Lecture 7

In lecture 6 we developed the spectral algorithm to solve the Lippmann-Schwinger integral eq.(L-S) using Chebyshev polynomials as basis functions. A sample program, called "*wave.m*" is included in the website.
Homework 7-1: If it works, play with program *wave.m* by changing the wave number k, the strength of the potential, even the sign of V (to make it repulsive), the range alpha, etc. Try to "dig out" from the program the quantity T described below, and see how it changes for different wave numbers k. If possible, also compare the spectral Chebyshev solution, embodied in *wave.m* with Euler-Cromer's solution of the differential equation $\left(d^2/dr^2 + k^2\right)\psi(r) = V(r)\psi(r)$.

The asymptotic form of the wave function $\psi(r)$ (its mesh-point values are called PSI in the program), obtained by the algorithm, is

$$\psi(r \to \infty) = \sin(kr) + T\cos(kr) \tag{1}$$

where

$$T = -\frac{1}{k}\int_0^{r_{\max}} F(r)V(r)\psi(r)dr \tag{2}$$

The corresponding wave function $\bar{\psi}(r) = \sin(kr + \varphi)$ is

$$\bar{\psi}(r) = \cos(\varphi)\psi(r) \tag{3}$$

where

$$\tan(\varphi) = T \tag{4}$$

Start of Lecture 7

1 Smoothness and spectral accuracy on a uniform mesh.

The rate of convergence of either the Fourier or Chebyshev expansion has been studied in the mathematical literature. The two references used for the present error (or convergence) properties of the Fourier and Chebyshev expansions are respecitvely [1] and [2]

b) Chebyshev differentiation matrices (Chapter 6 of Trefethen)

The mesh is the clustered Chebyshev mesh, the interpolating function is a polynomial of order N, the derivatives of a function $u(x)$ at the meshpoints $x_0,\ x_1, ..x_N$ are given in terms of the column vector $\vec{u} = [u(x_0),\ u(x_1),\ ...u(x_N)]^T$ as

$$\vec{u}' = D_N\ \vec{u}$$

where D_N is the $(N+1) \times (N+1)$ differentiation matrix given on p. 53 of Trefethen. These support points are given by $x_j = \cos(j\pi/N)$, $j = 0, 1, ..N$,

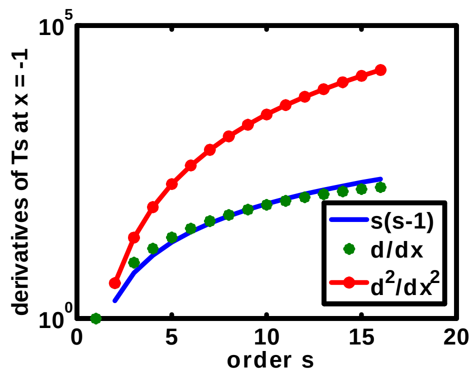

Figure 1: The first and second derivaties with respect to x of $T_s(x)$ for $x = -1$. As expected from Treferthen's derivative matrices, the first order derivative of T_s is proportional to s^2, while the second order derivtive is proportional to s^4

and are not identical to the support points ξ_j which are the zeros of T_{N+1}. The elements of matrix D_N are given below:

$$(D_N)_{00} = \frac{2N^2+1}{6}, \ (D_N)_{NN} = -\frac{2N^2+1}{6} \tag{5}$$

$$(D_N)_{jj} = \frac{-x_j}{2(1-x_j^2)}, \quad j = 1, 2, ..N-1 \tag{6}$$

$$(D_N)_{ij} = \frac{c_i}{c_j} \frac{(-1)^{i+j}}{(x_i - x_j)}, \quad i \neq j, \ i, j = 0, ...N \tag{7}$$

$$c_i = 2 \text{ for } i = 0 \text{ or } N; \quad c_i = 1 \text{ otherwise} \tag{8}$$

Trefethen gives an example in Fig. 11 (program "Output 11") for the function

$$u(x) = \sin(5x)\ e^x, \qquad [-1 \leq x \leq 1]$$

When $N = 10$, the error is in the vicinity of ± 0.02;
for $N = 20$, the error is in the vicinity of $\pm\ 5 \times 10^{-10}$, a big improvement.

However, GR does not like this method of taking derivatives by fitting a polynomial to the function, since when the order is high (say $N = 50$), the derivatives of the polynomial become large and can amplify numerical errors. Indeed, the corner values of the matrix D_N are given by $\pm(2N^2+1)/6$, and for the second derivative factors of order N^4 will be involved. This is illustrated in Fig.1 So, if large values of N are involved it is preferable to interpolate the function u to an equidistant mesh, and then apply the spectral derivative method described in Section I. There the interpolating function is given in terms of sin and cos functions, whose derivatives are easily obtained analytically, and do not involve large powers of x.

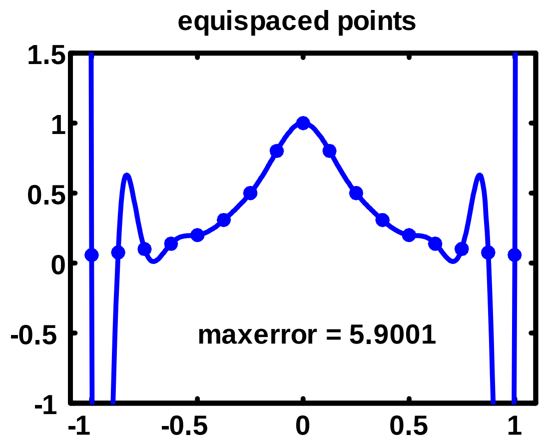

Figure 2: Interpolation of a function $u(x) = 1/(1 + 16\ x^2)$.with $N = 16 + 1$ equispaced points in the interval $x = [-1, 1]$. The program is Program9 of Trefethen, which uses $p = polyfit(x, u, N)$ to obtain the ceofficients of the polynomial $a_0 + a_1 x + ...a_N\ x^N$.The polynomial is then evaluted at all points xx by means of $pp = polyval(p, xx)$. The result $pp(x)$ is plotted by means of the solid line, the max error is shown below

2 Advantage of a non-linear mesh over a linear one, demonstrated by interpolation.

The function being interpolated by means of a polynomial is

$$u(x) = 1/(1 + 16\ x^2), \quad x = [-1, 1]$$

In both cases a polynomial is fitted to all the discrete values of the function, evaluated at the support points. Hence the method is spectral. In one case the points are equidistant, in the other they are clustered according to the Chebyshev method. The result is illustrated in Figs. 2, and 3. They show that non-equispaced points give a much higher accuracy. This property is demonstrated in general by a very interesting argument given by Trefethen in Chapter 5 of his book [1]

References

[1] L. N. Trefethen, *Spectral Methods in MATLAB*, (SIAM, Philadelphia, PA, 2000);

[2] A. Deloff, Ann. Phys. (NY) **322** (2007) 1373–1419.

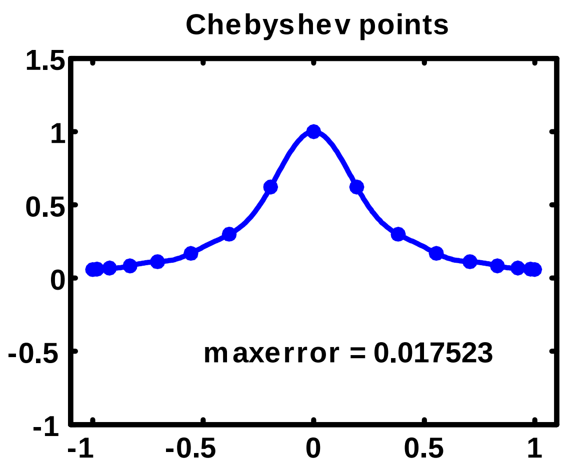

Figure 3: Non-equispaced Chebyshev points in the same interval $[-1, 1]$, interpolating the same function as in Fig (2), but using $N = 16 + 1$ non-equispaced Chebyshev points in $[-1, 1]$, given by $x_j = \cos(j\pi/N)$, $j = 0, 1, ...N$. These points include -1 and $+1$. In our later calculations a different set of Chebyshev points are used, that are the zeros of a Chebyshev function $T_{17}(x)$, which do not include -1 and $+1$., and are spaced slightly differently from the ones above.