

- 1.4 Run Program 1 to $N = 2^{16}$ instead of 2^{12} . What happens to the plot of the error vs. N ? Why? Use the MATLAB commands `tic` and `toc` to generate a plot of approximately how the computation time depends on N . Is the dependence linear, quadratic, or cubic?

Solution.

We can see from these two images that the plot of the error starts to behave differently at the end. This is due to the computations near the limit of the computational precision of the program. We can see that the error starts to go up as N gets closer to this precision after the point of $N = 2^{12}$.

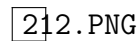
212.PNG

Figure 1: $N = 2^{12}$

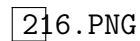
216.PNG

Figure 2: $N = 2^{16}$

Now, as we see by the time graph, the dependence appears to be more linear. Even with the cluster of time on the left, the data seems to be well approximated by a linear function.

tic4.PNG

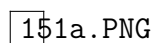
Figure 3: Time vs N

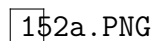
```
1 close all
2 clc
3 % p1.m - convergence of fourth-order finite differences
4
5 % For various N, set up grid in [-pi,pi] and function u(x):
6 Nvec= 2.^(3:16);
7 clf, subplot('position',[.1 .4 .8 .5])
8 for N = Nvec
9     tic
10    h = 2*pi/N; x = -pi + (1:N)*h;
11    u = exp(sin(x)); uprime = cos(x).*u;
12
13    % Construct sparse fourth-order differentiation matrix:
14    e = ones(N,1);
15    D = sparse(1:N,[2:N 1],2*e/3,N,N) ...
16        - sparse(1:N,[3:N 1 2],e/12,N,N);
17    D = (D-D')/h;
18
19    % Plot max(abs(D*u-uprime)):
20    error = norm(D*u-uprime,inf);
21    loglog(N,error, '.', 'markersize',15),
22    hold on;
23    T=toc;
24    %scatter(N,T);
25    %hold on;
26
27 end
28 grid on, xlabel N, ylabel error
29 title('Convergence of fourth-order finite differences')
30 semilogy(Nvec,Nvec.^(-4),'--')
31 text(105,5e-8,'N^{-4}','fontsize',18)
32 %grid on, xlabel N, ylabel Time
33 %title('Computation time for Different N Values')
34
35 % Uncomment out the last two lines in the for loop and the last ...
36 % above while commenting out the loglog line and lines 28-31 to ...
37 % code used to generate the time graph.
```

- 1.5 Run Programs 1 and 2 with $e^{\sin(x)}$ replaced by (a) $e^{\sin^2(x)}$ and (b) $e^{\sin(x)|\sin(x)|}$ and with uprime adjusted appropriately. What rates of convergence do you observe?

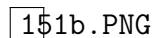
Solution.

a. The rates of convergence seem to be roughly the same as that of the original $e^{\sin(x)}$ function. They might be a little slower, though the convergence is still very much present and similar to that of the original.

151a.PNG

152a.PNG

b. The rates of convergence for $e^{\sin(x)|\sin(x)|}$ are drastically different than that of part a and the original. That is to say, the error tends upward instead of downward as the iterations increase. This is due to discontinuities in the derivative of the function.

151b.PNG

152b.PNG

```

1 clear
2 close all
3 clc
4 % p1.m - convergence of fourth-order finite differences
5
6 % For various N, set up grid in [-pi,pi] and function u(x):
7 Nvec = 2.^(3:12);
8 clf, subplot('position',[.1 .4 .8 .5])
9 for N = Nvec
10     h = 2*pi/N; x = -pi + (1:N)*h;
11     u = exp(sin(x).^2); uprime = 2.*sin(x).*cos(x).*u;
12
13     % Construct sparse fourth-order differentiation matrix:
14     e = ones(N,1);
15     D = sparse(1:N,[2:N 1],2*e/3,N,N)...
16         - sparse(1:N,[3:N 1 2],e/12,N,N);
17     D = (D-D')/h;
18
19     % Plot max(abs(D*u-uprime)):
20     error = norm(D*u-uprime,inf);
21     loglog(N,error, '.', 'markersize',15), hold on
22 end
23 grid on, xlabel N, ylabel error
24 title('Convergence of fourth-order finite differences')
25 semilogy(Nvec,Nvec.^(-4),'--')
26 text(105,5e-8,'N^{-4}','fontsize',18)

```

```

1 clear
2 close all
3 clc
4 % p2.m - convergence of periodic spectral method (compare p1.m)
5
6 % For various N (even), set up grid as before:
7 clf, subplot('position',[.1 .4 .8 .5])
8 for N = 2:2:100;
9     h = 2*pi/N;
10    x = -pi + (1:N)*h;
11    u = exp(sin(x).^2); uprime = 2.*sin(x).*cos(x).*u;
12
13    % Construct spectral differentiation matrix:
14    column = [0 .5*(-1).^(1:N-1).*cot((1:N-1)*h/2)];
15    D = toeplitz(column,column([1 N:-1:2]));
16
17    % Plot max(abs(D*u-uprime)):
18    error = norm(D*u-uprime,inf);
19    loglog(N,error, '.', 'markersize',15), hold on
20 end
21 grid on, xlabel N, ylabel error
22 title('Convergence of spectral differentiation')

```

```

1 clear
2 close all
3 clc
4 % p1.m - convergence of fourth-order finite differences
5
6 % For various N, set up grid in [-pi,pi] and function u(x):
7 Nvec = 2.^(3:12);
8 clf, subplot('position',[.1 .4 .8 .5])
9 for N = Nvec
10     h = 2*pi/N; x = -pi + (1:N)*h;
11     u = exp(sin(x).*abs(sin(x))); uprime = ...
        (sin(x).*sin(2.*x)/(abs(sin(x)))).*u;
12
13     % Construct sparse fourth-order differentiation matrix:
14     e = ones(N,1);
15     D = sparse(1:N,[2:N 1],2*e/3,N,N) ...
        - sparse(1:N,[3:N 1 2],e/12,N,N);
16     D = (D-D')/h;
17
18
19     % Plot max(abs(D*u-uprime)):
20     error = norm(D*u-uprime,inf);
21     loglog(N,error, '.', 'markersize',15), hold on
22 end
23 grid on, xlabel N, ylabel error
24 title('Convergence of fourth-order finite differences')
25 semilogy(Nvec,Nvec.^(-4),'--')
26 text(105,5e-8,'N^{-4}','fontsize',18)

```

```

1 clear
2 close all
3 clc
4 % p2.m - convergence of periodic spectral method (compare p1.m)
5
6 % For various N (even), set up grid as before:
7 clf, subplot('position',[.1 .4 .8 .5])
8 for N = 2:2:100;
9     h = 2*pi/N;
10    x = -pi + (1:N)*h;
11    u = exp(sin(x).*abs(sin(x))); uprime = ...
        (sin(x).*sin(2.*x)/(abs(sin(x)))).*u;
12
13    % Construct spectral differentiation matrix:
14    column = [0 .5*(-1).^(1:N-1).*cot((1:N-1)*h/2)];
15    D = toeplitz(column,column([1 N:-1:2]));
16
17    % Plot max(abs(D*u-uprime)):
18    error = norm(D*u-uprime,inf);
19    loglog(N,error, '.', 'markersize',15), hold on
20 end
21 grid on, xlabel N, ylabel error
22 title('Convergence of spectral differentiation')

```

- 1.6 By manipulating Taylor series, determine the constant C for an error expansion of (1.3) of the form $w_j - u'(x_j) \sim Ch^4 u^{(5)}(x_j)$, where $u^{(5)}$ denotes the fifth derivative. Based on this value of C and on the formula for $u^{(5)}(x)$ with $u(x) = e^{\sin(x)}$ determine the leading term in the expansion for $w_j - u'(x_j)$ for $u(x) = e^{\sin(x)}$ (You will have to find $\max_{x \in [-\pi, \pi]} |u^{(5)}(x)|$ numerically.) Modify Program 1 so that it plots the dashed line corresponding to this leading term rather than just N^{-4} . This adjusted dashed line should fit the data almost perfectly. Plot the difference between the two on a log-log scale and verify that it shrinks at the rate $\mathcal{O}(h^6)$.

Solution.

We are given the differential matrix

$$D = \frac{1}{h} \begin{pmatrix} \ddots & \ddots & \dots & \frac{1}{12} & -\frac{2}{3} \\ \ddots & \ddots & -\frac{1}{12} & \ddots & \frac{1}{12} \\ \ddots & \ddots & \frac{2}{3} & \ddots & \ddots \\ \ddots & \ddots & 0 & \ddots & \ddots \\ \ddots & \ddots & -\frac{2}{3} & \ddots & \ddots \\ -\frac{1}{12} & \ddots & \frac{1}{12} & \ddots & \ddots \\ \frac{2}{3} & -\frac{1}{12} & \dots & \ddots & \ddots \end{pmatrix}$$

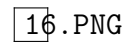
So we can, from 1.3, write the equation

$$w_j = \frac{1}{h} \left(-\frac{1}{12} u_{j-2} + \frac{2}{3} u_{j-1} - \frac{2}{3} u_{j+1} + \frac{1}{12} u_{j+2} \right)$$

Therefore, by expanding each non u_j term about u_j , we get

$$\begin{aligned} w_j &= \frac{1}{h} \left(\frac{1}{12} (-u + 2hu' - 2h^2u'' + \frac{8}{6}h^3u''' - \frac{2}{3}h^4u'''' + \frac{1}{45}h^5u''''') \right. \\ &\quad + \frac{2}{3} (u - hu' + \frac{1}{2}h^2u'' - \frac{1}{6}h^3u''' + \frac{1}{24}h^4u'''' - \frac{1}{120}h^5u''''') \\ &\quad + \frac{2}{3} (-u - hu' - \frac{1}{2}h^2u'' - \frac{1}{6}h^3u''' - \frac{1}{24}h^4u'''' - \frac{1}{120}h^5u''''') \\ &\quad \left. + \frac{1}{12} (u + 2hu' + 2h^2u'' + \frac{8}{6}h^3u''' + \frac{2}{3}h^4u'''' + \frac{1}{45}h^5u''''') \right) \\ &= \frac{1}{h} (-hu' + \frac{1}{30}h^5u''''') = -u' + \frac{1}{30}h^4u'''''' \end{aligned}$$

where u and its derivatives denote $u(x_j)$. Hence, we get a C value of $C = \frac{1}{30}$. Now, taking the max value of $u^{(5)}$ on $[-\pi, \pi]$, we get $\max |u^{(5)}| = 24.811$. Therefore, using

16.PNG

the above equation and C value that we found, we get the following graph where the approximation is almost perfectly in line with the dotted line.


```
1 clear
2 close all
3 clc
4 % p1.m - convergence of fourth-order finite differences
5
6 % For various N, set up grid in [-pi,pi] and function u(x):
7 Nvec = 2.^(3:12);
8 Mvec = zeros(length(Nvec),1);
9 counter = 1;
10 clf, subplot('position',[.1 .4 .8 .5])
11 for N = Nvec
12     h = 2*pi/N; x = -pi + (1:N)*h;
13     u = exp(sin(x)); uprime = cos(x).*u;
14
15     % Construct sparse fourth-order differentiation matrix:
16     e = ones(N,1);
17     D = sparse(1:N,[2:N 1],2*e/3,N,N) ...
18         - sparse(1:N,[3:N 1 2],e/12,N,N);
19     D = (D-D')/h;
20
21     % Plot max(abs(D*u-uprime)):
22     error = norm(D*u-uprime,inf);
23     Mvec(counter)=(1/30).*h^4.*24.811;
24     counter=counter+1;
25     loglog(N,error, '.', 'markersize',15), hold on
26 end
27 grid on, xlabel N, ylabel error
28 title('Convergence of fourth-order finite differences')
29 semilogy(Nvec,Mvec, '--')
```

2.6 We obtained the entries of (1.4) by differentiating the sinc function. Derive the same result by calculating the inverse semidiscrete Fourier transform of $ik\hat{\delta}(k)$.

Solution.

We need to calculate the inverse semidiscrete Fourier transform of the above expression by plugging it into

$$\frac{1}{2\pi} \int_{-\frac{\pi}{h}}^{\frac{\pi}{h}} e^{ikx} \hat{v}(k) dk$$

where $\hat{\delta}(k)$ is given by h . So, we get

$$\frac{1}{2\pi} \int_{-\frac{\pi}{h}}^{\frac{\pi}{h}} e^{ikx} ik\hat{\delta}(k) dk = \frac{2\pi x \cos(\frac{\pi x}{h}) - 2h \sin(\frac{\pi x}{h})}{2\pi x^2}$$

Now, when we differentiate the sinc function $\text{sinc}(\frac{\pi x}{h})$, we arrive at

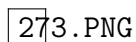
$$\frac{2\pi x \cos(\frac{\pi x}{h}) - 2h \sin(\frac{\pi x}{h})}{2\pi x^2}$$

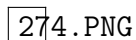
Thus, computing the inverse semidiscrete Fourier transform on $ik\hat{\delta}(k)$ gives the same results as differentiating $\text{sinc}(\frac{\pi x}{h})$.

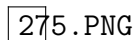
- 2.7 Modify Program 3 to determine the maximum error over \mathbb{R} in the sinc function interpolants of the square wave and the hat function, and to produce a log-log plot of these two error maxima as functions of h . (Good choices for h are 2^{-3} , 2^{-4} , 2^{-5} , 2^{-6} .) What convergence rates do you observe as $h \rightarrow 0$?

Solution.

Here, we have the graphs of the functions with increasing h values, and we can clearly see that the error for the computations decreases as $h \rightarrow 0$. It also seems to settle after $h = 2^{-5}$.

 273.PNG

 274.PNG

 275.PNG

I was not sure how to get the loglog plot to work.

276.PNG

```

1 % Stephanie Klumpe
2 % Problem 2.7 edited code
3
4 clear
5 close all
6 clc
7
8 hval = [2^-3 2^-4 2^-5 2^-6];           %% h-value vector
9 maxe = zeros(2,4);                     %% Max error matrix for ...
    both fxns
10 counter = 1;                           %% Indexing counter
11
12 for h = hval
13
14     xmax = 10; clf
15     x = -xmax:h:xmax;                   % computational grid %% ...
        Estimate?
16     xx = -xmax-h/20:h/10:xmax+h/20;     % plotting grid %% Actual?
17
18     for plt = 1:2
19         switch plt
20             case 1, v = (abs(x) ≤ 3);     % square wave
21                 vv = (abs(xx) ≤ 3);
22
23             case 2, v = max(0,1-abs(x)/3); % hat function
24                 vv = max(0,1-abs(xx)/3);
25         end
26
27         p = zeros(size(xx));
28         for i = 1:length(x)
29             p = p + v(i)*sin(pi*(xx-x(i))/h) ./ (pi*(xx-x(i))/h);
30         end
31
32         maxe(plt,counter) = norm(p-vv,Inf); %% Max error entries
33     end
34
35     counter = counter+1;
36 end
37
38 subplot(2, 1, 1);
39 loglog(hval,maxe(1,:))                  %% Plot of h vs error of sw
40 subplot(2, 1, 2);
41 loglog(hval,maxe(2,:))                  %% Plot of h vs error of hf

```

- 3.5 Using the commands `tic` and `toc`, study the time taken by MATLAB (which is far from optimal in this regard) to compute an N -point FFT, as a function of N . If $N = 2^k$ for $k = 0, 1, \dots, 15$, for example, what is the dependence of the time on N ? What if $N = 500, 501, \dots, 519, 520$? From a plot of the latter results, can you spot the prime numbers in this range? (Hints. The commands `isprime` and `factor` may be useful. To get good `tic/toc` data it may help to compute each FFT 10 or 100 times in a loop.)

Solution.

For N in the first case, we have that the time for the FFT on $2\sin(3x+2)$ decreases rapidly and then slowly increases. For N in the second case, we have that the FFT on the same function continuously decreases in terms of the computational time required.

```
1 clear
2 close all
3 clc
4
5 for k=[0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15]
6     N=2.^k;
7     tic
8     h = 2*pi/N; x = -pi + (1:N)*h;
9     u=2.*sin(3.*x+2);
10    uhat=fft(u,N);
11    toc
12 end
13 fprintf('\n')
14 for N = 500:1:520
15     tic
16     h = 2*pi/N; x = -pi + (1:N)*h;
17     u=2.*sin(3.*x+2);
18     uhat=fft(u,N);
19     toc
20 end
```

- 3.6 We have seen that a discrete function v can be spectrally differentiated by means of two complex FFTs (one forward, one inverse). Explain how two distinct discrete functions v and w can be spectrally differentiated at once by the same two complex FFTs, provided that v and w are real.

Solution.

Let v and w be two real valued discrete functions. Then define the function $u \equiv v + iw$. Then, we have that u is a complex function and we know that we can spectrally differentiate u by the use of the FFT and its inverse. So, by applying the FFT to u , we get $\hat{u} = \hat{v} + i\hat{w}$. We can then spectrally differentiate u and by consequence spectrally differentiate v and w . Succeeding this, we then apply the inverse FFT to \hat{u}' and obtain u' and therefore v' and w' .

- 4.3 (a) Determine the Fourier transform of $u(x) = \frac{1}{(1+x^2)}$. (Use a complex contour integral if you know how; otherwise, copy the result from (4.3).) (b) Determine $\hat{v}(k)$, where v is the discretization of u on the grid $h\mathbb{Z}$. (Hint. Calculating $\hat{v}(k)$ from the definition (2.3) is very difficult.) (c) How fast does $\hat{v}(k)$ approach $\hat{u}(k)$ as $h \rightarrow 0$? (d) Does this result match the predictions of Theorem 3?

Solution.

Using result 4.3 and taking $\sigma = 1$, we have $\hat{u}(k) = \pi e^{-|k|}$ as the Fourier transform of $u(x)$. Now, to determine the semidiscrete Fourier transform of u on the grid $h\mathbb{Z}$, let $v = u(x_j)$ where $x_j \in h\mathbb{Z}$. Using Theorem 2, we have that

$$\hat{v}(k) = \sum_{j=-\infty}^{\infty} \hat{u}\left(k + \frac{2\pi j}{h}\right) = \sum_{j=-\infty}^{\infty} \pi e^{-|k + \frac{2\pi j}{h}|}$$

So, leaving \hat{v} in this form, and by the fact that $u(x)$ has infinitely many continuous derivatives, we see that $|\hat{u} - \hat{v}| = \mathcal{O}(h^m)$ for all $m \in \mathbb{Z}^+$. That is the order of the difference of the two transformations is better than any power of h as $h \rightarrow 0$. This also fits with the predictions of Theorem 3, which is expected.