Lecture 1

Solving time dependent problems Edit

Prognostic models are integrated forward in time starting from some specified initial conditions. Although ideally both time and space should be considered together, we will describe how to solve time-dependent problems in general either ignoring the spatial dimensions or assuming perfect differentiation in space. This approach of treating time and space dimensions separately is formerly known as the "method of lines".

Consider how to solve the equation

$$\partial_t u = g(u(t)) \tag{1.1}$$

given sufficient initial conditions $u(t = t_o) = u_o$. Assuming a constant time step Δt, then time is $t = n\Delta t$ so that n is the time step number. Imagine that we know all values of the problem up to and including time level n and we wish to integrate forward in time to time level $n + 1$. The system can be integrated between $t = n\Delta t$ and $t = (n + 1)\Delta t$ yielding:

$$u^{n+1} - u^n = \int_{n\Delta t}^{(n+1)\Delta t} g(u(t))dt \tag{1.2}$$

The problem becomes one of approximating the integral on the right hand side and there are three basic approaches to making this approximation:

1. Implicit methods which express the integral in terms of both knowns, $u^n, u^{n-1}, \ldots$ and unknowns, $u^{n+1}, u^{n+2}, \ldots$. For example, the backward and trapezoidal methods.

2. Single stage, explicit methods which express the integral only in terms of knowns, $u^n, u^{n-1}, \ldots$. For example, the forward and Adams-Bashforth methods.

3. Multi-stage explicit methods that use both knowns, $u^n, u^{n-1}, \ldots$ and intermediate or iterated values, $u^{n+\frac{1}{m}}, u^{n+\frac{2}{m}}, \ldots$. For example, the Euler method and Runge-Kutta methods.

There is also a group of methods that require approximation of an integral over two time intervals:

$$u^{n+1} - u^{n-1} = \int_{(n-1)\Delta t}^{(n+1)\Delta t} g(u(t))dt \tag{1.3}$$

of which the leap-frog scheme is most famous. These methods also come in the three flavors of implicit, single-stage and multi-stage. We will concentrate on the schemes of the form (1.2) but will discuss the leap-frog scheme.

1.1 The damped oscillation equation Edit

We will be analyzing time-stepping methods applied to a particularly simple equation that describes damped harmonic motion. To illustrate the relevance of this equation, first consider the following, somewhat arbitrary, set of equations:

$$\begin{aligned}
\partial_t u + U\partial_x u - 2\Omega \sin(\phi) v &= -r_d u + \nu \partial_{xx} u - \nu_4 \partial_{xxxx} u \\
\partial_t v + U\partial_x v + 2\Omega \sin(\phi) u &= -r_d v + \nu \partial_{xx} v - \nu_4 \partial_{xxxx} v
\end{aligned}$$

where U, Ω, ϕ, r_d, ν and ν_4 are constants in space. Although linear, the terms are representative of the types of terms encountered in dynamical models. A Fourier transformation yields:

$$\begin{aligned}
\partial_t \tilde{u}_k + ick\tilde{u}_k - 2\Omega \sin(\phi)\tilde{v}_k &= -\left(r_d + \nu k^2 + \nu_4 k^4\right)\tilde{u}_k \\
\partial_t \tilde{v}_k + ick\tilde{v}_k + 2\Omega \sin(\phi)\tilde{u}_k &= -\left(r_d + \nu k^2 + \nu_4 k^4\right)\tilde{v}_k
\end{aligned}$$

for each mode k. Combining the real variables $\tilde{u}_k$ and $\tilde{v}_k$ into a complex variable $\tilde{z}_k = \tilde{u}_k + i\tilde{v}_k$ allows the entire system to be described by:

$$\partial_t \tilde{z}_k + if(k)\tilde{z}_k = -\epsilon(k)\tilde{z}_k$$

where $f(k) = ck + 2\Omega \sin(\phi)$ and $\epsilon(k) = r_d + \nu k^2 + \nu_4 k^4$ are both real. Each Fourier mode is therefore governed by a simple equation that has an oscillatory term (ifz) and a damping term ($-\epsilon z$). When considering how to integrate a system forward in time it is generally worth classifying terms as "oscillatory" or "damping" in order to decide how to treat each term. Some general methods do not make a disctinction but there are often savings to be made by using the most appropriate scheme for a particular type of evolution.

For the purposes of analysing time-stepping methods we will therefore initially consider the damped oscillation equation:

$$\partial_t u + ifu = -\epsilon u + \tau \tag{1.4}$$

where t is time, u is the prognostic (complex) variable, f, ϵ and τ are real constants. Initial conditions at $t = 0$ are $u(t = 0) = u_o$. f is the inherent frequency of the model and ϵ is a damping rate. The solution to (1.4) is:

$$u(t) = (u_o - \frac{\tau}{\epsilon + if})e^{-(if+\epsilon)t} + \frac{\tau}{\epsilon + if}$$

1.2 The forward method Edit

The forward method is the simplest and most obvious "explicit" scheme; explicit meaning all accelerations (functions of state and fixed) are calculable without solving for the future unknown state. The general form of the forward method is

$$u^{n+1} - u^n = \int_{n\Delta t}^{(n+1)\Delta t} g(u(t))dt \approx \Delta t g(u^n). \tag{1.5}$$

Applied to the the damped oscillator equation (1.4) this yields

$$u^{n+1} - u^n = \Delta t \left(-\epsilon u^n - ifu^n + \tau\right)$$

or

$$u^{n+1} = \left(1 - \Delta t\epsilon - i\Delta t f\right) u^n + \Delta t\tau. \tag{1.6}$$

We will analyze this system by finding the numerical solution; we can do this because the damped oscillator equation is easy to solve. We will assume no forcing ($\tau = 0$) and a damped oscillatory form of the solution

$$u^n \sim e^{-i\omega n\Delta t} e^{-\lambda n\Delta t}$$

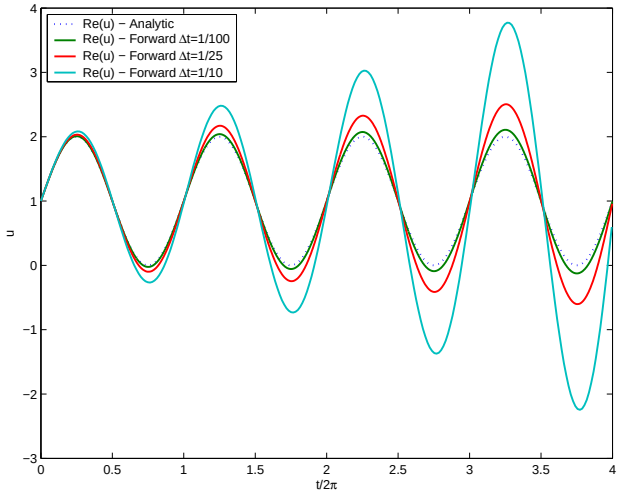

Figure 1.1: The forward method applied to a simple oscillation equation ($f = 1$ and with no damping, $\epsilon = 0$). The solutions all grow with time but as a function of the time-step.

where ω and λ are the frequency and decay rate of the numerical solution and are both assumed ot be real. Substituting into the forward equation (1.6) and cancelling common factors yields

$$e^{-i\omega\Delta t}e^{-\lambda\Delta t} = (1 - \Delta t\epsilon) - i\Delta t f.$$

This can be solved for ω and λ by taking the complex conjugate

$$e^{i\omega\Delta t}e^{-\lambda\Delta t} = (1 - \Delta t\epsilon) + i\Delta t f$$

and combining the two equations. The resulting "growth" or "stability" equation is

$$e^{-2\lambda\Delta t} = (1 - \Delta t\epsilon)^2 + \Delta t^2 f^2$$

and frequency is given by

$$\tan \Delta t\omega = \frac{\Delta t f}{1 - \Delta t\epsilon}.$$

Now consider two limits i) no damping ($\epsilon = 0$) and ii) no oscillations ($f = 0$).

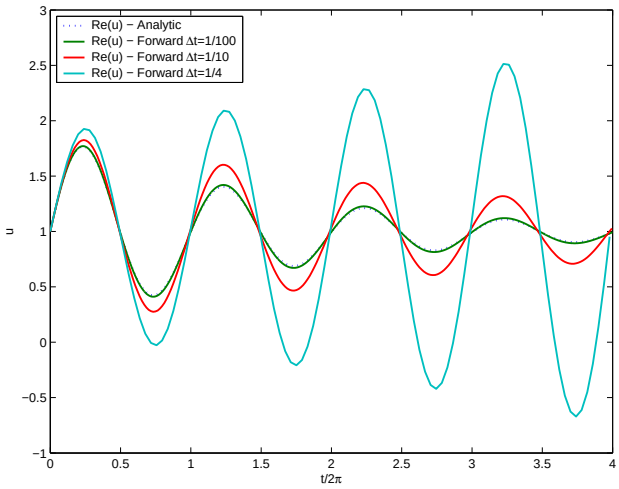

Figure 1.2: The forward method applied to the damped oscillator equation ($f = 1$ and with damping, $\epsilon = 1/10$). One solution is growing and clearly unstable. Two solutions are decaying and so the scheme is technically conditionally stable but they do not decay as fast as the true solution.

i) No damping with $\epsilon = 0$. The growth equation becomes

$$e^{-2\lambda\Delta t} = 1 + \Delta t^2 f^2$$

so that $\lambda < 0$ for all values of Δt and the solution is growing in time; a growing or unbounded solution is said to be unstable. In this case, the forward method is *unconditionally unstable* for oscillatory terms.

ii) Ignore the oscillations ($f = 0$) we see that the forward scheme is conditionally stable since the growth equation becomes

$$e^{-\lambda\Delta t} = 1 - \Delta t\epsilon$$

and is both positive and less than 1 if $\Delta t\epsilon < 1$. This is a stability criteria and the forward method is said to be *conditionally stabile* for damping terms.

Fig. 1.1 shows numerical solutions to the undamped oscillation equation using the forward method (with $f = 1$, $\epsilon = 0$, $\tau = 0 + i$ and $u_o = 1 + i$). All solutions grow with time while the true solution has constant amplitude but the rate of spurious growth is a function of time-step.

The next paragraph is included to help explain numerical methods we will cover later in the course which look like a foward method but are stable. The preceding stability results, obtained by analyzing each term in isolation, are often interpreted as meaning that the forward scheme being unusable oscillatory terms under all conditions. However, the scheme is conditionally stable if the dissipation and oscillations are considered together and satisfy:

$$1 - \sqrt{1 - \Delta t^2 f^2} \leq \Delta t \epsilon \leq 1$$

The right hand inequality arises from requiring the frequency to be finite. Note that for all quantities to be real the above criterion also implies $\Delta t f \leq 1$. In essence, the friction must be large enough to overcome the numerical growth of the oscillations. This property is used in the Lax-Wendroff time-stepping/advection scheme which will be discussed later. However, the use of explicit dissipation terms to counter spurious numerical growth is not generally advisable. Satisfying the above stability criteria, the solution remains finite but this does not necessarily mean that the solution is accurate or physically relevent. For instance, to apply the forward method to a non-dissipative system stabilized by explicit dissipation allows for a balance of terms involving the dissipation; there should be no such balance if the system in non-dissipative. *Using the forward method for oscillatory terms with explicit stabilization will lead to an inappropriate solution so do not use it.*

Fig. 1.2 shows numerical solutions to the damped oscillation equation (with $f = 1$, $\epsilon = 1/10$, $\tau = 0+i$ and $u_o = 1+i$). Two of the damped solutions appear to be stable but the rate of decay is reduced for the intermediate time-step. Here, the tendency for growth is balancing the explicit damping.

In general, the forward method should only be used for dissipatory terms and indeed its simplicity makes it the most widely used explicit method for such terms. However, it is only first order accurate in time: $O(\Delta t)$. This should be obvious from the nature of the finite difference equation (1.5); a Taylor series expansion about $t = n\Delta t$ shows the time-derivative to be approximated by a "side difference" which is of first order accuracy.

1.2.1 The backward method Edit

The backward method is also first order accurate in time but is an implicit scheme since it uses g_{n+1}. The general form of the backward method is

$$u^{n+1} - u^n = \int_{n\Delta t}^{(n+1)\Delta t} g(u(t))dt \approx \Delta t g(u^{n+1}). \tag{1.7}$$

Applied to the the damped oscillator equation (1.4) this yields

$$u^{n+1} - u^n = \Delta t \left(-\epsilon u^{n+1} - if u^{n+1} + \tau \right)$$

or

$$(1 + \Delta t\epsilon + i\Delta t f)\, u^{n+1} = u^n + \Delta t\tau. \tag{1.8}$$

As for the forward scheme, we will analyze this system by finding the numerical solution. Again, we will assume no forcing ($\tau = 0$) and a damped oscillatory form of the solution ($e^{-i\omega n\Delta t} e^{-\lambda n\Delta t}$). Substituting into the forward equation (1.8) and cancelling common factors yields

$$((1 + \Delta t\epsilon) + i\Delta t f)\, e^{-i\omega\Delta t} e^{-\lambda\Delta t} = 1$$

which is better written

$$e^{i\omega\Delta t} e^{\lambda\Delta t} = (1 + \Delta t\epsilon) + i\Delta t f.$$

This can be solved for ω and λ by taking the complex conjugate

$$e^{-i\omega\Delta t} e^{\lambda\Delta t} = (1 + \Delta t\epsilon) - i\Delta t f$$

and combining the two equations. The growth rate is given by

$$e^{2\lambda\Delta t} = (1 + \Delta t\epsilon)^2 + \Delta t^2 f^2$$

and frequency is given by

$$\tan \Delta t\omega = \frac{\Delta t f}{1 - \Delta t\epsilon}.$$

The growth factor (for one time step) is:

$$e^{-2\lambda\Delta t} = \frac{1}{(1 + \Delta t\epsilon)^2 + \Delta t^2 f^2}$$

and is less than one for all Δt, and approaches zero for infinite Δt. It is therefore unconditionally stable but damps oscillations for any finite Δt. The frequency equation becomes:

$$\tan(\omega\Delta t) = \frac{\Delta t f}{1 + \Delta t\epsilon}$$

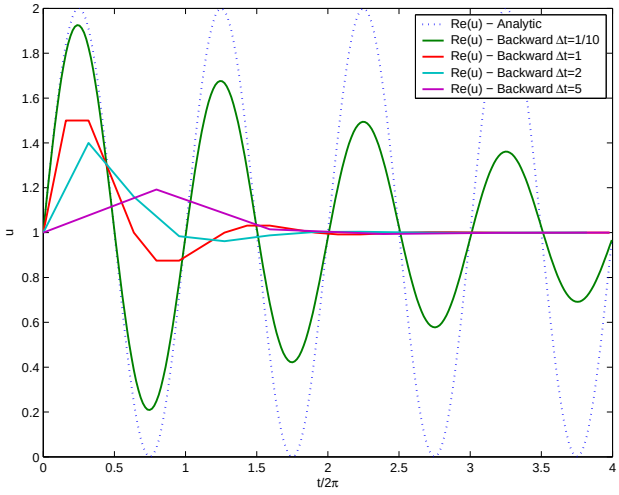

Figure 1.3: The backward method applied to a simple oscillation equation ($f = 1$ and with no damping, $\epsilon = 0$). The solutions all decay with time. For long time steps, the frequency of oscillation is reduced.

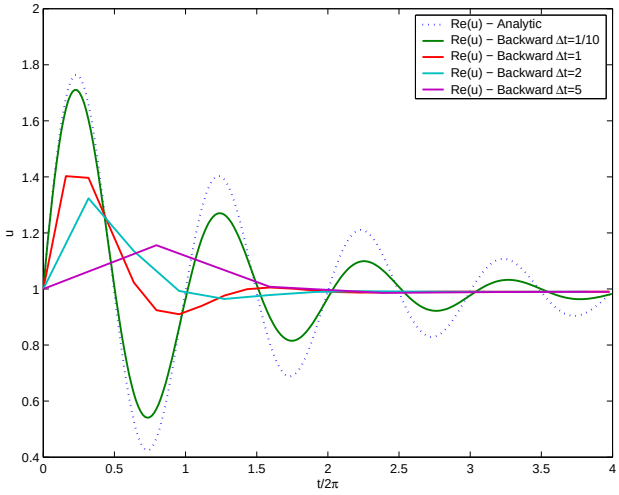

Figure 1.4: The backward method applied to the damped oscillator equation ($f = 1$ and with damping, $\epsilon = 1/10$).

which is invertable for all Δt; as the time step becomes very large the argument $\omega\Delta t$ approaches $\pi/2$. In other words, if the explicit frequency is not resolved($\Delta t f >> 1$) then the modeled oscillations are slowed down so that they have a period of $4\Delta t$.

The implicit nature of the backward scheme makes it difficult to use for non-linear and large stencil terms because those terms must be "inverted" to find the future state. The backward method is robust and has the preferential property of not aliasing unresolved motions onto lower frequencies but instead filtering them out. However, the backward scheme is only first order accurate, so even when motions are resolved they tend to be damped.

1.3 The trapezoidal method Edit

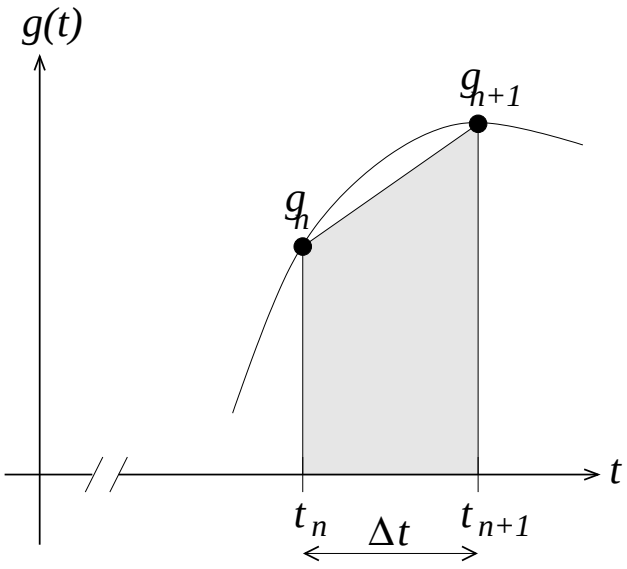

Figure 1.5: Approximating the integral of $g(u(t))$ between $t = n\Delta t$ and $t = (n+1)\Delta t$ using the trapezoidal method. The area under the curve can be approximated by the shaded area which is $\Delta t(g_n + g_{n+1})/2$.

One method to approximate the integral in 1.2 is the trapezoidal method, as illustrated in Fig. 1.5. Here, the area under the curve $g(t)$ between $t = n\Delta t$

and $t = (n+1)\Delta t$ is approximated as

$$\int_{n\Delta t}^{(n+1)\Delta t} g(u(t))dt \approx \Delta t \left(\frac{g_n + g_{n+1}}{2}\right)$$

Applying this rule to the oscillation equation (1.4), we get:

$$u^{n+1} - u^n = \Delta t\tau - (if + \epsilon)\Delta t \left(\frac{u^n + u^{n+1}}{2}\right)$$

Dividing through by Δt, we can express the discrete model using operator short-hand

$$\frac{1}{\Delta t}\delta_n u = -if\overline{u}^n - \epsilon\overline{u}^n + \tau$$

and by recognizing the second order centered finite differences we can see immediately that the model is second order accurate in time.

To actually solve the model equations, we re-arrange the equations, collecting knowns and unknowns:

$$\left(1 + i\frac{\Delta t f}{2} + \frac{\Delta t\epsilon}{2}\right) u^{n+1} = \left(1 - i\frac{\Delta t f}{2} - \frac{\Delta t\epsilon}{2}\right) u^n + \Delta t\tau$$

The forced part of the system, $\delta_n u = \Delta t\tau$, is trivially an exact solution, when the forcing is constant in time. For the purposes of analysis, we will ignore this part of the solution by setting $\tau = 0$.

To analyse the numerical properties of the model, we consider a solution of the form $Ae^{-(i\omega+\lambda)n\Delta t}$ which on substituting into the numerical model yields:

$$\left(1 + i\frac{\Delta t f}{2} + \frac{\Delta t\epsilon}{2}\right) e^{-(i\omega+\lambda)\Delta t} = \left(1 - i\frac{\Delta t f}{2} - \frac{\Delta t\epsilon}{2}\right)$$

Splitting the equation into two separate equations for the real and imaginary components, assuming ω and λ are real, gives:

$$\begin{aligned} e^{-\lambda\Delta t}\cos(\omega\Delta t) &= \frac{1 - \left(\frac{\Delta t\epsilon}{2}\right)^2 - \left(\frac{\Delta t f}{2}\right)^2}{\left(1 + \frac{\Delta t\epsilon}{2}\right)^2 + \left(\frac{\Delta t f}{2}\right)^2} \\ e^{-\lambda\Delta t}\sin(\omega\Delta t) &= \frac{\Delta t f}{\left(1 + \frac{\Delta t\epsilon}{2}\right)^2 + \left(\frac{\Delta t f}{2}\right)^2} \end{aligned}$$

from which it is easy to solve for

$$\tan\omega\Delta t = \frac{\Delta t f}{1 - \left(\frac{\Delta t\epsilon}{2}\right)^2 - \left(\frac{\Delta t f}{2}\right)^2}$$

and

$$e^{-2\lambda\Delta t} = 1 - \frac{2\Delta t\epsilon}{1 + \Delta t\epsilon + \left(\frac{\Delta t\epsilon}{2}\right)^2 + \left(\frac{\Delta t f}{2}\right)^2}$$

The numerical stability of the solution depends on the sign of λ. If $\lambda \leq 0$ then the solution is not growing with time and is thus stable. Here, we see that $e^{-2\lambda\Delta t} \leq 1$ and positive (i.e. $e^{-2\lambda\Delta t} > 0$ implies λ is real) for all Δt, assuming $\epsilon \geq 0$. The trapezoidal scheme is therefore "unconditionally stable".

In the special case of no explicit damping ($\epsilon = 0$) then the numerical solution is "neutral" ($\lambda = 0$) meaning that the oscillations have constant amplitude. This is curious since if $\Delta t f >> 1$ it means that the model can not resolve the period $2/f$. Expanding the expression for $\tan(\omega\Delta t)$:

$$\tan\omega\Delta t \approx \Delta t f \left(1 + \left(\frac{\Delta t f}{2}\right)^2 + \ldots\right)$$

and we see that the truncation terms makes ω larger than f as $\Delta t f$ increases. There is a critical point where the denominator vanishes:

$$1 - \left(\frac{\Delta t\epsilon}{2}\right)^2 - \left(\frac{\Delta t f}{2}\right)^2 = 0$$

or

$$\Delta t = \sqrt{\frac{4}{\epsilon^2 + f^2}}$$

then $\tan(\omega\Delta t)$ becomes infinite and $\omega = \pm\frac{\pi}{2}$. Even though the scheme is unconditionally stable, all skill in modeling the oscillations is lost once $\tan\omega\Delta t$ has become infinite.

The trapezoidal scheme is, by and large, a robust and stable scheme but is difficult to use for non-linear terms or for terms of a high spatial order (high order derivatives or interpolation) since it's implicit nature requires that these terms can be "inverted". The trapezoidal method is often called the Crank-Nicholson method.

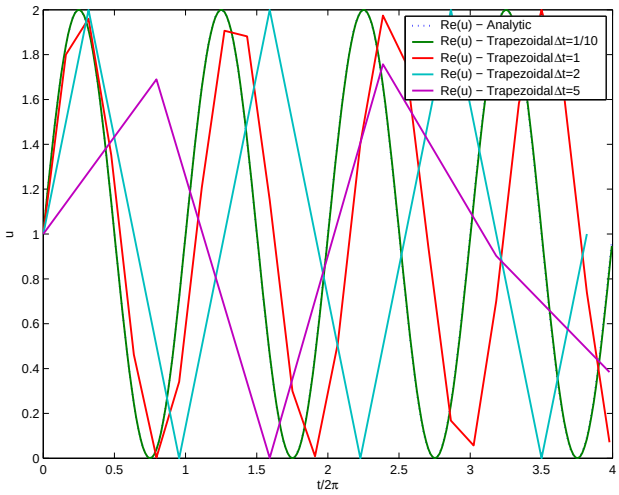

Figure 1.6: The Crank-Nicholson or trapezoidal method applied to a simple oscillation equation ($f = 1$ and with no damping, $\epsilon = 0$). The amplitude of solutions is conserved. For long time steps, the explicit frequency is aliased leading to energy at longer time scales.

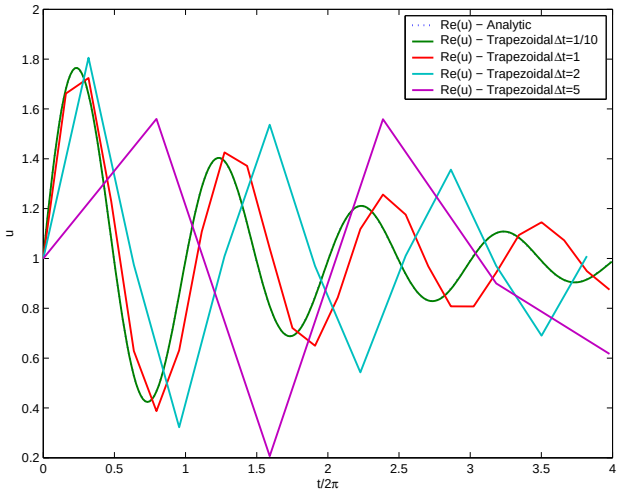

Figure 1.7: The trapezoidal method applied to the damped oscillator equation ($f = 1$ and with damping, $\epsilon = 1/10$).

1.4 General form for two-level schemes Edit

The general form of the family of single-stage two-time level schemes is:

$$u^{n+1} - u^n = \Delta t \left((\frac{1}{2} - \alpha) g_n) + (\frac{1}{2} + \alpha) g_{n+1} \right) \tag{1.9}$$

where $-\frac{1}{2} \leq \alpha \leq \frac{1}{2}$ is a weight and distinguishes between three different methods:

- $\alpha = 0$ recovers the trapezoidal scheme, discussed in section 1.3,
- $\alpha = \frac{1}{2}$ is the backward scheme, which is implicit,
- $\alpha = -\frac{1}{2}$ is the forward scheme, which is explicit.

Using the same approach as for the trapezoidal method, we can directly analyse the scheme by substituting a solution of the form $Ae^{-(i\omega+\lambda)n\Delta t}$ into the damped oscillator equation (1.9):

$$\begin{aligned} e^{-2\lambda\Delta t} &= 1 - \frac{2\Delta t\epsilon + 2\alpha\Delta t^2(\epsilon^2 + f^2)}{1 + 2(\frac{1}{2} + \alpha)\Delta t\epsilon + (\frac{1}{2} + \alpha)^2\Delta t^2(\epsilon^2 + f^2)} \\ \tan(\omega\Delta t) &= \frac{\Delta t f}{1 + 2\alpha\Delta t\epsilon - (\frac{1}{4} - \alpha^2)\Delta t^2(\epsilon^2 + f^2)} \end{aligned}$$

Substituting in $\alpha = 0$ recovers the expressions found for the trapezoidal scheme. Similarly for the forward and backward schemes.

1.5 Energy Method

The method of analysis used so far is known as the Von Neumann method. The factor $e^{-\lambda\Delta t}$ is the growth factor which must not be larger than one to avoid unbounded growth. The Von Neumann method can only be applied to linear equations because it can only account for growing-oscillatory motion. The energy method can be applied to non-linear systems and include boundary conditions but it relies on our ability to find a quadratic quantity that is either conserved or bounded; if the solution is bounded for all time then the method is said to be stable to the l_2-norm.

The damped oscillator equation (1.4) has an energy uu^* (u^* is the complex conjugate) governed by the equation

$$\partial_t(uu^*) = -2\epsilon uu^* + (\tau u^* + \tau^* u).$$

Energy is conserved by the oscillations (f does not appear in the energy equation) but is removed by dissipation ($-\epsilon uu^*$). Forcing can add or subtract energy depending on the phase of the solution.

Applying the energy method to the forward method, we multiply equation (1.6) by its complex conjugate

$$\begin{aligned}(uu^*)^{n+1} - (uu^*)^n &= \left(\Delta t^2\epsilon^2 - 2\Delta t\epsilon + \Delta t^2 f^2\right)(uu^*)^n + \Delta t^2\tau\tau^* \\ &\quad +(1 - \Delta t\epsilon - i\Delta t f)\Delta t\tau^* + (1 - \Delta t\epsilon + i\Delta t f)\Delta t\tau.\end{aligned}$$

For the energy to be bounded we need the unforced terms to be negative definite; in the absence of friction this term is positive definite and thus unconditionally unstable.

The energy method analysis of the backward scheme applied to the unforced damped oscillator yields an energy equation of the form

$$\left((1 + \Delta t\epsilon)^2 + \Delta t^2 f^2\right)(uu^*)^{n+1} = (uu^*)^n$$

which tells us that $(uu^*)^{n+1} < (uu^*)^n$ for all values of Δt; the backward method is unconditionally stable.

The advantage of the energy method is that it can handle non-linear terms. Consider the quadratic decay equation

$$\partial_t u = -C_d|u|u$$

which, using the forward method, is approximated

$$u^{n+1} = u^n - \Delta t C_d|u^n|u^n.$$

The change in energy over one time step is

$$(u^{n+1})^2 - (u^n)^2 = \Delta t C_d|u^n|u^n\left(-2 + \Delta t C_d|u^n|u^n\right)$$

so stability is conditional on $\Delta t C_d|u^n|u^n < 2$. Note that here, the stability is a function of state unlike in all the linear systems we've examined previously.

The advantage of the Von Neumann analysis is that it yields more information, particularly about the phase. The energy method and Von Neumann method generally agree about the growth of solutions but sign changes in phase of the solution provide more stringent criteria that help ensure the solution remains physical as well as bounded.

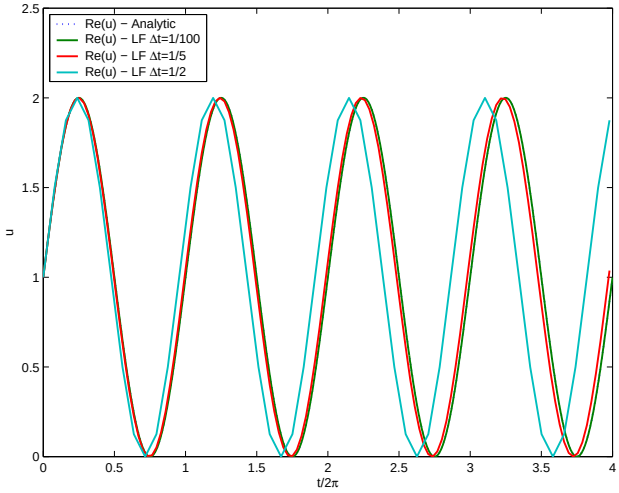

Figure 1.8: The leap-frog method applied to the oscillator equation ($f = 1$ and with no damping, $\epsilon = 0$).

1.6 Leap frog scheme Edit

The forward scheme is the simplest explicit method and we found it to not be useful for oscillatory motions. It is also first order accurate. To raise the order of accuracy we must use another time-level and to remain explicit that implies we must invoke time levels $n-1$ and n to explicitly predict time level $n+1$. The leap-frog scheme achieves second order accuracy by centering the time-derivative at time-level n and spanning $2\Delta t$:

$$u^{n+1} - u^{n-1} = 2\Delta t g_n. \tag{1.10}$$

Applying the leap-frog method to the damped oscillator equation and analyzing it with the Von Neumann method we obtain the amplification and and phase equation:

$$e^{-2(\lambda+i\omega)\Delta t} + 2\Delta t(\epsilon + if)e^{-(\lambda+i\omega)\Delta t} - 1 = 0$$

Solving for $e^{-(\lambda+i\omega)\Delta t}$ produces two roots:

$$e^{-(\lambda+i\omega)\Delta t} = -(\epsilon + if)\Delta t \pm \sqrt{1 + (\epsilon + if)^2\Delta t^2}$$

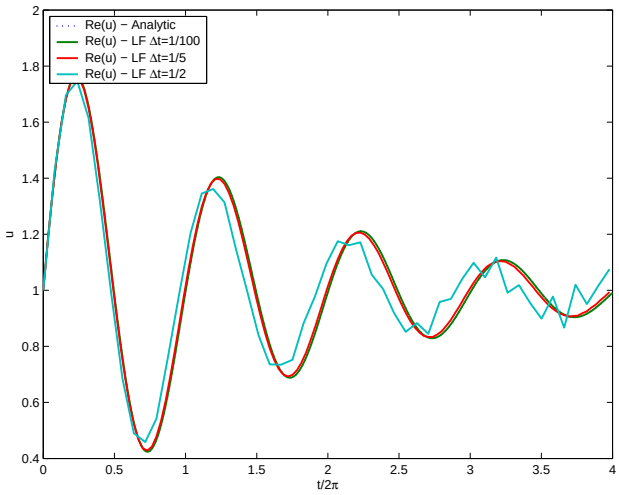

Figure 1.9: The leap-frog method applied to the damped oscillator equation ($f = 1$ and $\epsilon = 1/10$). Note the growing copmutational mode toward the end of the run.

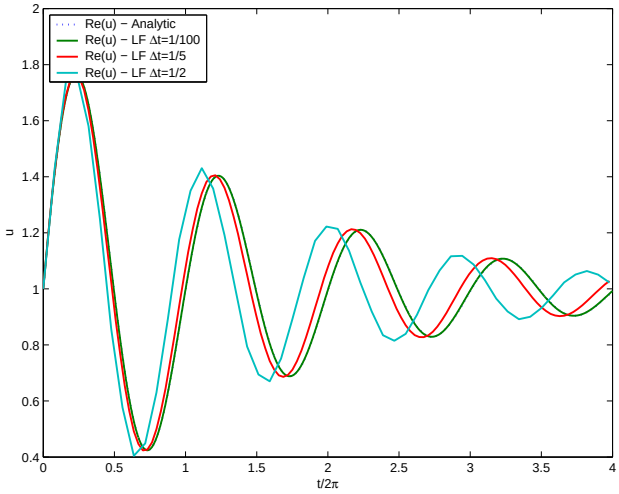

Figure 1.10: The leap-frog/forward method applied to the damped oscillator equation ($f = 1$ and $\epsilon = 1/10$). The leap-frog method is used for the oscillation term and the forward method is used for the damping term.

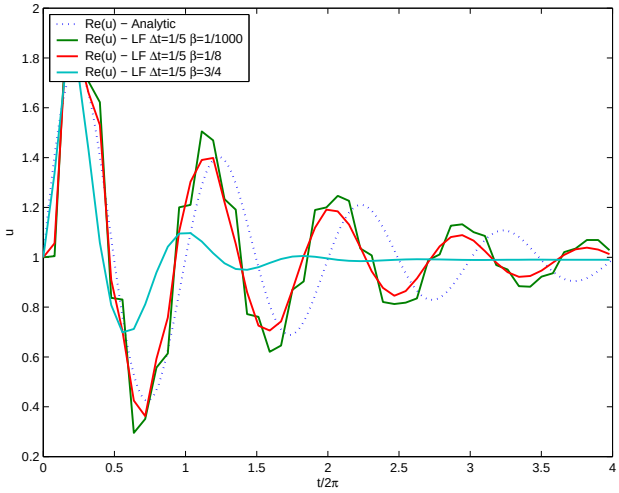

Figure 1.11: The leap-frog/forward method applied to the damped oscillator equation ($f = 1$ and $\epsilon = 1/10$) with an Asselin filter and no initialization. The lack of initialization gives the computational mode energy at the beginning of the integration term.

Consider first the pure decay problem by setting $f = 0$. The roots become:

$$e^{-(\lambda+i\omega)\Delta t} = -\epsilon\Delta t \pm \sqrt{1+\epsilon^2\Delta t^2}$$

These roots are both real. Although the positive root is less than 1 for small $\Delta t\epsilon$, the negative root is unconditionally less than -1. This means that the solution grows in amplitude and changes sign each step. Therefore, the leap-frog scheme is unconditionally unstable for damping terms.

Consider now pure oscillations by setting $\epsilon = 0$ and keeping f finite. The root of the amplification equation are:

$$e^{-(\lambda+i\omega)\Delta t} = -if\Delta t \pm \sqrt{1-f^2\Delta t^2}$$

These roots are complex but they both have the same absolute magnitude which is:

$$e^{-2(\lambda\Delta t} = \left(\pm\sqrt{1-f^2\Delta t^2}\right)^2 + f^2\Delta t^2 = 1$$

Therefore, the leap-frog scheme conserves the amplitude of oscillations. This is a significant advantage over most other schemes but we must remember

that both the computational mode and physical mode are undamped. The stability is also conditional on the square root term remaining real. That is, the leap-frog scheme is conditionally stable if $\Delta t f < 1$.

In principle, the computational mode does not grow so if intialization of the integration can minimize the computational mode then it should remain small. However, in practice, non-linear terms and time-dependent forcing will always induce a divergence in the trajectories of the physical and computational modes. It is therefore necessary to filter the computational mode.

1.7 Adams-Bashforth 2 Edit

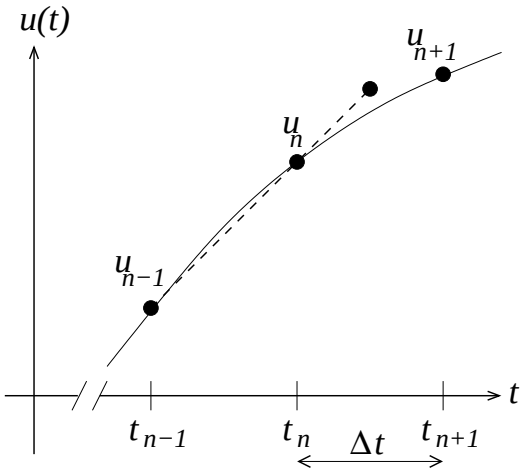

Figure 1.12: The Adams-Bashforth method extrapolates forward-in-time from known values to give a mid-point value. The second order method involves linear extrapolation, indicated by the dashed line.

The second order Adams-Bashforth scheme (AB2) is another explicit three level scheme and takes the form

$$u^{n+1} - u^n = \Delta t g(\frac{3}{2}u^n - \frac{1}{2}u^{n-1}). \tag{1.11}$$

If $g(u)$ is linear then the AB2 scheme can equivalently be written:

$$u^{n+1} - u^n = \Delta t \left(\frac{3}{2}g_n - \frac{1}{2}g_{n-1}\right)$$

but these two forms differ if $g(u)$ is non-linear. The first form is more likely to satisfy similar constraints to $g(u)$ and is therefore preferred, but the second former is easier to implement.

The AB2 method extrapolates from known values to a mid-point in the time-stepping interval, as illustrated in Fig. 1.12. Applying the AB2 scheme to the damped oscillation equation, we have:

$$u^{n+1} = \left(1 - \frac{3}{2}\Delta t(if + \epsilon)\right) u^n + \frac{1}{2}\Delta t(if + \epsilon)u^{n-1} + \Delta t\tau$$

For the purposes of simplifying the analysis, we will first consider pure oscillations (i.e. with $\epsilon = 0$). The amplification equation is then:

$$e^{-2(\lambda+i\omega)\Delta t} - (1 - \frac{3}{2}i\Delta tf)e^{-(\lambda+i\omega)\Delta t} - \frac{1}{2}i\Delta tf = 0$$

Solving for $e^{-(\lambda+i\omega)\Delta t}$ there are two roots:

$$\begin{aligned} e^{-(\lambda+i\omega)\Delta t} &= \frac{1}{2} - \frac{3}{4}i\Delta tf \pm \frac{1}{2}\sqrt{1 - i\Delta tf - \frac{9}{4}\Delta t^2 f^2} \\ &= \begin{cases} (1 - \frac{1}{2}\Delta t^2 f^2 - \frac{1}{8}\Delta t^4 f^4 - \ldots) & -i\Delta tf(1 + \frac{1}{4}\Delta t^2 f^2 + \ldots) \\ \frac{1}{2}\Delta t^2 f^2(1 + \frac{1}{4}\Delta t^2 f^2 - \ldots) & -\frac{1}{2}i\Delta tf(1 - \frac{1}{4}\Delta t^2 f^2 - \ldots) \end{cases} \end{aligned}$$

where we have expanded the root using series. The presence of two roots indicates that there is a computational mode. The first root is the physical mode since it approaches 1 as the resolution increases while the second root approaches 0. Since, with infinitesimal time-step, the computational mode vanishes the scheme is convergent; sometimes the presence of a computational mode means a scheme does not converge. The amplification factors of the two modes are:

$$1 + \frac{1}{4}\Delta t^4 f^4 + \ldots \quad ; \quad \frac{1}{2}\Delta tf + \ldots$$

respectively. That is, the computational mode is damped (if $\Delta tf << 1$) and the physical mode is unconditionally unstable. However, the fourth power in Δtf makes the growth of the physical mode "weak".

Next, consider the decay equation ($f = 0$). The amplification factors of the physical and computational modes are:

$$1 - \Delta t\epsilon + \frac{1}{2}\Delta t^2\epsilon^2 + \frac{1}{4}\Delta t^3\epsilon^3 + \ldots \quad ; \quad \frac{1}{2}\Delta tf + \ldots$$

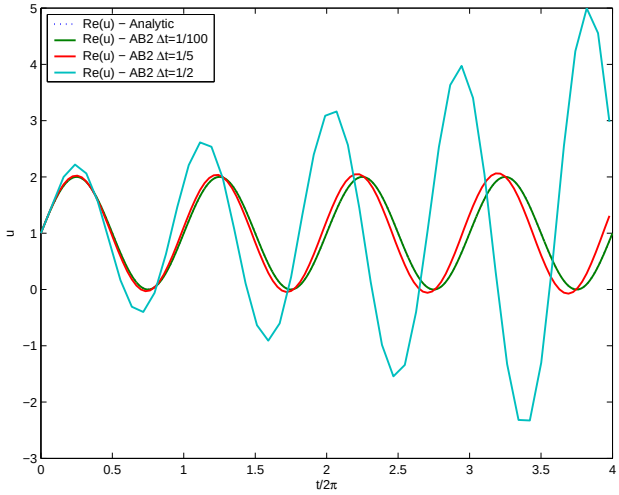

Figure 1.13: The Adams-Bashforth 2 (second order) method applied to the undamped oscillator equation ($f = 1$ and $\epsilon = 0$). Initialization is handled naively by assigning the inital conditions to all unknown time levels. The method is weakly unstable and the growth increases with the time step.

Again, the computational mode is damped. The series for the amplification of the physical mode can be compared to the series for $e^{-\lambda\Delta t}$:

$$1 - \Delta t\lambda + \frac{1}{2}\Delta t^2\lambda^2 - \frac{1}{6}\Delta t^3\lambda^3 + \ldots$$

and we see that the numerical solution decays more slowly than it should. In summary, the AB2 scheme is weakly unstable with a $O(\Delta t^4)$ growth for oscillations and stable but with a damping rate reduced by $O(\Delta t^2)$.

The AB2 scheme can be made more stable by over-extrapolating beyond the mid-point in the time interval. The scheme then becomes:

$$\begin{aligned} \overline{u} &= (\frac{3}{2} + \alpha)u^n - (\frac{1}{2} + \alpha)u^{n-1} \\ u^{n+1} - u^n &= \Delta t g(\overline{u}) \end{aligned}$$

where α is a small but positive parameter. Considering again the pure oscillation equation ($\epsilon = 0$), the amplification factor for the physical mode now is:

$$1 - \alpha\Delta t^2 f^2 + \frac{1}{4}(1 - 3\alpha)\Delta t^4 f^4 + \ldots$$

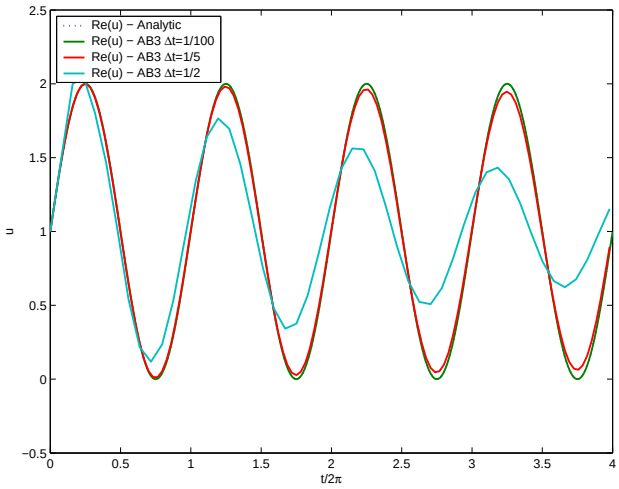

Figure 1.14: The Adams-Bashforth 3 (third order) method applied to the undamped oscillator equation ($f = 1$ and $\epsilon = 0$). Initialization is handled naively by assigning the inital conditions to all unknown time levels. Note that the method is conditionally stable and when the time step is large the method is damping.

which suggests that the scheme is stable if α is chosen such that:

$$\alpha > \frac{\Delta t^2 f^2}{4 + 3\Delta t^2 f^2}$$

However, for moderately large $\Delta t f$, the terms left out of the above analysis can cause this criteria to break down. More over, the method is no longer of second order accuracy though it is said to be "quasi-second order" since the small parameter α should be comparable to $\Delta t^2 f^2$.

1.8 Adams-Bashforth 3 Edit

In Fig. 1.12 we suggest the state variable extrapolated forward in time using second order linear extrapolation with the objective of using the extrpolated state in a "centered" time integration. We will now extend the idea to third order. Rather than use Taylor series we will fit a polynomial representaion to the evolution of time levels $n-2$, $n-1$ and n and then use the polynomial to extrapolate forward in time.

The three coefficients in the quadratic polynomial of the form

$$p(t) = a + \frac{2b}{\Delta t}(t - n\Delta t) + \frac{3c}{\Delta t^2}(t - n\Delta t)^2$$

are determined by fitting the polynomial at the time levels

$$\begin{aligned} p\,(n\Delta t) &- u^n \\ p\,((n-1)\Delta t) &- u^{n-1} \\ p\,((n-2)\Delta t) &- u^{n-2} \end{aligned}$$

which gives

$$\begin{aligned} a &= u^n \\ b &= \frac{3}{4}u^n - u^{n-1} + \frac{1}{4}u^{n-2} \\ c &= \frac{1}{6}u^n - \frac{1}{3}u^{n-1} + \frac{1}{6}u^{n-2}. \end{aligned}$$

Evaluating the polynomial at the mid-point $n + \frac{1}{2}$ gives

$$p((n + \frac{1}{2}\Delta t) = \frac{15}{8}u^n - \frac{10}{8}u^{n-1} + \frac{3}{8}u^{n-2}$$

which happens to be exactly what a Taylor series estimate would give; the Taylor series method and poynomial fitting method are generally equivalent. However, the poynomial method allows us to evaulate the *average* value over the interval $n\Delta t$ to $(n+1)\Delta t$:

$$\begin{aligned} \frac{1}{\Delta t}\int_{n\Delta t}^{(n+1)\Delta t} p(t)\,dt &= a + b + c \\ &= \frac{23}{12}u^n - \frac{16}{12}u^{n-1} + \frac{5}{12}u^{n-2} \end{aligned}$$

This is more fitting considering the integral form of equation (1.2) although a better approximation to (1.2) would be to fit the poynomial to $g(u(t))$ rather than $u(t)$. The two possible forms of AB3 are then

$$u^{n+1} = u^n + \Delta t g\left(\frac{23}{12}u^n - \frac{16}{12}u^{n-1} + \frac{5}{12}u^{n-2}\right) \tag{1.12}$$

and

$$u^{n+1} = u^n + \Delta t\left(\frac{23}{12}g(u^n) - \frac{16}{12}g(u^{n-1}) + \frac{5}{12}g(u^{n-2})\right). \tag{1.13}$$

As mentioned earlier, the first form is more likely to exhibit properties of g and might be preferred for this reason but strictly speaking the second form is a more accurate approximation of the general time integral equation (1.2).

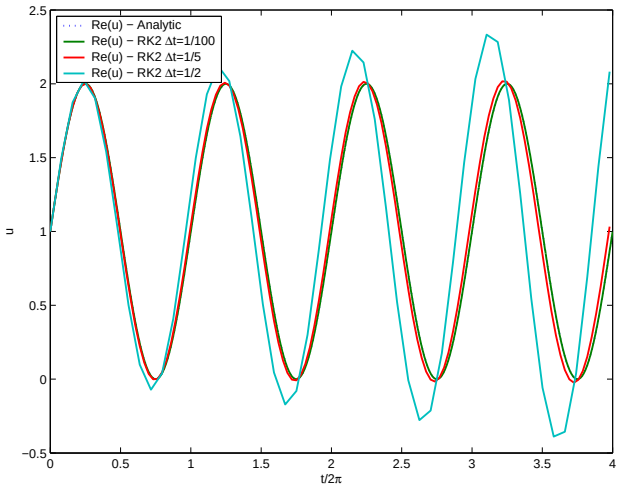

Figure 1.15: The Runge-Kutta 2 (second order) method or Heun scheme applied to the undamped oscillator equation ($f = 1$ and $\epsilon = 0$). Note that the method is weakly unstable.

1.9 Multi-stage schemes Edit

We have so far covered single-stage schemes and found that increasing from two to three time-levels has introduced a copmutational mode. An alternative approach is to return to the two-level scheme but make intermediate estimates of the solution at fractional intervals through out the time-step. We start with the two-stage schemes that can be written in the general form:

$$\begin{aligned} \tilde{u}^{n+\alpha} &= u^n + \alpha \Delta t g(u^n) \\ u^{n+1} &= u^n + \Delta t \left(\beta g(\tilde{u}^{n+\alpha}) + (1-\beta) g(u^n) \right) \end{aligned} \tag{1.14}$$

where $0 \leq \alpha \leq 1$ and $0 \leq \beta \leq 1$ are real parameters. The corresponding amplitude equation is:

$$e^{-(\lambda + i\omega)\Delta t} = 1 - (\epsilon + if)\Delta t + \alpha\beta(\epsilon + if)^2 \Delta t^2$$

First consider the decay problem by setting $f = 0$ in which case the amplitude expression is real:

$$e^{-\lambda \Delta t} = 1 - \Delta t \epsilon + \alpha\beta \Delta t^2 \epsilon^2$$

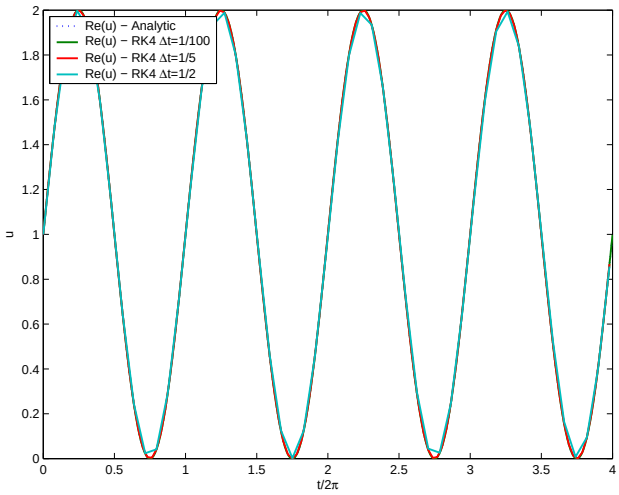

Figure 1.16: The Runge-Kutta 4 (fourth order) method applied to the undamped oscillator equation ($f = 1$ and $\epsilon = 0$). The method is conditionally stable for longer time-steps than lower order explicit schemes.

Notice that setting $\beta = 0$ recovers the forward method. The scheme is conditionally stable for damping terms if $0 \leq \alpha\beta \leq 1$ and $\Delta t\epsilon < 1$. Next consider pure oscillations by setting $\epsilon = 0$. The amplitude expression is then complex

$$e^{-(\lambda+i\omega)\Delta t} = 1 - i\Delta t f - \alpha\beta\Delta t^2 f^2$$

The amplitude and phase factors are then:

$$\begin{aligned} e^{-2\lambda\Delta t} &= 1 + (1 - 2\alpha\beta)\Delta t^2 f^2 + \alpha^2\beta^2\Delta t^4 f^4 \\ \tan\omega\Delta t &= \frac{\Delta t f}{1 - \alpha\beta\Delta t^2 f^2} \end{aligned}$$

Clearly, the only stable schemes have $\alpha\beta > \frac{1}{2}$ and that for $\alpha\beta = \frac{1}{2}$ the schemes are "weakly" unstable. The various rational choices of α and β are:

- $\alpha = 1$, $\beta = \frac{1}{2}$: Is the Heun method. This is an explicit analog to the trapezoidal method; a forward step is used to predict $\tilde{u}_{n+1}$ and then the result used in an explicit trapezoidal evaluation. Second order accurate and weakly unstable.

- $\alpha = \frac{1}{2}$, $\beta = 1$: The mid-point method (or second-order Runge-Kutta) has exactly the same stability properties as the Heun method for linear models. Second order accurate and weakly unstable.
- $\alpha = 1$, $\beta = 1$: The forward-backward, Euler-backward or Matsuno method. The forward method is used to predict $\tilde{u}_{n+1}$ and then the result used in an explicit backward step. First order accurate, conditonally stable ($\Delta t f < 1$) and damping (maximized at $\Delta t f = 1/\sqrt{2}$).

1.9.1 Derivation of Runge-Kutta methods Edit

We will now analyze the accuracy of the above two-stage schemes.

The Taylor series expansion for u^{n+1} about t_n is:

$$u^{n+1} = u^n + \Delta t u'(t_n) + \frac{1}{2!}\Delta t^2 u''(t_n) + \frac{1}{3!}\Delta t^3 u'''(t_n) + \ldots$$

Since $u'(t_n) = g(u^n, t_n)$ we can write:

$$\begin{aligned} u' &= g \\ u'' &= \partial_t g + u'\partial_u g = \partial_t g + g\partial_u g \\ u''' &= d_t(\partial_t g + g\partial_u g) = \partial_{tt} g + 2g\partial_{tu} g + g^2\partial_{uu} g + \partial_u g \partial_t g + g\partial_u g^2 \end{aligned}$$

so that

$$u^{n+1} = u^n + \Delta t g + \frac{1}{2}\Delta t^2 \left(\partial_t g + g\partial_u g\right) + O(\Delta t^3) \tag{1.15}$$

Now we write the algorithm in a series of steps as follows:

$$\begin{aligned} g_1 &= g(u^n, t_n) \\ u_1 &= u^n + \alpha \Delta t g_1 \\ g_2 &= g(u_1, t_n + \delta \Delta t) \\ u^{n+1} &= u^n + \gamma_1 \Delta t g_1 + \gamma_2 \Delta t g_2 \end{aligned}$$

where we have generalized the algorithm further than before by introducing the arbitrary parameters α, δ, γ_1 and γ_2. The objective now is to manipulate the last step into a form corresponding to (1.15). On inspecting the last step, we see that we need a Taylor expansion of g_2 which is:

$$\begin{aligned} g_2 &= g(u^n + \alpha \Delta t g_1, t_n + \delta \Delta t) \\ &= g(u^n + \alpha \Delta t g_1, t_n) + \delta \Delta t \partial_t g(u^n + \alpha \Delta t g_1, t_n) + O(\Delta t^2) \\ &= g(u^n, t_n) + \alpha \Delta t g_1 \partial_u g(u^n, t_n) + \delta \Delta t \partial_t g(u^n, t_n) + O(\Delta t^2) \end{aligned}$$

Substituting into the last step of the algorithm we get:

$$u^{n+1} = u^n + \Delta t\,(\gamma_1 + \gamma_2)\,g + \Delta t^2 \gamma_2\,(\alpha \partial_t g + \delta g \partial_u g) + O(\Delta t^3)$$

To make terms match with those in equation (1.15) we must chose:

$$\begin{aligned} \gamma_1 + \gamma_2 &= 1 \\ \gamma_2 \alpha &= \frac{1}{2} \\ \gamma_2 \delta &= \frac{1}{2} \end{aligned}$$

in which case the scheme is then of order $O(\Delta t^2)$. These three equations in four unknowns can be solved in terms of jsut one parameter:

$$\delta = \alpha \quad ; \quad \gamma_2 = \frac{1}{2\alpha} \quad ; \quad \gamma_1 = 1 - \frac{1}{2\alpha}$$

The algorithm can now be written:

$$\begin{aligned} g_1 &= g(u^n, t_n) \\ u_1 &= u^n + \alpha \Delta t g_1 \\ g_2 &= g(u_1, t_n + \alpha \Delta t) \\ u^{n+1} &= u^n + \left(1 - \frac{1}{2\alpha}\right) \Delta t g_1 + \frac{1}{2\alpha} \Delta t g_2 \end{aligned}$$

which corresponds to the two-stage method if we set $\beta = \frac{1}{2\alpha}$ in equation (1.14). For the two-stage method we found that stability is conditional on $\alpha\beta > \frac{1}{2}$ and that if $\alpha\beta = \frac{1}{2}$ then the two-stage method was weakly unstable due to a $O(\Delta t^4)$ term. This means that the second order accurate Runge-Kutta methods are weakly unstable.

1.9.2 Higher order Runge-Kutta Edit

Derivation of higher order Runge-Kutta methods uses the same technique. However, the pages of algebra entailed in find the coefficients are unrevealing. Instead, we supply the "Maple" code to illustrate how to obtain the coefficients:

```
> n:=3;
```

```
> alias( G=g(t,u(t)), Gt=D[1](g)(t,u(t)), Gu=D[2](g)(t,u(t)),
  Gtt=D[1,1](g)(t,u(t)), Gtu=D[1,2](g)(t,u(t)), Guu=D[2,2](g)(t,u(t)) );
> D(u):=t->g(t,u(t));
> TaylorExpr:=(mtaylor(u(t+h),h,n+1)-u(t))/h;
> g1:=mtaylor( g(t,u(t)) ,h,n);
> g2:=mtaylor( g(t+beta[1]*h,u(t)+h*alpha[1]*g1) ,h,n);
> g3:=mtaylor( g(t+beta[2]*h,u(t)+h*alpha[2,1]*g1+h*alpha[2,2]*g2) ,h,n);
> RungeKuttaExpr:=( gamma[1]*g1+gamma[2]*g2+gamma[3]*g3 );
> eq:=simplify(RungeKuttaExpr-TaylorExpr);
> eqns:={coeffs(eq,[h,G,Gt,Gu,Gtt,Gtu,Guu])};
> indets(eqns);
> solve(eqns,indets(eqns));
```

Extending the above script to fourth order involves adding the necessary definitions for u_3 and g_4. The most common fourth order method is:

$$\begin{aligned} g_1 &= g(u^n, t_n) \\ g_2 &= g(u^n + \frac{1}{2}\Delta t g_1, t_n + \frac{1}{2}\Delta t) \\ g_3 &= g(u^n + \frac{1}{2}\Delta t g_2, t_n + \frac{1}{2}\Delta t) \\ g_4 &= g(u^n + \Delta t g_3, t_n + \Delta t) \\ u^{n+1} &= u^n + \frac{1}{6}\Delta t \left(g_1 + 2g_2 + 2g_3 + g_4\right) \end{aligned}$$

and is widely used. It is both accurate and near neutrally stable. Higher than fourth order Runge-Kutta methods exist and can be found in text books but are rarely used in models of the ocean or atmosphere.

1.10 Side-by-side comparison Edit

A simple P-Z model is

$$\begin{aligned} N &= N_t - P - Z \\ \partial_t P &= \frac{uPN}{N + N_o} - gZP \\ \partial_t Z &= agZP - dZ \end{aligned} \tag{1.16}$$

where $N_t = 5$, $N_o = 0.1$, $u = 0.03$, $g = 0.2$, $a = 0.4$ and $d = 0.08$ are all constants.

A slightly different model has a wider separation of inherent time-scales and behaves more non-linearly:

$$\begin{aligned} N &= N_t - P - Z \\ \partial_t P &= \frac{uPN}{N+N_o} - \frac{gZP}{P+P_o} \\ \partial_t Z &= \frac{agZP}{P+P_o} - dZ \end{aligned} \tag{1.17}$$

where $N_t = 5$, $N_o = 0.1$, $P_o = 0.5$, $u = 0.01$, $g = 0.1$, $a = 1$ and $d = 0.08$ are all constants.

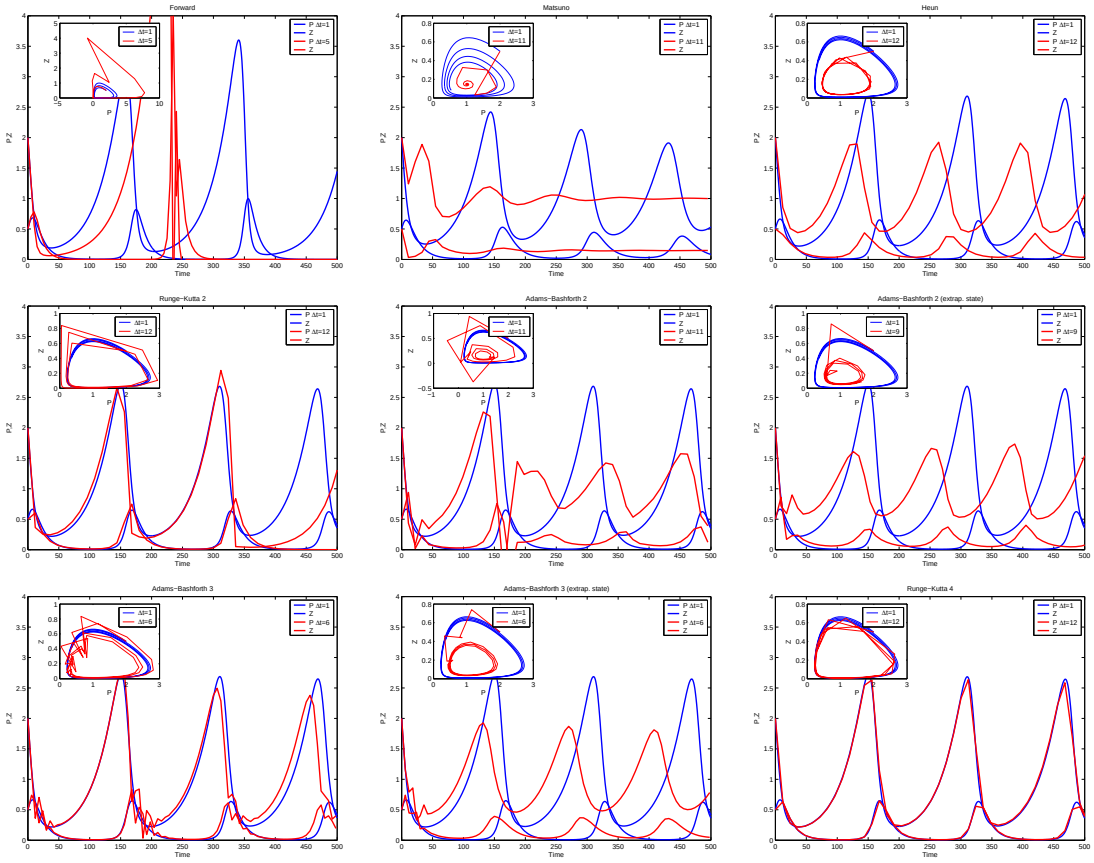

Figure 1.17: Solutions to the P-Z model (equations 1.16) obtained using a "small" $\Delta t = 1$ and the largest "stable" Δt for each scheme.

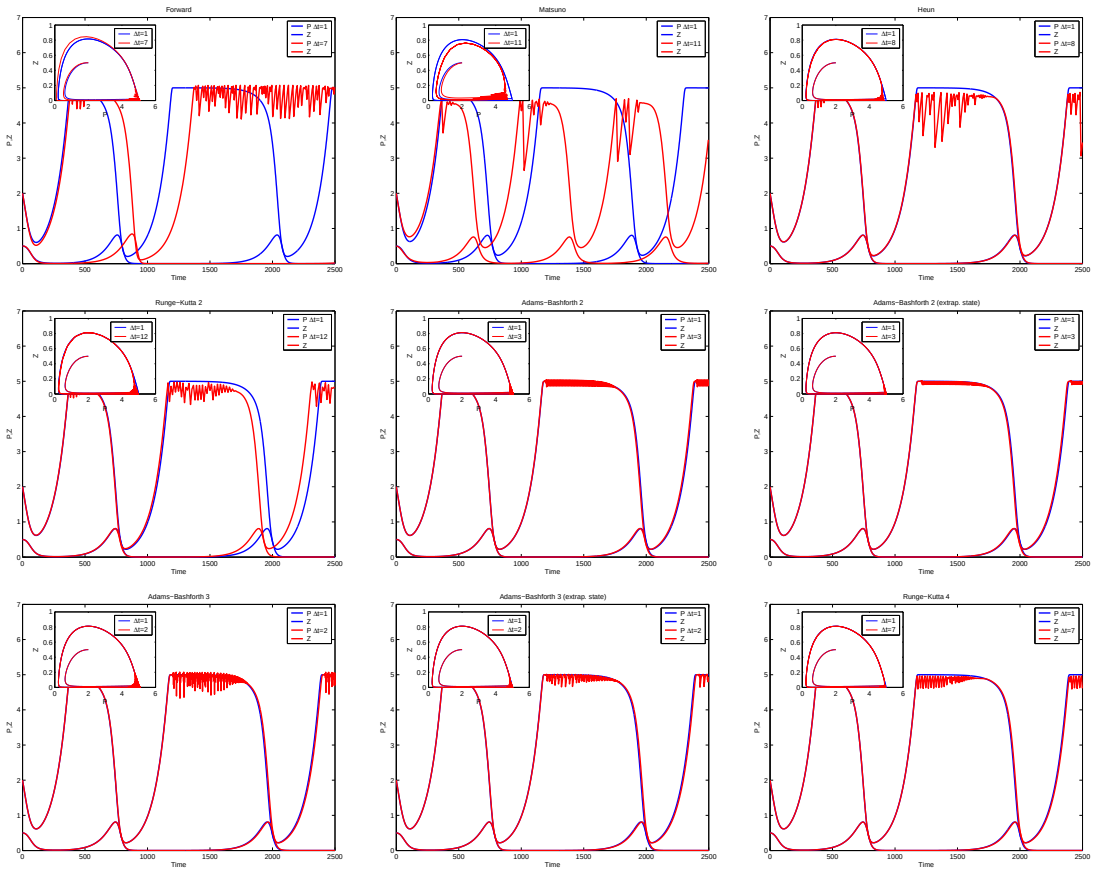

Figure 1.18: Solutions to the P-Z model (equations 1.17) obtained using a "small" $\Delta t = 1$ and the largest "stable" Δt for each scheme.