Homework – due December 1st

(**1**) Consider the four functions:

$$\begin{aligned}
&\text{(i)} \quad f(x) = \sin 50x \\
&\text{(ii)} \quad f(x) = \sin(\cos(x)) \\
&\text{(iii)} \quad f(x) = (1 - a\sin^2 x)^{1/2} \quad \text{with} \quad a = 0.95 \\
&\text{(iv)} \quad f(x) = (x - 2\pi n) - \pi \text{ for } x \in [2\pi n, 2\pi(n+1)) \\
&\qquad \text{and } n = \ldots, -3, -2, -1, 0, +1, +2, +3, \ldots,
\end{aligned}$$

i.e. $x - \pi \in [0, 2\pi)$ periodically copied. Sample these functions on the uniform grid $x_j = jh$, with $j = 0(1)(N-1)$ and $h = 2\pi/N$. Use the DFT representation combined with the FFT algorithm to approximate $f'(x_j)$. For each function, plot $\log_{10}\left[\max_j \|f'(x_j) - g_j\|\right]$, where g_j is your approximation, as a function of $N = 2^m$ for $m = 4(1)10$ (you might also plot it as a function of m). For each of these functions, discuss the rate of observed convergence, speculate on why each example has the form that it does, and compare and contrast these examples.

Relevant to your discussion should be that (i) is a simple trigonometric function, (ii) is an *entire* function, (iii) has complex singularities near the real axis, and that (iv) is a 2π-periodic function with jump discontinuities.

(**2**) A very common numerical problem is to approximate convolution integrals of the form

$$h(x) = \frac{1}{2\pi}\int_0^{2\pi} f(x-y)\, g(y) dy ,$$

where f and g are 2π-periodic (function of the same period can always be rescaled to this basic period). Given the uniform partition $x_k = kh$ for $k = 0(1)(N-1)$ with $h = 2\pi/N$, the trapezoidal rule approximation is

$$h_k = \frac{1}{N}\sum_{m=0}^{N-1} f_{k-m}\, g_m \text{ for } k = 0(1)(N-1) .$$

(**a**) Prove the discrete version of the Fourier convolution theorem, which states

$$\tilde{h}_j = \tilde{f}_j\, \tilde{g}_j \quad \text{for } j = (-\frac{N}{2} + 1)(1)\frac{N}{2} .$$

Given the discrete Fourier convolution theorem, discuss how the DFT and FFT would be applied to calculate h_k. Discuss the relative advantages of such an approach over “direct summation”, i.e., the straightforward evaluation of the sum. If f and g are smooth periodic functions, discuss the accuracy of this approach.

(**b**) Implement both direct and an FFT based “fast” summation to evaluate h_k. Using

$$f(x) = (1 - a\cos^2 x)^{1/2},\ g(x) = \sin x ,\ \text{with } a = 0.95 ,$$

show from your computations that, for $N = 16$, both methods give the same answer (this is called debugging!). How accurate is the numerical solution? Demonstrate and discuss convergence of the fast method by considering successive doublings of N.

(**3**) The Lorenz system is given by

$$\dot{x} = \sigma(y - x)$$
$$\dot{y} = \rho x - y - xz$$
$$\dot{z} = -\beta z + xy$$

with $\sigma,\ \rho,\ \beta > 0$. In his original paper Lorenz chose $\sigma = 10$, $\beta = 8/3$, and $\rho = 28$, and solved the equations numerically as an initial value problem. Let's do the same. Solve the Lorenz system using the RK4 method with fixed time-step, starting from the initial data $(x_0, y_0, z_0) = (0.1, -0.1, 0.0)$. Demonstrate that your code is fourth-order by applying the ratio test to your computed solution at $t = 1.0$. Using the time-step $h = 0.01$, run the code up to $t = 100$, and plot the orbit $(x(t), y(t), z(t))$ every 0.1 time-units. Rerun the code using the very slightly perturbed initial data $(x_0, y_0, z_0) = (0.10001, -0.10002, 0.0)$. Replot $(x(t), y(t), z(t))$, and plot the difference between these two "approximate" orbits that start from slightly different initial data. Explain and interpret your results.

Now, using the original initial data $(x_0, y_0, z_0) = (0.1, -0.1, 0.0)$, successively halve the time-step from $h = 0.01$, running up to $t = 100$, and saving data at the same time-points (every 0.1 time-units), and examine and discuss the differences between the "approximate" numerical solutions (for different h) over this long time. What does the ratio test tell you? Your own discussion will be enhanced by reading Goodman's discussion of this system.