# 15. Spectral numerical differentiation

## Spectral numerical differentiation

For *continuous* periodic function $f(x)$, $f(x+2\pi) = f(x)$, represented by a Fourier series:

$$f(x) = \sum_{n=-\infty}^{\infty} \hat{f}_n \, e^{inx}$$

The differentiation of $f(x)$ can then be evaluated by:

$$\frac{df(x)}{dx} = \sum_{n=-\infty}^{\infty} \underbrace{(in\hat{f}_n)}_{\text{Fourier coefficient of } f'(x)} \, e^{inx}$$

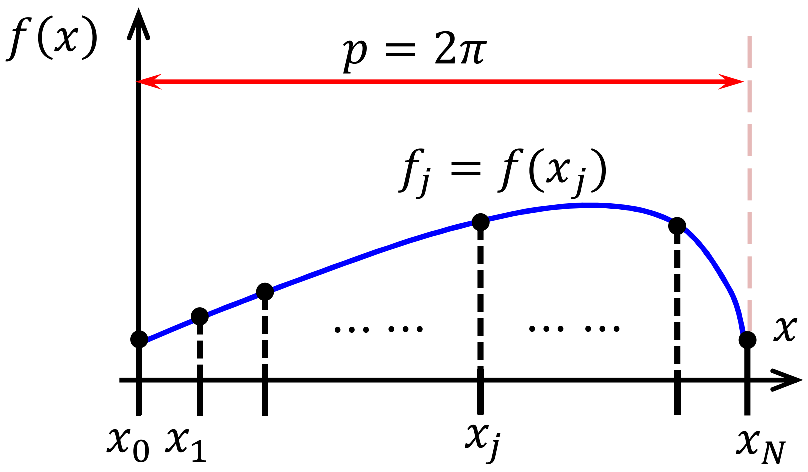
∴ Once the coefficients of the Fourier series $\hat{f}_n$ is obtained, the differentiation can be evaluated by summing the Fourier series with new coefficients $(in\hat{f}_n)$

- Now, for discrete periodic function $f_j$ defined at $x_j, j = 0, 1, \ldots, N-1$:

$$f_j = \sum_{n=-\frac{N}{2}}^{\frac{N}{2}-1} \hat{f}_n\, e^{inx_j} = \sum_{n=0}^{N-1} \hat{f}_n\, e^{inx_j}$$

where $\hat{f}_n$ is the discrete Fourier transform:

$$\hat{f}_n = \frac{1}{N} \sum_{j=0}^{N-1} f(x_j) e^{-inx_j}$$



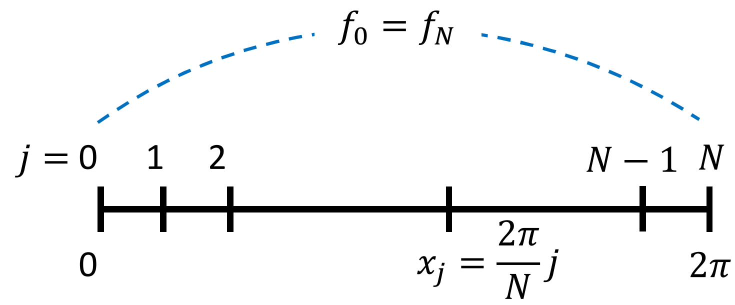

- Can the differentiation of $f$ defined at $x_j$, i.e., $\left.\frac{df}{dx}\right|_j$ be evaluated by:

$$\left.\frac{df}{dx}\right|_j = \sum_{n=-\frac{N}{2}}^{\frac{N}{2}-1} (in) \hat{f}_n\, e^{inx_j}$$

Similarly,

$$\left.\frac{d^2 f}{dx^2}\right|_j = \sum_{n=-\frac{N}{2}}^{\frac{N}{2}-1} (in)^2 \hat{f}_n\, e^{inx_j} = -\sum_{n=-\frac{N}{2}}^{\frac{N}{2}-1} n^2 \hat{f}_n\, e^{inx_j}$$

- **First derivative:**




$$\left.\frac{df}{dx}\right|_j = \sum_{n=-\frac{N}{2}}^{\frac{N}{2}-1} (in\hat{f}_n)\, e^{inx_j}$$

$$= i\left(-\frac{N}{2}\right)\hat{f}_{-\frac{N}{2}}\, e^{i\left(-\frac{N}{2}\right)\frac{2\pi}{N}j} + \sum_{n=-\frac{N}{2}+1}^{-1}(in\hat{f}_n)\, e^{inx_j} + 0 + \sum_{n=1}^{\frac{N}{2}-1}(in\hat{f}_n)\, e^{inx_j}$$

$$= -i\left(\frac{N}{2}\right)\hat{f}_{-\frac{N}{2}}\,(-1)^j + \sum_{m=\frac{N}{2}-1}^{1}(-im\hat{f}_{-m})\, e^{-imx_j} + \sum_{n=1}^{\frac{N}{2}-1}(in\hat{f}_n)\, e^{inx_j}$$

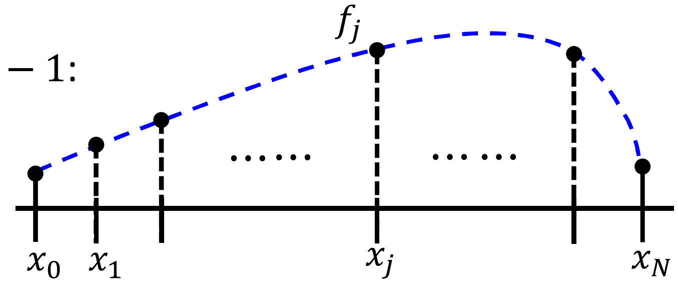
$$= -i\left(\frac{N}{2}\right)\hat{f}_{-\frac{N}{2}}\,(-1)^j + \sum_{n=1}^{\frac{N}{2}-1} in\left(-\hat{f}_n^*\, e^{-inx_j} + \hat{f}_n\, e^{inx_j}\right)$$

$$= \underbrace{-i\left(\frac{N}{2}\right)\hat{f}_{-\frac{N}{2}}\,(-1)^j}_{\text{complex}} + \underbrace{\sum_{n=1}^{\frac{N}{2}-1} in\left[\begin{array}{l}(-\cancel{\hat{f}_n^r\cos nx_j} + \cancel{\hat{f}_n^i\sin nx_j} + \cancel{\hat{f}_n^r\cos nx_j} - \cancel{\hat{f}_n^i\sin nx_j})\\ +i(\hat{f}_n^i\cos nx_j + \hat{f}_n^r\sin nx_j + \hat{f}_n^i\cos nx_j + \hat{f}_n^r\sin nx_j)\end{array}\right]}_{\text{real}}$$

$\because \hat{f}_{-\frac{N}{2}}$ is real

Since the derivative is real value, this complex term must be set to zero.

- Given the discrete periodic function $f_j$ defined at $x_j$, $j = 0, 1, \dots, N-1$:




$$f_j = \sum_{n=-\frac{N}{2}}^{\frac{N}{2}-1} \hat{f}_n \, e^{inx_j}$$

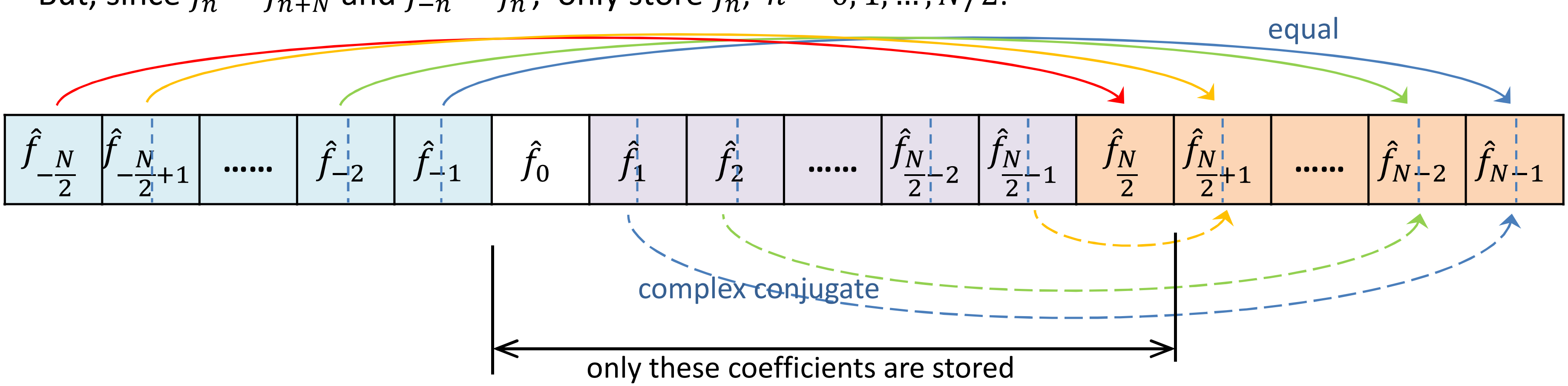
| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $\hat{f}_{-\frac{N}{2}}$ | $\hat{f}_{-\frac{N}{2}+1}$ | ...... | $\hat{f}_{-2}$ | $\hat{f}_{-1}$ | $\hat{f}_0$ | $\hat{f}_1$ | $\hat{f}_2$ | ...... | $\hat{f}_{\frac{N}{2}-2}$ | $\hat{f}_{\frac{N}{2}-1}$ | | | | | |

- The 1st derivative of the discrete periodic function at $x_j$, $j = 0, 1, \dots, N-1$ can be evaluated by:

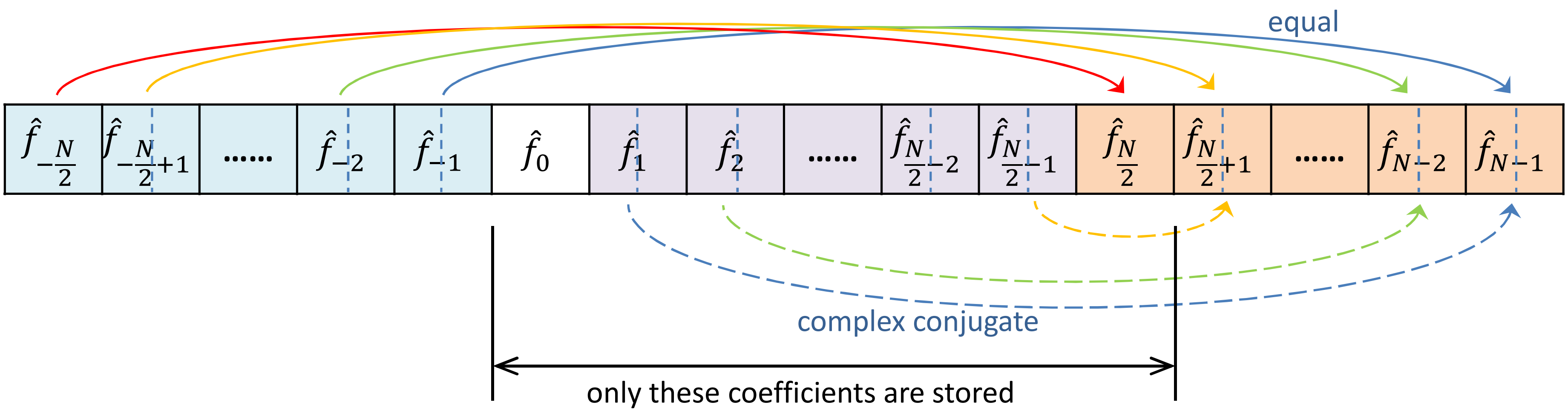
$$\left.\frac{df}{dx}\right|_j = \sum_{n=-\frac{N}{2}+1}^{\frac{N}{2}-1} (in\hat{f}_n) \, e^{inx_j}$$

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $\cancel{\hat{f}_{-\frac{N}{2}}}$ | $i(-\frac{N}{2}+1) \times \hat{f}_{-\frac{N}{2}+1}$ | ...... | $i(-2) \times \hat{f}_{-2}$ | $i(-1) \times \hat{f}_{-1}$ | $0 \times \hat{f}_0$ | $i(1) \times \hat{f}_1$ | $i(2) \times \hat{f}_2$ | ...... | $i(\frac{N}{2}-2) \times \hat{f}_{\frac{N}{2}-2}$ | $i(\frac{N}{2}-1) \times \hat{f}_{\frac{N}{2}-1}$ | | | | | |

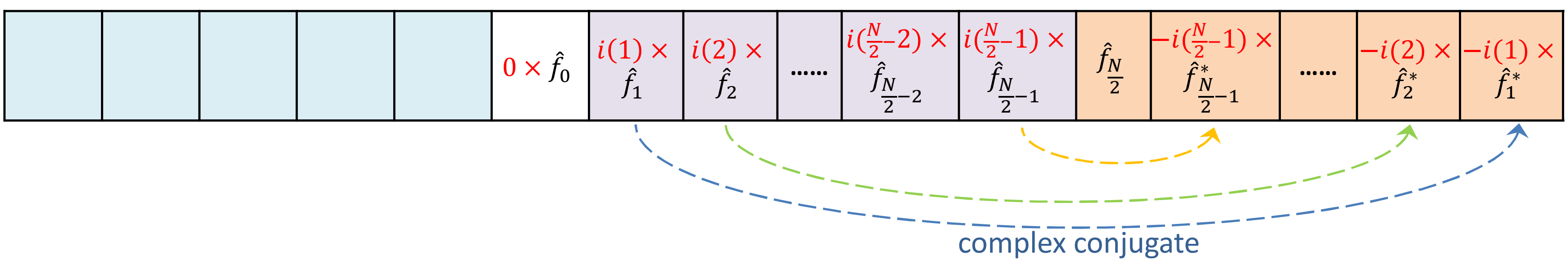
- But, since $\hat{f}_n = \hat{f}_{n+N}$ and $\hat{f}_{-n} = \hat{f}_n^*$, only store $\hat{f}_n$, $n = 0, 1, \dots, N/2$:



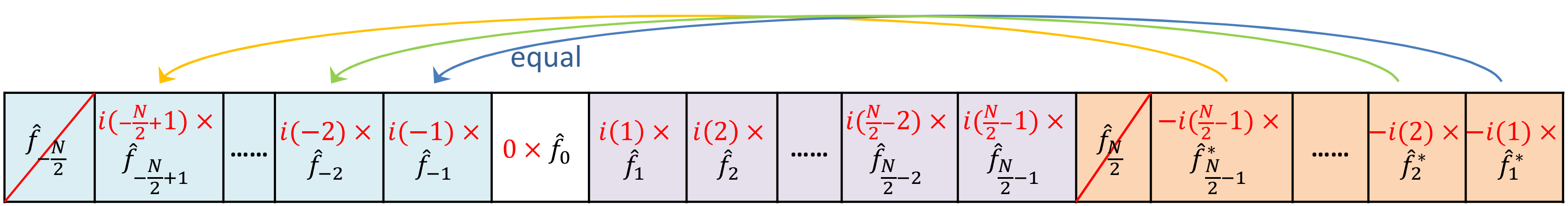
- The discrete Fourier transform $\hat{f}_n$ has the properties: $\hat{f}_0 = \text{real}$, $\hat{f}_{-\frac{N}{2}} = \text{real}$, $\hat{f}_n = \hat{f}_{n+N}$, $\hat{f}_{-n} = \hat{f}_n^*$




- Since $\left(in\hat{f}_n\right)^* = -in\hat{f}_n^*$ , the un-stored DFFT of 1st derivatives are $-in\hat{f}_n^*$:




- Apply the property $\hat{f}_{-n} = \hat{f}_n^*$, we have $i(-n)\hat{f}_{-n} = -in\hat{f}_n^*$:



- **The algorithm to compute first-order differentiation of discrete real $f_j$ using spectral approximation is:**

1. Given $f_j$ at $x_j$, call forward FFT to compute $\hat{f}_n$: $f_j = \sum_{n=-\frac{N}{2}}^{\frac{N}{2}-1} \hat{f}_n\, e^{inx_j} = \sum_{n=0}^{N-1} \hat{f}_n\, e^{inx_j}$

| $j=0$ | $1$ | | | | | | | | | $N-2$ | $N-1$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $f_0$ | $f_1$ | $f_2$ | $f_3$ | ...... | | | | | ...... | $f_{N-2}$ | $f_{N-1}$ |

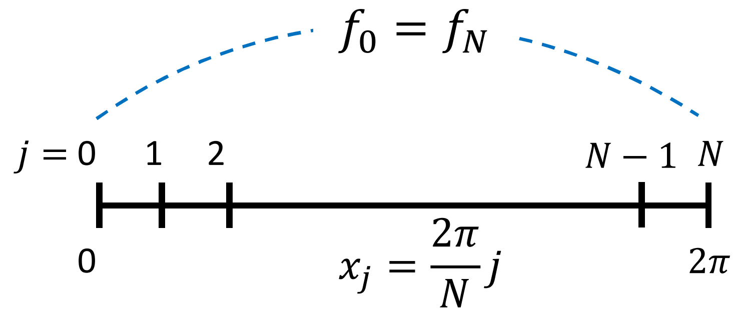
| $n=0$ | $1$ | $1$ | $2$ | $2$ | | | | | $\frac{N}{2}-1$ | $\frac{N}{2}-1$ | $\frac{N}{2}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $\hat{f}_0^r$ | $\hat{f}_1^r$ | $\hat{f}_1^i$ | $\hat{f}_2^r$ | $\hat{f}_2^i$ | ...... | | | ...... | $\hat{f}^r_{\frac{N}{2}-1}$ | $\hat{f}^i_{\frac{N}{2}-1}$ | $\hat{f}^r_{\frac{N}{2}}$ |

2. Compute $\hat{f}_n' = in\hat{f}_n$ , $n = 1, \dots, \frac{N}{2} - 1$, and $\hat{f}'_{-\frac{N}{2}} = \hat{f}'_{\frac{N}{2}} \equiv 0$

| $n=0$ | $1$ | $1$ | $2$ | $2$ | | | | $\frac{N}{2}-1$ | $\frac{N}{2}-1$ | $\frac{N}{2}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| **0** | $-\hat{f}_1^i$ | $\hat{f}_1^r$ | $-2\hat{f}_2^i$ | $2\hat{f}_2^r$ | ...... | | ...... | $-\left(\frac{N}{2}-1\right)\hat{f}^i_{\frac{N}{2}-1}$ | $\left(\frac{N}{2}-1\right)\hat{f}^r_{\frac{N}{2}-1}$ | **0** |

3. Call backward FFT to compute $\left.\frac{df}{dx}\right|_j = \sum_{n=-\frac{N}{2}}^{\frac{N}{2}-1} \hat{f}_n'\, e^{inx_j}$

- **Second derivative:**




$$\left.\frac{d^2 f}{dx^2}\right|_j = \sum_{n=-\frac{N}{2}}^{\frac{N}{2}-1} (in)^2 \hat{f}_n \, e^{inx_j}$$

$$= -\left(-\frac{N}{2}\right)^2 \hat{f}_{-\frac{N}{2}} \, e^{i\left(-\frac{N}{2}\right)\frac{2\pi}{N}j} + \sum_{n=-\frac{N}{2}+1}^{-1} (in)^2 \hat{f}_n \, e^{inx_j} + {\color{red}0} + \sum_{n=1}^{\frac{N}{2}-1} (in)^2 \hat{f}_n \, e^{inx_j}$$

$$= -\frac{N^2}{4} \hat{f}_{-\frac{N}{2}} (-1)^j + \sum_{m=\frac{N}{2}-1}^{1} (-im)^2 \hat{f}_{-m} \, e^{-imx_j} + \sum_{n=1}^{\frac{N}{2}-1} (in)^2 \hat{f}_n \, e^{inx_j}$$

$$= -\frac{N^2}{4} \hat{f}_{-\frac{N}{2}} (-1)^j + \sum_{n=1}^{\frac{N}{2}-1} (-n^2)\left(\hat{f}_n^* \, e^{-inx_j} + \hat{f}_n \, e^{inx_j}\right)$$

$$= -\frac{N^2}{4} \hat{f}_{-\frac{N}{2}} (-1)^j + \sum_{n=1}^{\frac{N}{2}-1} (-n^2) \left[\begin{array}{c} (\hat{f}_n^r \cos nx_j - \hat{f}_n^i \sin nx_j + \hat{f}_n^r \cos nx_j - \hat{f}_n^i \sin nx_j) \\ +i(\cancel{-\hat{f}_n^i \cos nx_j} - \cancel{\hat{f}_n^r \sin nx_j} + \cancel{\hat{f}_n^i \cos nx_j} + \cancel{\hat{f}_n^r \sin nx_j}) \end{array}\right]$$

$= {\color{red}\text{real}}$

- **The algorithm to compute second-order differentiation of discrete real $f_j$ using spectral approximation is:**

1. Given $f_j$ at $x_j$, call forward FFT to compute $\hat{f}_n$: $\quad f_j = \sum_{n=-\frac{N}{2}}^{\frac{N}{2}-1} \hat{f}_n \, e^{inx_j}$

| $j=0$ | $1$ | | | | | | | | | $N-2$ | $N-1$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $f_0$ | $f_1$ | $f_2$ | $f_3$ | …… | | | | | …… | $f_{N-2}$ | $f_{N-1}$ |

| $n=0$ | $1$ | $1$ | $2$ | $2$ | | | | | $\frac{N}{2}-1$ | $\frac{N}{2}-1$ | $\frac{N}{2}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $\hat{f}_0^r$ | $\hat{f}_1^r$ | $\hat{f}_1^i$ | $\hat{f}_2^r$ | $\hat{f}_2^i$ | …… | $\hat{f}_n^r$ | $\hat{f}_n^i$ | …… | $\hat{f}_{\frac{N}{2}-1}^r$ | $\hat{f}_{\frac{N}{2}-1}^i$ | $\hat{f}_{\frac{N}{2}}^r$ |

2. Compute $\hat{f}_n'' = (in)^2 \hat{f}_n = -n^2 \hat{f}_n, \quad n = 1, \ldots, \frac{N}{2} - 1$

| $n=0$ | $1$ | $1$ | $2$ | $2$ | | | | | $\frac{N}{2}-1$ | $\frac{N}{2}-1$ | $\frac{N}{2}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $\hat{f}_0^r$ | $-\hat{f}_1^r$ | $-\hat{f}_1^i$ | $-2^2\hat{f}_2^r$ | $-2^2\hat{f}_2^i$ | …… | $-n^2\hat{f}_n^r$ | $-n^2\hat{f}_n^i$ | …… | | | $-\left(\frac{N}{2}\right)^2 \hat{f}_{\frac{N}{2}}^r$ |

3. Call backward FFT to compute $\quad \left.\frac{d^2 f}{dx^2}\right|_j = \sum_{n=-\frac{N}{2}}^{\frac{N}{2}-1} \hat{f}_n'' \, e^{inx_j}$

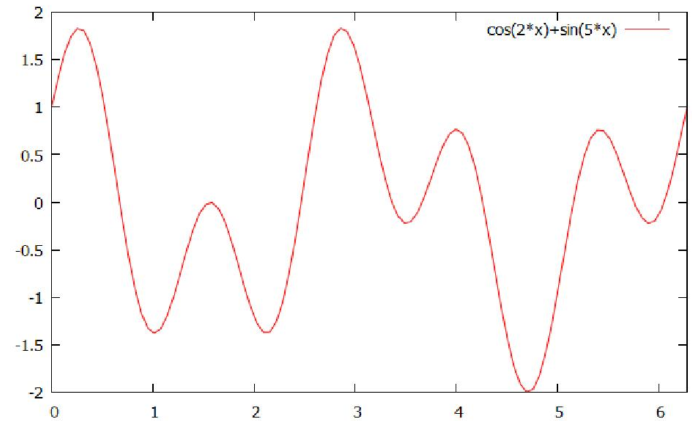
## Notes about spectral numerical differentiation:

- The spectral derivative is much more accurate than any finite-difference schemes for *periodic functions*.
- The major cost involved is the use of fast Fourier transform.
- However, it is inaccurate and does not converge when the derivative is discontinuous.

## ➔ Example of computing first derivative of a real function using discrete Fourier transform and calling NCAR FFTPACK

```
> gfortran  t_specderiv.f90  NCAR_fft.f
```

$f(x) = \cos(2x) + \sin(5x)$




```
program t_specderiv
! Example of computing first derivative of a real function
! using discrete spectral transform and calling NCAR FFTPACK

implicit none
integer,parameter :: nn = 16
real :: x(0:nn-1), f(0:nn-1), df(0:nn-1)
real :: trig(2*nn+15), tmp
real,parameter :: pi = acos(-1.0)
integer :: i, ii

!-- Give values: f(x) = cos2x + sin5x
do i = 0, nn-1
  x(i)  = 2.0*pi/nn*i
  f(i)  = cos(2.0*x(i))+sin(5.0*x(i))
  df(i) = -2.0*sin(2.0*x(i))+5.0*cos(5.0*x(i))
end do

!-- Forward transform to compute the complex coefficients
call rffti(nn,trig)
call rfftf(nn,f,trig)

!-- Set 0 to the coefficient of nn/2 mode
f(nn-1) = 0.0
```

(continued)

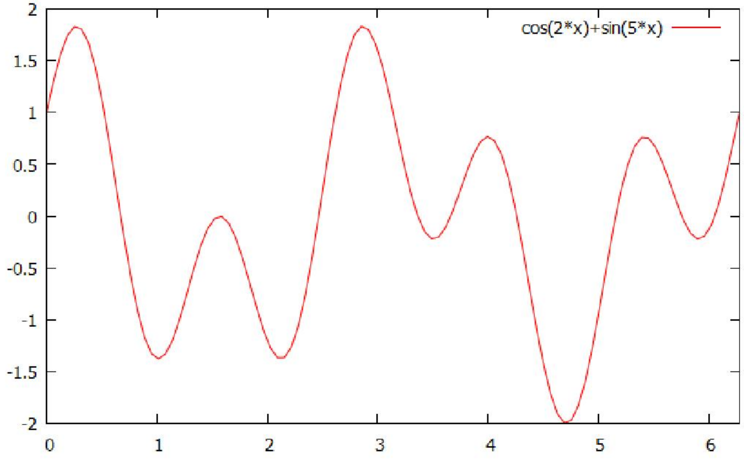
```
!-- Multiply and swap the Fourier coefficients for first derivative
ii = 1
do i = 1, nn-3, 2
  tmp = -ii*f(i+1)
  f(i+1) = ii*f(i)
  f(i)   = tmp
  ii = ii + 1
end do
f = f/nn

!-- Backward transform
call rfftb(nn,f,trig)

!-- Output and compare with exact values
write(*,*) '          j    spectral        exact'
do i = 0, nn-1
  write(*,*) i, f(i), df(i)
end do
write(*,*) ''
write(*,*) 'Max error: ', maxval(abs(f-df))

end program
```

**output:**

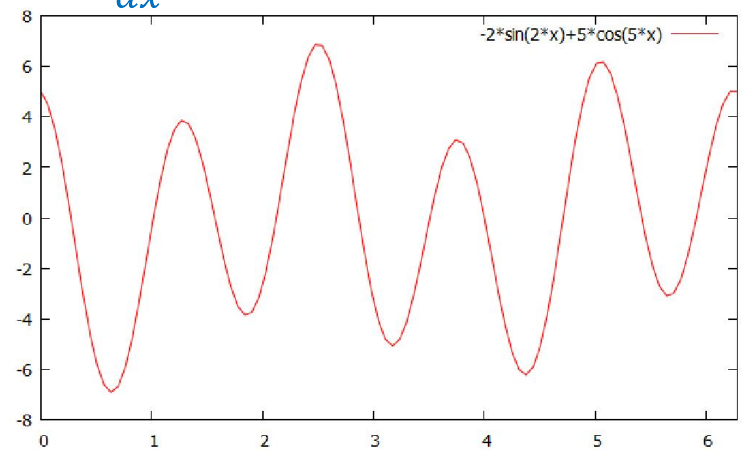
| j | spectral | exact |
|---|---|---|
| 0 | 4.9999995 | 5.0000000 |
| 1 | -3.3276315 | -3.3276312 |
| 2 | -5.5355334 | -5.5355330 |
| 3 | 3.2051830 | 3.2051840 |
| 4 | -2.86202066E-08 | -1.51398558E-06 |
| 5 | -3.2051840 | -3.2051833 |
| 6 | 5.5355349 | 5.5355339 |
| 7 | 3.3276296 | 3.3276265 |
| 8 | -5.0000024 | -5.0000005 |
| 9 | 0.49920809 | 0.49920601 |
| 10 | 1.5355268 | 1.5355327 |
| 11 | -6.0336103 | -6.0336146 |
| 12 | 6.30433988E-06 | 3.45821547E-07 |
| 13 | 6.0336065 | 6.0336118 |
| 14 | -1.5355327 | -1.5355399 |
| 15 | -0.49920332 | -0.49920550 |

Max error: 7.15255737E-06

$$f(x) = \cos(2x) + \sin(5x)$$



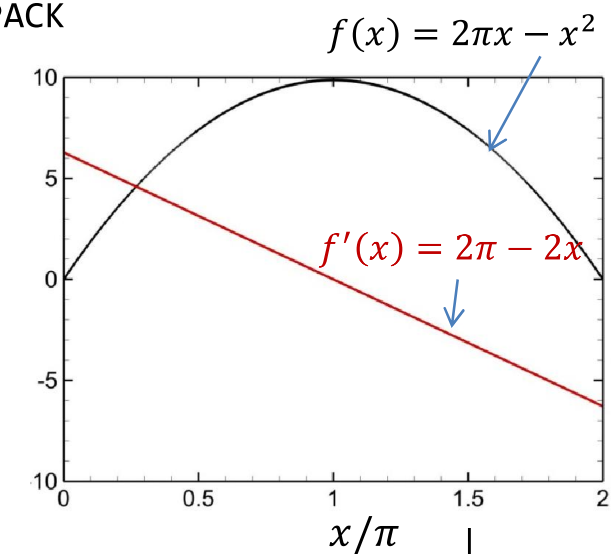

$$\frac{df(x)}{dx} = -2\sin(2x) + 5\cos(5x)$$



➔ **Example of computing first derivative of a real function using discrete Fourier transform when the derivatives at the periodic boundaries are not continuous**

```
> gfortran t_specderiv_2.f90 NCAR_fft.f
```

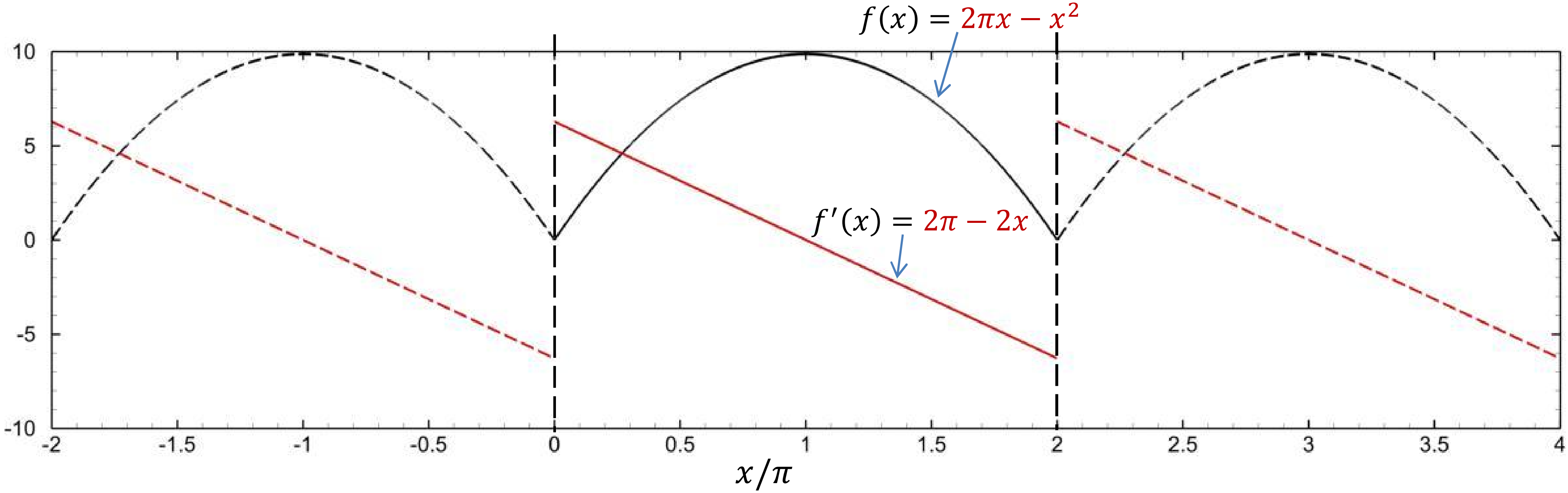
```
program t_specderiv_2
! Example of computing first derivative of a real function
! using discrete spectral transform and calling NCAR FFTPACK
!
! > gfortran t_specderiv_2.f90 NCAR_fft.f

implicit none
integer :: nn
real,allocatable,dimension(:) :: x, f, df, trig
real :: tmp
real,parameter :: pi = acos(-1.0)
integer :: i, ii

!-- Input number of grids & allocate arraies
write(*,*) "number of grid n=?"
read(*,*) nn
allocate (x(0:nn-1), f(0:nn-1), df(0:nn-1), trig(2*nn+15))

!-- Give values: f(x) = 2.*pi*x -x**2
do i = 0, nn-1
  x(i)  = 2.0*pi/nn*i
  f(i)  = 2.*pi*x(i) -x(i)**2
  df(i) = 2.*pi -2.*x(i)
end do
```



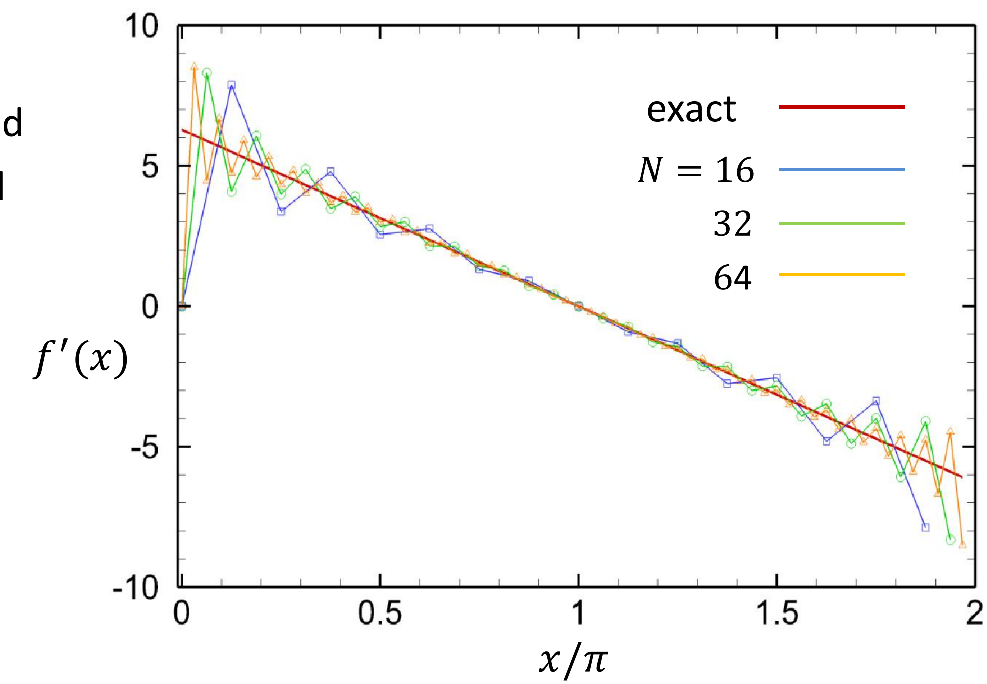

(continued)

```
!-- Forward transform
call rffti(nn,trig)
call rfftf(nn,f,trig)

!-- Set 0 to 0 & nn/2 modes
f(0) = 0.0
f(nn-1) = 0.0

!-- Multiply and swap the Fourier coefficients
ii = 0
do i = 1, nn-3, 2
  ii = ii + 1
  tmp =   -ii*f(i+1)
  f(i+1) = ii*f(i)
  f(i)   = tmp
end do
f = f/nn

!-- Backward transform
call rfftb(nn,f,trig)

!-- Output the derivatives by spectral method and compare with exact values
write(*,*) '          j       x            spectral         exact'
do i = 0, nn-1
  write(*,*) i, x(i)/pi, f(i), df(i)
end do

write(*,*) ''
write(*,*) 'Max error: ', maxval(abs(f-df))

end program
```

- Although $f(x)$ is continuous at the two boundaries $x = 0$ and $2\pi$, its derivative $f'(x)$ is not.



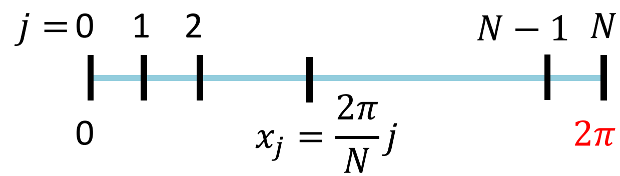

- The discontinuity of $f'(x)$ at $x = 0$ and $2\pi$ results in highly oscillatory spectral approximation near the boundaries.



# Discrete Fourier Transform with a period $L$

- **Discrete Fourier Transform with a period $2\pi$ :** $f(x) = f(x + 2\pi)$

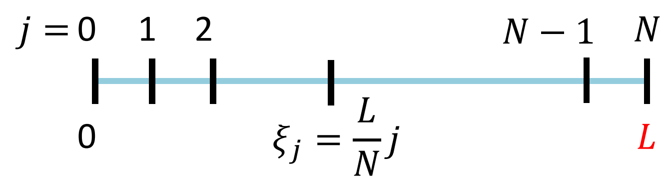
$$f_j = \sum_{n=-\frac{N}{2}}^{\frac{N}{2}-1} \hat{f}_n \, e^{inx_j} = \sum_{n=-\frac{N}{2}}^{\frac{N}{2}-1} \hat{f}_n \, e^{in\frac{2\pi}{N}j}$$




$$\hat{f}_n = \frac{1}{N} \sum_{j=0}^{N-1} f_j \, e^{-inx_j} = \frac{1}{N} \sum_{j=0}^{N-1} f_j \, e^{-in\frac{2\pi}{N}j}$$

- **For $f$ with a general period $L$:** $f(\xi) = f(\xi + L)$

$$\xi \equiv \frac{L}{2\pi} x \qquad x \equiv \frac{2\pi}{L} \xi$$




$$f_j = \sum_{n=-\frac{N}{2}}^{\frac{N}{2}-1} \hat{f}_n \, e^{inx_j} = \sum_{n=-\frac{N}{2}}^{\frac{N}{2}-1} \hat{f}_n \, e^{in\frac{2\pi}{L}\xi_j} = \sum_{n=-\frac{N}{2}}^{\frac{N}{2}-1} \hat{f}_n \, e^{in\frac{2\pi}{L}\frac{L}{N}j} = \sum_{n=-\frac{N}{2}}^{\frac{N}{2}-1} \hat{f}_n \, e^{in\frac{2\pi}{N}j}$$

$$\hat{f}_n = \frac{1}{N} \sum_{j=0}^{N-1} f_j \, e^{-inx_j} = \frac{1}{N} \sum_{j=0}^{N-1} f_j \, e^{-in\frac{2\pi}{L}\xi_j} = \frac{1}{N} \sum_{j=0}^{N-1} f_j \, e^{-in\frac{2\pi}{L}\frac{L}{N}j} = \frac{1}{N} \sum_{j=0}^{N-1} f_j \, e^{-in\frac{2\pi}{N}j}$$

- **The discrete Fourier transform pair has the same form regardless the period.**

- But to use spectral approximation to compute the derivative of $f$ with a period $L$,

$$f(\xi) = f(\xi + L)$$

$$x \equiv \frac{2\pi}{L}\xi$$

$$f(x) = f(x + 2\pi)$$

$$\Rightarrow \frac{df}{d\xi} = \frac{df}{dx}\frac{dx}{d\xi} = \frac{2\pi}{L}\frac{df}{dx}$$

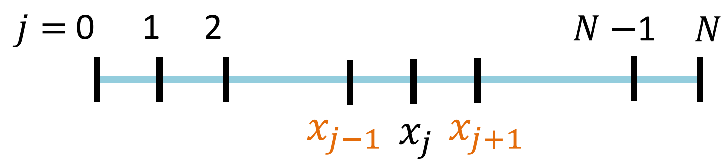
wavenumber coefficient when the period is $L$

## An Alternative Measure for the Accuracy of Finite Difference

- **Common measure:**

e.g., 2nd order finite-difference scheme mesh refinement by a factor of 2 improves the accuracy by fourfold:

$$\left.\frac{df}{dx}\right|_j = \frac{f_{j+1} - f_{j-1}}{2\Delta} + \mathcal{O}(\Delta^2) \equiv \left.\frac{\delta f}{\delta x}\right|_j + \mathcal{O}(\Delta^2)$$




$$\Delta = \frac{L}{N} = x_j - x_{j-1}$$

- **Alternative measure:**
  - Modified wavenumber approach.
  - Measure how well a finite-difference scheme differentiates sinusoidal functions.

Consider pure harmonic functions of periods $L/n$:

$$f(x) = e^{i\kappa x} \qquad \boxed{\kappa = \frac{2\pi}{L} n} \qquad n = 0, 1, 2, \cdots, \frac{N}{2}$$

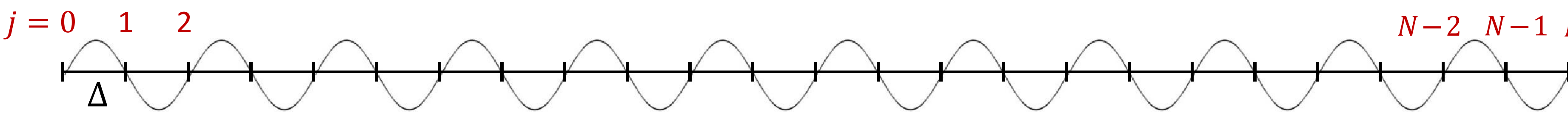
The exact derivative: $\dfrac{df}{dx} = i\kappa f$

($k$ vs. $\kappa$)

$$f(x) = e^{i\kappa x} = e^{i\frac{2\pi}{L}nx} \qquad n = 0, 1, 2, \cdots, \frac{N}{2}$$

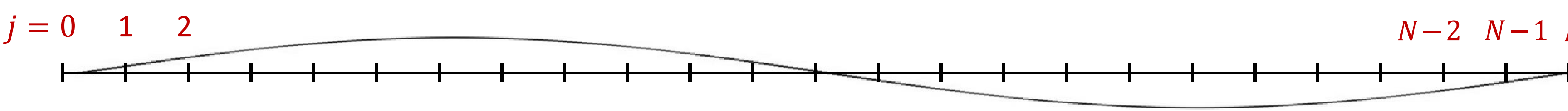
The shortest wave $f(x)$ represents: $n = \frac{N}{2}$ $\quad f(x) = e^{i\frac{2\pi}{L}\frac{N}{2}x} = e^{i2\pi\frac{N}{2L}x}$

The shortest wavelength $= \frac{2L}{N} = 2\Delta$




The longest wave $f(x)$ represents: $n = 1$ $\quad f(x) = e^{i\frac{2\pi}{L}1x}$

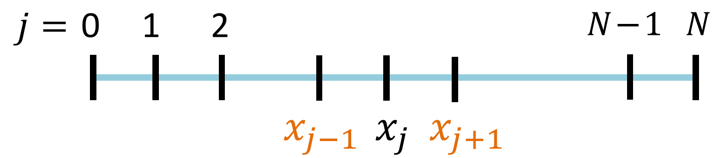
The longest wavelength $= L$



- If $f(x)$ is defined at discrete grid $x_j$ :

$$x_j = \frac{L}{N} j \qquad j = 0, 1, 2, \cdots, \mathrm{N} - 1$$

$$f_j = f(x_j) = e^{i\kappa x_j} = e^{i\frac{2\pi}{L} n \cdot \frac{L}{N} j} = e^{i2\pi\frac{nj}{N}}$$




$$\Delta = \frac{L}{N}$$

- The second-order finite difference approximation :

$$\left.\frac{\delta f}{\delta x}\right|_j = \frac{f_{j+1} - f_{j-1}}{2\Delta}$$

$$= \frac{e^{i2\pi\frac{n(j+1)}{N}} - e^{i2\pi\frac{n(j-1)}{N}}}{2\Delta} = \frac{e^{i2\pi\frac{nj}{N}}\left(e^{i2\pi\frac{n}{N}} - e^{-i2\pi\frac{n}{N}}\right)}{2\Delta} = \frac{\left(e^{i2\pi\frac{n}{N}} - e^{-i2\pi\frac{n}{N}}\right)}{2\Delta} f_j = \frac{i \sin\left(2\pi\frac{n}{N}\right)}{\Delta} f_j$$

$$= i\kappa' f_j$$

$$\boxed{\kappa' \equiv \frac{\sin\left(\frac{2\pi n}{N}\right)}{\Delta} = \frac{\sin\left(\frac{2\pi n}{L}\frac{L}{N}\right)}{\Delta} = \frac{\sin(\kappa\Delta)}{\Delta}}$$

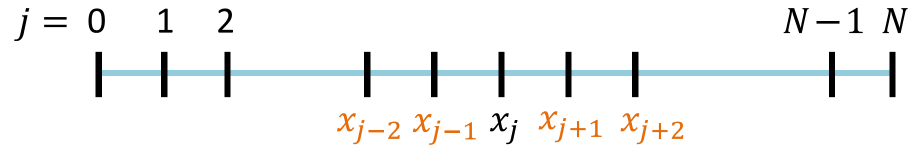
- Exact differention:

$$\frac{df}{dx} = i\kappa f$$

$$\boxed{\kappa = \frac{2\pi}{L} n}$$

- $\kappa'$ is called the *modified wavenumber* for the 2nd-order central difference scheme.
- A measure of accuracy of a finite difference scheme is provided by comparing the modified wavenumber $\kappa'$ with $\kappa$.

The fourth-order finite approximation :




$$f_j = f(x_j) = e^{i\kappa x_j}$$

$$= e^{i\frac{2\pi}{L}n\cdot\frac{L}{N}j}$$

$$= e^{i2\pi\frac{nj}{N}}$$

$$\left.\frac{\delta f}{\delta x}\right|_j = \frac{4}{3}\frac{f_{j+1}-f_{j-1}}{2\Delta} - \frac{1}{3}\frac{f_{j+2}-f_{j-2}}{4\Delta}$$

$$= \frac{4}{3}\frac{\left(e^{i2\pi\frac{n(j+1)}{N}} - e^{i2\pi\frac{n(j-1)}{N}}\right)}{2\Delta} - \frac{1}{3}\frac{\left(e^{i2\pi\frac{n(j+2)}{N}} - e^{i2\pi\frac{n(j-2)}{N}}\right)}{4\Delta}$$

$$= \frac{4}{3}e^{i2\pi\frac{nj}{N}}\frac{\left(e^{i2\pi\frac{n}{N}} - e^{-i2\pi\frac{n}{N}}\right)}{2\Delta} - \frac{1}{3}e^{i2\pi\frac{nj}{N}}\frac{\left(e^{i2\pi\frac{2n}{N}} - e^{-i2\pi\frac{2n}{N}}\right)}{4\Delta}$$

$$= \frac{4}{3}\frac{i\sin\left(2\pi\frac{n}{N}\right)}{\Delta}f_j - \frac{1}{3}\frac{i\sin\left(2\pi\frac{2n}{N}\right)}{2\Delta}f_j$$

$$= i\left(\frac{4}{3}\frac{\sin\left(2\pi\frac{n}{N}\right)}{\Delta} - \frac{1}{3}\frac{\sin\left(2\pi\frac{2n}{N}\right)}{2\Delta}\right)f_j$$

$$= i\kappa' f_j$$

$$\boxed{\kappa' = \frac{4}{3}\frac{\sin\left(2\pi\frac{n}{N}\right)}{\Delta} - \frac{1}{3}\frac{\sin\left(2\pi\frac{2n}{N}\right)}{2\Delta} = \frac{4}{3}\frac{\sin(\kappa\Delta)}{\Delta} - \frac{1}{3}\frac{\sin(2\kappa\Delta)}{2\Delta}}$$

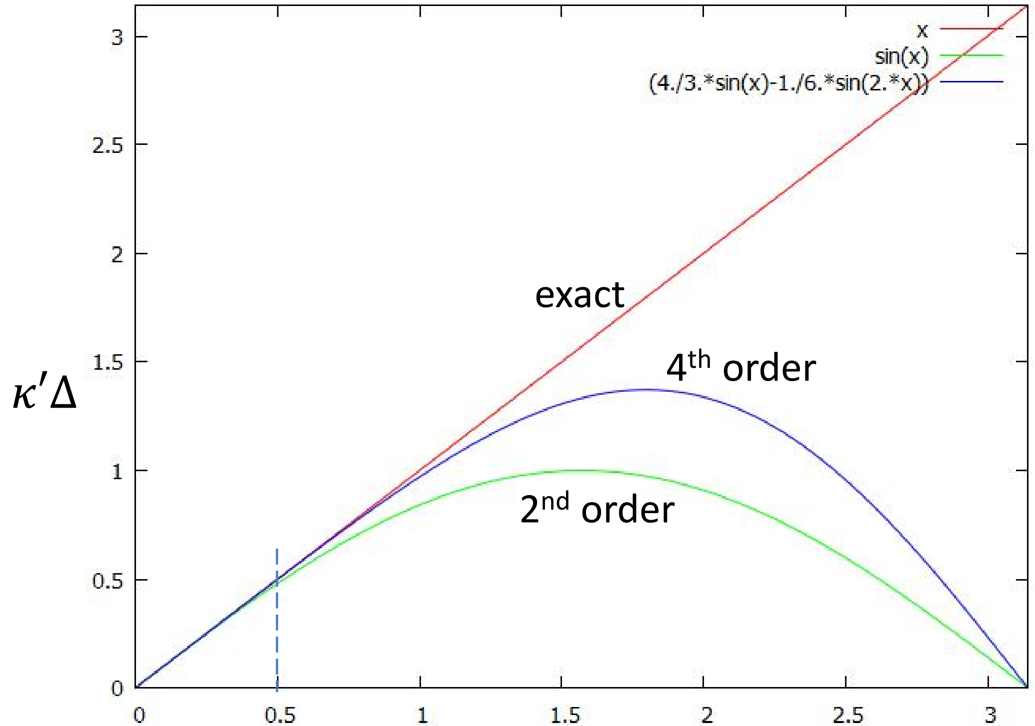
- Modified wavenumber of

  $2^{nd}$ order finite-difference scheme:

$$\kappa' = \frac{\sin(\kappa\Delta)}{\Delta}$$

  $4^{th}$ order finite-difference scheme:

$$\kappa' = \frac{4}{3}\frac{\sin(\kappa\Delta)}{\Delta} - \frac{1}{3}\frac{\sin(2\kappa\Delta)}{2\Delta}$$




$$\kappa\Delta = \frac{2\pi}{L} n \frac{L}{N} = 2\pi\frac{n}{N}$$

- The modified wavenumber $\kappa'$ is in good agreement with the exact wavenumber $\kappa$ at small value of $\kappa\Delta$, i.e., when $f$ is slowly varying.

- For higher $\kappa\Delta$, i.e., when $f$ varies rapidly, the finite-difference scheme provides a poor approximation.

- For $2^{nd}$ order finite-difference scheme, to have accurate approximation of the $1^{st}$ derivative:

$$\kappa\Delta < 0.5 \quad \Longrightarrow \quad \frac{2\pi}{L}\Delta < 0.5 \quad \Longrightarrow \quad \Delta < \frac{L}{4\pi} \quad \Longrightarrow \quad \frac{L}{\Delta} > 4\pi$$

  i.e., the number of grids per wavelength must be more than $4\pi \approx 12$

- The number of grids needed for simulating such a wavy flow of multiple scales will be enormous when using finite-difference scheme!

- This justifies the use of spectral numerical differentiation.

# Exercise

- Write two subroutines which compute the *first* and *second* derivatives of a real discrete function sampled at NUM equal spacing grids.
- Use the 1-D real FFT routine in FFTW.
- Test your subroutines by comparing with the derivatives of known functions, e.g., $f(x) = \sin(\cos(x))^3$.
- Note that the array storage arrangement of DFT in FFTW is:

| $n=0$ | $1$ | $2$ | | | | | $\frac{N}{2}-1$ | $\frac{N}{2}$ | $\frac{N}{2}-1$ | | | | | $2$ | $1$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $\hat{f}_0^r$ | $\hat{f}_1^r$ | $\hat{f}_2^r$ | …… | | | …… | $\hat{f}_{\frac{N}{2}-1}^r$ | $\hat{f}_{\frac{N}{2}}^r$ | $\hat{f}_{\frac{N}{2}-1}^i$ | …… | | | …… | $\hat{f}_2^i$ | $\hat{f}_1^i$ |