

# CS711008Z Algorithm Design and Analysis

## Lecture 5. FFT and DIVIDE AND CONQUER

Dongbo Bu

Institute of Computing Technology  
Chinese Academy of Sciences, Beijing, China

- DFT: evaluate a polynomial at  $n$  special points;
- FFT: an efficient implementation of DFT;
- Applications of FFT: multiplying two polynomials (and multiplying two  $n$ -bits integers); time-frequency transform; solving partial differential equations;
- Appendix: relationship between continuous and discrete Fourier transforms.

# DFT: Discrete Fourier Transform

- DFT evaluates a polynomial  $A(x) = a_0 + a_1x + \dots + a_{n-1}x^{n-1}$  at  $n$  distinct points  $1, \omega, \omega^2, \dots, \omega^{n-1}$ , where  $\omega = e^{\frac{2\pi}{n}i}$  is the  $n$ -th complex root of unity.
- Thus, it transforms the complex vector  $a_0, a_1, \dots, a_{n-1}$  into another complex vector  $y_0, y_1, \dots, y_{n-1}$ , where  $y_k = A(\omega^k)$ , i.e.,

$$\begin{array}{ccccccccc} y_0 & = & a_0 & & + & a_1 & & + & a_2 & & \dots & + & a_{n-1} \\ y_1 & = & a_0 & & + & a_1\omega^1 & & + & a_2\omega^2 & & \dots & + & a_{n-1}\omega^{n-1} \\ \dots & & \dots & & & \dots & & & \dots & & \dots & & \dots \\ y_{n-1} & = & a_0 & & + & a_1\omega^{n-1} & & + & a_2\omega^{2(n-1)} & & \dots & + & a_{n-1}\omega^{(n-1)^2} \end{array}$$

- Matrix form:

$$\begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_{n-1} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & \omega^1 & \omega^2 & \dots & \omega^{n-1} \\ 1 & \omega^2 & \omega^4 & \dots & \omega^{2(n-1)} \\ \dots & \dots & \dots & \dots & \dots \\ 1 & \omega^{n-1} & \omega^{2(n-1)} & \dots & \omega^{(n-1)^2} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_{n-1} \end{bmatrix}$$

# FFT: a fast way to implement DFT [Cooley and Tukey, 1965]

- Direct matrix-vector multiplication requires  $O(n^2)$  operations when using the Horner's method, i.e.,

$$A(x) = a_0 + x(a_1 + x(a_2 + \dots + xa_{n-1})).$$

- FFT: reduce  $O(n^2)$  to  $O(n \log_2 n)$  using divide-and-conquer technique.
- How does FFT achieve this? Or what calculations are redundant in the direct matrix-vector multiplication approach?
- Note: The idea of FFT was proposed by James Cooley and John Tukey in 1965 when analyzing earth-quake data, but the idea can be dated back to F. Gauss.

## Let's evaluate $A(x)$ at four special points

- It is easy to evaluate any 1-degree polynomial  $A(x) = a_0 + a_1x$  at two points  $1, -1$ . Now let's evaluate a 3-degree polynomial  $A(x) = a_0 + a_1x + a_2x^2 + a_3x^3$  at four special points  $1, i, -1, -i$ .
- **Divide:** Break the polynomial into even and odd terms, i.e.,
  - $A_{\text{even}}(x) = a_0 + a_2x^2$
  - $A_{\text{odd}}(x) = a_1x + a_3x^3$

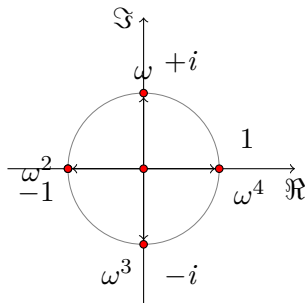
Then we have the following equations:

- $A(x) = (a_0 + a_2x^2) + x(a_1 + a_3x^2) = A_{\text{even}}(x^2) + xA_{\text{odd}}(x^2)$
- $A(-x) = (a_0 + a_2x^2) - x(a_1 + a_3x^2) = A_{\text{even}}(x^2) - xA_{\text{odd}}(x^2)$
- **Combine:** For the 4 special points  $1, i, -1, -i$ , we have
  - $A(1) = A_{\text{even}}(1) + A_{\text{odd}}(1)$
  - $A(i) = A_{\text{even}}(-1) + iA_{\text{odd}}(-1)$
  - $A(-1) = A_{\text{even}}(1) - A_{\text{odd}}(1)$
  - $A(-i) = A_{\text{even}}(-1) - iA_{\text{odd}}(-1)$
- In other words, the values of  $A(x)$  at **4 points**  $1, i, -1, -i$  can be calculated based on the values of  $A_{\text{even}}(x), A_{\text{odd}}(x)$  at **2 points**  $1, -1$ .

# An example: $n = 4$

$$\begin{aligned}y_0 &= a_0 & + a_1 & & + a_2 & & + a_3 \\y_1 &= a_0 & + a_1\omega^1 & + a_2\omega^2 & + a_3\omega^3 \\y_2 &= a_0 & + a_1\omega^2 & + a_2\omega^4 & + a_3\omega^6 \\y_3 &= a_0 & + a_1\omega^3 & + a_2\omega^6 & + a_3\omega^9\end{aligned}$$

- Objective: Evaluate  $A(x)$  at 4 points:  $1, \omega, \omega^2, \omega^3$ , where  $\omega = e^{\frac{1}{4}2\pi i}$ .



# Step 1: Simplification

$$\begin{aligned}y_0 &= a_0 + a_1 + a_2 + a_3 \\y_1 &= a_0 + a_1\omega^1 + a_2\omega^2 + a_3\omega^3 \\y_2 &= a_0 + a_1\omega^2 + a_2 + a_3\omega^2 \\y_3 &= a_0 + a_1\omega^3 + a_2\omega^2 + a_3\omega^1\end{aligned}$$

## Step 2. Divide into odd- and even-terms

$$\begin{aligned}y_0 &= a_0 & + a_2 & & + a_1 & & + a_3 \\y_1 &= a_0 & + a_2\omega^2 & & + a_1\omega^1 & & + a_3\omega^3 \\y_2 &= a_0 & + a_2 & & + a_1\omega^2 & & + a_3\omega^2 \\y_3 &= a_0 & + a_2\omega^2 & & + a_1\omega^3 & & + a_3\omega^1\end{aligned}$$



# Key observation: redundant calculations

$$\begin{array}{rcl} y_0 & = & a_0 + a_2 + a_1 + a_3 \\ y_1 & = & a_0 + a_2\omega^2 + a_1\omega^1 + a_3\omega^3 \\ y_2 & = & a_0 + a_2 + a_1\omega^2 + a_3\omega^2 \\ y_3 & = & a_0 + a_2\omega^2 + a_1\omega^3 + a_3\omega^1 \end{array}$$

Note that the calculations in the two red frames are identical, and the calculations in the blue frames are also identical after multiplying by  $\omega^2$ . Here  $\omega^2 = -1$  as  $\omega = e^{\frac{2\pi}{4}i}$ .

## Step 3: Divide-and-conquer

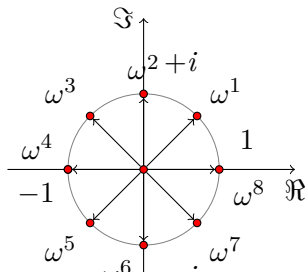
$$\begin{aligned}y_0 &= a_0 + a_2 + a_1 + a_3 \\y_1 &= a_0 + a_2\omega^2 + a_1\omega^1 + a_3\omega^3 \\y_2 &= a_0 + a_2 + a_1\omega^2 + a_3\omega^2 \\y_3 &= a_0 + a_2\omega^2 + a_1\omega^3 + a_3\omega^1\end{aligned}$$

Thus the calculations in the top-left and bottom-right frames are redundant. We need only  $2 + 4 + 2 = 4 \times \log_2 4$  calculations.

## Another example: $n = 8$

$$\begin{array}{l} y_0 = a_0 + a_1 + a_2 + a_3 + a_4 + a_5 + a_6 + a_7 \\ y_1 = a_0 + a_1\omega^1 + a_2\omega^2 + a_3\omega^3 + a_4\omega^4 + a_5\omega^5 + a_6\omega^6 + a_7\omega^7 \\ y_2 = a_0 + a_1\omega^2 + a_2\omega^4 + a_3\omega^6 + a_4\omega^8 + a_5\omega^{10} + a_6\omega^{12} + a_7\omega^{14} \\ y_3 = a_0 + a_1\omega^3 + a_2\omega^6 + a_3\omega^9 + a_4\omega^{12} + a_5\omega^{15} + a_6\omega^{18} + a_7\omega^{21} \\ y_4 = a_0 + a_1\omega^4 + a_2\omega^8 + a_3\omega^{12} + a_4\omega^{16} + a_5\omega^{20} + a_6\omega^{24} + a_7\omega^{28} \\ y_5 = a_0 + a_1\omega^5 + a_2\omega^{10} + a_3\omega^{15} + a_4\omega^{20} + a_5\omega^{25} + a_6\omega^{30} + a_7\omega^{35} \\ y_6 = a_0 + a_1\omega^6 + a_2\omega^{12} + a_3\omega^{18} + a_4\omega^{24} + a_5\omega^{30} + a_6\omega^{36} + a_7\omega^{42} \\ y_7 = a_0 + a_1\omega^7 + a_2\omega^{14} + a_3\omega^{21} + a_4\omega^{28} + a_5\omega^{35} + a_6\omega^{42} + a_7\omega^{49} \end{array}$$

- Objective: Evaluate  $A(x)$  at 8 points:  $1, \omega, \omega^2, \dots, \omega^7$ , where  $\omega = e^{\frac{1}{8}2\pi i}$ .



# Step 1: Simplification

$$\begin{array}{lclclclclcl} y_0 & = & a_0 & + & a_1 & + & a_2 & + & a_3 & + & a_4 & + & a_5 & + & a_6 & + & a_7 \\ y_1 & = & a_0 & + & a_1\omega^1 & + & a_2\omega^2 & + & a_3\omega^3 & + & a_4\omega^4 & + & a_5\omega^5 & + & a_6\omega^6 & + & a_7\omega^7 \\ y_2 & = & a_0 & + & a_1\omega^2 & + & a_2\omega^4 & + & a_3\omega^6 & + & a_4 & + & a_5\omega^2 & + & a_6\omega^4 & + & a_7\omega^6 \\ y_3 & = & a_0 & + & a_1\omega^3 & + & a_2\omega^6 & + & a_3\omega^1 & + & a_4\omega^4 & + & a_5\omega^7 & + & a_6\omega^2 & + & a_7\omega^5 \\ y_4 & = & a_0 & + & a_1\omega^4 & + & a_2\omega^8 & + & a_3\omega^4 & + & a_4 & + & a_5\omega^4 & + & a_6 & + & a_7\omega^4 \\ y_5 & = & a_0 & + & a_1\omega^5 & + & a_2\omega^2 & + & a_3\omega^7 & + & a_4\omega^4 & + & a_5\omega^1 & + & a_6\omega^6 & + & a_7\omega^3 \\ y_6 & = & a_0 & + & a_1\omega^6 & + & a_2\omega^4 & + & a_3\omega^2 & + & a_4 & + & a_5\omega^6 & + & a_6\omega^4 & + & a_7\omega^2 \\ y_7 & = & a_0 & + & a_1\omega^7 & + & a_2\omega^6 & + & a_3\omega^5 & + & a_4\omega^4 & + & a_5\omega^3 & + & a_6\omega^2 & + & a_7\omega^1 \end{array}$$

## Step 2. Divide into odd- and even-terms

$$\begin{aligned}y_0 &= a_0 & + a_4 & + a_2 & + a_6 & + a_1 & + a_5 & + a_3 & + a_7 \\y_1 &= a_0 & + a_4\omega^4 & + a_2\omega^2 & + a_6\omega^6 & + a_1\omega^1 & + a_5\omega^5 & + a_3\omega^3 & + a_7\omega^7 \\y_2 &= a_0 & + a_4 & + a_2\omega^4 & + a_6\omega^4 & + a_1\omega^2 & + a_5\omega^2 & + a_3\omega^6 & + a_7\omega^6 \\y_3 &= a_0 & + a_4\omega^4 & + a_2\omega^6 & + a_6\omega^2 & + a_1\omega^3 & + a_5\omega^7 & + a_3\omega^1 & + a_7\omega^5 \\y_4 &= a_0 & + a_4 & + a_2 & + a_6 & + a_1\omega^4 & + a_5\omega^4 & + a_3\omega^4 & + a_7\omega^4 \\y_5 &= a_0 & + a_4\omega^4 & + a_2\omega^2 & + a_6\omega^6 & + a_1\omega^5 & + a_5\omega^1 & + a_3\omega^7 & + a_7\omega^3 \\y_6 &= a_0 & + a_4 & + a_2\omega^4 & + a_6\omega^4 & + a_1\omega^6 & + a_5\omega^6 & + a_3\omega^2 & + a_7\omega^2 \\y_7 &= a_0 & + a_4\omega^4 & + a_2\omega^6 & + a_6\omega^2 & + a_1\omega^7 & + a_5\omega^3 & + a_3\omega^5 & + a_7\omega^1\end{aligned}$$

The specific order of these terms will be explained later.

# Key observation: redundant calculations

$y_0 =$	$a_0$	$+$	$a_4$	$+$	$a_2$	$+$	$a_6$	$+$	$a_1$	$+$	$a_5$	$+$	$a_3$	$+$	$a_7$
$y_1 =$	$a_0$	$+$	$a_4\omega^4$	$+$	$a_2\omega^2$	$+$	$a_6\omega^6$	$+$	$a_1\omega^1$	$+$	$a_5\omega^5$	$+$	$a_3\omega^3$	$+$	$a_7\omega^7$
$y_2 =$	$a_0$	$+$	$a_4$	$+$	$a_2\omega^4$	$+$	$a_6\omega^4$	$+$	$a_1\omega^2$	$+$	$a_5\omega^2$	$+$	$a_3\omega^6$	$+$	$a_7\omega^6$
$y_3 =$	$a_0$	$+$	$a_4\omega^4$	$+$	$a_2\omega^6$	$+$	$a_6\omega^2$	$+$	$a_1\omega^3$	$+$	$a_5\omega^7$	$+$	$a_3\omega^1$	$+$	$a_7\omega^5$
$y_4 =$	$a_0$	$+$	$a_4$	$+$	$a_2$	$+$	$a_6$	$+$	$a_1\omega^4$	$+$	$a_5\omega^4$	$+$	$a_3\omega^4$	$+$	$a_7\omega^4$
$y_5 =$	$a_0$	$+$	$a_4\omega^4$	$+$	$a_2\omega^2$	$+$	$a_6\omega^6$	$+$	$a_1\omega^5$	$+$	$a_5\omega^1$	$+$	$a_3\omega^7$	$+$	$a_7\omega^3$
$y_6 =$	$a_0$	$+$	$a_4$	$+$	$a_2\omega^4$	$+$	$a_6\omega^4$	$+$	$a_1\omega^6$	$+$	$a_5\omega^6$	$+$	$a_3\omega^2$	$+$	$a_7\omega^2$
$y_7 =$	$a_0$	$+$	$a_4\omega^4$	$+$	$a_2\omega^6$	$+$	$a_6\omega^2$	$+$	$a_1\omega^7$	$+$	$a_5\omega^3$	$+$	$a_3\omega^5$	$+$	$a_7\omega^1$

Note that the calculations in the two red frames are identical, and the calculations in the blue frames are also identical after multiplying by  $\omega^4$ . Here  $\omega^4 = -1$  as  $\omega = e^{\frac{2\pi}{8}i}$ .

## Step 3: Divide-and-conquer

$$\begin{array}{rclclclclcl} y_0 = & a_0 & + & a_4 & + & a_2 & + & a_6 & + & a_1 & + & a_5 & + & a_3 & + & a_7 \\ y_1 = & a_0 & + & a_4\omega^4 & + & a_2\omega^2 & + & a_6\omega^6 & + & a_1\omega^1 & + & a_5\omega^5 & + & a_3\omega^3 & + & a_7\omega^7 \\ y_2 = & a_0 & + & a_4 & + & a_2\omega^4 & + & a_6\omega^4 & + & a_1\omega^2 & + & a_5\omega^2 & + & a_3\omega^6 & + & a_7\omega^6 \\ y_3 = & a_0 & + & a_4\omega^4 & + & a_2\omega^6 & + & a_6\omega^2 & + & a_1\omega^3 & + & a_5\omega^7 & + & a_3\omega^1 & + & a_7\omega^5 \\ y_4 = & a_0 & + & a_4 & + & a_2 & + & a_6 & + & a_1\omega^4 & + & a_5\omega^4 & + & a_3\omega^4 & + & a_7\omega^4 \\ y_5 = & a_0 & + & a_4\omega^4 & + & a_2\omega^2 & + & a_6\omega^6 & + & a_1\omega^5 & + & a_5\omega^1 & + & a_3\omega^7 & + & a_7\omega^3 \\ y_6 = & a_0 & + & a_4 & + & a_2\omega^4 & + & a_6\omega^4 & + & a_1\omega^6 & + & a_5\omega^6 & + & a_3\omega^2 & + & a_7\omega^2 \\ y_7 = & a_0 & + & a_4\omega^4 & + & a_2\omega^6 & + & a_6\omega^2 & + & a_1\omega^7 & + & a_5\omega^3 & + & a_3\omega^5 & + & a_7\omega^1 \end{array}$$

Thus the calculations in the top-left and bottom-right frames are redundant.

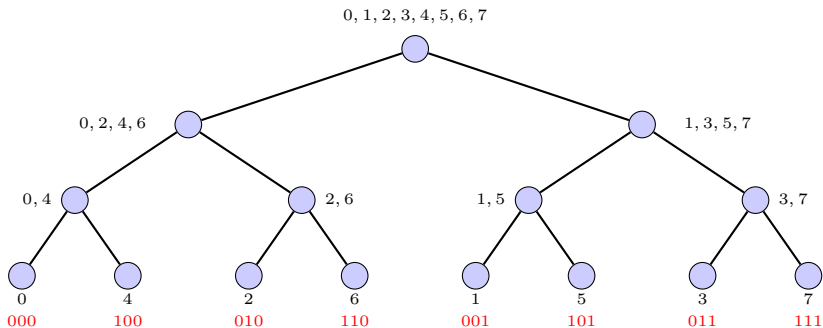
## Step 3: Divide-and-conquer

$$\begin{aligned}y_0 &= a_0 + a_4 + a_2 + a_6 + a_1 + a_5 + a_3 + a_7 \\y_1 &= a_0 + a_4\omega^4 + a_2\omega^2 + a_6\omega^6 + a_1\omega^1 + a_5\omega^5 + a_3\omega^3 + a_7\omega^7 \\y_2 &= a_0 + a_4 + a_2\omega^4 + a_6\omega^4 + a_1\omega^2 + a_5\omega^2 + a_3\omega^6 + a_7\omega^6 \\y_3 &= a_0 + a_4\omega^4 + a_2\omega^6 + a_6\omega^2 + a_1\omega^3 + a_5\omega^7 + a_3\omega^1 + a_7\omega^5 \\y_4 &= a_0 + a_4 + a_2 + a_6 + a_1\omega^4 + a_5\omega^4 + a_3\omega^4 + a_7\omega^4 \\y_5 &= a_0 + a_4\omega^4 + a_2\omega^2 + a_6\omega^6 + a_1\omega^5 + a_5\omega^1 + a_3\omega^7 + a_7\omega^3 \\y_6 &= a_0 + a_4 + a_2\omega^4 + a_6\omega^4 + a_1\omega^6 + a_5\omega^6 + a_3\omega^2 + a_7\omega^2 \\y_7 &= a_0 + a_4\omega^4 + a_2\omega^6 + a_6\omega^2 + a_1\omega^7 + a_5\omega^3 + a_3\omega^5 + a_7\omega^1\end{aligned}$$

Finally, we need only  $2 + 4 + 2 + 8 + 2 + 4 + 2 = 8 \times \log_2 8$  calculations.



# The final order



# FFT Algorithm

FFT( $n, a_0, a_1, \dots, a_{n-1}$ )

```
1: if  $n == 1$  then  
2:   return  $a_0$  ;  
3: end if  
4:  $(E_0, E_1, \dots, E_{\frac{n}{2}-1}) = \text{FFT}(\frac{n}{2}, a_0, a_2, \dots, a_{n-2})$ ;  
5:  $(O_0, O_1, \dots, O_{\frac{n}{2}-1}) = \text{FFT}(\frac{n}{2}, a_1, a_3, \dots, a_{n-1})$ ;  
6: for  $k = 0$  to  $\frac{n}{2} - 1$  do  
7:    $\omega^k = e^{\frac{2\pi}{n}ki}$ ;  
8:    $y_k = E_k + \omega^k O_k$ ;  
9:    $y_{\frac{n}{2}+k} = E_k - \omega^k O_k$ ;  
10: end for  
11: return  $(y_0, y_1, \dots, y_{n-1})$ ;
```

- Here  $\text{FFT}(\frac{n}{2}, a_0, a_2, \dots, a_n)$  computes the polynomial

$A_{\text{even}}(x) = a_0 + a_2x + \dots + a_nx^{\frac{n}{2}}$  at  $\frac{n}{2}$  points

$1, \omega^2, \omega^4, \dots, \omega^{n-2}$ , and  $\text{FFT}(\frac{n}{2}, a_1, a_3, \dots, a_{n-1})$  computes the polynomial  $A_{\text{odd}}(x) = a_1 + a_3x + \dots + a_{n-1}x^{\frac{n}{2}}$  at these points.

# Inverse Discrete Fourier Transform

- Inverse Discrete Fourier Transform: to determine coefficients of a polynomial  $a_0, a_1, \dots, a_{n-1}$  based on  $n$  point-value pairs  $(1, y_0), (\omega, y_1), \dots, (\omega^{n-1}, y_{n-1})$ , where  $y_k = A(\omega^k)$ , and  $A(x) = a_0 + a_1x + a_2x^2 + \dots + a_{n-1}x^{n-1}$ .
- Matrix form

$$\begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_{n-1} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & \omega^1 & \omega^2 & \dots & \omega^{n-1} \\ 1 & \omega^2 & \omega^4 & \dots & \omega^{2(n-1)} \\ \dots & \dots & \dots & \dots & \dots \\ 1 & \omega^{n-1} & \omega^{2(n-1)} & \dots & \omega^{(n-1)^2} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_{n-1} \end{bmatrix}$$

- It takes  $O(n^3)$  to calculate the inverse matrix when using the Gaussian elimination technique.

# Inverse Discrete Fourier Transform cont'd

- Matrix form

$$\begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_{n-1} \end{bmatrix} = \frac{1}{n} \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & \bar{\omega}^1 & \bar{\omega}^2 & \dots & \bar{\omega}^{n-1} \\ 1 & \bar{\omega}^2 & \bar{\omega}^4 & \dots & \bar{\omega}^{2(n-1)} \\ \dots & \dots & \dots & \dots & \dots \\ 1 & \bar{\omega}^{n-1} & \bar{\omega}^{2(n-1)} & \dots & \bar{\omega}^{(n-1)^2} \end{bmatrix} \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_{n-1} \end{bmatrix}$$

- Reason: it turns out that it is nearly its own inverse. More precisely, the conjugate transpose of this matrix is its own inverse.

# IFFT Algorithm

IFFT( $n, y_0, y_1, \dots, y_{n-1}$ )

```
1: if  $n == 1$  then  
2:   return  $y_0$  ;  
3: end if  
4:  $(E_0, E_1, \dots, E_{\frac{n}{2}-1}) = \text{IFFT}(\frac{n}{2}, y_0, y_2, \dots, y_{n-2})$ ;  
5:  $(O_0, O_1, \dots, O_{\frac{n}{2}-1}) = \text{IFFT}(\frac{n}{2}, y_1, y_3, \dots, y_{n-1})$ ;  
6: for  $k = 0$  to  $\frac{n}{2} - 1$  do  
7:    $\omega^k = e^{-\frac{2\pi}{n}ki}$ ;  
8:    $a_k = E_k + \omega^k O_k$ ;  
9:    $a_{\frac{n}{2}+k} = E_k - \omega^k O_k$ ;  
10: end for  
11: return  $\frac{1}{n}(a_0, a_1, \dots, a_{n-1})$  ;
```

Here we assume  $n$  is the power of 2 for simplicity. The normalization factors multiplying FFT and IFFT (here 1 and  $\frac{1}{n}$ ) and the signs of exponents are merely conventions, and differ in some treatments.

Application: fast multiplication of two polynomials (or two integers)

# Multiply two polynomials: convolution

- Given two polynomials

$$A(x) = a_0 + a_1x + a_2x^2 + \dots + a_{n-1}x^{n-1}, \text{ and}$$

$$B(x) = b_0 + b_1x + b_2x^2 + \dots + b_{n-1}x^{n-1}$$

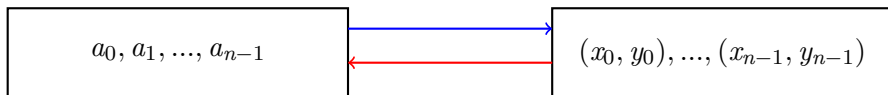
- Let's calculate its product

$$C(x) = A(x)B(x) = c_0 + c_1x + c_2x^2 + \dots + c_{2n-2}x^{2n-2}$$

- Brute-force (convolution):  $c_k = \sum_{i=0}^k a_i b_{k-i}$ .
- It costs  $O(n^2)$  time if using the convolution technique.

# Conversion between two representations of polynomials

- An efficient conversion between these two representations is extremely useful when multiplying two polynomials.





# Using FFT to speed up multiplication

- Given two polynomials

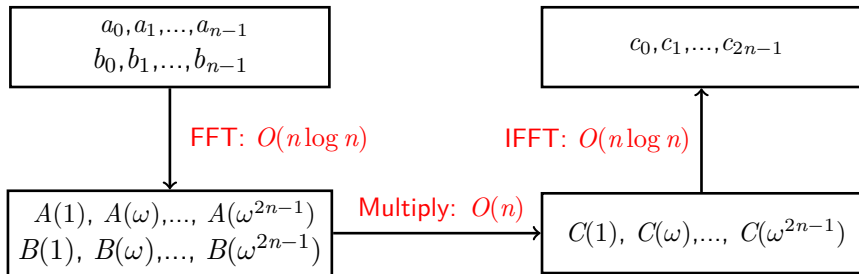
$$A(x) = a_0 + a_1x + a_2x^2 + \dots + a_{n-1}x^{n-1}, \text{ and}$$

$$B(x) = b_0 + b_1x + b_2x^2 + \dots + b_{n-1}x^{n-1}$$

- Let's calculate its product

$$C(x) = A(x)B(x) = c_0 + c_1x + c_2x^2 + \dots + c_{2n-2}x^{2n-2}$$

- Brute-force:  $c_k = \sum_{i=0}^k a_i b_{k-i}$ . Cost  $O(n^2)$  time
- Using FFT and IFFT:  $O(n \log n)$



# An example

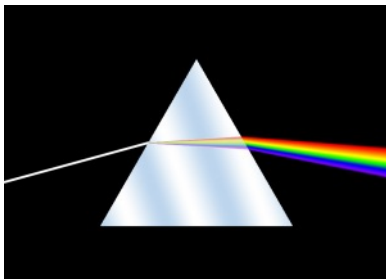
- $A(x) = 1 + 2x$
- $B(x) = 3 + 4x$
- $C(x) = A(x)B(x) = c_0 + c_1x + c_2x^2 + c_3x^3$

$x$	1	$-i$	$-1$	$i$
$A(x)$	3	$1 - 2i$	$-1$	$1 + 2i$
$B(x)$	7	$3 - 4i$	$-1$	$3 + 4i$
$C(x)$	21	$-5 - 10i$	1	$-5 + 10i$

- By running  $\text{IFFT}(4, (21, -5 - 10i, 1, -5 + 10i))$ , we obtained the coefficients as  $c_0 = 3$ ,  $c_1 = 10$ ,  $c_2 = 8$ , and  $c_3 = 0$ .
- Extension: given two  $n$ -bit integers  $a = a_{n-1} \dots a_1 a_0$ , and  $b = b_{n-1} \dots b_1 b_0$ , it takes  $O(n \log n)$  complex arithmetic steps to calculate  $c = a \times b$ .
- In 1971, A. Schönhage and V. Strassen proposed an algorithm for multiplication that uses  $O(n \log n \log \log n)$  bit operations.

Application: time-frequency transform

- One analogy for the type of thing a Fourier Transform does is a prism which splits white light into a spectrum of colors.



- Extension: Integer factorization using quantum computing

# DFT: time-domain vs. frequency-domain

- DFT, denoted as  $\mathbf{X} = \mathcal{F}\{\mathbf{x}\}$ , transforms a sequence of  $N$  complex numbers  $x_0, x_1, \dots, x_{N-1}$  (time-domain) into a  $N$ -periodic sequence of complex numbers  $X_0, X_1, \dots, X_{N-1}$  (frequency-domain):

$$X_k = \sum_{n=0}^{N-1} x_n e^{-\frac{2\pi}{N} i k n}, \quad k = 0, 1, \dots, N-1$$

- Here,  $X_k$  encodes both amplitude and phase of a sinusoidal component  $e^{-\frac{2\pi}{N} k n i}$  of the function  $x_n$  (the sinusoid's frequency is  **$k$  cycles per  $N$  samples**).
- **Inverse transform** of DFT:

$$x_n = \frac{1}{N} \sum_{k=0}^{N-1} X_k e^{\frac{2\pi}{N} i k n}$$

An interpretation of DFT is that its inverse transform is the **discrete analogy** of the formula for **a Fourier series**:

$$f(x) = \sum_{n=-\infty}^{+\infty} F_n e^{i n x}, \quad F_n = \frac{1}{2\pi} \int_{-\infty}^{\infty} f(x) e^{-i n x} dx$$

# Time-frequency transformation

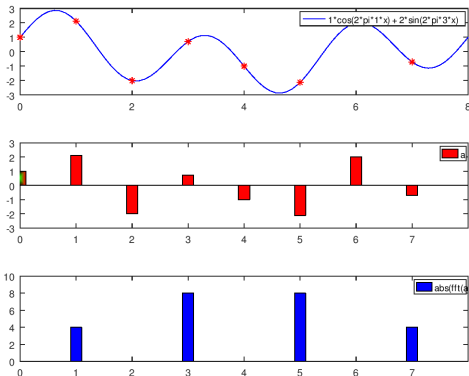
- FFT transforms the input data  $a_0, a_1, \dots, a_{n-1}$  (time-domain samples) into  $y_0, y_1, \dots, y_{n-1}$  (frequency domain). For example,

$$\begin{array}{rclclclclcl} y_0 & = & a_0 & + & a_1 & + & a_2 & + & a_3 & + & a_4 & + & a_5 & + & a_6 & + & a_7 \\ y_1 & = & a_0 & + & a_1\omega^1 & + & a_2\omega^2 & + & a_3\omega^3 & + & a_4\omega^4 & + & a_5\omega^5 & + & a_6\omega^6 & + & a_7\omega^7 \\ y_2 & = & a_0 & + & a_1\omega^2 & + & a_2\omega^4 & + & a_3\omega^6 & + & a_4\omega^8 & + & a_5\omega^{10} & + & a_6\omega^{12} & + & a_7\omega^{14} \\ y_3 & = & a_0 & + & a_1\omega^3 & + & a_2\omega^6 & + & a_3\omega^9 & + & a_4\omega^{12} & + & a_5\omega^{15} & + & a_6\omega^{18} & + & a_7\omega^{21} \\ y_4 & = & a_0 & + & a_1\omega^4 & + & a_2\omega^8 & + & a_3\omega^{12} & + & a_4\omega^{16} & + & a_5\omega^{20} & + & a_6\omega^{24} & + & a_7\omega^{28} \\ y_5 & = & a_0 & + & a_1\omega^5 & + & a_2\omega^{10} & + & a_3\omega^{15} & + & a_4\omega^{20} & + & a_5\omega^{25} & + & a_6\omega^{30} & + & a_7\omega^{35} \\ y_6 & = & a_0 & + & a_1\omega^6 & + & a_2\omega^{12} & + & a_3\omega^{18} & + & a_4\omega^{24} & + & a_5\omega^{30} & + & a_6\omega^{36} & + & a_7\omega^{42} \\ y_7 & = & a_0 & + & a_1\omega^7 & + & a_2\omega^{14} & + & a_3\omega^{21} & + & a_4\omega^{28} & + & a_5\omega^{35} & + & a_6\omega^{42} & + & a_7\omega^{49} \end{array}$$

- Here  $y_k$  encodes both amplitude and phase of a sinusoidal component of the time-domain samples  $a_0, a_1, \dots, a_7$ .

# FFT: an example

```
N = 8;  
t = 0:1/N:1-1/N;  
a = 1*cos(2*pi*1*t) + 2*sin(2*pi*3*t);  
Freq = 0:N-1;  
bar( Freq, abs(fft(a)), "b", 0.2 );
```



## $y_k$ encodes amplitude and phase of a sinusoidal component

- $y_k = a_0 + a_1\omega^k + a_2\omega^{2k} + \dots + a_7\omega^{7k}$  computes the dot product of two vectors: the time-domain samples  $a_0, a_1, \dots, a_7$ , and a sinusoid signal  $1, \omega^k, \omega^{2k}, \dots, \omega^{7k}$ . Here  $\omega = e^{\frac{2\pi}{8}i}$  and thus the sinusoid has frequency of  $k$  cycles per 8 samples.
- The dot product  $y_k = 0$  if the time-domain samples  $a_0, a_1, \dots, a_7$  do not consist of any sinusoidal component of such frequency. The reason is that:
  - The dot product of these two vectors  $y_k = \sum_{j=0}^7 a_j \omega^{jk}$  is essentially a discrete analogy of the integral of two sinusoids, say  $\int_0^{2\pi} \cos mx \cdot \cos nx dx$ .
  - The orthogonality of sinusoids states that for two integers  $m, n$ ,

$$\int_0^{2\pi} \cos mx \cdot \sin nx dx = 0, \text{ and}$$

$$\int_0^{2\pi} \sin mx \cdot \sin nx dx = 0, \int_0^{2\pi} \cos mx \cdot \cos nx dx = 0 \quad (m \neq n)$$



# Calculation of $y_1$

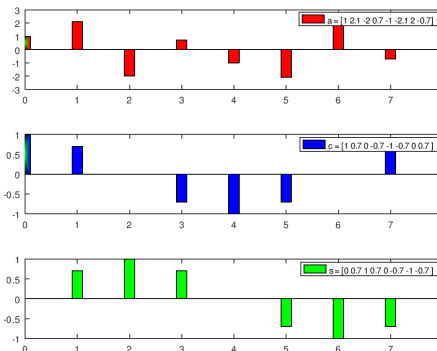
- $1, \omega^1, \omega^2, \dots, \omega^7$  represents a sinusoidal signal of frequency 1 cycle per 8 samples, and the existence of such sinusoidal component in the time-domain samples  $a_0, a_1, \dots, a_7$  is encoded by the dot product  $y_1 = a_0 + a_1\omega^1 + a_2\omega^2 + \dots + a_7\omega^7$ .

- $\mathbf{a} = [1 \ 2.1 \ -2 \ 0.7 \ -1 \ -2.1 \ 2 \ -0.7]$

$$\mathbf{c} = [1 \ 0.7 \ 0 \ -0.7 \ -1 \ -0.7 \ 0 \ 0.7]$$

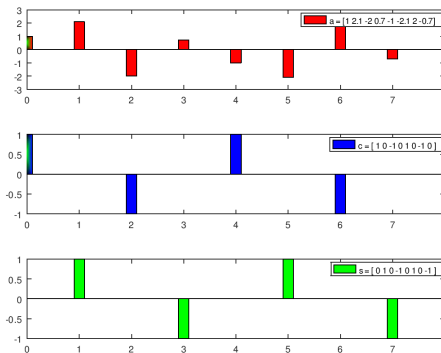
$$\mathbf{s} = [0 \ 0.7 \ 1 \ 0.7 \ 0 \ -0.7 \ -1 \ -0.7]$$

$$\text{sqrt}((\mathbf{a} \cdot \mathbf{c})^2 + (\mathbf{a} \cdot \mathbf{s})^2) = 4$$



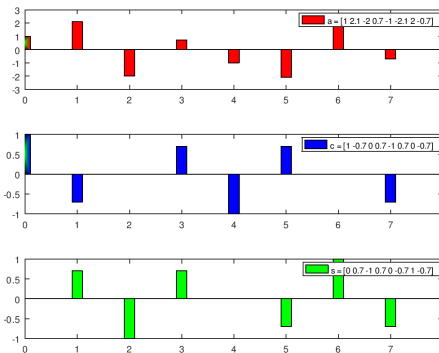
# Calculation of $y_2$

- $1, \omega^2, \omega^4, \dots, \omega^{14}$  represents a sinusoidal signal of frequency 2 cycles per 8 samples, and the existence of such sinusoidal component in time-domain samples  $a_0, a_1, \dots, a_7$  is encoded by the dot product  $y_2 = a_0 + a_1\omega^2 + a_2\omega^4 + \dots + a_7\omega^{14}$ .
- $a = [1 \ 2.1 \ -2 \ 0.7 \ -1 \ -2.1 \ 2 \ -0.7]$   
 $c = [1 \ 0 \ -1 \ 0 \ 1 \ 0 \ -1 \ 0]$   
 $s = [0 \ 1 \ 0 \ -1 \ 0 \ 1 \ 0 \ -1]$   
 $\text{sqrt}((a*c')^2 + (a*s')^2) = 0$



# Calculation of $y_3$

- $1, \omega^3, \omega^6, \dots, \omega^{21}$  represents a sinusoidal signal of frequency 3 cycles per 8 samples, and the existence of such sinusoidal component in time-domain samples  $a_0, a_1, \dots, a_7$  is encoded by the dot product  $y_3 = a_0 + a_1\omega^3 + a_2\omega^6 + \dots + a_7\omega^{21}$ .
- $a = [1 \ 2.1 \ -2 \ 0.7 \ -1 \ -2.1 \ 2 \ -0.7]$   
 $c = [1 \ -0.7 \ 0 \ 0.7 \ -1 \ 0.7 \ 0 \ -0.7]$   
 $s = [0 \ 0.7 \ -1 \ 0.7 \ 0 \ -0.7 \ 1 \ -0.7]$   
 $\text{sqrt}((a \cdot c')^2 + (a \cdot s')^2) = 8$



## Appendix: Relationship between continuous and discrete Fourier transforms

# Fourier series, Fourier transform, DTFT, and DFT

- Fourier series decomposes a periodic function into a set of sine/cosine waves, and one of the motivations of Fourier transform comes from the extension of Fourier series to non-periodic functions.
- DTFT uses discrete-time samples of a continuous function as input, and generates a continuous function of frequency.
- Using a finite sequence of equally-spaced samples of a function as input, DFT computes a sequence of identical length, representing equally-spaced samples of DTFT. The interval at which the DTFT is sampled is reciprocal of the duration of the input sequence.
- The inverse DFT is a Fourier series using the DTFT samples as coefficients of corresponding frequency, and it is essentially a periodic summation of the original input sequence.



Figure: Jean-Baptiste Joseph Fourier (1768-1830)

- In 1807, Joseph Fourier proposed the idea of Fourier series when solving heat equation, a partial differential equation.
- Prior to Fourier's work, no solution to heat equation was known in the general case. However, when the heat source was a simple sine or cosine wave, solutions were known (called eigensolutions).
- Thus, Fourier modelled complicated heat source as a superposition of simple sine/cosine waves, and rewrote the solution as superposition of corresponding eigensolutions.

- Fourier series is a way to represent a **periodic function** of time as the sum of a set of simple sines and cosines (or, equivalently, complex exponentials). For example, the Fourier series of a periodic function  $f(x)$  (period  $2\pi$ ) is:

$$f(x) = a_0 + \sum_{n=1}^{\infty} (a_n \cos nx + b_n \sin nx)$$

where

$$a_0 = \frac{1}{2\pi} \int_0^{2\pi} f(t) dt$$

$$a_n = \frac{1}{\pi} \int_0^{2\pi} f(t) \cos ntdt \quad (n = 1, 2, \dots)$$

$$b_n = \frac{1}{\pi} \int_0^{2\pi} f(t) \sin ntdt \quad (n = 1, 2, \dots)$$

# Fourier series: orthogonality of basis functions

- Unlike Taylor's expansion, the basis functions of Fourier series are orthogonal over  $[0, 2\pi]$ , i.e., for two integers  $m, n$ ,

$$\int_0^{2\pi} 1 \cdot \sin x dx = 0, \quad \int_0^{2\pi} 1 \cdot \cos x dx = 0$$

$$\int_0^{2\pi} \sin mx \cdot \sin nx dx = 0, \quad \int_0^{2\pi} \cos mx \cdot \cos nx dx = 0 \quad (m \neq n)$$

$$\int_0^{2\pi} \cos mx \cdot \sin nx dx = 0$$

- The orthogonality plays an important role in solving coefficients  $a_0, a_n, b_n$ .



# Fourier series: complex exponential form

- According to the Euler's formula  $e^{ix} = \cos x + i \sin x$ , we have  $\cos x = \frac{1}{2}(e^{ix} + e^{-ix})$ ,  $\sin x = \frac{1}{2i}(e^{ix} - e^{-ix})$ , and

$$\begin{aligned} f(x) &= a_0 + \sum_{n=1}^{\infty} (a_n \cos nx + b_n \sin nx) \\ &= a_0 + \sum_{n=1}^{\infty} \left( a_n \frac{1}{2} (e^{inx} + e^{-inx}) + b_n \frac{1}{2i} (e^{inx} - e^{-inx}) \right) \\ &= a_0 + \sum_{n=1}^{\infty} \left( \frac{1}{2} (a_n - ib_n) e^{inx} + \frac{1}{2} (a_n + ib_n) e^{-inx} \right) \end{aligned}$$

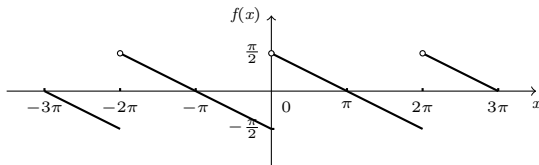
- Define  $F_0 = a_0$ , and  $F_n = \frac{1}{2}(a_n - ib_n)$  ( $n > 0$ ). We have  $F_{-n} = \frac{1}{2}(a_n + ib_n)$ , and thus rewrite the Fourier series as:

$$f(x) = \sum_{n=-\infty}^{+\infty} F_n e^{inx}, \quad F_n = \frac{1}{2}(a_n - ib_n) = \frac{1}{2\pi} \int_0^{2\pi} f(t) e^{-int} dt$$

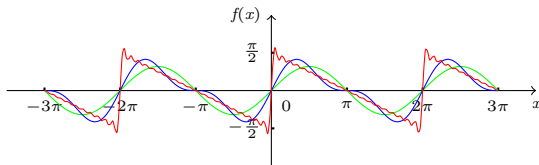
- Complex exponential form is necessary as the complex coefficients  $F_n$  (called *frequency spectrum*) could encode both amplitude and phase of basic waves.

# Fourier series: example 1

- Periodic function  $f(x) = \begin{cases} \frac{1}{2}(\pi - x) & 0 < x \leq 2\pi \\ f(x + 2\pi) & \text{otherwise} \end{cases}$



- Fourier series:  $f(x) = \sum_{n=1}^{\infty} \frac{1}{n} \sin nx$  (since  $a_n = 0$ ,  $b_n = \frac{1}{n}$ )



# Fourier series: extension to $f(x)$ with period of $2L$

- For a periodic function  $f(x)$  with period of  $2L$ , the Fourier series is:

$$f(x) = a_0 + \sum_{n=1}^{\infty} \left( a_n \cos \frac{\pi}{L} nx + b_n \sin \frac{\pi}{L} nx \right)$$

- The coefficients are:

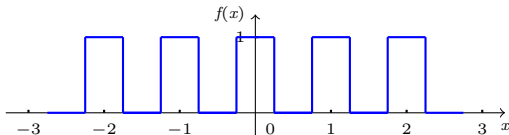
$$a_0 = \frac{1}{2L} \int_0^{2L} f(t) dt$$

$$a_n = \frac{1}{L} \int_0^{2L} f(t) \cos \frac{\pi}{L} n t dt$$

$$b_n = \frac{1}{L} \int_0^{2L} f(t) \sin \frac{\pi}{L} n t dt$$

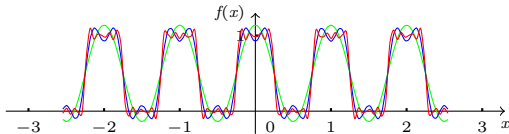
## Fourier series: example 2

- Periodic function  $f(x) = \begin{cases} 1 & |x| \leq \frac{1}{4} \\ 0 & \frac{1}{4} < |x| \leq \frac{T}{2} \end{cases}$ , and  $f(x)$  has a period  $T = 1$ .



- Fourier series:

$$f(x) = \frac{1}{2T} + \frac{2}{\pi} \sum_{n=1}^{\infty} \frac{1}{n} \sin\left(\frac{\pi}{2T}n\right) \cos\left(\frac{2\pi}{T}nx\right)$$



# Convergence of Fourier series: Dirichlet's conditions

- Dirichlet's theorem states the sufficient conditions for the convergence of Fourier series, i.e., if  $f(x)$  satisfies the following conditions:
  - 1  $f(x)$  is periodic, and absolutely integrable over a period;
  - 2  $f(x)$  must have a finite number of maxima and minima in any bounded interval;
  - 3  $f(x)$  must have a finite number of discontinuities in any bounded interval, and the discontinuity cannot be infinite.

Then

$$a_0 + \sum_{n=1}^m (a_n \cos nx + b_n \sin nx) \rightarrow \frac{1}{2}(f(x+0) + f(x-0))$$

when  $m \rightarrow \infty$ .

- A succinct proof using Dirac's  $\delta$  function can be found in *Mathematical Methods for Physics* (by Q. Gu).

## Proof.

- Since  $a_n \cos nx + b_n \sin nx = \frac{1}{\pi} \int_{-\pi}^{\pi} f(t) \cos n(x-t) dt$ , the partial sum of Fourier series is:

$$\begin{aligned} S_m(x) &= \frac{1}{2\pi} \int_{-\pi}^{\pi} f(t) \left[ 1 + 2 \sum_{n=1}^m \cos n(x-t) \right] dt \\ &= \int_{-\pi}^{\pi} f(t) \frac{\sin((m + \frac{1}{2})(x-t))}{2\pi \sin \frac{1}{2}(x-t)} dt \\ &= \int_{-\pi}^{\pi} f(t) D_m(x-t) dt \end{aligned}$$

- Here  $D_m(x) = \frac{1}{2\pi} (1 + 2 \cos x + 2 \cos 2x + \dots + 2 \cos mx)$ .
- Note that  $\lim_{m \rightarrow \infty} D_m(x) = \delta(x)$  since  $\int_{-\pi}^{\pi} D_m(x) dx = 1$  and  $D_m(0) = \frac{1}{2\pi} (2m+1) \rightarrow \infty$ .
- Thus, we have  $\lim_{m \rightarrow \infty} S_m(x) = \int_{-\pi}^{\pi} f(t) \delta(x-t) = f(x)$  (when  $f(x)$  is continuous at  $x$ ). Please refer to *Mathematical Methods for Physics* (by Q. Gu) for complete proof.

# Fourier transform (in terms of angular frequency $\omega$ )

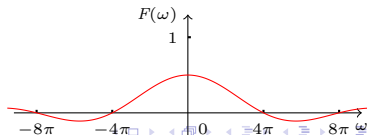
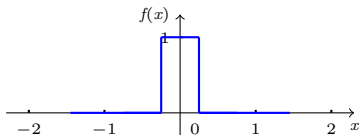
- Fourier transform of a function of time (a *signal*) is a complex-valued function of frequency (represented as *angular frequency*  $\omega$ ), whose absolute value represents the amount of that frequency present in the original function.

$$F(\omega) = \int_{-\infty}^{\infty} f(x) e^{-i\omega x} dx, \quad f(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega) e^{i\omega x} d\omega$$

- Fourier transform, denoted as  $F(\omega) = \mathcal{F}\{f(x)\}$ , is called *frequency representation of the original signal*, and  $F(\omega)$  is called *spectral density*.

- For example, the Fourier transform of  $f(x) = \begin{cases} 1 & |x| \leq \frac{1}{4} \\ 0 & \text{otherwise} \end{cases}$  is

$$F(\omega) = \int_{-\infty}^{\infty} f(x) e^{-i\omega x} dx = \frac{2}{\omega} \sin\left(\frac{\omega}{4}\right)$$



# Fourier transform (in terms of ordinary frequency $\nu$ )

- For a sinusoidal wave with period  $T$  (measured in *seconds*), its frequency can be measured using angular frequency  $\omega$  (measured in *radians per second*) or using ordinary frequency  $\nu$  (measured in *cycles per second*, or hertz), where  $\omega = 2\pi\nu$ , and  $\nu = \frac{1}{T}$ .
- When using angular frequency  $\omega$ , Fourier transform is defined as:

$$F(\omega) = \int_{-\infty}^{\infty} f(x) e^{-i\omega x} dx$$
$$f(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega) e^{i\omega x} d\omega$$

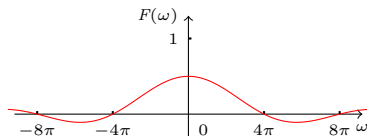
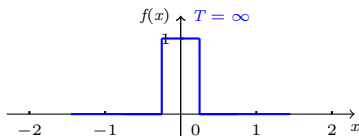
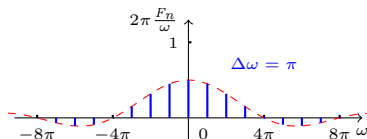
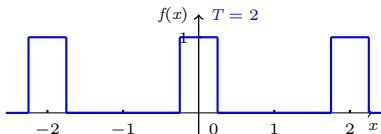
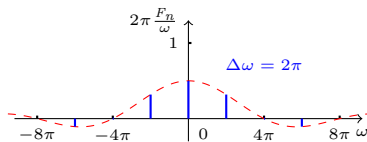
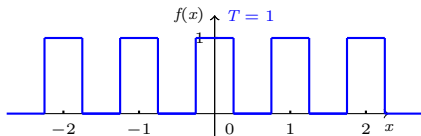
- Replacing  $\omega$  with  $\omega = 2\pi\nu$ , we obtain another representation of Fourier transform in terms of ordinary frequency  $\nu$ :

$$F(\nu) = \int_{-\infty}^{\infty} f(x) e^{-2\pi i \nu x} dx$$
$$f(x) = \int_{-\infty}^{\infty} F(\nu) e^{2\pi i \nu x} d\nu$$



# Connection between Fourier series and Fourier transform

- For a function that are zero outside an interval, we can calculate Fourier series on any larger interval. As we lengthen the interval, the coefficients of Fourier series will approach Fourier transform.



# Fourier transform: deduction

- Consider a periodic function  $f(x)$  with period  $2L$ . Its Fourier series  $f(x) = a_0 + \sum_{n=1}^{\infty} (a_n \cos \frac{\pi}{L} nx + b_n \sin \frac{\pi}{L} nx)$  can be rewritten as  $f(x) = a_0 + \sum_{n=1}^{\infty} (a_n \cos \omega_n x + b_n \sin \omega_n x)$ , where  $\omega_n = \frac{\pi}{L} n$  represents angular frequency.
- Intuitively, when  $L \rightarrow \infty$ ,  $f(x)$  becomes a non-periodic function over  $(-\infty, \infty)$ , and

$$\sum_{n=1}^{\infty} \dots \Delta\omega \rightarrow \int_0^{\infty} \dots d\omega.$$

- In particular, we have  $a_0 = \frac{1}{2L} \int_{-L}^L f(t) dt \xrightarrow{L \rightarrow \infty} 0$  since  $f(x)$  is absolutely integrable, and

$$\begin{aligned} \sum_{n=1}^{\infty} a_n \cos \omega_n x &= \sum_{n=1}^{\infty} \frac{1}{L} \left[ \int_{-L}^L f(t) \cos \omega_n t dt \right] \cos \omega_n x \\ &= \sum_{n=1}^{\infty} \frac{\Delta\omega}{\pi} \left[ \int_{-L}^L f(t) \cos \omega_n t dt \right] \cos \omega_n x \\ &\rightarrow \int_0^{\infty} d\omega \left[ \frac{1}{\pi} \int_{-\infty}^{\infty} f(t) \cos \omega t dt \right] \cos \omega x \end{aligned}$$

# Fourier transform: deduction cont'd

- Similarly, we have

$$\sum_{n=1}^{\infty} b_n \sin \omega_n x \rightarrow \int_0^{\infty} d\omega \left[ \frac{1}{\pi} \int_{-\infty}^{\infty} f(t) \sin \omega t dt \right] \sin \omega x$$

and rewrite Fourier series as:

$$\begin{aligned} f(x) &= \frac{1}{\pi} \int_{\omega=0}^{\infty} \int_{t=-\infty}^{\infty} f(t) (\cos \omega x \cos \omega t + \sin \omega x \sin \omega t) dt d\omega \\ &= \frac{1}{\pi} \int_{\omega=0}^{\infty} \int_{t=-\infty}^{\infty} f(t) \cos \omega(x-t) dt d\omega \\ &= \frac{1}{2\pi} \int_{\omega=0}^{\infty} \int_{t=-\infty}^{\infty} f(t) (e^{i\omega(x-t)} + e^{-i\omega(x-t)}) dt d\omega \\ &= \frac{1}{2\pi} \int_{-\infty}^{\infty} \left[ \int_0^{\infty} f(t) e^{i\omega(x-t)} d\omega + \int_0^{\infty} f(t) e^{-i\omega(x-t)} d\omega \right] dt \\ &= \frac{1}{2\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(t) e^{i\omega(x-t)} d\omega dt \\ &= \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega) dt \end{aligned}$$

# Fourier transform: properties

- Linear operations performed in one domain (time or frequency) have corresponding operations in the other domain.
- Differentiation in time domain corresponds to multiplication in the frequency domain, usually making it easier to analyze.
- Convolution in the time domain corresponds to the ordinary multiplication in the frequency domain.
- Functions that are localized in one domain have Fourier transforms that are spread out across the other domain, known as the *uncertainty principle*.
- The Fourier transform of a Gaussian function is another Gaussian function.

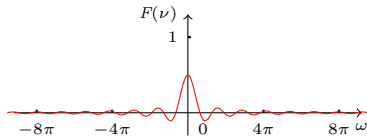
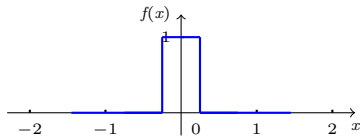
# Fourier transform: Poisson summation formula

- For a function  $f(x)$  with its Fourier transform (in terms of ordinary frequency)  $F(\nu) = \int_{-\infty}^{\infty} f(x) e^{-2\pi i x \nu} dx$ , the Poisson summation formula states  $\sum_{k=-\infty}^{\infty} f(k) = \sum_{k=-\infty}^{\infty} F(k)$ .
- For example, the Fourier transform of  $f(x) = \begin{cases} 1 & |x| \leq \frac{1}{4} \\ 0 & \text{otherwise} \end{cases}$  is

$$F(\nu) = \int_{-\infty}^{\infty} f(x) e^{-2\pi i \nu x} dx = \frac{1}{\pi \nu} \sin\left(\frac{\pi \nu}{2}\right)$$

- Poisson summation formula states that

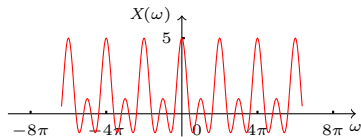
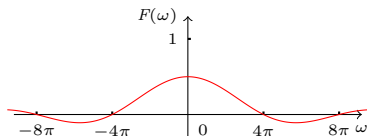
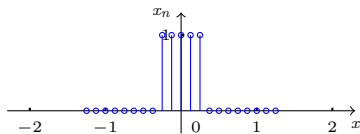
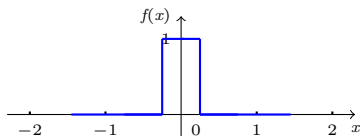
$$\sum_{k=-\infty}^{\infty} f(k) = 1 = \sum_{k=-\infty}^{\infty} F(k).$$



- Discrete-time Fourier transform (DTFT) refers to Fourier analysis on the uniformly-spaced samples of a continuous function, i.e., a Fourier series with  $x_n$  as coefficients:

$$X(\omega) = \sum_{n=-\infty}^{\infty} x_n e^{-in\omega}$$

Here, the frequency variable  $\omega$  has normalized units of *radians/sample*.



# Inverse transform of DTFT

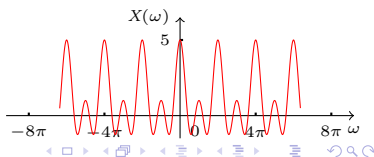
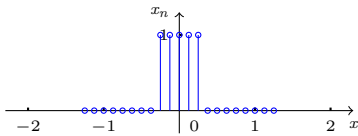
- DTFT is itself a periodic function of frequency  $X(\omega)$ . From this function, the original samples  $x_n$  can be readily recovered as below:

$$x_n = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\omega) e^{in\omega} d\omega$$

- For example, the DTFT is  $X(\omega) = 1 + 2 \cos \omega + 2 \cos 2\omega$ . The original samples can be recovered as:

$$x_0 = \frac{1}{2\pi} \int_{-\infty}^{\infty} (1 + 2 \cos \omega + 2 \cos 2\omega) d\omega = 1$$

Similarly, we obtained  $x_{-1} = x_1 = x_2 = x_{-2} = 1$ .



- From these samples, DTFT produces a function of frequency that is a periodic summation of the Fourier transform of the original continuous function.
- The sampling theorem states the theoretical conditions under which the original function can be perfectly recovered from DTFT of the samples.
- When the input data sequence  $x_n$  is  $N$ -periodic, DTFT reduces to DFT, i.e.,

$$X_k = \sum_{n=0}^{N-1} x_n e^{-\frac{2\pi}{N} kni}$$

- Alternatively, DTFT is itself a continuous function, and the discrete samples of it can be efficiently calculated using DFT.



# Appendix: Dirac's $\delta$ function

- Dirac's  $\delta$  function has the following two properties:

①  $\delta(x) = \begin{cases} \infty & x = 0 \\ 0 & \text{otherwise} \end{cases}$

②  $\int_{-\infty}^{+\infty} \delta(x) dx = 1$

- We can prove the following properties:

- For any continuous function  $f(x)$ ,

$$\int_{-\infty}^{+\infty} f(x) \delta(x - x_0) dx = f(x_0)$$

- $\delta(x)$  is the Fourier transform of 1 since

$$\mathcal{F}\{\delta(x)\} = \int_{-\infty}^{+\infty} \delta(x) e^{-i\omega x} dx = 1$$

- According to the inverse Fourier transform of 1, we have:

$$\delta(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{i\omega x} d\omega = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{-i\omega x} d\omega$$