Computational Methods in Physics

Pawel Scharoch

March 19, 2017

Wroclaw University of Technology, Wyb. Wyspianskiego 27, 50-370 Wroclaw, Poland

Contents

1 Introduction.

Numerical modeling is a powerful scientific tool. Its spectacular development could be observed over the last half century, as one of the consequences of advances in computer technology. Today, no field of science or engineering can be found were the computational methods would not play one of crucial roles. The list of their advantages is long. New opportunities have emerged, like: the ability of "predicting", of investigating into properties unavailable experimentally, obtaining quantitative data requiring huge amount of mathematical operations or processing of huge amount of data. Those new opportunities are accompanied by the relative easiness of application and low costs. Nowadays, the computational research, owing to the mentioned above features, constitutes an independent scientific tool (besides traditional experiment and theory), which on the one hand, is an extension of mathematical modeling and could not exist without it, but on the other hand, creates its own methodology and often resembles experimental rather than theoretical work (computational experiments).

In view of the above remarks, it is obvious that various aspects of numerical modeling should become an important element of academic education, particularly at science and engineering faculties. In fact, such an element of education has been offered for many years by universities all over the world, even in the form of a separate specialization (like e.g. Applied Computer Science), which concentrates on various aspects of applications of computers in science and technology, and on numerical modeling in particular. This course (Computational Methods in Physics) is a response to this need. This is a "Numerical Modeling" course, whose philosophy is similar to that proposed by S.Koonin [1], but has been modified, adjusted to more elementary level, and supplemented with additional issues and examples. The idea of the course is that it is purpose-oriented, i.e. a numerical method is introduced as a tool to solve a particular physical problem, so that a participant has the feeling of applicability of the method he or she learns. An additional advantage is learning physics, since the computer programs on which the participants work allow them to investigate into properties of chosen systems. The course is constructed as a series of Projects, which are here to provide a good reason and motivation for learning numerical methods. Thus, every Subject Unit (Project) begins with presenting the physics problem, discussing it and identifying the mathematical tools needed. Then, an appropriate numerical method is introduced, the algorithm is developed and a respective computer code is shown and discussed.

The methods taught cover basic mathematical operations (finding roots and minima), derivatives and integrals, ordinary and partial differential equations (initial value problem (IVP), boundary value problem (BVP), eigenvalue problem (EVP)). Most methods are based on discretisations of variables. The physical systems are chosen so that their simulation could be relatively simple (the programming, tests and experiments could be done during 1-2 sessions), thus the methods specific for the partial differential equations are illustrated by 1D

systems. The main objective of the course is to learn numerical methods, not a particular programming language, thus the algorithms can be easily implemented in various programming environments (FORTRAN, C, PYTHON, etc.). The students are given preliminary versions of the codes in FORTRAN77 and PYTHON, to be used alternatively during practical classes. During computer laboratory classes students have an opportunity to test and modify the codes, and finally perform some experiments which resemble the real ones, but offer much more freedom in setting the conditions, and therefore open large space for creativity and learning physics. For the freeware FORTRAN language the *Force* environment is used and for the graphical representation of data the freeware GNUPLOT is recommended, alternatively the PYTHON freeware compiler can be used. The advantage of PYTHON is that has its own graphical environment and thus no any additional graphical tool is needed. Using the freeware software allows students to work on the lab projects also at home. Getting familiar with this software can be very useful for them also in the future.

This material is organized in the following way. Each chapter begins with a brief summary of its content followed by the theoretical background section, containing a summary of the theoretical foundation of the considered problem, and the numerical procedures section, introducing the respective numerical algorithms and presenting the final formulae of the schemes to be applied in the programs. No derivations in those chapters are presented, only some hints, how a particular formula can be derived, are given, thus further reading is strongly recommended of any standard physics textbook (at academic level) as well as on numerical methods, e.g. [1, 2, 3] (at the beginning of each section the ***preliminary reading*** note points at the physics part which should be recalled or learned in advance). It is also recommended that a student derives by himself every formula presented in this book (this should be treated as an **obligatory exercise** for each session which has not been explicitly mentioned in the text). Finally a section **Exercises** follows in which students will find a set of exercises to be done with the particular computer code. The exercises are divided into 3 categories: *Obligatory* (which have to be done), *Supplementary* (which serve for training) and *Challenge* (for students particularly interested in the subject). The computer codes have been prepared and tested in advance and are provided as an integral part of this book. The codes have not been optimized so that the implementation of the algorithms is clearer. Students are free to modify the codes when doing the exercises or to implement their own ideas.

2 First steps

Prerequisities: *basic computer knowledge (Windows or Linux operating system).*

This is the first computer lab session during which students learn (or recall) basic elements of FORTRAN or PYTHON programming language (program structure, declaration of variables, input/output declarations and instructions), and (if needed) graphical program GNUPLOT. Basic numerical procedures are also introduced: tabulating 1D and 2D functions, finding roots and finding a minimum in 1D.

2.1 Basic mathematical operations

Content: *tabulating a function 1D i 2D, finding roots (Bisection, Secant, Newton-Raphson method), finding a minimum in 1D (Golden Section Search for a minimum in 1D, method based on the parabolic interpolation, Simplex method in 1D).*

Finding roots, Bisection method

We start from the point where we are sure that the root of a function $f(x)$ is somewhere within the interval (x_l, x_r). To reduce the interval we cut it half and find the point in the middle x_m. We get then two new intervals and we have to identify the one within which the function crosses zero. This can be easily done by checking the condition e.g. $f(x_l) \cdot f(x_m) < 0$ (the function changes its sign). We denote the ends of the new interval again by (x_l, x_r) and repeat the procedure. After n steps the lengths of the domain segment containing zero reduces by factor 2^n and when it is less then assumed accuracy ϵ the whole procedure stops.

The bisection method is very safe but not most efficient. The popular alternatives are Secant and Newton-Raphson methods. The Newton-Raphson method is useful when together with the function $f(x)$ also its analytical derivative is known. Using the derivative we construct a linear function being tangent to $f(x)$ at the starting point. Then we approximate the root of $f(x)$ by the zero of the linear function. This root is treated as the new starting point and the construction repeats. In many cases the procedure is very quickly convergent but this is not always the case, thus should be applied with a caution. The secant method is similar to the Newton-Raphson one except the derivative of a function is found numerically, using e.g. a 3-point scheme (see Chapter 4.2)

Finding minimum in 1D, Golden Section Search

To identify the domain interval containing a minimum we need a checking point x_c within the interval (x_l, x_r). We can be sure that the minimum is present in the interval if $f(x_c) < f(x_l)$ and $f(x_c) < f(x_r)$. In constructing the algorithm one should focus on a proper choice of the new, fourth checking point x_n. By introducing the fourth point in the interval (x_l, x_r) we obtain two overlapping regions each defined by 3 points. We can then identify the one containing the minimum (the condition above). To assure the optimum

convergence the two regions should have the smallest possible and the same length (at each step). This can be achieved if at each step the fourth point splits the longer of the two segments (x_l, x_c) and (x_c, x_r) in golden section proportion, i.e. $h_l/(h_l + h_s) = h_s/h_l$ (h_l - longer segment, h_s - shorter segment) so that the two new intervals containing 3 point have equal length. By substituting $x = h_s/h_l$ into the above golden section condition we get the equation $x^2 + x - 1 = 0$ whose one of solutions $(\sqrt{5} - 1)/2 \approx 0.62$ is the golden section ratio. Thus, after n steps the length of starting interval is reduced by factor $\approx 0.62^n$, which is a little bit slower then in bisection algorithm.

The Golden Section Search is always convergent but not necessarily most effective. Usually, very effective is the algorithm in which the forth checking point is found as the minimum of the second order polynomial interpolation of the function between 3 initial points. The effectiveness is connected with the fact that many functions near minimum have leading second order term in the power series expansion.

The Simplex method is the most simple search of the minimum which uses a "test window" rather than 3 points. Starting from certain point the domain is scanned in the direction where function decreases (let us say to the right) by moving window (x_l, x_r) of certain length $h = x_r - x_l$. The minimum is found if the function begins to increase, i.e. $f(x_l) < f(x_r)$. At this point the window length is cut half and the search continues from the x_l where the increase has been noticed. The procedure stops if $h < \epsilon$, ϵ - the assumed accuracy.

If the analytical form of the first derivative is known then the minimum can be found by finding a root of the derivative. The derivative can be also approximated by a numerical formula, this leads to more general form of the described above secant scheme.

2.2 Exercises

Obligatory

1. Visualize chosen 1D function (set your own function).

2. Modify the program FTABLE so that it could tabulate a 2D function, set your own function and visualise it.

3. Test the program BISEC by finding the roots of a chosen 2nd order polynomial and comparing the results with the analytical solutions.

4. Find the value of the number π as a zero of sinus function. What precision can you achieve, explain why ?

Challenge

5. Write the program 1DMINIMUM that finds the minimum of a 1D function with one of algorithms discussed on the lecture (golden section, parabolas or 1D simplex). Test the

program by finding the minimum of a chosen 2nd order polynomial and comparing the results with the analytical solution. Find the value of the number π as a minimum position of cosine function. What precision can you achieve, explain why ?

3 Project: Rectangular quantum well

Preliminary reading: *stationary Schrödinger equation, rectangular quantum well as the simplest model of a low dimensional or high symmetry quantum system.*

In this project the participants use the finding roots procedure to solve the eigenvalue problem of a rectangular quantum well (program QWELL). The program is then applied to fit the simple quantum well results to a real system (hydrogen atom) by trial and error method.

3.1 Stationary SchrĂ¶dinger equation in 1D

Content: *eigenvalue problem for a rectangular quantum well and a method of its numerical solution.*

Stationary Schrödinger equation for a rectangular quantum well:

$$\Big[-\frac{\hbar^2}{2m}\frac{d^2}{dx^2} + V(x)\Big]\psi(x) = \varepsilon\psi(x) \tag{1}$$

where:

$$V(x) = \begin{cases} -V_o & \text{if } -a/2 \le x \le a/2 \\ 0 & \text{if } x < -a/2 \text{ or } x > a/2 \end{cases}$$

In Hartree atomic units: $\hbar = m_e = e = 1$:

$$\Big[\frac{d^2}{dx^2} + k^2(x)\Big]\psi(x) = 0 \tag{2}$$

where $k^2(x) = 2(\varepsilon - V(x))$.
Solutions:

$$\psi(x) = \begin{cases} A\cos(kx) & \text{for } -a/2 \le x \le a/2 \quad (even) \\ A\sin(kx) & \text{for } -a/2 \le x \le a/2 \quad (odd) \\ B\exp(\mp ikx) & \text{for } x < -a/2 \text{ or } x > a/2 \end{cases} \tag{3}$$

The condition for the eigenvalue is that the two solutions (inside and outside the region of the well) must join smoothly, i.e. their values and the values of their first derivatives must be equal e.g. at $a/2$ (because of symmetry it is sufficient to consider only one border). Thus, we have:

for even solutions:

$$\begin{cases} \pm A\cos(ka/2) = \pm B\exp(-\kappa a/2) \\ \mp Ak\sin(ka/2) = \mp B\kappa\exp(-\kappa a/2) \end{cases}$$

and for odd solutions:

$$\begin{cases} \pm A\sin(ka/2) = \pm B\exp(-\kappa a/2) \\ \pm Ak\cos(ka/2) = \mp B\kappa\exp(-\kappa a/2) \end{cases}$$

where $k = \sqrt{2(\varepsilon + V_o)}$ and $\kappa = \sqrt{-2\varepsilon}$

Dividing the first equation by the second one in the above systems we obtain two conditions, for even and odd solution:

$$\begin{cases} F_{even}(\varepsilon) = sin(ka/2) - \kappa/k \cdot cos(ka/2) = 0 \quad (even) \\ F_{odd}(\varepsilon) = sin(ka/2) + k/\kappa \cdot cos(ka/2) = 0 \quad (odd) \end{cases} \tag{4}$$

The eigenvalues ε are found by solving the above equations.

3.2 Exercises

Obligatory

1. Using the program QWELL tabulate functions $F_{even}(\varepsilon)$ and $F_{odd}(\varepsilon)$ (they are the functions, corresponding to even and odd solutions respectively, whose zeros are energies of quantum levels)

2. Visualize (in one figure) the functions $F_{even}(\varepsilon)$ i $F_{odd}(\varepsilon)$. Repeat the calculation and visualization of $F_{even}(\varepsilon)$ i $F_{odd}(\varepsilon)$ for different values of the well parameters (a and V_0).

3.(Square finite quantum well as a model of the hydrogen atom). Try to fit the first two energy levels to the hydrogen ones through variation of the parameters a and Vo, by trial and error method (hint: in the beginning set the values $a = 3Bohr$ and $V_o = 1Hartree$). What is the value of the third energy level? (Note that in atomic units the energy levels should be $\varepsilon_n = -1/(2n^2)$.)

4 Project: Diffraction of light by a slit

Preliminary reading: *interference and diffraction phenomena.*

By working on this project, the participants learn the numerical differentiation and quadrature procedures. In particular, the important issue of convergence with respect to the grid parameter is discussed. The numerical quadrature procedure is used to construct a program (DIFFRACTION) simulating the diffraction of a scalar wave by a single infinite slit and a system of parallel infinite slits. The program then serves for studying the physics of the system.

4.1 Interference and diffraction

Content: *interference and diffraction phenomena, the complex amplitude concept, discrete and continuous sources, diffraction integral, single infinite slit diffraction, diffraction grating.*

The diffraction integral represents a superposition of complex amplitudes of waves emitted by elementary sources forming the whole continuous source (Huygens principle):

$$D = \int_{Source} \frac{A_o(s)}{r} \exp\left(-ikr + \phi(s)\right) ds \tag{5}$$

where $A_o(s)$ - amplitude at the elementary source, $\phi(s)$ - initial phase (at the elementary source), $k = 2\pi/\lambda$ - wave number, r - a distance from the elementary source to the observation point, $(A_o(s)/r)\exp\left(-ikr + \phi(s)\right)$ - complex amplitude.

The intensity (forming the diffraction pattern) is given by:

$$I = |D|^2 = Re(D)^2 + Im(D)^2 \tag{6}$$

In the case of an infinite slit of width a the problem becomes 2-dimensional, in the sense that no quantity varies in the direction parallel to the slit, so only two remaining directions are of interest. Assuming that initial amplitude and phase (A_o, ϕ) are constant along the slit the diffraction integral on the screen placed at the distance d takes the form:

$$D = \int_{-a/2}^{a/2} A(r) \exp\left(-ikr\right) dx \tag{7}$$

where $r = \sqrt{(y-x)^2 + d^2}$, $A(r) = A_o/\sqrt{r}$ (for a cylindrical wave emitted by an elementary line source), A_o - the amplitude at the source, x is the coordinate of the elementary source and y - the coordinate of the observation point on the screen.

4.2 Numerical derivatives and quadrature

Content: *various multipoint derivative and quadrature schemes based on the local interpolation of a function by a power series.*

Derivatives

Suppose we want to calculate numerically derivatives of various orders of a function $f(x)$, at certain point x. We start from power series expansions:

$$\begin{cases} f(x \pm h) = f(x) \pm f'(x)h + \frac{1}{2}f''(x)h^2 \pm \frac{1}{6}f'''(x)h^3 + ... \\ f(x \pm 2h) = f(x) \pm f'(x)2h + \frac{1}{2}f''(x)(2h)^2 \pm \frac{1}{6}f'''(x)(2h)^3 + ...) \\ ... \end{cases} \tag{8}$$

where h is a small grid parameter. The above power series expansions, when cut at certain order, form a system of linear equations whose unknowns are subsequent derivatives. For example, if only the terms up to second order are preserved we get:

$$\begin{cases} f(x+h) = f(x) + f'(x)h + \frac{1}{2}f''(x)h^2 + O(h^3) \\ f(x-h) = f(x) - f'(x)h + \frac{1}{2}f''(x)h^2 + O(h^3) \end{cases} \tag{9}$$

The symbol $O(h^3)$ means that the leading term in the rest of the series is on the order of h^3.

By subtracting or adding the equations (9) we easily derive the 3-point schemes for the first and the second derivative, respectively:

$$f'(x) = \frac{f(x+h) - f(x-h)}{2h} + O(h^2) \tag{10}$$

$$f''(x) = \frac{f(x+h) + f(x-h) - 2f(x)}{h^2} + O(h^2) \tag{11}$$

Note that the uncertainty of the second derivative is of the order of h^2, this is because the leading terms of the third order in the two expansions cancel out. The above formulae are called 3-point schemes because the local interpolation of the function based on 3 points (parabolic) is used. The formulae give exact values of derivatives for quadratic functions. It is also interesting to note that the first derivative formula uses only 2 points, in spite of the fact that this is a 3-point scheme.

Using the above procedure it is easy to derive schemes based on a larger number of points (higher order polynomial interpolations) as well as derivatives of a higher order. It is also easy to incorporate a nonuniform grid (h is different at each step), although then the formulae become more complicated.

Quadrature

The schemes presented here are based on the assumption that the function $f(x)$ which we want to integrate is tabulated, i.e. its values are known at some points of the domain uniformly distributed along the integration interval. The grid parameter h is the distance between subsequent values of the argument in the grid: $h = x_{i+1} - x_i$. The idea behind the construction of the schemes is simple. Once we can calculate numerically the derivatives of any order at the points of the argument grid, the function can be interpolated by a polynomial of any order along certain local interval. Thus, it can also be locally integrated along the same local interval. The quadrature along the whole interval of interest is obtained by summing up the values of local integrals.

Very popular Simpson algorithm is based on the given above 3-point schemes for the first and the second derivatives (and local quadratic interpolation of the function):

$$D = \int_{-h}^{h} f(x)dx = \frac{h}{3}(f(x+h) + 4f(x) + f(x-h)) + O(h^5) \tag{12}$$

Note that the uncertainty of the quadrature is very low (on the order of h^5). Could you explain why?

It should be pointed out that the schemes of higher orders do not always lead to better performance of computer programs since they usually need more evaluations of the function (at a larger number of points). The same goal (better accuracy) can be achieved by reducing the grid parameter h in the schemes of lower order. This can be easily done with modern computers and this is perhaps the reason why in the computational practice the 3-point schemes are most popular.

4.3 Exercises

Numerical procedures

Obligatory

1. (Testing the program DERIV) Substitute your own function in the segment FUNC and calculate its derivatives. Compare the results with analytical values.

2. Check the convergence of the 1^{st} and the 2^{nd} derivative with respect to the grid parameter h (draw derivatives as functions of $-log_{10}(h)$. Do the tests for a single and double precision of real numbers. Discuss the results.

3. Test the program QUADRAT by integrating a function whose analytical quadrature is known.

4. Check the convergence of the Simpson algorithm with respect to the grid parameter h. How does it compare with the derivative convergence (discuss) ?
5. Evaluate the number π from the length of an arc with use of the QUADRAT program.

Supplementary
6. Derive the 5-point finite difference formulae for the 1st and the 2nd derivatives of a function. Construct the functions *fp5* and *fpp5* (the 1^{st} and the 2^{nd} derivative withe use of the 5-point scheme)in DERIV program and repeat the Ex.2. Compare the convergence of 3-point and the 5-point schemes.
7. Substitute a 2^{nd} and a 5^{th} order polynomial in the segment FUNC and test the convergence with respect to h. Discuss of the results.

Diffraction
Obligatory
1. Test the program DIFFRACTION by comparing the first minimum position with the analytical value.
2. Evaluate of diffraction patterns for different sets of input parameters and draw respective graphs (near and far field, narrow and wide slit etc.).
3. Modify the program so that it could calculate the diffraction pattern produced by a system of two slits. Discuss the evaluated diffraction pattern.

Supplementary
4. Modify the program so that it could calculate the diffraction pattern produced by a system of many parallel slits (hint: use a loop to simulate a big number of slits.
5. Use the modified program to simulate a diffraction grating and draw graphs of diffraction patterns for different grating parameters (the width and the separation of slits). Discuss the results.

Challenge
6. Consider various initial phase distributions $\Phi(s)$ across a single slit (or at slits in diffraction grating). How do they affect the diffraction pattern ?
7. Consider the case of diffraction due to a circular aperture or an aperture of arbitrary shape.

5 Project: Pendulum as a standard of the unit of time

Preliminary reading: *Newton's lows of motion, harmonic and anharmonic oscillators, driven oscillations, concept of a phase space.*

This project is devoted to the Initial Value Problem for the ordinary differential equations. Participants learn various multipoint recursion schemes, apply them to some examples of equations whose analytical solutions are known, check the convergence with respect to the grid parameter and compare the quality of the schemes. Finally a chosen scheme is applied to study the properties of the compound pendulum, in particular the dependence of the period of oscillation on energy. The results may serve to discuss the applicability of the pendulum as a standard of the unit of time.

5.1 Simple pendulum

Content: *pendulum equation, initial conditions, scaling the equation, the case of small oscillations (isochronism), the period in the whole range of energies, decomposition of the 2nd order equation into a system of two 1st order equations, phase space.*

Simple pendulum (point mass m suspended from a massless, stiff rod of length l) equation of motion:

$$\frac{d^2\theta}{dt^2} = -\frac{g}{l} sin(\theta) \tag{13}$$

θ is the swing angle, g - free fall acceleration.

For small swing angles θ $sin(\theta) \approx \theta$ and the oscillation is harmonic:

$$\frac{d^2\theta}{dt^2} = -\frac{g}{l}\theta \tag{14}$$

$$\theta(t) = \theta_o sin(\omega t) \tag{15}$$

where θ_o - swing angle amplitude, $\omega = 2\pi/T = \sqrt{g/l}$.

For small swing angles the period of oscillation T does not depend on the amplitude. This phenomenon (isochronism) allows us to use a pendulum as the standard of the unit of time in pendulum clocks (the first one constructed in 1656 by Christiaan Huygens). However, for bigger swing angles the period begins to change and to find it, it is necessary to solve numerically the differential equation (13) for given inial values of angle and its first derivative (initial value problem).

A convenient way of setting the state of the pendulum is to use its total energy $E = mgl(1-cos(\theta))+ml^2\omega^2/2$ expressed in units of the maximum potential energy (with respect to the lowest position), i.e. $\epsilon = E/2mgl$. Then we expect the period to be constant for $\epsilon \ll 1$, to tend to infinity for ϵ approaching 1, and to behave like $1/\sqrt{\epsilon}$ for $\epsilon \gg 1$ (explain why?).

5.2 Ordinary differential equations

Content: *Euler, Adams-Bashforth and Runge-Kutta schemes, the implicit and 'predictor-corrector' algorithms.*

The system of first order, linear, nonuniform differential equations:

$$\begin{cases} \dfrac{dy_i}{dx} = f(y_1, ..., y_N, x) \end{cases} \tag{16}$$

where x - independent variable, $\{y_1, ..., y_N\}$ - dependent variables (functions of x).

The linear differential equations of higher order can be decomposed into the system (16) by defining auxiliary functions being derivatives of the function $y(x)$.

Developing the numerical methods for a single equation:

$$\frac{dy}{dx} = f(y, x) \tag{17}$$

suffices, since the methods can be easily adopted to the system (16).

A regular mesh (grid) of points $\{x_0, x_1, ..., x_N\}$, in the interval (x_0, x_N) is introduced. The distance between neighboring points: $h = (x_{i+1} - x_i)$, also $h = (x_N - x_0)/N$, is called the grid parameter h. We seek a recursion formula of the form:

$$y_{n+1} = F(y_n, y_{n-1}, y_{n-2}, ...) \tag{18}$$

which would allow to develop the function $y(x)$, beginning from the initial point $y_0(x_0)$.

A starting formula is an exact integral of the equation (17) over the interval (x_n, x_{n+1}):

$$y_{n+1} = y_n + \int_{x_n}^{x_{n+1}} f(x, y)dx \tag{19}$$

In general the explicit form of $f(x, y)$ is not known since the $y(x)$ is not known (it is to be found). However, values of $y(x)$ at previous points of mesh $y_n, y_{n-1}, y_{n-2}, ...$ are known and they can be used to extrapolate $y(x)$ over the interval (x_n, x_{n+1}). Depending on the number

of points used in the extrapolation we get so called explicit schemes of various orders. For example the most simple Euler's formula (based on the extrapolation by a step function) reads:

$$y_{n+1} = y_n + f_n h + O(h^2) \tag{20}$$

where $f_n = F(x_n, y_n)$. The leading term of the local deviation from the exact solution if on the order of h^2 (explain why?). It should be noted that after N steps the uncertainty (global) will be one order lower because $h \simeq 1/N$, thus $h^2 \cdot N = h^2/h = h$.

The three step Adams-Bashforth scheme, based on the parabolic extrapolation has the form:

$$y_{n+1} = y_n + \frac{h}{12}(5f_{n-2} - 16f_{n-1} + 23f_n) + O(h^4) \tag{21}$$

The Runge-Kutta methods are based on more complicated mathematical idea which will not be discussed here, although the integral (19) is still used as the starting point (for reference see e.g. [1, 2, 3]). They are regarded as the best integration schemes, whose great advantage is the fact that only one, already known, point x_n, y_n (e.g. the initial condition) is needed to perform an integration step, whereas the multipoint recursion schemes cannot be started from a single initial condition. As an example, the 4^{th} order Runge-Kutta scheme has the form:

$$\begin{aligned} k_1 &= hf(x_n, y_n) \\ k_2 &= hf(x_n + 1/2h, y_n + 1/2k_1) \\ k_3 &= hf(x_n + 1/2h, y_n + 1/2k_2) \\ k_4 &= hf(x_n + h, y_n + k_3) \\ y_{n+1} &= y_n + \tfrac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) + O(h^6) \end{aligned} \tag{22}$$

The unknown function $y(x)$ appearing in the integrand (19) can be also interpolated by a polynomial of any order. In this case the unknown value y_{n+1} appears also on the right hand side of the formula and has to be evaluated by solving the equation. This strategy leads to a group of implicit schemes. They are most often used in the predictor-corrector methods in which a predicted value of y_{n+1} found from one of explicit is then corrected by the implicit scheme of higher order.

5.3 Exercises

Numerical procedures
Obligatory

1. Use the IVP program to evaluate the function being a solution to the differential equation $dy/dx = -y, y(0) = 1$, with the use of three schemes: Euler, 3^{rd} order Adams-Bashforth and 5^{th} order Runge-Kutta, visualize the results.
2. Modify the code so that it gives at the output the $y(1)$ value as a function of the grid parameter $-log_{10}(h)$ (the logarithmic scale). Check the convergence of the three schemes with respect to the grid parameter. Classify the schemes with respect to their efficiency.
3. Modify the Euler method function so that it solves a system of differential equations. Use it to solve the system obtained from decomposing the equation $d^2y/dx^2 = -k \cdot y$. *Hint:* the equation can be decomposed into: $dy/dx = v$, $dv/dx = -ky$.

Supplementary
4. Use the IVP2D program to Evaluate the function being a solution to the second order differential equation $d^2y/dx^2 = -k \cdot y$ (the harmonic oscillator) with the use of a chosen scheme. Add in the code a line calculating the total energy of the oscillator and output the result. Find the range of the grid parameter at which the energy is conserved.
5. Modify the code IVP2D code so that it could solve the equation $d^2y/dx^2 + k \cdot y - \beta \cdot dy/dx = Asin(\omega \cdot x)$ (driven oscillator); visualize and discuss the results for different parameters (damping, elastic constant, frequency of driving force).

Challenge
6. Consider the case of coupled pendulums, described by the system of equations (1D system in which 2 oscillators of spring constants k_1, k_2 are connected by a spring of k constant)

$$\begin{aligned} d^2y_1/dx^2 &= -k_1y_1 - k(y_1 - y_2) \\ d^2y_2/dx^2 &= -k_2y_2 - k(y_2 - y_1) \end{aligned} \tag{23}$$

Study the behavior of the system as dependent on its physical parameters, in particular the periodic energy transfer from one oscillator to the other.

Compound pendulum
1. Test the program PENDULUM by comparing the results for small oscillations with the analytical solution.
2. Investigate into the dynamics of the pendulum for different total energies (the energy is expressed in units of the maximum potential energy). Check the asymptotic behavior of frequency or period, for the energy tending to zero and to infinity. Evaluate the function $(T(E) - T_o)/T_o$, where T_o is a small oscillation period (the relative deviation of the period from its "small oscillation" value). The calculations can be limited to a few points in characteristic regions: the low, the system characteristic and the high energy.

6 Project: Planetary system

Preliminary reading: *Newton's law of universal gravitation, Newton's laws of motion.*

The molecular dynamics (MD) is one of most important kinds of simulation in physics. It is just solving the Newtonian equations of motion for a system of many particles. The rigorous analytical solution exists only for two-particle systems (and three-particle in special cases), thus the computer simulation seems to provide the only possibility of studying such systems theoretically. The MD simulation is very popular in studying the dynamics and thermodynamics of polyatomic systems but they are also used on a cosmic scale (motion of planets, stars, galaxies). All we need to know to construct the MD code is the law of interaction between the particles and Newton's (or perhaps relativistic) equations of motion. In this project a simple 2-dimensional planetary system will be considered with two planets and a fixed star as the source of the central force. Newton's law of universal gravitation is used as the interaction law between planets and between planets and the star. The Verlet algorithm (most often used in MD simulation) for solving the initial value problem will be applied.

6.1 Motion of planets

Content: *equations of motion of planets, energy and angular momentum conservation.*

The forces acting on planets due to the Star fixed in the origin of a reference system and to the other planet moving in the same plane:

$$\begin{aligned} \mathbf{F}_1 &= -G\frac{m_1 M}{r_1^2}\hat{r}_1 - G\frac{m_1 m_2}{r_{12}^2}\hat{r}_{12} \\ \mathbf{F}_2 &= -G\frac{m_2 M}{r_2^2}\hat{r}_2 - G\frac{m_2 m_1}{r_{21}^2}\hat{r}_{21} \end{aligned} \tag{24}$$

where $\mathbf{r}_{ij} = \mathbf{r}_j - \mathbf{r}_i$, and $\hat{r} = \mathbf{r}/r$ is a unit vector associated with $\mathbf{r}$.
The Newton equations of motion for the two planets:

$$\begin{cases} d\mathbf{p}_1/dt = \mathbf{F}_1 \\ d\mathbf{r}_1/dt = \mathbf{p}_1/m_1 \\ d\mathbf{p}_2/dt = \mathbf{F}_2 \\ d\mathbf{r}_2/dt = \mathbf{p}_2/m_2 \end{cases} \tag{25}$$

6.2 Verlet algorithm

Content: *Verlet algorithm.*

The Verlet algorithm is a simple scheme developed for solving Newton's equations of motion for a system of particles (Molecular Dynamics), and is most often used in MD simulations. Here, the idea of the algorithm is presented on the example of 1D motion, but it can be easily generalized to the many dimensional case of a system of particles.

Newton's equation of motion of a point mass m in 1D:

$$\frac{d^2x}{dt^2} = F/m \tag{26}$$

where F - force acting on a particle of mass m, x - coordinate of the particle on the X axis.

To study the trajectory in phase space also the velocity $v = dx/dt$ is needed.

We use 3-point numerical formulae (10,11) to express position and velocity, which, when rearranged lead to the following recursion schemes (Verlet's algorithm):

$$\begin{aligned} x_{n+1} &= 2x_n - x_{n-1} + \tau^2 F/m + O(\tau^4) \\ v_n &= (x_{n+1} - x_{x-1})/(2\tau) + O(\tau^2) \end{aligned} \tag{27}$$

where τ is the time step.

It should be noted that as initial conditions usually position and velocity (or momentum) are given (x_o, v_o), thus, there is certain difficulty in starting the recursion scheme since it needs 2 initial points. The difficulty can be overcome e.g. by assuming that over the first time step the system performs the uniformly accelerated motion:

$$x_1 = x_o + v_o\tau + (F/m)\tau^2/2 + O(\tau^3) \tag{28}$$

which, however, is less accurate by 1 order of magnitude of τ and thus should be applied with caution. Alternatively, more accurate schemes (e.g. Runge-Kutta) over the first time step can be used.

6.3 Exercises

Obligatory

1. (Testing the program) Test the program on the example of motion of a single planet. For that purpose modify the code PLANETS so that one of the planets is fixed far aside and its mass is very low. Check the energy and the momentum conservation. Find the maximum time-step for which the energy and the angular momentum are conserved. Compare the results with an analytical solution for a circular orbit.

2. Try to find elliptic, parabolic and hyperbolic trajectories for a single planet (is it possible distinguish between the parabolic and the hyperbolic orbit in the computer simulation ?) (hint: consider the total energy).
3. For an elongated elliptical orbit observe the evolution of the total energy, explain the observed behavior.
4. Include the motion of the second planet. Observe the dynamics of the system for different initial conditions. Check the energy and the angular momentum conservation. For example try to arrange a situation where the second planet is a satellite of the first one, or the second planet plays a role of the second star.

Supplementary
5. By changing the mass of the fixed planet and its position observe its effect on the trajectory of the second planet (a perturbation). Check the energy and the momentum conservation (should those quantities be both conserved ?). Try to characterize the effect of the perturbation on motion of the planet.

Challenge
6. By introducing proper numerical values of the parameters observe the effect of the presence of Venus (or Mars) on the trajectory of the Earth (hint: keep the position of Venus (Mars) fixed). Note that the period of motion of the Earth is about 365 days, what is the time-step then ?
7. Check the third Keppler law of planetary motion.
8. Propose an algorithm in which the time step would dynamically change to minimize the CPU time but preserving assumed accuracy of calculations. Test the proposed algorithm by repeating Ex.3.

7 Project: Gravitation inside a star

Preliminary reading: *Poisson's equation for spherical charge density distribution, analogy to gravitation.*

This project is devoted to the Boundary Value Problem (BVP) for differential equations. The case of gravitational field inside a star of model radial mass density distribution is considered. The problem is formally equivalent to the problem of the electric field inside an atom. The partial differential equation of Poisson's type, because of high symmetry (spherical), reduces to the 2^{nd} order ordinary differential equation. At variance with the initial value problem, here, the two conditions necessary to define uniquely the solution are

given at two ends of the independent variable range (not at one end, like in the initial value problem). Usually the values of the function being the solution are given at the boundaries (Boundary Value Problem). However, to start one of the recursion algorithms the values of the function at least two points at one end are needed. Some solutions to this problem are presented.

7.1 Poison's equation

Content: *Poison's equation and examples of its application.*

Poison's equation:

$$\nabla^2\phi(\mathbf{r}) = -4\pi\rho(\mathbf{r}) \tag{29}$$

is in general a partial differential equation which together with the boundary conditions forms the boundary value problem whose solution is the electric potential function $\phi(\mathbf{r})$ for a given charge density distribution $\rho(\mathbf{r})$. Formally identical equation (with the gravitational constant G multiplying the right hand side) can be used to find the gravitational potential for a given mass density distribution.

In the spherical coordinates and for a system of spherical symmetry it acquires a simpler 1D form:

$$\frac{1}{r^2}\frac{d}{dr}\left(r^2\frac{d\phi(r)}{dr}\right) = -4\pi\rho(r) \tag{30}$$

A standard substitution $\phi(r) = \varphi(r)/r$ simplifies the formula even more:

$$\frac{d^2\varphi(r)}{dr^2} = -4\pi r\rho(r) \tag{31}$$

This is a second-order, linear, inhomogeneous, ordinary differential equation. Such an equation needs two conditions to define uniquely the solution. In the case of initial value problem these are usually the value of the function and its first derivative at certain initial point which allows to start the 2-point recursion numerical algorithm. However, in the case of the above equation there are boundary conditions, i.e. values of the function at two ends of a certain argument interval. This makes the application of the recursion schemes very inconvenient because an additional condition must be found to use them. In Chapter 9 a very useful matrix method will be discussed to solve this problem but here we will try to apply the initial value problem (IVP) methods.

We will use the above equations to find gravitational potential inside a star. Suppose the mass density distribution inside a star is given by the formula:

$$\rho(r) = \frac{1}{8\pi} e^{-r} \tag{32}$$

in which case the total mass of the star:

$$M = \int \rho(\mathbf{r}) d^3\mathbf{r} = \int_0^\infty \rho(r) 4\pi r^2 dr = 1 \tag{33}$$

the exact solution to this problem is

$$\varphi(r) = 1 - \frac{r+2}{2} e^{-r} \tag{34}$$

from which $\phi(r) = \varphi/r$ follows immediately (note that some specific units for the gravitational potential are used here). Through some independent speculations it is possible to find the additional initial condition to start the recursion Numerow-Cowling algorithm (described in the next section), but for the purpose of this project we will use the above formula to find the necessary conditions (e.g. values $\varphi(r = 0)$ and $\varphi(r = h)$).

7.2 Numerow-Cowling algorithm

Content: *Numerow-Cowling algorithm.*

We consider a class of the second order, linear, inhomogeneous differential equations, which describe many various physical systems (including almost all discussed so far):

$$\frac{d^2y}{dx^2} + k^2(x)y = S(x) \tag{35}$$

where $k^2(x)$ - a real function and $S(x)$ - an inhomogeneous ("driving") term.
Using the expansions (8) one can derive:

$$\frac{y_{n+1} - 2y_n + y_{n-1}}{h^2} = y_n'' + \frac{h^2}{12} y_n'''' + O(h^4) \tag{36}$$

On the right hand side of the above equation the second derivative is given by the differential equation:

$$y_n'' = (-k^2 y)_n + S_n \tag{37}$$

and the forth derivative can be evaluated numerically using the 3-point scheme as the second derivative of $y''(x)$:

$$y''''_n = -\frac{(k^2y)_{n+1} - 2(k^2y)_n + (k^2y)_{n-1}}{h^2} + \frac{S_{n+1} - 2S_n + S_{n_1}}{h^2} + O(h^2) \tag{38}$$

which, when substituted into (36), leads to the numerical expression (Numerow-Cowling algorithm):

$$(1+\frac{h^2}{12}k^2_{n+1})y_{n+1} - 2(1-\frac{5h^2}{12}k^2_n)y_n + (1+\frac{h^2}{12}k^2_{n-1})y_{n-1} = \frac{h^2}{12}(S_{n+1}+10S_n+S_{n-1}) + O(h^6) \tag{39}$$

Note that the substitution does not spoil the overall accuracy ($O(h^4)$) and the final local accuracy of the scheme remains very high ($O(h^6)$).

An important advantage of the expression is that it can be rearranged to give recursion formulae either in "forward" or "backward" direction. It can be also treated as a three-diagonal system of equations.

7.3 Exercises

Obligatory

1. Run the program BVP1D (Boundary Value Problem in 1D) at different values of control parameters: step parameter, upper boundary of integration interval, initial conditions. Compare the numerical solution for the gravitational potential with the analytical one.

2. Modify the program so that the Numerow-Cowling scheme works "backward", i.e. starting the recursion procedure in "infinity". Compare the results with analytical solutions, draw some conclusions.

3. Extend the code so that it could calculate the gravitational field. Do the calculations and represent the results graphically.

Challenge

4. Consider the case of an ideal gas at constant temperature forming a ball under gravitation (the simplest model of a star). Find the radial distribution of pressure, density, gravitational potential and gravitational field.

8 Project: Normal modes in a cylindrical waveguide

Preliminary reading*: waves, wave equation, standing waves, stationary wave equation.*

The Eigenvalue Problem (EVP) appears e.g. in the description of standing waves or stationary quantum systems. Many theoretical techniques have been developed to solve this problem. Here, a method is presented which is based on recursion schemes for solving ordinary differential equations: the shooting method. The method, although looking rather simple and coarse, since it deals with 1D case and thus can be applied only to low dimensional or high symmetry structures, is exploited in top research, e.g. in finding the atomic structure within Density Functional Theory. In this project a simple classical system is analyzed, namely a cylindrical waveguide (e.g. optical fiber) in which the normal modes of a scalar wave have to be found.

8.1 Standing waves

Content: *stationary wave equation in cylindrical symmetry, the eigenvalue problem, other eigenvalue problems in physics.*

The stationary wave equation in cylindrical coordinates and cylindrical symmetry:

$$\left(\frac{d^2}{dr^2} + \frac{1}{r}\frac{d}{dr}\right)\phi(r) = -k^2\phi(r) \tag{40}$$

with boundary conditions: $\phi(r = 0) = 1, \phi(r = 1) = 0$ This is the eigenvalue problem, thus, we expect the solution as a sequence of pairs $(\phi_n(r), k_n)$, i.e. the normal modes (standing waves) and associated wavenumbers. Upon substitution $\phi(r) = \varphi/\sqrt{r}$ the equation changes the form to be one for which the Numerow-Cowling algorithm can be applied:

$$\left[\frac{d^2}{dr^2} + \left(\frac{1}{4r^2} + k^2\right)\right]\varphi(r) = 0 \tag{41}$$

Note that the asymptotic behavior of the function $\varphi(r)$ in the limit $r \mapsto 0$ must be $\sim \sqrt{r}$. This fact can be used to avoid singularity at $r = 0$ in numerical calculations.

8.2 The shooting method

Content: *the shooting method for finding eigenvalues.*

The idea of the shooting method is simple. We perform the recursion procedure (like the Numerow-Cowling one) with a certain trial eigenvalue k^2, starting at one end of the domain. When the other end is reached we check the value of the function. If it obeys the boundary condition then the trial k^2 is the eigenvalue. In practice the problem reduces to finding the roots of the function: $\varphi_{r=1}(k^2)$.

In the case of a 1D quantum well:

$$\Big[-\frac{1}{2}\frac{d^2}{dx^2}+V(x)\Big]\psi(x)=\varepsilon\psi(x) \tag{42}$$

the problem is slightly more complicated. When integrating into the classically forbidden region (negative kinetic energy), if the eigenenergy differs from the true one (even by a very small amount), the very rapidly exponentially increasing solution appears and it is very difficult to hit the eigenvalue. In this case the recursion procedure must be conducted from both ends of the domain, i.e. always out off the classically forbidden region into the allowed one. The criterium for finding the eigenvalue is the smooth connection of the two solutions ("from the left" one and "from the right" one) at certain testing point which should be situated in the classically allowed region. Since the quantum states can always be normalized so that they connect the next condition is that their derivatives are equal. This leads to the criterium:

$$\frac{1}{\psi_<(x_m)}\Big[\psi_<(x_m-h)-\psi_>(x_m-h)\Big]=0 \tag{43}$$

where $\varphi_<$ and $\varphi_>$ are recursively found functions "from the left" and "from the right", respectively, and x_m is the testing point.

8.3 Exercises

Obligatory
1. Run the program WAVEGUIDE and test the convergence of results with respect to control parameters (initial value of r, r step)
2. Calculate the eigenvalues of the wave number and compare the numerical results with the analytical values: 2.404826, 5.520078, 8.653728, 11.791534.
3. Extend the code so that it could output the radial functions of normal modes amplitudes. Visualize the results.

Challenge
5. Consider the case of a nonuniform but still cylindrically symmetric refractive index $k(r)=n(r)*k_c$, where k_c is a wave number in a vacuum and $n(r)$ is the radial distribution of the refractive index (must not exceed 2.0 and its value at the waveguide wall is 1.0 - vacuum). How various functions $n(r)$ affect the wave numbers of normal modes ? (try e.g. parabolic or exponential)

6. Change the code so that it could find the energy levels of a rectangular quantum well discussed in Chapter 3, compare the results with the results obtained with the QWELL program.

9 Project: Thermal insulation properties of a wall

Preliminary reading: *diffusion phenomena, diffusion equation, steady-state diffusion.*

In this project students will study the insulating properties of the wall in a house, in particular, how the temperature across the wall depends on the thermal conductivity distribution of the insulating material. To do this we have to apply the steady-state diffusion equation. Students will also learn the **Finite Difference** (FD) numerical technique.

9.1 Steady-state diffusion

Content: *steady-state diffusion equation in 3D, the boundary value problem for partial differential equations.*

The steady-state diffusion equation:

$$-S(\mathbf{r}) = \nabla \cdot \left[D(\mathbf{r})\, \nabla \phi(\mathbf{r}) \right] \tag{44}$$

where $\phi(\mathbf{r})$ is the density of diffusing material at location $\mathbf{r}$ (does not depend on time), $D(\mathbf{r})$ is the diffusion coefficient at location $\mathbf{r}$, and ∇ is the vector differential operator:

$$\nabla = \widehat{i}\frac{\partial}{\partial x} + \widehat{j}\frac{\partial}{\partial y} + \widehat{k}\frac{\partial}{\partial z} \tag{45}$$

The $S(\mathbf{r})$ and $\phi(\mathbf{r})$ may have different meanings, depending on the kind of system which is considered. In particular they can be interpreted as the source of heat and the temperature, respectively. The diffusion coefficient $D(\mathbf{r})$ has the meaning of thermal conductivity in that case.

The equation is the elliptic partial differential equation which, together with the boundary conditions (values of the function and/or values of its derivatives at some borders surrounding the region of interest, and sometimes also inside the region), forms the Boundary Value Problem. For the sake of simplicity, in this project we will consider the 1D case of heat diffusion through a wall. The equation simplifies then to:

$$-S(x) = \frac{d}{dx}\left(D(x)\frac{d\phi(x)}{dx} \right) \tag{46}$$

or

$$-S(x) = D'(x)\frac{d\phi(x)}{dx} + D(x)\frac{d^2\phi(x)}{dx^2} \tag{47}$$

9.2 Finite Difference (FD) method

Content: *finite difference method (FD), the Gauss elimination with backward substitution algorithm for tridiagonal system of equations.*

In this project the **Finite Difference** (FD) method will be applied. Firstly, a regular grid is introduced, with the grid parameter h. Secondly, the differential equation is discretized, using e.g. the 3-point formula for derivatives. In the example considered, the discretization of equation (47) takes the form:

$$-S_i = D'_i\frac{\phi_{i+1} - \phi_{i-1}}{2h} + D_i\frac{\phi_{i+1} + \phi_{i-1} - 2\phi_i}{h^2} \tag{48}$$

or

$$\left(D_i - \frac{h}{2}D'_i\right)\phi_{i-1} - 2D_i\phi_i + \left(D_i + \frac{h}{2}D'_i\right)\phi_{i+1} = -S_i h^2 \tag{49}$$

As a result we obtain a system of linear equations whose unknowns are the values of the function ϕ at grid nodes. Thus, the finite difference method converts the boundary value problem for differential equation into the system of linear equations.

In our case this is a tri-diagonal system of linear equations of the form:

$$\left\{ A_i^-\phi_{i-1} + A_i^0\phi_i + A_i^+\phi_{i+1} = b_i \right. \tag{50}$$

Note that the boundary conditions ϕ_0 and ϕ_N have to be moved to the right-hand side of the equation.

To solve this system of equations we will use the very efficient Gaussian elimination with back-substitution algorithm. We assume that the solution satisfies a one-term forward recursion relation of the form:

$$\phi_{i+1} = \alpha_i\phi_i + \beta_i \tag{51}$$

where α_i and β_i are the coefficients to be determined. Substituting this into (50), we get

$$A_i^-(\alpha_i\phi_i + \beta_i) + A_i^0\phi_i + A_i^+\phi_{i-1} = b_i \tag{52}$$

which can be solved for ϕ_i to yield

$$\phi_i = \gamma_i A_i^- \phi_{i-1} + \gamma_i (A_i^+ \beta_i - b_i) \tag{53}$$

with

$$\gamma_i = -\frac{1}{A_i^0 + A_i^+ \alpha_i} \tag{54}$$

Upon comparing (51) with (53) we obtain the backward recursion relations for the $\alpha's$ and $\beta's$:

$$\alpha_{i-1} = \gamma_i A_i^- \tag{55}$$

$$\beta_{i-1} = \gamma_i (A_i^+ \beta_i - b_i) \tag{56}$$

The strategy now is as follows. We use the recursion relations (55, 56) to determine the α_i and β_i, for i running from $N-2$ down to 0. The starting values to be used are

$$\alpha_{N-1} = 0, \beta_{N-1} = \phi_N \tag{57}$$

which guarantee the correct value of ϕ at the upper boundary. Having known these coefficients, we use the recursion relation (51) in a forward sweep from $i = 0$ to $N-1$ to find the solution, with the starting value ϕ_0 known from the boundary condition. Thus, the solution is determined in only two sweeps of the lattice, involving of order N arithmetic operations.

Note that this technique can be also applied to the boundary value problem discussed in Chapter 7 (Gravitation inside a star). The method of solving the tri-diagonal system of linear equations can be treated here as a tricky way of "moving" one of boundary conditions to the opposite side.

9.3 Exercises

Obligatory

1. Using the program FTABLE tabulate the thermal conductivity coefficient D and its first derivative D' for a a few chosen functions (encreasing/decreasing linearly, Fermi-Dirac like function, your own choice).

2. Test the program DIFFUSION for the uniform linear equation: $\frac{d^2y}{dx^2} = 0$ ($D' = 0$,$D = 1$) for different boundary values and grid parameters.

3. Calculate the temperature across the wall for the thermal conductivity coefficient increasing from a small to a big value (according to a chosen function). Repeat the calculation for

the diffusion coefficient decreasing in the same way and having the same values at the boundaries, draw conclusions. (Assume the S source function has zero value in the whole range of x variable). Does the speed of the heat flow depend on whether the thermal conductivity is decreasing or increasing (according to same but reflected with respect to the wall center function)? Give the arguments.

Challenge
4. Note that the method considered here may serve for solving the Poisson equation. Try to use it for solving the problem of gravitation inside a star (Chapter 7) and confront the solution with the analytical one.
5. An interesting problem from applications point of view is a quantum well with certain number of charged particles occupying the energy levels. Here, an interaction between the particles (e.g. electrons) leads to a significant modification of the potential, which in turn causes the changes in the energy levels. Some components of the interaction are of purely quantum nature (exchange, correlation), but the dominant is the classical coulomb interaction between the charges, leading to so called Hartree potential. The method described in Chapter 8 may serve to find the energy levels in a quantum well of arbitrary shape, and the method developed in this Chapter allows to solve the 1D Poisson equation for the Hartree potential. Thus, the two methods when used together in a self-consistent procedure may give a solution to the problem. We start with a rectangular quantum well with certain number of electrons inside. The energy levels found are occupied by electron (following the Pauli exclusion principle) and form certain charge density, which is evaluated from modulus squared of respective wave functions. The charge density is then used in the Poisson equation to find the Hartree potential which when added to the initial (external) quantum well leads to a new quantum well. The procedure must be repeated until the convergence is reached.

10 Project: Cylindrical capacitor

Preliminary reading: *electrostatics, Poisson's and Laplace's equation.*

The electrostatics problem described by Poisson's equation, i.e. finding the electric potential for a given boundary conditions and charge density distribution has its variational formulation (as many other problems in physics). In physics language this principle says that the potential being the solution minimizes the total energy of the system (energy of the field plus the potential energy of the charge). However, there exists also a rigorous mathematical proof that solving the Poisson's differential equation is equivalent to the problem of minimization of a certain functional (variational principle), thus the method can be also

used in the cases where the "energy" interpretation does not apply. In this project, for the sake of simplicity, a high symmetry (cylindrical) electrostatic system has been chosen. This is a capacitor consisting of two coaxial metallic cylinders of different radii $\phi_0 = \phi(r_1)$ and $\phi_N = \phi(r_2)$. The partial differential Laplace's equation becomes one dimensional. The Gauss-Seidle iterative minimization of the function will be applied.

10.1 Variational principle for electrostatic systems.

Content: *Poison's equation in 3D, the boundary value problem - variational formulation.*

An electrostatic system is described by Poison's equation:

$$-\nabla^2\phi(\mathbf{r}) = 4\pi\rho(\mathbf{r}) \tag{58}$$

which, for a given charge density distribution $\rho(\mathbf{r})$ and boundary values, forms the boundary value problem for an elliptic partial differential equation (see Chapter 7), the solution to which is the electric potential function $\phi(\mathbf{r})$. However, there exists an alternative approach to the problem: the **variational principle**. It can be formally proved that the function $\phi(\mathbf{r})$ being the solution to (58) minimizes the functional:

$$F[\phi] = \int d^3\mathbf{r}\big[\frac{1}{2}(\nabla\phi(\mathbf{r}))^2 - 4\pi\rho(\mathbf{r})\phi(\mathbf{r})\big] \tag{59}$$

In the case of electrostatics the above functional is the total electrostatic energy of the system, but since the proof is general it can be treated just as a useful functional in other cases. The problem is then converted into the problem of finding the minimum of (59). Upon any parametrization of the function $\phi(\mathbf{r})$ with the set of parameters $\{\alpha_i\}$ the functional becomes a function $F(\{\alpha_i\})$ and the problem reduces to finding the minimum of the function. In the **finite elements** method the function is represented by its values at the grid nodes $\{\phi_i\}$ which become the parameters to be found. At variance with the **finite difference** method the grid can be shaped in arbitrary way, e.g. can be denser in the regions where the function is expected to vary at a high rate, or can have a triangular rather than rectangular geometry.

The system considered in this project has cylindrical symmetry. The Poisson equation takes the form:

$$-4\pi\rho(r)r = \frac{\partial}{\partial r}\left(r\frac{\partial\phi}{\partial r}\right) \tag{60}$$

which together with the boundary values $\phi(r_1)$ and $\phi(r_2)$ leads to the boundary value problem (similar to that discussed in Chapter 7). The above equation is also formally equivalent to the diffusion equation discussed in the previous chapter, and thus the method introduced there can be also applied. In this project however, we will use the variational principle. A respective functional in the cylindrical coordinates reads:

$$F[\phi] = \int_{r_1}^{r_2} dr \left[\frac{1}{2} \left(\frac{d\phi(r)}{dr} \right)^2 - 4\pi\rho(r)\phi(r) \right] r \tag{61}$$

10.2 Finite Elements (FE) method

Content: *finite elements method, the Gauss-Seidel iterative minimization.*

The simplest possible representation of (61) after discretization has the form:

$$F = \frac{1}{2h} \sum_{i=1}^{N} r_{i-1/2} (\phi_i - \phi_{i-1})^2 - h \sum_{i=0}^{N-1} 4\pi\rho_i \phi_i r_i \tag{62}$$

where $r_i = r_1 + ih$, $r_{i-1/2} = r_1 + (i - 1/2)h$, $h = (r_2 - r_1)/N$.
The condition for the minimum:

$$\frac{\partial F}{\partial \phi_i} = 0 \tag{63}$$

leads to:

$$\phi_{i+1} r_{i+1/2} - \phi_i (r_{i+1/2} + r_{i-1/2}) + \phi_{i-1} r_{i-1/2} + h^2 4\pi\rho_i r_i = 0 \tag{64}$$

This is a starting equation for the Gauss-Seidel iterative minimization procedure. Let us solve it for ϕ_i:

$$\phi_i = \phi_{i+1} \frac{r_{i+1/2}}{2r_i} + \phi_{i-1} \frac{r_{i-1/2}}{2r_i} + 2\pi h^2 \rho_i \tag{65}$$

or

$$\phi_i = \frac{1}{2}(\phi_{i+1} + \phi_{i-1}) + \frac{h}{4r_i}(\phi_{i+1} - \phi_{i-1}) + 2\pi h^2 \rho_i \tag{66}$$

The strategy is then as follows: (i) we make an initial guess of the function ϕ (e.g. the linear function between $\phi_0 = \phi(r_1)$ and $\phi_N = \phi(r_2)$, (ii) a new value of ϕ_i is calculated from (66) at each point of the grid r_i (except for the boundary values), (iii) at the new set $\{\phi_i\}$

the value of the functional (62) is calculated, (iv) the whole procedure is repeated until (62) stabilizes (does not change any more).

This procedure can be further improved by the mixing of the new guess for ϕ_i with the old one. This is done via the mixing parameter ω in the following way:

$$\phi_i' = \phi_i^{old}(1-\omega) + \phi_i^{new}\omega \tag{67}$$

where ϕ_i^{new} is the new guess (66) and ϕ_i' is the value to be taken in the nest run.

It can be proved that the procedure is convergent for a certain range of ω values (to be found as an exercise). In the following exercises we will also assume that the density of charge ρ is zero in the space between metallic cylinders, only the values of the potential at the cylinders $\phi(r_1)$ and $\phi(r_2)$ are given (boundary conditions).

10.3 Exercises

Obligatory

1. Test the program CAPACITOR for different boundary conditions and control parameters.

2. Investigate into the effect of "mixing"; find the range of the parameter ω at which the procedure is convergent and find its value at which the relaxation is the fastest.

3. Perform the fully converged calculations for chosen boundary conditions and compare the results with the analytical solution: $\varphi(r) = A\ln(r) + B$. Using an independent graphical tool (e.g. GNUPLOT) fit the analytical function to the numerical results.

Challenge

4. Think of an algorithm (similar to that considered in this section) and write an appropriate program that solves the 2D Laplace equation on a 2D rectangular lattice for given values at the boundaries and on certain points inside the domain. Note that the Gauss-Seidel iterative minimization must be done alternately vertically and horizontally across the lattice. Visualize the evaluated potential. (The program can be used to model some typical electrostatic systems like electric dipole, a system of charges, parallel plate capacitor, Faraday cage etc.)

References

[1] Koonin S., Meredith D., *Computational Physics*, Addison-Wesley, 1990

[2] Press W. H., Flannery B. P., Teukolsky S. A., Vetterling W. T., *Numerical Recipes*, Cambridge University Press, 1987

[3] Tao Pang, *An introduction to Computational Physics*, Cambridge University Press, 1997