b, g, ing],
       {a, 1, Length[g]}, {b, 1, Length[g]}]]];
```

Having these tensors available, we can proceed to the derivation of Einstein's field equations.

6.4.4 Einstein's Field Equations

Einstein's vacuum equations are expressed by the Ricci tensor and the Ricci scalar:

```
Einstein[m_, n_, g_, ing_] :=
                                 RicciScalar[g, ing] g[[m, n]]
 Ricci[m, n, g, ing]  -  ------------------------------------
                                          2
```

The function **Einstein[]** gives the left-hand side of the equations and the right-hand side is equal to zero. The derived equations are nonlinear partial differential equations of second order in space and time. In addition to the field equations, there are four side conditions given by the Bianchi identities; these identities are a form of energy conservation:

```
Bianchi[a_, g_, ing_] := Block[ {},
Expand[
Sum[ D[Sum[ ing[[n, m]] Einstein[m, a, g.ing],
  {m, 1, Length[g]}], IndepVar[[n]] ],
       {n, 1, Length[g]}]
+ Sum[ Sum[ Christoffel[n, m, n, g, ing]
 Sum[ ing[[m, l]] Einstein[l, a, g, ing],
  {l, 1, Length[g]}], {m, 1, Length[g]}],
       {n, 1, Length[g]}]
- Sum[ Sum[ Christoffel[n, m, a, g, ing]
 Sum[ ing[[m, l]] Einstein[l, n, g, ing],
  {l, 1, Length[g]}], {m, 1, Length[g]}],
       {n, 1, Length[g]}] ]
];
```

The calculation of the 10 coefficients of the metric tensor g is an incompletely formulated mathematical problem since there are fewer equations than unknowns (6 equations with 10 unknowns). Since the metric tensor is a solution of the field equations, it is apparent that a coordinate transformation does not change the problem. When choosing a coordinate system, we are free to introduce gauge conditions. For example, Gaussian or normal coordinates are often introduced by setting $g_{00} = 1$ and $g_{0a} = 0$ for $a = 1, 2, 3$.

We now examine some examples for which we can use the functions defined above. The first is again the three-dimensional flat cartesian space.

6.4.5 The Cartesian Space

The cartesian space in three dimensions is characterized by the line element

dsc = (d x)^2 + (d y)^2 + (d z)^2

$$(dx)^2 + (dy)^2 + (dz)^2$$

with the independent variables

```
IndepVar = {x, y, z}
```

$$\{x, y, z\}$$

The metric form of this space is given by

***g* = metric(dsc, IndepVar)**

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

The inverse of the metric tensor follows by

```
ing = Inverse[g]
```

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

which is simply the identity matrix. Then we calculate some of the Christoffel symbols to see which of them are not equal to zero.

```
Christoffel[1, 1, 1, g, ing]
```

0

```
Christoffel[1,1,1,g,ing]
```

0

```
Christoffel[1,2,1,g,ing]
```

0

```
Ricci[1,2,g,ing]
```

0

It is trivial to see that all Christoffel symbols of this metric vanish. Consequently, the coefficients of the Riemann tensor vanish, too. This fact is expected because a cartesian space is flat. We now examine the cartesian space in different coordinate systems.

6.4.6 Cartesian Space in Cylindrical Coordinates

The line element of cartesian space with cylindrical coordinates is expressed by

```
IndepVar = {r, ϕ, z}
```

$\{r, \phi, z\}$

```
dscy = (dr)^2 + (dz)^2 + r^2 (dϕ)^2
```

$(dr)^2 + (dz)^2 + r^2 (d\phi)^2$

The metric tensor is given by

```
g = metric[dscy,IndepVar]
```

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & r^2 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

and the inverse of the metric tensor is

```
ing = Inverse[g]
```

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{r^2} & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Contrary to the case of the cartesian coordinate system, the Christoffel symbols do not all vanish.

```
Table[Christoffel[i,j,k,g,ing],{i,1,3},{j,1,3},
{k,1,3}]
```

$$\begin{pmatrix} \{0, 0, 0\} & \{0, -r, 0\} & \{0, 0, 0\} \\ \{0, \frac{1}{r}, 0\} & \{\frac{1}{r}, 0, 0\} & \{0, 0, 0\} \\ \{0, 0, 0\} & \{0, 0, 0\} & \{0, 0, 0\} \end{pmatrix}$$

Nevertheless, the Riemann tensor has to be zero for flat cartesian space in spite of the coordinate transformation:

```
Table[Riemann[a,b,c,d,g,ing],{a,1,3},{b,1,3},
{c,1,3},{d,1,3}]
```

$$\left(\begin{pmatrix} \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} & \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} & \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \\ \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} & \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} & \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \\ \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} & \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} & \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \end{pmatrix}\right)$$

The disappearance of the Riemann tensor in flat cartesian space is independent of the corresponding coordinate system. To illustrate the situation, we next examine the Euclidean space in polar coordinates.

6.4.7 Euclidean Space in Polar Coordinates

With the spherical coordinates

```
IndepVar = {r,θ,ϕ}
```

$\{r, \theta, \phi\}$

the line element and the corresponding metric are given by

```
dscp = (dr)^2 + r^2 (dθ)^2 + r^2 (dϕ)^2 sin^2(θ)
```

$$(dr)^2 + r^2\,(d\theta)^2 + r^2\,(d\phi)^2\,\sin^2(\theta)$$

```
g = metric[dscp,IndepVar]
```

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & r^2 & 0 \\ 0 & 0 & r^2\sin^2(\theta) \end{pmatrix}$$

The inverse metric tensor is

```
ing = Inverse[g]
```

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{r^2} & 0 \\ 0 & 0 & \frac{\csc^2(\theta)}{r^2} \end{pmatrix}$$

The Christoffel symbols read

```
Table[Christoffel[i,j,k,g,ing],{i,1,3},{j,1,3},{k,1,3
}]
```

$$\begin{pmatrix} \{0,0,0\} & \{0,-r,0\} & \{0,0,-r\sin^2(\theta)\} \\ \{0,\frac{1}{r},0\} & \{\frac{1}{r},0,0\} & \{0,0,-\cos(\theta)\sin(\theta)\} \\ \{0,0,\frac{1}{r}\} & \{0,0,\cot(\theta)\} & \{\frac{1}{r},\cot(\theta),0\} \end{pmatrix}$$

As in the previous example, the Christoffel symbols do not vanish and are now even more complicated. However, again, as expected, the coefficients of the Riemann tensor are zero:

```
Simplify[Table[Riemann[a,b,c,d,g,ing],{a,1,3},{b,1,3}
,{c,1,3},{d,1,3}] ]
```

$$\left(\begin{pmatrix}0&0&0\\0&0&0\\0&0&0\end{pmatrix}\begin{pmatrix}0&0&0\\0&0&0\\0&0&0\end{pmatrix}\begin{pmatrix}0&0&0\\0&0&0\\0&0&0\end{pmatrix}\right)$$
$$\left(\begin{pmatrix}0&0&0\\0&0&0\\0&0&0\end{pmatrix}\begin{pmatrix}0&0&0\\0&0&0\\0&0&0\end{pmatrix}\begin{pmatrix}0&0&0\\0&0&0\\0&0&0\end{pmatrix}\right)$$
$$\left(\begin{pmatrix}0&0&0\\0&0&0\\0&0&0\end{pmatrix}\begin{pmatrix}0&0&0\\0&0&0\\0&0&0\end{pmatrix}\begin{pmatrix}0&0&0\\0&0&0\\0&0&0\end{pmatrix}\right)$$

6.5 The Schwarzschild Solution

6.5.1 The Schwarzschild Metric in Eddington–Finkelstein Form

In this section, we discuss a nontrivial solution of Einstein's field equations, the famous Schwarzschild metric given in special coordinate representations. The Schwarzschild solution is a solution of Einstein's field equations with the highest symmetry (i.e., with spherical symmetry).

In this representation, there are, as usual, a timelike variable t, a variable r related to distance, and two angle variables θ and ϕ.

```
IndepVar = {t,r,ϴ,ϕ}
```

$\{t, r, \theta, \phi\}$

According to the Eddington–Finkelstein line element,

$$\mathbf{dss} = -((d\theta)^2 + (d\phi)^2 \sin^2(\theta))\, r^2 - \left(\frac{2m}{r} + 1\right)(dr)^2 + \left(1 - \frac{2m}{r}\right)(dt)^2 - \frac{(4m)\, dt\, dr}{r}$$

$$(-(d\theta)^2 - (d\phi)^2 \sin^2(\theta))\, r^2 - \left(\frac{2m}{r} + 1\right)(dr)^2 + \left(1 - \frac{2m}{r}\right)(dt)^2 - \frac{4m\, dr\, dt}{r}$$

The meaning of r and t is different from the standard Schwarzschild solution. Due to our choice of r, a nondiagonal element between r and t appears. Here, the diagonal elements of r and t are in a more symmetric form. Yet, the metric possesses the required symmetries: spherical symmetry and time independence. This metric is special in that it is regular at point $r = 2m$, whereas the Schwarzschild line element in its standard form is singular at this point. This metric can be interpreted as an analytical extension of the standard form in the region $2m < r < \infty$ to the region $0 < r < \infty$. With the metric tensor

```
g = metric[dss,IndepVar]
```

$$\begin{pmatrix} 1 - \frac{2m}{r} & -\frac{2m}{r} & 0 & 0 \\ -\frac{2m}{r} & -\frac{2m}{r} - 1 & 0 & 0 \\ 0 & 0 & -r^2 & 0 \\ 0 & 0 & 0 & -r^2 \sin^2(\theta) \end{pmatrix}$$

and its inverse

```
ing = Inverse[g]
```

$$\left\{\left\{-\frac{\text{Csc}[\theta]^2\,(-2\,\text{m}\,\text{r}^3\,\text{Sin}[\theta]^2 - \text{r}^4\,\text{Sin}[\theta]^2)}{\text{r}^4}, -\frac{2\,\text{m}}{\text{r}}, 0, 0\right\},\right.$$
$$\left\{-\frac{2\,\text{m}}{\text{r}}, -\frac{\text{Csc}[\theta]^2\,(-2\,\text{m}\,\text{r}^3\,\text{Sin}[\theta]^2 + \text{r}^4\,\text{Sin}[\theta]^2)}{\text{r}^4}, 0, 0\right\},$$
$$\left.\left\{0, 0, -\frac{1}{\text{r}^2}, 0\right\}, \left\{0, 0, 0, -\frac{\text{Csc}[\theta]^2}{\text{r}^2}\right\}\right\}$$

the Christoffel symbols and Ricci tensor are easily calculated.

```
Table[Christoffel[i,j,k,g,ing],{i,1,4},{j,1,4},{k,1,4
}]
```

$$\left\{\left\{\left\{\frac{2\,m^2}{r^3}, \frac{2\,m^2}{r^3}+\frac{m}{r^2}, 0, 0\right\},\right.\right.$$
$$\left\{\frac{2\,m^2}{r^3}+\frac{m}{r^2}, \frac{2\,m^2}{r^3}+\frac{2\,m}{r^2}, 0, 0\right\},$$
$$\{0, 0, -2\,m, 0\}, \{0, 0, 0, -2\,m\,\text{Sin}[\theta]^2\}\},$$
$$\left\{\left\{-\frac{2\,m^2}{r^3}+\frac{m}{r^2}, -\frac{2\,m^2}{r^3}, 0, 0\right\}, \left\{-\frac{2\,m^2}{r^3}, -\frac{2\,m^2}{r^3}-\frac{m}{r^2}, 0, 0\right\},\right.$$
$$\{0, 0, 2\,m-r, 0\}, \{0, 0, 0, 2\,m\,\text{Sin}[\theta]^2-r\,\text{Sin}[\theta]^2\}\},$$
$$\left\{\{0, 0, 0, 0\}, \left\{0, 0, \frac{1}{r}, 0\right\}, \left\{0, \frac{1}{r}, 0, 0\right\},\right.$$
$$\{0, 0, 0, -\text{Cos}[\theta]\,\text{Sin}[\theta]\}\}, \left\{\{0, 0, 0, 0\},\right.$$
$$\left.\left.\left\{0, 0, 0, \frac{1}{r}\right\}, \{0, 0, 0, \text{Cot}[\theta]\}, \left\{0, \frac{1}{r}, \text{Cot}[\theta], 0\right\}\right\}\right\}$$

```
Table[Ricci[i,j,g,ing],{i,1,4},{j,1,4}]
```

$$\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -\cot^2(\theta)+\csc^2(\theta)-1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

With these quantities in hand, we can verify that the form of the Eddington–Finkelstein line element is a solution of Einstein's vacuum field equations:

```
Simplify[ Table[Einstein[a,b,g,ing],{a,1,4},{b,1,4}]
]
```

$$\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

In addition to the field equations, the Bianchi identities are satisfied also.

6.5.2 Dingle's Metric

The metric of Dingle with three space coordinates and one timelike coordinate

```
IndepVar = {t,x,y,z}
```

$$\{t, x, y, z\}$$

is the most general metric in diagonal form.

```
dsd = A1(t, x, y, z) (dt)^2 - B1(t, x, y, z) (dx)^2 -
      C1(t, x, y, z) (dy)^2 - D1(t, x, y, z) (dz)^2
```

$$\mathrm{A1}(t, x, y, z)\,(dt)^2 - \mathrm{B1}(t, x, y, z)\,(dx)^2 - \mathrm{C1}(t, x, y, z)\,(dy)^2 - \mathrm{D1}(t, x, y, z)\,(dz)^2$$

Hence, the metric tensor is a diagonal tensor

```
g = metric[dsd,IndepVar]
```

$$\begin{pmatrix} \mathrm{A1}(t, x, y, z) & 0 & 0 & 0 \\ 0 & -\mathrm{B1}(t, x, y, z) & 0 & 0 \\ 0 & 0 & -\mathrm{C1}(t, x, y, z) & 0 \\ 0 & 0 & 0 & -\mathrm{D1}(t, x, y, z) \end{pmatrix}$$

and so is its inverse

```
ing = Inverse[g]
```

$$\begin{pmatrix} \frac{1}{A1(t,x,y,z)} & 0 & 0 & 0 \\ 0 & -\frac{1}{B1(t,x,y,z)} & 0 & 0 \\ 0 & 0 & -\frac{1}{C1(t,x,y,z)} & 0 \\ 0 & 0 & 0 & -\frac{1}{D1(t,x,y,z)} \end{pmatrix}$$

Due to the form of the metric tensor, the Christoffel symbols are fairly simple expressions.

```
Table[Christoffel[i,j,k,g,ing],{i,1,4},{j,1,4},{k,1,4
}]
```

$$\left\{\left\{\left\{\frac{A1^{(1,0,0,0)}[t, x, y, z]}{2\,A1[t, x, y, z]}, \frac{A1^{(0,1,0,0)}[t, x, y, z]}{2\,A1[t, x, y, z]}, \right.\right.\right.$$
$$\left.\left.\frac{A1^{(0,0,1,0)}[t, x, y, z]}{2\,A1[t, x, y, z]}, \frac{A1^{(0,0,0,1)}[t, x, y, z]}{2\,A1[t, x, y, z]}\right\},$$
$$\left\{\frac{A1^{(0,1,0,0)}[t, x, y, z]}{2\,A1[t, x, y, z]}, \frac{B1^{(1,0,0,0)}[t, x, y, z]}{2\,A1[t, x, y, z]}, 0, 0\right\},$$
$$\left\{\frac{A1^{(0,0,1,0)}[t, x, y, z]}{2\,A1[t, x, y, z]}, 0, \frac{C1^{(1,0,0,0)}[t, x, y, z]}{2\,A1[t, x, y, z]}, 0\right\},$$
$$\left.\left\{\frac{A1^{(0,0,0,1)}[t, x, y, z]}{2\,A1[t, x, y, z]}, 0, 0, \frac{D1^{(1,0,0,0)}[t, x, y, z]}{2\,A1[t, x, y, z]}\right\}\right\},$$
$$\left\{\left\{\frac{A1^{(0,1,0,0)}[t, x, y, z]}{2\,B1[t, x, y, z]}, \frac{B1^{(1,0,0,0)}[t, x, y, z]}{2\,B1[t, x, y, z]}, 0, 0\right\},\right.$$
$$\left\{\frac{B1^{(1,0,0,0)}[t, x, y, z]}{2\,B1[t, x, y, z]}, \frac{B1^{(0,1,0,0)}[t, x, y, z]}{2\,B1[t, x, y, z]}, \right.$$
$$\left.\frac{B1^{(0,0,1,0)}[t, x, y, z]}{2\,B1[t, x, y, z]}, \frac{B1^{(0,0,0,1)}[t, x, y, z]}{2\,B1[t, x, y, z]}\right\},$$
$$\left\{0, \frac{B1^{(0,0,1,0)}[t, x, y, z]}{2\,B1[t, x, y, z]}, -\frac{C1^{(0,1,0,0)}[t, x, y, z]}{2\,B1[t, x, y, z]}, 0\right\},$$
$$\left\{0, \frac{B1^{(0,0,0,1)}[t, x, y, z]}{2\,B1[t, x, y, z]}, \right.$$
$$\left.\left.0, -\frac{D1^{(0,1,0,0)}[t, x, y, z]}{2\,B1[t, x, y, z]}\right\}\right\},$$
$$\left\{\left\{\frac{A1^{(0,0,1,0)}[t, x, y, z]}{2\,C1[t, x, y, z]}, 0, \frac{C1^{(1,0,0,0)}[t, x, y, z]}{2\,C1[t, x, y, z]}, 0\right\},\right.$$

$$\left\{0, -\frac{B1^{(0,0,1,0)}[t, x, y, z]}{2\,C1[t, x, y, z]}, \frac{C1^{(0,1,0,0)}[t, x, y, z]}{2\,C1[t, x, y, z]}, 0\right\},$$
$$\left\{\frac{C1^{(1,0,0,0)}[t, x, y, z]}{2\,C1[t, x, y, z]}, \frac{C1^{(0,1,0,0)}[t, x, y, z]}{2\,C1[t, x, y, z]},\right.$$
$$\left.\frac{C1^{(0,0,1,0)}[t, x, y, z]}{2\,C1[t, x, y, z]}, \frac{C1^{(0,0,0,1)}[t, x, y, z]}{2\,C1[t, x, y, z]}\right\}, \{0,$$
$$0, \frac{C1^{(0,0,0,1)}[t, x, y, z]}{2\,C1[t, x, y, z]}, -\frac{D1^{(0,0,1,0)}[t, x, y, z]}{2\,C1[t, x, y, z]}\}\},$$
$$\left\{\left\{\frac{A1^{(0,0,0,1)}[t, x, y, z]}{2\,D1[t, x, y, z]}, 0, 0, \frac{D1^{(1,0,0,0)}[t, x, y, z]}{2\,D1[t, x, y, z]}\right\},\right.$$
$$\left\{0, -\frac{B1^{(0,0,0,1)}[t, x, y, z]}{2\,D1[t, x, y, z]}, 0, \frac{D1^{(0,1,0,0)}[t, x, y, z]}{2\,D1[t, x, y, z]}\right\},$$
$$\left\{0, 0, -\frac{C1^{(0,0,0,1)}[t, x, y, z]}{2\,D1[t, x, y, z]}, \frac{D1^{(0,0,1,0)}[t, x, y, z]}{2\,D1[t, x, y, z]}\right\},$$
$$\left\{\frac{D1^{(1,0,0,0)}[t, x, y, z]}{2\,D1[t, x, y, z]}, \frac{D1^{(0,1,0,0)}[t, x, y, z]}{2\,D1[t, x, y, z]},\right.$$
$$\left.\left.\frac{D1^{(0,0,1,0)}[t, x, y, z]}{2\,D1[t, x, y, z]}, \frac{D1^{(0,0,0,1)}[t, x, y, z]}{2\,D1[t, x, y, z]}\right\}\right\}$$

Still, one equation of Einstein's vacuum field equations is complicated

```
Einstein[1,1,g,ing]
```

$$-\frac{A1^{(0,0,0,1)}[t, x, y, z]^2}{4\,A1[t, x, y, z]\,D1[t, x, y, z]} +$$
$$\frac{A1^{(0,0,0,1)}[t, x, y, z]\,B1^{(0,0,0,1)}[t, x, y, z]}{4\,B1[t, x, y, z]\,D1[t, x, y, z]} +$$
$$\frac{A1^{(0,0,0,1)}[t, x, y, z]\,C1^{(0,0,0,1)}[t, x, y, z]}{4\,C1[t, x, y, z]\,D1[t, x, y, z]} -$$
$$\frac{A1^{(0,0,0,1)}[t, x, y, z]\,D1^{(0,0,0,1)}[t, x, y, z]}{4\,D1[t, x, y, z]^2} +$$
$$\frac{A1^{(0,0,0,2)}[t, x, y, z]}{2\,D1[t, x, y, z]} - \frac{A1^{(0,0,1,0)}[t, x, y, z]^2}{4\,A1[t, x, y, z]\,C1[t, x, y, z]} +$$
$$\frac{A1^{(0,0,1,0)}[t, x, y, z]\,B1^{(0,0,1,0)}[t, x, y, z]}{4\,B1[t, x, y, z]\,C1[t, x, y, z]} -$$
$$\frac{A1^{(0,0,1,0)}[t, x, y, z]\,C1^{(0,0,1,0)}[t, x, y, z]}{4\,C1[t, x, y, z]^2} +$$
$$\frac{A1^{(0,0,1,0)}[t, x, y, z]\,D1^{(0,0,1,0)}[t, x, y, z]}{4\,C1[t, x, y, z]\,D1[t, x, y, z]} +$$

$$\frac{A1^{(0,0,2,0)}[t,x,y,z]}{2\,C1[t,x,y,z]}-\frac{A1^{(0,1,0,0)}[t,x,y,z]^2}{4\,A1[t,x,y,z]\,B1[t,x,y,z]}-\frac{A1^{(0,1,0,0)}[t,x,y,z]\,B1^{(0,1,0,0)}[t,x,y,z]}{4\,B1[t,x,y,z]^2}+\frac{A1^{(0,1,0,0)}[t,x,y,z]\,C1^{(0,1,0,0)}[t,x,y,z]}{4\,B1[t,x,y,z]\,C1[t,x,y,z]}+\frac{A1^{(0,1,0,0)}[t,x,y,z]\,D1^{(0,1,0,0)}[t,x,y,z]}{4\,B1[t,x,y,z]\,D1[t,x,y,z]}+\frac{A1^{(0,2,0,0)}[t,x,y,z]}{2\,B1[t,x,y,z]}+\frac{A1^{(1,0,0,0)}[t,x,y,z]\,B1^{(1,0,0,0)}[t,x,y,z]}{4\,A1[t,x,y,z]\,B1[t,x,y,z]}+\frac{B1^{(1,0,0,0)}[t,x,y,z]^2}{4\,B1[t,x,y,z]^2}+\frac{A1^{(1,0,0,0)}[t,x,y,z]\,C1^{(1,0,0,0)}[t,x,y,z]}{4\,A1[t,x,y,z]\,C1[t,x,y,z]}+\frac{C1^{(1,0,0,0)}[t,x,y,z]^2}{4\,C1[t,x,y,z]^2}+\frac{A1^{(1,0,0,0)}[t,x,y,z]\,D1^{(1,0,0,0)}[t,x,y,z]}{4\,A1[t,x,y,z]\,D1[t,x,y,z]}+\frac{D1^{(1,0,0,0)}[t,x,y,z]^2}{4\,D1[t,x,y,z]^2}-\frac{B1^{(2,0,0,0)}[t,x,y,z]}{2\,B1[t,x,y,z]}-\frac{C1^{(2,0,0,0)}[t,x,y,z]}{2\,C1[t,x,y,z]}-\frac{D1^{(2,0,0,0)}[t,x,y,z]}{2\,D1[t,x,y,z]}-\frac{1}{2}A1[t,x,y,z]\left(-\frac{A1^{(0,0,0,1)}[t,x,y,z]^2}{2\,A1[t,x,y,z]^2\,D1[t,x,y,z]}+\frac{A1^{(0,0,0,1)}[t,x,y,z]\,B1^{(0,0,0,1)}[t,x,y,z]}{2\,A1[t,x,y,z]\,B1[t,x,y,z]\,D1[t,x,y,z]}-\frac{B1^{(0,0,0,1)}[t,x,y,z]^2}{2\,B1[t,x,y,z]^2\,D1[t,x,y,z]}+\frac{A1^{(0,0,0,1)}[t,x,y,z]\,C1^{(0,0,0,1)}[t,x,y,z]}{2\,A1[t,x,y,z]\,C1[t,x,y,z]\,D1[t,x,y,z]}+\frac{B1^{(0,0,0,1)}[t,x,y,z]\,C1^{(0,0,0,1)}[t,x,y,z]}{2\,B1[t,x,y,z]\,C1[t,x,y,z]\,D1[t,x,y,z]}-\frac{C1^{(0,0,0,1)}[t,x,y,z]^2}{2\,C1[t,x,y,z]^2\,D1[t,x,y,z]}-\frac{A1^{(0,0,0,1)}[t,x,y,z]\,D1^{(0,0,0,1)}[t,x,y,z]}{2\,A1[t,x,y,z]\,D1[t,x,y,z]^2}-\frac{B1^{(0,0,0,1)}[t,x,y,z]\,D1^{(0,0,0,1)}[t,x,y,z]}{2\,B1[t,x,y,z]\,D1[t,x,y,z]^2}-$$

$$\frac{C1^{(0,0,0,1)}[t,x,y,z]\,D1^{(0,0,0,1)}[t,x,y,z]}{2\,C1[t,x,y,z]\,D1[t,x,y,z]^2}+\frac{A1^{(0,0,0,2)}[t,x,y,z]}{A1[t,x,y,z]\,D1[t,x,y,z]}+\frac{B1^{(0,0,0,2)}[t,x,y,z]}{B1[t,x,y,z]\,D1[t,x,y,z]}+\frac{C1^{(0,0,0,2)}[t,x,y,z]}{C1[t,x,y,z]\,D1[t,x,y,z]}-\frac{A1^{(0,0,1,0)}[t,x,y,z]^2}{2\,A1[t,x,y,z]^2\,C1[t,x,y,z]}+\frac{A1^{(0,0,1,0)}[t,x,y,z]\,B1^{(0,0,1,0)}[t,x,y,z]}{2\,A1[t,x,y,z]\,B1[t,x,y,z]\,C1[t,x,y,z]}-\frac{B1^{(0,0,1,0)}[t,x,y,z]^2}{2\,B1[t,x,y,z]^2\,C1[t,x,y,z]}-\frac{A1^{(0,0,1,0)}[t,x,y,z]\,C1^{(0,0,1,0)}[t,x,y,z]}{2\,A1[t,x,y,z]\,C1[t,x,y,z]^2}-\frac{B1^{(0,0,1,0)}[t,x,y,z]\,C1^{(0,0,1,0)}[t,x,y,z]}{2\,B1[t,x,y,z]\,C1[t,x,y,z]^2}+\frac{A1^{(0,0,1,0)}[t,x,y,z]\,D1^{(0,0,1,0)}[t,x,y,z]}{2\,A1[t,x,y,z]\,C1[t,x,y,z]\,D1[t,x,y,z]}+\frac{B1^{(0,0,1,0)}[t,x,y,z]\,D1^{(0,0,1,0)}[t,x,y,z]}{2\,B1[t,x,y,z]\,C1[t,x,y,z]\,D1[t,x,y,z]}-\frac{C1^{(0,0,1,0)}[t,x,y,z]\,D1^{(0,0,1,0)}[t,x,y,z]}{2\,C1[t,x,y,z]^2\,D1[t,x,y,z]}-\frac{D1^{(0,0,1,0)}[t,x,y,z]^2}{2\,C1[t,x,y,z]\,D1[t,x,y,z]^2}+\frac{A1^{(0,0,2,0)}[t,x,y,z]}{A1[t,x,y,z]\,C1[t,x,y,z]}+\frac{B1^{(0,0,2,0)}[t,x,y,z]}{B1[t,x,y,z]\,C1[t,x,y,z]}+\frac{D1^{(0,0,2,0)}[t,x,y,z]}{C1[t,x,y,z]\,D1[t,x,y,z]}-\frac{A1^{(0,1,0,0)}[t,x,y,z]^2}{2\,A1[t,x,y,z]^2\,B1[t,x,y,z]}-\frac{A1^{(0,1,0,0)}[t,x,y,z]\,B1^{(0,1,0,0)}[t,x,y,z]}{2\,A1[t,x,y,z]\,B1[t,x,y,z]^2}+\frac{A1^{(0,1,0,0)}[t,x,y,z]\,C1^{(0,1,0,0)}[t,x,y,z]}{2\,A1[t,x,y,z]\,B1[t,x,y,z]\,C1[t,x,y,z]}-\frac{B1^{(0,1,0,0)}[t,x,y,z]\,C1^{(0,1,0,0)}[t,x,y,z]}{2\,B1[t,x,y,z]^2\,C1[t,x,y,z]}-$$

$$\frac{\text{C1}^{(0,1,0,0)}[t, x, y, z]^2}{2\,\text{B1}[t, x, y, z]\,\text{C1}[t, x, y, z]^2} + \frac{\text{A1}^{(0,1,0,0)}[t, x, y, z]\,\text{D1}^{(0,1,0,0)}[t, x, y, z]}{2\,\text{A1}[t, x, y, z]\,\text{B1}[t, x, y, z]\,\text{D1}[t, x, y, z]} - \frac{\text{B1}^{(0,1,0,0)}[t, x, y, z]\,\text{D1}^{(0,1,0,0)}[t, x, y, z]}{2\,\text{B1}[t, x, y, z]^2\,\text{D1}[t, x, y, z]} + \frac{\text{C1}^{(0,1,0,0)}[t, x, y, z]\,\text{D1}^{(0,1,0,0)}[t, x, y, z]}{2\,\text{B1}[t, x, y, z]\,\text{C1}[t, x, y, z]\,\text{D1}[t, x, y, z]} - \frac{\text{D1}^{(0,1,0,0)}[t, x, y, z]^2}{2\,\text{B1}[t, x, y, z]\,\text{D1}[t, x, y, z]^2} + \frac{\text{A1}^{(0,2,0,0)}[t, x, y, z]}{\text{A1}[t, x, y, z]\,\text{B1}[t, x, y, z]} + \frac{\text{C1}^{(0,2,0,0)}[t, x, y, z]}{\text{B1}[t, x, y, z]\,\text{C1}[t, x, y, z]} + \frac{\text{D1}^{(0,2,0,0)}[t, x, y, z]}{\text{B1}[t, x, y, z]\,\text{D1}[t, x, y, z]} + \frac{\text{A1}^{(1,0,0,0)}[t, x, y, z]\,\text{B1}^{(1,0,0,0)}[t, x, y, z]}{2\,\text{A1}[t, x, y, z]^2\,\text{B1}[t, x, y, z]} + \frac{\text{B1}^{(1,0,0,0)}[t, x, y, z]^2}{2\,\text{A1}[t, x, y, z]\,\text{B1}[t, x, y, z]^2} + \frac{\text{A1}^{(1,0,0,0)}[t, x, y, z]\,\text{C1}^{(1,0,0,0)}[t, x, y, z]}{2\,\text{A1}[t, x, y, z]^2\,\text{C1}[t, x, y, z]} - \frac{\text{B1}^{(1,0,0,0)}[t, x, y, z]\,\text{C1}^{(1,0,0,0)}[t, x, y, z]}{2\,\text{A1}[t, x, y, z]\,\text{B1}[t, x, y, z]\,\text{C1}[t, x, y, z]} + \frac{\text{C1}^{(1,0,0,0)}[t, x, y, z]^2}{2\,\text{A1}[t, x, y, z]\,\text{C1}[t, x, y, z]^2} + \frac{\text{A1}^{(1,0,0,0)}[t, x, y, z]\,\text{D1}^{(1,0,0,0)}[t, x, y, z]}{2\,\text{A1}[t, x, y, z]^2\,\text{D1}[t, x, y, z]} - \frac{\text{B1}^{(1,0,0,0)}[t, x, y, z]\,\text{D1}^{(1,0,0,0)}[t, x, y, z]}{2\,\text{A1}[t, x, y, z]\,\text{B1}[t, x, y, z]\,\text{D1}[t, x, y, z]} - \frac{\text{C1}^{(1,0,0,0)}[t, x, y, z]\,\text{D1}^{(1,0,0,0)}[t, x, y, z]}{2\,\text{A1}[t, x, y, z]\,\text{C1}[t, x, y, z]\,\text{D1}[t, x, y, z]} + \frac{\text{D1}^{(1,0,0,0)}[t, x, y, z]^2}{2\,\text{A1}[t, x, y, z]\,\text{D1}[t, x, y, z]^2} - \frac{\text{B1}^{(2,0,0,0)}[t, x, y, z]}{\text{A1}[t, x, y, z]\,\text{B1}[t, x, y, z]} - \frac{\text{C1}^{(2,0,0,0)}[t, x, y, z]}{\text{A1}[t, x, y, z]\,\text{C1}[t, x, y, z]} - \frac{\text{D1}^{(2,0,0,0)}[t, x, y, z]}{\text{A1}[t, x, y, z]\,\text{D1}[t, x, y, z]}\Bigg)$$

6.5.3 Schwarzschild Metric in Kruskal Coordinates

The Kruskal solution is the most general analytical extension of the Schwarzschild metric. Whereas the Eddington–Finkelstein solution is developed for the time region $0 \le t < \infty$ or $-\infty < t \le 0$, the Kruskal solution is extended to both time regions.

The Kruskal solution consists of the two angle variables θ and ϕ, a spacelike variable x and a timelike variable t.

```
IndepVar = {t,x,ϴ,ϕ}
```

$\{t, x, \theta, \phi\}$

The radial distance r is defined implicitly by the equation

```
gld = t^2 - x^2 == -(r(x, t) - 2 m) ℯ^(r(x,t)/(2 m))
```

$$t^2 - x^2 == e^{\frac{r(x,t)}{2m}} (2m - r(x, t))$$

For later calculations, this equation is solved for t:

```
seq = Last[Solve[gld /. r(x, t) → r, t]]
```

$$\left\{t \to \sqrt{x^2 + e^{\frac{r}{2m}} (2m - r)}\right\}$$

The line element is given by the radial coordinate r:

$$\mathbf{dsk} = \frac{e^{-\frac{r(x,t)}{2m}}\,(16\,m^2)\,(dt)^2}{r} - \frac{(16\,m^2)\,e^{-\frac{r(x,t)}{2m}}\,(dx)^2}{r(x,t)} - r(x,t)^2\,((d\theta)^2 + (d\phi)^2\sin^2(\theta))$$

$$\frac{16\,e^{-\frac{r(x,t)}{2m}}\,m^2\,(dt)^2}{r} - r(x,t)^2\,((d\theta)^2 + (d\phi)^2\sin^2(\theta)) - \frac{16\,e^{-\frac{r(x,t)}{2m}}\,m^2\,(dx)^2}{r(x,t)}$$

The metric is again in the shape of a diagonal matrix and its inverse

```
g = metric[dsk,IndepVar]
```

$$\begin{pmatrix} \frac{16\,e^{-\frac{r(x,t)}{2m}}\,m^2}{r} & 0 & 0 & 0 \\ 0 & -\frac{16\,e^{-\frac{r(x,t)}{2m}}\,m^2}{r(x,t)} & 0 & 0 \\ 0 & 0 & -r(x,t)^2 & 0 \\ 0 & 0 & 0 & -r(x,t)^2\sin^2(\theta) \end{pmatrix}$$

```
ing = Inverse[g]
```

$$\begin{pmatrix} \frac{e^{\frac{r(x,t)}{2m}}\,r}{16\,m^2} & 0 & 0 & 0 \\ 0 & -\frac{e^{\frac{r(x,t)}{2m}}\,r(x,t)}{16\,m^2} & 0 & 0 \\ 0 & 0 & -\frac{1}{r(x,t)^2} & 0 \\ 0 & 0 & 0 & -\frac{\csc^2(\theta)}{r(x,t)^2} \end{pmatrix}$$

To calculate the Christoffel symbols and the Einstein tensor, we compute the derivatives of $r[x, t]$ up to second order following from equation **gld**.

```
s1= Flatten[Simplify[Solve[D[gld,x],D[r[x,t],x]]]];

s2 = Flatten[Simplify[Solve[D[gld,t],D[r[x,t],t]]]];

s3 =
Flatten[Simplify[Solve[D[gld,x,x],D[r[x,t],x,x]]
/.s1 ]];

s4 =
Flatten[Simplify[Solve[D[gld,t,t],D[r[x,t],t,t]] /.
s2 ]];
```

```
sg = Flatten[{s1,s2,s3,s4}]
```

$$\left\{r^{(1,0)}(x,\,t) \to \frac{4\,e^{-\frac{r(x,t)}{2m}}\,m\,x}{r(x,\,t)},\; r^{(0,1)}(x,\,t) \to -\frac{4\,e^{-\frac{r(x,t)}{2m}}\,m\,t}{r(x,\,t)},\right.$$
$$r^{(2,0)}(x,\,t) \to \frac{4\,e^{-\frac{r(x,t)}{m}}\,m\left(-4\,m\,x^2 - 2\,r(x,\,t)\,x^2 + e^{\frac{r(x,t)}{2m}}\,r(x,\,t)^2\right)}{r(x,\,t)^3},$$
$$\left.r^{(0,2)}(x,\,t) \to -\frac{4\,e^{-\frac{r(x,t)}{m}}\,m\left(4\,m\,t^2 + 2\,r(x,\,t)\,t^2 + e^{\frac{r(x,t)}{2m}}\,r(x,\,t)^2\right)}{r(x,\,t)^3}\right\}$$

With the list of **sg** rules, the Christoffel symbols and the Einstein tensor are calculated as follows:

```
Table[ Simplify[ Christoffel[i,j,k,g,ing] /. sg
],{i,1,4},{j,1,4},{k,1,4}]
```

$$\left\{\left\{\left\{\frac{e^{-\frac{r[x,t]}{2m}}\,t}{r[x,t]},\ -\frac{e^{-\frac{r[x,t]}{2m}}\,x}{r[x,t]},\ 0,\ 0\right\},\right.\right.$$
$$\left\{-\frac{e^{-\frac{r[x,t]}{2m}}\,x}{r[x,t]},\ \frac{e^{-\frac{r[x,t]}{2m}}\,r\,t\,(2m+r[x,t])}{r[x,t]^3},\ 0,\ 0\right\},$$
$$\left\{0,\ 0,\ -\frac{r\,t}{4m},\ 0\right\},\ \left.\left\{0,\ 0,\ 0,\ -\frac{r\,t\,\mathrm{Sin}[\theta]^2}{4m}\right\}\right\},$$
$$\left\{\left\{-\frac{e^{-\frac{r[x,t]}{2m}}\,x}{r},\ \frac{e^{-\frac{r[x,t]}{2m}}\,t\,(2m+r[x,t])}{r[x,t]^2},\ 0,\ 0\right\},\right.$$
$$\left\{\frac{e^{-\frac{r[x,t]}{2m}}\,t\,(2m+r[x,t])}{r[x,t]^2},\right.$$
$$\left.-\frac{e^{-\frac{r[x,t]}{2m}}\,x\,(2m+r[x,t])}{r[x,t]^2},\ 0,\ 0\right\},$$
$$\left\{0,\ 0,\ -\frac{x\,r[x,t]}{4m},\ 0\right\},\ \left\{0,\ 0,\ 0,\ -\frac{x\,r[x,t]\,\mathrm{Sin}[\theta]^2}{4m}\right\}\Big\},$$
$$\left\{\left\{0,\ 0,\ -\frac{4\,e^{-\frac{r[x,t]}{2m}}\,m\,t}{r[x,t]^2},\ 0\right\},\ \left\{0,\ 0,\ \frac{4\,e^{-\frac{r[x,t]}{2m}}\,m\,x}{r[x,t]^2},\ 0\right\},\right.$$
$$\left\{-\frac{4\,e^{-\frac{r[x,t]}{2m}}\,m\,t}{r[x,t]^2},\ \frac{4\,e^{-\frac{r[x,t]}{2m}}\,m\,x}{r[x,t]^2},\ 0,\ 0\right\},$$
$$\left\{0,\ 0,\ 0,\ -\mathrm{Cos}[\theta]\,\mathrm{Sin}[\theta]\right\},\ \left\{\left\{0,\ 0,\ 0,\ -\frac{4\,e^{-\frac{r[x,t]}{2m}}\,m\,t}{r[x,t]^2}\right\},\right.$$
$$\left\{0,\ 0,\ 0,\ \frac{4\,e^{-\frac{r[x,t]}{2m}}\,m\,x}{r[x,t]^2}\right\},\ \{0,\ 0,\ 0,\ \mathrm{Cot}[\theta]\},$$
$$\left.\left\{-\frac{4\,e^{-\frac{r[x,t]}{2m}}\,m\,t}{r[x,t]^2},\ \frac{4\,e^{-\frac{r[x,t]}{2m}}\,m\,x}{r[x,t]^2},\ \mathrm{Cot}[\theta],\ 0\right\}\right\}\right\}$$

To verify Einstein's field equations, we calculate, for example, the (1,1) coefficient of the Einstein tensor:

```
es1 = Simplify[ Einstein[1,1,g,ing] /. sg  ]
```

$$\frac{8\,e^{-\frac{r(x,t)}{m}}\,m\left(-r\,t^2-e^{\frac{r(x,t)}{2m}}\,r(x,t)^2+\left(x^2+2\,e^{\frac{r(x,t)}{2m}}\,m\right)r(x,t)\right)}{r\,r(x,t)^3}$$

With the aid of the defining equation for r, the above expression vanishes.

```
es1 = es1 /. { r[x,t] → r}
```

$$\frac{8\, e^{-\frac{r}{m}}\, m\left(-e^{\frac{r}{2m}}\, r^2 - t^2\, r + \left(x^2 + 2\, e^{\frac{r}{2m}}\, m\right) r\right)}{r^4}$$

```
Simplify[ PowerExpand[es1 /. seq ] ]
```

0

6.6 The Reissner–Nordstrom Solution for a Charged Mass Point

The Reissner–Nordstrom solution is a spherically symmetric metric for a massive body with charge ε. This type of solution allows the study of the coupling of Einstein's field equations with Maxwell's equations via the energy momentum tensor. Consequently, we have to solve the inhomogeneous field equations. Because of the spherical symmetry, we can use the Kruskal variables:

```
IndepVar = {t,r,Θ,ϕ}
```

$\{t, r, \theta, \phi\}$

The same shape of the line element is also given:

```
dsr = -((dθ)^2 + (dϕ)^2 sin^2(θ)) r^2 - e^λ(r) (dr)^2 + e^ν(r) (dt)^2
```

$$(-(d\theta)^2 - (d\phi)^2 \sin^2(\theta))\, r^2 - e^{\lambda(r)}\, (dr)^2 + e^{\nu(r)}\, (dt)^2$$

The metric tensor follows

```
g = metric[dsr,IndepVar]
```

$$\begin{pmatrix} e^{\nu(r)} & 0 & 0 & 0 \\ 0 & -e^{\lambda(r)} & 0 & 0 \\ 0 & 0 & -r^2 & 0 \\ 0 & 0 & 0 & -r^2 \sin^2(\theta) \end{pmatrix}$$

The related inverse metric tensor is

```
ing = Inverse[g]
```

$$\begin{pmatrix} e^{-\nu(r)} & 0 & 0 & 0 \\ 0 & -e^{-\lambda(r)} & 0 & 0 \\ 0 & 0 & -\frac{1}{r^2} & 0 \\ 0 & 0 & 0 & -\frac{\csc^2(\theta)}{r^2} \end{pmatrix}$$

Since the Reissner–Nordstrom solution possesses spherical symmetry, the coordinates can be chosen so that the metric is static and ν and λ depend only on the radial distance r. At the same time, the Reissner–Nordstrom solution satisfies Einstein's field equations and Maxwell's vacuum equations. Consequently, the Maxwell tensor F also depends on the distance r. Its form is determined by a purely radial electrostatic field.

```
F = {{ 0, -
Ee[r],0,0},{Ee[r],0,0,0},{0,0,0,0},{0,0,0,0}}
```

$$\begin{pmatrix} 0 & -\mathrm{Ee}(r) & 0 & 0 \\ \mathrm{Ee}(r) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

According to Maxwell's equations, the covariant divergence of the Maxwell tensor must vanish. The conditions deliver the substitution rule

```
sm = {Ee[r] → ε E^(1/2 (λ[r]+ν[r])) / r^2}
```

$$\left\{\text{Ee}(r) \to \frac{e^{\frac{1}{2}(\lambda(r)+\nu(r))}\,\varepsilon}{r^2}\right\}$$

and the Maxwell tensor

```
F = F /. sm
```

$$\begin{pmatrix} 0 & -\frac{e^{\frac{1}{2}(\lambda(r)+\nu(r))}\,\varepsilon}{r^2} & 0 & 0 \\ \frac{e^{\frac{1}{2}(\lambda(r)+\nu(r))}\,\varepsilon}{r^2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

with the corresponding covariant tensor.

```
Fc = Simplify[ing . F . ing ]
```

$$\begin{pmatrix} 0 & \frac{e^{-\frac{1}{2}(\lambda(r)+\nu(r))}\,\varepsilon}{r^2} & 0 & 0 \\ -\frac{e^{-\frac{1}{2}(\lambda(r)+\nu(r))}\,\varepsilon}{r^2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

The energy momentum tensor T is computed by

```
T = Simplify[
  Table[ Sum[ Sum[ (-1/(4 π)) ing[[c, d]] F[[a, c]] F[[b, d]] +
        1/(16 π) g[[a, b]] F[[c, d]] Fc[[c, d]], {c, 1, 4}], {d, 1, 4}],
   {a, 1, 4}, {b, 1, 4}]]
```

$$\begin{pmatrix} \frac{e^{\nu(r)}\,\varepsilon^2}{8\pi r^4} & 0 & 0 & 0 \\ 0 & -\frac{e^{\lambda(r)}\,\varepsilon^2}{8\pi r^4} & 0 & 0 \\ 0 & 0 & \frac{\varepsilon^2}{8\pi r^2} & 0 \\ 0 & 0 & 0 & \frac{\varepsilon^2 \sin^2(\theta)}{8\pi r^2} \end{pmatrix}$$

It should be pointed out that the energy momentum tensor for a source-free electromagnetic field is traceless since the Maxwell tensor – a fully antisymmetric tensor – is traceless. According to this property of the energy momentum tensor, the Ricci scalar vanishes as well. Consequently, the field equations reduce to $R = 8\pi T$, where R is the Ricci tensor.

```
Simplify[Table[Ricci[a,b,g,ing] -8 π
T[[a,b]],{a,1,4},{b,1,4}] ]
```

$$\left\{\left\{\frac{1}{4\,r^4}\left(e^{-\lambda[r]+\nu[r]}\left(-4\,e^{\lambda[r]}\,\varepsilon^2 + r^3\,(4 - r\,\lambda'[r])\,\nu'[r] + r^4\,\nu'[r]^2 + 2\,r^4\,\nu''[r]\right)\right), 0, 0, 0\right\},\right.$$
$$\left\{0, \frac{1}{4}\left(\frac{4\,e^{\lambda[r]}\,\varepsilon^2}{r^4} - \nu'[r]^2 + \lambda'[r]\left(\frac{4}{r} + \nu'[r]\right) - 2\,\nu''[r]\right), 0, 0\right\},$$
$$\left\{0, 0, \frac{1}{2}\left(2 - 2\,e^{-\lambda[r]} - \frac{2\,\varepsilon^2}{r^2} + e^{-\lambda[r]}\,r\,\lambda'[r] - e^{-\lambda[r]}\,r\,\nu'[r]\right), 0\right\},$$
$$\left.\left\{0, 0, 0, \frac{1}{2\,r^2}\left(e^{-\lambda[r]}\,\mathrm{Sin}[\theta]^2\left(-2\,r^2 + 2\,e^{\lambda[r]}\,r^2 - 2\,e^{\lambda[r]}\,\varepsilon^2 + r^3\,\lambda'[r] - r^3\,\nu'[r]\right)\right)\right\}\right\}$$

The solutions of these differential equations can easily be verified. With the coordinates

```
IndepVar = {t,r,θ,ϕ}
```

$\{t, r, \theta, \phi\}$

the line element is given by

```
dsrn =
```

$$-((d\theta)^2 + (d\phi)^2 \sin^2(\theta))\, r^2 - \frac{(dr)^2}{\frac{\varepsilon^2}{r^2} - \frac{2m}{r} + 1} + \left(\frac{\varepsilon^2}{r^2} - \frac{2m}{r} + 1\right)(dt)^2$$

$$(-(d\theta)^2 - (d\phi)^2 \sin^2(\theta))\, r^2 - \frac{(dr)^2}{\frac{\varepsilon^2}{r^2} - \frac{2m}{r} + 1} + \left(\frac{\varepsilon^2}{r^2} - \frac{2m}{r} + 1\right)(dt)^2$$

and the metric tensor

```
g = metric[dsrn,IndepVar];g//MatrixForm
```

$$\begin{pmatrix} \frac{\varepsilon^2}{r^2} - \frac{2m}{r} + 1 & 0 & 0 & 0 \\ 0 & -\frac{r^2}{r^2 - 2mr + \varepsilon^2} & 0 & 0 \\ 0 & 0 & -r^2 & 0 \\ 0 & 0 & 0 & -r^2 \sin^2(\theta) \end{pmatrix}$$

with the corresponding inverse

```
ing = Simplify[ Inverse[g] ];ing//MatrixForm
```

$$\begin{pmatrix} \frac{r^2}{r^2 - 2mr + \varepsilon^2} & 0 & 0 & 0 \\ 0 & -\frac{\varepsilon^2}{r^2} + \frac{2m}{r} - 1 & 0 & 0 \\ 0 & 0 & -\frac{1}{r^2} & 0 \\ 0 & 0 & 0 & -\frac{\csc^2(\theta)}{r^2} \end{pmatrix}$$

The two parameters can be interpreted as the charge ε of the body and the geometric mass m. Of course, in reality, a body of considerable mass has

no net charge. Therefore, the Reissner–Nordstrom solution is only of hypothetical interest. However, the Reissner–Nordstrom solution can help in the study of the more complicated Kerr solution for a rotating black hole due to the similarity of its structure.

The determinant for the Reissner–Nordstrom solution is the same as for the Schwarzschild solution. It is plotted in Figure 6.6.8.

```
detg = Simplify[Det[g]]
```

$$-r^4 \sin^2(\theta)$$

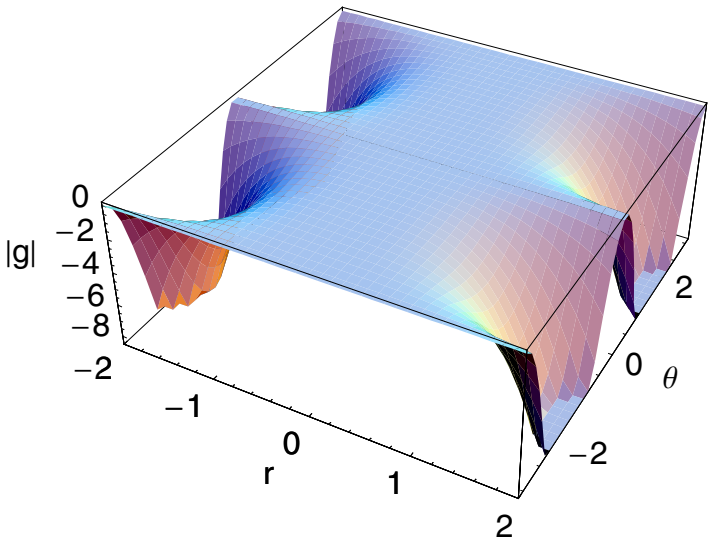


Figure 6.6.8. The determinant $|g|$ for the Reissner–Nordstrom solution.

According to the metric of the Maxwell tensor, the energy momentum tensor reduces to

```
sme = {ν[r] → - λ[r] }
```

$$\{\nu(r) \to -\lambda(r)\}$$

```
F = F /. sme
```

$$\begin{pmatrix} 0 & -\frac{\varepsilon}{r^2} & 0 & 0 \\ \frac{\varepsilon}{r^2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

```
Fc = Fc /.sme
```

$$\begin{pmatrix} 0 & \frac{\varepsilon}{r^2} & 0 & 0 \\ -\frac{\varepsilon}{r^2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

```
T = Simplify[
  Table[∑_{c=1}^{4} ∑_{d=1}^{4} (-1/(4 π) ing[[c, d]] F[[a, c]] F[[b, d]] +
      1/(16 π) g[[a, b]] F[[c, d]] Fc[[c, d]]),
   {a, 1, 4}, {b, 1, 4}]]
```

$$\begin{pmatrix} \frac{\varepsilon^2 (r^2-2mr+\varepsilon^2)}{8\pi r^6} & 0 & 0 & 0 \\ 0 & -\frac{\varepsilon^2}{8\pi r^2 (r^2-2mr+\varepsilon^2)} & 0 & 0 \\ 0 & 0 & \frac{\varepsilon^2}{8\pi r^2} & 0 \\ 0 & 0 & 0 & \frac{\varepsilon^2 \sin^2(\theta)}{8\pi r^2} \end{pmatrix}$$

We have so far calculated all quantities sufficient to verify the field equations in a modified form:

```
Simplify[Table[ Ricci[a,b,g,ing] - 8 π
T[[a,b]],{a,1,4},{b,1,4}]]
```

$$\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

The field equations in their original forms are verified as follows:

```
Simplify[ Table[Einstein[a,b,g,ing] - 8 π
T[[a,b]],{a,1,4},{b,1,4}] ]
```

$$\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

As a consequence, the Ricci scalar obviously vanishes:

```
Simplify[RicciScalar[g,ing]]
```

0

6.7 Exercises

1. Extend the databases in the package **PerihelionShift'** to other planets and planetary systems.

2. Find a representation of the perihelion shift using the classical parameters of an orbit. Compare your calculations to the approximations given in literature.

3. Change the package **LightBending'** in such a way that you are able to treat arbitrary masses in the calculations of light bending.*Caution*: Save the package before making changes in the program!

4. Create a three-dimensional representation of the relation for light bending (6.53) which considers changes in the mass and diameter of the star.

5. The line element in a three-dimensional space in a particular coordinate system is

$$ds^2 = dx_1^2 + x_1\, dx_2^2 + x_1 \sin^2(x_2)\, dx_3^2.$$

First, identify the coordinates and, second, examine the flatness of the metric.

6. The Minkowski line element in Minkowski coordinates

$$x_a = (x_0,\, x_1,\, x_2,\, x_3) = (t,\, x,\, y,\, z)$$

is given by

$$ds^2 = dt^2 - dx^2 - dy^2 - dz^2.$$

Is the metric flat? Determine the metric tensor.

7. Find the nonzero components of the Christoffel symbols Γ^a_{bc} of Bondi's radiating metric:

$$\begin{aligned} ds^2 = &\left(\frac{V}{r}\, e^{2\beta} - U^2\, r^2\, e^{2\gamma}\right) du^2 + 2\, e^{2\beta}\, du\, dr + \\ &2\, U\, r^2\, e^{2\gamma}\, du\, d\theta - r^2\, (e^{2\gamma}\, d\theta^2 + e^{-2\gamma} \sin^2(\theta)\, d\phi^2), \end{aligned}$$

where V, U, β and γ are four arbitrary functions of the three coordinates u, r, and θ.

8. Verify that the Kerr form is a solution of the Einstein field equations. The Kerr form is

$$\begin{aligned} ds^2 = &\, dt^2 - dx^2 - dy^2 - dz^2 - \\ &\frac{2\, m\, r^3}{r^4 + a^2\, z^2}\, \left(dt^2 + \frac{r}{a^2 + r^2}\, (x\, dx + y\, dy) + \right. \\ &\left. \frac{a}{a^2 + r^2}\, (y\, dx - x\, dy) + \frac{z}{r}\, dz\right)^2, \end{aligned}$$

where m and a are constants.

9. Check that the Boyer–Lindquist form of Kerr's solution is a solution of Einstein's field equations

$$\begin{aligned} ds^2 = &\frac{\Delta}{\rho^2}\, (dt - a \sin^2(\theta)\, d\phi)^2 - \\ &\frac{\sin^2(\theta)}{\rho^2}\, ((r^2 + a^2)\, d\phi - a\, dt)^2 - \frac{\rho^2}{\Delta}\, dr^2 - \rho^2\, d\theta^2, \end{aligned}$$

where $\rho^2 = r^2 + a^2 \cos^2(\theta)$ and $\Delta = r^2 - 2\, m\, r + a^2$.

6.8 Packages and Programs

6.8.1 EulerLagrange Equations

This section gives some support in calculating the Euler–Lagrange equations. First, the notation package is loaded.

```
<< Utilities`Notation`
```

Then, the path where you have located the package follows. Please change the path if you have stored the package in a different directory

```
$EulerLagrangePath =
 $AddOnsDirectory <> "/Applications/EulerLagrange/";
AppendTo[$Path, $EulerLagrangePath];
```

The next line loads the package.

```
<< EulerLagrange.m
```

```
================================================

EulerLagrange™ 1.0 (Dos/Windows®)

© 1992-2005 Dr. Gerd Baumann

Runs with Mathematica® Version 3.0 or later

Licensed to one machine only, copying prohibited

================================================
```

Here, we define a symbolic notation for the function

```
Notation[ℰ^x_u_[den_] ⟺ EulerLagrange[den_, u_, x_]]
```

The following pallet allows you to generate the shorthand notation for the Euler–Lagrange operator. You can generate the pallet by selecting the following cell and use the File+Generate Pallet from Selection button to activate the pallet.

$$\mathcal{E}^{\square}_{\square[\square]}$$

6.8.2 PerihelionShift

This package calculates the perihelion shift for different planets. The planets are collected in a database which can be extended by the user.

```
BeginPackage["PerihelionShift`"];

Clear[e1, e2, e3, g2, g3, omega1, omega2,
  Orbit, orbit, Energy, AngularMomentum,
  PerihelionShift, Planets, D0Orbit, Schwarzschild];

Planets::usage = "Planets[planet_String]
    creates a list of data for planets and
planetoids stored in the data base of the
    package PerihelionShift. The data
base contains the names of the planets,
    their major axes, their eccentricity
and the mass of the central planet.
    Planets['List'] creates a list of the
planets in the data base. Planets['name']
    delivers the data of the planet
given in the argument.";

orbit::usage =
  "orbit[phiend_,minorAxes_,majorAxes_,mass_]
creates a graphical representation of
    the perihelion shift if the major and
minor axes and the mass are given.";


Orbit::usage = "Orbit[planet_String]
```

```
    creates a graphical representation of the
perihelion shift for the planets
    contained in the data base.";

PerihelionShift::usage =
  "PerihelionShift[minorAxes_,majorAxes_,mass_]
Calculates the numerical value
    of the perihelion shift.";

AngularMomentum::usage =
  "AngularMomentum[minorAxes_,majorAxes_,mass_]
calculates the angular momentum of a planet.";

Energy::usage = "Energy[minorAxes_,majorAxes_,mass_]
calculates the energy of a planet.";

D0Orbit::usage = "D0Orbit[planet_String,
    phiend_,options___] plots the orbit
in the case of vanishing determinants (see text).";


Begin["`Private`"];

(*---data bases of several planets---*)

data =
  {{"Mercury", 0.5791 10^(11), 0.2056, MassOfTheSun},
   {"Venus", 1.0821 10^(11), 0.0068, MassOfTheSun},
   {"Earth", 1.4967 10^(11), 0.0167, MassOfTheSun},
   {"Icarus", 1.61 10^(11), 0.827, MassOfTheSun},
   {"Mars", 2.2279 10^(11), 0.093, MassOfTheSun},
   {"Ceres", 4.136 10^(11), 0.076, MassOfTheSun},
   {"Jupiter", 7.78 10^(11), 0.048, MassOfTheSun},
   {"Saturn", 14.27 10^(11), 0.056, MassOfTheSun},
   {"Uranus", 28.70 10^(11), 0.047, MassOfTheSun},
   {"Neptune", 44.96 10^(11), 0.009, MassOfTheSun},
   {"Pluto", 59.10 10^(11), 0.25, MassOfTheSun},
   {"PSR1916", 7.0204020286 10^(8), 0.6171313,
    2.82837 MassOfTheSun}, {"TestPlanet",
    5.2327 10^(8), 0.6171313, 2828.37 MassOfTheSun}};

(*---information on the planets---*)
```

```
Planets[planet_String] :=
  Block[{gh, kh, ma}, MassOfTheSun = 1.993 10^(30);
   If[planet == "List",
    Print[DisplayForm[GridBox[Prepend[
        Map[Map[PaddedForm[#, {5, 3}] &, #] &, data],
        {StyleForm["planet", FontWeight → "Bold"],
         StyleForm["mean radius",
          FontWeight → "Bold"], StyleForm[
          "eccentricity", FontWeight → "Bold"],
         StyleForm["mass", FontWeight → "Bold"]}],
       RowLines → True, ColumnLines → True,
       GridFrame → True,
       ColumnAlignments → {Left}]]], gh = 0;
    kh = 0;
    ma = 0;
    Do[If[planet == data[[k, 1]], Planet = data[[k, 1]];
      gh = data[[k, 2]];
      kh = N[data[[k, 2]] Sqrt[1 - data[[k, 3]]]];
      ma = data[[k, 4]];
      Print[DisplayForm[
        GridBox[{{data[[k, 1]], " "}, {"mass", ma},
          {"minor axes", kh}, {"major axes", gh},
          {"eccentricity", data[[k, 3]]}}, RowLines →
          True, ColumnLines → True, GridFrame → True,
         ColumnAlignments → {Left}]]], gh = gh;
      kh = kh;
      ma = ma], {k, 1, Length[data]}];
    MajorAxes = gh;
    MinorAxes = kh;
    Mass = ma;
    If[gh ≠ 0, PerihelionShift[kh, gh, ma], 0]]];

(*---Schwarzschild radius---*)

SchwarzSchild[mass_] :=
  Block[{Gravitation, SpeedOfLight},
   Gravitation = 6.6732 10^(-11);
   SpeedOfLight = 2.9979250 10^8;
   2 Gravitation mass / SpeedOfLight^2];

(*---roots of the characteristic polynomial---*)
```

```
e2[minorAxes_, majorAxes_, mass_] :=
  Block[{Schwarzschild, eh},
   Schwarzschild = SchwarzSchild[mass];
   eh = -(1 - 3 majorAxes Schwarzschild / minorAxes^2
         (1 - Sqrt[majorAxes^2 - minorAxes^2] /
            majorAxes)) / 12];

e3[minorAxes_, majorAxes_, mass_] :=
  Block[{Schwarzschild, eh},
   Schwarzschild = SchwarzSchild[mass];
   eh = -(1 - 3 majorAxes Schwarzschild / minorAxes^2
         (1 + Sqrt[majorAxes^2 - minorAxes^2] /
            majorAxes)) / 12];

e1[minorAxes_, majorAxes_, mass_] :=
  Block[{}, -(e3[minorAxes, majorAxes, mass] +
      e2[minorAxes, majorAxes, mass])];

(*---g2 and g3 of the Weierstrass function---*)

g2[minorAxes_, majorAxes_, mass_] :=
  Block[{}, 2 (e1[minorAxes, majorAxes, mass]^2 +
      e2[minorAxes, majorAxes, mass]^2 +
      e3[minorAxes, majorAxes, mass]^2)];

g3[minorAxes_, majorAxes_, mass_] :=
  Block[{}, 4 e1[minorAxes, majorAxes, mass] *
    e2[minorAxes, majorAxes, mass] *
    e3[minorAxes, majorAxes, mass]];

(*---frequencies of the Weierstrass function---*)

omega1[minorAxes_, majorAxes_, mass_] :=
  Block[{integrand, x, om1, e11, e21, e31, module},
   integrand = 4 x^3 - g2[minorAxes, majorAxes, mass] x -
     g3[minorAxes, majorAxes, mass];
   integrand = 1 / Sqrt[integrand];
   e11 = e1[minorAxes, majorAxes, mass];
   e21 = e2[minorAxes, majorAxes, mass];
   e31 = e3[minorAxes, majorAxes, mass];
   module = (e31 - e21) / (e11 - e21);
```

```
    om1 = EllipticK[module] / Sqrt[e11 - e21]];

omega2[minorAxes_, majorAxes_, mass_] :=
  Block[{integrand, x, om2, e11,
    e21, e31, module}, integrand =
    Abs[4 x^3 - g2[minorAxes, majorAxes, mass] x -
      g3[minorAxes, majorAxes, mass]];
   integrand = 1 / Sqrt[integrand];
   e11 = e1[minorAxes, majorAxes, mass];
   e21 = e2[minorAxes, majorAxes, mass];
   e31 = e3[minorAxes, majorAxes, mass];
   module = (e31 - e21) / (e11 - e21);
   module = 1 - module;
   om2 = I EllipticK[module] / Sqrt[e11 - e21]];

(*---creates the orbit
   from the orbit parameters---*)

orbit[phiend_, minorAxes_, majorAxes_, mass_, planet_] :=
  Block[{Schwarzschild, bh, omega3, l2, l3, l4, l5,
    phi}, Schwarzschild = SchwarzSchild[mass];
   om1 = omega1[minorAxes, majorAxes, mass];
   om2 = omega2[minorAxes, majorAxes, mass];
   omega3 = om1 + om2;
   l2 = g2[minorAxes, majorAxes, mass];
   l3 = g3[minorAxes, majorAxes, mass];
   l4 = Chop[WeierstrassP[phi - omega3, {l2, l3}]];
   l5 = 1 + 12 l4;
   bh = Re[3 Schwarzschild / l5];
   ParametricPlot[{Cos[phi] bh, Sin[phi] bh}, {phi, 0,
     phiend}, PlotRange → All, AspectRatio → Automatic,
    Prolog → Thickness[0.001], PlotLabel → planet]];

(*---creates the orbit with the data base---*)

Orbit[planet_String] := Block[{}, Planets[planet];
   orbit[6 Pi, MinorAxes, MajorAxes, Mass, planet]];

(*---numerical value of the perihelion shift---*)

PerihelionShift[minorAxes_, majorAxes_, mass_] :=
  Block[{ph, ph1}, ph =
```

```
    N[2 (omega1[minorAxes, majorAxes, mass] - Pi), 16];
   ph1 = ph 2.062648062451 0^5;
   Print[" "];
   Print[" Perihelion shift = ", ph1, " arcs"];
   ph];

(*---constants of motion---*)

AngularMomentum[minorAxes_, majorAxes_, mass_] :=
  Block[{Schwarzschild, l1},
   Schwarzschild = SchwarzSchild[mass];
   l1 = g2[minorAxes, majorAxes, mass];
   l1 = Schwarzschild/ (2 (1 / 12 - l1))];

Energy[minorAxes_, majorAxes_, mass_] :=
  Block[{Schwarzschild, energy, l2, l3},
   Schwarzschild = SchwarzSchild[mass];
   l2 = g2[minorAxes, majorAxes, mass];
   l3 = g3[minorAxes, majorAxes, mass];
   energy = -2 Sqrt[(1 / 54 - l2 / 6 - l3) / (1 / 12 - l2)] /
      SpeedOfLight];

(*---asymptitic orbits---*)

D0Orbit[planet_String, phiend_, options___] := Block[
   {Schwarzschild, e0, n2, phi}, Planets[planet];
   Schwarzschild = SchwarzSchild[Mass];
   e0 = 1 / 24 - Schwarzschild / (4 MajorAxes);
   n2 = 3 e0;
   bh1 = 4 / Schwarzschild
     (1 / 12 + n2 / 3 - n2 / Cosh[Sqrt[n2] phi]^2);
   bh1 = 1 / bh1;
   ParametricPlot[{Cos[phi] bh1, Sin[phi] bh1},
    {phi, -phiend, phiend}, options]];
End[];
EndPackage[];
```

6.8.3 LightBending

This package determines the bending of a light beam in a gravitational field.

```
BeginPackage["LightBending`"];


Remove[e1, e2, e3, g2, g3, omega1, omega2, Orbit,
Deviation];

Deviation::usage = "Deviation[radius_,mass_]
calculates the numerical value
of the light bending in a gravitational field of a
planet with mass M in a
distance radius of the center.";

Orbit::usage = "Orbit[radius_,mass_] plots the orbit
of a light beam near
a mass in the distance radius. The calculation is
done in Schwarzschild
metric.";

MassOfTheSun::usage;
RadiusOfTheSun::usage;

Begin["`Private`"];

(* --- mass and radius of the sun --- *)

MassOfTheSun = 1.993 10^(30);
RadiusOfTheSun = 7 10^8;

(* --- Schwarzschild radius --- *)

SchwarzSchild[mass_]:=
      Block[{Gravitation,SpeedOfLight},
      Gravitation = 6.6732 10^(-11);
      SpeedOfLight = 2.9979250 10^8;
      Schwarzschild = 2 Gravitation
mass/SpeedOfLight^2
      ];

(* --- roots of the characteristic polynomial --- *)

e1[radius_,mass_]:=
      Block[{eh,e31},
      e21 = e2[radius,mass];
```

```
      eh = N[-1/2 e21 + Sqrt[3] Sqrt[1-36
e21^2]/12]];

e2[radius_,mass_]:=
      Block[{Schwarzschild,eh},
      Schwarzschild = SchwarzSchild[mass];
      eh = -1/12 (1 - 3 Schwarzschild/radius)
       ];

e3[radius_,mass_]:=
      Block[{eh},
      eh = N[-(e2[radius,mass] + e1[radius,mass])]];

(* --- frequencies of the Weierstrass function --- *)

omega1[radius_,mass_]:=
      Block[{om1,e11,e21,e31,modulus},
      e11 = e1[radius,mass];
      e21 = e2[radius,mass];
      e31 = e3[radius,mass];
      modulus = (e21-e31)/(e11-e31);
      om1 = EllipticK[modulus]/Sqrt[e11-e31]
    ];

omega2[radius_,mass_]:=
      Block[{om2,e11,e21,e31,modulus},
      e11 = e1[radius,mass];
      e21 = e2[radius,mass];
      e31 = e3[radius,mass];
      modulus = (e21-e31)/(e11-e31);
      modulus = 1 - modulus;
      om2 = I EllipticK[modulus]/Sqrt[e11-e31]
    ];

(* --- g2 and g3 of the Weierstrass function --- *)

g2[radius_,mass_]:=Block[{},N[1/12]];

g3[radius_,mass_]:=Block[{},
      4 e1[radius,mass] e2[radius,mass]
e3[radius,mass]];

(* --- creates the path of the light beam --- *)

Orbit[radius_,mass_]:=

Block[{Schwarzschild,bh,l2,l3,l4,l5,phi,phia,deltaphi
,
```

```
         erg,omega3},
      Schwarzschild = SchwarzSchild[mass];
      om1 = omega1[radius,mass];
      om2 = omega2[radius,mass];
      omega3 = om1 + om2;
      l2 = g2[radius,mass];
      l3 = g3[radius,mass];
      l4 = WeierstrassP[phi-omega3,{l2,l3}]+1/12;
      erg = FindRoot[l4==0,{phi,Pi/2}];
      phia = phi /. erg;
      phia = Re[phia];
      l4 = Re[WeierstrassP[phi-omega3,{l2,l3}]];
      l5 = 1 + 12 l4;
      bh = 3 Schwarzschild/l5;
      ParametricPlot[{Cos[phi] bh,Sin[phi] bh},
            {phi,-phia 0.9,phia 0.9},
            Prolog->Thickness[0.001],Ticks->False]
      ];

(* --- determination of the deviation angle --- *)

Deviation[radius_,mass_]:=

Block[{Schwarzschild,om1,om2,omega3,l2,l3,l4,phi,
         deltaphi,dphi,phia,erg},
      Schwarzschild = SchwarzSchild[mass];
      om1 = omega1[radius,mass];
      om2 = omega2[radius,mass];
      omega3 = om1+om2;
      l2 = g2[radius,mass];
      l3 = g3[radius,mass];
      l4 = WeierstrassP[phi-omega3,{l2,l3}]+1/12;
      erg =
FindRoot[l4==0,{phi,Pi/2},AccuracyGoal\[Rule]34,Worki
ngPrecision\[Rule]34,
    MaxIterations\[Rule]50];
      phia = phi /. erg;
      phia = Re[phia];
      deltaphi = N[2 phia-Pi,16];
(* --- the factor 2.06264806245 10^5 converts radian
to arcsecond --- *)
```

```
        dphi = deltaphi   2.06264806245 10^5;
        Print[" "];
        Print[" Deviation = ",dphi," arcs"];
        deltaphi];
    End[];
    EndPackage[];
```

7
Fractals

7.1 Introduction

Fractals are, today, a basic tool to phenomenologically describe natural objects. The properties of these objects can be the length of a border, the relaxation time spectrum of a process, the geometric structure of trees, the circumference of cells and so forth. All of the measures derived from such objects are related to the choice of the scale length with which the object is examined. Fractals are also a tool to describe natural objects such as biological and medical objects. Fractals are geometric as well as temporal objects having a long-lasting history such as the monster curves in mathematics. Fractals are not only restricted to geometric objects but also have its appearance in time-dependent processes and differential objects. The observation by Mandelbrot [7.4] of the existence of a "Geometry of Nature" has led us to think in a new way about natural objects.

The coastline of Norway, a snowflake in Bavaria, the Mississippi River all of these share a common characteristic that is very common in nature. They all have a certain amount of geometric complexity. The boundary of

the snowflake is difficult to define in geometric terms. The same holds for the other objects. Indeed, the snowflake must have a very long perimeter, but it is a very small geometric structure. The mentioned natural examples provide, with a little reflection, a crisis of definition. If we define a geometric measure as the determination of a quantifiable measure of these examples such as length or area, then the geometric measures of physical characteristics are hard to establish. In fact, the measure could only be approached on an operational level; that is if one wants to measure the length of the perimeter of a snowflake, one would have to know by what means to measure it. Felix Hausdorff (see Figure 7.1.1) was one of the few mathematicians who thought about these problems in the 20th century. At the age of 50, Hausdorff was a well-respected mathematician and well known as a set theoretician. In 1918, Hausdorff published an important paper contributing to measure theory. This 22-page article published by *Mathematische Annalen* gave a new treatment of Lebesgue measure. He contributed a large amount of knowledge with his own words "Hierzu geben wir im folgenden einen kleinen Beitrag". This "little contribution" is his entire theory of measure and of fractional dimension, presented in a clear and general form. This article is a gem. Few people have read it, yet it has brought its author more fame, today, than all the rest of his works put together. The principal application of his theory concerns a family $\mathcal{U}$ of bounded sets associated with a weight $\ell(U)$, where U are the countable sets; thus $\ell(U)$ is a function of the diameter $\rho(U) = \lambda(\rho(U))$, with $\lambda(x) = x^a$. This functional relation is the point at which Hausdorff defines his fractional dimension.

Figure 7.1.1. Felx Hausdorff: born November 8, 1868; died Junuary 26, 1942.

About 60 years after Hausdorff's paper, Benoit Mandelbrot (see Figure 7.1.2) coined the term fractal in his "Geometry of Nature". Mandelbrot examined a large number of natural, artificial, and geometric objects. He also introduced numerical experiments to demonstrate the fractal beauty of mappings. The famous Mandelbrot set is one example demonstrating the fractal nature by an iterated map. Benoit Mandelbrot is the founding father of the fractal community incorporating fields from physics, biology, chemistry, material science, architecture, and so forth. The application of fractal concepts in today's science is omnipresent in all disciplines.

Figure 7.1.2. Benoit Mandelbrot: born November 20, 1924.

This chapter introduces the fractal concept for geometric objects. It discusses the experimental determination of fractal dimensions for geometric structures. In Section 7.4 a monofractal is generalized to the notion of multifractals. The renormalization group theory in Section 7.5 makes a link between renormalization and fractality. Section 7.6 introduces a generalization of derivatives to fractional derivatives.

7.2 Measuring a Borderline

A natural borderline separating two objects can be a complicated curve. When looking at a distant object governed by a geometrical structure, a skyscraper, for example, we get the impression that its borderlines are straight lines. Looking through binoculars, we observe that there are wrinkles and loops in its borderline, and a closer look reveals that the object has an even more complicated shape. Following this reasoning, we may wonder whether natural objects can be described fully by Euclidean geometry. In fact, nowhere in nature will we observe the idealized *straight line*. Nature itself uses straight lines connecting two different points only as an approximation and on small scales. Objects in our natural environment have different geometrical structures at different scales of magnification.

Let us consider a tree as an object of our study. If we are far away from the tree, we can imagine that the picture we see is similar to a point or a short line on the horizon. If we get closer to the tree, the appearance changes. First, we see the extension in a plane, and coming closer, we see the spatial arrangements of its branches. Up close enough, we recognize small branches and leaves. The building blocks of a tree are not geometrical objects like cylinders, balls, cones, and the like. The branches of a tree exhibit self-similarity: After scaling of a branch, a subbranch forms from which another subbranch can be scaled, and so on. This type of self-similar scaling law was discovered by Leonardo da Vinci, who experimented with this subject back in the 16th century [7.4]. In modern-day mathematics Benoit Mandelbrot has introduced the term *fractals* to describe such scaling laws of self-similarity.

When studying complicated natural objects, we simplify the problem by considering the three-dimensional object in a projection plane. In the case of the tree, we study the shadow of the tree in order to reduce the problem. The picture of the shadow is easily created with *Mathematica* following Gray and Glynn [7.1] (see Figure 7.2.3). To construct the tree, simple building blocks are put together in a self-similar way. The package **Tree`** contains all the necessary functions to create branches, **branchLine[]**, to

rotate lines, **rotateLine[]**, and to scale branches, **BranchScaling**. A listing of the package is given in the section packages of this Chapter 7. A typical application of the main function is given below. Here, we generate a tree consisting of 10 branch generations and a natural coloring of the branches.

```
Tree[Generation → 10, BranchColor → 11];
```

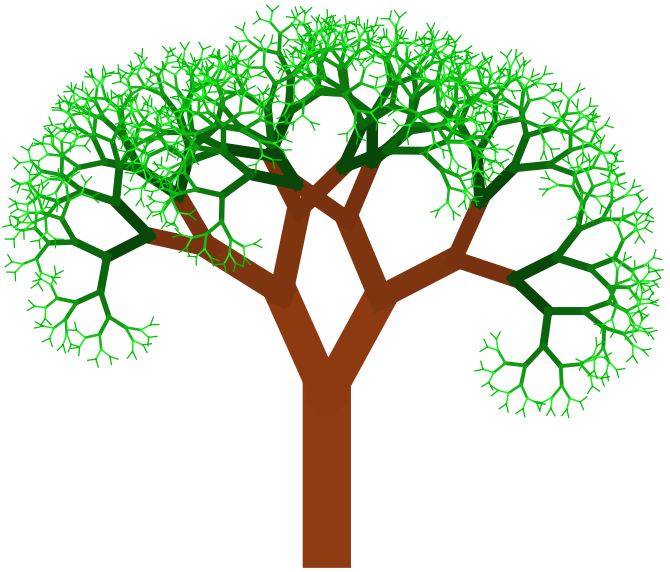


Figure 7.2.3. Fractal tree.

The result is a tree that you will observe in a similar shape in spring or autumn.

One of the characteristic properties of a projected tree is the length of its boundary line. If we choose a fixed yardstick length for determining the length of the boundary line, we get its total length by the number of yardsticks multiplied by the length of the yardstick. The mathematical formula is $L = N(\varepsilon)\,\varepsilon$, where L is the resulting length, ε is the length of the yardstick, and $N(\varepsilon)$ is the number of yardsticks used to cover the boundary.

In a second experiment, we change the length of the yardstick ε. We again count a number $N(\varepsilon)$ and calculate the length L by the same formula as above. The first observation we make is that the calculated length L has a

different value compared to the first measurement. For example, if we choose the yardstick length measuring our tree to be the vertical height of the tree, we get a different length compared to measuring the tree with a small yardstick of about 1 cm. The first measurement of the boundary line is a very crude estimation of its actual length. The accuracy of the measurement increases with the decrease in length of the yardstick used. Not only does the accuracy of the measurement increase, but the numerical value of the total length L increases as well. The method of measuring the length of the boundary line by means of a yardstick is called the yardstick method.

Another method for determining the length of a boundary line is the box counting method. In this method, the object is superimposed on a lattice with mesh size ε. If we count the squares which contain a part of the boundary and multiply the number of boxes $N(\varepsilon)$ by mesh size ε, we get an approximated length of the boundary line. Again, we observe that with decreasing mesh size ε, the accuracy of the measurement and the total length L increases. The number of boxes counted in the box counting method is nearly of the same order as the number of yardsticks in the yardstick method.

If the length L increases while the yardstick ε decreases, the question arises of whether there exists a finite length of the boundary of the tree. If the length of the boundary is finite, we expect that the number of yardsticks $N(\varepsilon)$ must increase proportionally to $1/\varepsilon$ (i.e., $N(\varepsilon) = L_N / \varepsilon$). In other words, if the length of the boundary is $L = N(\varepsilon)\,\varepsilon = L_N$, where L_N is a constant for any $\varepsilon \to 0$, we can say that the length is constant. If we apply this mind game to a natural object and count the number of boxes, we observe a completely different behavior.

The measurement of natural objects like blood cells or the bronchial tree using the yardstick or box counting method shows a different relationship between the yardstick length and the number $N(\varepsilon)$. The actual relation observed in experiments ([7.2, 7.3]) is $N(\varepsilon) = a\,\varepsilon^{-D}$, where D is a number greater than 1 for plane objects. If we insert the experimentally observed relation for the number of yardsticks into the length relation $L = N\varepsilon$, we get

$$L(\varepsilon) = N(\varepsilon)\varepsilon = a\,\varepsilon^{1-D}. \tag{7.2.1}$$

This relation applies to any boundary line. For an Euclidean curve which is smooth and differentiable at any point, we expect that parameter a represents the finite length L_N and that dimension D equals 1 as $\varepsilon \to 0$. For natural objects, the dimension D is not equal to 1. The property that the dimension of a natural object is different from its topological dimension was used by Mandelbrot to define the term "fractal" [7.4]. The experimental determination of dimension D follows from the slope of a log-log plot in which the length of the curve is plotted versus the length of the yardstick. The slope of the plot is equal to $1 - D$. In fractal theory, the quantity

$$D = 1 + \frac{\log(L(\varepsilon))}{\log(1/\varepsilon)} - \frac{\log(a)}{\log(1/\varepsilon)} \tag{7.2.2}$$

is called the fractal dimension. This parameter characterizes the plane filling of the curve. The tree example used earlier in this chapter is illustrative for our purposes but too complicated to determine the fractal dimension by analytical methods. Another example of a fractal object is the curve as defined by Koch, who at the turn of the century introduced the mathematical *monster* known as the Koch snowflake. At the same time, other mathematicians, including Cantor, Peano, and Weierstrass, discussed sets of points and curves with very strange properties. An example of the type of curve is given in Figure 7.2.4, which shows the Koch snowflake. Using the Koch curve, we can show how the fractal dimension of such a curve (which is nowhere differentiable) is determined and how self-similarity occurs. First, we will describe the box counting method used to determine the fractal dimension. After this experimental approach, we will return to the more analytic approach for fractal curves.

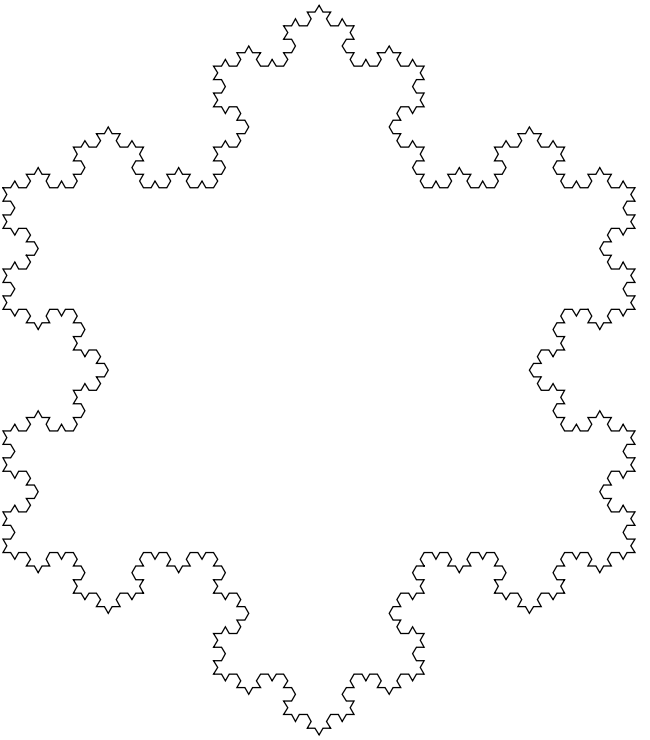


Figure 7.2.4. Koch's snowflake.

7.2.1 Box Counting

As mentioned earlier, the determination of a contour length can be carried out in different ways. One method to determine the total length of a contour is the application of the yardstick method to gain an approximation of the length. Another method which will be elaborated here in more detail is the box counting method. The box counting method gained its name from the counting of disjunct boxes or squares in the plane. The squares or boxes can be replaced by other geometric objects like spheres, ellipsoids, cylinders, and so forth. The explicit form of the used basic measuring element is of minor importance in the estimation of the length of a contour. Here, we use the box counting method to demonstrate its application to plain objects. We apply the box counting algorithm in its simplest form to show how the method works and how we can improve the basic procedure to refine the results.

Box counting is one of the most widely used methods to determine the fractal dimension. Its popularity is largely due to its relative ease of mathematical calculation and empirical estimation. The definition goes back at least to the 1930s and it has been variously termed Kolmogorov entropy, entropy dimension, capacity dimension, metric dimension, logarithmic density, and information dimension. We will always refer to box or box counting dimension to avoid confusion.

Let $\mathcal{S}$ be a nonempty bounded subset of $\mathbb{R}^2$ and let $N(\varepsilon)$ be the number of sets of diameter at most ε which can cover $\mathcal{S}$. We refer to the value as the box counting dimension or box dimension of $\mathcal{S}$ as

$$D = \lim_{\varepsilon \to 0} \left(\frac{\log(N(\varepsilon))}{\log(1/\varepsilon)} \right). \tag{7.2.3}$$

This version of the definition is widely used empirically. To find the box dimension of the set $\mathcal{S}$, we can draw a mesh of squares of side ε and count the number $N(\varepsilon)$ that overlap the set for various small ε. The dimension is the logarithmic rate at which $N(\varepsilon)$ increases as $\varepsilon \to 0$ and can be estimated by the gradient of the graph of $\log(N(\varepsilon))$ against $\log(1/\varepsilon)$.

The box counting method is based on the division of a plane into squares of edge length ε. The box counting method delivers an estimate of the length of a contour by counting the number of boxes $N(\varepsilon)$ of a given size. Each box containing at least one point is counted in $N(\varepsilon)$. Starting with the largest ε scale (the maximal extension of the object) the grid length ε is decreased successively. In a log-log plot of $N(\varepsilon)$ versus ε, a scaling range for self-similar structures is obtained.

To demonstrate how this mathematical definitions works in practice, we will examine each step of the box counting method starting with the generation of an object, the generation of the squares for different ε's, the counting of the relevant boxes, and the determination of the scaling exponent.

First, we start with the generation of the object which we will examine. Suppose we have to measure the contour length of a human cell. The planar projection of a human cell is mainly described by a disturbed circle. We assume that the radial coordinate of a circle of radius 1 is increased by random numbers in the range (0, 0.2) for the x coordinate and (0, 0.1) for the y coordinate. The sequence of points is generated by the following table:

```
points = Table[{Sin[i] + Random[Real,{0,.2}],
       Cos[i] + Random[Real,{0,.1}]}
//N,{i,0,2Pi,.05}];
```

To generate a contour line from these points, we will link each neighboring points by straight lines. This is carried out by the following function generating the contour.

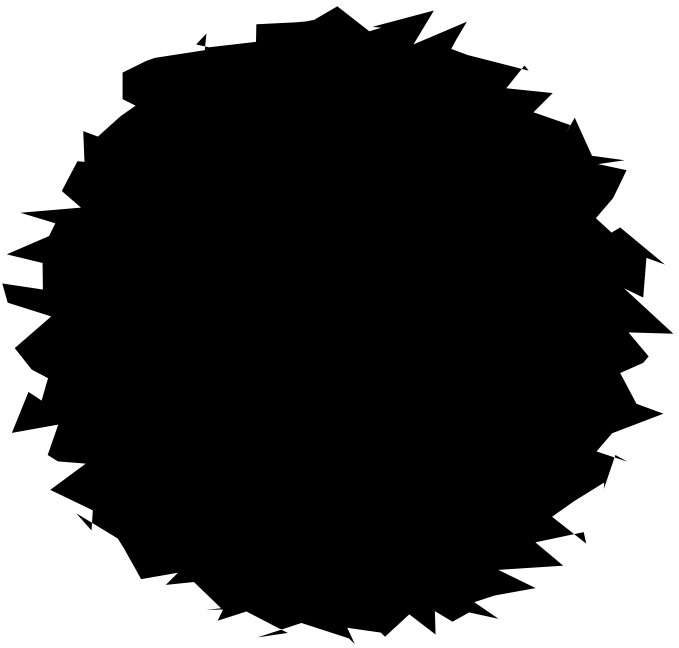
```
Contour[points_]:=Module[{contour},
contour = {};
Do[
  AppendTo[contour,Line[{points[[i]],points[[i+1]]}]
],
{i,1,Length[points]-1}];
AppendTo[contour,Line[{Last[points],First[points]}]];
contour
]
```

The actual contour is then generated by applying this function to the set of points:

```
c1 = Contour[points];
```

A graphical representation of the artificial cell is given next:

```
pl1 = Show[Graphics[Polygon[points]],
AspectRatio->Automatic];
```

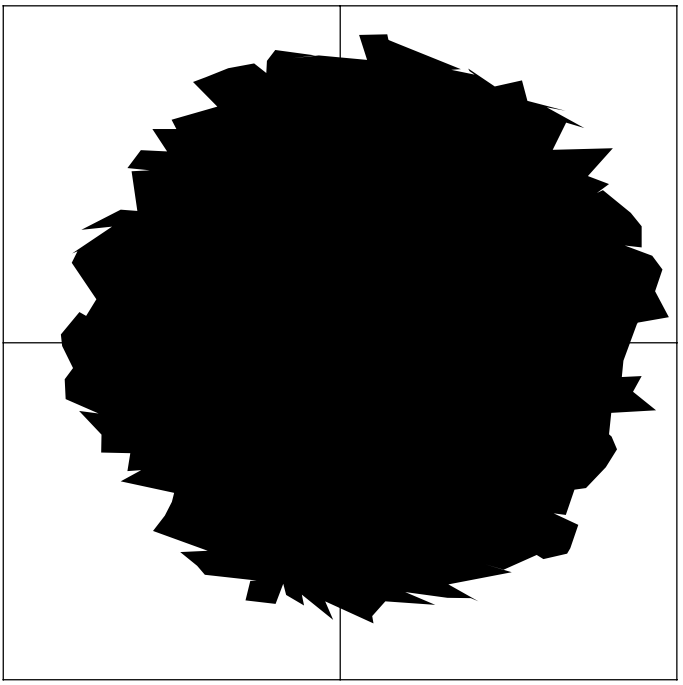


In the next step, we need to generate the grids allowing us to count the occupied squares by the contour. The following function generates a square of total side length $l_{\max}$ divided into subsquares of length ε.

```
Clear[Grid]
Grid[lmax_,eps_]:=Module[{l1={}},
AppendTo[l1,Table[Line[{{-lmax,y},{lmax,y}}],
                {y,-lmax,lmax,2 lmax/eps}]];
AppendTo[l1,Table[Line[{{x,-lmax},{x,lmax}}],
                {x,-lmax,lmax,2 lmax/eps}]];
l1
]
```

Using this function, we can generate an animation showing the principal situation for the measurement process by decreasing the length ε:

```
Do[Show[pl1,Graphics[Grid[1.2,eps]],AspectRatio->Auto
matic,
PlotRange->All],
{eps,2,75,5}]
```



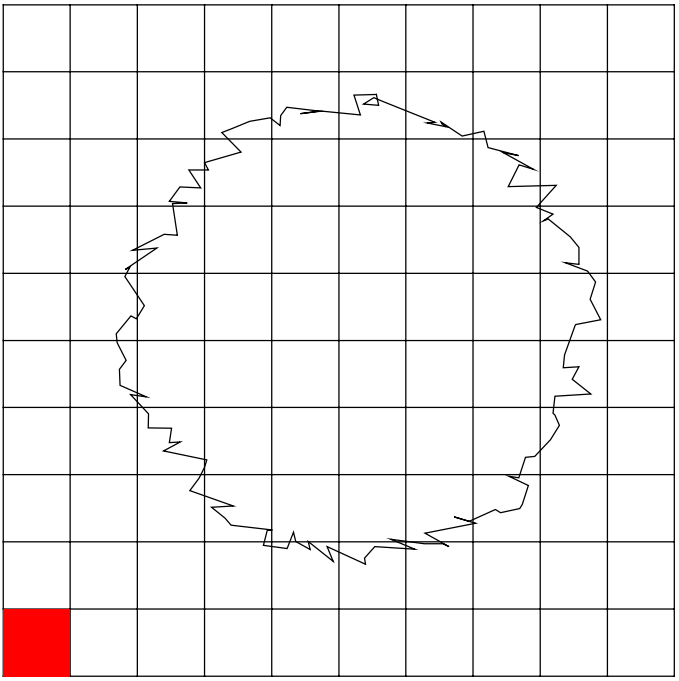
The next step in the determination of the box dimension is to count all squares occupied by the contour line of the cell. For this step, we have to

check whether the contour line intersects with a specific square or the square is empty. The following function scans over the total square and counts the occupied squares:

```
Clear[PointSearchG];
PointSearchG[lmax_,eps_,points_]:=Module[{
deltaeps,xgmin,xgmax,ygmin,ygmax,occupied,presentPoly
},

deltaeps = 2 lmax/eps;

xgmin = -lmax;
xgmax = xgmin + deltaeps;
ygmin = -lmax;
ygmax = ygmin + deltaeps;
occupied = {};

Do[
  Do[
    Do[
If[xgmin <= points[[i,1]] < xgmax &&
   ygmin <= points[[i,2]] < ygmax,
   AppendTo[occupied,{RGBColor[1,1,0],
            Polygon[{{xgmin,ygmin},{xgmax,ygmin},
                    {xgmax,ygmax},{xgmin,ygmax},
                    {xgmin,ygmin}}]} ]
],
    {i,1,Length[points]}];
presentPoly = {RGBColor[1,0,0],
            Polygon[{{xgmin,ygmin},{xgmax,ygmin},
                    {xgmax,ygmax},{xgmin,ygmax},
                    {xgmin,ygmin}}]};
Show[Graphics[Grid[1.5,eps]],
Graphics[presentPoly],
Graphics[occupied],
Graphics[c1],AspectRatio->Automatic];
xgmin = xgmin + deltaeps;
xgmax = xgmin + deltaeps,
  {jx,1,eps}];
xgmin = -lmax;
xgmax = xgmin + deltaeps;
ygmin = ygmin + deltaeps;
ygmax = ygmin + deltaeps,
{jy,1,eps}];
]
```

The application of this function to the cell contour demonstrates the detection and counting of occupied squares

```
PointSearchG[1.5,10,points]
```

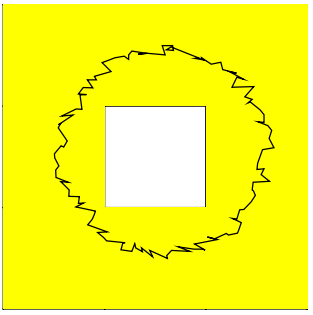


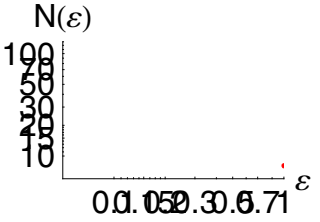
The numeric counterpart to this graphical representation is realized in the following function. This function counts the occupied squares and collects those squares containing a point of the contour in a list. This list is used to determine the total number of squares for a certain box length ε.

```
Clear[PointSearch];
PointSearch[lmax_,eps_,points_]:=Module[{
deltaeps,xgmin,xgmax,ygmin,ygmax,occupied},

deltaeps = 2 lmax/eps;

xgmin = -lmax;
xgmax = xgmin + deltaeps;
ygmin = -lmax;
ygmax = ygmin + deltaeps;
occupied = {};
(* --- detect the occupied squares --- *)
Do[
  Do[
    Do[
If[xgmin <= points[[i,1]] < xgmax &&
   ygmin <= points[[i,2]] < ygmax,
   AppendTo[occupied,{RGBColor[1,1,0],
            Polygon[{{xgmin,ygmin},{xgmax,ygmin},
                    {xgmax,ygmax},{xgmin,ygmax},
                    {xgmin,ygmin}}]} ];
   Return[]
],
    {i,1,Length[points]}];
xgmin = xgmin + deltaeps;
xgmax = xgmin + deltaeps,
  {jx,1,eps}];
xgmin = -lmax;
xgmax = xgmin + deltaeps;
ygmin = ygmin + deltaeps;
ygmax = ygmin + deltaeps,
{jy,1,eps}];
AppendTo[data,{deltaeps,Length[occupied]}];
occupied
]
```

To count the squares for decreasing ε, we iterate this function in a certain range of ε. In addition, we graphically represent the measuring process and the data gained in a sequence of figures.

```
dat = {};
data = {};
j = 1;
Do[
Show[
 GraphicsArray[{Graphics[{Grid[1.5,n],
                          PointSearch[1.5,n,points],
                          c1},AspectRatio->Automatic],
LogLogListPlot[AppendTo[dat,data[[j]]],
     PlotStyle->{PointSize[0.02],RGBColor[1,0,0]},
     PlotRange->{{0.05,1},{6,130}},
     AxesLabel->{"ε","N(ε)"},
DisplayFunction->Identity]}],
AspectRatio->Automatic,DisplayFunction->$DisplayFunct
ion];
j = j + 1,
{n,3,25,5}]
```





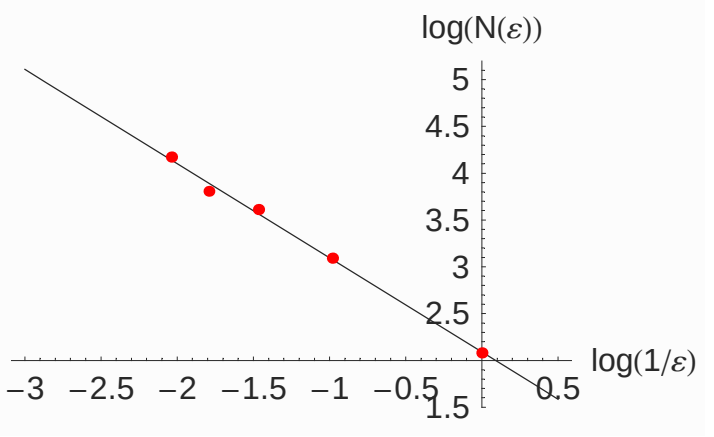
The result shows that the number of occupied squares increases if ε is decreased. Two remarks of caution are appropriate here. Since the representation of the cell contour is given by a relatively small number of points, the accuracy of the gained results are not very high. Second, to increase the reliability of the estimation, the origin of the grid should be changed. From the different measurements, a mean value of the occupied squares can be determined and used in the estimation of the scaling exponent. To estimate the scaling exponent for the present artificial cell contour, we can fit the data to a straight line in a log-log representation of the data.

```
f1 = Fit[Log[dat],{1,x},x]
```

```
2.08955 - 1.0062 x
```

The result shows that a small deviation from a straight line occurs. The scaling law of the artificial cell is shown in the following:

```
Show[Plot[f1,{x,-3,.5},DisplayFunction->Identity],Lis
tPlot[Log[dat],

PlotStyle->{PointSize[0.02],RGBColor[1,0,0]},
          DisplayFunction->Identity],
DisplayFunction->$DisplayFunction,AxesLabel->{"log(1/
ε)","log(N(ε))"}];
```



It is obvious that the gained data can be represented as a straight line in a log-log plot. However, we observe that a scattering of the data points around the line occurs. This chitter has two main causes. First, the small number of data points used in the representation of the cell contour results in fluctuations of the number of occupied squares. Second, there are two limits of the scaling region for small and large values of ε, where a major deviation from the straight line occurs. In the range of large ε we have a cutoff at the diameter of the cell where the scaling relation fails. For very small ε, we reach a region where the discrete representation of the contour cannot be resolved by the box length due to lack of points. Thus, only in

the middle where the box length and the number of points of the contour are commensurable, the scaling behavior is observed. The lower and upper limits in ε are thus determined by the extension of the object itself and the resolution of the contour discretization. The experimental determination of fractal dimensions by means of the box counting method should only be trusted if a range of two or three decades in the box length is spanned.

7.3 The Koch Curve

We have been discussing self-similarity, especially of self-similar curves, but have not explained what is meant by a self-similar object. An example of a self-similar object from geometry is the congruent triangle. Everybody knows that the theorem by Pythagoras, $c^2 = a^2 + b^2$, is satisfied for a right triangle. In this formula, c denotes the hypotenuse and a and b represent the legs of a right triangle (see Figure 7.3.5). The proof of the Pythagorean theorem is given by the self-similar properties of the triangle.

The area of a right triangle is determined by the length of the hypotenuse and the smaller of the two angles between the hypotenuse and its legs ϕ (i.e., $F = f(c, \phi)$). Since F has the dimension of area and c has the dimension of length, we can write $F = c^2\Phi(\phi)$. Drawing the normal line of the hypotenuse through the right angle, we divide the total triangle into two self-similar triangles (see Figure 7.3.5). The areas of the self-similar triangles are $F_1 = a^2\Phi(\phi)$ and $F = b^2\Phi(\phi)$, where $\Phi(\phi)$ is the same function for both (similar) triangles. The sum of the areas F_1 and F_2 is the total area F of the triangle:

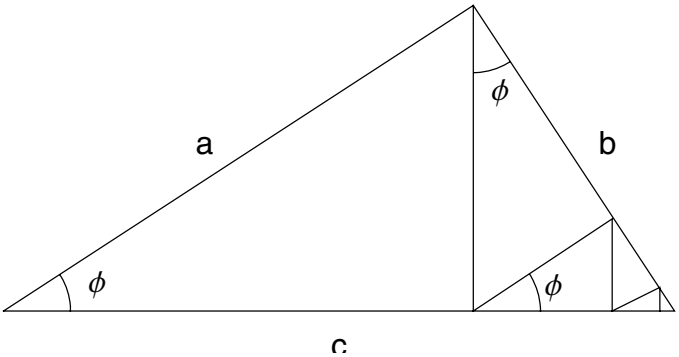


Figure 7.3.5. Self-similarity on a rectangular triangle.

$$F = F_1 + F_2, \tag{7.3.4}$$

$$c^2\,\Phi(\phi) = a^2\Phi(\phi) + b^2\Phi(\phi). \tag{7.3.5}$$

Cancellation of the mutual function yields

$$c^2 = a^2 + b^2 \tag{7.3.6}$$

QED.

This sort of self-similarity is known as congruence in geometry. If we apply this construction again to divide the right triangle for each triangle and repeat the procedure ad infinitum, we get a sequence of triangles which are scaled versions of the original triangle. At each level of division, we find the same triangles, but scaled by a different factor. This behavior of repetition and scaling was used by Helge von Koch to construct the Koch curve.

The initial element of the Koch curve is a straight line of length L_N=1. The first step in constructing the Koch curve is a scaling of the total length by a factor r = 1/3. In the second step, four elements are arranged as shown in Figure 7.3.6. From this figure, we see that the curve loses its differentiability at the connection points of the four lines. These two fundamental steps can be infinitely applied to each of the line elements. In a kth iteration step, we get a total scaling factor of $r_k = (1/3)^k$. The number of line elements increases up to $N_k = 4^k$. The first three steps of this construction are shown in Figures 7.3.6-7.3.8. If we measure the

length of the Koch curve by a yardstick of the same length as the scaling factor $\varepsilon = r$, we find the equation from the length relation $L(\varepsilon) = N(\varepsilon)\,\varepsilon$,

$$D = \frac{\log(N)}{\log(1/\varepsilon)} \tag{7.3.7}$$

and one obtains $D = \log(4)/\log(3) = 1.218\ldots.$ for the Koch curve. Thus, the fractal dimension for a self-similar curve follows from the number of building blocks N of the generator and the scaling factor r, which is used as the yardstick length. The geometrical structure of the line elements is not contained in the fractal dimension because the fractal dimension is not a unique property of a curve. Thus, we get the same fractal dimension for curves with completely different appearances (compare Figures 7.3.8 and 7.3.9).

The Koch curves of the Figures 7.3.6-7.6.9 are constructed in *Mathematica* with the function **Line[]**. We define the generator of the Koch curve in the **Koch[]** function, which is part of the **Koch`** package, and use the *Mathematica* function **Map[]** to generate the higher iterations of the generator (see Section 7.8.2). By keeping the generator and the iteration separate in the creation process of the fractal curve, we are able to mix two or more generators into the iteration process. In Figure 7.3.10 the Koch generator is mixed with a rectangular representation. The first two iterations are done with the original Koch generator. The next two iterations use the rectangular Koch generator. Separating the iteration process from the definition of the fundamental generators allows any mixing of generators in any state of the iteration. In package **Koch`**, we define a number of generators of fractal curves. Their combinations are accessed by the function **Fractal[]**. This function uses a string containing one of the possible fractals as the first argument. The second argument of the function changes the default values of the generators.

Another form of the Koch curve is obtained if we change the base angle α of the triangle in the generator. If we again use four line elements to set up the generator and alter the scaling factor to $r = 1/(2 + 2\cos\alpha)$, we find a fractal dimension of

$$D = \frac{\log 4}{\log 2 + \log(1+\cos\alpha)}. \tag{7.3.8}$$

A representation of the dimension D versus the angle α is given in Figure 7.3.11. In the case of $\alpha = 0$, the dimension is reduced to $D = 1$ and for

$\alpha = \pi/2$, the maximum dimension $D = 2$ occurs. For $D = 2$, we have a plane filling curve. For the specific value $\alpha = 1.4$, the sixth iteration of the Koch curve with a variable base angle is given in Figure 7.3.12.

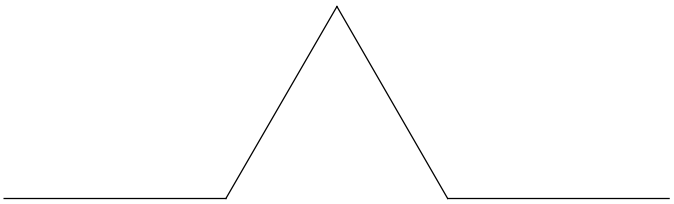


Figure 7.3.6. First iteration of the Koch curve.

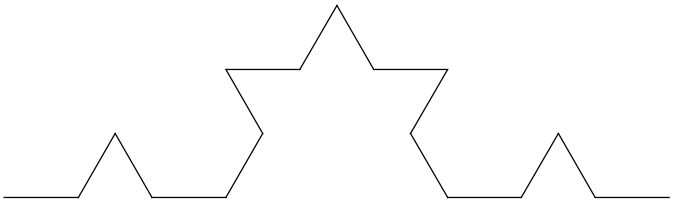


Figure 7.3.7. Second iteration of the Koch curve.

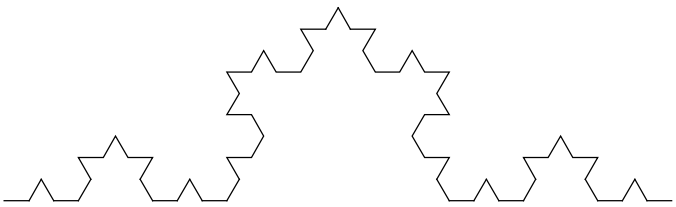


Figure 7.3.8. Third iteration of the Koch curve.

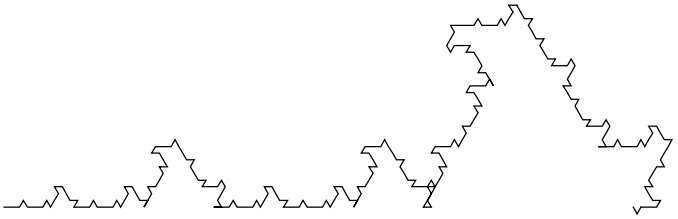


Figure 7.3.9. Fourth iteration of an altered Koch curve. The triangle is located at the right end of the unit base element.

```
Fractal["Mixture"]
```

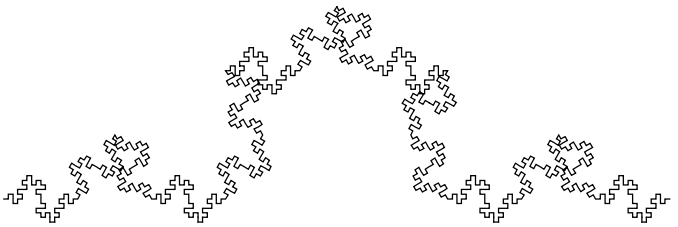


Figure 7.3.10. Mixing of two generators. The first two iteration steps are governed by the original Koch generator. In the last two iteration steps, a rectangular Koch generator is used.

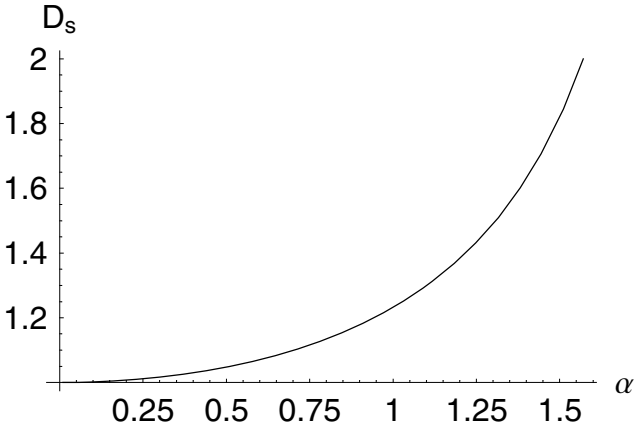


Figure 7.3.11. Change of the fractal dimension under a change of base angle.

```
Fractal["WKoch", Angle -> 1.4, Generations -> 5]
```

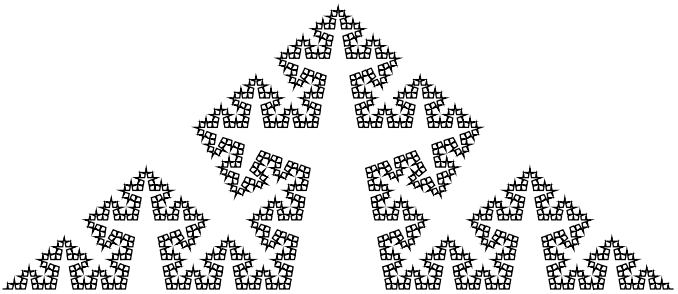


Figure 7.3.12. Koch curve with base angle α =1.4. The scaling factor is r = 0.42736....

7.4 Multifractals

In the previous sections, we discussed structures with mutual scaling factors. This kind of self-similarity is a special case of fractals. A more common type of fractal uses several scaling factors in competition with one another. If in the same system different scaling factors occur with different probabilities, we speak of multifractal behavior. The first step in the construction of a multifractal consists of the division of a set into j components, in which each is scaled by the factor $1/r_j < 1$. We assume that each part of the j-fold set is related to a probability P_j. The probabilities P_j are normalized so that $\sum_{j=1}^{n} P_j = 1$, where n counts the number of subsets. The second step in constructing $k = 2$ is a repetition of the first step applied to each subset. The n subsets are each divided into n subsets and are related to the corresponding probabilities. A graphical representation of this division is given in Figure 7.4.13. The multifractal is created as $k \to \infty$.

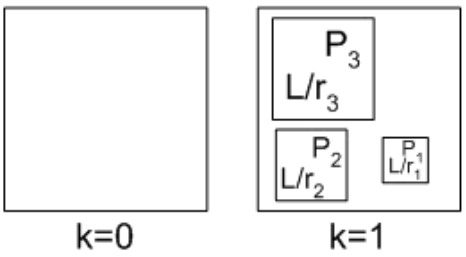


Figure 7.4.13. Representation of a multifractal. The initial state $k = 0$ and the first iteration $k = 1$ are shown. The scaling factors are r_1, r_2, and r_3. The related probabilities are P_1, P_2, and P_3.

The consequence of this construction is that we can divide the total fractal into n parts. Each part of the fractal is scaled by a factor $1/r_j$ and the measure of the jth part is determined by P_j. Using these quantities, we can define one of the characteristic functions of a multifractal by

$$\chi_{q,j}(\varepsilon)=\sum_{i=1}^{N} p_{j,i}^{q}(\varepsilon) = P_j^q \chi_q(\varepsilon r_j), \tag{7.4.9}$$

where $\chi_{q,j}(\varepsilon)$ characterizes the jth part of the fractal by a probability $p_{j,i}(\varepsilon)$ ($p_{j,i}(\varepsilon)$ is the ith probability for the jth part of the total fractal). For the total fractal, we get

$$\chi_q(\varepsilon)=\sum_{j=1}^{n} \chi_{q,j}(\varepsilon). \tag{7.4.10}$$

Using the relation $\chi_q(\varepsilon) = \varepsilon^{(q-1)D_q}$ and Eq. (7.4.9), we get the expressions

$$\chi_{q,j}(\varepsilon) = P_j^q \chi_q(\varepsilon r_j) = P_j^q r_j^{(q-1)D_q} \varepsilon^{(q-1)D_q}, \tag{7.4.11}$$

$$\chi_q = \sum_{j=1}^{n} P_j^q r_j^{(q-1)D_q} \varepsilon^{(q-1)D_q}, \tag{7.4.12}$$

which define the implicit equation for determining the generalized dimension D_q by

$$\sum_{j=1}^{n} P_j^q r_j^{(q-1)D_q} = 1. \tag{7.4.13}$$

Depending on the choice of probabilities P_j and scaling factors r_j, we can use Eq. (7.4.13) to derive several special cases for a multifractal. For $q = 0$ we get the fractal dimension $D = D_0$. This dimension was introduced by Mandelbrot [7.4] for a fractal

$$\sum_{j=1}^{n} r_j^{-D_0} = 1. \tag{7.4.14}$$

For arbitrary q and identical scaling factors $r_j = r$, we get the representation of D_q by

$$\sum_{j=1}^{n} P_j^q \, r_j^{(q-1) D_q} = 1, \tag{7.4.15}$$

$$(q-1) D_q \ln r = -\ln \sum_{j=1}^{n} P_j^q, \tag{7.4.16}$$

$$D_q = \frac{1}{q-1} \frac{\ln \sum_{j=1}^{n} P_j^q}{\ln \frac{1}{r}}. \tag{7.4.17}$$

Once the probabilities P_j and the scaling factors r_j are equal for each individual j, the multifractal properties no longer occur.

Knowing the dependence of D_q on q, alternate representations of the fractal dimensions emerge. By a Legendre transformation, we can introduce

$$(q-1) D_q = (\tau(q)) = q \, \alpha_q - f_q, \tag{7.4.18}$$

where f_q is the multifractal distribution and α_q is the Hölder exponent. The Hölder exponent α_q is defined by the derivative of $\tau(q)$:

$$\alpha_q = \frac{d}{d\,q} \tau(q). \tag{7.4.19}$$

Once we know the fractal dimensions D_q, we are able to determine the Hölder exponent and f_q by relations (7.4.19) and (7.4.18), respectively. Knowing both quantities, we can plot $f = f(\alpha)$ versus α, eliminating q. Calculating the derivative of $\tau(q)$ given in Eq. (7.4.19) causes numerical problems. Finding the numerical derivative of the Legendre transformation of D_q is the main problem in our calculation. In the package **MultiFractal`** (see Section 7.8.3), we use a symmetric difference procedure (see Section 3.5 of Chapter 3) for representing the numerical values of the derivatives of $\tau(q)$. The transformation to τ is defined in the function **Tau[]**. The approximations of derivatives by their differences result in a numerical error, but it is sufficiently small if we choose steps dq in q as a small quantity.

MultiFractal[] calculates the multifractal characteristics. Probabilities P_j and scaling factors r_j are input parameters for this function. The fractal dimension D_q, the function $\tau(q)$, and the Legendre transfor- mation are determined by the functions **Dq[]**, **Tau[]**, and **Alpha[]**, respectively. After their calculation, these quantities are graphically represented by the

Mathematica function **ListPlot[]**. An example of a transformation is given in Figures 7.4.14-7.4.17.

7.4.1 Multifractals with Common Scaling Factor

We now consider a multifractal with a fixed and a mutual scaling factor $r_i = r$. To determine the generalized dimensions D_q, we use Eq. (7.4.17), which gives

$$D_q = \frac{1}{q-1} \frac{\ln \sum_{j=1}^{n} P_j^q}{\ln \frac{1}{r}}. \tag{7.4.20}$$

In the following, we consider a model that contains three independent sets, $n = 3$, characterized by the probabilities $P_1 = 1/5$, $P_2 = 3\,P_1$, and $P_3 = P_1$. If we use relation (7.4.17) for these three processes, we get

$$D_q = \frac{1}{q-1} \frac{\ln(P_1^q + P_2^q + P_3^q)}{\ln(1/r)}. \tag{7.4.21}$$

Normalizing the probability by $P_3 = 1 - P_1 - P_2$ simplifies expression (7.4.21) to

$$D_q = \frac{1}{q-1} \frac{\ln(P_1^q + P_2^q + (1-P_1-P_2)^q)}{\ln(1/r)}. \tag{7.4.22}$$

The numerical results are represented in Figure 7.4.15 which is created by **MultiFractal[{1/5,3/5,1/5},{1/2,1/2,1/2}]**. In the above case, probabilities $P_1 = P_3$ and $P_2 = 3\,P_1$ simplify Eq. (7.4.22) to

$$D_q = \frac{1}{q-1} \frac{\ln(2 + 3^q) + q \ln(P_1)}{\ln(1/r)}. \tag{7.4.23}$$

From relation (7.4.23), we can derive analytic relations for the Hölder exponent α_q, and for the spectrum f_q by using relations (7.4.19) and (7.4.18). We get for α_q the expression

$$\alpha_q = \frac{1}{\ln(1/r)} \left(\frac{3^q \ln 3}{2+3^q} - \ln(P_1) \right). \tag{7.4.24}$$

The spectrum of the fractal dimensions is given by

$$f_q = \frac{1}{\ln(1/r)} \left(q \frac{d}{d\,q} \ln(2 + 3^q) - \ln(2 + 3^q) \right). \tag{7.4.25}$$

Relation (7.4.25) is independent of P_1 and only contains the ratios of the probabilities. Since the expressions for D_q, α_q, and f_q can not be solved explicitly, we use the numerical method implemented in the function

MultiFractal[] to find the solution. Figures 7.4.14–7.4.17 show the results of our calculation. The fractal dimension D_0 of our model is $D_0 = 1.58...$.

Figure 7.4.14 represents the auxiliary function $\tau(q) = (q-1)\,D_q$, which is the basis of the numerical calculations. Figure 7.4.15 contains the representation of the generalized dimension D_q. Relations (7.4.19) and (7.4.18) for f_q and α_q are shown in Figure 7.4.16. We observe that α_q is a monotonically decreasing function and that f_q shows its maximum at $q = 0$. The Legendre transform of these relations results in the function $f(\alpha)$ as shown in Figure 7.4.17. We observe that the values of $f(\alpha)$ are almost equally spaced at the maximum and become denser at the boundaries of the α interval. In the $\alpha_{-\infty}$ limit, the function $f(\alpha)$ tends to 0, but for α_{∞}, a finite value $f(\alpha)$ results. This means that for $\alpha = \alpha_{\infty}$, a finite dimension of the subsets exists which is smaller than D_0 but greater than zero.

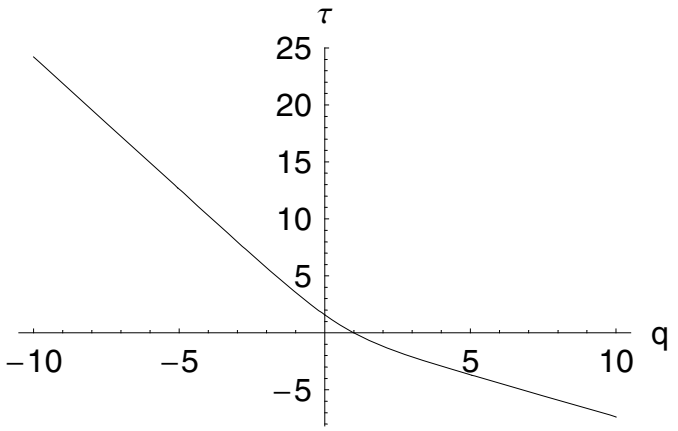


Figure 7.4.14. Function $\tau_q = (q-1)\,D_q$ versus q in the range $q \in [-10, 10]$ for the model fixed by $n = 3$ and $r = 1/2$. The probabilities are $P_1 = 1/5$, $P_2 = 3/5$, and $P_3 = 1/5$.

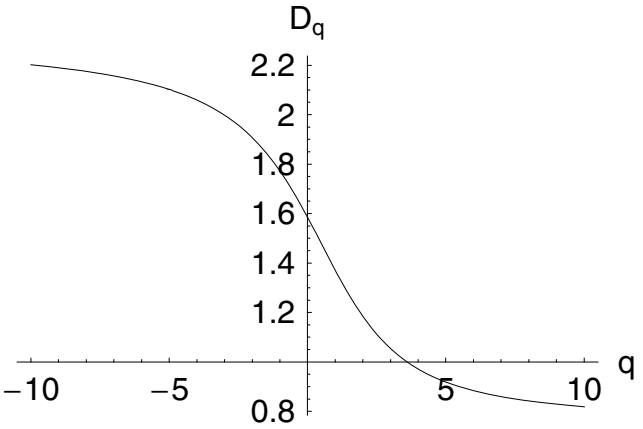


Figure 7.4.15. Generalized fractal dimension D_q for the model $n = 3$, $r = 1/2$, $P_1 = 1/5$, $P_2 = 3/5$, $P_3 = 1/5$, and $q \in [-10, 10]$.

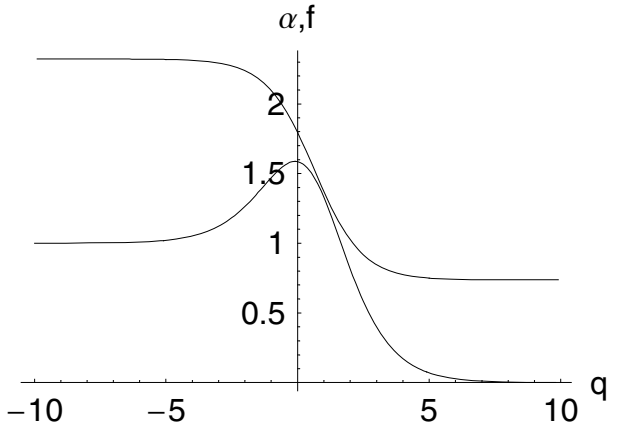


Figure 7.4.16. The exponent α_q (top) and f_q (bottom) versus q for the model $n = 3$, $r = 1/2$, P_1=P_3= 1/5, $P_2 = 3/5$, and $q \in [-10, 10]$.

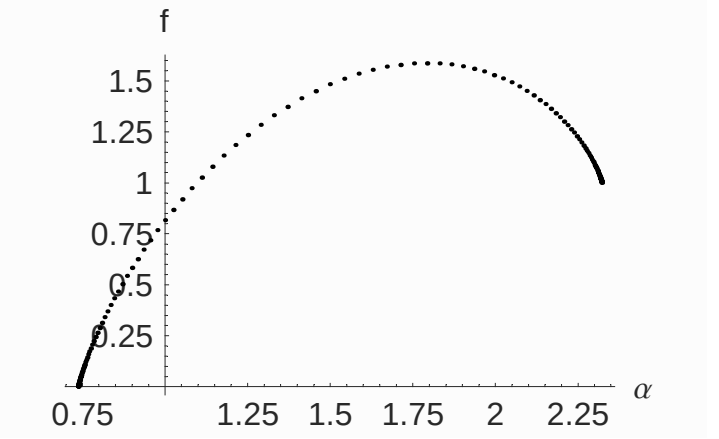


Figure 7.4.17. The fractal spectrum $f(\alpha)$ for a multifractal with $n = 3$, $r = 1/2$, $P_1 = P_3 = 1/5$, and $P_2 = 3/5$.

7.5 The Renormalization Group

Renormalization group theory is useful for describing physical phenomena that show the same behavior on different scales. We assume that p is a quantity measured with a certain accuracy. The same physical quantity is measured in a second experiment, yielding p' with an accuracy which is smaller by a factor of 2 than the first measurement. We assume there is a resolution transformation f_2 connecting the two measurements by

$$p' = f_2(p), \tag{7.5.26}$$

where subscript 2 denotes the order of resolution. If we decrease the resolution of the measurement by another factor of 2, we get the relation

$$p'' = f_2(p') = f_2(f_2(p)) = f_2 \cdot f_2(p) = f_4(p). \tag{7.5.27}$$

The general representation of our resolution transformation for two arbitrary resolutions a and b is given by

$$f_a \cdot f_b = f_{ab}, \tag{7.5.28}$$
$$f_1 = 1, \tag{7.5.29}$$

where 1 represents the identity transform. Applying the resolution transformation to any physical state, a reduced state containing less information is created. Decreasing the resolution from a state with small

resolution is, in general, not possible. In other words, the function f cannot be inverted in general. A set of functions which is not 1 to1 is called a semigroup in mathematics. In physics, the transformation reducing the resolution is called renormalization. (Strictly speaking, f should be called a semirenormalization group.) By definition, the renormalization group is closely related to the definition of a fractal.

Since a fractal stays invariant under a scaling transformation, it is evident that a fractal also stays invariant under a renormalization transformation. Chronologically both terms – fractal and renorma- lization – were introduced in the 1970s. Both describe the behavior of an object with changing scales. The difference between the two terms is that a fractal is based on geometrical properties, whereas renorma- lization considers the physical properties in a scaling process. However, recent developments in fractal theory also consider physical properties, whereas renormalization theory is also applied to geometric objects. Consequently, the distinction between a fractal and renormalization theory is disappearing.

Renormalization theory is a tool describing critical phenomena like phase transitions in a liquid. Liquids, for example, possess a critical point in their phase diagrams. Renormalization theory is used to describe the behavior of the system in the immediate neighborhood of the critical point. Let us consider a state of liquid below the critical point where a mixture of liquid and gas coexists. Below the critical point, the mixture contains more liquid than gas. If we "coarse grain" our observation, we get a system which is dominated by the liquid phase. The combination of cells containing liquid and gas components produce a liquid state under renormalized conditions. The repetition of the "coarse graining" process results in a global liquid state. If, on the other hand, the initial state of the phase diagram contains more gas than liquid, the renormalization results in a gaseous state.

In another example, we consider the renormalization procedure in connection with percolation theory. Percolation theory is a theory describing the connections in a network of random links. The theoretical basis for this theory was created by P.G. de Gennes [7.7], winner of the 1991 Nobel Prize. He applied percolation theory for disordered materials in polymer science. Percolation phenomena are widespread in nature, occurring in biological, chemical, and physical systems.

Percolation theory allows the connection of two different boundaries with a cluster of particles on a lattice. Specifically, let us examine the transport of electrons through a porous medium which is located between two metal plates. The transport of the charge is carried by a percolation cluster connecting both plates. In order to study the transport of electrons, picture the simulation of a current in a porous medium on a two-dimensional lattice. Atoms carry the charge on the lattice. The atoms are randomly scattered. Using the probability p, an atom at a certain location on the lattice can be located.

The renormalization step on this lattice is defined by the rule valid for a 2 × 2 sublattice, which is called the virtual lattice. We are able to replace the region of the virtual lattice with a new lattice point in the renormalized lattice. The resultant lattice is called the superlattice. The (2 × 2) cells of the virtual lattice are called blocks (see Figure 7.5.18).

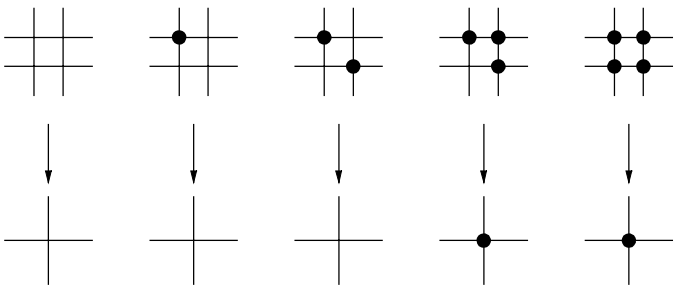


Figure 7.5.18. Renormalization steps with (2×2) blocks.

The transition from the original lattice to the superlattice follows rules for replacing old atoms with new ones. The simplest rule applies if we have four atoms in a block. In this case, the new point in the super lattice is an atom. If we only have three atoms in a block, another new atom emerges on the superlattice. Accordingly, percolation clusters can form horizontally as well as vertically. If a block only contains one or two particles, it is impossible for a percolation cluster to occur which is independent of any direction. Therefore, no atom appears on the superlattice. Applying the transition rules as defined in a probability projection, we can write down the probability of finding an atom on the superlattice by

$$(p') = f_2(p) = p^4 + 4\,p^3(1 - p). \tag{7.5.30}$$

The first term describes the probability that all four atoms are present in a block. The second term takes into account the four possible arrangements of three atoms in a block. Since we now know the function f_2, we can determine the phase transition by using the properties of f_2.

Generalizing relation (7.5.30) for a lattice with $n = b \times b$ locations on which m empty points exist is given by the expression

$$f((p)^n_m) = \sum_{i=0}^{m} \binom{n}{i} p^{(n-i)}(1 - p)^i. \tag{7.5.31}$$

Equation (7.5.31) specifies the probability on a lattice if the block contains n locations of which all m points are empty.

The critical point p_c on the (2×2) lattice is defined in such a way that the probability will not change under the transformation f_2. The fixed point p_c is derived from the relation

$$p_c = p_c^4 + 4\,p_c^3(1 - p_c) \tag{7.5.32}$$

with solutions

$$p_c = \left\{0,\ 1,\ \frac{1 \pm \sqrt{13}}{6}\right\}. \tag{7.5.33}$$

The numerical values of the third and fourth solutions are -0.434 and 0.768. Since p is a probability which is always greater than 0, we have to exclude the solution $p_c = -0.434$ from the physical solution set. The cases $p_c = 0$ and $p_c = 1$ are trivial since they correspond to an empty or occupied lattice. The remaining value of $p_c = 0.768$ seems to be the critical value for which a percolation takes place. We observe a gap if we compare the theoretical value with the value $p_c = 0.59$ yielded by computer simulations. However, the experimentally determined value of $p_c = 0.752$ is fairly close to its theoretical counterpart [7.5, 7.6]. A graphical representation of the critical probability versus the number of lattice points is given in Figure 7.5.19. The curves in this figure represent different superlattices.

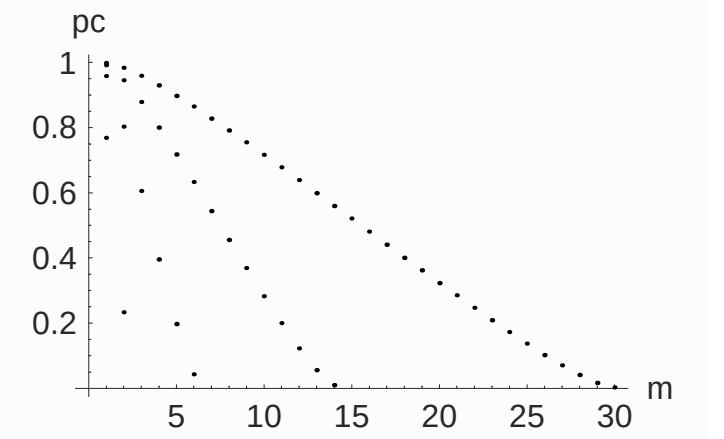


Figure 7.5.19. Percolation probability for super lattices with 4, 8, 16, and 32 lattice points. The probability is plotted versus the number of empty lattice points.

To see how other solutions of (7.5.33) are reached, we first consider the case $p < p_c$. In this case, we get the inequalities

$$p_c > p > f_2(p) > f_{2^2}(p) > \cdots > f_{2^n}(p). \tag{7.5.34}$$

Relation (7.5.34) shows that the probability p decreases in each renormalization step. After infinitely many renormalization steps, we get the limit $f_\infty(p) = 0$. In other words, a point with an atom somewhere on the lattice is impossible, since the lattice is empty.

For the case $p > p_c$, the reverse occurs and $f_\infty(p) = 1$. After infinitely many renormalization steps, the superlattice is fully occupied. This means that all initial values in the neighborhood of $p_c = 0.768$ will tend to be $p_c = 0$ or $p_c = 1$. The fixed point at $p_c = 0.768$ is unstable (see Figure 7.5.20).

Figure 7.5.20. Stability of the fixed points in the renormalization procedure.

In the following, we determine the fractal dimension of the cluster at percolation $p_c = 0.768$. If an atom is present on the superlattice, we know

that there are either three or four atoms in a block. The expectation value $p_c\, N_c$ of occupied lattice points is thus given by

$$(p_c\, N_c) = 4\, p_c^4 \;+\; 3 \cdot 4\, p_c^3\, (1 \;-\; p_c) \tag{7.5.35}$$

$$\Longleftrightarrow \quad N_c = 4\, p_c^3 \;+\; 3 \cdot 4\, p_c^2\, (1 \;-\; p_c), \tag{7.5.36}$$

where N_c is the mean value of atoms provided that the superlattice is occupied. The general formula for a square grid has the representation

$$N_c\, ((p)_m^n) = \sum_{i=0}^{m} \binom{n}{i} (n - i)\, p^{(n-i-1)} (1 - p)^i\,. \tag{7.5.37}$$

Equation (7.5.37) counts the mean number of occupied lattice points for a square lattice with n locations and with m empty locations. A graphical representation of N_c versus m is given in Figure 7.5.21. The curves in the figure represent different block sizes.

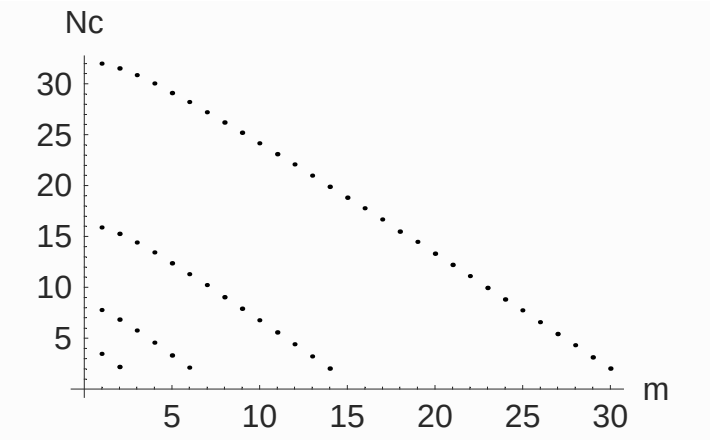


Figure 7.5.21. Mean number of occupied locations in a square lattice. The block size is 4, 8, 16, and 32 as shown in the curves from bottom to top.

The meshsize in the superlattice is twice that of the original lattice. If we divide the meshsize by 2 in the superlattice, we observe N_c atoms, the average in the original lattice. Generalizing this observation when reducing the observation scale by a factor of $1/b$ yields

$$N_c\,(b) \;=\; b^{-D}. \tag{7.5.38}$$

In the example discussed above, $b = 2$. From relation (7.5.38) we get for the specific case,

$$D = \frac{\ln N_c}{\ln 2} = 1.79, \tag{7.5.39}$$

where the constant D represents the fractal dimension of the percolation cluster. $D = 1.79$ is in good agreement with the value found in computer simulations. However, the experimental value of the fractal dimension is different ($D = 1.9$ [7.5]). Figure 7.5.22 represents the fractal dimension compared to the empty lattice points for several block sizes. We observe from this figure that the fractal dimension decreases with an increase of empty lattice points. The dimension D approaches 2 if the lattice is almost fully occupied.

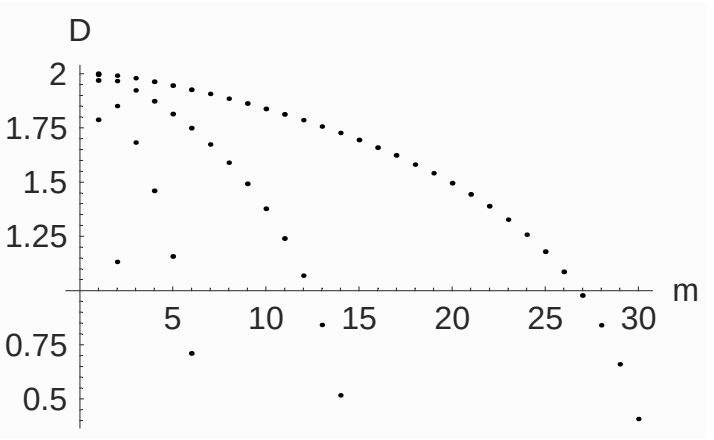


Figure 7.5.22. Fractal dimension of a percolation cluster versus empty locations for four block sizes 4, 8, 16, and 32.

In our previous considerations, we calculated the fractal cluster dimension at the critical point. Other interesting quantities in the neighborhood of the critical point are the critical exponents. The critical exponents are easy to derive if we again use the renormalization procedure. As an example, we determine the critical exponent of the correlation length.

For $p < p_c$ and p in the neighborhood of p_c, we can represent the correlation length ξ by

$$\xi = \xi_0 \mid p_c - p \mid^{-\nu}, \tag{7.5.40}$$

where ξ_0 is a characteristic length of the system (e.g., the meshsize). If we consider the rescaled superlattice, we find for the invariant correlation length,

$$\xi = \xi_0' \mid p_c - p' \mid^{-\nu} \tag{7.5.41}$$

with $\xi_0' = 2\,\xi_0$. From Eq. (7.5.39) and (7.5.40), we derive the critical exponent ν:

$$\nu = \frac{\log(2)}{\log(p_c) - p'/(p_c - p)}. \tag{7.5.42}$$

At the limit where p and p' tend to p_c, we can replace

$$\frac{p_c - p'}{p_c - p} \longrightarrow \frac{\partial p'}{\partial p} \Big|_{p=p_c}. \tag{7.5.43}$$

The final result for the critical exponent is

$$\nu = \frac{\log(2)}{\log\left(\frac{\partial f_2(p)}{\partial_p}\right)\Big|_{p=p_c}}. \tag{7.5.44}$$

Using the functional relation f_2 in Eq. (7.5.44), the numerical value $\nu = 1.4$ is close to the experimental value of $\nu = 1.35$.

The renormalization group theory is useful for determining fractal and critical properties of a system. Note that the renormalization theory is a kind of perturbation theory. Errors occur in the renormalization procedure when defining renormalization rules. For example, blocks containing more than two atoms are replaced by atoms on the superlattice, whereas blocks containing one or two atoms are given by a void. This coarse graining process is the source of renormalization errors; that is, we create a crude picture of the original lattice in the superlattice containing links and gaps on sites where no links were present in the original lattice (see Figure 7.5.23). To minimize errors, we use large block sizes. If we use blocks of size b, we have b^2 lattice points. The number of states in the block is given by 2^{b^2} and increases rapidly with block size b. From a practical point of view, $b = 4$ is the upper limit for which we can calculate the renormalized function f_b.

The package **Renormalization`** (see Section 7.8.4) contains the functions **Nc[]** for determining the mean number of occupied lattice points, **Dim[]** for calculating the fractal dimension, and **Pcrit[]** for calculating the critical probability of percolation. Function **RenormPlot[]** allows the graphical

representation of the above functions. Examples of the plots are given in Figures 7.5.19, 7.5.21 and 7.5.22.

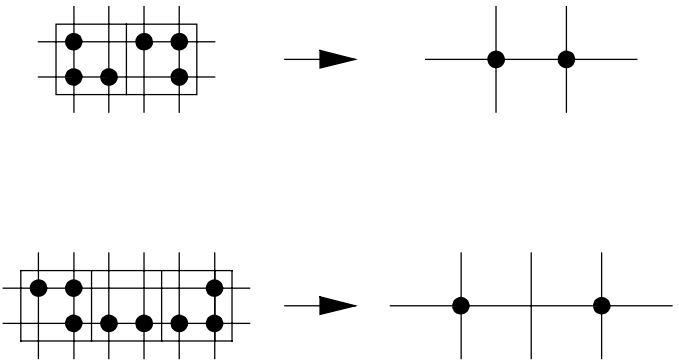


Figure 7.5.23. Errors in the renormalization of a 2×2 lattice.

7.6 Fractional Calculus

Fractional calculus, contrary to fractal geometry, is an old subject in mathematics. This kind of calculus is useful to describe phenomenological models for different chemical and physical processes. Among these processes are temporal relaxations of polymeric material and diffusion processes in space and time. Fractional calculus is an approach to mathematically describe natural phenomena which are mainly connected to power law behavior in the limit of large arguments. The power-law behavior of large arguments for natural systems is typically accompanied by a deviation from these power laws for small arguments. Thus, fractional calculus is a tool to interpolate between these two regimes by means of fractional differentiations.

7.6.1 Historical Remarks on Fractional Calculus

The term *fractional calculus* is by no means new. It is a generalization of the ordinary differentiation by noninteger derivatives. The subject is as old as the calculus of differentiation and goes back to times when Leibniz (see Figure 7.6.24), Gauß, and Newton invented this kind of calculation. In a letter to L`Hospital in 1695, Leibniz raised the following question:

Figure 7.6.24. Gottfried Wilhelm von Leibniz: born July 1, 1646; died November 14, 1716.

Can the meaning of derivatives with integral order $d^n y(x)/dx^n$ be generalized to derivatives with nonintegral orders, so that, in general, $n \in \mathbb{C}$? This question goes back to a query of Bernoulli, who was interested in the noninteger differentiation of a product. The story goes that L`Hospital was somewhat curious about that question of Leibniz and replied by another question. What if $n = \frac{1}{2}$? Leibniz in a letter dated September 30, 1695 replied: *Il y a de l'apparence qu'on tirera un jour des consequences bien utiles de ces paradoxes, car il n'y a gueres de paradoxes sans utilité.* The translation reads: *It will lead to a paradox, from which one day useful consequences will be drawn.* The question raised by Leibniz for a fractional derivative was an ongoing topic in the last 300 years. Several mathematicians contributed to this subject over the years. People like Liouville, Riemann, and Weyl made major contributions to the theory of fractional calculus.

In fact, a fractional derivative is useful for some types of function. For example, let us consider the nth derivative of a power x^m. We know that the general expression for the nth derivative is given by

$$\frac{d^n x^m}{\mathrm{d}x^n} = \frac{m!}{(m-n)!} x^{m-n}. \tag{7.6.45}$$

We also know that a factorial is connected with Euler's Γ function by the relation $n! = \Gamma(n+1)$. Replacing the factorials in Eq. (7.6.45) by the Γ function, we can write

$$\frac{d^n x^m}{\mathrm{d}x^n} = \frac{\Gamma(m+1)}{\Gamma(m-n+1)} x^{m-n}. \tag{7.6.46}$$

This representation is equivalent to Eq. (7.6.45); however, it contains the potential of a generalization. We know that the Γ function is defined for continuous arguments over the complex domain. If we now change the integer value of n to a number $q \in \mathbb{C}$, we are able to generalize the meaning of an integer differentiation to a noninteger form. We can even define a complex differentiation. Replacing n by q in Eq. (7.6.46) results in general in

$$\frac{d^q x^m}{\mathrm{d}x^q} = \frac{\Gamma(m+1)}{\Gamma(m-q+1)} x^{m-q}. \tag{7.6.47}$$

Relation (7.6.47) has a well-defined meanin; however, it is restricted to powers x^m. However, if we try to fractionally differentiate such simple functions with *Mathematica*, we end up with the following result:

```
∂{x,1/2} x^2
```

```
D::dvar :
  Multiple derivative specifier {x, 1/2} does not have the form
    {variable, n} where n is a nonnegative machine integer.
```

$$\partial_{\{x,\frac{1}{2}\}} x^2$$

This shows us that *Mathematica* is not capable of dealing with fractional differentiation orders. The developer of *Mathematica*, however, designed the system in such a way that the user can extend the definition of derivatives. This extension will be our subject in the following. Telling *Mathematica* that fractional derivatives of powers are useful mathematical constructs is realized by the following lines:

```
Unprotect[D];
```

First, unprotect the differentiation and then add a new definition:

```
D[x_^m_., {x_, q_}] :=
  Gamma[m + 1]/Gamma[m - q + 1] x^(m-q) /; Head[q] == Real ||
   Head[q] == Rational || Head[q] == Complex
```

Protect the differential operator again:

```
Protect[D];
```

The definition of the fractional derivative of powers is based on Eq. (7.6.47) and restricts the order of differentiation either to the rational, the real or the complex numbers. An example for a rational number reads

```
∂{x, 1/2} x
```

$$\frac{2\sqrt{x}}{\sqrt{\pi}}$$

If we set the order of differentiation q to a real number, we find

```
∂{x,2.1} x^2
```

$$\frac{1.87156}{x^{0.1}}$$

Even if we use complex numbers differentiation order, we get a result:

$\partial_{\{x,11.5+I\}}\, x^4$
$(57152.1 - 143371.\, i)\, x^{-7.5-i}$

This kind of formula was discussed by Lacroix in 1819 [7.8] based on the work by Euler in 1738 [7.9]. In retrospect, these formulas are the first analytical answer of Leibniz's question on fractional derivatives. The answer lied 100 years dormant and needed the work of Euler to get a preliminary answer. The story on fractional calculus continued with contributions from Fourier, Abel, Liouville, Riemann, and Weyl. For a historical survey, the reader can consult the books of Oldham and Spanier [7.10] or Miller and Ross [7.11]. The historical developments culminated in two main calculi based on the work of Riemann [7.12] and Liouville [7.13] on the one hand and on the work of Weyl [7.14] on the other hand. Both formulations are connected and Weyl's calculus forms a subset of the Riemann–Liouville (RL) calculus. In Section 7.6.2 we will discuss the RL calculus. Section 7.6.3 is concerned with the Mellin transform used in the solution of fractional differential equations. Section 7.6.4 discusses the solution of different fractional differential equations.

7.6.2 The Riemann–Liouville Calculus

The development of fractional calculus within the framework of classical functions is well known and no purpose would be served by a detailed exposition. However, the present subsection has the aim to provide the reader with the basic tools to carry out such calculations by computer. We not only present the theoretical background of the calculus but also show how symbolic computation is instrumental in calculating fractional expressions. Most of the basic analysis is discussed in the book by Oldham and Spanier [7.10]. The more theoretical issues as well as historical remarks are collected in the book by Miller and Ross [7.11].

In the previous subsection, we introduced the fractional derivative by heuristics using properties of Euler's Γ function. In this subsection, we will

define an operator to calculate fractional derivatives. This operator is based on works by Riemann and Liouville (RL). Paradoxically, the basis of this differential operator is not a derivative but an integral. However, we can understand an integration as a differentiation if we introduce a differentiation with negative exponents. For example the negative first-order derivative is defined by

$$\frac{d^{-1}}{\mathrm{d}x^{-1}} f(x) := \int_0^x f(t)\, dt. \tag{7.6.48}$$

The negative second-order derivative is

$$\frac{d^{-2}}{\mathrm{d}x^{-2}} f(x) := \int_0^x \int_0^t f(s)\, ds\, dt \ldots \tag{7.6.49}$$

The negative order of differentiation means nothing more than an integration. Higher orders of differentiation are calculated by nesting the integrals on the right-hand side. We will abbreviate this kind of recursion by the symbol $\mathcal{D}_{0,x}^{-n}$, where n is a positive integer. Thus, Eq. (7.6.48) is reduced to

$$\mathcal{D}_{0,x}^{-1} f(x) = \int_0^x f(t)\, dt. \tag{7.6.50}$$

The symbol $\mathcal{D}_{0,x}^{-n}$ contains the complete information for the calculation of the negative differential in a nutshell. The lower two indices denote the lower and upper boundaries of the integral. The superscript represents the order of differentiation. A weak generalization of the above notation is gained if we allow an arbitrary starting point a as the lower boundary in the integral; that is,

$$\mathcal{D}_{a,x}^{-1} f(x) = \int_a^x f(t)\, dt. \tag{7.6.51}$$

If we consider the nth derivative $\mathcal{D}_{a,x}^{-n}$ of an arbitrary function $f(x)$, we write

$$\mathcal{D}_{a,x}^{-n} f(x) = \int_a^x \int_{a\ n-1}^{x} f(x_0)\, dx_{n-1} \ldots dx_0. \tag{7.6.52}$$

Recalling Cauchy's integral formula

$$\frac{d^n}{\mathrm{d}x^n} f(x) = \frac{n!}{2\pi \mathrm{i}} \int_C (\zeta - z)^{-n-1} f(\zeta)\, d\zeta, \tag{7.6.53}$$

we can reduce Eq. (7.6.52) to

$$\mathcal{D}_{a,x}^{-n} f(x) = \frac{1}{(n-1)!} \int_a^x (x - x_0)^{n-1} f(x_0)\, dx_0. \tag{7.6.54}$$

Using the well-known relations of the Γ function and factorials discussed in the previous subsection, we can generalize the result to an arbitrary

order of fractional differentiation by replacing $n!$ by $\Gamma(n+1)$. The general formula follows thus by

$$\mathcal{D}_{a,x}^{-q} f(x) = \frac{1}{\Gamma(q)} \int_a^x (x - x_0)^{q-1} f(x_0)\, d x_0 \qquad \text{with} \quad \mathrm{Re}(q) > 0. \tag{7.6.55}$$

This kind of operator is denoted as the Riemann (R) version of the fractional integral by Miller and Ross [7.11]. The Liouville (L) version of this operator follows if we replace the lower boundary a of the integral by $-\infty$; that is, $\mathcal{D}_{-\infty,x}^{-q} f(x)$ is called the Liouville fractional integral. A sufficient condition that this integral converges is that $f(-x) = o(x^{-q-\epsilon})$ for $\epsilon > 0$ and $x \to \infty$. The special case where $a = 0$

$$\mathcal{D}_{0,x}^{-q} f(x) = \frac{1}{\Gamma(q)} \int_0^x (x - x_0)^{q-1} f(x_0)\, d x_0, \qquad q > 0, \tag{7.6.56}$$

is known as the Riemann–Liouville (RL) fractional integral. A sufficient condition that the RL integral converges is given by $f(1/x) = O(x^{1-\epsilon})$ for $\epsilon > 0$. Functions satisfying this relation are called functions of the Riemann–Liouville type. For example, the functions x^α with $\alpha > -1$ and a constant belong to this class of functions. We recognize that the different definitions of Riemann–Liouville fractional integrals differ only in the lower boundary of the integral. The reader might suppose that this small difference is of minor importance. The following subsection will demonstrate that this assumption is not correct. The change of the lower boundary has very far-reaching consequences in the calculation of fractional derivatives.

So far, we introduced the notation of the fractional integral. A fractional derivative is connected with a fractional integral by introducing a positive order of differentiation in the operator $\mathcal{D}_{a,x}^{-q}$. This shift of order can be obtained by introducing an ordinary differentiation followed by a fractional integration. We thus define a fractional differentiation by

$$\mathcal{D}_{a,x}^{s} f(x) := \left(\frac{d^n}{\mathrm{d}x^n}\right) \mathcal{D}_{a,x}^{-(n-s)} f(x) \qquad \text{with} \quad n \in \mathbb{N},\ s > 0,\ n - s > 0. \tag{7.6.57}$$

In this Riemann notation, the fractional derivative depends on a lower boundary a of the integral. This dependence disappears if we consider only the RL operator with $a = 0$.

Up to the present point, we discussed the essentials of the theory of RL integrals. If we intend to use computer algebra in connection with RL operators, we need to know how RL operators are implemented. Thus, the next step is to create a function in *Mathematica* which carries out the calculation. We call this function **RiemannLiouville[]**. Since the RL integral is applied to functions depending on one independent variable, say x, we need to supply this information to the function. Another quantity which must be given by the user is the order of differentiation q. In addition to these two input variables, we need information on the lower boundary of the integration interval. Thus, our function needs, in addition to the function on which we apply the RL operator, three input quantities. The lower boundary is superfluous if we treat a RL integral. The following definition of the Riemann–Liouville fractional integral incorporates the theoretical considerations discussed above:

```
Remove[RiemannLiouville];
RiemannLiouville[1, {x_, order_, a_: 0}] :=
  (x - a) ^ -order / Gamma[1 - order];
(*--- main function --- *)
RiemannLiouville[f_, {x_, order_, a_: 0}] :=
 Block[{n, int, y},
        If[NumericQ[order] && Simplify[order > 0],
   n = Floor[order]; q = order - n];
        int = Integrate[(x - y)^-q-1 (f /. x → y),
    {y, a, x}, GenerateConditions → False];
        D[int / Gamma[-q], {x, n}] /; FreeQ[int, y]
        ]
```

At this stage, we know how functions are treated by a RL integral. Before we apply **RiemannLiouville[]** to a mathematical problem or use it in physical models, we introduce some general properties of the fractional derivative. These properties are important for manual as well as for automatic calculations. They also serve to extend the properties of the function **RiemannLiouville[]**.

7.6.2.1 Properties of Riemann-Liouville Operators

The main properties needed in an implementation of RL operators are linearity and the composition rule. These two properties are basic properties in addition to the Leibniz rule of differentiation and the chain rule. Let us discuss these properties in more detail. In the implementation of the mathematical properties, linearity and the composition of derivatives are of importance. The other two relations are of minor practical importance.

1 Linearity

Linearity is one of the basic properties of a RL operator. This property guarantees that the superposition of a RL operators applied to different functions is the same as the application of the RL operator on the superposition of functions. Linearity of a RL operator means

$$\mathcal{D}^s_{a,x}(\alpha f(x) + \beta g(x)) = \alpha \mathcal{D}^s_{a,x} f(x) + \beta \mathcal{D}^s_{a,x} g(x), \tag{7.6.58}$$

with α and β as real constants. Relation (7.6.58) is implemented by two functions. The first function removes common constants from the argument of the input function:

```
RiemannLiouville[c_ f_, {x_, order_, a_: 0}] :=
  c RiemannLiouville[f, {x, order, a}] /; FreeQ[c, x];
```

The second part of the linearity represents a superposition of two functions. This property is implemented as

```
RiemannLiouville[f_ + g_, {x_, order_, a_: 0}] :=
 RiemannLiouville[f, {x, order, a}] +
  RiemannLiouville[g, {x, order, a}]
```

Both definitions combined represent relation (7.6.58). Linearity of the RL operator means that the operator $\mathcal{D}^s_{a,x}$ can be distributed through the terms of a finite sum; that is,

$$\mathcal{D}^s_{a,x} \sum_{i=0}^{n} f_i(x) = \sum_{i=0}^{n} \mathcal{D}^s_{a,x} f_i(x). \tag{7.6.59}$$

Another important relation is the composition rule of fractional differentiation.

2 Composition Rule

In the case of RL integrals for $\mu, \nu > 0$ and $f(x)$ continuous, the relation

$$\mathcal{D}_{0,x}^{-\mu}\, \mathcal{D}_{0,x}^{-\nu}\, f(x) \;=\; \mathcal{D}_{0,x}^{-(\mu+\nu)}\, f(x) \tag{7.6.60}$$

holds.

The composition rule combining two fractional derivatives of different order is

$$\mathcal{D}_{a,x}^{s}\, \mathcal{D}_{a,x}^{p}\, f(x) \;=\; \mathcal{D}_{a,x}^{s+p}\, f(x), \tag{7.6.61}$$

with $p < 0$ and $f(x)$ finite at $x = a$. This property is another rule to extend the definition of the function **RiemannLiouville[]**. The following lines represent the above relation

```
RiemannLiouville[ RiemannLiouville[f_,
   {x_, order1_, a_: 0}], {x_, order2_, a_: 0}] :=
 RiemannLiouville[f, {x, order1 + order2, a}] /;
  order1 < 0
```

In the case of $p > 0$, the following relation holds:

$$\mathcal{D}_{a,x}^{s}\, \mathcal{D}_{a,x}^{p}\, f(x) \;=\; \mathcal{D}_{a,x}^{s+p}\, f(x) - \mathcal{D}_{a,x}^{s+p}(f(x) - \mathcal{D}_{a,x}^{-p}\, \mathcal{D}_{a,x}^{p}\, f(x)) \tag{7.6.62}$$

where the last term is

$$\mathcal{D}_{a,x}^{-p}\, \mathcal{D}_{a,x}^{p}\, f(x) = f(x) \;- \sum_{k=1}^{m} c_k\, x^{p-k}, \tag{7.6.63}$$

with $0 < p \le m < p + 1$. The constants c_k in Eq. (7.6.63) are constants of integration. In the case of the RL integral ($a = 0$), these constants are given by

$$c_k \;=\; \frac{1}{\Gamma(p - \mathcal{D}k + 1)}\; \mathcal{D}_{0,x}^{p-k}\; f(x)\; \Big|_{x=0}. \tag{7.6.64}$$

The difference of $p > 0$ or $p < 0$ can be demonstrated by the example

$$\mathcal{D}_{a,x}^{1}\, \mathcal{D}_{a,x}^{-1}\, f(x) \;=\; f(x) \tag{7.6.65}$$

for $p < 0$ and

$$\mathcal{D}^{-1}_{a,x}\,\mathcal{D}^{1}_{a,x}\,f(x) \;=\; f(x)+c \tag{7.6.66}$$

with c a constant. This example also demonstrates the general property that RL integrals do not commute.

3 Chain Rule

The chain rule of a RL operator is

$$\mathcal{D}^{q}_{a,x}\,f(g(x)) \;=\; \sum_{j=0}^{\infty}\binom{q}{j}\frac{x^{j-q}}{\Gamma(1+j-q)}\;\frac{d^{j}\,f(g(x))}{dx^{j}}. \tag{7.6.67}$$

The complexity of this result will inhibit its general utility in connection with computer algebra. The chain rule creates an infinite series that offers little hope of being expressible in closed form.

4 Leibniz's Rule

The rule for differentiation of a product of two functions is a familiar result in calculus. It states that

$$\frac{d^{n}(f(x)\,g(x))}{dx^{n}} \;=\; \sum_{j=0}^{n}\binom{n}{j}\frac{d^{n-j}\,f(x)}{dx^{n-j}}\;\frac{d^{j}\,g(x)}{dx^{j}} \tag{7.6.68}$$

for non-negative integers n. The generalization of Leibniz's rule to negative numbers is given by

$$\mathcal{D}^{q}_{a,x}(f(x)\,g(x)) \;=\; \sum_{j=0}^{\infty}\binom{q}{j}\mathcal{D}^{q-j}_{a,x}\,f(x)\,\mathcal{D}^{j}_{a,x}\,g(x), \tag{7.6.69}$$

where the binomial $\binom{q}{j} = \Gamma(q+1)/(\Gamma(j+1)\,\Gamma(q-j+1))$ is expressed by Euler's Γ function. Again we face the problem that Leinbiz's rule results into an infinite series. This series may collapse to a simple expression if the functions f and g are simple. However, in general computer algebra cannot handle this relation.

The discussed *Mathematica* code shows that it is sufficient for an implementation to use the definition given by the RL operator in Eq. (7.6.56)–(7.6.59). The mathematical formulas and the *Mathematica* code above show that the RL operator in mathematical and *Mathematica* notation is quite similar. To make this similarity to an identity, we introduce a special *Mathematica* notation identical with the RL operator symbol. The notation $\mathcal{D}^{\square}_{\square,\square}[\square]$ is connected with the function

RiemannLiouville[]. The template is designed in such a way that it is identical with the mathematical notation given above. However, this notation differs somewhat from the standard notation used in the literature. Since in *Mathematica* it is safer to handle the lower indices of the operator $\mathcal{D}_{a,x}^{-q}$ on the right side of the $\mathcal{D}$ symbol, we changed the notation given by Davis [7.15], who used ${}_a\mathcal{D}_x^{-q}$ for the RL operator. The function **RiemannLiouville[]** and the template $\mathcal{D}_{a,x}^{-q}$ allow us to carry out different calculations. The following examples show how the function **RiemannLiouville[]** is used and what kind of calculations are supported by this function.

We note that the following calculations are based on the package **FractionalCalculus`** developed by Südland and myself. This package is available from the author by request. To support the future development of the package **FractionalCalculus`**, we have to charge the user for the package.

7.6.2.2 Examples

An example frequently discussed in the literature [7.10, 7.11] is the differentiation of a constant. From standard calculus, we know that an ordinary integer differentiation of a constant vanishes. Applying the RL operator of order $q = 1/2$ to a numeric constant, say $c = 1$, we get

```
𝒟[1/2,0,x][1]
```

$$\frac{1}{\sqrt{\pi}\,\sqrt{x}}$$

This result compared with our knowledge of ordinary calculus is surprising. Contrary to an ordinary differentiation, the result of a fractional differentiation does not vanish but depends on the original variable, here x. The same result follows by applying the function **RiemannLiouville[]** to the constant. The difference is that we do not need to specify the lower boundary. The function **RiemannLiouville[]** assumes by default that the lower boundary is zero. However, we can change this boundary value by providing a third input variable in the second argument of

RiemannLiouville[]. Let us demonstrate this by first using **RiemannLiouville[]** with two arguments at the second input position

```
RiemannLiouville[1, {x, 1 / 2}]
```

Conditions to solve the fractional integral:
$x > 0$

$$\frac{1}{\sqrt{\pi}\,\sqrt{x}}$$

The result of both calculations is the same. However, we have the freedom to choose the lower boundary as a third entry in the function **RiemannLiouville[]**.

The gained results might contradict the general knowledge that the differentiation of a constant vanishes. Contrary to the ordinary calculus, in fractional calculus it is not true that the differentiation of a constant vanishes. This behavior is obvious if we recall the definition of a fractional derivative by an integration in Eq. (7.6.56). This nonvanishing of a RL operator applied to a constant is even true if we allow a general order of differentiation. Before we can apply the RL operator to the constant, we have to tell the package **FractionalCalculus** that we restrict the order of differentiation to positive values, meaning $\nu > 0$. This mathematical assumption is incorporated into the package **FractionalCalculus** by the function **Assume[]**. This function allows one to specify conditions under which the integrals are calculated. For our example, we set

```
Assume[ν > 0]
```

```
{{{ν > 0}, {Im[ν] → 0, Re[ν] → ν}}}
```

This assumption tells the RL operator that ν is a positive real number. The calculation of the RL integral in the general form then gives

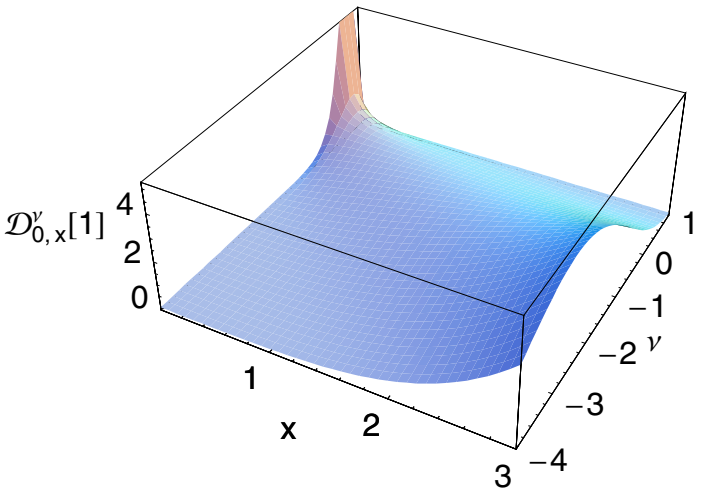
```
d1 = 𝒟[0, x, ν][K]
```

```
Conditions to solve the fractional integral:
x > 0 && Re[ν] < 1
```

$$\frac{K\, x^{-\nu}}{\Gamma[1-\nu]}$$

where K is a constant. The expression shows that for positive $\nu < 1$, the RL operator provides a nonvanishing result containing Euler's Γ function. A graphical representation of the result for different ν's is given in the following plot:

```
Plot3D[d1 /. K -> 1, {x, .01, 3},
  {ν, -4, 1}, PlotPoints -> 40, Mesh -> False,
  AxesLabel -> {"x", "ν", "𝒟[0, x, ν][1]"}];
```



The above calculations show some printings in between the input and output. These printouts inform you about the conditions under which the calculation was carried out. The output of conditional information is controlled by an option of **RiemannLiouville[]**. The options of the RL function are

```
Options[RiemannLiouville]
```

```
{ShowConditions → True, UniqueSymbols → False,
 OldhamSpanierConstants → False,
 FractionalIntegrationVariable → y,
 ShowFinalResult → False,
 ShowLiterature → False, ShowResults → False}
```

To suppress the information on solution conditions, we set the option ShowConditions to False.

```
SetOptions[RiemannLiouville, ShowConditions → False]
```

```
{ShowConditions → False, UniqueSymbols → False,
 OldhamSpanierConstants → False,
 FractionalIntegrationVariable → y,
 ShowFinalResult → False,
 ShowLiterature → False, ShowResults → False}
```

Now, **RiemannLiouville[]** does not display any information about the calculation. An example of a RL integration demonstrates this. The example uses a power function x^{μ} to which we apply the RL operator. Let us assume that the fractional order of integration is any positive number greater than zero and let μ be a real number. The application of the RL operator to this function gives

```
Assume[ν > 0];
```

```
D[x^μ, {x, -ν, 0}]
```

$$\frac{x^{\mu+\nu}\,\Gamma[1+\mu]}{\Gamma[1+\mu+\nu]}$$

The result is again a power function containing both parameters μ and ν as exponents. The behavior of projecting a function into the same class of function is not typical for the RL operator. The application to other classes of functions like exponentials, sines, and cosines demonstrates that we get higher transcendental functions. An example for this behavior is the function $e^{\alpha x}$ with $\alpha > 0$. The application of the RL integral delivers

```
Assume[α > 0];
```

```
D[e^(α x), {x, -ν, 0}]
```

$$\frac{e^{x\alpha}\,\alpha^{-\nu}\,\gamma[\nu, x\alpha]}{\Gamma[\nu]}$$

which represents the Mittag–Leffler function in *Mathematica* notation. The Mittag–Leffler function $E_x(\nu, \alpha)$ is defined by

$$E_x(\nu, \alpha) = \frac{e^{\alpha x}}{\alpha^{\nu}}\left(1 - \frac{\gamma(\nu,\alpha x)}{\Gamma(\nu)}\right). \tag{7.6.70}$$

Other examples showing the same behavior are the trigonometric functions

```
D[Sin[ω x], {x, -ν, 0}]
```

$$\frac{x^{1+\nu}\,\omega\,F_{p,q}\left[\begin{matrix}\{1\}\\ \{1+\frac{\nu}{2},\ \frac{3}{2}+\frac{\nu}{2}\}\end{matrix};\ -\frac{1}{4}x^2\omega^2\right]}{\Gamma[2+\nu]}$$

```
𝒟[0, x]^(-ν)[Cos[ω x]]
```

$$\frac{x^{\nu}\, F_{p,q}\left[\{1\};\ \{\frac{1}{2}+\frac{\nu}{2},\ 1+\frac{\nu}{2}\};\ -\frac{1}{4}x^2\omega^2\right]}{\Gamma[1+\nu]}$$

Both results are connected with hypergeometric functions $F_{p,q}$. Let us consider some slightly more complicated functions

```
f[x_] := (α + x)^λ
```

and assume that

```
Assume[λ > 0];
```

Then, the fractional integral of this function follows by

```
𝒟[0, x]^(-ν)[f[x]] // FunctionExpand
```

$$\frac{x^{\nu}\,\alpha^{\lambda}\, F_{2,1}\left[1,\ -\lambda,\ 1+\nu,\ -\frac{x}{\alpha}\right]}{\Gamma[1+\nu]}$$

If we change the sign of x in f, we get

```
ϕ[x_] := (α - x)^λ
```

```
𝒟[0, x]^(-ν)[ϕ[x]] // FunctionExpand
```

$$\frac{x^{\nu}\,\alpha^{\lambda}\, F_{2,1}\left[1,\ -\lambda,\ 1+\nu,\ \frac{x}{\alpha}\right]}{\Gamma[1+\nu]}$$

The difference between the two results is the minus sign in the argument of $F_{2,1}$.

As a another example, let us examine functions containing logarithms. The fractional integral of ln(*x*) is given by

```
D[-ν, 0, x][Log[x]]
```

$$\frac{x^{\nu}\,(-\text{HarmonicNumber}[\nu]+\text{Log}[x])}{\Gamma[1+\nu]}$$

A more general example is the combination of powers and logarithms by

```
D[-ν, 0, x][x^μ Log[x]]
```

$$\frac{1}{\Gamma[1+\mu+\nu]}\,(x^{\mu+\nu}\,\Gamma[1+\mu]\,(\text{HarmonicNumber}[\mu]-\text{HarmonicNumber}[\mu+\nu]+\text{Log}[x]))$$

If we combine a power and an exponential, we find a sum of hypergeometric functions:

```
D[-ν, 0, x][x^μ Exp[-1 / x]] // FunctionExpand
```

$$\frac{\pi\,x^{-1+\nu}\,\text{Csc}[\pi\mu]\,F_{1,1}[1-\nu,\,2+\mu,\,-\frac{1}{x}]}{\Gamma[2+\mu]\,\Gamma[\nu]}-\frac{\pi\,x^{\mu+\nu}\,\text{Csc}[\pi\mu]\,F_{1,1}[-\mu-\nu,\,-\mu,\,-\frac{1}{x}]}{\Gamma[-\mu]\,\Gamma[1+\mu+\nu]}$$

As a result, a combination of power laws and hypergeometric functions follows from

```
D[-ν, 0, x][Cos[α x]^2]
```

$$\frac{x^{\nu}\left(1+F_{p,q}\left[\begin{matrix}\{1\}\\ \{\frac{1}{2}+\frac{\nu}{2},\,1+\frac{\nu}{2}\}\end{matrix};\,-x^2\alpha^2\right]\right)}{2\,\Gamma[1+\nu]}$$

A completely different result follows for rational functions. First, let us set the integration order to the special value $-1/2$. For the function $(1-x)/(1+\alpha x)$, we find

```
D[-1/2, x][(1 - x)/(1 + α x)]
```

$$\frac{2\left(-\sqrt{x}\,\sqrt{\alpha}\,\sqrt{1+x\,\alpha}+(1+\alpha)\,\text{ArcSinh}\left[\sqrt{x}\,\sqrt{\alpha}\right]\right)}{\alpha^{3/2}\sqrt{\pi+\pi\,x\,\alpha}}$$

The result contains hyperbolic functions. For arbitrary ν, we find

```
hyper = D[-ν, x][(1 - x)/(1 + α x)] // FunctionExpand
```

```
Conditions to solve the fractional integral:
x > 0 && Re[ν] > 0 && (1/(x α) ≥ 0 || 1 + 1/(x α) ≤ 0 || Im[1/(x α)] ≠ 0)
```

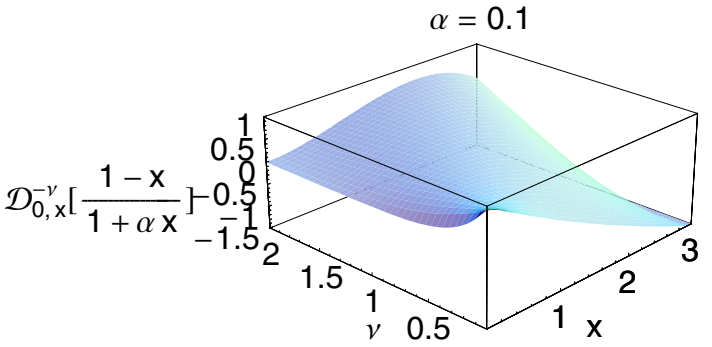
$$\frac{x^{\nu}\,F_{2,1}[1,\,1,\,1+\nu,\,-x\,\alpha]}{\Gamma[1+\nu]}-\frac{x^{1+\nu}\,F_{2,1}[1,\,2,\,2+\nu,\,-x\,\alpha]}{\Gamma[2+\nu]}$$

If we choose $\nu = 1/2$, the result reduces to the previous result:

```
hyper /. ν → 1/2 // Simplify
```

$$\frac{2\left(-\sqrt{x}\,\sqrt{\alpha}\,\sqrt{1+x\,\alpha}+(1+\alpha)\,\text{ArcSinh}\left[\sqrt{x}\,\sqrt{\alpha}\right]\right)}{\alpha^{3/2}\sqrt{\pi+\pi\,x\,\alpha}}$$

The following plot shows the result of the fractional derivative where x and ν are used as coordinates and α as a changing parameter. The static picture shows the transition to the value at $\nu = 1/2$. In addition, the variation of α visualizes the change of the surface. We observe that an increase in α will stretch out the surface to a more or less flat plane:



Including hyperbolic functions as arguments for the RL operator, we find

```
𝒟[0, x]^-ν [x^μ Sinh[γ x] e^(α x)] // FunctionExpand
```

$$-\frac{1}{2\,\Gamma[1+\mu+\nu]}\left(x^{\mu+\nu}\,\Gamma[1+\mu]\,\left(F_{1,1}[1+\mu,\,1+\mu+\nu,\,x\,(\alpha-\gamma)]-F_{1,1}[1+\mu,\,1+\mu+\nu,\,x\,(\alpha+\gamma)]\right)\right)$$

The pure Sinh with a square root of the independent variable as argument in the RL integral reduces to

```
𝒟[0, x]^-ν [Sinh[x^(1/2)]] // FunctionExpand
```

$$2^{-\frac{1}{2}+\nu}\,\sqrt{\pi}\;x^{\frac{1}{2}+\frac{1}{2}\,\left(-\frac{1}{2}-\nu\right)+\nu}\;I_{\frac{1}{2}\,(1+2\,\nu)}\left[\sqrt{x}\,\right]$$

The result is a Bessel function of I type multiplied by a power function. Even if we look at special functions like the Bessel functions, we can calculate the RL integral. The following example takes a Bessel J as argument in the RL integral:

```
D[0, x]^(-ν)[BesselJ[n, x]] // FunctionExpand
```

$$\frac{2^{-n}\, x^{n+\nu}\, F_{p,q}\left[\begin{matrix}\{\frac{1}{2}+\frac{n}{2},\ 1+\frac{n}{2}\}\\ \{1+n,\ \frac{1}{2}+\frac{n}{2}+\frac{\nu}{2},\ 1+\frac{n}{2}+\frac{\nu}{2}\}\end{matrix};\ -\frac{x^2}{4}\right]}{\Gamma[1+n+\nu]}$$

The result of this calculation is a hypergeometric function of general $F_{p,q}$ type multiplied by a power function. Combining a Bessel functions with a power, we get

```
D[0, x]^(-ν)[x^μ BesselJ[n, x]] // FunctionExpand
```

$$\left(2^{-n}\, x^{n+\mu+\nu}\, \Gamma[1+n+\mu]\ F_{p,q}\left[\begin{matrix}\{\frac{1}{2}+\frac{n}{2}+\frac{\mu}{2},\ 1+\frac{n}{2}+\frac{\mu}{2}\}\\ \{1+n,\ \frac{1}{2}+\frac{n}{2}+\frac{\mu}{2}+\frac{\nu}{2},\ 1+\frac{n}{2}+\frac{\mu}{2}+\frac{\nu}{2}\}\end{matrix};\ -\frac{x^2}{4}\right]\right) \Big/ \left(\Gamma[1+n]\ \Gamma[1+n+\mu+\nu]\right)$$

Again, we find a hypergeometric function $F_{q,p}$ multiplied by an extended power function. A semifractional derivative of $1/\sqrt{x}$ is given by

```
D[0, x]^(1/2)[1/Sqrt[x]]
```

```
0
```

Surprisingly, this differentiation vanishes. The reason why this result occurs is obvious from the more general derivative

```
D[0, x]^ν [1/Sqrt[x]]
```

$$\frac{\sqrt{\pi}\, x^{-\frac{1}{2}-\nu}}{\Gamma\left[\frac{1}{2}-\nu\right]}$$

We see that if $\nu = 1/2$, the Γ function approaches infinity and, thus, the overall behavior is reduced to zero.

The above examples serve to demonstrate that the function **RiemannLiouville[]** is designed in such a way that a large class of function is accessible via integration and differentiation. We already observed that the application of the RL operators deliver extraordinary results for simple functions. How these results are useful in connection with physical applications is discussed in Section 7.6.4.

7.6.3 Mellin Transforms

Frequently in mathematical physics we encounter pairs of functions related by an expression of the form

$$g(x) = \int_a^b f(k)\, K(x, k)\, dk. \tag{7.6.71}$$

The function $g(x)$ is called the integral transform of $f(k)$ by the kernel $K(x, k)$.

One of the most useful of the infinite number of possible transforms is the Fourier transform given by

$$g(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(k)\, e^{ixk}\, dk. \tag{7.6.72}$$

Two modifications of this transformation are the Fourier cosine and the Fourier sine transforms

$$g_c(x) = \sqrt{\frac{2}{\pi}} \int_0^{\infty} f(k) \cos(xk)\, dk \tag{7.6.73}$$

$$g_s(x) = \sqrt{\frac{2}{\pi}} \int_0^{\infty} f(k) \sin(xk)\, dk \tag{7.6.74}$$

The Fourier transform is based on the kernel $e^{i\,x\,k}$ and its real and imaginary parts taken separately, $\cos(k\,x)$ and $\sin(k\,x)$, because these kernels are the functions used to describe waves. Fourier transforms appear frequently in studies of waves. The output of a stellar interferometer, for instance, involves a Fourier transform of the brightness across a stellar disk. The electron distribution in an atom can be obtained from a Fourier transform of the amplitude of scattered X-rays.

Three other useful kernels in defining integral transforms are $e^{-k\,x}$, $k\,J_n(k\,x)$, and k^{x-1}. These give rise to the following transformations:

$$g(x) = \int_0^\infty f(k)\,e^{-k\,x}\,dk \tag{7.6.75}$$

defining the Laplace transform,

$$g(x) = \int_0^\infty f(k)\,k\,J_n(k\,x)\,dk, \tag{7.6.76}$$

known as the Hankel transform, and

$$g(x) = \int_0^\infty f(k)\,k^{x-1}\,dk, \tag{7.6.77}$$

the Mellin transform. Clearly, the possible types are unlimited. The following subsection will outline the Mellin transform in more detail.

7.6.3.1 Definition of the Mellin Transform

This subsection is concerned with the theory and application of the Mellin transform. We define the Mellin transform and its inverse. Several examples and the basic operational properties of the Mellin transform are discussed. Historically, Riemann in 1876 [7.16] first recognized the Mellin transform in his famous memoir on prime numbers. Its explicit formulation was given by Cahen in 1894 [7.17]. Almost simultaneously, Mellin, in two papers from 1896 and 1902 [7.18, 7.19], gave an elaborate discussion of the Mellin transform and its inversion formula.

In this subsection, we study the Mellin transform, which, although closely related to the Fourier transform, has its own peculiar uses. In particular, it turns out to be a most convenient tool for solving fractional integral equations. We recall first that the Fourier transform pair can be written in the form

$$F(\omega) = \int_{-\infty}^{\infty} e^{\mathrm{i}\omega t} f(t)\, dt, \qquad \text{with } \alpha < \mathrm{Im}(\omega) < \beta \tag{7.6.78}$$

and

$$f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{-\mathrm{i}\omega t} F(\omega)\, d\omega, \qquad \alpha < \gamma < \beta. \tag{7.6.79}$$

The Mellin transform and its inverse follow if we introduce the variable changes $p = i\omega$, $x = e^t$, and $\phi(t) = f(\ln(t))$, so that Eq. (7.6.78) and (7.6.79) become

$$\mathcal{M}(f(t)) = F(p) = \int_0^{\infty} t^{p-1} f(t)\, dt \tag{7.6.80}$$

and

$$\mathcal{M}^{-1}(F(p)) = f(t) = \frac{1}{2\pi i} \int_{c-i\infty}^{c+i\infty} t^{-p} F(p)\, dp, \tag{7.6.81}$$

respectively. Equation (7.6.80) is the Mellin transform and, (7.6.81) is the inversion formula for the Mellin transform. The transform normally exists only in the range $\alpha < \mathrm{Re}(p) < \beta$, and the inversion contour must lie in this strip.

The following theorem collects the main properties of the Mellin transform.

Theorem: Properties of Mellin Transform

If $\mathcal{M}(f(t)) = F(p)$, then the following properties hold:

No.	Property	
i)	Scaling	$\mathcal{M}(f(a\,t)) = \frac{1}{a^p}\,F(p),\quad a>0$
ii)	Shifting	$\mathcal{M}(t^a\,f(t)) = F(p+a)$
iii)	Derivatives	$\mathcal{M}(f^{(n)}(t)) = (-1)^n\,\frac{\Gamma(p)}{\Gamma(p-n)}\,F(p-n)$
iv)	Derivative multiplied with a power	$\mathcal{M}(t^n\,f^{(n)}(t)) = (-1)^n\,\frac{\Gamma(p+n)}{\Gamma(p)}\,F(p)$
v)	Differential operator	$\mathcal{M}\left(\left(t\,\frac{d}{dt}\right)^n f(t)\right) = (-1)^n\,p^n\,F(p)$
vI)	Integrals	$\mathcal{M}\left(\int_0^t f(u)\,du\right) = -\frac{F(p+1)}{p}$
vii)	nth repeated Integral	$\mathcal{M}(I_n\,f(t)) = \mathcal{M}\left(\int_0^t I_{n-1}\,f(u)\,du\right) = (-1)^n\,\frac{\Gamma(p)}{\Gamma(p+n)}\,F(p+n)$
viii)	Convolution type I	$\mathcal{M}(f(t)*g(t)) = \mathcal{M}(\int_0^\infty f(u)\,g(\frac{t}{u})\,\frac{1}{u}\,du) = F(p)\,G(p)$
ix)	Convolution type II	$\mathcal{M}(f(t)\circ g(t)) = \mathcal{M}(\int_0^\infty f(t\,u)\,g(u)\,du) = F(p)\,G(1-p)$

In this table, $I_n\,f(t)$ denotes the nth repeated integral if $f(t)$ defined by $I_n\,f(t) = \int_0^t I_{n-1}\,f(u)\,du$. ●

The package **FractionalCalculus** contains a function **MellinTransform[]**, which is accessible by the template $\mathcal{M}^{\square}_{\square}[\square]$, where the lower placeholder represents the original variable and the upper placeholder represents the Mellin variable. The placeholder in [] contains the function which is transformed. The following examples demonstrate the application of the Mellin transform to different functions.

7.6.3.2 Examples for Mellin Transforms

Before we discuss specific examples and applications of the Mellin transform, let us demonstrate some general properties. The scaling property of the Mellin transform for an arbitrary function f is given by

```
Remove[f, ϕ];
Assume[λ > 0];
```

```
MellinTransform[f[λ t], t, p]
```

$$\lambda^{-p}\, \mathcal{M}_t^p[f[t]]$$

The result is identical with property i) of the above table. The shifting property follows from

```
MellinTransform[t^λ f[t], t, p]
```

$$\mathcal{M}_t^{p+\lambda}[f[t]]$$

The following relations demonstrate that the Mellin transform is defined for powers:

```
MellinTransform[f[t^λ], t, p]
```

$$\frac{\mathcal{M}_t^{\frac{p}{\lambda}}[f[t]]}{\lambda}$$

for rational functions:

```
MtP[f[1/t]/t]
```

```
Mt^(1-p)[f[t]]
```

and for logarithms:

```
MtP[Log[t] f[t]]
```

```
MellinTransform^(0,0,1)[f[t], t, p]
```

Even general derivatives can be handled by the function **MellinTransform[]**:

```
MtP[∂t f[t]] // FunctionExpand
```

```
(1 - p) Mt^(-1+p)[f[t]]
```

```
MtP[∂t,t f[t]] // FunctionExpand
```

```
(-2 + p) (-1 + p) Mt^(-2+p)[f[t]]
```

The results are special cases of the general formula from above. The Mellin transform of an integral is given by

```
MtP[∫0^t f[τ] dτ] // FunctionExpand
```

$$-\frac{\mathcal{M}_t^{1+p}[f[t]]}{p}$$

The convolution properties viii) and ix) are

```
ℳ_t^p[∫_0^∞ f[τ] g[t/τ]/τ dτ]
```

```
ℳ_t^p[f[t]] ℳ_t^p[g[t]]
```

or

```
ℳ_t^p[∫_0^∞ f[t τ] g[τ] dτ]
```

```
ℳ_t^p[f[t]] ℳ_t^(1-p)[g[t]]
```

These general properties are important in the treatment of the following applications. Before we discuss the capabilities of the Mellin transform in connection with integrals, integral equations, and differential equations, we demonstrate the application of the Mellin transform to special functions.

The first example is concerned with the function $f(t) = e^{-n\,t}$ with $n > 0$. The Mellin transform of the exponential function follows by applying the operator $\mathcal{M}_{\square}^{\square}[\square]$ to the function

```
ℳ_t^p[Exp[-n t]]
```

```
n^-p Γ[p]
```

This result is characteristic for an exponential function. In the Mellin space, this kind of function is represented by the Γ function divided by n to the power of p denoting the factor in the exponent. The function **MellinTransform[]** also tells us that the real part of n and the real part of p must be greater than zero. Another example of interest is given by the rational function $1/(1+t)$. The Mellin transform of this function is

```
MellinTransform[p][1/(1 + t)]
```

```
π Csc[p π]
```

The Mellin transform of the generalized expression $f(t) = 1/(1+t)^n$ follows from

```
MellinTransform[p][1/(1 + t)^n]
```

```
Γ[n - p] Γ[p]
--------------
    Γ[n]
```

The result is represented by a fraction of Γ functions depending on the Mellin variable p and on the exponent n. The representation of the Mellin transform in terms of Γ functions is very useful in connection with the solution of fractional differential equations. Another interesting example containing an exponential function is the Mellin transform of the function $f(t) = 1/(e^t \pm 1)$. The two Mellin transforms read

```
MellinTransform[p][1/(Exp[t] - 1)]
```

```
Γ[p] Zeta[p]
```

The result contains a special function the so-called Riemann ζ function. The second representation of $f(t)$ with the − sign replaced by the + sign gives

```
MellinTransform[p][1/(Exp[t] + 1)]
```

```
2^-p (-2 + 2^p) Γ[p] Zeta[p]
```

Here again, the Γ function and the ζ function are involved in the representation of the Mellin transform. An example containing trigonometric functions is

```
MellinTransform[p, t][Sin[(1 - t)]]
```

$$\Gamma[p]\,\text{Sin}\left[1-\frac{p\,\pi}{2}\right]$$

The result contains trigonometric as well as the Γ function. The Mellin transform of the pure Cos[] is given by

```
MellinTransform[p, t][Cos[ω t]]
```

$$(\omega^2)^{-p/2}\,\text{Cos}\left[\frac{p\,\pi}{2}\right]\,\Gamma[p]$$

where ω is a positive constant. Other special functions are logarithms. An example containing a logarithmic expression is given by

```
MellinTransform[p, t][Log[1 + t]]
```

$$\frac{\pi\,\text{Csc}[(1+p)\,\pi]\,\Gamma[-p]}{\Gamma[1-p]}$$

These few examples demonstrate that the Mellin transform of special functions can be calculated in a direct way. We note that the package **FractionalCalculus** is capable to calculate all the Mellin transforms and more tabulated by Oberhettinger [7.20].

The inverse of the Mellin transform (**InverseMellinTransform[]**) in the package **FractionalCalculus** is accessible by the operator $(\mathcal{M}^{-1})^{\square}_{\square}[\square]$. The subscript denotes the Mellin variable and the superscript denotes the original variable. The template of the inverse Mellin transform is connected with the function **InverseMellinTransform[]**. A simple example for an inversion is

```
(M^-1)_p^t[Gamma[p]]
```

$$e^{-t}$$

which just delivers the exponential function. Another simple example is

```
(M^-1)_p^t[Gamma[p + n]]
```

$$e^{-t}\, t^{n}$$

where n is a positive number. More complicated results follow from

```
(M^-1)_p^t[Gamma[1 + p] Gamma[p/π] Gamma[1 - p/π]]
```

$$\mathcal{H}^{2,1}_{1,2}\left[t \,\middle|\, \begin{matrix} \{\{0, \frac{1}{\pi}\}\} & | & \{\} \\ \{\{0, \frac{1}{\pi}\}, \{1, 1\}\} & | & \{\} \end{matrix}\right]$$

where $\mathcal{H}^{2,1}_{1,2}$ represents a generalized hypergeometric function, so called Fox functions. A similar result follows from

```
(M^-1)_p^t[Gamma[1 + p]/Sin[p]]
```

$$\frac{\mathcal{H}^{2,1}_{1,2}\left[t \,\middle|\, \begin{matrix} \{\{0, \frac{1}{\pi}\}\} & | & \{\} \\ \{\{0, \frac{1}{\pi}\}, \{1, 1\}\} & | & \{\} \end{matrix}\right]}{\pi}$$

If we combine a Γ function and a trigonometric function in the Mellin space by a product, we find

```
(M^-1)_p^t[Gamma[p] Sin[p]] // FunctionExpand
```

$$\pi\,\mathcal{H}_{1,2}^{1,0}\left[t \,\middle|\, \begin{matrix} \{\} \mid & \{\{0,\ \frac{1}{\pi}\}\} \\ \{\{0,\ 1\}\} \mid & \{\{0,\ \frac{1}{\pi}\}\} \end{matrix}\right]$$

Another rational expression of Γ functions and the **Sin[]** gives

```
(M^-1)_p^t[Gamma[1 + p] Sin[p] / (Gamma[p] Gamma[2 p])]
```

$$\pi\,\mathcal{H}_{3,2}^{1,0}\left[t \,\middle|\, \begin{matrix} \{\} \mid & \{\{0,\ 1\},\ \{0,\ 2\},\ \{0,\ \frac{1}{\pi}\}\} \\ \{\{1,\ 1\}\} \mid & \{\{0,\ \frac{1}{\pi}\}\} \end{matrix}\right]$$

These small selection of special combinations of Γ functions demonstrate that the inverse Mellin transform is a powerful tool to represent special functions. The package **FractionalCalculus** allows one to calculate a large number of special functions, including Fox's *H* function, which is a generalization of the Meijer *G* function. The following applications demonstrate how a Mellin transform can be used to solve specific mathematical and physical problems.

7.6.3.3 Solution of Integrals

Let us first discuss the solution of specific integrals of the form

```
F[t] == Integrate[f[τ], {τ, 0, t}];
```

We first apply the Mellin transform on both sides of the equation:

```
r1 = M_t^p[F[t] == ∫_0^t f[τ] dτ]
```

$$\mathcal{M}_t^p[F[t]] == \frac{\Gamma[-p]\,\mathcal{M}_t^{1+p}[f[t]]}{\Gamma[1-p]}$$

The result represents the solution of the integral in Mellin space. The inversion of the Mellin transform provides us with the integral value:

```
(M^-1)_p^t[r1]
```

$$F[t] == (\mathcal{M}^{-1})_p^t\left[\frac{\Gamma[-p]\,\mathcal{M}_t^{1+p}[f[t]]}{\Gamma[1-p]}\right]$$

under the condition that $\partial_t F = f(t)$ and $F(0) = 0$. An integral satisfying these conditions is given by $\int_0^t \cos(\tau)\,\mathrm{d}\tau$. The Mellin transform according to the above formula gives for the integrand

```
intM = M_t^(p+1)[Cos[t]] Γ[-p] / Γ[1 - p] // FullSimplify
```

$$\Gamma[p]\,\mathrm{Sin}\left[\frac{p\pi}{2}\right]$$

Since the inversion of the Mellin transform is essentially based on Γ functions, we first have to represent the trigonometric function by Γ functions. The package **FractionalCalculus** contains general transformation rules to carry out this transformation. Applying the rules **TrigToGammaRules** to the result **intM**, we find

```
intM = intM /. TrigToGammaRules
```

$$\frac{\pi\,\Gamma[p]}{\Gamma[1-\frac{p}{2}]\,\Gamma[\frac{p}{2}]}$$

containing only Γ functions. The inverse Mellin transform now follows by

```
(M^-1)_p^t[intM] // FunctionExpand // PowerExpand
```

```
FoxH::changedstructure : Warning: 1/(2 Sqrt[π]) FoxH[
   {{}, {}} , {{{1/2, 1/2}}, {{0, 1/2}}} , t/2 , {1, 0, 0, 2} ]:
This Fox H-function has a changed structure in comparison
   with the input. Please check your input data.
```

```
Sin[t]
```

The inverse Mellin transform is based on the definition of Fox's *H* function. This connection is always used by **FractionalCalculus** to reduce the result to a special function. The direct integration using *Mathematica* provides the same result:

```
∫_0^t Cos[τ] dτ
```

```
Sin[t]
```

Another integral also satisfying the above conditions is given by

```
∫_0^t τ^α / (1 - (b τ)^β) dτ // Timing
```

```
{6.81 Second, If[t > 0 && Re[α] > -1 && Re[β] > 0,
  t^(1+α) F_{2,1}[1, (1+α)/β, (1+α+β)/β, b^β t^β] / (1 + α), ∫_0^t τ^α / (1 - (b τ)^β) dτ]}
```

Let us assume that the parameters α, β, and b are positive constants:

```
Assume[α > 0]; Assume[β > 0]; Assume[b > 0];
```

The Mellin transform of the integrand extended by the two Γ functions then follows as

```
vh =
 M_t^(p+1)[ t^α / (1 - (b t)^β) ] Γ[-p] / Γ[1 - p] /. TrigToGammaRules //
   Simplify // Timing
```

$$\left\{1.1\,\text{Second},\ \frac{b^{-1-p-\alpha}\,\pi\,\Gamma[-p]\,\Gamma\left[\frac{1+p+\alpha}{\beta}\right]\,\Gamma\left[-\frac{1+p+\alpha-\beta}{\beta}\right]}{\beta\,\Gamma[1-p]\,\Gamma\left[\frac{-2-2p-2\alpha+\beta}{2\beta}\right]\,\Gamma\left[\frac{2+2p+2\alpha+\beta}{2\beta}\right]}\right\}$$

We represent the result by Γ functions because the inversion of the Mellin transform relies on this functions. The inversion of the Mellin transform gives

```
(M^-1)_p^t[vh[[2]]] // Timing
```

$$\left\{1.37\,\text{Second},\ \frac{1}{\beta}\left(b^{-1-\alpha}\,\pi\,\mathcal{H}^{1,2}_{3,3}\left[\,b\,t\,\middle|\,\begin{matrix}\{\{1,1\},\{\frac{1+\alpha}{\beta},\frac{1}{\beta}\}\} & \{\{\frac{2+2\alpha+\beta}{2\beta},\frac{1}{\beta}\}\} \\ \{\{\frac{1+\alpha}{\beta},\frac{1}{\beta}\}\} & \{\{0,1\},\{\frac{2+2\alpha+\beta}{2\beta},\frac{1}{\beta}\}\}\end{matrix}\right]\right)\right\}$$

representing the result in terms of a Fox function. The direct integration with integrate has a different representation by hypergeometric functions

```
∫_0^t τ^α / (1 - (b τ)^β) dτ
```

$$\text{If}\left[t>0\ \&\&\ \text{Re}[\alpha]>-1\ \&\&\ \text{Re}[\beta]>0,\ \frac{t^{1+\alpha}\,F_{2,1}\left[1,\frac{1+\alpha}{\beta},\frac{1+\alpha+\beta}{\beta},b^{\beta}\,t^{\beta}\right]}{1+\alpha},\ \int_0^t\frac{\tau^{\alpha}}{1-(b\,\tau)^{\beta}}\,d\tau\right]$$

Another application of the Mellin transform is the calculation of the moments of the Kohlrausch–William–Watts (KWW) distribution given by

```
KWW[x_] := e^-(b x)^β
```

The moments of this distribution are given by

```
∫_0^t τ^α KWW[τ] dτ
```

```
If[t > 0 && Re[β] > 0 && β (1 + Arg[b]) < π/2,
 (b^β)^(-(1+α)/β) γ[(1+α)/β, b^β t^β] / β, ∫_0^t e^(-(b τ)^β) τ^α dτ]
```

where α, β, and b are positive constants.

```
Assume[α > 0]; Assume[β > 0]; Assume[b > 0];
```

The Mellin transform of the integrand follows by

```
intM = M_τ^(p+1)[τ^α KWW[τ]] Γ[-p] / Γ[1 - p]
```

```
b^(-1-p-α) Γ[-p] Γ[(1+p+α)/β] / (β Γ[1 - p])
```

The inversion shows the coincidence with the direct method:

```
res = (M^(-1))_p^t[intM] // PowerExpand // Simplify
```

```
b^(-1-α) γ[(1+α)/β, b^β t^β] / β
```

7.6.3.4 Integral Equations

Another application of the Mellin transform is the solution of integral equations of the convolution type. Let us consider the general form of a first-kind Fredholm convolution integral equation. The unknown function in this equation is denoted by f:

```
firstFredholm = ∫_0^∞ f[τ] k[t τ] dτ == g[t]
```

$$\int_0^\infty f[\tau]\, k[t\, \tau]\, d\tau == g[t]$$

k and g are the kernel and the inhomogenity of the equation. If g equals zero, we have a homogenous integral equation of Fredholm type. The Mellin transform applied to this equation gives

```
MellinFirstFredholm = M_t^p[firstFredholm]
```

$$\mathcal{M}_t^{1-p}[f[t]]\, \mathcal{M}_t^{p}[k[t]] == \mathcal{M}_t^{p}[g[t]]$$

If we replace p by $1 - p$ in the above expression, we find a standardized representation of the Fredholm equation in Mellin space:

```
standard = MellinFirstFredholm /. p → -p + 1
```

$$\mathcal{M}_t^{p}[f[t]]\, \mathcal{M}_t^{1-p}[k[t]] == \mathcal{M}_t^{1-p}[g[t]]$$

which is solved by

```
solMellin = Solve[standard, M_t^p[f[t]]]
```

$$\left\{\left\{\mathcal{M}_t^{p}[f[t]] \to \frac{\mathcal{M}_t^{1-p}[g[t]]}{\mathcal{M}_t^{1-p}[k[t]]}\right\}\right\}$$

The inversion of the Mellin transform gives us the formal solution of the integral equation:

```
(ℳ^-1)^t_p[solMellin]
```

$$\left\{\left\{f[t] \to (\mathcal{M}^{-1})^{t}_{p}\left[\frac{\mathcal{M}_t^{1-p}[g[t]]}{\mathcal{M}_t^{1-p}[k[t]]}\right]\right\}\right\}$$

The second type of convolution-type integral connected with a Mellin transform is given by the equation

```
secondFredholm = g[t] == ∫_0^∞ f[τ] k[t/τ]/τ dτ
```

$$g[t] == \int_0^\infty \frac{f[\tau]\, k[\frac{t}{\tau}]}{\tau}\, d\tau$$

Again, k and g are the kernel and the inhomogenity, respectively. For this second kind of convolution equation, a Mellin transform provides

```
MellinSecondFredholm = ℳ^p_t[secondFredholm]
```

$$\mathcal{M}_t^p[g[t]] == \mathcal{M}_t^p[f[t]]\, \mathcal{M}_t^p[k[t]]$$

We realize that for the second kind of equation, we do not need to shift the Mellin variable p in any way. Thus, we can proceed to solve the resulting relation to derive the solution in Mellin space:

```
solMellin = Solve[MellinSecondFredholm, ℳ^p_t[f[t]]]
```

$$\left\{\left\{\mathcal{M}_t^p[f[t]] \to \frac{\mathcal{M}_t^p[g[t]]}{\mathcal{M}_t^p[k[t]]}\right\}\right\}$$

The inversion of the relation gives the formal solution in original variables:

```
(ℳ^-1)_p^t[solMellin]
```

$$\left\{\left\{f[t] \to (\mathcal{M}^{-1})_p^t\left[\frac{\mathcal{M}_t^p[g[t]]}{\mathcal{M}_t^p[k[t]]}\right]\right\}\right\}$$

Thus, the algorithmic procedure to solve the two integral equations of convolution type must distinguish two cases of specific kernels. The characteristic is even more pronounced in the Mellin space, where the two cases differ by shifts in the Mellin variable from each other. A function which solves first Fredholm equations of convolution type has to be sensitive on this case. The following function realizes an automatic solution procedure for the two types of integral equation:

```
Clear[ISolveFirstFredholm]
ISolveFirstFredholm[equation_, depend_, independ_] :=
 Block[{mtr, solmtr, p, k, vh, solexp},
  mtr = MellinTransform[equation, independ, p];
        vh = k == First[Cases[
      Level[mtr, Depth[mtr]], a_. MellinTransform[
         Apply[depend, {independ}], t_, p_] :> p]];
        solexp = Solve[vh, k] /. k → p // Flatten;
    mtr = mtr /. solexp;
        solmtr = Solve[mtr, MellinTransform[
     Apply[depend, {independ}], independ, p]];
        InverseMellinTransform[solmtr, p, independ]]
```

The above lines carry out first the Mellin transform of the integral equation. In a second step, the Mellin variable for the unknown function is determined. If a shift in the Mellin variable occurs, this shift is eliminated by an appropriate transformation. Next, the solution in Mellin space is calculated. The last step transforms the solution in Mellin space to the original variables. The general integral equations are solved automatically by

```
ISolveFirstFredholm[firstFredholm, f, t]
```

$$\left\{\left\{f[t] \to (\mathcal{M}^{-1})_p^t\left[\frac{\mathcal{M}_t^{1-p}[g[t]]}{\mathcal{M}_t^{1-p}[k[t]]}\right]\right\}\right\}$$

and the solution of the second integral equation follows from

```
ISolveFirstFredholm[secondFredholm, f, t]
```

$$\left\{\left\{f[t] \to (\mathcal{M}^{-1})_p^t\left[\frac{\mathcal{M}_t^{p}[g[t]]}{\mathcal{M}_t^{p}[k[t]]}\right]\right\}\right\}$$

Thus, we have a general procedure to solve first-kind Fredholm integral equations of the convolution type. A special example of the first convolution type is given by

```
equation1 = ∫_0^∞ Sin[t τ] f[τ] dτ == 1/(1 + t)^n
```

$$\int_0^\infty f[\tau]\,\text{Sin}[t\,\tau]\,d\tau == (1+t)^{-n}$$

where the kernel is given by a trigonometric function. The solution of this integral equation then follows by

```
solution1 =
 ISolveFirstFredholm[equation1, f, t] // FunctionExpand
```

FoxH::changedstructure : Warning: $\frac{1}{\sqrt{\pi}}$ FoxH[$\{\{\{\frac{1}{2}, \frac{1}{2}\}\}, \{\}\}$, $\{\{\{\frac{1}{2}, \frac{1}{2}\}, \{-1+n, 1\}\}, \{\}\}$, t , {2, 1, 1, 2}]:
This Fox H-function has a changed structure in comparison with the input. Please check your input data.

$$\left\{\left\{f[t] \to \frac{1}{\Gamma[n]}\left((-i\,t)^{-\frac{1}{2}+\frac{n}{2}}\,(i\,t)^{-\frac{1}{2}+\frac{n}{2}}\,\text{Cos}[t]\,\text{Csc}\left[\left(\frac{1}{2}+\frac{1-n}{2}\right)\pi\right]\right.\right.$$
$$\text{Csc}\left[\left(\frac{1-n}{2}+\frac{n}{2}\right)\pi\right]\Big) + \frac{1}{(-2+n)\,(-1+n)\,\pi}\left(\sqrt{-i\,t}\right.$$
$$\left.\sqrt{i\,t}\,\text{Csc}\left[\left(-\frac{1}{2}+\frac{1}{2}\,(-1+n)\right)\pi\right]\text{Csc}\left[\left(-\frac{1}{2}+\frac{n}{2}\right)\pi\right]\right.$$
$$\left.F_{p,q}\left[\begin{matrix}\{1\}\\ \{\frac{3}{2}-\frac{n}{2},\,2-\frac{n}{2}\}\end{matrix};\,-\frac{t^2}{4}\right]\text{Sin}[n\,\pi]\right) +$$
$$\frac{1}{t^3\,\Gamma[n]}\left((-i\,t)^{1+\frac{n}{2}}\,(i\,t)^{1+\frac{n}{2}}\,\text{Csc}\left[\left(\frac{1}{2}-\frac{n}{2}\right)\pi\right]\right.$$
$$\left.\left.\text{Csc}\left[\left(\frac{1}{2}\,(-1+n)-\frac{n}{2}\right)\pi\right]\text{Sin}[t]\right)\right\}\right\}$$

An example for the second convolution type integral equation is given by the equation:

```
equation2 = 0 == Exp[-t] - Integral_0^Infinity φ[τ] Cos[t τ]/τ dτ
```

$$0 == e^{-t} - \int_0^\infty \frac{\text{Cos}[t\,\tau]\,\phi[\tau]}{\tau}\,d\tau$$

Again, we replaced the kernel by a trigonometric function. The solution of this equation follows from

```
solution2 = ISolveFirstFredholm[equation2, ϕ, t]
```

```
FoxH::changedstructure :
 Warning: 1/(2 √π) FoxH[ {{{1/2, 1/2}}, {}} ,
   {{{1/2, 1/2}}, {}} , t , {1, 1, 1, 1} ]:
This Fox H-function has a changed structure in comparison
   with the input. Please check your input data.
```

$$\left\{\left\{\phi[t] \to \frac{2\,t}{\pi\,(1+t^2)}\right\}\right\}$$

Another example is concerned with the Laplace integral equation

$$\text{equation3} = \int_0^\infty e^{-t\,\tau}\,f[\tau]\,d\tau == \frac{1}{(1+t)^n}$$

$$\int_0^\infty e^{-t\,\tau}\,f[\tau]\,d\tau == (1+t)^{-n}$$

which has the solution

```
solution3 = ISolveFirstFredholm[equation3, f, t]
```

$$\left\{\left\{f[t] \to \frac{e^{-t}\,t^{-1+n}}{\Gamma[n]}\right\}\right\}$$

Replacing in *equation3* the exponential constant E by an arbitrary number a, we get the equation

$$\text{equation4} = \int_0^\infty a^{-t\,\tau}\,f[\tau]\,d\tau == \frac{1}{(1+t)^n}$$

$$\int_0^\infty a^{-t\,\tau}\,f[\tau]\,d\tau == (1+t)^{-n}$$

The solution of this integral equation is

```
solution4 = ISolveFirstFredholm[equation4, f, t]
```

$$\left\{\left\{f[t] \to \frac{a^{-t}\,\text{Log}[a]\,(t\,\text{Log}[a])^{-1+n}}{\Gamma[n]}\right\}\right\}$$

The second kind of Fredholm equations allows the occurrence of the unknown function outside of the integral. One of the two standard forms of the convolution-type Fredholm integral equations of the second kind is given by

```
secondFredholm1 = f[t] == g[t] + Integrate[k[t τ] f[τ], {τ, 0, ∞}]
```

$$f[t] == g[t] + \int_0^\infty f[\tau]\, k[t\,\tau]\, d\tau$$

This equation also can be solved by means of a Mellin transform.

```
secondF1Mellin1 = MellinTransform[p][secondFredholm1]
```

$$\mathcal{M}_t^p[f[t]] == \mathcal{M}_t^p[g[t]] + \mathcal{M}_t^{1-p}[f[t]]\,\mathcal{M}_t^p[k[t]]$$

The application of the Mellin operator to the equation shows that the Mellin transform of the unknown function f occurs with two different Mellin variables (i.e., with p and $1 - p$). This is also true if we carry out the Mellin transform on the original equation a second time with the second Mellin variable chosen as $1 - p$:

```
secondF1Mellin2 = MellinTransform[1-p][secondFredholm1]
```

$$\mathcal{M}_t^{1-p}[f[t]] == \mathcal{M}_t^{1-p}[g[t]] + \mathcal{M}_t^p[f[t]]\,\mathcal{M}_t^{1-p}[k[t]]$$

Both transforms are equivalent and are the basis for the solution in Mellin space following from

```
solutionMellin =
 Solve[{secondF1Mellin1, secondF1Mellin2},
  {M_t^p[f[t]], M_t^(1-p)[f[t]]}]
```

$$\left\{\left\{\mathcal{M}_t^p[f[t]] \to -\frac{\mathcal{M}_t^p[g[t]] + \mathcal{M}_t^{1-p}[g[t]]\,\mathcal{M}_t^p[k[t]]}{-1 + \mathcal{M}_t^{1-p}[k[t]]\,\mathcal{M}_t^p[k[t]]},\right.\right.$$
$$\left.\left.\mathcal{M}_t^{1-p}[f[t]] \to -\frac{\mathcal{M}_t^{1-p}[g[t]] + \mathcal{M}_t^p[g[t]]\,\mathcal{M}_t^{1-p}[k[t]]}{-1 + \mathcal{M}_t^{1-p}[k[t]]\,\mathcal{M}_t^p[k[t]]}\right\}\right\}$$

The first formal solution follows from

```
(M^-1)_p^t[solutionMellin[[1, 1]]]
```

$$f[t] \to -(\mathcal{M}^{-1})_p^t\left[\frac{\mathcal{M}_t^p[g[t]] + \mathcal{M}_t^{1-p}[g[t]]\,\mathcal{M}_t^p[k[t]]}{-1 + \mathcal{M}_t^{1-p}[k[t]]\,\mathcal{M}_t^p[k[t]]}\right]$$

and the second one from the inversion

```
(M^-1)_(1-p)^t[solutionMellin[[1, 2]]]
```

$$f[t] \to -(\mathcal{M}^{-1})_{1-p}^t\left[\frac{\mathcal{M}_t^{1-p}[g[t]] + \mathcal{M}_t^p[g[t]]\,\mathcal{M}_t^{1-p}[k[t]]}{-1 + \mathcal{M}_t^{1-p}[k[t]]\,\mathcal{M}_t^p[k[t]]}\right]$$

The second solution is equivalent with the first solution. This is shown by replacing p by $1 - p$ in the second Mellin solution. Applying to the result the standard Mellin transform, we find

```
(M^-1)_p^t[solutionMellin[[1, 2]] /. p → 1 - p]
```

$$f[t] \to -(\mathcal{M}^{-1})_p^t\left[\frac{\mathcal{M}_t^p[g[t]] + \mathcal{M}_t^{1-p}[g[t]]\,\mathcal{M}_t^p[k[t]]}{-1 + \mathcal{M}_t^{1-p}[k[t]]\,\mathcal{M}_t^p[k[t]]}\right]$$

which is identical with the first solution. Thus, in an automatic solution procedure, we only need to treat one of the solutions in Mellin space. The second type of a second-kind Fredholm equation is given by

```
secondFredholm2 = f[t] == g[t] + ∫_0^∞ k[t/τ] f[τ]/τ dτ
```

$$f[t] == g[t] + \int_0^\infty \frac{f[\tau]\, k[\frac{t}{\tau}]}{\tau}\, d\tau$$

The Mellin transform of this equation shows that the Mellin representation of the unknown function occurs at both places with the same Mellin variable *p*:

```
M_t^p[secondFredholm2]
```

$$\mathcal{M}_t^p[f[t]] == \mathcal{M}_t^p[g[t]] + \mathcal{M}_t^p[f[t]]\, \mathcal{M}_t^p[k[t]]$$

This indicates that the solution procedure of the first-kind Fredholm equations can be applied to this type of convolution equation. The formal solution follows from

```
ISolveFirstFredholm[secondFredholm2, f, t]
```

$$\{\{f[t] \to (\mathcal{M}^{-1})_p^t\left[\frac{\mathcal{M}_t^p[g[t]]}{1 - \mathcal{M}_t^p[k[t]]}\right]\}\}$$

Thus, the second type of Fredholm equations can be automatically solved by the following function:

```
Clear[ISolveSecondFredholm]
ISolveSecondFredholm[equation_, depend_,
  independ_] := Block[{mtr, solmtr, p, k, vh, solexp},
  mtr = MellinTransform[equation, independ, p];
        vh = Complement[
    Union[Cases[Level[mtr, Depth[mtr]],
      a_. MellinTransform[Apply[depend, {independ}],
         t_, p_] :> p]], {p}];
        If[Length[vh] >= 1,
        mtr1 = Map[
     MellinTransform[equation, independ, #] &, vh];
        solmtr = Solve[Flatten[{mtr, mtr1}],
      {MellinTransform[Apply[depend, {independ}],
        independ, p], MellinTransform[Apply[depend,
         {independ}], independ, 1 - p]}] // Flatten;
            solmtr = Cases[solmtr,
     Rule[MellinTransform[eq_, t_, p], y___] :>
      Rule[MellinTransform[eq, t, p], y]],
            solmtr = Solve[mtr,
       MellinTransform[Apply[depend, {independ}],
        independ, p]] // Flatten;
                    ];
        InverseMellinTransform[solmtr, p, independ]]
```

The formal solution of the second Fredholm equation then follows by

```
ISolveSecondFredholm[secondFredholm1, f, t]
```

$$\left\{f[t] \to -(\mathcal{M}^{-1})^{t}_{p}\left[\frac{\mathcal{M}_t^{p}[g[t]] + \mathcal{M}_t^{1-p}[g[t]]\,\mathcal{M}_t^{p}[k[t]]}{-1 + \mathcal{M}_t^{1-p}[k[t]]\,\mathcal{M}_t^{p}[k[t]]}\right]\right\}$$

This solution is actually a formal solution because the inverse Mellin transform fails to reduce to Fox functions. The main obstacle to prevent the inversion is the -1 in the denominator preventing a pure representation by Γ functions. At this point, we reach the limit of the solution class based on Fox functions. A specific example demonstrates this behavior more clearly. Let us examine the Fredholm equation of the second kind:

```
sF1 = f[t] == (1 + t)^-α + ∫_0^∞ Sin[t τ] f[τ] dτ
```

$$f[t] == (1+t)^{-\alpha} + \int_0^\infty f[\tau]\, \text{Sin}[t\,\tau]\, d\tau$$

The solution should follow by

```
ISolveSecondFredholm[sF1, f, t] // FunctionExpand
```

$$\left\{f[t] \to -(\mathcal{M}^{-1})_p^t\left[\frac{\frac{2\,\Gamma[1-\frac{p}{2}]\,\Gamma[p]\,\Gamma[-p+\alpha]}{\Gamma[\alpha]} + \frac{2^p\sqrt{\pi}\,\Gamma[1-p]\,\Gamma[\frac{1}{2}+\frac{p}{2}]\,\Gamma[-1+p+\alpha]}{\Gamma[\alpha]}}{-2\,\Gamma[1-\frac{p}{2}] + \pi\,\Gamma[1-\frac{p}{2}]}\right]\right\}$$

The result demonstrates that the inverse Mellin transform is, in principle, possible if we extend the meaning of the Barns integral. However, the resulting function lies outside the class of Fox functions. The second type of second-kind Fredholm integral equation of convolution type has the formal solution

```
ISolveSecondFredholm[secondFredholm2, f, t]
```

$$\left\{f[t] \to (\mathcal{M}^{-1})_p^t\left[\frac{\mathcal{M}_t^p[g[t]]}{1-\mathcal{M}_t^p[k[t]]}\right]\right\}$$

Again, the problem is the same as in the first convolution type. A specific example shows the problem more clearly:

```
sF2 = f[t] == 1/(1 + t)^n + ∫_0^∞ Sin[t/τ] f[τ]/τ dτ
```

$$f[t] == (1+t)^{-n} + \int_0^\infty \frac{f[\tau]\,\text{Sin}[\frac{t}{\tau}]}{\tau}\, d\tau$$

```
ISolveSecondFredholm[sF2, f, t]
```

$$\left\{f[t] \to \frac{(\mathcal{M}^{-1})_{p}^{t}\left[\frac{\Gamma[n-p]\,\Gamma[p]}{2^{-1+p}\sqrt{\pi}\,\Gamma[\frac{1}{2}+\frac{p}{2}]}\right]}{1-\frac{\Gamma[n-p]\,\Gamma[p]}{\Gamma[1-\frac{p}{2}]}}\right\}$$

The occurrence of the -1 in the denominator again prevents a solution by Fox functions.

At this point, we reach the limits of special functions which serve to solve the second kind of Fredholm equations. If we are able to enlarge the definitions of special functions, we will have access to the explicit solution of the equation. However, so far we did not extend the package **FractionalCalculus** to this kind of special functions.

7.6.4 Fractional Differential Equations

The current chapter deals with the formulation and solution of fractional differential equations (FDEs). We introduce the solution procedure by recalling the techniques for linear ordinary differential equations (ODEs). The generalization of these techniques allows us to treat FDEs in different physical and chemical applications. We discuss relaxation phenomenons in complex systems like polymers and anomalous diffusion processes.

7.6.4.1 Linear Ordinary Differential Equations

Linear ordinary differential equations (ODEs) occur frequently in mathematical and physical applications. In general, a differential equation is an equation that relates an unknown function u and one or more derivatives or differentials of that unknown function with respect to one or more independent variables. An ODE contains one or more unknown functions but depends only on one independent variable. A linear ODE is an equation containing the dependent variable and its derivatives linearly. Examples of ODEs are

$$\frac{d\,u(t)}{d\,t} = f(t, u(t)), \tag{7.6.82}$$

representing a general first-order ODE for the unknown u. The order of a differential equation is the order of the highest derivative that appears in the equation. A linear first-order ODE consists of an equation containing u linearly. The most general linear first-order ODE is given by

$$\frac{d\,u(t)}{d\,t} = \alpha(t)\,u(t) + \beta(t), \tag{7.6.83}$$

where α and β are real functions of t. This equation is connected with the Langevin equation if $\alpha(t) = -\gamma$ and $\beta(t)$ represents a random force.

A general second-order equation is given by the relation

$$\frac{d^2\,u(t)}{d\,t^2} = g\Big(t,\, u(t),\, \frac{d\,u(t)}{d\,t}\Big). \tag{7.6.84}$$

The most general linear second-order ODE is

$$\frac{d^2\,u(t)}{d\,t^2} = \alpha(t)\,\frac{d\,u(t)}{d\,t} + \beta(t)\,u(t) + \kappa(t)$$

with α, β, and κ arbitrary functions of t. The next step in increasing the order is a general nth-order ODE like

$$\frac{d^{\,n}\,u(t)}{d\,t^n} = h\Big(t,\, u(t),\, \frac{d\,u(t)}{d\,t},\, \ldots,\, \frac{d^{n-1}\,u(t)}{d\,t^{n-1}}\Big). \tag{7.6.85}$$

So far, we introduced the basic notations to classify ODEs. The question now is, how can we solve these equations? Before we start to solve the equations, let us recall the meaning of a solution of ODEs. To say that $u = v(t)$ is a solution of the differential equation (7.6.85) on an interval K means that

$$\frac{d^{\,n}\,v(t)}{d\,t^n} = h\Big(t,\, v(t),\, \frac{d\,v(t)}{d\,t},\, \ldots,\, \frac{d^{n-1}\,v(t)}{d\,t^{n-1}}\Big)$$

is satisfied for every choice of t in the interval K. In other words, a solution, when substituted into the ODE, makes the equation identically true for t in K. How these solutions, especially for linear ODEs, can be derived is the subject of the next section. The solution of general nonlinear ODEs and PDEs is discussed in the book by Baumann [7.21].

7.6.4.2 Solution of ODEs by Integral Transforms

In this subsection, we repeat the solution procedure of linear ODEs by means of integral transforms. Integral transforms are one of the efficient methods to solve initial value problems. In detail, we discuss the Laplace transform technique to solve ODEs. We study this kind of technique because it is also instrumental in solving fractional differential equations.

One of the key properties of a Laplace transform is the reduction of a differential equation to an algebraic equation. This property is based on the transformation of differentials. The result is that an *n*th-order derivative $f^{(n)}(t)$ transforms like

$$\mathfrak{L}(f^{(n)}(t)) = s^n F(s) - \sum_{m=1}^{n} s^{n-m} f^{(m-1)}(0). \tag{7.6.86}$$

The right-hand side of Eq. (7.6.86) consists of a term containing the Laplace transform of f, displayed as $F(s)$, multiplied by the nth power of the Laplace variable s. The other terms contain the initial conditions represented by derivatives of f up to order $n - 1$. We observe that a single derivative transforms to a polynomial in the Laplace variable s. This behavior simplifies an ODE to a pure algebraic relation.

The following example demonstrate this for a first-order ODE. The equation under discussion is the relaxation equation

$$\frac{d f(t)}{d t} = -\frac{1}{\tau} f(t) \tag{7.6.87}$$

with τ, the relaxation time, a constant. Here, we denote the dependent variable by f to separate the mathematical representation of the equation from the physical meaning. This equation is represented in *Mathematica* by

```
Remove[f];
```

```
relaxation = ∂t f[t] == -1/τ f[t]
```

```
f'[t] == -f[t]/τ
```

The Laplace transform of the above equation follows with

```
lrelax = ℒ_t^s[relaxation]
```

$$-\mathtt{f[0]} + \mathtt{s}\,\mathcal{L}_{\mathtt{t}}^{\mathtt{s}}[\mathtt{f[t]}] == -\frac{\mathcal{L}_{\mathtt{t}}^{\mathtt{s}}[\mathtt{f[t]}]}{\tau}$$

representing an algebraic equation in Laplace space. The Laplace transform of f is denoted by $\mathcal{L}_t^s[f[t]]$. The solution of this equation in Laplace space follows by solving it with respect to the Laplace transform:

```
lsol = Solve[lrelax, ℒ_t^s[f[t]]] // Flatten
```

$$\left\{\mathcal{L}_{\mathtt{t}}^{\mathtt{s}}[\mathtt{f[t]}] \to \frac{\mathtt{f[0]}}{\mathtt{s} + \frac{1}{\tau}}\right\}$$

The result shows that the solution in Laplace space is represented by a function depending on the Laplace variable s and the initial condition $f(t = 0)$. The solution in the original variables results by inverting the Laplace transform:

```
sol = (ℒ^-1)_s^t[ℒ_t^s[f[t]] /. lsol]
```

$$e^{-\frac{t}{\tau}}\, f(0)$$

The solution of the relaxation equation is thus given by an exponential multiplied by the initial condition $f(0)$.

This simple example contains the necessary steps to derive a solution for an initial value problem. We realize that the method presented is completely algorithmic and can be incorporated into a *Mathematica* function. The steps solving a linear initial value problem for an ODE in f can be summarized as follows:

1. Laplace transform the ODE.

2. Solve the resulting algebraic equation to find the solution in the Laplace variable.

3. Invert the Laplace transform to find the solution in original coordinates.

These three steps are graphically shown in Figure 7.6.25.

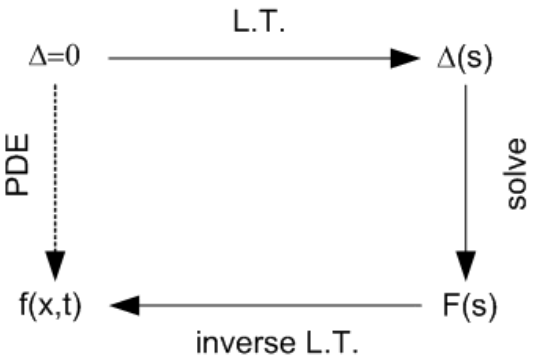


Figure 7.6.25. Solution procedure based on the Laplace transform for linear ODEs.

We start from a linear ODE $\Delta = 0$ of arbitrary order. Laplace transform this equation and solve for the Laplace variable F. The inversion of the Laplace solution F delivers the solution of the ODE. These steps are always feasible if the coefficients of the derivatives and the functions are constants. If we encounter analytic coefficients, we end up with an ODE in Laplace space.

So far, we demonstrated the solution technique for a homogeneous ODE. If the equation contains a nonvanishing inhomogeneity, the procedure works as well. We demonstrate this by extending the relaxation equation with an inhomogeneity representing an external force, for example. If we add to the right-hand side of the relaxation equation a term consisting of a power of t,

```
inHomRelaxation = ∂t f[t] == -1/τ f[t] + (t^(ν-1)/Gamma[ν])
```

$$f'[t] == -\frac{f[t]}{\tau} + \frac{t^{-1+\nu}}{\Gamma[\nu]}$$

where $\nu > 0$ is a real constant. The Laplace transform of the extended relaxation equation is

```
lrelax = ℒ_t^s[inHomRelaxation]
```

$$-f[0] + s\,\mathcal{L}_t^s[f[t]] == s^{-\nu} - \frac{\mathcal{L}_t^s[f[t]]}{\tau}$$

Solving with respect to the Laplace variable, we find

```
lsol = Solve[lrelax, ℒ_t^s[f[t]]] // Flatten
```

$$\left\{\mathcal{L}_t^s[f[t]] \to \frac{s^{-\nu} + f[0]}{s + \frac{1}{\tau}}\right\}$$

The inversion of this result gives us the solution of the inhomogeneous relaxation equation:

```
sol = (ℒ^-1)_s^t[ℒ_t^s[f[t]] /. lsol]
```

$$e^{-\frac{t}{\tau}} f[0] + \frac{e^{-\frac{t}{\tau}} \left(-\frac{1}{\tau}\right)^{-\nu} \gamma[\nu, -\frac{t}{\tau}]}{\Gamma[\nu]}$$

The result is a solution consisting of the homogenous solution and a part determined by the inhomogeneity. This second part is independent of any initial condition.

The three steps necessary to solve an initial value problem for ODEs are incorporated in the function **FractalDSolve[]**; this function not only

allows the solution of ODEs but is especially designed to solve linear fractional differential equations. The following line demonstrates the application of this function to the inhomogeneous relaxation equation:

```
FractalDSolve[inHomRelaxation, f, t]
```

```
{f → Function[t,
    (ℯ^(-t/τ) (-1/τ)^(-ν) ((-1/τ)^ν f[0] Γ[ν] + Γ[ν] - Γ[ν, -t/τ])) / Γ[ν]]}
```

The result is identical to the result derived in the interactive calculation. The function is also useful in solving linear fractional differential equations. The following subsection discusses the solution steps necessary.

7.6.4.3 Linear Fractional Differential Equations

Linear fractional differential equations FDEs are integral equations of the Volterra type. These equations have in common that one part of the equation consists of an integral operator of Riemann–Liouville or Weyl type. In general, a FDE is given by

$$\Delta(t,\, u,\, \mathcal{D}_{0,t}^{-\nu}\, u) \;=\; 0, \tag{7.6.88}$$

where $\nu > 0$ denotes the order of the FDE. An important property of (7.6.88) is the linearity of the equation, meaning

$$\begin{aligned}\Delta(t,\, \alpha\, u + \beta\, v,\; \mathcal{D}_{0,t}^{-\nu}(\alpha\, u + \beta\, v)) = \\ \alpha\, \Delta(t,\; u,\; \mathcal{D}_{0,t}^{-\nu}\, u) + \beta\, \Delta(t,\, v,\; \mathcal{D}_{0,t}^{-\nu}\, v),\end{aligned} \tag{7.6.89}$$

where α and β are constants and $u = u(t)$ and $v = v(t)$ are functions of the independent variable t. This property guarantees that the superposition principle holds and that we can apply integral transforms to solve FDEs. The solution steps are discussed in the following subsection.

7.6.4.4 Solution of FDEs by Integral Transforms

This section describes a solution procedure of linear FDEs by using integral transforms. Integral transforms are efficient methods to solve initial value problems for fractional differential equations. In detail, we discuss the Laplace and Mellin transform technique to solve FDEs.

One of the key steps in solving FDEs is the Laplace transform as a first step. This step allows us to reduce a fractional differential equation to an algebraic equation. We demonstrate this behavior by means of the generalized relaxation equation:

```
Remove[f];
Assume[q > 0];
```

```
Frelaxation = D^q_{0,t}[f[t]] == -1/τ f[t] + α
```

$$\mathcal{D}_t^q[f[t]] == \alpha - \frac{f[t]}{\tau}$$

where q is a positive number and α is related to the initial condition. The Laplace transform of the above equation delivers the algebraic equation

```
lrelax = L^s_t[Frelaxation]
```

```
Conditions to solve the integral:

-1 + Re[q] < 0
```

$$s^q\,\mathcal{L}_t^s[f[t]] == \frac{\alpha}{s} - \frac{\mathcal{L}_t^s[f[t]]}{\tau}$$

The Laplace transform of f is denoted by $\mathcal{L}_t^s[f[t]]$. The solution of this equation in Laplace space follows by solving the above equation with respect to the Laplace representation of f:

```
lsol = Solve[lrelax, ℒ_t^s[f[t]]] // Flatten
```

$$\left\{\mathcal{L}_t^s[f[t]] \to \frac{\alpha}{s\left(s^q + \frac{1}{\tau}\right)}\right\}$$

If we try to apply the inverse Laplace transformation, we end up with an integral which cannot be solved by *Mathematica*:

```
(ℒ^-1)_s^t[ℒ_t^s[f[t]] /. lsol]
```

$$\alpha\, \tau^{1-\frac{1}{q}} \int_0^t (s\, \tau^{-1/q})^{-1+q}\, E_{q,q}\left[-(s\, \tau^{-1/q})^q\right] \mathrm{d}s$$

However, the resolution of the problem is an additional application of a Mellin transform to the Laplace representation of the solution. If we, in addition, shift the Mellin variable, we gain

```
melEq = ℳ_s^σ[lsol] /. {σ -> -σ + 1, Rule -> Equal}
```

$$\left\{\Gamma[1-\sigma]\, \mathcal{M}_t^{\sigma}[f[t]] == -\frac{\pi\, \alpha\, \tau^{1+\frac{\sigma}{q}}\, \mathrm{Csc}\left[\frac{\pi - \pi\,(1-\sigma)}{q}\right]}{q}\right\}$$

This representation of the solution can be solved for the Mellin transform of f, providing us with the solution in Mellin space:

```
smelEq = Solve[melEq, ℳ_t^σ[f[t]]]
```

$$\left\{\left\{\mathcal{M}_t^{\sigma}[f[t]] \to -\frac{\pi\, \alpha\, \tau^{1+\frac{\sigma}{q}}\, \mathrm{Csc}\left[\frac{\pi - \pi\,(1-\sigma)}{q}\right]}{q\, \Gamma[1-\sigma]}\right\}\right\}$$

The inversion of the Mellin transform to the original independent coordinate t by means of the inverse Mellin transform delivers the solution in original coordinates:

```
solution = (M^-1)^t_σ[smelEq] // PowerExpand // Flatten
```

$$\left\{f[t] \to -\alpha\, \tau\, E_{q,1}\left[-\frac{t^q}{\tau}\right]\right\}$$

The derived solution is given by the Mittag–Leffler function. This example contains the necessary steps to derive a solution for an initial value problem. We realize that the method presented is completely algorithmic and can be incorporated into a *Mathematica* function. The steps solving a linear initial value problem for an FDE in f can be summarized as follows:

1. Laplace transform the FDE.

2. Solve the resulting algebraic equation to find the solution in the Laplace variable.

3. Apply the Mellin transform to find a representation of the Laplace solution in Mellin space

4. Invert the Mellin transform to find the solution in original coordinates.

These four steps are graphically shown in Figure 7.6.26.

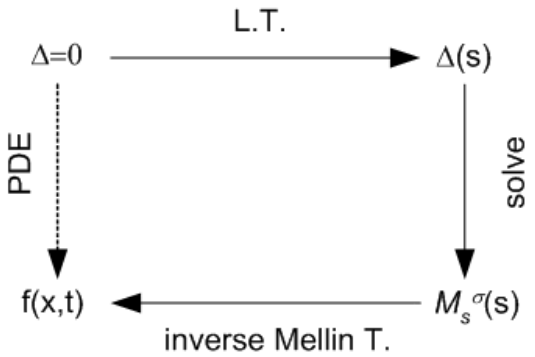


Figure 7.6.26. Solution steps for FDEs by means of Laplace and Mellin transforms.

The method used is restricted to those functions which can be represented by the inverse Mellin transform. In other words, the functions must be given by a Mellin–Barns integral. If this is not the case, the procedure fails to deliver a solution. However, the solution class derived by this method is much larger than the solutions derived by a simple Laplace transform.

To summarize the solution procedure, we started from a linear FDE $\Delta = 0$ of arbitrary order. Laplace transform this equation and solve for the Laplace variable F. An additional transformation to a Mellin representation allows us to gain the solution by an inverse Mellin transform. The inversion of the Mellin solution delivers the solution of the FDE. These steps are always feasible if the coefficients of the derivatives and the functions of the FDE are constants.

The three steps necessary to solve an initial value problem for FDEs are also incorporated in the function **FractalDSolve[]**; this function not only allows the solution of ODEs but is especially designed to solve linear fractional differential equations. The following line demonstrates the application of this function to the inhomogeneous relaxation equation:

```
FractalDSolve[Frelaxation, f, t]
```

$$\{f \to \text{Function}\left[t, -\alpha\, \tau\, E_{q,1}\left[-(t\, \tau^{-1/q})^{q}\right]\right]\}$$

The result is identical to the result derived in the interactive calculation.

7.6.4.5 Fractional Relaxation Equation

Relaxation processes are numerous in physical applications. One of the famous examples is the decay of a β particle. The temporal behavior of such a decay is usually described by a first-order ordinary differential equations. This standard relaxation is also called a Debye process or Debye relaxation.

```
relaxation = ∂t f[t] + 1/τ f[t] == 0
```

$$\frac{f[t]}{\tau} + f'[t] == 0$$

The solution of this equation follows by applying the function **DSolve[]** to the equation

```
sol1 = DSolve[relaxation, f, t]
```

$$\{\{f \to \text{Function}[\{t\}, e^{-\frac{t}{\tau}} C[1]]\}\}$$

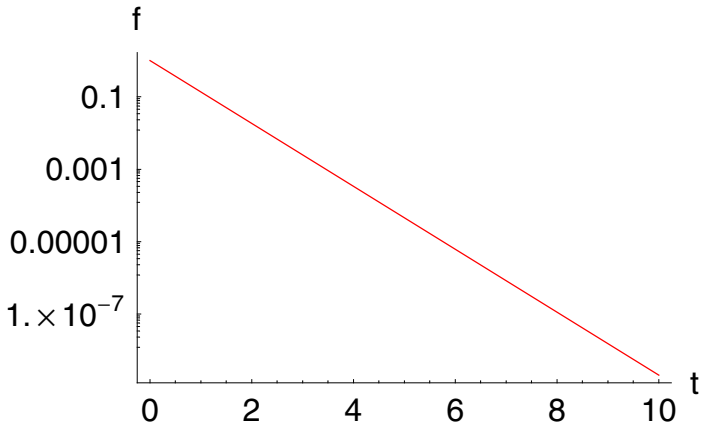
The same solution follows by applying the function **FractalDSolve[]**:

```
sol2 = ℱ𝒟[t, f][relaxation]
```

$$\{f \to \text{Function}[t, e^{-\frac{t}{\tau}} f[0]]\}$$

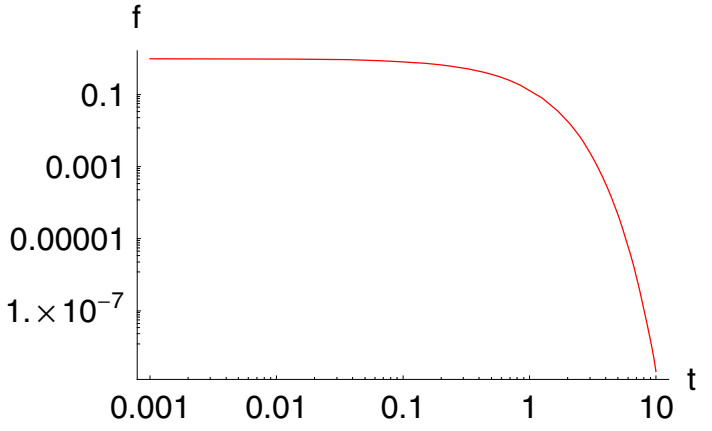
Both solutions contain a single constant $C[1]$ and $f[0]$ determining the initial condition of the relaxation process. The characteristic behavior of a relaxation process is the exponential decay in time, which is a straight line in a log plot of the function

```
LogPlot[f[t] /. sol2 /. {f[0] → 1, τ → 1 / 2},
  {t, .001, 10}, PlotStyle → RGBColor[0.996109, 0, 0],
  AxesLabel → {"t", "f"}];
```



The double logarithmic plot of a relaxation process shows a shoulder and a decay

```
pl1 = LogLogPlot[f[t] /. sol2 /. {f[0] → 1, τ → 1 / 2},
   {t, .001, 10}, PlotStyle → RGBColor[0.996109, 0, 0],
   AxesLabel → {"t", "f"}];
```



Both the log and the log-log plot show that a standard relaxation process decays very fast. The decay of a β particle is a process determined by a single physical cause. However, relaxation processes in complex materials show a different characteristic pattern. The decay in complex materials is much slower than in the standard relaxation case. The asymptotic behavior observed can be described by a power law in time:

$$f(t) \propto t^{-q}, \text{ with } 0 < q < 1. \tag{7.6.90}$$

The range of time extends over many decades. Examples are current distributions at rough blocking electrodes [7.22], charge-carrier transport in amorphous semiconductors [7.23], the dielectric relaxation of liquids [7.24], and relaxation of polymeric networks [7.25–7.27].

One of the nonstandard relaxation models to describe the behavior of complex materials assumes that the material has a memory. This memory includes the total decay for all times. The model discussed by Nonnenmacher [7.28] is applicable to models in which an integral net effect determines the relaxation process. The relaxation equation is generalized in such a way that a regular behavior at the initial time is incorporated into the model. The equation is given by a Fredholm integral equation of first kind expressed by RL differential operators. This kind of relaxation process assumes that the order of differentiation is a positive real number:

```
Assume[q, q > 0];
```

The equation in terms of a RL operator reads

```
eq = D[f[t], {t, q}] - f0 t^-q / Gamma[1 - q] + τ^-q f[t] == 0
```

$$\tau^{-q} f[t] - \frac{f0\, t^{-q}}{\Gamma[1-q]} + \mathcal{D}_t^q[f[t]] == 0$$

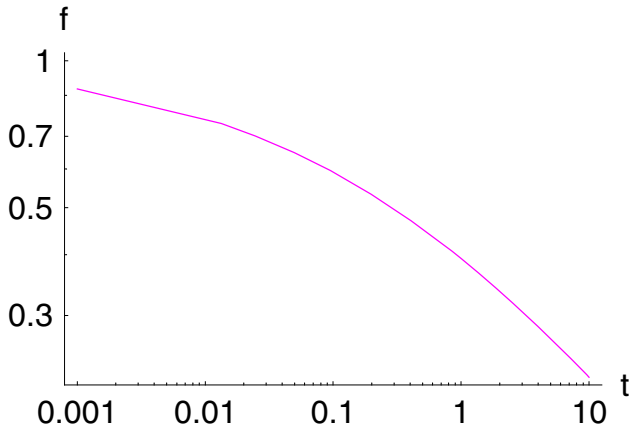
The solution of this equation follows by applying **FractalDSolve[]** to the fractional equation

```
solf = FD_f^t[eq]
```

$$\left\{\mathtt{f} \to \mathtt{Function}\left[\mathtt{t},\ \mathtt{f0}\, E_{\mathtt{q},1}\left[-\left(\frac{\mathtt{t}}{\tau}\right)^{\mathtt{q}}\right]\right]\right\}$$

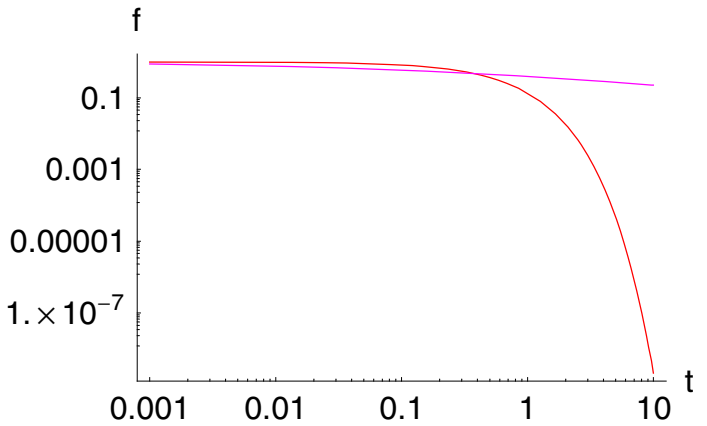
The solution consists of a regular solution containing the initial condition f_0. The generalized Mittag–Leffler function is nonstandard in *Mathematica.* The graphical representation of the Mittag–Leffler function for $q = 1/3$ and $f_0 = 1$ is given by

```
pl2 =
  LogLogPlot[f[t] /. solf /. {f0 → 1, τ → 1 / 2, q → 1 / 3},
   {t, .001, 10}, PlotStyle → RGBColor[0.996109, 0, 1],
   AxesLabel → {"t", "f"}];
```



Comparing the derived nonstandard relaxation result with the standard relaxation solution demonstrates

```
Show[pl1, pl2];
```



that a fractional relaxation process decays much slower than a Debye relaxation. This slower decay of a relaxation process is frequently observed in natural systems.

7.6.4.6 Relaxation Oscillation Equation

Next, let us consider an equation which interpolates between the ordinary relaxation and the oscillation equation. This kind of equation can be considered as a weak form of Newton's equation or a generalization of relaxation processes. The main assumption is that we restrict the order of differentiation to the interval $1 \leq q \leq 2$.

```
Assume[1 < q && q <= 2]
```

```
{{{q > 1, q ≤ 2}, {Im[q] → 0, Re[q] → q}}}
```

The equation under consideration is given by

```
relaxOscill = 𝒟[0, t, q][f[t]] + f[t] == f[0] t^(-q-α)
```

```
f[t] + 𝒟[t, q][f[t]] == t^(-q-α) f[0]
```

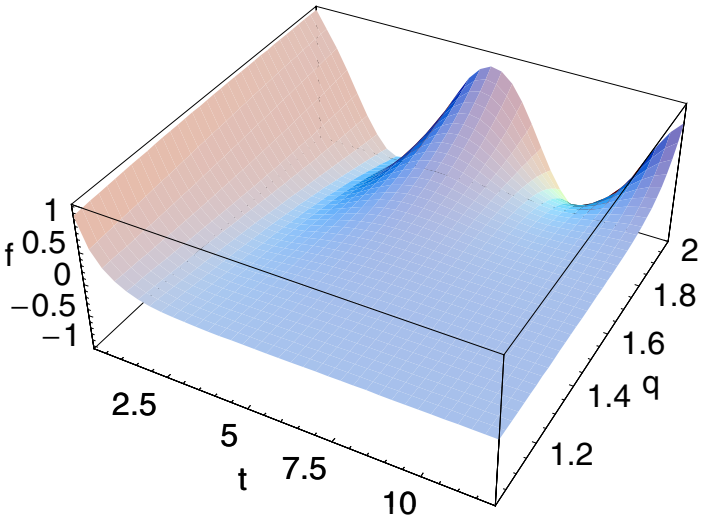
where we specialized the left-hand side of the equation to a power function. This equation is called the relaxation oscillation equation. Applying the fractional solution operator to this equation will deliver the solution

```
sol = ℱ𝒟[t, f][relaxOscill]
```

```
{f → Function[t, t^(-α) f[0] Γ[1 - q - α] E[q, 1-α][-t^q]]}
```

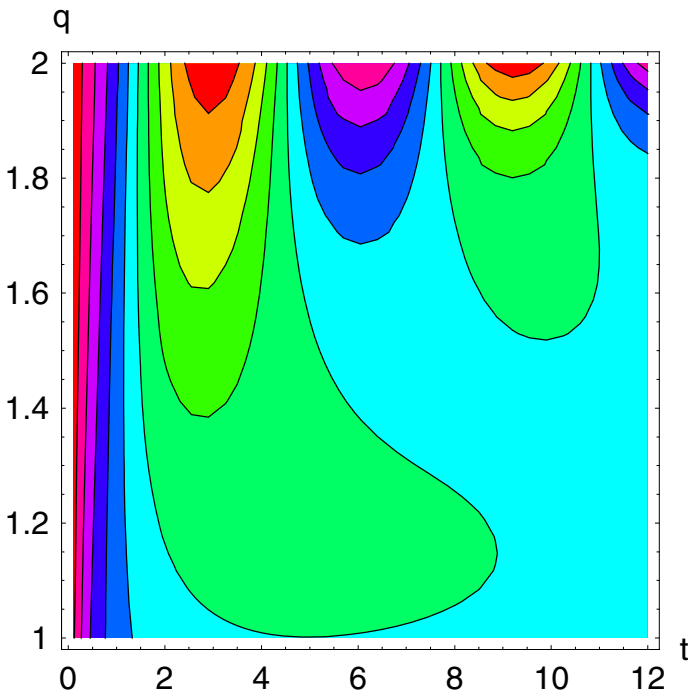
The result is a function determined by the generalized Mittag–Leffler function $E_{q,p}(t)$ providing us the solution manifold for different differentiation orders q. Since the gamma function contained in this solution possesses singularities at different negative integer orders of the arguments, we have to choose the initial conditions in such a way that this singularity is eliminated. We introduce a scaled initial condition $g(0)/\Gamma(1-q-\alpha)$, allowing us to exclude the singularity from the functional domain. However, we must keep in mind that at certain values of $q = 1-\alpha$, negative integers singularities of the function occur. The following plot of the singularity free function shows the transition from relaxation behavior to oscillations. Depending on the fractional order q, we observe that the total relaxation phenomenon is converted to a damped oscillation and then to an undamped oscillation if q increases from 1 to 2.

```
Plot3D[Evaluate[
  f[t] /. sol /. f[0] -> g[0] / Γ[1 - q - α] /. α -> 0.1 /.
   g[0] -> 1], {t, 0.1, 12}, {q, 1.0001, 2},
 AxesLabel -> {"t", "q", "f   "}, PlotRange -> All,
 PlotPoints -> 35, Mesh -> False];
```



The following contour plot of the solution shows that the frequency decreases slightly if q is increased. However, this frequency shift disappears for q values near 2.

```
ContourPlot[Evaluate[
  f[t] /. sol /. f[0] -> g[0] / Γ[1 - q - α] /. α -> 0.1 /.
   g[0] -> 1], {t, 0.1, 12}, {q, 1.0001, 2}, Axes -> True,
 AxesLabel -> {"t", "q"}, PlotRange -> All,
 ColorFunction -> Hue, PlotPoints -> 35];
```



7.6.4.7 Semifractional Differential Equations

Semifractional differential equations are those equations with differential order $q = 1/2$. This kind of equation is in use in different fields of chemistry and physics such as electroanalysis, polymer physics, and so forth. A characteristic equation of relaxation type for a positive relaxation time constant

```
Assume[τ > 0]
```

```
{{{τ > 0}, {Im[τ] → 0, Re[τ] → τ}}}
```

is given by

```
sfDG = 𝒟[0, 1/2, t][f[t]] - f0 t^(-1/2) / Gamma[1/2] - τ^(-1/2) f[t]
```

$$-\frac{f0}{\sqrt{\pi}\,\sqrt{t}}-\frac{f[t]}{\sqrt{\tau}}+\mathcal{D}_t^{\frac{1}{2}}[f[t]]$$

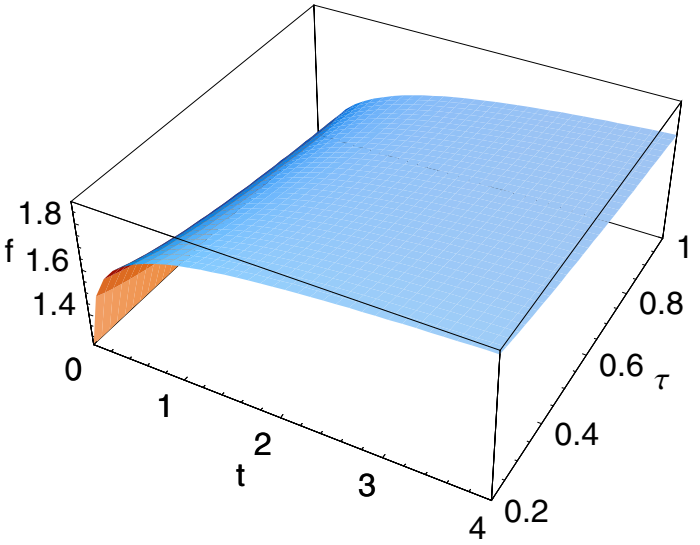
The solution of this equation is derived by applying the fractional solution operator $\mathcal{FD}_f^t$ to the fractional differential equation:

```
sDGSol = ℱ𝒟f[t][sfDG]
```

$$\left\{f\to \text{Function}\left[t,\ f0\left(1-\left(-1+e^{t/\tau}\right)\sqrt{\frac{1}{\tau}}\,\sqrt{\tau}+e^{t/\tau}\sqrt{\frac{1}{\tau}}\,\sqrt{\tau}\,\text{Erf}\left[\sqrt{t}\,\sqrt{\frac{1}{\tau}}\,\right]\right)\right]\right\}$$

The result is a function combining exponentials and error functions. A plot of the solution is given next for different relaxation constants τ.

```
Plot3D[f[t] /. sDGSol /. {f0 -> 1}, {t, 0, 4},
  {τ, 0.2, 1}, AxesLabel -> {"t", "  τ", "f  "},
  Mesh -> False, PlotPoints -> 35];
```



Another example for a semifractional equation is given by the driven rubber equation:

```
drfDG = Φ[ t ]  + τ^-b 𝒟_{0,t}^{-1/2}[Φ[t]]  - a_0 Sin[ω t] == 0
```

$$\tau^{-b}\,\mathcal{D}_t^{-\frac{1}{2}}[\Phi[t]] - \text{Sin}[t\,\omega]\,a_0 + \Phi[t] == 0$$

This kind of equation is used to model the relaxation behavior of rubber driven by a harmonic external force. The solution of the equation is gained by application of the fractional solution operator:

```
solution = FD^t_Φ[drfDG]
```

$$\left\{\Phi \to \text{Function}\left[t,\ \frac{1}{4\,(1+\tau^{4\,b}\,\omega^2)}\right.\right.$$

$$\left(\tau^b\sqrt{\omega}\,\left(\frac{2\sqrt{2\,\pi}\,\sqrt{t\,\omega}\;F_{p,q}\left[\begin{matrix}\{1\}\\ \{\frac{3}{4},\ \frac{5}{4}\}\end{matrix};\ -\frac{1}{4}\,t^2\,\omega^2\right]}{\Gamma\left[\frac{3}{4}\right]\,\Gamma\left[\frac{5}{4}\right]}+\right.\right.$$

$$\tau^b\sqrt{\omega}\,\left(\frac{4\,\pi\,\tau^{2\,b}\,\sqrt{t^2\,\omega^2}}{t\,\Gamma\left[\frac{\sqrt{t^2\,\omega^2}}{\pi}\right]\,\Gamma\left[1-\frac{\sqrt{t^2\,\omega^2}}{\pi}\right]}-\right.$$

$$\frac{4\,\pi}{\Gamma\left[\frac{1}{2}-\frac{\sqrt{t^2\,\omega^2}}{\pi}\right]\,\Gamma\left[\frac{1}{2}+\frac{\sqrt{t^2\,\omega^2}}{\pi}\right]}-$$

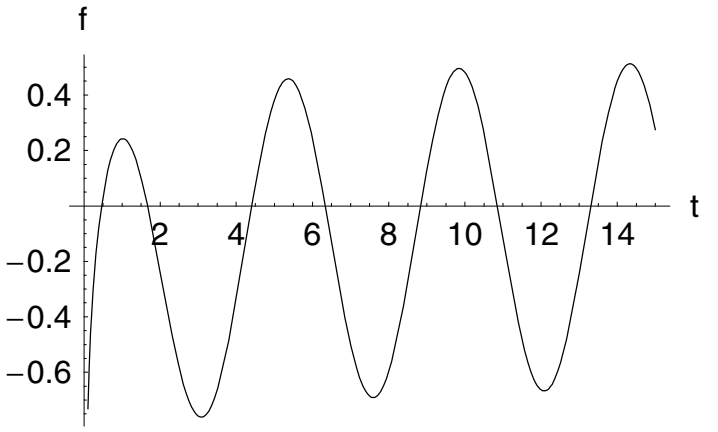
$$\frac{\sqrt{2\,\pi}\;\tau^b\,\sqrt{\omega}\;(t\,\omega)^{3/2}\;F_{p,q}\left[\begin{matrix}\{1\}\\ \{\frac{5}{4},\ \frac{7}{4}\}\end{matrix};\ -\frac{1}{4}\,t^2\,\omega^2\right]}{\Gamma\left[\frac{5}{4}\right]\,\Gamma\left[\frac{7}{4}\right]}$$

$$-$$

$$\left.\left.\left.\left.\left.\frac{4\,\Psi_{1,0}\left[\begin{matrix}\{\{1,\ \frac{1}{2}\}\}\\ \{\}\end{matrix};\ -2\,\sqrt{t\,\tau^{-2\,b}}\right]}{\sqrt{\pi}\,\sqrt{t\,\tau^{-2\,b}}}\right)\right)\,a_0\right)\right]\right\}$$

The result is a combination of generalized hypergeometric functions. This solution demonstrates that a more or less simple initial equation results in a complicated structure of the solution. An example of the solution is shown in the following plot. The parameters used are $\tau = 1$, $b = 1/3$, $a_0 = 1$, and $\omega = 1.4$.

```
Plot[Evaluate[Φ[ t ] /. solution /.
    {τ -> 1, b -> 1 / 3, a0 -> 1, ω -> 1.4}],
  {t, 0.1, 15}, AxesLabel -> {"t", "f"}];
```



7.6.4.8 Anomalous Diffusion

Many experiments indicate that diffusion processes usually do not follow the standard Gaussian behavior. In turn, the mean square displacement $\langle r(t)^2 \rangle \propto t$ for a Gaussian process changes to $\langle r(t)^2 \rangle \propto t^{2/d_w}$, where the anomalous diffusion exponent d_w differs from 2, the value for standard (Fickean) diffusion. The deviation from a linear dependence to a power law is an indication for anomalous diffusion. Anomalous diffusion in which the mean square distance between diffusing quantities increases slower or faster than linearly in time has been observed in different physical and biological systems from macroscopic surface growth to DNA sequences [7.29]. One of the first investigations discussing fractional diffusion goes back to Wyss [7.30] and O'Shaugnessy and Procaccia [7.31]. A method for solving fractional diffusion equations using Fox's H functions has been presented by Schneider and Wyss [7.32] and more recently by Metzler et al. [7.33].

The motivation for the anomalous diffusion equation being discussed now follows the ideas already outlined in the section on fractional relaxation starting from the standard model and generalizing the equation by

incorporating the initial condition. The standard (Fickean) diffusion equation in 1+1-dimensions reads

$$\partial_t \rho(x, t) = \partial_{x,x} \rho(x, t), \tag{7.6.91}$$

with an additional initial condition $\rho(x, t = 0) = \rho_0(x)$. Equation (7.6.91) is given in a scaled representation where the diffusion constant is incorporated as a factor in the time variable. Let us start with the memory-diffusion equation

$$\partial_t \rho(x, t) = \int_0^t K(t - \tau)\, \partial_{x,x} \rho(x, \tau)\, d\tau, \tag{7.6.92}$$

which has already been motivated and derived recently [7.34, 7.35].

Again, as in the case of relaxation, we assume that the memory kernel takes on a power law $K(t) = D_0\, t^{\beta-1} / \Gamma(\beta)$ with $0 < \beta < 1$. Then we can express Eq. (7.6.92) by

$$\partial_t \rho = \frac{D_0}{\Gamma(\beta)} \int_0^t (t - \tau)^{\beta-1}\, \partial_{x,x} \rho(x, \tau)\, d\tau \quad \text{and} \quad \beta > 0, \tag{7.6.93}$$

which, in terms of Riemann–Liouville operators $\mathcal{D}^q_{0,t}$, can be written as

$$\mathcal{D}^1_{0,t} \rho(x, t) = D_0\, \mathcal{D}^{-\beta}_{0,t}(\partial_{x,x} \rho(x, t)). \tag{7.6.94}$$

Applying an integration $\mathcal{D}^{-1}_{0,t}$ to both sides of Eq. (7.6.94), we find

$$\rho(x, t) - \rho_0(x) = D_0\, \mathcal{D}^{-(1+\beta)}_{0,t}(\partial_{x,x} \rho(x, t)). \tag{7.6.95}$$

A differentiation of order $(1 + \beta)$ of Eq. (7.6.95) replaces the integral representation of the generalized diffusion equation by its differential representation

$$\mathcal{D}^{1+\beta}_{0,t} \rho(x, t) - \rho_0(x) \frac{t^{-(1+\beta)}}{\Gamma(-\beta)} = D_0\, \partial_{x,x} \rho(x, t). \tag{7.6.96}$$

This generalized diffusion equation incorporates, in addition to the fractional differentiation in time, the initial condition ρ_0 for the density ρ. Replacing the fractional order $1 + \beta$ by q, we find the simplified equation

$$\mathcal{D}^q_{0,t} \rho(x, t) - \rho_0(x) \frac{t^{-q}}{\Gamma(1-q)} = D_0\, \partial_{x,x} \rho(x, t) \quad \text{with } 1 < q < 2. \tag{7.6.97}$$

The solution of Eq. (7.6.97) follows by the following steps. First, let us assume

```
Assume[q > 0];
```

Next, define the equation

```
equation18 =
 D[q, {0, t}][ρ[x, t]] - ρ0[x] t^-q / Γ[1 - q] == D0 ∂{x,x} ρ[x, t]
```

$$\mathcal{D}^{q}_{t}[\rho[x,\,t]] - \frac{t^{-q}\,\rho_0[x]}{\Gamma[1-q]} == D_0\,\rho^{(2,0)}[x,\,t]$$

Then, apply the Laplace transform to Eq. (7.6.97):

```
equation18Laplace = LaplaceTransform[equation18, t, s]
```

```
Conditions to solve the integral:

-1 + Re[q] < 0
```

$$s^{q}\,\mathcal{L}^{s}_{t}[\rho[x,\,t]] - s^{-1+q}\,\rho_0[x] == \mathcal{L}^{s}_{t}[\rho^{(2,0)}[x,\,t]]\,D_0$$

The second step of the transformation consists of a Fourier transform of the equation in Laplace space:

```
foulapgl2 = FourierTransform[equation18Laplace /.
   {LaplaceTransform[ρ[x, t], t, s] → L[x], LaplaceTransform[∂{x,x} ρ[x, t], t, s] → ∂{x,x} L[x],
    ρ0[x] → DiracDelta[x], C1[x] → 0}]
```

$$-2\,s^{-1+q} + s^{q}\,\mathcal{F}^{k}_{x}[L[x]] == -k^2\,\mathcal{F}^{k}_{x}[L[x]]\,D_0$$

The algebraic solution in Fourier and Laplace space follows by

```
foulaploes2 = Solve[foulapgl2, FourierTransform[L[x]]] // Flatten
```

$$\left\{\mathcal{F}^{k}_{x}[L[x]] \to \frac{2\,s^{-1+q}}{s^{q} + k^2\,D_0}\right\}$$

The application of the inverse Fourier transform on this solution delivers the solution in spatial and Laplacian variables (x, s):

```
laploes2 =
 Map[(ℱ^-1)^x_k[#] &, foulaploes2, {2}] /. L[x] → ℒ^s_t[ρ[x, t]]
```

$$\left\{\mathcal{L}_t^s[\rho[x,\,t]] \to \frac{e^{-(|x|)\sqrt{\frac{s^q}{D_0}}}\, s^{-1+q}\,\sqrt{s^{-q}}}{\sqrt{D_0}}\right\}$$

The result shows that the Laplace solution contains a stretched exponential multiplied by a power function. If we restrict our consideration to the half-space $x > 0$ and assume that the diffusion constant D_0 is positive,

```
Assume[x > 0, C1 > 0];
```

we can represent the result in Mellin space as

```
mellaploes2 =
 ℳ^z_s[laploes2 /. D0 → C1 // PowerExpand] // PowerExpand //
  Simplify
```

$$\left\{\Gamma[z]\,\mathcal{M}_t^{1-z}[\rho[x,\,t]] \to \frac{2\,C1^{\frac{-1+z}{q}}\,x^{-\frac{-2+q+2z}{q}}\,\Gamma\left[\frac{-2+q+2z}{q}\right]}{q}\right\}$$

A shift of the Mellin variable by 1 gives us the Mellin solution

```
melloes2 = Solve[mellaploes2 /. {z → 1 - z, Rule → Equal},
   ℳ^z_t[ρ[x, t]]] // Flatten
```

$$\left\{\mathcal{M}_t^z[\rho[x,\,t]] \to \frac{2\,C1^{-\frac{z}{q}}\,x^{-\frac{-2+q+2(1-z)}{q}}\,\Gamma\left[\frac{-2+q+2(1-z)}{q}\right]}{q\,\Gamma[1-z]}\right\}$$

The inversion of the Mellin transform provides the final result:

```
solution =
 (ℳ^-1)^t_z[melloes2] /. C1 → D0 // PowerExpand // Simplify
```

$$\left\{\rho[x,t] \to \frac{2\,\mathcal{H}^{1,0}_{1,1}\left[\frac{x^{2/q}\,D_0^{-1/q}}{t}\,\middle|\,\begin{matrix}\{\}\mid & \{\{1,1\}\}\\ \{\{1,\frac{2}{q}\}\}\mid & \{\}\end{matrix}\right]}{q\,x}\right\}$$

The solution of the generalized diffusion equation (7.6.97) thus is represented by a Fox's H function of $\mathcal{H}^{1,0}_{1,1}$. This function is given by a series representation as follows:

$$\mathcal{H}^{1,0}_{1,1}\left[\frac{x^{2/q}\,D_0^{-1/q}}{t}\,\middle|\,\begin{matrix}\{\}\mid & \{\{1,1\}\}\\ \{\{1,\frac{2}{q}\}\}\mid & \{\}\end{matrix}\right] = \sum_{k=0}^{\infty}\frac{q\,(-1)^k}{2\,\Gamma\left(1-\frac{q}{2}(1+k)\right)k!}\left(\frac{x^{\frac{2}{q}}\,D_0^{-1/q}}{t}\right)^{\frac{q}{2}(1+k)}. \tag{7.6.98}$$

A graphical representation of the solution is given in Figure 7.6.27.

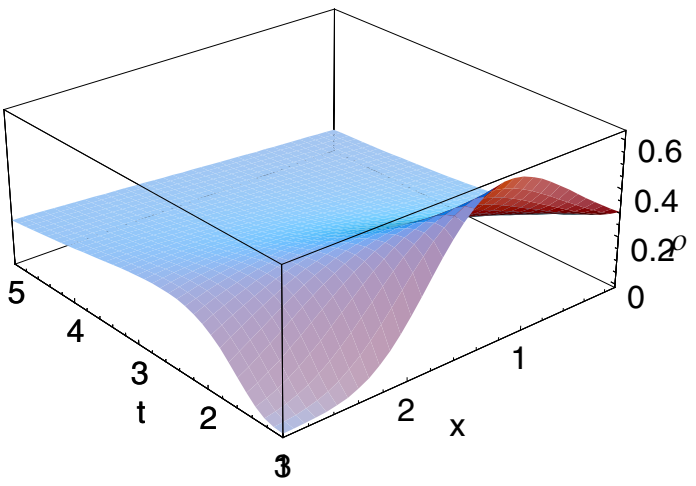


Figure 7.6.27. Solution of the fractional diffusion equation (7.6.97) in the series representation (7.6.98). The fractional exponent is $q = 3/2$ and $D_0 = 1$.

7.7 Exercises

1. Use the **Tree[]** function to create different kinds of trees. Which option determines the shape of a tree?

2. Extend the **Koch[]** function to other generators (e.g., the Peano curve). For a set of generators consult the book by Mandelbrot [7.4].

3. Examine the multifractal properties of a system with different numbers of probabilities and scaling factors. Determine the fractal dimensions D_0 and D_1 and give a graphical representation of these dimensions versus the number of scaling factors.

4. Use hexagonal lattices in the renormalization procedure for the percolation model.

7.8 Packages and Programs

7.8.1 Tree Generation

This package allows one to generate a fractal tree.

```
BeginPackage["Tree`"];

Needs["Geometry`Rotations`"];

Clear[Tree, rotateLine, branchLine, createBranches];

Tree::usage = "Tree[options___] creates a fractal
tree. The options of
the function Tree determine the form of the fractal
created. Options are
        Generation -> 10, \n
        BranchRotation -> 0.65, \n
        BranchScaling -> 0.75, \n
        BranchThickness -> 0.7,  \n
        OriginalThickness -> 0.07, \n
        BranchColor -> {RGBColor[0,0,0]} \n
Example: Tree[BranchColor->l1,BranchRotation->0.3] \n
```

```
     ll is a list created in the package Tree.";

(* --- global variables --- *)

Generation::usage;
BranchRotation::usage;
BranchSkaling::usage;
BranchThickness::usage;
OriginalThickness::usage;
BranchColorn::usage;

Begin["`Private`"];

(* --- rotate a line --- *)
rotateLine[Line[{start_, end_}], angle_] :=
Line[{end, end + branchScaling*
         Rotate2D[end - start, angle Random[Real,
{0.5,1.5}]
                 ]}];

(* --- branch a line --- *)
branchLine[Line[points_]] :=
{rotateLine[Line[points],
                  branchRotation],
                  rotateLine[Line[points],
                  - branchRotation]};

(* --- change thickness --- *)
branchLine[Thickness[th_]] := Thickness[th
branchThickness];

(* --- define color of a branch --- *)
branchLine[RGBColor[r_, g_, b_]] := (
            branchColor = RotateLeft[branchColor];
            First[branchColor] );

(* --- create branches --- *)
createBranches[lines_] := Flatten[Map[branchLine,
lines]];

(* --- options if Tree[] --- *)
Options[Tree] =
      {
        Generation -> 10,
        BranchRotation -> 0.65,
        BranchScaling -> 0.75,
        BranchThickness -> 0.7,
        OriginalThickness -> 0.07,
```

```
        BranchColor ->
                {
                  RGBColor[0,0,0]}
      };

(* --- create a tree --- *)

Tree[options___] := Block[
   {generations, branchRotation,
    branchScaling, branchThickness,
    originalThickness, branchColor},
(* --- check options --- *)
    {generations, branchRotation, branchScaling,
branchThickness,
     originalThickness, branchColor} =
    {Generation, BranchRotation, BranchScaling,
BranchThickness,
     OriginalThickness, BranchColor}  /. {options}
                /. Options[Tree];
(* --- iterate the functions and display the tree
--- *)
    Show[
         Graphics[
            NestList[
        createBranches,
                {
                        First[branchColor],
            Thickness[originalThickness],
                        Line[{{0,0},{0,1}}]
                 },
         generations]],
        FilterOptions[Show, options],
            AspectRatio -> Automatic,
            PlotRange -> All]];

(* --- filter for options --- *)

FilterOptions[ command_Symbol, opts___] :=
      Block[{keywords = First /@ Options[command]},
           Sequence @@ Select [{opts},
                MemberQ[keywords, First[#]]&]];
End[];
EndPackage[];

(* --- an example of a color list --- *)


l1 = {RGBColor[0.5620000000000001, 0.236, 0.071],
```

```
      RGBColor[0.5470000000000001, 0.229,
0.06900000000000001],
      RGBColor[0.5, 0.21, 0.063], RGBColor[0.469,
0.196, 0.059],
      RGBColor[0.033, 0.281, 0.035], RGBColor[0.046,
0.395, 0.05],
      RGBColor[0.055, 0.469, 0.059],
      RGBColor[0.07000000000000001, 0.602, 0.076],
      RGBColor[0.085, 0.727, 0.092], RGBColor[0.109,
0.937, 0.118],
      RGBColor[0.013, 0.75, 0.028]};
```

7.8.2 Koch Curves

This package generates fractal curves of a different kind.

```
BeginPackage["Koch`"];

Clear[Koch,VKoch,WKoch,QKoch,Quad,NGon,docurve,Fracta
l,FaktalPlot];

Needs["Geometry`Rotations`"];

Fractal::usage = "Fractal[curve_String, options___]
creates a graphical
representation of a fractal curve. The type of curve
is determined by
the first argument. A list of available curves is
obtained by calling
Fractal[List] or Fractal[Help]. The second argument
allows to change the
options of the function. The default values are
Generations -> 3,
Angle -> Pi/6 and Corners -> 6.";

Generations::usage;
Angle::usage;
Corners::usage;

Begin["`Private`"];

(* --- generator of the Koch curve --- *)
(*        __/\__                *)

Koch[Line[{StartingPoint_,EndPoint_}]]:=Block[{fActor
```

```
, angle, liste={}},
      fActor = 1/3;
      angle  = Pi/3;
      l1 = StartingPoint;
      l2 = StartingPoint+(EndPoint -
StartingPoint)*fActor;
      AppendTo[liste,Line[{l1,l2}]];
      l1 = l2;
      l2 = l2 +
Rotate2D[EndPoint-StartingPoint,-angle,{0,0}]*fActor;
      AppendTo[liste,Line[{l1,l2}]];
      l1 = l2;
      l2 = l2 +
Rotate2D[EndPoint-StartingPoint,angle,{0,0}]*fActor;
      AppendTo[liste,Line[{l1,l2}]];
      l1 = l2;
      l2 = l2 +
Rotate2D[EndPoint-StartingPoint,0,{0,0}]*fActor;
      AppendTo[liste,Line[{l1,l2}]]
      ];

(* --- generator of an altered Koch curve --- *)
(*        ____/\                    *)

VKoch[Line[{StartingPoint_,EndPoint_}]]:=Block[{fActo
r, angle, liste={}},
      fActor = 1/3;
      angle  = Pi/3;
      l1 = StartingPoint;
      l2 = StartingPoint+(EndPoint -
StartingPoint)*fActor;
      AppendTo[liste,Line[{l1,l2}]];
      l1 = l2;
      l2 = l2 +
Rotate2D[EndPoint-StartingPoint,0,{0,0}]*fActor;
      AppendTo[liste,Line[{l1,l2}]];
      l1 = l2;
      l2 = l2 +
Rotate2D[EndPoint-StartingPoint,-angle,{0,0}]*fActor;
      AppendTo[liste,Line[{l1,l2}]];
      l1 = l2;
      l2 = l2 +
Rotate2D[EndPoint-StartingPoint,angle,{0,0}]*fActor;
      AppendTo[liste,Line[{l1,l2}]]
      ];

(* --- generator of the Koch curve with variable
base angle --- *)
```

```
WKoch[Line[{StartingPoint_,EndPoint_}]]:=Block[{fActo
r, liste={}},
      fActor = 1/(2*(1+Cos[angle]));
      l1 = StartingPoint;
      l2 = StartingPoint+(EndPoint -
StartingPoint)*fActor;
      AppendTo[liste,Line[{l1,l2}]];
      l1 = l2;
      l2 = l2 +
Rotate2D[EndPoint-StartingPoint,-angle,{0,0}]*fActor;
      AppendTo[liste,Line[{l1,l2}]];
      l1 = l2;
      l2 = l2 +
Rotate2D[EndPoint-StartingPoint,angle,{0,0}]*fActor;
      AppendTo[liste,Line[{l1,l2}]];
      l1 = l2;
      l2 = l2 +
Rotate2D[EndPoint-StartingPoint,0,{0,0}]*fActor;
      AppendTo[liste,Line[{l1,l2}]]
      ];

(* --- generator of the rectangular Koch curve --- *)
(*
          __
        __|  |   __
          |__|               *)

QKoch[Line[{StartingPoint_,EndPoint_}]]:=Block[{fActo
r, angle, liste={}},
      fActor = 1/4;
      angle  = Pi/2;
      l1 = StartingPoint;
      l2 = StartingPoint+(EndPoint -
StartingPoint)*fActor;
      AppendTo[liste,Line[{l1,l2}]];
      l1 = l2;
      l2 = l2 +
Rotate2D[EndPoint-StartingPoint,-angle,{0,0}]*fActor;
      AppendTo[liste,Line[{l1,l2}]];
      l1 = l2;
      l2 = l2 +
Rotate2D[EndPoint-StartingPoint,0,{0,0}]*fActor;
      AppendTo[liste,Line[{l1,l2}]];
      l1 = l2;
      l2 = l2 +
Rotate2D[EndPoint-StartingPoint,angle,{0,0}]*fActor;
      AppendTo[liste,Line[{l1,l2}]];
      l1 = l2;
```

```
      l2 = l2 +
Rotate2D[EndPoint-StartingPoint,angle,{0,0}]*fActor;
      AppendTo[liste,Line[{l1,l2}]];
      l1 = l2;
      l2 = l2 +
Rotate2D[EndPoint-StartingPoint,0,{0,0}]*fActor;
      AppendTo[liste,Line[{l1,l2}]];
      l1 = l2;
      l2 = l2 +
Rotate2D[EndPoint-StartingPoint,-angle,{0,0}]*fActor;
      AppendTo[liste,Line[{l1,l2}]];
      l1 = l2;
      l2 = l2 +
Rotate2D[EndPoint-StartingPoint,0,{0,0}]*fActor;
      AppendTo[liste,Line[{l1,l2}]]
      ];

(* --- generator for a rectangular curve --- *)
(*
           __
        __|  |__           *)

Quad[Line[{StartingPoint_,EndPoint_}]]:=Block[{fActor
, angle, liste={}},
      fActor = 1/3;
      angle  = Pi/2;
      l1 = StartingPoint;
      l2 = StartingPoint+(EndPoint -
StartingPoint)*fActor;
      AppendTo[liste,Line[{l1,l2}]];
      l1 = l2;
      l2 = l2 +
Rotate2D[EndPoint-StartingPoint,-angle,{0,0}]*fActor;
      AppendTo[liste,Line[{l1,l2}]];
      l1 = l2;
      l2 = l2 +
Rotate2D[EndPoint-StartingPoint,0,{0,0}]*fActor;
      AppendTo[liste,Line[{l1,l2}]];
      l1 = l2;
      l2 = l2 +
Rotate2D[EndPoint-StartingPoint,angle,{0,0}]*fActor;
      AppendTo[liste,Line[{l1,l2}]];
      l1 = l2;
      l2 = l2 +
Rotate2D[EndPoint-StartingPoint,0,{0,0}]*fActor;
      (* l2 = l2 + EndPoint*fActor;*)
      AppendTo[liste,Line[{l1,l2}]]
      ];
```

```
(* --- generator for a N gon --- *)

NGon[Line[{StartingPoint_,EndPoint_}]]:=Block[{l1,
l2, angle, liste={}},
      angle = 2*Pi/corners;
      l1 = StartingPoint;
      l2 = StartingPoint+(EndPoint - StartingPoint);
      AppendTo[liste,Line[{l1,l2}]];
      Do[
      l1 = l2;
      l2 = l2 +
Rotate2D[EndPoint-StartingPoint,k*angle,{0,0}];
      AppendTo[liste,Line[{l1,l2}]],
      {k,1,corners-1}];
      liste
      ];

(* --- calculate the higher iterations --- *)

docurve[Type_,Linie_]:=Block[{},
      Flatten[Map[Type,Linie]]
      ];

(* --- plot of a line sequence --- *)

FractalPlot[x_]:=Show[Graphics[x],AspectRatio->Automa
tic];

(* --- options for Fractal[] --- *)

Options[Fractal]:={
        Generations -> 3,
        Angle -> Pi/6,
        Corners -> 6
        };

(* --- create the fractal curve --- *)

Fractal[curve_, options___]:=Block[{generations,
angle, corners},
(* --- check options --- *)
      {generations,angle,corners} =
{Generations,Angle,Corners}
        /. {options} /. Options[Fractal];
(* --- menu for the different fractal curves --- *)
      If[curve == "List" || curve == "Help",
      Print[" "];
      Print[" --------- available curves ---------"];
```

```
      Print["  Koch   :  Koch curve"];
      Print["  QKoch  :  rectangular Koch curve"];
      Print["  VKoch  :  altered Koch curve"];
      Print["  WKoch  :  variable angle Koch curve"];
      Print["  Quad   :  rectangular curve"];
      Print["  Star   :  Koch star"];
      Print["  Square :  Koch square"];
      Print["  N-gon  :  Koch N gon"];
      Print["  Mixture:  2 x Koch and 2 x QKoch"];
      Print["  Random :  random generation"]];

(* --- plot the Koch curves --- *)
          If[curve == "Koch" ||
          curve == "QKoch" ||
          curve == "VKoch" ||
          curve == "WKoch" ||
          curve == "Quad",
(* --- ToExpression transforms a string to an
expression --- *)
         k1 =
ToExpression[curve][Line[{{0,0},{1,0}}]];
          Do[
          k1 = docurve[ToExpression[curve],k1],
          {k,1,generations}];
          FractalPlot[k1]
        ];
(* --- plot a Koch star --- *)
        If[curve == "Star",
          corners = 3;
          k1 = NGon[Line[{{0,0},{1,0}}]];
          Do[
          k1 = docurve[Koch,k1],
          {k,1,generations}];
          FractalPlot[k1]
          ];
(* --- plot a Koch square --- *)
        If[curve == "Square",
          corners = 4;
          k1 = NGon[Line[{{0,0},{1,0}}]];
          Do[
          k1 = docurve[Koch,k1],
          {k,1,generations}];
          FractalPlot[k1]
          ];
(* --- plot a Koch N gon --- *)
        If[curve == "N-gon",
          k1 = NGon[Line[{{0,0},{1,0}}]];
          Do[
```

```
          k1 = docurve[Koch,k1],
          {k,1,generations}];
          FractalPlot[k1]
          ];
(* --- plot a mixture of Koch curves --- *)
        If[curve == "Mixture",
          k1 = Koch[Line[{{0,0},{1,0}}]];
          k1 = docurve[Koch,k1];
          k1 = docurve[QKoch,k1];
          k1 = docurve[QKoch,k1];
          FractalPlot[k1]
          ];
(* --- plot a random sequence of Koch curves --- *)
        If[curve == "Random",
          listec ={Koch,QKoch,VKoch,WKoch,Quad,NGon};
          k2 = Random[Integer,{1,6}];
          k3 = Random[Integer,{1,6}];
          If[k2 == 6 || k3 == 6, corners =
                          Random[Integer,{3,12}]];
          name1 = listec[[k2]];
          name2 = listec[[k3]];
          k1 = name1[Line[{{0,0},{1,0}}]];
          k1 = docurve[name1,k1];
          Do[
          k1 = docurve[name2,k1],
          {k,1,generations-1}];
          FractalPlot[k1]
          ];
      ];
End[];
EndPackage[];
```

7.8.3 Multifactals

The multifractal package provides functions for the determination of multifractal spectra.

```
BeginPackage["MultiFractal`"];

Clear[Dq, Tau, Alpha, MultiFractal];

MultiFractal::usage =
  "MultiFractal[p_List,r_List] calculates the
multifractal spectrum D_q for a model
```

```
    based on the probabilities p and the
scaling factors r. This function plots
    five functions Tau(q), D_q(q),
Alpha(q), f(q) and f(Alpha).";

Begin["`Private`"];

(*---calculate the multifractal dimensions---*)

Dq[p_List, r_List] := Block[{l1, l2, listrg = {}},
   (*---length of the lists---*)l1 = Length[p];
   l2 = Length[r];
   If[l1 == l2,
    (*---variation of q and determination of D_q---*)
    Do[gl1 = Sum[p[[j]]^q r[[j]]^((q - 1) Dfractal),
        {j, 1, l1}] - 1;
     result = FindRoot[gl1 == 0, {Dfractal, -3, 3}];
     result = -Dfractal /. result;
     (*---collect the results in a list---*)
     AppendTo[listrg, {q, result}],
     {q, -10, 10, .101}], Print[" "];
    Print["  Lengths of lists are different!"];
    listrg = {}];
   listrg];

(*---calculate Tau---*)

Tau[result_List] := Block[{l1, listtau = {}},
   (*---lengths of the lists---*)
   l1 = Length[result];
   (*---calculate Tau---*)Do[
    AppendTo[listtau, {result[[k, 1]], result[[k, 2]]
       (1 - result[[k, 1]])}], {k, 1, l1}];
   listtau];

(*---Legendre transform---*)

Alpha[result_List] := Block[{l1, dq, listalpha = {},
    listf = {}, listleg = {}, mlist = {}, pl1, pl2},
   (*---lengths of the lists---*)
   l1 = Length[result];
   (*---determine the differential dq---*)
```

```
    dq = (result[[2, 1]] - result[[1, 1]]) 2;
    (*---calculate Alpha by numerical
       differentiation---*)Do[
     AppendTo[listalpha, {result[[k, 1]],
       (result[[k + 1, 2]] - result[[k - 1, 2]]) /
        dq}], {k, 2, l1 - 1}];
    l2 = Length[listalpha];
    (*---calculate f and
       collect the results in a list---*)Do[
     AppendTo[listf, {result[[k, 1]], -(result[[k, 1]]
           listalpha[[k, 2]] - result[[k, 2]])}];
     listalpha[[k, 2]] = -listalpha[[k, 2]],
     {k, 1, l2}];
    (*---list of the Legendre transforms---*)
    Do[AppendTo[listleg,
      {listalpha[[k, 2]], listf[[k, 2]]}];
     AppendTo[mlist, listf[[k, 2]]], {k, 1, l2}];
    (*---plot f and alpha versus q---*)pl1 =
     ListPlot[listalpha, PlotJoined → True, AxesLabel →
       {"q", "α"}, Prolog → Thickness[0.001]];
    pl2 = ListPlot[listf, PlotJoined → True, AxesLabel →
       {"q", "f"}, Prolog → Thickness[0.001]];
    Show[{pl1, pl2}, AxesLabel → {"q", "α,f"}];
    (*---plot the Legendre
       transform f versus alpha---*)
     ListPlot[listleg, AxesLabel → {"α", "f"}];
    (*---print the maximum of f=D_ 0---*)
    maxi = Max[mlist];
    Print[" "];
    Print["   D_0 = ", maxi]];

(*---calculate the multifractal properties---*)

MultiFractal[p_List, r_List] :=
  Block[{listDq, listTau},
   (*---determine D_q---*)listDq = Dq[p, r];
   ListPlot[listDq, PlotJoined → True, AxesLabel →
     {"q", "Dq"}, Prolog → Thickness[0.001]];
   (*---calculate Tau---*)listTau = Tau[listDq];
   ListPlot[listTau, PlotJoined → True, AxesLabel →
     {"q", "τ"}, Prolog → Thickness[0.001]];
   (*---determine the Hoelder exponent---*)
```

```
   Alpha[listTau]];
End[];
EndPackage[];
```

7.8.4 Renormalization

This package supports the calculations of renormalization.

```
BeginPackage["Renormalization`"];

Clear[f, Pcrit, Nc, Dim, RenormPlot];

Nc::usage = "Nc[n_] determines the mean number of
atoms at the probability
p_c if m is changed in the range 1 <= m <= n-2. The
size of the block is
determined by n.";

Dim::usage = "Dim[n_] calculates the fractal
dimension for the critical
probability p_c. The dimension depends on m where 1
<= m <= n-2, n is
the size of the block used.";

Pcrit::usage = "Pcrit[n_] determines the critical
probability for an n x n
grid under the variation of m where m is the number
of empty locations in the
grid. The range of m is 1 <= m <= n-2.";

RenormPlot::usage = "RenormPlot[n_,typ_String] plots
the functions Nc, Dim
or Pcrit.";

Begin["`Private`"];

(* --- auxilary function --- *)

f[p_,n_,m_]:=
Sum[Binomial[n,i]*p^(n-i)*(1-p)^i,{i,0,m}];

(* --- mean number of particles on a grid --- *)

Nc[n_]:= Block[{p, ncliste={}},
          p = Pcrit[n];
```

```
          Do[
         AppendTo[ncliste,

Sum[Binomial[n,i]*(n-i)*p[[k]]^(n-i-1)*(1-p[[k]])^i,
            {i,0,k}]],
          {k,1,n-2}];
          ncliste
          ];
(* --- fractal dimension at the critical probability
--- *)

Dim[n_]:=  N[Log[Nc[n]]/Log[Sqrt[n]]];


(* --- critical probability on a n x n grid --- *)

Pcrit[n_]:=Block[{ph, p, erg, erg1, gl1, pliste1={}},
       If[n > 2,
          Do[
          gl1 = p - f[p,n,i];
      (* --- solution of the fixpoint equation --- *)
          erg = NSolve[gl1==0,p];
          erg = p /. erg;
      (* --- use only real solutions --- *)
          erg1 = {};

Do[If[Head[erg[[k]]]==Real,AppendTo[erg1,erg[[k]]]],
          {k,1,Length[erg]}];
      (* --- looking for solutions between 0 and 1
--- *)
          erg = Sort[erg1];
          erg1 = {};
          Do[If[erg[[k]] > 0.0 ,
            AppendTo[erg1, erg[[k]] ] ],
          {k,1,Length[erg]}];
          ph = Min[erg1];
          AppendTo[pliste1,ph],
          {i,1,n-2}],
        Print["   "];
        Print[" choose n > 2 "]];
          pliste1];

(* --- plot the results --- *)

RenormPlot[n_,typ_String]:=Block[{},
      If[typ == "Pcrit",
     ListPlot[Pcrit[n],AxesLabel->{"m","pc"}],
     If[typ == "Nc",
```

```
        ListPlot[Nc[n],AxesLabel->{"m","Nc"}],
        If[typ == "Dim",
          ListPlot[Dim[n],AxesLabel->{"m","D"}],
          Print[" "];
          Print["  Wrong key word use: "];
          Print["  Pcrit, Nc or Dim.   "];
          Print[" "]
        ]
     ]
      ]
      ];
End[];
EndPackage[];
```

7.8.5 Fractional Calculus

Define the global variable $FractionalCalculusPath in such a way that the location of the package **FractionalCalculus** is uniquely defined.

```
$FractionalCalculusPath =
 $AddOnsDirectory <> "/Applications/FracCalc/";
AppendTo[$Path, $FractionalCalculusPath];
```

Load the package:

```
<< FractionalCalculus.m
```

```
--> FractionalCalculus ready <--
```



```
<< Integral.m
```

```
-- "Integral.m" is available. --
```

```
NotebookClose[foxtitle];
```

A
Appendix

This appendix contains some information on the installation of the accompanying software and a short description of the functions defined in the packages. It also summarizes the *Mathematica* functions used in the book.

A.1 Program Installation

The book is accompanied by a CD containing all *Mathematica* notebooks. These notebooks can be used as interactive text. The notebooks are linked to a style file called ScriptStyle.nb and Vortrag.nb. You should copy these two files to the location where the additional style files are stored. For example, on a PC, the style files are located at C:\WINDOWS\Profiles\All_Users\Applications\Mathematica\SystemFiles\FrontEnd\StyleSheets.

In addition to the notebooks, there is the package **EulerLagrange** which is delivered with the text. The package is used in Chapter 2. In this chapter you have to change the path name in the sections Packages and Programs.

You can either use the package from the CD or you can copy the package to your preferred location. In any case, you have to change the path name of the package.

Other packages supporting calculations of the text are located in the section Packages and Programs in the appropriate notebook. For these packages, there is no need to set any path names. They are ready to use for your calculations.

A.2 Glossary of Files and Functions

This section contains a short description of all functions defined in the packages of this book. The packages are alphabetically listed.

A.2.1 Anharmonic Oscillator

Anharmonic oscillator of quantum mechanics.

- **AsymptoticPT**

AsymptoticPT[N_,kin_] determines the asymptotic approximation for $|x| \to \infty$ for the continuous case of eigenvalues in a Pöschel–Teller potential. The function yields an analytical expression for $|b(k)|^2$. The variables Transmission and Reflection contain the expressions for the transmission and the reflection coefficients. w1a and w2a contain the approximations for $x \to -\infty$ and $x \to \infty$, respectively.

- **PoeschelTeller**

PoeschelTeller[x_, n_, N_] calculates the eigenfunction of the Pöschel–Teller potential for discrete eigenvalues. N determines the depth of the potential $V_0 \, \mathrm{Sech}(x)$ by $V_0 = N(N+1)$. n fixes the state where $0 < n \leq N$.

- **PlotPT**

PlotPT[kini_,kend_,type_] gives a graphical representation of the reflection or transmission coefficient depending on the value of the variable type. If type is set to the string *r,* the reflection coefficient is plotted. If type is set to *t,* the transmission coefficient is represented. This function creates five different curves.

- **Reflection**

Variable containing the reflection coefficient. The independent variables are N and k.

- **Transmission**

Variable containing the expression for the transmission coefficient. The independent variables are N and k.

- **w1a**

The variable contains the analytic expression for the asymptotic approximation for $x \to -\infty$.

- **w2a**

The variable contains the analytic expression for the asymptotic approximation for $x \to \infty$.

A.2.2 Boundary Value Problem of Electrodynamics

Boundary value problem of electrodynamics.

- **Potential**

Potential[boundary_,R_,alpha_,n_] calculates the potential in a circular segment. Input parameters are the potential on the circle, the radius R of the circle, and the angle of the segment of the circle. The last argument n determines the number of expansion terms used to represent the solution.

A.2.3 Central Field Problem in Quantum Mechanics

Quantum mechanical description of motion in a central field.

- **Angle**

  Angle[theta_, phi_, l_, m_] calculates the angular part of the wave function for an electron in the Coulomb potential. The numbers L and m denote the quantum numbers for the angular momentum operator. θ and ϕ are the angles in the spherical coordinate system.

- **AnglePlot**

  AnglePlot[pl_,theta_,phi_] gives a graphical representation of the function contained in pl. The range of representation is $\pi \leq \phi < 5\pi/2$ and $0 < \theta < \pi$. θ is measured with respect to the vertical axis. This function is useful for plotting the orbitals and the angular part of the eigenfunction.

- **Orbital**

  Orbital[theta_,phi_,l_,m_,type_String] calculates the superposition of two wave functions for the quantum numbers $m_l = +m$ and $m_l = -m$. The variable type allows the creation of the sum or the difference of the wave functions. The string values of type are either plus or minus.

- **Radial**

  Radial[ro_, n_, l_, Z_] calculates the radial representation of the eigenfunctions for an electron in the Coulomb potential. The numbers n and l are the quantum numbers for the energy and the angular momentum operator. Z specifies the number of charges in the nucleus. The radial distance between the center and the electron is given by ρ.

A.2.4 Harmonic Oscillator in Quantum Mechanics

- **a**

a[psi_, xi_:x] is annihilation operator for eigenfunction ψ. The second argument specifies the independent variable of the function ψ.

- **across**

across[psi_, xi_:x] is creation operator for eigenfunction ψ. The second argument specifies the independent variable of ψ.

- **Psi**

Psi[xi_,n_] represents the eigenfunction of the harmonic oscillator. The first argument ξ is the spatial coordinate. The second argument n fixes the eigenstate.

- **wcl**

wcl[xi_,n_] calculates the classical probability of locating the particle in the harmonic potential. The first argument ξ is the spatial coordinate and n determines the energy given as the eigenvalue.

- **wqm**

wqm[xi_,n_] calculates the quantum mechanical probability for an eigenvalue state n. The first argument represents the spatial coordinate.

_

A.2.5 Korteweg–de Vries Equation

Multisoliton solution of the Korteweg–de Vries equation.

- **Soliton**

Soliton[x_,t_,N_] creates the N soliton solution of the KdV equation.

- **PlotKdV**

PlotKdV[tmin_,tmax_,dt_,N_] calculates a sequence of pictures for the N soliton solution of the KdV equation. The time interval of the

representation is $[t_{min}, t_{max}]$. The variable *dt* measures the length of the time step.

A.2.6 Korteweg de Vries equation and its derivation

Derivation of the Korteweg de Vries equation.

- **Equation**

Equation[n_] calculates the evolution equation up to order *n*.

A.2.7 Korteweg–de Vries Equation and Integrals of Motion

Integral of motion of the Korteweg–de Vries equation.

- **Gardner**

Gardner[N_] calculates the densities of the integrals of motion for the KdV equation using Gardner's method. The integrals are determined up to the order *N*.

A.2.8 Korteweg–de Vries Equation Numerical Solution

Numerical solution of the Korteweg–de Vries equation.

- **KdVNIntegrate**

KdVNIntegrate[initial_,dx_,dt_,M_] carries out a numerical integration of the KdV equation using the procedure of [3.5]. The input parameter initially determines the initial solution in the procedure (e.g., $-6\,\mathrm{Sech}^2(x)$). The infinitesimals *dx* and *dt* are the steps with respect to the spatial and temporal directions. *M* fixes the number of steps along the *x*-axis.

A.2.9 Koch Curves

Fractal curves.

- **Fractal**

Fractal[curve_String, options___] creates a graphical representation of a fractal curve. The type of curve is determined by the first argument. A list of available curves is obtained by calling Fractal[List] or Fractal[Help]. The second argument allows changing the options of the function. The default values are Generations $\rightarrow 3$, Angle $\rightarrow \pi/6$ and Corners $\rightarrow 6$.

A.2.10 Light Beam Near a Planet

The bending of a light beam near a planet is discussed.

- **Deviation**

Deviation[radius_,mass_] calculates the numerical value of the light bending in a gravitational field of a star with mass M in a distance radius of the center.

- **Orbit**

Orbit[radius_,mass_] plots the orbit of a light beam near a mass in the distance radius. The calculation is done in Schwarzschild metric.

A.2.11 Multifractal Properties

Multifractal properties of point sets.

- **MultiFractal**

MultiFractal[p_List,r_List] calculates the multi-fractal spectrum D_q for a model based on the probabilities p and the scaling factors r. This function plots five functions $\tau(q)$, $D_q(q)$, $\alpha(q)$, $f(q)$, and $f(\alpha)$.

A.2.12 Penning Trap

Motion of two ions in a Penning trap.

- **PenningCMPlot**

PenningCMPlot[x0_,y0_,x0d_,y0d_,w_] gives a graphical represen- tation of the center of mass motion for two ions in the Penning trap. The plot is created for a fixed cyclotron frequency w in cartesian coordinates (x, y, z). x_0, y_0, $x_0\,d$, and $y_0\,d$ are the initial conditions for integration.

- **PenningI**

PenningI[r0_,z0_,e0_,n_,l_,te_] determines the numerical solution of the equation of motion for the relative components. To integrate the equations of motion, the initial conditions $r_0 = r(t = 0)$, $z_0 = z(t = 0)$, and the total energy e_0 are needed as input parameters. The momentum with respect to the r-direction is set to $\mathrm{pr}_0 = 0$. Parameters l and n determine the shape of the potential. The last argument t_e specifies the endpoint of the integration.

A.2.13 Perihelion Shift

Perihelion shift of a planet.

- **AngularMomentum**

AngularMomentum[minorAxes_,majorAxes_,mass_] calculates the angular momentum of a planet.

- **D0Orbit**

D0Orbit[planet_String,phiend_,options___] plots the orbit in the case of vanishing determinants (see text).

- **Energy**

Energy[minorAxes_,majorAxes_,mass_] calculates the energy of a planet.

- **orbit**

  orbit[phiend_,minorAxes_,majorAxes_,mass_] creates a graphical representation of the perihelion shift if the major and minor axes and the mass are given.

- **Orbit**

  Orbit[planet_String] creates a graphical representation of the perihelion shift for the planets contained in the database.

- **PerihelionShift**

  PerihelionShift[minorAxes_,majorAxes_,mass_] calculates the numeri- cal value of the perihelion shift.

- **Planets**

  Planets[planet_String] creates a list of data for planets and planetoids stored in the database of the package PerihelionShift. The database contains the names of the planets, their major axes, their eccentricities, and the mass of the central planet. **Planets['List']** creates a list of the planets in the data base. **Planets['name']** delivers the data of the planet given in the argument.

A.2.14 Point Charges

Fields of point charges.

- **EnergyDensity**

  EnergyDensity[coordinates_List] calculates the density of the energy for an ensemble of point charges. The cartesian coordinates are lists in the form of {{x,y,z,charge},{...},...}.

- **Field**

Field[coordinates_List] calculates the electric field for an ensemble of point charges. The cartesian coordinates are lists in the form {{x,y,z,charge},{...},...}.

- **FieldPlot**

FieldPlot[coordinates_List,type_,options___] creates a contour plot for an ensemble of point charges. The plot type (Potential, Field, or Density) is specified as a string in the second input variable. The third argument allows a change of the Options of ContourPlot and PlotGradientField.

- **Potential**

Potential[coordinates_List] creates the potential of an assembly of point charges. The cartesian coordinates of the locations of the charges are given in the form of {{x,y,z,charge},{x,y,z,charge},...}.

A.2.15 Poisson Bracket

Canonical Poisson bracket.

- **PoissonBracket**

PoissonBracket[a_, b_, q_List, p_List] calculates the Poisson bracket for two functions a and b which depend on the variables p and q. Example: **PoissonBracket[q,p,{q},{p}]** calculates the fundamental bracket relation between the coordinate and momentum.

A.2.16 Quantum Well

Quantum well in one dimension.

- **PsiASym**

PsiASym[x_,k_,a_] determines the antisymmetric eigenfunction for a potential well of depth $-V_0$. The input parameter k fixes the energy and $2\,a$

the width of the well. **PsiASym** is useful for a numerical representation of eigenfunctions.

- **PsiSym**

PsiSym[x_,k_,a_] determines the symmetric eigenfunction for a potential well of depth $-V_0$. The input parameter k fixes the energy and $2a$ the width of the well. **PsiSym** is useful for a representation of eigenfunctions.

- **Spectrum**

Spectrum[V0_,a_] calculates the negative eigenvalues in a potential well. V_0 is the potential depth and $2a$ the width of the well. The eigenvalues are returned as a list and are available in the variables **lsym** and **lasym** as replacement rules. The corresponding eigenfunctions are stored in the variables **Plsym** and **Plasym**. The determining equation for the eigenvalues is plotted.

A.2.17 Renormalization

Renormalization and percolation.

- **Dim**

Dim[n_] calculates the fractal dimension for the critical probability p_c. The dimension depends on m where $1 \le m \le n-2$, n is the size of the block used.

- **Nc**

Nc[n_] determines the mean number of atoms at the probability p_c if m is changed in the range $1 \le m \le n-2$. The size of the block is determined by n.

- **Pcrit**

Pcrit[n_] determines the critical probability for an $n \times n$ grid under the variation of m where m is the number of empty locations in the grid. The range of m is $1 \leq m \leq n - 2$.

- **RenormPlot**

  RenormPlot[n_,type_String] plots the functions **Nc**, **Dim** or **Pcrit**.

A.2.18 Tree as a Fractal

Fractal tree.

- **Tree**

  Tree[options___] creates a fractal tree. The options of the function Tree determine the form of the fractal created. Options are **Generation → 10**, **BranchRotation → 0.65**, **BranchSkaling → 0.75**, **Branch- Thickness → 0.7**, **OriginalThickness → 0.07**, **BranchColor → {RGBColor[0,0,0]}**.
  Example: **Tree[BranchColor → l1**, **BranchRotation → 0.3]**, **l1** is a list created in the package Tree.

A.3 *Mathematica* Functions

This appendix contains a short description of the *Mathematica* functions used in the book. It is a small selection of the approximately 1200 functions available in the *Mathematica* kernel. The description given does not replace the text of the handbook by S. Wolfram ([1.1]).

The first few items describe the use of the shorthand notation of symbols frequently used in the programming examples. The *Mathematica* functions used in the programs and in the notebooks follow.

- **lhs = rhs** evaluates rhs and assigns the result to lhs. From then on, lhs is replaced by rhs whenever it appears. {l1, l2, ...}= {r1, r2, ...} evaluates the ri and assigns the results to the corresponding li.

- **lhs → rhs** represents a rule that transforms lhs to rhs.
- **expr /. rules** applies a rule or list of rules to transform each subpart of an expression expr.
- **lhs := rhs** assigns rhs to be the delayed value of lhs. rhs is maintained in an unevaluated form. When lhs appears, it is replaced by rhs, evaluated afresh each time.
- **lhs :> rhs** represents a rule that transforms lhs to rhs, evaluating rhs only when the rule is used.
- **lhs == rhs** returns True if lhs and rhs are identical.
- **expr //. rules** repeatedly performs replacements until expr no longer changes.
- **AppendTo[s, elem]** appends elem to the value of s and resets s to the result.
- **Apply[f, expr]** or **f @@ expr** replaces the head of expr by f. Apply[f, expr, levelspec] replaces heads in parts of expr specified by levelspec.
- **ArcSin[z]** gives the arc sine of the complex number z.
- **ArcTan[z]** gives the inverse tangent of z. ArcTan[x, y] gives the inverse tangent of y/x, where x and y are real, taking into account the quadrant in which the point (x, y) is located.
- **Begin["context`"]** resets the current context.
- **BeginPackage["context`"]** makes context` and System` the only active contexts. BeginPackage[ "context` ",{"need1` "}, { "need2` "},...}] calls Needs on the needi.
- **BesselJ[n, z]** gives the Bessel function of the first kind J(n, z).
- **Block[{x, y, ...}, expr]** specifies that expr is to be evaluated with local values for the symbols x, y, ... Block[{x = x0, ...}, expr] defines initial

local values for x,... . Block[{vars}, body /; cond] allows local variables to be shared between conditions and function bodies.

- **C[i]** is the default form for the *i*th constant of integration produced in solving a differential equation with DSolve.

- **Chop[expr]** replaces approximate real numbers in expr that are close to zero by the exact integer 0. Chop[expr, tol] replaces with 0 approximate real numbers in expr that differ from zero by less than tol.

- **Circle[{x, y}, r]** is a two-dimensional graphics primitive that represents a circle of radius r centered at the point {x, y}. Circle[{x, y}, {rx, ry}] yields an ellipse with semiaxes rx and ry. Circle[{x, y}, r, {theta1, theta2}] represents a circular arc.

- **Clear[symbol1, symbol2, ...]** clears values and definitions of the specified symbols. Clear["pattern1", "pattern2", ...] clears values and definitions of all symbols whose names match any of the specified string patterns.

- **Coefficient[expr, form]** gives the coefficient of form in the polynomial expr. Coefficient[expr, form, n] gives the coefficient of form^n in expr.

- **ContourPlot[f, {x, xmin, xmax}, {y, ymin, ymax}]** generates a contour plot of f as a function of x and y.

- **Cos[z]** gives the cosine of z.

- **Cosh[z]** gives the hyperbolic cosine of z.

- **Cot[z]** gives the cotangent of z.

- **D[f, x]** gives the partial derivative of f with respect to x. D[f, {x, n}] gives the nth partial derivative with respect to x. D[f, x1, x2, ...] gives a mixed derivative.

- **f'** represents the derivative of a function f of one argument. Derivative[n1, n2, ...][f] is the general form, representing a function obtained from f by

differentiating n1 times with respect to the first argument, n2 times with respect to the second argument, and so on.

- **Det[m]** gives the determinant of the square matrix m.

- **Disk[{x, y}, r]** is a two-dimensional graphics primitive that represents a filled disk of radius r centered at the point {x, y}. Disk[{x, y}, {r_x, r_y}] yields an elliptical disk with semiaxes r_x and r_x. Disk[{x, y}, r, {θ_1, θ_2}] represents a segment of a disk.

- **Display[channel, graphics]** writes graphics or sound to the specified output channel.

- **Do[expr, {imax}]** evaluates expr imax times. Do[expr, {i, imax}] evaluates expr with the variable i successively taking on the values 1 through imax (in steps of 1). Do[expr, {i, imin, imax}] starts with i = imin. Do[expr, {i, imin, imax, di}] uses steps di. Do[expr,{i, imin, imax}, {j, jmin, jmax},...] evaluates expr looping over different values of j, etc. for each i. Do[] returns Null, or the argument of the first Return it evaluates.

- **DSolve[eqn, y[x], x]** solves a differential equation for the functions y[x], with independent variable x. DSolve[{eqn1, eqn2, ...},{y1[x1], ...}, {x1, ...}] solves a list of differential equations.

- **Dt[f, x]** gives the total derivative of f with respect to x. Dt[f] gives the total differential of f. Dt[f, {x, n}] gives the nth total derivative with respect to x. Dt[f, x1, x2, ...] gives a mixed total derivative.

- **EllipticK[m]** gives the complete elliptic integral of the first kind K(m).

- **End[ ]** returns the present context, and reverts to the previous one.

- **EndPackage[ ]** restores $Context and $ContextPath to their values before the preceding BeginPackage, and prefixes the current context to the list $ContextPath.

- **lhs == rhs** returns True if lhs and rhs are identical.

- **Evaluate[expr]** causes expr to be evaluated, even if it appears as the argument of a function whose attributes specify that it should be held unevaluated.

- **Exp[z]** is the exponential function.

- **Expand[expr]** expands products and positive integer powers in expr. Expand[expr, patt] avoids expanding elements of expr which do not contain terms matching the pattern patt.

- **FindRoot[lhs == rhs, {x,** x_0**}]** searches for a numerical solution to the equation lhs == rhs, starting with x = x_0.

- **Flatten[list]** flattens out nested lists. Flatten[list, n] flattens to level n. Flatten[list, n, h] flattens subexpressions with head h.

- **Floor[x]** gives the greatest integer less than or equal to x.

- **FontForm[expr, {"font", size}]** specifies that expr should be printed in the specified font and size.

- **Function[body]** or body& is a pure function. The formal parameters are # (or #1), #2, etc. Function[x, body] is a pure function with a single formal parameter x. Function[{x1, x2,...}, body] is a pure function with a list of formal parameters. Function[{x1, x2, ...}, body, {attributes}] has the given attributes during evaluation.

- **<<name** reads in a file, evaluating each expression in it, and returning the last one. Get["name ", key] gets a file that has been encoded with a certain key.

- **Graphics[primitives, options]** represents a two-dimensional graphical image.

- **GraphicsArray[{g1, g2, ...}]** represents a row of graphics objects. GraphicsArray[{{g11, g12, ...}, ...}] represents a two-dimensional array of graphics objects.

- **HermiteH[n, x]** gives the nth Hermite polynomial.
- **Hold[expr]** maintains expr in an unevaluated form.
- **Hue[h]** specifies that graphical objects which follow are to be displayed, if possible, in a color corresponding to hue h. Hue[h, s, b] specifies colors in terms of hue, saturation, and brightness.
- **If[condition, t, f]** gives t if condition evaluates to True, and f if it evaluates to False. If[condition, t, f, u] gives u if condition evaluates to neither True nor False.
- **Im[z]** gives the imaginary part of the complex number z.
- **Infinity** is a symbol that represents a positive infinite quantity.
- **Input[ ]** interactively reads in one *Mathematica* expression. Input["prompt"] requests input, using the specified string as a prompt.
- **Integrate[f,x]** gives the indefinite integral of f with respect to x. Integrate[f,{x, xmin,xmax}] gives the definite integral. Integrate[f,{x,xmin,xmax},{y,ymin,ymax}] gives a multiple integral.
- **InterpolatingFunction[range, table]** represents an approximate function whose values are found by interpolation.
- **JacobiAmplitude[u, m]** gives the amplitude for Jacobi elliptic functions.
- **JacobiSN[u, m]** gives the Jacobi elliptic function sn at u for the parameter m.
- **Join[list1, list2,...]** concatenates lists together. Join can be used on any set of expressions that have the same head.
- **LaguerreL[n, x]** gives the *n*th Laguerre polynomial. LaguerreL[n, a, x] gives the *n*th generalized Laguerre polynomial.
- **LegendreP[n, x]** gives the *n*th Legendre polynomial. LegendreP[n, m, x] gives the associated Legendre polynomial.

- **Length[expr]** gives the number of elements in expr.
- **Limit[expr, $x \to x_0$]** finds the limiting value of expr when x approaches x_0.
- **Line[{pt1, pt2,...}]** is a graphics primitive which represents a line joining a sequence of points.
- **{e1, e2, ...}** is a list of elements.
- **ListPlot[{y1, y2, ...}]** plots a list of values. The x coordinates for each point are taken to be 1, 2, ... ListPlot[{{x1, y1}, {x2, y2}, ...}] plots a list of values with specified x and y coordinates.
- **Log[z]** gives the natural logarithm of z (logarithm to base E). Log[b, z] gives the logarithm to base b.
- **Map[f, expr]** or **f /@** expr applies f to each element on the first level in expr. Map[f, expr, levelspec] applies f to parts of expr specified by levelspec.
- **MapAt[f, expr, n]** applies f to the element at position n in expr. If n is negative, the position is counted from the end. MapAt[f, expr, {i, j, ...}] applies f to the part of expr at position {i, j, ...}. MapAt[f, expr, {{i1, j1,...}, {i2, j2, ...}, ...}] applies f to parts of expr at several positions.
- **MatrixForm[list]** prints the elements of list arranged in a regular array.
- **Max[x1, x2, ...]** yields the numerically largest of the xi. Max[{x1, x2, ...}, {y1, ...}, ...] yields the largest element of any of the lists.
- **Min[x1, x2, ...]** yields the numerically smallest of the xi. Min[{x1, x2, ...}, {y1,...},...] yields the smallest element of any of the lists.
- **Mod[m, n]** gives the remainder on division of m by n. The result has the same sign as n.
- **N[expr]** gives the numerical value of expr. N[expr, n] does computations to n-digit precision.

- **NDSolve[eqns, y, {x, xmin, xmax}]** finds a numerical solution to the differential equations eqns for the function y with the independent variable x in the range xmin to xmax. NDSolve[eqns, {y1, y2,...}, {x, xmin, xmax}] finds numerical solutions for the functions yi. NDSolve[eqns, y, {x, x1, x2, ...}] forces a function evaluation at each of x1, x2, ... The range of numerical integration is from Min[x1, x2, ...] to Max[x1, x2,...].

- **Needs["context` ", "file"]** loads file if the specified context is not already in $Packages. Needs["context`"] loads the file specified by ContextToFilename["context`"] if the specified context is not already in $Packages.

- **Nest[f, expr, n]** gives an expression with f applied n times to expr.

- **NestList[f, expr, n]** lists the results of applying f to expr 0 through n times.

- **NIntegrate[f, {x, xmin, xmax}]** gives a numerical approximation to the integral of f with respect to x over the interval xmin to xmax.

- **Normal[expr]** converts expr to a normal expression, from a variety of special forms.

- **NSolve[eqns, vars]** attempts to solve numerically an equation or set of equations for the variables vars. Any variable in eqns but not vars is regarded as a parameter. NSolve[eqns] treats all variables encountered as vars above. NSolve[eqns, vars, prec] attempts to solve numerically the equations for vars using prec digits precision.

- **Off[symbol::tag]** switches off a message, so that it is no longer printed. Off[s] switches off tracing messages associated with the symbols. Off[m1, m2, ...] switches off several messages. Off[ ] switches off all tracing messages.

- **On[symbol::tag]** switches on a message, so that it can be printed. On[s] switches on tracing for the symbol s. On[m1, m2, ...] switches on several messages ma, m2, ... On[ ] switches on tracing for all symbols.

- **ParametricPlot[{fx, fy}, {t, tmin, tmax}]** produces a parametric plot with x and y coordinates fx and fy generated as a function of t.

ParametricPlot[{{fx, fy}, {gx, gy}, ...}, {t, tmin, tmax}] plots several parametric curves.

- **ParametricPlot3D[{fx, fy, fz}, {t, tmin, tmax}]** produces a three-dimensional space curve parameterized by a variable t which runs from tmin to tmax. ParametricPlot3D[{fx, fy, fz}, {t, tmin, tmax}, {u, umin, umax}] produces a three-dimensional surface parameterized by t and u. ParametricPlot3D[{fx, fy, fz, s}, ...] shades the plot according to the color specifications. ParametricPlot3D[{{fx, fy, fz}, {gx, gy, gz}, ...}, ...] plots several objects together.

- **expr[[i]]** or **Part[expr, i]** gives the ith part of expr. expr[[-i]] counts from the end. expr[[0]] gives the head of expr. expr[[i, j, ...]] or Part[expr, i, j, ...] is equivalent to expr[[i]][[j]] ... . expr[[{i1, i2, ...}]] gives a list of the parts i1, i2, ... of expr.

- **Partition[list, n]** partitions list into non-overlapping sublists of length n. Partition[list, n, d] generates sublists with offset d. Partition[list, {n1, n2, ...}, {d1, d2, ...}] partitions successive levels in list into length ni sublists with offsets di.

- **Pi** is pi, with numerical value 3.14159... .

- **Plot[f, {x, xmin, xmax}]** generates a plot of f as a function of x from xmin to xmax. Plot[{f1, f2, ...}, {x, xmin, xmax}] plots several functions fi.

- **x + y + z** represents a sum of terms.

- **Point[coords]** is a graphics primitive that represents a point.

- x^y gives x to the power y.

- **PowerExpand[expr]** expands nested powers, powers of products, logarithms of powers, and logarithms of products. PowerExpand[expr,{x1, x2,...}] expands expr with respect to the x1. Use PowerExpand with caution because PowerExpand does not pay attention to branch cuts.

- **Print[expr1, expr2,...]** prints the expri, followed by a newline (line feed).

- **Protect[s1, s2, ...]** sets the attribute Protected for the symbols si. Protect["form1", "form2 ", ...] protects all symbols whose names match any of the string patterns formi.

- **Quit[ ]** terminates a *Mathematica* session.

- **Random[ ]** gives a uniformly distributed pseudorandom Real in the range 0 to 1. Random[type, range] gives a pseudorandom number of the specified type, lying in the specified range. Possible types are Integer, Real, and Complex. The default range is 0 to 1. You can give the range {min, max} explicitly; a range specification of max is equivalent to {0, max}.

- **Re[z]** gives the real part of the complex number z.

- **ReleaseHold[expr]** removes Hold and HoldForm in expr.

- **Replace[expr, rules]** applies a rule or list of rules in an attempt to transform the entire expression expr.

- **expr /. rules** applies a rule or list of rules in an attempt to transform each subpart of an expression expr.

- **expr //. rules** repeatedly performs replacements until expr no longer changes.

- **RGBColor[red, green, blue]** specifies that graphical objects which follow are to be displayed, if possible, in the color given.

- **lhs →rhs** represents a rule that transforms lhs to rhs.

- **Save["filename", symb1, symb2, ...]** appends the definitions of the symbols symbi to a file.

- **Series[f, {x, x_0, n}]** generates a power series expansion for f about the point $x = x_0$ to order $(x - x_0)^n$. Series[f, {x, x_0, nx}, {y, y_0, ny}] successively finds series expansions with respect to y, then x.

- **Show[graphics, options]** displays two- and three-dimensional graphics, using the options specified. Show[g1, g2, ...] shows several plots combined. Show can also be used to play Sound objects.

- **Simplify[expr]** performs a sequence of transformations on expr and returns the simplest form it finds.

- **Sin[z]** gives the sine of z.

- **Sinh[z]** gives the hyperbolic sine of z.

- **Solve[eqns, vars]** attempts to solve an equation or set of equations for the variables vars. Any variable in eqns but not vars is regarded as a parameter. Solve[eqns] treats all variables encountered as vars above. Solve[eqns, vars, elims] attempts to solve the equations for vars, eliminating the variables elims.

- **Sort[list]** sorts the elements of list into canonical order. Sort[list, p] sorts using the ordering function p.

- **SphericalHarmonicY[l, m, theta, phi]** gives the spherical harmonic $Y_{l,m}(\theta, \phi)$.

- **Sqrt[z]** gives the square root of z.

- **Sum[f, {i, imax}]** evaluates the sum of f with i running from 1 to imax. Sum[f, {i, imin, imax}] starts with i = imin. Sum[f, {i, imin, imax, di}] uses steps di. Sum[f, {i, imin, imax}, {j, jmin, jmax},...] evaluates a multiple sum.

- **Table[expr, {imax}]** generates a list of imax copies of expr. Table[expr, {i, imax}] generates a list of the values of expr when i runs from 1 to imax. Table[expr, {i, imin, imax}] starts with i = imin. Table[expr, {i, imin, imax, di}] uses steps di. Table[expr, {i, imin, imax}, {j, jmin, jmax},...] gives a nested list. The list associated with i is outermost.

- **Take[list, n]** gives the first n elements of list. Take[list, -n] gives the last n elements of list. Take[list, {m, n}] gives elements m through n of list.

- **Tan[z]** gives the tangent of z.

- **Text[expr, coords]** is a graphics primitive that represents text corresponding to the printed form of expr, centered at the point specified by coords.

- **Thread[f[args]]** ``threads" f over any lists that appear in args. Thread[f[args], h] threads f over any objects with head h that appear in args. Thread[f[args], h, n] threads f over objects with head h that appear in the first n args. Thread[f[args], h, -n] threads over the last n args. Thread[f[args], h, {m, n}] threads over arguments m through n.

- **Unprotect[s1, s2, ...]** removes the attribute Protected for the symbols si. Unprotect["form1","form2", ...] unprotects all symbols whose names textually match any of the formi.

- **Which[test1, value1, test2, value2, ...]** evaluates each of the testi in turn, returning the value of the valuei corresponding to the first one that yields True.

References

Volume I

[1] **Chapter 1**

[1.1] S. Wolfram, The *Mathematica* book, 5th ed. Wolfram Media/Cambridge University Press, Cambridge 2003.

[1.2] M. Abramowitz & I.A. Stegun, Handbook of Mathematical Functions. Dover Publications, Inc., New York, 1968.

[1.3] N. Blachman, *Mathematica*: A Practical Approach. Prentice Hall, Englewood Cliffs, 1992.

[1.4] Ph. Boyland, A. Chandra, J. Keiper, E. Martin, J. Novak, M. Petkovsek, S. Skiena, I. Vardi, A. Wenzlow, T. Wickham-Jones, D. Withoff, and others, Technical Report: Guide to Standard *Mathematica* Packages, Wolfram Research, Inc. 1993.

[2] **Chapter 2**

[2.1] R. Maeder, Programming in *Mathematica*. Addison-Wesley Publ. Comp. Inc., Redwood City, 1991.

[2.2] L.D. Landau & E.M. Lifshitz, Mechanics. Addison-Wesley, Reading, Massachusetts, 1960.

[2.3] J. B. Marion, Classical Dynamics of Particles and Systems. Academic Press, New York, 1970.

[2.4] R. Courant & D. Hilbert, Methods of Mathematical Physics, Vol. 1+2. Wiley (Interscience), New York, 1953.

[2.5] R.H. Dicke, Science 124, 621, (1959)

[2.6] R.V. Eötvös,Ann.Phys. 59, 354, (1896)

[2.7] L. Southerns,Proc.Roy.Soc.(London),A, 84, 325, (1910)

[2.8] P. Zeeman,Proc.Amst.,20,542,(1917)

[2.9] G. Baumann, Symmetry Analysis of Differential equations using *Mathematica,* Springer, New York, (2000).

[2.10] H. Geiger and E. Marsden, The Laws of Deflexion of α Particles through Large Angles, Phil. Mag. 25, 605, 1913.

[2.11] Ph. Blanchard and E. Brüning, Variational Methods in Mathematical Physics, Springer, Wien, 1982.

[3] **Chapter 3**

[3.1] F. Calogero & A. Degasperis, Spectral Transform and Solitons: Tools to solve and investigate nonlinear evolution equations. North-Holland Publ. Comp., Amsterdam, 1982.

[3.2] V.A. Marchenko, On the Reconstruction of the Potential Energy from Phases of the Scattered Waves. Doklady Akademii Nauk SSSR, **104**, 695, 1955.

R.M. Miura, C. Gardner & M.D. Kruskal. Korteweg-de Vries equation and generalizations. II Existence of Conservation Laws and Constants of Motion. Journal of Mathematical Physics **9**, 1204, 1968.

[3.4] T.R. Taha & M.J. Ablowitz, Analytical and numerical solutions of certain nonlinear evolution equations. I. Analytical. Journal of Computational Physics **55**, 192, 1984.

[3.5] N.J. Zabusky & M.D. Kruskal, Interactions of 'solitons' in a collisionless plasma and the recurrence of initial states. Physical Review Letters **15**, 240, 1965.

Volume II

[4] **Chapter 4**

[4.1] G. Arfken, Mathematical Methods for Physicists. Academic Press, New York, 1966.

[4.2] P.M. Morse & H. Feshbach, Methods of Theoretical Physics. McGraw-Hill, New York, 1953.

W. Paul, O. Osberghaus & E. Fischer, Ein Ionenkäfig. Forschungsbericht des Wissenschafts- und Verkehrsministeriums Nordrhein-Westfalen, **415**, 1, 1958.

[4.3] Similar work has been done by H. G. Dehmelt, Radiofrequency Spectroscopy of Stored Ions I: Storage, Advances in Atomic and Molecular Physics 3(1967) 53-72; D. J. Wineland, W.M. Itano and R.S. van Dyck Jr., High-Resolution Spectroscopy of Stored Ions, Advances in Atomic and Molecular Physics 19(1983)135-186

F.M. Penning, Die Glimmentladung bei niedrigem Druck zwischen koaxialen Zylindern in einem axialen Magnetfeld. Physica **3**, 873,

1936.
Similar work has been done by D. Wineland, P. Ekstrom and H. Dehmelt, Monoelectron Oscillator, Physical Review Letters 31(1973)1279-1282

[4.5] G. Baumann, The Paul trap: a completely integrable model? Phys. Lett. **A 162**, 464, 1992.

[5] **Chapter 5**

[5.1] E. Schrödinger, Quantisierung als Eigenwertproblem. Annalen der Physik, **79**, 361, 1926.

[5.2] N. Rosen & P.M. Morse, On the Vibrations of Polyatomic Molecules. Physical Review **42**, 210, 1932.

[5.3] G. Pöschel & E. Teller, Bemerkungen zur Quantenmechanik des anharmonischen Oszillators. Z. Physik, **83**, 143, 1933.

[5.4] W. Lotmar, Zur Darstellung des Potentialverlaufs bei zweiatomigen Molekülen. Z. Physik, **93**, 518, 1935

[5.5] S. Flügge, Practical Quantum Mechanics I + II. Springer-Verlag, Berlin, 1971.

[5.6] C. Cohen-Tannoudji, B. Diu & F. Laloë, Quantum Mechanics I + II. John Wiley & Sons, New York, 1977.

[5.7] Rowlinson J.S.; *Mol. Phys.* 1963, 6, 75-83

[5.8] Lennard-Jones J.E.; *Proc. Roy. Soc.* 1924, A106, 463-477

[5.9] London F.; *Z. Phys.* 1930, 63, 245-279

[5.10] Hirschfelder J.O., Curtiss R.F., Bird R.B. Molecular Theory of Gases and Liquids. Wiley: New York, 1954

[5.11] Mason E.A., Spurling T.H. The virial Equation of State; Pergamon Press, Oxford, 1969

[5.12] McQuarrie D.A.; *Statistical Thermodynamics*, Harper and Row: New York 1973, p. 307

[5.13] Sinanoglu O. and Pitzer K.S.; *J. Chem. Phys.* 1959, 31, 960-967

[5.14] Friend D.G.; *J. Chem. Phys.* 1985, 82, 967-971

[5.15] Kihara T.; *Suppl. Progs. Theor. Phys.* 1967, 40, 177-206

[5.16] Stogryn D.E., Hirschfelder J.O. J. Chem. Phys. 1959, 31, 1531-1545

[5.17] Phair R., Biolsi L., Holland P.M. Int. J. Thermophys., 1990, 11, 201-211

[5.18] Mies F.H., Julienne P.S. J. Chem. Phys. 1982, 77, 6162-61176

[6] **Chapter 6**

[6.1] W. Rindler, Essential Relativity. Springer Verlag, New York, 1977.

[6.2] C.W. Misner, K.S. Thorne & J.A. Wheeler, Gravitation. Freeman, San Francisco, 1973.

[6.3] H. Stephani, General relativity: An introduction to the gravitational field. Cambridge University Press, 1982.

[6.4] M. Berry, Principles of Cosmology and Gravitation. Cambridge University Press, Cambridge, 1976.

[7] **Chapter 7**

[7.1] T.W. Gray & J. Glynn, Exploring Mathematics with *Mathematica*. Addison-Wesley Publ. Comp. Inc., Redwood City, 1991.

[7.2] T.F. Nonnenmacher, G. Baumann & G. Losa, Self organization and fractal scaling patterns in biological systems. In: Trends in Biological Cybernetics, World Scientific, Singapore, **1**, 65, 1990.

[7.3] A. Barth, G. Baumann & T.F. Nonnenmacher, Measuring Rényi-dimensions by a modified box algorithm. Journal of Physics A: Mathematical and General **25**, 381, 1992.

[7.4] B. Mandelbrot, The fractal geometry of nature. W.H. Freeman a. Comp., New York, 1983.

[7.5] A. Aharony, Percolation. In: Directions in condensed matter physics (Eds. G. Grinstein & G. Mazenko). World Scientific, Singapore, 1986.

[7.6] T. Grossman & A. Aharony, Structure and perimeters of percolation clusters. Journal of Physics A: Mathematical and General **19**, L745, 1986.

[7.7] P.G. Gennes, Percolation - a new unifying concept. La Recherche **7**, 919, 1980.

[7.8] S.F. Lacroix, Traité du Calcul Différentiel et du Calcul Intégral, 2nd ed., Vol.3 pp. 409-410. Courcier, Paris (1819).

[7.9] L. Euler, De progressionibvs transcendentibvs, sev qvarvm termini generales algebraice dari negvevnt, In: Comment Acad. Sci. Imperialis petropolitanae, **5**, 36-57, (1738).

[7.10] K.B. Oldham and J. Spanier, The Fractional Calculus, Academic Press, New York, (1974).

[7.11] K.S. Miller and B. Ross, An Introduction to the Fractional Calculus and Fractional Differential Equations, John Wiley & Sons, Inc., New York, (1993).

[7.12] G.F.B. Riemann, Gesammelte Werke, pp.353-366, Teubner, Leipzig, (1892).

[7.13] J. Liouville, Mémoiresur le calcul des différentielles à indices quelconques, J. École Polytech., **13**, 71-162, (1832).

[7.14] H. Weyl, Bemerkungen zum Begriff des Differentialquotienten gebrochener Ordnung, Vierteljahresschr. Naturforsch. Ges. Zürich, **62**, 296-302, (1917).

[7.15] H.T. Davis, The Theory of Linear Operators, Principia Press, Bloomington, Ind., (1936).

[7.16] B. Riemann, Über die Anzahl der Primzahlen unter einer gegebenen Größe, Gesammelte Math. Werke, 136-144, (1876).

[7.17] E. Cahen, Sur la fonction ζ(s) de Riemann et sur des Fonctions analoges, Ann de l'Ec. Norm, **11**, 75-164, (1894).

[7.18] H. Mellin, Über die fundamentale Wichtigkeit des Satzes von Cauchy für die Theorie der Gamma- und der hypergeometrischen Funktion, Acta Soc. Fennicae. **21**, 1-115, (1896).

[7.19] H. Mellin, Über den Zusammenhang zwischen den linearen Differential- und Differenzengleichungen, Acta Math. **25**, 139-164, (1902).

[7.20] F. Oberhettinger, Mellin Transforms, Springer, Berlin, (1974)

[7.21] G. Baumann, Symmetry Analysis of Differential equations using *Mathematica,* Springer, New York, (2000).

[7.22] J.B. Bates and Y.T. Chu, Surface Topography and Electrical Response of Metal-Electrolyte Interfaces, Solid State Ionics, **28-30**, 1388-1395, (1988).

[7.23] H. Scher and E.W. Montroll, Anomalous Transit-Time Dispersion in Amorphous Solids, Phys. Rev. B, **12**, 2455-2477, (1975).

[7.24] K.S. Cole and R.H. Cole, Dispersion and Absorption in Dielectrics, J. Chem. Phys., 9, 341-351, (1941).

[7.25] W.G. Glöckle, Anwendungen des fraktalen Differentialkalküls auf Relaxationen, Thesis, Ulm, (1993).

[7.26] R. Metzler, Modellierung spezieller dynamischer Probleme in komplexen Materialien, Thesis, Ulm, (1996).

[7.27] H. Schiessel and A. Blumen, Mesoscopic Pictures of the Sol-Gel Transition: Ladder Models and Fractal Networks, Macromolecules, 28, 4013-4019, (1995).

[7.28] T.F. Nonnenmacher, On the Riemann-Liouville Fractional Calculus and some Recent Applications, Fractals, 3, 557-566, (1995).

[7.29] B.J. West and W. Deering, Fractal physiology for physicists: Lévy statistics, Phys. Rep. 246, 1-100, (1994).

[7.30] W. Wyss, The Fractional Diffusion Equation, J. Math. Phys., **27**, 2782-2785, (1986).

[7.31] B. O'Shaugnessy and I. Procaccia, Analytical Solutions for Diffusion on Fractal Objects, Phys. Rev. Lett., **54**, 455-458, (1985).

[7.32] W.R. Schneider and W. Wyss, Fractional Diffusion and Wave Equations, J. Math. Phys. **30**, 134-144, (1989).

[7.33] R. Metzler, W.G. Glöckle, and T:F. Nonnenmacher, Fractional Model Equation for Anomalous Diffusion, Physica, **211A**, 13-24, (1994).

[7.34] A. Compte, Stochastic foundations of fractional dynamics, Phys. Rev. E, **53**, 4191-4193, (1996)

[7.35] B.J. West, P. Grigolini, R. Metzler, and T.F. Nonnenmacher, Fractional diffusion and Lévy stable processes,Phys. Rev. E, **55**, 99-106, (1997).

Index