5 Higher-order ODEs and systems of ODEs

5.1 General-purpose discretization schemes for systems of ODEs

The strategy of generalizing a discretization scheme from one to $N > 1$ ODEs is, for the most part, straightforward. Therefore, below we will consider only the case of a system of $N = 2$ ODEs. This case will also allow us to investigate a certain issue that is specific to systems of ODEs and does not occur for a single ODE. We will denote the exact solutions of the ODE system in question as $y^{(1)}(x)$ and $y^{(2)}(x)$, while the corresponding numerical solutions of this system, as $Y^{(1)}$ and $Y^{(2)}$; the functions appearing on the r.h.s. of the system will be denoted as $f^{(1)}$ and $f^{(2)}$. Thus, the IVP for the two unknowns, $y^{(1)}$ and $y^{(2)}$, is:

$$\begin{aligned} y^{(1)\prime} &= f^{(1)}(x, y^{(1)}, y^{(2)}), \quad y^{(1)}(x_0) = y_0^{(1)}, \\ y^{(2)\prime} &= f^{(2)}(x, y^{(1)}, y^{(2)}), \quad y^{(2)}(x_0) = y_0^{(2)}. \end{aligned} \tag{5.1}$$

We now consider generalizations of some of the methods introduced in Lectures 1 and 2.

Simple Euler method

Probably the most intuitive form of this method for two ODEs is

$$\begin{aligned} Y_{n+1}^{(1)} &= Y_n^{(1)} + hf^{(1)}\left(x_n, \left\{Y_n^{(1)}, Y_n^{(2)}\right\}\right), \\ Y_{n+1}^{(2)} &= Y_n^{(2)} + hf^{(2)}\left(x_n, \left\{Y_n^{(1)}, Y_n^{(2)}\right\}\right). \end{aligned} \tag{5.2}$$

Already for this most basic example, we can identify the issue, mentioned above, that is specific to systems of ODEs and does not occur for a single first-order ODE. Namely, notice that once we have found the new value $Y_{n+1}^{(1)}$ for the first component of the solution, we can substitute it into the second equation instead of substituting $Y_n^{(1)}$, as it is done in (5.2). The result is:

$$\begin{aligned} Y_{n+1}^{(1)} &= Y_n^{(1)} + hf^{(1)}\left(x_n, \left\{Y_n^{(1)}, Y_n^{(2)}\right\}\right), \\ Y_{n+1}^{(2)} &= Y_n^{(2)} + hf^{(2)}\left(x_n, \left\{Y_{n+1}^{(1)}, Y_n^{(2)}\right\}\right). \end{aligned} \tag{5.3}$$

Since the components $Y^{(1)}$ and $Y^{(2)}$ enter Eqs. (5.2) on equal footing, we can interchange their order in (5.3) and obtain:

$$\begin{aligned} Y_{n+1}^{(2)} &= Y_n^{(2)} + hf^{(2)}\left(x_n, \left\{Y_n^{(1)}, Y_n^{(2)}\right\}\right), \\ Y_{n+1}^{(1)} &= Y_n^{(1)} + hf^{(1)}\left(x_n, \left\{Y_n^{(1)}, Y_{n+1}^{(2)}\right\}\right). \end{aligned} \tag{5.4}$$

It is rather straightforward to see that all the three implementations of the simple Euler method are first-order methods.

An obvious question that now comes to mind is this: Is there any aspect because of which methods (5.3) and (5.4) may be preferred over method (5.2)? The short answer is 'yes, for a certain form of $f^{(1)}$ and $f^{(2)}$, there is'. We will present more detail in Sec. 5.3 below. For now we continue with presenting the discretization scheme for the Modified Euler equation for two first-order ODEs.

Modified Euler method

$$\begin{aligned}
\overline{Y^{(k)}} &= Y_n^{(k)} + h f^{(k)}\left(x_n, \left\{Y_n^{(1)}, Y_n^{(2)}\right\}\right), \\
Y_{n+1}^{(k)} &= Y_n^{(k)} + \tfrac{h}{2}\left[f^{(k)}\left(x_n, \left\{Y_n^{(1)}, Y_n^{(2)}\right\}\right) + f^{(k)}\left(x_{n+1}, \left\{\overline{Y^{(1)}}, \overline{Y^{(2)}}\right\}\right)\right], \qquad (5.5)\\
& \quad k = 1, 2\,.
\end{aligned}$$

Let us verify that (5.5) is a second-order method, as it has been for a single ODE. We proceed in exactly the same steps as in Lecture 1. We will also use the shorthand notations:

$$\vec{\mathbf{Y}}_n = \left\{Y_n^{(1)}, Y_n^{(2)}\right\}, \qquad \vec{\mathbf{y}}_n = \left\{y_n^{(1)}, y_n^{(2)}\right\}, \qquad \vec{\mathbf{f}}_n = \left\{f^{(1)}(x_n, \vec{\mathbf{Y}}_n), f^{(2)}(x_n, \vec{\mathbf{Y}}_n)\right\} \equiv \left\{f_n^{(1)}, f_n^{(2)}\right\}.$$

Since in the derivation of the local truncation error we always assume that $Y_n^{(k)} = y_n^{(k)}$, then also

$$\left\{f^{(1)}(x_n, \vec{\mathbf{y}}_n), f^{(2)}(x_n, \vec{\mathbf{y}}_n)\right\} = \left\{f_n^{(1)}, f_n^{(2)}\right\}.$$

Expanding the r.h.s. of the second of Eqs. (5.5) about the "point" $(x_n, \vec{\mathbf{Y}}_n)$ in a Taylor series, we obtain:

$$\begin{aligned}
Y_{n+1}^{(k)} &= Y_n^{(k)} + \frac{h}{2}\left[f^{(k)}\left(x_n, \vec{\mathbf{Y}}_n\right) + f^{(k)}\left(x_{n+1}, \vec{\mathbf{Y}}_n + h\vec{\mathbf{f}}_n\right)\right] \\
|_{\text{Taylor expansion}} &= Y_n^{(k)} + \frac{h}{2}\left[f_n^{(k)} + \left\{f_n^{(k)} + h\frac{\partial f_n^{(k)}}{\partial x} + h f_n^{(1)}\frac{\partial f_n^{(k)}}{\partial y^{(1)}} + h f_n^{(2)}\frac{\partial f_n^{(k)}}{\partial y^{(2)}}\right\}\right] + O(h^3) \\
&= Y_n^{(k)} + h f_n^{(k)} + \frac{h^2}{2}\left[\frac{\partial f_n^{(k)}}{\partial x} + f_n^{(1)}\frac{\partial f_n^{(k)}}{\partial y^{(1)}} + f_n^{(2)}\frac{\partial f_n^{(k)}}{\partial y^{(2)}}\right] + O(h^3)\,. \qquad (5.6)
\end{aligned}$$

Now expanding the exact solution $y_{n+1}^{(k)} = y^{(k)}(x_{n+1})$ in a Taylor series, we obtain:

$$\begin{aligned}
y_{n+1}^{(k)}\,|_{\text{Taylor expansion}} &= y_n^{(k)} + h\frac{d}{dx}y_n^{(k)} + \frac{h^2}{2}\frac{d^2}{dx^2}y_n^{(k)} + O(h^3) \\
|_{\text{definitions of } dy^{(k)}/dx} &= y_n^{(k)} + h f_n^{(k)} + \frac{h^2}{2}\frac{d}{dx}f_n^{(k)} + O(h^3) \\
|_{\text{Chain rule}} &= y_n^{(k)} + h f_n^{(k)} + \frac{h^2}{2}\left[\frac{\partial f_n^{(k)}}{\partial x} + f_n^{(1)}\frac{\partial f_n^{(k)}}{\partial y^{(1)}} + f_n^{(2)}\frac{\partial f_n^{(k)}}{\partial y^{(2)}}\right] + O(h^3)\,. \qquad (5.7)
\end{aligned}$$

Here the coefficient of the h^2-term has been computed by using the fact that $f^{(k)} = f^{(k)}(x, y^{(1)}(x), y^{(2)}(x))$ and then using the Chain rule. Comparing the last lines in (5.6) and (5.7), we see that $y_{n+1}^{(k)} = Y_{n+1}^{(k)} + O(h^3)$, which confirms that the order of the local truncation error in the Modified Euler method (5.5) is 3, and hence the method is second-order accurate.

In the homework problems, you will be asked to write out the forms of discretization schemes for the Midpoint and cRK methods for a system of two ODEs.

To conclude this subsection, we note that any higher-order IVP, say,

$$y''' + f(x, y, y', y'') = 0, \qquad y(x_0) = y_0, \quad y'(x_0) = z_0, \quad y''(x_0) = w_0\,, \qquad (5.8)$$

can be rewritten as a system of first-order ODEs with appropriate initial conditions:

$$\begin{aligned}
y_1' &= y_2\,, \\
y_2' &= y_3\,, \qquad (5.9)\\
y_3' &= -f(x, y_1, y_2, y_3)\,,
\end{aligned}$$

$$y_1(x_0) = y_0, \quad y_2(x_0) = z_0, \quad y_3(x_0) = w_0\,.$$

Above we have denoted $y = y_1$, and then y_2 and y_3 get defined by the first two equations in (5.9). To solve this system of three first-order ODEs, one can use any of the general-purpose discretization schemes considered above. Similarly, any higher-order ODE that can be explicitly solved for the highest derivative, can be dealt with along the same lines.

5.2 Special methods for the second-order ODE $y'' = f(y)$. I: Central-difference methods

A second-order ODE, along with the appropriate initial conditions:

$$y'' = f(y), \qquad y(x_0) = y_0, \quad y'(x_0) = y'_0\,, \tag{5.10}$$

occurs in applications quite frequently because it describes the motion of a Newtonian particle (i.e. a particle that obeys the laws of Newtonian mechanics) in the presence of a conservative force (i.e. a force that depends only on the position of the particle but not on its speed and/or the time). In the remainder of this subsection, it will be convenient to think of y as the position of the particle, of x — as the time, and of y' — as the particle's velocity.

The first special method that we introduce for Eq. (5.10) (and for systems of such equations) uses a *second-order accurate* approximation for y'':

$$y''_n = \frac{y_{n+1} - 2y_n + y_{n-1}}{h^2} + O(h^2)\,; \tag{5.11}$$

you encountered a similar formula in Lecture 3 (see Sec. 3.1). Combining Eqs. (5.10) and (5.11), we arrive at the central-difference method for Eq. (5.10):

$$Y_{n+1} - 2Y_n + Y_{n-1} \;=\; h^2 f_n\,. \tag{5.12}$$

(Method (5.12) is sometimes referred to as the *simple* central-difference method, because the r.h.s. of the ODE (5.10) enters it in the simplest possible way.) Since this is a two-step method, one needs two initial points to start it. The first point, Y_0, is simply the initial condition for the particle's position: $Y_0 = y_0$. The second point, Y_1, has to be determined from the initial position y_0 and the initial velocity y'_0. The natural **question** then is: To what accuracy should we determine Y_1 so as to be consistent with the accuracy of the method (5.12)?

To answer this question, we first show that the global error in the simple central-difference method is $O(h^2)$. Indeed, the local truncation error is $O(h^4)$, as it follows from (5.10)–(5.12). For numerical methods for a first-order ODE, considered earlier, this would imply that the global error must be $O(\frac{1}{h}) \cdot O(h^4) = O(h^3)$, since the local error of $O(h^4)$ would accumulate over $O(\frac{1}{h})$ steps. However, (5.10) is a *second-order* ODE, and for it, the error accumulates differently than for a first-order one. Below we will explain this qualitatively; for a rigorous derivation, the reader is referred to Appendix 1.

Consider the simplest case where the same error is made at every step, and all these errors simply add together. This can then be modeled by the following second-order "ODE" in the discrete variable n:

$$d^2(\text{GlobalError})/dn^2 = \text{LocalError}, \qquad \text{where} \quad \text{LocalError=const}; \tag{5.13}$$

$$\text{GlobalError}(0) = 0, \quad \text{GlobalError}\,'(0) = \text{StartupError}\,. \tag{5.14}$$

The "StartupError" is actually the error one makes in computing Y_1. If we now treat the discrete variable n as continuous (which is acceptable if we want to obtain an estimate for the answer), then the solution of the above is, obviously,

$$\text{GlobalError}(n) = \text{StartupError} \cdot n \;+\; \text{LocalError} \cdot \frac{n^2}{2}\,, \tag{5.15}$$

which on the account of

$$n \;=\; \frac{(b-a)}{h} \;=\; O\left(\frac{1}{h}\right), \qquad ([a,b] \text{ being the interval of integration})$$

becomes

$$\text{GlobalError}(n) = \text{StartupError} \cdot O\left(\frac{1}{h}\right) \;+\; \text{LocalError} \cdot O\left(\frac{1}{h^2}\right). \tag{5.16}$$

In Appendix 1, we derive an analog of (5.16) for the *discrete* equation (5.12) rather than for the continuous equation (5.13); that derivation confirms the validity of our replacing the discrete equation by its continuous equivalent for the purposes of the estimation of error accumulation.

Equation (5.16) along with the aforementioned fact that the local truncation error is $O(h^4)$ imply that the global error is indeed $O(h^2)$, provided that the "startup error" (i.e., the error in Y_1) is appropriately small. Using the same equation (5.16), it is now easy to see that Y_1 needs to be determined with accuracy $O(h^3)$. Therefore, we supplement Eq. (5.12) with the following

$$\text{Initial conditions for method (5.12):}$$
$$Y_0 = y_0, \qquad Y_1 = y_0 + hy_0' + \frac{h^2}{2} f(y_0)\,, \tag{5.17}$$

where in the last equation we have used the ODE $y'' = f(y)$.

Another method that uses the central-difference approximation (5.11) for y'' is:

$$Y_{n+1} - 2Y_n + Y_{n-1} \;=\; \frac{h^2}{12}\left(f_{n+1} + 10f_n + f_{n-1}\right). \tag{5.18}$$

This is called Numerov's method, or the Royal Road formula. The local truncation error of this method is $O(h^6)$. Therefore, the global error will be $O(h^4)$ (i.e., 2 orders better than the global error in the simple central-difference method), provided we calculate Y_1 with accuracy $O(h^5)$. In principle, this can be done using, for example, the Taylor expansion:

$$Y_1 = y_0 + hy_0' + \frac{h^2}{2}y_0'' + \frac{h^3}{6}y_0''' + \frac{h^4}{24}y_0^{(iv)}\,, \tag{5.19}$$

where y_0 and y_0' are given as the initial conditions and the higher-order derivatives are computed successively as follows:

$$\begin{aligned} y_0'' &= f(y_0), \\ y_0''' &= \tfrac{d}{dx} f(y)\big|_{y=y_0} \;=\; f_y(y_0)y_0'\,, \\ y_0^{(iv)} &= \tfrac{d}{dx}\left[f_y(y)\,y'(x)\right]\big|_{y=y_0} \;=\; f_{yy}(y_0)y_0' + f_y(y_0)f(y_0)\,. \end{aligned} \tag{5.20}$$

However, Numerov's method is *implicit* (why?), which makes it unpopular for numerical integration of the IVPs (5.10). The only exception would be the case when $f(y) = ay + b$, a linear function of y, when the equation for Y_{n+1} can be easily solved. We will encounter Numerov's method later in this course when we study *boundary* value problems; there, this method is the method of choice because of its high accuracy.

5.3 Special methods for the second-order ODE $y'' = f(y)$. II: Methods that approximately preserve energy

5.3.1 Analytical background

As we said at the beginning of the previous subsection, Eq. (5.10) describes the motion of a particle in the field of a conservative force. For example, the gravitational or electrostatic force is conservative, but any form of friction is not. We now rename the independent variable x as t (the time) and denote $v(t) = y'(t)$ (the velocity). As before, $y(t)$ denotes the particle's position. Equation (5.10) can be rewritten as

$$y' = v, \tag{5.21}$$
$$v' = f(y), \tag{5.22}$$
$$y(t_0) = y_0, \quad v(t_0) = v_0\,.$$

In Eq. (5.22), the r.h.s. can be thought of as a force acting on the particle of unit mass. Note that these equations admit a conserved quantity, called the *Hamiltonian* (which in Newtonian mechanics is just the total energy of the particle):

$$H(v,y) = \frac{1}{2}v^2 + U(y)\,, \qquad U(y) = -\int f(y)dy\,. \tag{5.23}$$

The first and second terms on the r.h.s. of (5.23) are the kinetic and potential energies of the particle. Using the equations of motion, (5.21) and (5.22), it is easy to see that the Hamiltonian (i.e., the total energy) is indeed conserved:

$$\frac{dH}{dt} = \frac{\partial H}{\partial v}\frac{dv}{dt} + \frac{\partial H}{\partial y}\frac{dy}{dt} = v\cdot f(y) + \frac{dU}{dy}\cdot v = 0 \quad \forall t\,. \tag{5.24}$$

It will be useful to refer to a simple model that is Hamiltonian. As such a model, we will use the equation of a simple harmonic oscillator

$$y'' = -\omega^2 y. \tag{5.25a}$$

Here $f(y) = -\omega^2 y$ and thus $U(y) = \omega^2 y^2/2$. As an initial condition for (5.25a) we will use

$$y(0) = 0, \quad y'(0) = v_0. \tag{5.25b}$$

The parameter ω is called the frequency of the oscillator.

Model (5.25a) arises in a great variety of applications where energy of the system is conserved (as one should expect from (5.24)). It is also a good first approximation for systems whose energy is nearly conserved. More specifically, this model describes small oscillations (not surprisingly!) when the system is given a small initial displacement or kick from its equilibrium state.

To analyze (5.25a), we write it in matrix form as follows:

$$\begin{pmatrix} \omega y \\ v \end{pmatrix}' = \begin{pmatrix} 0 & \omega \\ -\omega & 0 \end{pmatrix}\begin{pmatrix} \omega y \\ v \end{pmatrix}, \tag{5.26}$$

so that $y^{(1)} = \omega y$ and $y^{(2)} = v$. Note that using the factor ω in the definition of $y^{(1)}$ has made the matrix in (5.26), as well as subsequent formulas, more "symmetric-looking". The matrix in (5.26) has the eigenvalues $\lambda_{1,2} = \pm i\omega$; indeed:

$$\begin{vmatrix} 0-\lambda & \omega \\ -\omega & 0-\lambda \end{vmatrix} = 0 \qquad \Rightarrow$$
$$\lambda^2 + \omega^2 = 0 \qquad \Rightarrow$$
$$\lambda = \pm i\omega\,. \tag{5.27}$$

We will now demonstrate that Eq. (5.25a) does indeed describe an oscillating solution, which is to be expected of a model named 'harmonic oscillator'. First, from (5.26) and (5.27) one can state, using standard techniques from a course on differential equations, that

$$\omega y = C\, e^{i\omega x} + C^* e^{-i\omega x}, \tag{5.28}$$

where C is some (complex-valued) constant and C^* is its complex conjugate. (Such a relation between the coefficients in (5.28) ensures that this solution is real-valued.) Then, using the Euler formula for complex numbers,

$$e^{i\omega x} = \cos(\omega x) + i\, \sin(\omega x)\,,$$

in (5.28), one obtains

$$\omega y = B\, \sin(\omega x + \phi), \qquad v = B\, \cos(\omega x + \phi)\,, \tag{5.29}$$

where B are ϕ are some constants (related to constant C). It is now clear that solution (5.29) of Eq. (5.25a) indeed describes oscillations.

Finally, let us point out that the Hamiltonian (5.23) for the harmonic oscillator can be seen to be constant by a direct calculation using (5.29):

$$\frac{1}{2}v^2 + \frac{1}{2}(\omega y)^2 = \frac{1}{2}B^2. \tag{5.30}$$

One can interpret the last equation by saying that if the oscillator's coordinates $y^{(1)} = \omega y$ and $y^{(2)} = v$ are plotted in the $(y^{(1)}, y^{(2)})$-plane, they would form a circle. The oscillator's motion is then represented as a rotation around that circle. This will soon be illustrated in the next subsection.

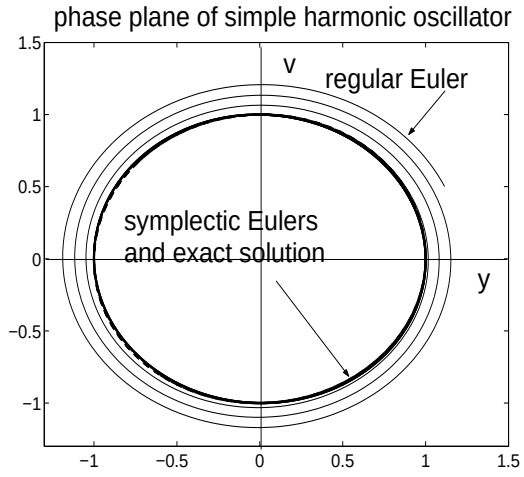
5.3.2 Numerical methods

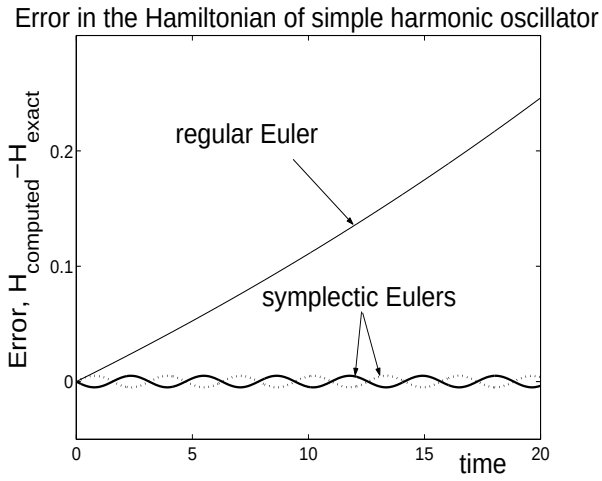
It is now natural to ask: Do any of the methods considered so far conserve the Hamiltonian? That is, if $\{Y_n,\, V_n\}$ is a numerical solution of (5.21) and (5.22), is $H(Y_n, V_n)$ independent of n? The answer is '**no**'. However, some of the methods do conserve the Hamiltonian *approximately* over very long time intervals. We now consider specific examples of such methods.

Consider the three implementations of the simple Euler method, given by Eqs. (5.2)–(5.4). We will refer to method (5.2) as the *regular* Euler method; the other two methods are conventionally referred to as *symplectic*[12] Euler methods. Let us apply these methods with $h = 0.02$ to integration of the simple harmonic oscillator model (5.25). For simplicity, in this section we set $\omega = 1$.

The results are presented below. We plot the numerical solutions along with the exact one in the phase plane for $t \le 20$, which corresponds to slightly more than 3 oscillation periods. The orbits of the solutions obtained by the symplectic methods lie very close to the orbit of the exact solution, while the orbit corresponding to the regular Euler method winds off into infinity (provided one waits infinitely long, of course).

[12] "Symplectic" is a term from Hamiltonian mechanics that means "preserving areas in the phase space". If this explanation does not make the matter clearer to you, simply ignore it and treat the word "symplectic" just as a new adjective in your vocabulary.
Perhaps surprisingly, systematic studies of symplectic methods began relatively recently, in the late 1980s. The theory behind these methods goes far beyond the scope of this course (and the expertise of this instructor). A review of such methods is posted on the website of this course.





At this point, we are ready to ask two more questions.

Question: What feature of the symplectic Euler methods allows them to maintain the Hamiltonian of the numerical solution near that of the exact solution?

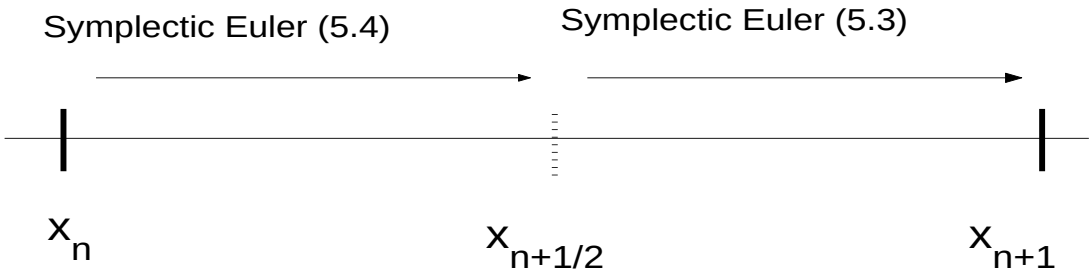
Answer: A short answer is 'stability'. We will provide more details on this in Sec. 5.4. Another point of view is based on the concept of a so-called *modified equation*. It is mentioned in the paper posted on the course website alongside this Lecture. While it is a useful concept, we do not have the time to cover it in this course.

Question: Among the methods we have considered in this Section, are there other methods that possess the property of near-conservation of the Hamiltonian?

Answer: A short answer is 'yes'. To present a more detailed answer, let us look back at the figure for the error in the Hamiltonian, obtained with the two symplectic Euler methods. We see that these errors are nearly opposite to each other and hence, being added, will nearly cancel one another. Therefore, if we somehow manage to combine the symplectic methods (5.3) and (5.4) so that the Hamiltonian error of the new method is the sum of those two "old" errors, then that new error will be dramatically reduced in comparison with either of the "old" errors. (This is similar to how the second-order trapezoidal rule for integration of $f(x)$ is obtained as the average of the first-order accurate left and right Riemann sums, whose errors are nearly opposite and thus, being added, nearly cancel each other.) Below we produce such a combination of methods (5.3) and (5.4).

Let us split the step from x_n to x_{n+1} into two substeps: from x_n to $x_{n+\frac{1}{2}}$ and then from $x_{n+\frac{1}{2}}$ to x_{n+1} (see the figure below). Let us now advance the solution in the first half-step using method (5.4) and then advance it in the second half-step using method (5.3). Here is this process in detail:

$$
\begin{aligned}
&\underline{x_n \quad \to \quad x_{n+\frac{1}{2}}, \quad \text{use (5.4)}} : \quad && V_{n+\frac{1}{2}} = V_n + \tfrac{h}{2} f(Y_n)\,, \\
& && Y_{n+\frac{1}{2}} = Y_n + \tfrac{h}{2} V_{n+\frac{1}{2}}\,, \\
&\underline{x_{n+\frac{1}{2}} \quad \to \quad x_{n+1}, \quad \text{use (5.3)}} : && Y_{n+1} = Y_{n+\frac{1}{2}} + \tfrac{h}{2} V_{n+\frac{1}{2}}\,, \\
& && V_{n+1} = V_{n+\frac{1}{2}} + \tfrac{h}{2} f(Y_{n+1})\,.
\end{aligned}
\tag{5.31}
$$



Combining the above equations (simply add the 2nd and 3rd equations, and then add the 1st and 4th ones), we obtain:

$$\begin{aligned} Y_{n+1} &= Y_n + hV_n + \tfrac{h^2}{2} f(Y_n)\,, \\ V_{n+1} &= V_n + \tfrac{h}{2}\left(f(Y_n) + f(Y_{n+1})\right)\,. \end{aligned} \tag{5.32}$$

Method (5.32) is called the Verlet method, after Loup Verlet, who "discovered" it in 1967. Later, however, Verlet himself found accounts of his method in works dated as far back as the late 18th century. In particular, in 1907, G. Störmer used higher-order versions of this method for computation of the motion of the ionized particles in the Earth's magnetic field. About 50 years earlier, J.F. Encke had used method (5.32) for computation of planetary orbits. For this reason, this method is also sometimes related with the names of Störmer and/or Encke.

The Verlet method is extensively used in applications dealing with long-time computations, such as molecular dynamics, planetary motion, and computer animation[13]. Its benefits are:
(i) It nearly conserves the energy of the modeled system;
(ii) It is second-order accurate; and
(iii) It requires only one function evaluation per step.

To make the value of these benefits evident, in a homework problem you will be asked to compare the performance of the Verlet method with that of the higher-order cRK method, which is not symplectic and does not have the property of near-conservation of energy.

We now complete the answer to the question asked about a page ago and show that the Verlet method is equivalent to the simple central-difference method. To this end, let us write the Verlet equations at two consecutive steps:

$$\begin{aligned} Y_{n+1} &= Y_n + hV_n + \tfrac{h^2}{2} f(Y_n)\,, \\ V_{n+1} &= V_n + \tfrac{h}{2}\left(f(Y_n) + f(Y_{n+1})\right)\,, \\ Y_{n+2} &= Y_{n+1} + hV_{n+1} + \tfrac{h^2}{2} f(Y_{n+1})\,, \\ V_{n+2} &= V_{n+1} + \tfrac{h}{2}\left(f(Y_{n+1}) + f(Y_{n+2})\right)\,. \end{aligned} \tag{5.33}$$

In fact, we will only need the first three of the above equations. Subtracting the 1st equation from the 3rd and slightly rearranging the terms, we obtain:

$$Y_{n+2} - 2Y_{n+1} + Y_n = \left\{ hV_{n+1} + \frac{h^2}{2} f(Y_{n+1}) \right\} - \left\{ hV_n + \frac{h^2}{2} f(Y_n) \right\}\,. \tag{5.34}$$

[13]For example, you may visit a game-developers' website at http://www.gamedev.net, go to their Forums and there do a search for 'Verlet'.

We now use the 2nd equation of (5.33) to eliminate V_{n+1}. The straightforward calculation yields

$$Y_{n+2} - 2Y_{n+1} + Y_n = h^2 f(Y_{n+1}),$$

which is the simple central-difference method (5.12). Thus, we have shown that the simple central-difference method nearly conserves the energy of the system.

To conclude this section, we note that although the Verlet method nearly conserves the Hamiltonian of the simulated system, it may not always conserve or nearly-conserve other constants of the motion, whenever such exist. As an example, consider the Kepler two-body problem (two particles in each other's gravitational field):

$$q'' = -\frac{q}{(q^2+r^2)^{3/2}}, \qquad r'' = -\frac{r}{(q^2+r^2)^{3/2}}, \tag{5.35}$$

where q and r are the Cartesian coordinates of a certain radius vector relative to the center of mass of the particles. Let us denote the velocities corresponding to q and r as Q and R, respectively. This problem has the following three constants of the motion:

Hamiltonian of (5.35):

$$H = \frac{1}{2}(Q^2+R^2) - \frac{1}{\sqrt{q^2+r^2}}\,, \tag{5.36}$$

Angular momentum of (5.35):

$$A = qR - rQ\,, \tag{5.37}$$

Runge–Lenz vector of (5.35):

$$L = \vec{\mathbf{i}}\left(R(qR-rQ) - \frac{q}{\sqrt{q^2+r^2}}\right) + \vec{\mathbf{j}}\left(-Q(qR-rQ) - \frac{r}{\sqrt{q^2+r^2}}\right). \tag{5.38}$$

It turns out that the Verlet method nearly conserves the Hamiltonian and exactly conserves the angular momentum A, but does not conserve the Runge–Lenz vector L. In a homework problem, you will be asked to examine what effect this nonconservation has on the numerical solution.

5.4 Stability of numerical methods for systems of ODEs and higher-order ODEs

Following the lines of the previous three subsections, we will first comment on the stability of general-purpose methods, and then on that of special methods for $y'' = f(y)$ and similar equations.

In Lecture 4, we showed that in order to analyze stability of numerical methods for a single first-order ODE $y' = f(x,y)$, we needed to consider that stability for the model problem (4.15), $y' = \lambda y$ with $\lambda = \text{const}$. The motivation for this was given in Sec. 4.2. Namely, we related the stability concept with the deviation $(y-u)$ between two nearby solutions y and u of the ODE in question, and this deviation satisfies Eq. (4.16), which we re-state here for the reader's convenience:

$$(y-u)' \approx f_y(x,y)\cdot(y-u). \tag{5.39}$$

Replacing the variable coefficient $f_y(x,y)$ with a constant λ, one arrives at the model problem (4.15). Recall that when we analyze the stability of a particular numerical method, we need to

keep into account what range of values f_y, and hence λ, can take on. Thus, as we stated earlier, the *stability of the numerical method depends on the ODE that it is applied to*. We will see below that the same statement also pertains to a system of ODEs.

Question: What is the counterpart of (5.39) for a system of ODEs?

Answer, stated for two ODEs:

$$(\vec{y}-\vec{u})' = \frac{\partial(f^{(1)},f^{(2)})}{\partial(y^{(1)},y^{(2)})}\ (\vec{y}-\vec{u})\,, \tag{5.40}$$

where

$$\vec{y} = \begin{pmatrix} y^{(1)} \\ y^{(2)} \end{pmatrix}, \qquad \vec{u} = \begin{pmatrix} u^{(1)} \\ u^{(2)} \end{pmatrix}, \qquad \frac{\partial(f^{(1)},f^{(2)})}{\partial(y^{(1)},y^{(2)})} = \begin{pmatrix} \frac{\partial f^{(1)}}{\partial y^{(1)}} & \frac{\partial f^{(1)}}{\partial y^{(2)}} \\ \frac{\partial f^{(2)}}{\partial y^{(1)}} & \frac{\partial f^{(2)}}{\partial y^{(2)}} \end{pmatrix}. \tag{5.41}$$

The last matrix is called the Jacobian of the r.h.s. of system (5.1). Equations (5.40) and (5.41) generalize straightforwardly for more than two equations.

We now give a brief derivation of Eqs. (5.40) and (5.41) which parallels that of Eq. (4.16) for a single first-order ODE. For convenience of the reader, we re-state here the ODEs from (5.1):

$$\begin{aligned} y^{(1)\prime} &= f^{(1)}(x,y^{(1)},y^{(2)}), & u^{(1)\prime} &= f^{(1)}(x,u^{(1)},u^{(2)}), \\ y^{(2)\prime} &= f^{(2)}(x,y^{(1)},y^{(2)}), & u^{(2)\prime} &= f^{(2)}(x,u^{(1)},u^{(2)}). \end{aligned} \tag{5.42}$$

We subtract the second equation in the kth line ($k=1$ or 2) of (5.42) from the first equation in the same line and obtain:

$$\begin{aligned} (y^{(k)}-u^{(k)})' &= f^{(k)}(x,y^{(1)},y^{(2)}) - f^{(k)}(x,u^{(1)},u^{(2)}) \\ &= \frac{\partial f^{(k)}}{\partial y^{(1)}}\,(y^{(1)}-u^{(1)}) + \frac{\partial f^{(k)}}{\partial y^{(2)}}\,(y^{(2)}-u^{(2)}) + O(\epsilon^2)\,. \end{aligned} \tag{5.43}$$

It is now easy to see that Eq. (5.43) is the same as Eqs. (5.40) and (5.41).

Since we do not know the values of the entries in the Jacobian matrix in Eq. (5.40), we simply replace that matrix by a matrix with constant terms. Thus, the model problem that one should use to analyze stability of numerical methods for system of ODEs is

$$\vec{\mathbf{y}}' = A\vec{\mathbf{y}}\,, \qquad A \text{ is a constant matrix.} \tag{5.44}$$

Now, for a single first-order ODE, we had only one parameter, λ, in the model problem.

Question: How many parameters do we have in the model problem (5.44) for a system of N ODEs?

The **answer** depends on which of the two categories of methods one uses. Namely, in Sec. 5.1, we saw that some methods (e.g., the regular Euler (5.2) and the Modified Euler (5.5)) use the solution $\vec{\mathbf{Y}}_n$ at $x=x_n$ to *simultaneously* advance to the next step, $x=x_{n+1}$. Moreover, each component $Y^{(k)}$ is obtained *using the same discretization rule*. To be consistent with the terminology of Sec. 5.1, we will call this first category of methods, the *general purpose methods*. Methods of the other category, which included the symplectic Euler and Verlet, obtain a component $Y_{n+1}^{(m)}$ at x_{n+1} by using previously obtained components at x_{n+1}, $Y_{n+1}^{(k)}$ with $k<m$, as well as the components $Y_n^{(p)}$ with $p\geq m$ at x_n. In other words, they apply *different discretization rules for different components*. We will call methods from this category, *special methods*.

Returning to the above question, we will show below that for the general-purpose methods (regular Euler, modified Euler, cRK, etc.), the answer is 'N' (even though matrix A contains N^2 entries!). We will explain this using the regular Euler method as an example. Details for

other general purpose methods are more involved, but follow the same logic. First, we will present a *diagonalization* process (see below) for the model problem (5.44). We will then show that this process straightforwardly extends to general-purpose numerical methods, but not to special methods.

Most matrices, or at least those that we will encounter in this course, are diagonalizable[14]. This means that there exists a matrix S and a diagonal matrix D such that

$$A = S^{-1} D S\,; \qquad D = \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_N)\,. \tag{5.45}$$

Moreover, the diagonal entries of D are the eigenvalues of A. Substitution of (5.45) into (5.44) leads to the following sequence of transformations:

$$\begin{aligned}
\vec{\mathbf{y}}\,' &= A\vec{\mathbf{y}} \qquad \Rightarrow \\
\vec{\mathbf{y}}\,' &= S^{-1} D\, S\vec{\mathbf{y}} \qquad \Rightarrow \\
S\vec{\mathbf{y}}\,' &= D\, S\vec{\mathbf{y}} \qquad \Rightarrow \\
(S\vec{\mathbf{y}})' &= D\,(S\vec{\mathbf{y}}) \qquad \Rightarrow \\
\vec{\mathbf{z}}\,' &= D\,\vec{\mathbf{z}}, \qquad \text{where} \quad \vec{\mathbf{z}} = S\vec{\mathbf{y}}\,.
\end{aligned} \tag{5.46}$$

Therefore, the important (for the stability analysis) information about a diagonalizable matrix A is concentrated in its eigenvalues. Namely, given the diagonal form (5.45) of matrix D, the last equation in (5.46) can be written as

$$z^{(k)\,\prime} = \lambda^{(k)} z^{(k)}\,, \qquad \text{for} \quad k = 1, \ldots N\,, \tag{5.47}$$

which means that the matrix model problem (5.44) reduces to the model problem (4.15) for a single first-order ODE.

Now, when we apply the regular Euler method to system (5.44), we get

$$\vec{\mathbf{Y}}_{n+1} - \vec{\mathbf{Y}}_n = hA\vec{\mathbf{Y}}_n\,. \tag{5.48}$$

Repeating now the steps of (5.46), we rewrite this as

$$\vec{\mathbf{Z}}_{n+1} - \vec{\mathbf{Z}}_n = hD\vec{\mathbf{Z}}_n\,, \tag{5.49}$$

where $\vec{\mathbf{Z}}_n$ is the numerical approximation to $\vec{\mathbf{z}}_n$. Given the diagonal form of D, for the components of $\vec{\mathbf{Z}}_n$ we obtain:

$$Z^{(k)}_{n+1} - Z^{(k)}_n = h\lambda^{(k)} Z^{(k)}_n\,, \tag{5.50}$$

which is just the simple Euler method applied separately to individual model problems (5.47). Thus, we have confirmed our earlier statement that for a general purpose method, the stability analysis for a system of ODEs reduces to the stability analysis for a single equation.

We will now show that the above statement does *not* apply to special methods like the symplectic Euler etc.. Indeed, let us apply symplectic Euler (5.3) to a 2×2 model problem (5.44), assuming that $A = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}$. We have (verify):

$$\vec{\mathbf{Y}}_{n+1} - \vec{\mathbf{Y}}_n = h \begin{pmatrix} a_{11} & a_{12} \\ 0 & a_{22} \end{pmatrix} \vec{\mathbf{Y}}_n + h \begin{pmatrix} 0 & 0 \\ a_{21} & 0 \end{pmatrix} \vec{\mathbf{Y}}_{n+1}\,. \tag{5.51}$$

[14] Some matrices, e.g. $\begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$, are not diagonalizable. However, if we perturb it as, say, $\begin{pmatrix} 1.01 & 1 \\ 0 & 1 \end{pmatrix}$, this latter matrix *is* diagonalizable.

This can no longer be written in the form (5.48), and hence the subsequent calculations that led to (5.50) are no longer valid. Therefore, the only venue to proceed with the stability analysis for special methods is to consider the original matrix model problem (5.44); obviously, this model problem has, in general, as many parameters as the matrix A, i.e. N^2.

Below we give an example of doing stability analyses for the regular and symplectic Euler methods. In a homework problem, you will be asked to do similar calculations for the modified Euler and Verlet methods. As a model problem, we choose that of a simple harmonic oscillator (5.25), whose solution was derived at the end of Sec. 5.3.1.

If we use the regular Euler, which is a general purpose method, then according to the discussion that led to Eq. (5.50), it suffices to study stability of the simple Euler method for a single ODE

$$y' = \lambda y \qquad \text{with } \lambda = i\omega \text{ or } \lambda = -i\omega, \tag{5.52}$$

as these are the eigenvalues of the simple harmonic oscillator; see (5.27). Equation (4.20) of Lecture 4 yielded the following:

$$|\rho| = |1 + h \cdot i| = \sqrt{1 + h^2} \approx 1 + \frac{1}{2}h^2\,, \tag{5.53}$$

so that

$$|Y_n| \propto |\rho|^n \;=\; |\rho|^{(x/h)} \;\approx\; (1 + \frac{1}{2}h^2)^{(x/h)} \;=\; (1 + h \cdot \frac{1}{2}h)^{(x/h)} \;\approx\; e^{xh/2}\,. \tag{5.54}$$

Since the absolute value $|\rho|^n$ determines the amplitude of the numerical solution, we see that Eq. (5.54) shows that this amplitude *grows* with x, whereas the amplitude of the exact solution of (5.26) is constant (see (5.29))[15]. Therefore, the regular Euler method applied to (5.26) is *unstable* for any step size h!

This result is corroborated by the figure accompanying Eq. (4.20). Namely, for $\lambda = i$ or $-i$ (recall that in Sec. 5.3.2 we set $\omega = 1$), the value $h\lambda$ lies on the imaginary axis, which is *outside* the stability region for the simple Euler method. Since the magnitude of any error will then grow exponentially, so will the amplitude of the solution, because, as we discussed in Lecture 4, for linear equations the error and the solution satisfy the same equation. In a homework problem, you will be asked to show that the behavior of the Hamiltonian of the numerical solution shown in a figure in Sec. 5.3 quantitatively agrees with Eq. (5.54).

Now we turn to the stability analysis of the symplectic Euler method (say, (5.3)). To that end, we will apply these finite-difference equations to our model problem $y'' = -\omega^2 y$ written in the form (5.26):

$$\begin{pmatrix} \omega y \\ v \end{pmatrix}' = \begin{pmatrix} 0 & \omega \\ -\omega & 0 \end{pmatrix} \begin{pmatrix} \omega y \\ v \end{pmatrix}. \tag{5.26}$$

The finite-difference equations are:

[15] To better visualize what is going on, you may imagine that the numerical solution at each x_n is simply the exact solution multiplied by $|\rho|^n$.

$$
\begin{array}{rcl}
\omega Y_{n+1} &=& \omega Y_n + (h\omega) V_n \\
V_{n+1} &=& V_n - (h\omega)\, \omega Y_{n+1}
\end{array}
\quad \Rightarrow
$$

$$
\begin{pmatrix} 1 & 0 \\ h\omega & 1 \end{pmatrix} \begin{pmatrix} \omega Y \\ V \end{pmatrix}_{n+1} = \begin{pmatrix} 1 & h\omega \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \omega Y \\ V \end{pmatrix}_{n} \quad \Rightarrow
$$

$$
\begin{pmatrix} \omega Y \\ V \end{pmatrix}_{n+1} = \begin{pmatrix} 1 & 0 \\ h\omega & 1 \end{pmatrix}^{-1} \begin{pmatrix} 1 & h\omega \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \omega Y \\ V \end{pmatrix}_{n} \quad \Rightarrow
$$

$$
\begin{pmatrix} \omega Y \\ V \end{pmatrix}_{n+1} = \begin{pmatrix} 1 & 0 \\ -h\omega & 1 \end{pmatrix} \begin{pmatrix} 1 & h\omega \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \omega Y \\ V \end{pmatrix}_{n} \quad \Rightarrow
$$

$$
\begin{pmatrix} \omega Y \\ V \end{pmatrix}_{n+1} = \begin{pmatrix} 1 & h\omega \\ -h\omega & 1-(h\omega)^2 \end{pmatrix} \begin{pmatrix} \omega Y \\ V \end{pmatrix}_{n} . \tag{5.55}
$$

Once we have obtained this matrix relation between the solutions at the nth and $(n+1)$st steps, we need to obtain the eigenvalues of the matrix on the r.h.s. of (5.55). Indeed, it is known from Linear Algebra that the solution of (5.55) is

$$
\begin{pmatrix} \omega Y \\ V \end{pmatrix}_{n} = \vec{\mathbf{u}}_1\, \rho_1^n + \vec{\mathbf{u}}_2\, \rho_2^n , \tag{5.56}
$$

where $\rho_{1,2}$ are the eigenvalues of the matrix in question and $\vec{\mathbf{u}}_{1,2}$ are the corresponding eigenvectors. (We have used the notation $\rho_{1,2}$ instead of $\lambda_{1,2}$ for the eigenvalues in order to emphasize the connection with the characteristic root ρ that arises in the stability analysis of a single ODE.) If we find that the modulus of either of the eigenvalues ρ_1 or ρ_2 exceeds 1, this would mean that the symplectic method is unstable (well, we know already that it is not, but we need to demonstrate that). A simple calculation similar to that found after Eq. (5.26) yields

$$
\rho_{1,2} \,=\, 1 - \frac{1}{2}\left(h^2\omega^2 \pm \sqrt{h^4\omega^4 - 4h^2\omega^2} \right) . \tag{5.57}
$$

With some help from *Mathematica*, one can show that

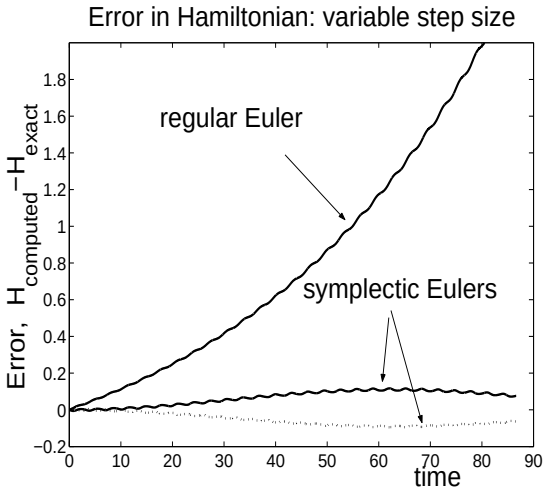
$$
\begin{array}{ll}
|\rho_1| = |\rho_2| = 1 & \text{for } \; -2 \le h\omega \le 2 \\
\text{either } |\rho_1| > 1 \text{ or } |\rho_2| > 1, & \text{for any other complex } h\omega\,.
\end{array} \tag{5.58}
$$

Thus, the symplectic Euler method is stable for the simple harmonic oscillator equation (and, in general, other oscillatory models), provided that h is sufficiently small, so that $|h\omega| < 2$. Note that ω in Eq. (5.25a) is just λ_I (i.e., $\mathrm{Im}(\lambda)$), where λ is the coefficient in the model equation (4.15): see (5.26) and (5.27). Using this relation between λ and ω, we then observe that the stability region for the symplectic Euler method, given by the first line of (5.58), is reminescent of the stability region of the Leap-frog method (see Eq. (4.35)). This may suggest that the Leap-frog method, applied to an oscillatory equation, will also have the property of near-conservation of the total energy. While this is indeed so, we will not consider this issue here.

The symplectic Euler (and higher-order symplectic) methods may lose their remarkable property of near-conservation of energy if the step size is varied. This is discussed, e.g., in the article by J. Dummer posted alongside this Lecture. An explanation of this fact was given by Robert Skeel in 1993; his paper is also posted on the course website. In Appendix 2 we summarize the

main idea of his explanation.

However, in practice, a varying step size does *not* always cause the symplectic Euler method to become unstable. To illustrate this fact, we show the error in the Hamiltonian obtained for the same Eq. (5.25) as in Sec. 5.3 when the step size is sinusoidally varied with a frequency incommensurable with that of the oscillator itself. Specifically, we took $h = 0.02+0.01\sin(1.95t)$. We see, however, that the error in the symplectic methods is still much smaller than that obtained by the regular Euler method.



To conclude this subsection, let us mention that for the simple central-difference method (5.12) and Numerov's method (5.18), the stability analysis should also be applied to the model equation (5.25a). For example, substituting $Y_n = \rho^n$ into the simple central-difference equation, where $f(y) = -\omega^2 y$, one finds

$$\rho^2 - (2 - h^2\omega^2)\rho + 1 = 0\,. \tag{5.59}$$

Again, with the help from *Mathematica*, one can show that the two roots $\rho_{1,2}$ of Eq. (5.59) satisfy Eq. (5.58). This is not at all surprising, given that the simple central-difference method is equivalent to the Verlet method (see the text around (5.33)) and the latter, in its turn, is simply a composition of two symplectic Euler methods.

Similarly, one can show that the stability region for Numerov's method is given by

$$-\sqrt{6} \le h\omega \le \sqrt{6}\,, \qquad \text{where} \quad |\rho_1| = |\rho_2| = 1\,, \tag{5.60}$$

whereas for any other complex $h\omega$, either $|\rho_1| > 1$ or $|\rho_2| > 1$.

5.5 Stiff equations

Here we will encounter, for the first time in this course, a class of equations that are very difficult to solve numerically. These equations are called numerically *stiff*. It is important to be able to recognize cases where one has to deal with such systems of equation; otherwise, the numerical solution that one would obtain will have no connection to the exact one.

Let us consider an IVP

$$\begin{pmatrix} u \\ v \end{pmatrix}' = \begin{pmatrix} 998 & 1998 \\ -999 & -1999 \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix}, \qquad \begin{pmatrix} u \\ v \end{pmatrix}\Big|_{x=0} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}. \tag{5.61}$$

Its exact solution is

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} 2 \\ -1 \end{pmatrix} e^{-x} + \begin{pmatrix} -1 \\ 1 \end{pmatrix} e^{-1000x}\,. \tag{5.62}$$

The IVP (5.61) is an example of a stiff equation. Although there is no rigorous definition of numerical stiffness, it is often accepted that a stiff system should satisfy the following two criteria: (i) The system of ODEs must contain at least two groups of solutions, where solutions in one

group vary rapidly relatively to the solutions of the other group. That is, among the eigenvalues of the corresponding matrix A there must be two, $\lambda^{\rm slow}$ and $\lambda^{\rm rapid}$, such that

$$\frac{|\mathrm{Re}\,\lambda^{\rm rapid}|}{|\lambda^{\rm slow}|} \gg 1\,. \tag{5.63}$$

(ii) The rapidly changing solution(s) must be stable. That is, the *large in magnitude* eigenvalues, $\lambda^{\rm rapid}$, of the matrix A in Eq. (5.44) must have $\mathrm{Re}\,\lambda^{\rm rapid} < 0$. As for the slowly changing solutions, they may be either stable or unstable.

Let us verify that system (5.61) is stiff. Indeed, criterion (i) above is satisfied for this system because of its two solutions, given by the two terms in (5.62), the first (with $\lambda^{\rm slow} = -1$) varies slowly compared to the other term (with $\lambda^{\rm rapid} = -1000$). Criterion (ii) is satisfied because the rapidly changing solution has $\lambda^{\rm rapid} < 0$.

Another example of a stiff system is

$$\begin{pmatrix} u \\ v \end{pmatrix}' = -\begin{pmatrix} 499 & 501 \\ 501 & 499 \end{pmatrix}\begin{pmatrix} u \\ v \end{pmatrix}, \qquad \begin{pmatrix} u \\ v \end{pmatrix}\Big|_{x=0} = \begin{pmatrix} 0 \\ 2 \end{pmatrix}, \tag{5.64}$$

whose solution is

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} -1 \\ 1 \end{pmatrix} e^{2x} + \begin{pmatrix} 1 \\ 1 \end{pmatrix} e^{-1000x}\,. \tag{5.65}$$

Here, again, the first and second terms in (5.65) represent the slow and fast parts of the solution, with $\lambda^{\rm slow} = 2$ and $\lambda^{\rm rapid} = -1000$, so that $|\lambda^{\rm rapid}| \gg |\lambda^{\rm slow}|$. Thus, criterion (i) is satisfied. Criterion (ii) is satisfied because the rapid solution is stable: $\lambda^{\rm rapid} < 0$.

The difficulty with stiff equations can be understood from the above examples (5.61), (5.62) and (5.64), (5.65). Namely, the rapid parts of those solutions are important only very close to $x = 0$ and are almost zero everywhere else. However, in order to integrate, e.g., (5.61) using, say, the simple Euler method, one would require to keep $h \cdot 1000 \le 2$ (see Eq. (4.19)), i.e. $h \le 0.002$. That is, we are forced to use a very small step size in order to avoid the numerical instability caused by *the least important part of the solution*!

Thus, in layman terms, a problem that involves processes evolving on two (or more) disparate scales, with the rapid process(es) being stable, is stiff. Moreover, as the above example shows, the meaning of stiffness is that one needs to work the hardest (i.e., use the smallest h) to resolve the least important part of the solution (i.e., the second terms on the r.h.s.'es of (5.62) and (5.65)).

An obvious way to deal with a stiff equation is to use an A-stable method (implicit or modified implicit Euler). This would eliminate the issue of numerical instability; however, the problem of (low) accuracy will still remain.

In practice, one strikes a compromise between the accuracy and stability of the method. Matlab, for example, uses a family of methods known as BDF (backward-difference formula) methods. Matlab's built-in solvers for stiff problems are `ode15s` (this uses a method of order between 1 and 5) and `ode23s`.

5.6 Appendix 1: Derivation of Eq. (5.12) with $f_n = \mathrm{const}$

Here we will derive the solution of Eq. (5.12) with its right-hand side being replaced by a constant:

$$Y_{n+1} - 2Y_n + Y_{n-1} = M, \qquad M = \mathrm{const}. \tag{5.66}$$

This will provide a rigorous justification for the solution (5.15) of the system (5.13)–(5.14) with StartupError= 0. We will indicate at the end how this derivation can be extended for StartupError$\neq$ 0.

The method that we will use closely follows the lines of the method of variation of parameters for the second-order ODE

$$y'' + By' + Cy = F(x). \tag{5.67}$$

In what follows we will refer to Eq. (5.67) as the *continuous case*. Namely, we first obtain the solutions of the homogeneous version of (5.66):

$$Y_n = c^{(1)} + c^{(2)} n\,, \qquad c^{(1)} \text{ and } c^{(2)} \text{ are arbitrary constants.} \tag{5.68}$$

Solution (5.68) was obtained by the substitution into (5.66) with $M = 0$ of the *ansätze* $Y_n = \rho^n$ and $Y_n = n\rho^n$. This is analogous to how the solution $y = c^{(1)} + c^{(2)}x$ of the ODE $y'' = 0$ is obtained.

Next, to solve Eq. (5.66) with $M \neq 0$, we allow the constants $c^{(1)}$ and $c^{(2)}$ to depend on n. Substituting the result into (5.66), we obtain:

$$\left(c^{(1)}_{n+1} - 2c^{(1)}_n + c^{(1)}_{n-1} \right) + \left((n+1)c^{(2)}_{n+1} - 2nc^{(2)}_n + (n-1)c^{(2)}_{n-1} \right) = M. \tag{5.69}$$

Now, similarly to how in the continuous case the counterparts of our $c^{(1)}$ and $c^{(2)}$ are *set* to satisfy an equation

$$\left(c^{(1)}\right)' y^{(1)} + \left(c^{(2)}\right)' y^{(2)} = 0,$$

where $y^{(k)}$, $k = 1, 2$ are the homogeneous solutions of (5.67), here we impose the following condition:

$$\underline{k = n:} \qquad \left(c^{(1)}_{k+1} - c^{(1)}_k \right) + k \left(c^{(2)}_{k+1} - c^{(2)}_k \right) = 0. \tag{5.70}$$

Subtracting from (5.70) its counterpart for $k = n - 1$, one obtains:

$$\left(c^{(1)}_{n+1} - 2c^{(1)}_n + c^{(1)}_{n-1} \right) + \left(n\, c^{(2)}_{n+1} - (2n-1)c^{(2)}_n + (n-1)c^{(2)}_{n-1} \right) = 0. \tag{5.71}$$

Next, subtracting the last equation from (5.69), we obtain a recurrence equation for $c^{(2)}$ only, which has a simple solution (assuming $c^{(2)}_0 = 0$):

$$c^{(2)}_{n+1} - c^{(2)}_n = M, \quad \Rightarrow \quad c^{(2)}_n = nM. \tag{5.72}$$

From (5.72) and (5.70) one obtains the solution for $c^{(1)}$:

$$c^{(1)}_{n+1} - c^{(1)}_n = -nM, \quad \Rightarrow \quad c^{(1)}_n = -\frac{n(n-1)}{2}\, M. \tag{5.73}$$

(Again, we have assumed that $c^{(1)}_0 = 0$.) Finally, combining the results of (5.68), (5.72), and (5.73), we obtain the solution of Eq. (5.66):

$$Y_n = -\frac{n(n-1)}{2} M + n^2 M = \frac{n(n+1)}{2} M = O(n^2)\, M. \tag{5.74}$$

The leading-order dependence on n of this solution is that corresponding to the LocalError term in formula (5.15).

To obtain the StartupError term in that formula, one needs to reconsider the initial conditions for $c^{(1)}_0$ and $c^{(2)}_0$. Let StartupError= S. This means that we now have

$$Y_0 = 0 \quad \text{(as before)}, \qquad Y_1 = S, \tag{5.75}$$

which says that our error at the initial node, $n = 0$, is still zero, but at the first computed node, $n = 1$, it is S. Next, for arbitrary $c_0^{(1)}$ and $c_0^{(2)}$, expressions in (5.72) and (5.73) generalize straightforwardly to

$$c_n^{(2)} = c_0^{(2)} + nM, \qquad c_n^{(1)} = c_0^{(1)} - \frac{n(n-1)}{2}\, M. \tag{5.76}$$

Substituting this along with (5.68) into (5.75), one finds

$$c_0^{(1)} = 0, \qquad c_0^{(2)} = S - M. \tag{5.77}$$

Finally, substituting (5.77) into (5.76) and then the result into (5.68), one obtains

$$Y_n = nS + \frac{n(n-1)}{2} M, \tag{5.78}$$

which agrees with (5.15).

5.7 Appendix 2: Symplectic methods with a variable step size

Consider the symplectic Euler method performed with two alternating step sizes, h_1 and h_2. The corresponding relations follow from (5.55):

$$\begin{pmatrix} \omega Y \\ V \end{pmatrix}_{n+1} = M_1 \begin{pmatrix} \omega Y \\ V \end{pmatrix}_n, \qquad \begin{pmatrix} \omega Y \\ V \end{pmatrix}_{n+2} = M_2 \begin{pmatrix} \omega Y \\ V \end{pmatrix}_{n+1}. \tag{5.79a}$$

$$M_k = \begin{pmatrix} 1 & h_k\omega \\ -h_k\omega & 1-(h_k\omega)^2 \end{pmatrix}, \qquad k = 1, 2. \tag{5.79b}$$

Thus,

$$\begin{pmatrix} \omega Y \\ V \end{pmatrix}_{n+2} = M_2 M_1 \begin{pmatrix} \omega Y \\ V \end{pmatrix}_n, \tag{5.80}$$

which means that the matrix whose eigenvalues determine the behavior of the numerical solution (see the text after (5.55)) is (M_2M_1). These eigenvalues *would have been* the products of the corresponding eigenvalues (5.57) computed with $h = h_1$ and $h = h_2$ if M_1 and M_2 *had commuted*. However, these matrices do not commute (as one can straightforwardly verify). Hence the only way to compute the eigenvalues of M_2M_1 is by direct calculation. In general, such calculations for the Verlet method (which, as you may recall, is just a superposition of two symplectic Euler methods) are performed in a paper "Variable step size destabilizes the Störmer/Leapfrog/Verlet method" by R. Skeel, which is posted on the course webpage.

Here is a numeric example illustrating the above statement. Take h_1 and h_2 such that $h_1\omega = 3/4$ and $h_2\omega = 7/4$. Both values are within the stability region of the symplectic Euler method, according to (5.58). However, the eigenvalues of M_2M_1 are $(-647 \pm 3\sqrt{17385})/512$, and the more negative of them has modulus that is slightly greater than 2. Thus, the symplectic Euler method with a variable step size implemented as per (5.79) with the $h_{1,2}\omega$ values as above not only will not conserve the energy, but also will be strongly unstable.

5.8 Questions for self-assessment

1. Verify (5.6).
2. Verify (5.7).

3. What would be your first step to solve a 5th-order ODE using the methodology of Sec. 5.1?

4. Use Eq. (5.11) to explain why the local truncation error of the simple central-difference method (5.12) is $O(h^4)$.

5. Explain why Y_1 for that method needs to be calculated with accuracy $O(h^3)$.

6. What is the global error of the simple central-difference method?

7. How does the rate of the error accumulation (with the number of steps) for a second-order ODE differ from the rate of the error accumulation for a first-order ODE?

8. Explain the last term in the expression for Y_1 in (5.17).

9. Why is Numerov's method implicit?

10. What is the physical meaning of the Hamiltonian for a Newtonian particle?

11. Verify (5.24).

12. What is the advantage of the symplectic Euler methods over the regular Euler method?

13. State the observation that prompted us to combine the two symplectic Euler methods into the Verlet method.

14. Obtain (5.32) from (5.31).

15. Obtain (5.34) and the next (unnumbered) equation.

16. What is the model problem for the stability analysis for a system of ODEs?

17. Show that for a "general-purpose" method, the stability analysis for a system of ODEs reduces to the stability analysis for the model problem (4.15).

18. Why is this not so for the "special" methods, like the symplectic Euler?

19. Make sure you can follow (5.55).

20. Verify that (5.56) is the solution of (5.55). That is, substitute (5.56), with the corresponding subindices, into both sides of the last equation of (5.55). Then for the expression on the r.h.s., use the stated fact that $\vec{\mathbf{u}}_1$ and $\vec{\mathbf{u}}_2$ are the eigenvectors of the matrix appearing on the r.h.s. of that equation. (You do not need to use the explicit form of that matrix.)

21. Obtain (5.57).

22. Would you apply the Verlet method to the following *strongly* damped oscillator:

$$y'' = -(2+i)^2 y\,?$$

Please explain.

23. Same question for Numerov's method.

24. What is the numerical stiffness, in layman terms?