Introduction to Computational Physics

University of Maribor,
Summer Term 2003

Franz J. Vesely
University of Vienna

www.ap.univie.ac.at/users/Franz.Vesely/

Reading Matter:

Vesely:

Computational Physics – An Introduction Plenum Press, New York 1994. Second edition to appear 2001.

Do not ask me about this one.

Potter:

Computational Physics. Wiley, New York 1980.

Very valuable text; in some places too demanding for the beginner.

Hockney, Eastwood:

Computer Simulation Using Particles. McGraw-Hill, New York 1981.

Very good, particularly, but not exclusively, for plasma physicists; covers large areas of computational physics, in spite of the seemingly restrictive title.

Press, Flannery, Teukolsky, Vetterling:

Numerical Recipes in Fortran. Cambridge University Press, Cambridge 1992.

Excellent handbook of modern numerical mathematics; comes with sample programs in various programming languages.

Giordano:

Computational Physics. Prentice-Hall, New Jersey 1997.

This is one of those texts in which little is said about the origin of the the algorithms used. However, it is redeemed by its large collection of charming physical applications. Use it together with a more method-oriented text.

Gould, Tobochnik:

Introduction to Computer Simulation Methods: Application to Physical Systems. Addison-Wesley, Reading 1996.

Nice "hands-on" introduction; starts out with elementary physics problems and works up to such cutting-edge applications as dynamical quantum simulation and renormalization.

Garcia:

Numerical Methods for Physics. Prentice Hall, New Jersey, 1999.

Carefully organized introduction to the field; presents many examples, including code and graphics.

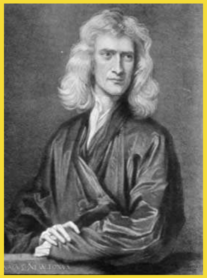
Gershenfeld:

The Nature of Mathematical Modeling. Cambridge University Press, Cambridge 1999.

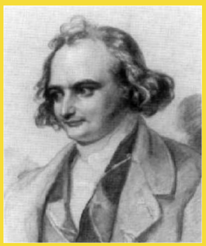
Grand tour through applied mathematics, covering analytical, numerical and observational models.

BASIC TOOLS OF OUR TRADE

Most of the methods used by computational physicists are drawn from three areas of numerical mathematics, namely from



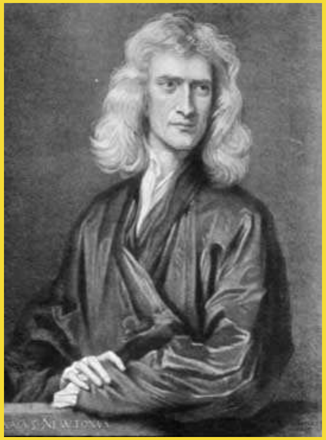
Calculus of Differences



Linear Algebra

Stochastics

1. Finite Differences

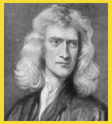


Yes, Computational Physics is that old!

Difference calculus:

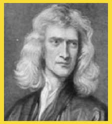
Use *finite differences* in place of *infinitesimal differentials*:

Given $f(x)$, let $x_k \equiv k\Delta x \, (k = 1, 2, ...)$



1. Finite differences

- History: opposite route
- Kepler 2- and 3-body problems (chaos!)
- Difference calculus remains applicable for any number of bodies and any potential
- Price paid: only tabulated trajectory



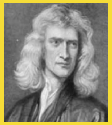
Given equidistant table values $f_k \equiv f(x_k)$, define

$$
\begin{aligned}
\Delta f_k &\equiv f_{k+1} - f_k \quad \text{Forward Difference} \\
\nabla f_k &\equiv f_k - f_{k-1} \quad \text{Backward Difference} \\
\delta f_k &\equiv f_{k+1/2} - f_{k-1/2} \quad \text{Central Difference } (*)
\end{aligned}
$$

(*) Table values at $x_{k\pm 1/2}$ not given; please have patience! In an emergency, use the "central mean"

$$
\begin{aligned}
\delta f_k \to \mu\delta f_k &\equiv \frac{1}{2}\left[\delta f_{k+1/2} + \delta f_{k-1/2}\right] \\
&= \frac{1}{2}\left[f_{k+1} - f_{k-1}\right]
\end{aligned}
$$

which uses only table values.

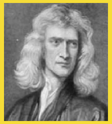


Recursive definition:

$$\begin{aligned}\Delta^2 f_k &\equiv \Delta f_{k+1} - \Delta f_k = f_{k+2} - 2f_{k+1} + f_k \\ \nabla^2 f_k &\equiv \ldots \ \text{(Exercise!?)} \\ \delta^2 f_k &\equiv \delta f_{k+1/2} - \delta f_{k-1/2} \\ &= f_{k+1} - 2f_k + f_{k-1} \quad (*)\end{aligned}$$

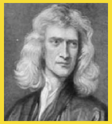
etcetera.

(*) Here is the reward for your patience!



Application of Finite Differences: Difference Quotients

- Construct interpolation polynomials $F(x)$ using forward, backward, or central differences. These are known as Newton-Gregory (Forward or Backward) and Stirling polynomials: NGF, NGB, ST, for short.
- Differentiate these polynomials, arriving at $F'(x)$ and $F''(x)$ etc.
- Insert $x = x_k$ to get approximations to the derivatives at x_k.



First Derivatives

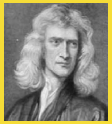
Replacing dx by Δx and df by Δf_k, ∇f_k, or δf_k we arrive at various approximations to the first derivative of f at x_k:

DNGF
(Differentiated Newton-Gregory Forward):

$$F_k{}' \approx \frac{1}{\Delta x}\left[\Delta f_k - \frac{\Delta^2 f_k}{2} + \frac{\Delta^3 f_k}{3} - \dots\right]$$

Example:

$$\begin{aligned} F_k{}' &= \frac{1}{\Delta x}\left[\Delta f_k - \frac{\Delta^2 f_k}{2}\right] + O[(\Delta x)^2] \\ &= \frac{1}{\Delta x}\left[-\frac{1}{2}f_{k+2} + 2f_{k+1} - \frac{3}{2}f_k\right] + O[(\Delta x)^2] \end{aligned}$$

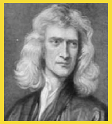


DNGB
(Differentiated Newton-Gregory Backward):

$$F_k{}' \approx \frac{1}{\Delta x}\left[\nabla f_k + \frac{\nabla^2 f_k}{2} + \frac{\nabla^3 f_k}{3} + \ldots\right]$$

Example:

$$\begin{aligned} F_k{}' &= \frac{1}{\Delta x}\left[\nabla f_k + \frac{\nabla^2 f_k}{2}\right] + O[(\Delta x)^2] \\ &= \frac{1}{\Delta x}\left[\frac{3}{2} f_k - 2 f_{k-1} + \frac{1}{2} f_{k-2}\right] + O[(\Delta x)^2] \end{aligned}$$

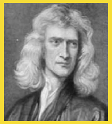


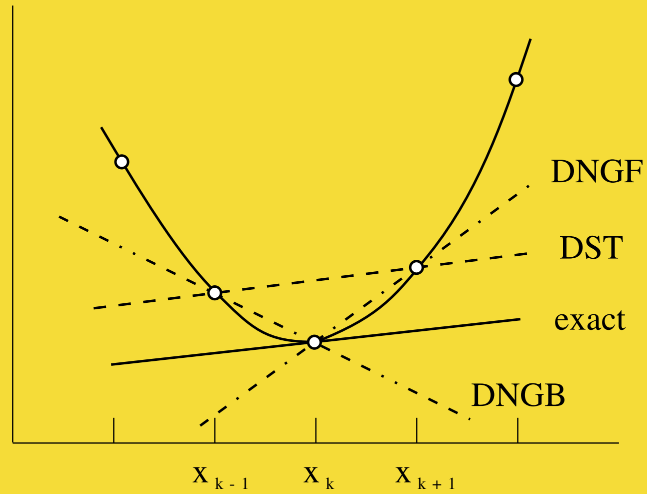
DST
(Differentiated Stirling):

$$F_k{}' \approx \frac{1}{\Delta x}\left[\mu\delta f_k - \frac{1}{6}\mu\delta^3 f_k + \frac{1}{30}\mu\delta^5 f_k + \ldots\right]$$

Example:

$$\begin{aligned} F_k{}' &= \frac{1}{\Delta x}\left[\mu\delta f_k\right] + O[(\Delta x)^2] \\ &= \frac{1}{2\Delta x}\left[f_{k+1} - f_{k-1}\right] + O[(\Delta x)^2] \end{aligned}$$



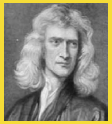


Comparison of various simple approximations to the first differential quotient:

$$DNGF: \quad F'_k = \frac{\Delta f_k}{\Delta x} + O[\Delta x] = \frac{1}{\Delta x}\,[f_{k+1} - f_k] + O[\Delta x]$$

$$DNGB: \quad F'_k = \frac{\nabla f_k}{\Delta x} + O[\Delta x] = \frac{1}{\Delta x}\,[f_k - f_{k-1}] + O[\Delta x]$$

$$DST: \quad F'_k = \frac{\mu\delta f_k}{\Delta x} + O[(\Delta x)^2] = \frac{1}{2\Delta x}\,[f_{k+1} - f_{k-1}] + O[(\Delta x)^2]$$



Second Derivatives

The same procedure as before yields

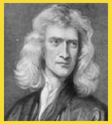
DDNGF:

$$F_k{}'' \approx \frac{1}{(\Delta x)^2}\left[\Delta^2 f_k - \Delta^3 f_k + \frac{11}{12}\Delta^4 f_k - \ldots\right]$$

Example:

$$\begin{aligned} F_k{}'' &= \frac{1}{(\Delta x)^2}\Delta^2 f_k + O(\Delta x) \\ &= \frac{1}{(\Delta x)^2}\left[f_{k+2} - 2f_{k+1} + f_k\right] + O(\Delta x) \end{aligned}$$

(pretty bad!)



Let's try again....

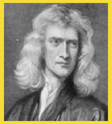
DDNGB:

$$F_k'' \approx \frac{1}{(\Delta x)^2}\left[\nabla^2 f_k + \nabla^3 f_k + \frac{11}{12}\nabla^4 f_k + \dots\right]$$

Example:

$$\begin{aligned} F_k'' &= \frac{1}{(\Delta x)^2}\nabla^2 f_k + O(\Delta x) \\ &= \frac{1}{(\Delta x)^2}\left[f_k - 2f_{k-1} + f_{k-2}\right] + O(\Delta x) \end{aligned}$$

(pretty bad, too!)



And the winner is...

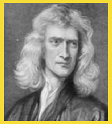
DDST:

$$F_k{}'' \approx \frac{1}{(\Delta x)^2}\left[\delta^2 f_k - \frac{1}{12}\delta^4 f_k + \frac{1}{90}\delta^6 f_k - \dots\right]$$

Example:

$$\begin{aligned} F_k{}'' &= \frac{1}{(\Delta x)^2}\delta^2 f_k + O\left[(\Delta x)^2\right] \\ &= \frac{1}{(\Delta x)^2}\left[f_{k+1} - 2f_k + f_{k-1}\right] + O\left[(\Delta x)^2\right] \end{aligned}$$

(much better!)



Finite Differences in Two Dimensions

Let $f(x, y)$ be given for equidistant values of x and y, respectively:

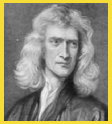
$$f_{i,j} \equiv f(x_0 + i\,\Delta x, y_0 + j\,\Delta y)$$

We will use the short notation

$$f_x \equiv \frac{\partial f(x, y)}{\partial x}$$

etc. for the partial derivatives of the function f with respect to its arguments.
Note: One of the arguments may be the time t: $f = f(x, t)$ etc.

For the numerical treatment of partial differential equations (PDEs) we again have to construct discrete approximations to the partial derivatives at the base points (x_i, y_j). $\Longrightarrow$



First derivatives in 2 dimensions

Using the DNGF, DNGB, or DST approximation of lowest order, we have

$$[F_x]_{i,j} \approx \frac{1}{\Delta x}\left[f_{i+1,j} - f_{i,j}\right] + O[\Delta x] \equiv \frac{\Delta_i f_{i,j}}{\Delta x} + O[\Delta x]$$

or

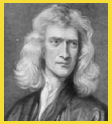
$$[F_x]_{i,j} \approx \frac{1}{\Delta x}[f_{i,j} - f_{i-1,j}] + O[\Delta x] \equiv \frac{\nabla_i f_{i,j}}{\Delta x} + O[\Delta x]$$

or

$$[F_x]_{i,j} \approx \frac{1}{2\Delta x}\left[f_{i+1,j} - f_{i-1,j}\right] + O[(\Delta x)^2] \equiv \frac{\mu\delta_i f_{i,j}}{\Delta x} + O[(\Delta x)^2]$$

Again, the *central* difference scheme is superior.

But what about *second* derivatives? $\Longrightarrow$



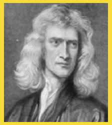
Second derivatives in 2 dimensions

By again fixing one of the independent variables – y, say – and considering only f_{xx}, we obtain, in terms of the Stirling (centered) approximation,

$$\begin{aligned} [F_{xx}]_{i,j} &\approx \frac{1}{(\Delta x)^2}[f_{i+1,j} - 2f_{i,j} + f_{i-1,j}] + O[(\Delta x)^2] \\ &\equiv \frac{\delta_i^2 f_{i,j}}{(\Delta x)^2} + O[(\Delta x)^2] \end{aligned}$$

Analogous (and less accurate) formulae are valid within the NGF- and NGB-approximations, respectively.

What about *mixed* derivatives? $\Longrightarrow$



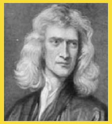
Mixed derivatives

Approximating f_{xy} use the same kind of approximation with respect to both the x- and the y-direction. (This may not hold if x and y have a different character, e.g. one space and one time variable.)

Stirling:

$$
\begin{aligned}
&[F_{xy}]_{i,j} \approx \\
&\frac{1}{4\Delta x \Delta y}\left[f_{i+1,j+1} - f_{i+1,j-1} - f_{i-1,j+1} + f_{i-1,j-1}\right] + O[\Delta x \Delta y] \\
&\equiv \frac{\mu\delta_i}{\Delta x}\left[\frac{\mu\delta_j f_{i,j}}{\Delta y}\right] + O[\Delta x \Delta y]
\end{aligned}
$$

Now for the *curvature* of $f(x, y)$: $\Longrightarrow$



Curvature of a function $f(x, y)$

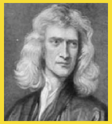
To find the local curvature at the grid point (i, j) we have to apply the nabla operator ∇ twice.(*) There are two ways:
Either "difference" along the grid axes,

$$\nabla^2 f(x,y) \approx \frac{1}{(\Delta l)^2}\left[f_{i+1,j} + f_{i,j+1} + f_{i-1,j} + f_{i,j-1} - 4f_{i,j}\right]$$

or apply "diagonal differencing", writing

$$\nabla^2 f(x,y) \approx \frac{1}{2(\Delta l)^2}\left[f_{i+1,j+1} + f_{i-1,j+1} + f_{i-1,j-1} + f_{i+1,j-1} - 4f_{i,j}\right]$$

(*) Note that the nabla operator ∇ mentioned here is not to be mixed up with the *backward difference* for which we use the same symbol.



Physical Examples:

A. Newton' equation for an oscillator

$$\frac{d^2x}{dt^2} = -\omega_0^2\, x$$

$\Longrightarrow$

$$\frac{\delta^2 x_n}{(\Delta t)^2} = -\omega_0^2 x_n + O[(\Delta t)^2]$$

$\Longrightarrow$

$$\boxed{x_{n+1} = 2x_n - x_{n-1} + (-\omega_0^2 x_n)(\Delta t)^2 + O[(\Delta t)^4]}$$

Step-by-step integration!

Applicable for any right hand side, and for any number of coupled equations of motion!

Exercise:

a) Write a program to tabulate and/or draw the analytical solution to the HO equation. (You may achieve a very concise visualization by displaying the trajectory in *phase space*, i.e. in the coordinate system $\{x; \dot{x}\}$; where for $\dot{x}$ the approximation $\dot{x} \approx (x_{n+1} - x_{n-1})/2\Delta t$ may be used.) Choose values of ω_0^2, Δt and $x_0, \dot{x}_0$, and use these to determine the exact value of x_1. Starting with x_0 and x_1, employ the above algorithm to compute the further path $\{x_n\,; n = 2, 3, \ldots\}$. Test the performance of your program by varying Δt and ω_0^2.

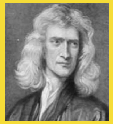
b) Now apply your code to the *anharmonic* oscillator

$$\frac{d^2x}{dt^2} = -\omega_0^2\, x - \beta\, x^3$$

To start the algorithm you may use the approximate value given by

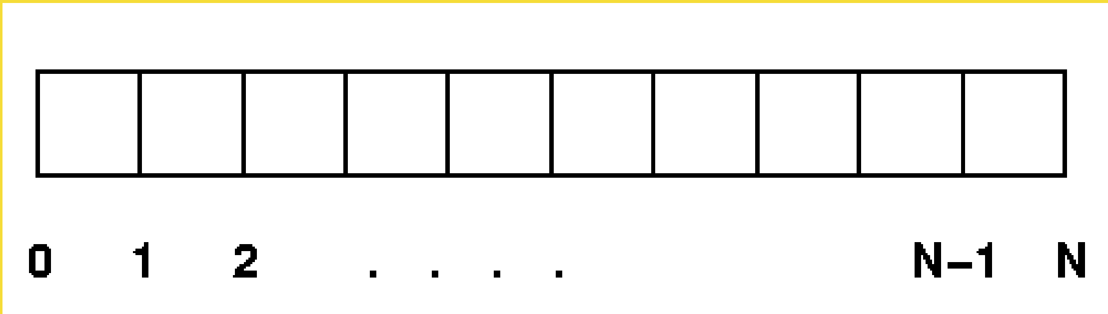
$$x_1 \;\approx\; x_0 + \dot{x}_0 \Delta t + \ddot{x}_0 \frac{(\Delta t)^2}{2}$$

$\Longrightarrow$Applet HarmOsci



B. Thermal Conduction

$$\frac{\partial T(x,t)}{\partial t} = \lambda \frac{\partial^2 T(x,t)}{\partial x^2}$$



Writing $T_i^n \equiv T(x_i, t_n)$, $\Longrightarrow$ "FTCS scheme" ("forward-time, centered-space")

$$\frac{1}{\Delta t}[T_i^{n+1} - T_i^n] \approx \frac{\lambda}{(\Delta x)^2}[T_{i+1}^n - 2T_i^n + T_{i-1}^n]$$

or

$$\boxed{T_i^{n+1} = (1-2a)T_i^n + a(T_{i-1}^n + T_{i+1}^n)\,, \quad \text{with} \quad a \equiv D\,\Delta t/(\Delta x)^2}$$

for $i = 1, \ldots N-1$ (and T_0^{n+1}, T_N^{n+1} given as boundary conditions).

<u>Exercise:</u> Let us divide the rod into $N = 10$ pieces of equal length, with node points $i = 0, \ldots N$, and assume the boundary conditions $T(0,t) \equiv T_0^n = 1.0$ and $T(L,t) \equiv T_{10}^n = 0.5$. The values for the temperature at time $t = 0$ (the *initial values*) are $T_1^0 = T_2^0 = \ldots T_5^0 = 1.0$ and $T_6^0 = T_7^0 = \ldots T_9^0 = 0.5$ (step function).