MAT 271: Applied & Computational Harmonic Analysis
Lecture 7: *Discrete Fourier Transform* (DFT)

Naoki Saito

Department of Mathematics
University of California, Davis

January 30, 2014

Outline

Outline

Definitions

- The DFT can be viewed as either an *approximation to the Fourier transform* or an *approximation to the Fourier series coefficients*.
- Suppose $f \in L^2[-A/2, A/2]$, and $f(x) = 0$ for $|x| > A/2$. That is, f is a *space-limited*, square integrable function, which is a reasonable assumption in practice.
- Then, we can invoke the *dual* version of *the Sampling Theorem* in the *frequency* domain, which can also be stated in terms of Fourier coefficients of the Fourier series (Recall that *the Fourier transform of the periodic functions gives the line spectrum in the frequency domain*).
- In fact, we have the following relationship:

$$\hat{f}(k/A) = \int_{-A/2}^{A/2} f(x) \mathrm{e}^{-2\pi \mathrm{i} kx/A} \mathrm{d}x = \left\langle f, \mathrm{e}^{2\pi \mathrm{i} k \cdot /A} \right\rangle = \sqrt{A} \alpha_k, \quad k \in \mathbb{Z}. \quad (1)$$

Definitions

- The DFT can be viewed as either an *approximation to the Fourier transform* or an *approximation to the Fourier series coefficients*.
- Suppose $f \in L^2[-A/2, A/2]$, and $f(x) = 0$ for $|x| > A/2$. That is, f is a *space-limited*, square integrable function, which is a reasonable assumption in practice.
- Then, we can invoke the *dual* version of *the Sampling Theorem* in the *frequency* domain, which can also be stated in terms of Fourier coefficients of the Fourier series (Recall that *the Fourier transform of the periodic functions gives the line spectrum in the frequency domain*).
- In fact, we have the following relationship:

$$\hat{f}(k/A) = \int_{-A/2}^{A/2} f(x) \mathrm{e}^{-2\pi \mathrm{i} kx/A} \mathrm{d}x = \left\langle f, \mathrm{e}^{2\pi \mathrm{i} k \cdot /A} \right\rangle = \sqrt{A}\alpha_k, \quad k \in \mathbb{Z}. \quad (1)$$

Definitions

- The DFT can be viewed as either an *approximation to the Fourier transform* or an *approximation to the Fourier series coefficients*.
- Suppose $f \in L^2[-A/2, A/2]$, and $f(x) = 0$ for $|x| > A/2$. That is, f is a *space-limited*, square integrable function, which is a reasonable assumption in practice.
- Then, we can invoke the *dual* version of *the Sampling Theorem* in the *frequency* domain, which can also be stated in terms of Fourier coefficients of the Fourier series (Recall that *the Fourier transform of the periodic functions gives the line spectrum in the frequency domain*).
- In fact, we have the following relationship:

$$\hat{f}(k/A) = \int_{-A/2}^{A/2} f(x) \mathrm{e}^{-2\pi \mathrm{i} kx/A} \mathrm{d}x = \left\langle f, \mathrm{e}^{2\pi \mathrm{i} k \cdot /A} \right\rangle = \sqrt{A}\alpha_k, \quad k \in \mathbb{Z}. \quad (1)$$

Definitions

- The DFT can be viewed as either an *approximation to the Fourier transform* or an *approximation to the Fourier series coefficients*.
- Suppose $f \in L^2[-A/2, A/2]$, and $f(x) = 0$ for $|x| > A/2$. That is, f is a *space-limited*, square integrable function, which is a reasonable assumption in practice.
- Then, we can invoke the *dual* version of *the Sampling Theorem* in the *frequency* domain, which can also be stated in terms of Fourier coefficients of the Fourier series (Recall that *the Fourier transform of the periodic functions gives the line spectrum in the frequency domain*).
- In fact, we have the following relationship:

$$\hat{f}(k/A) = \int_{-A/2}^{A/2} f(x) e^{-2\pi i k x/A} \, dx = \left\langle f, e^{2\pi i k \cdot /A} \right\rangle = \sqrt{A}\alpha_k, \quad k \in \mathbb{Z}. \quad (1)$$

- In general, $f \in L^2[-A/2, A/2]$ is *not* a *band-limited* function; Recall the *uncertainty principles*!
- Therefore, to have a finite length vector representing the frequency samples (or the Fourier coefficients), we need to truncate the frequency information for $|\xi| > \Omega/2$ for some $\Omega > 0$.
- This is the *first* source of error of DFT approximation to FT/FS.
- This truncation allows us to consider only k with $|k| \le A\Omega/2$.

$$\hat{f}(k/A) = \int_{-A/2}^{A/2} f(x) e^{-2\pi i k x/A} \, dx = \left\langle f, e^{2\pi i k \cdot /A} \right\rangle = \sqrt{A}\alpha_k, \quad k \in \mathbb{Z}. \quad (1).$$

- In general, $f \in L^2[-A/2, A/2]$ is *not* a *band-limited* function; Recall the *uncertainty principles*!
- Therefore, to have a finite length vector representing the frequency samples (or the Fourier coefficients), we need to truncate the frequency information for $|\xi| > \Omega/2$ for some $\Omega > 0$.
- This is the *first* source of error of DFT approximation to FT/FS.
- This truncation allows us to consider only k with $|k| \le A\Omega/2$.

$$\hat{f}(k/A) = \int_{-A/2}^{A/2} f(x) \mathrm{e}^{-2\pi \mathrm{i} kx/A} \, \mathrm{d}x = \left\langle f, \mathrm{e}^{2\pi \mathrm{i} k \cdot /A} \right\rangle = \sqrt{A} \alpha_k, \quad k \in \mathbb{Z}. \quad (1).$$

- In general, $f \in L^2[-A/2, A/2]$ is *not* a *band-limited* function; Recall the *uncertainty principles*!
- Therefore, to have a finite length vector representing the frequency samples (or the Fourier coefficients), we need to truncate the frequency information for $|\xi| > \Omega/2$ for some $\Omega > 0$.
- This is the *first* source of error of DFT approximation to FT/FS.
- This truncation allows us to consider only k with $|k| \le A\Omega/2$.

$$\hat{f}(k/A) = \int_{-A/2}^{A/2} f(x) \mathrm{e}^{-2\pi \mathrm{i} kx/A} \, \mathrm{d}x = \left\langle f, \mathrm{e}^{2\pi \mathrm{i} k \cdot /A} \right\rangle = \sqrt{A}\alpha_k, \quad k \in \mathbb{Z}. \quad (1).$$

- In general, $f \in L^2[-A/2, A/2]$ is *not* a *band-limited* function; Recall the *uncertainty principles*!
- Therefore, to have a finite length vector representing the frequency samples (or the Fourier coefficients), we need to truncate the frequency information for $|\xi| > \Omega/2$ for some $\Omega > 0$.
- This is the *first* source of error of DFT approximation to FT/FS.
- This truncation allows us to consider only k with $|k| \le A\Omega/2$.

$$\hat{f}(k/A) = \int_{-A/2}^{A/2} f(x) \mathrm{e}^{-2\pi \mathrm{i} kx/A} \, \mathrm{d}x = \left\langle f, \mathrm{e}^{2\pi \mathrm{i} k \cdot /A} \right\rangle = \sqrt{A}\alpha_k, \quad k \in \mathbb{Z}. \quad (1).$$

- We now need to approximate the integration in (1) numerically. We use the *trapezoid rule*. Here is the *second* source of the error of DFT.
- Let's divide the interval $[-A/2, A/2]$ into N (*positive even integer*[1]) subintervals of equal length of $\Delta x = A/N$. Let $x_\ell = \ell \Delta x$, $\ell = (-N/2):(N/2)$ be the points used in the trapezoid rule. Let $g(x) = f(x)\mathrm{e}^{-2\pi \mathrm{i} kx/A}$. Then we have

$$\hat{f}(k/A) \approx \frac{\Delta x}{2}\left\{ g(-A/2) + 2 \sum_{\ell=-N/2+1}^{N/2-1} g(x_\ell) + g(A/2) \right\}.$$

- If we assume $f(-A/2) = f(A/2)$ (which we can do by extending f by reflection, windowing, or zero-padding followed by redefining A), then the above approximation is simplified:

$$\hat{f}(k/A) \approx \Delta x \sum_{\ell=-N/2+1}^{N/2} g(x_\ell) = \frac{A}{N} \sum_{\ell=-N/2+1}^{N/2} f(\ell A/N)\, \mathrm{e}^{-2\pi \mathrm{i} k\ell/N},$$

[1] All the subsequent matrix representations assume this. See [2, Sec. 3.1] for N being positive odd integer as well as the other cases, e.g., different starting and ending indices.

- We now need to approximate the integration in (1) numerically. We use the *trapezoid rule*. Here is the *second* source of the error of DFT.
- Let's divide the interval $[-A/2, A/2]$ into N (*positive even integer*[1]) subintervals of equal length of $\Delta x = A/N$. Let $x_\ell = \ell \Delta x$, $\ell = (-N/2):(N/2)$ be the points used in the trapezoid rule. Let $g(x) = f(x)\mathrm{e}^{-2\pi \mathrm{i} kx/A}$. Then we have

$$\hat{f}(k/A) \approx \frac{\Delta x}{2}\left\{ g(-A/2) + 2\sum_{\ell=-N/2+1}^{N/2-1} g(x_\ell) + g(A/2) \right\}.$$

- If we assume $f(-A/2) = f(A/2)$ (which we can do by extending f by reflection, windowing, or zero-padding followed by redefining A), then the above approximation is simplified:

$$\hat{f}(k/A) \approx \Delta x \sum_{\ell=-N/2+1}^{N/2} g(x_\ell) = \frac{A}{N} \sum_{\ell=-N/2+1}^{N/2} f(\ell A/N)\,\mathrm{e}^{-2\pi \mathrm{i} k\ell/N},$$

[1] All the subsequent matrix representations assume this. See [2, Sec. 3.1] for N being positive odd integer as well as the other cases, e.g., different starting and ending indices.

- We now need to approximate the integration in (1) numerically. We use the *trapezoid rule*. Here is the *second* source of the error of DFT.
- Let's divide the interval $[-A/2, A/2]$ into N (*positive even integer*[1]) subintervals of equal length of $\Delta x = A/N$. Let $x_\ell = \ell \Delta x$, $\ell = (-N/2):(N/2)$ be the points used in the trapezoid rule. Let $g(x) = f(x)\mathrm{e}^{-2\pi \mathrm{i} k x/A}$. Then we have

$$\hat{f}(k/A) \approx \frac{\Delta x}{2}\left\{ g(-A/2) + 2\sum_{\ell=-N/2+1}^{N/2-1} g(x_\ell) + g(A/2) \right\}.$$

- If we assume $f(-A/2) = f(A/2)$ (which we can do by extending f by reflection, windowing, or zero-padding followed by redefining A), then the above approximation is simplified:

$$\hat{f}(k/A) \approx \Delta x \sum_{\ell=-N/2+1}^{N/2} g(x_\ell) = \frac{A}{N}\sum_{\ell=-N/2+1}^{N/2} f(\ell A/N)\,\mathrm{e}^{-2\pi \mathrm{i} k \ell/N},$$

[1] All the subsequent matrix representations assume this. See [2, Sec. 3.1] for N being positive odd integer as well as the other cases, e.g., different starting and ending indices.

- Now, let $f_\ell := f(\ell A/N)$ Then, *the N-point DFT* is defined as follows:

$$F_k := \frac{1}{\sqrt{N}} \sum_{\ell=-N/2+1}^{N/2} f_\ell \, \mathrm{e}^{-2\pi \mathrm{i} k\ell/N}, \quad k = -\frac{N}{2}+1, \ldots, \frac{N}{2}. \tag{2}$$

- The factor $1/\sqrt{N}$ is to make DFT a *unitary* transformation (i.e., ℓ^2-norm (energy) preserving transformation, so that the Parseval & Plancherel equalities holds.)[2]
- We now have the following relationship.

$$\hat{f}(k/A) = \sqrt{A}\alpha_k \approx \frac{A}{\sqrt{N}} F_k.$$

- *The N-point inverse DFT* is defined, as you can imagine, as follows.

$$f_\ell := \frac{1}{\sqrt{N}} \sum_{k=-N/2+1}^{N/2} F_k \, \mathrm{e}^{2\pi \mathrm{i} k\ell/N}, \quad \ell = -\frac{N}{2}+1, \ldots, \frac{N}{2}.$$

The proof of this formula gets easier when we use the vector-matrix notation later in this lecture.

[2] Note that the definition used in the standard book [2] uses the factor $1/N$ instead, which makes DFT non-unitary. You need to be careful about various definitions of DFT!

- Now, let $f_\ell := f(\ell A/N)$ Then, *the N-point DFT* is defined as follows:

$$F_k := \frac{1}{\sqrt{N}} \sum_{\ell=-N/2+1}^{N/2} f_\ell \, \mathrm{e}^{-2\pi i k \ell / N}, \quad k = -\frac{N}{2}+1, \ldots, \frac{N}{2}. \tag{2}$$

- The factor $1/\sqrt{N}$ is to make DFT a *unitary* transformation (i.e., ℓ^2-norm (energy) preserving transformation, so that the Parseval & Plancherel equalities holds.)[2]
- We now have the following relationship.

$$\hat{f}(k/A) = \sqrt{A}\alpha_k \approx \frac{A}{\sqrt{N}} F_k.$$

- *The N-point inverse DFT* is defined, as you can imagine, as follows.

$$f_\ell := \frac{1}{\sqrt{N}} \sum_{k=-N/2+1}^{N/2} F_k \, \mathrm{e}^{2\pi i k \ell / N}, \quad \ell = -\frac{N}{2}+1, \ldots, \frac{N}{2}.$$

The proof of this formula gets easier when we use the vector-matrix notation later in this lecture.

[2] Note that the definition used in the standard book [2] uses the factor $1/N$ instead, which makes DFT non-unitary. You need to be careful about various definitions of DFT!

- Now, let $f_\ell := f(\ell A/N)$ Then, *the N-point DFT* is defined as follows:

$$F_k := \frac{1}{\sqrt{N}} \sum_{\ell=-N/2+1}^{N/2} f_\ell \, \mathrm{e}^{-2\pi \mathrm{i} k\ell/N}, \quad k = -\frac{N}{2}+1, \ldots, \frac{N}{2}. \tag{2}$$

- The factor $1/\sqrt{N}$ is to make DFT a *unitary* transformation (i.e., ℓ^2-norm (energy) preserving transformation, so that the Parseval & Plancherel equalities holds.)[2]
- We now have the following relationship.

$$\hat{f}(k/A) = \sqrt{A}\alpha_k \approx \frac{A}{\sqrt{N}} F_k.$$

- *The N-point inverse DFT* is defined, as you can imagine, as follows.

$$f_\ell := \frac{1}{\sqrt{N}} \sum_{k=-N/2+1}^{N/2} F_k \, \mathrm{e}^{2\pi \mathrm{i} k\ell/N}, \quad \ell = -\frac{N}{2}+1, \ldots, \frac{N}{2}.$$

The proof of this formula gets easier when we use the vector-matrix notation later in this lecture.

[2] Note that the definition used in the standard book [2] uses the factor $1/N$ instead, which makes DFT non-unitary. You need to be careful about various definitions of DFT!

- Now, let $f_\ell := f(\ell A/N)$ Then, *the N-point DFT* is defined as follows:

$$F_k := \frac{1}{\sqrt{N}} \sum_{\ell=-N/2+1}^{N/2} f_\ell \, \mathrm{e}^{-2\pi \mathrm{i} k\ell/N}, \quad k = -\frac{N}{2}+1, \ldots, \frac{N}{2}. \tag{2}$$

- The factor $1/\sqrt{N}$ is to make DFT a *unitary* transformation (i.e., ℓ^2-norm (energy) preserving transformation, so that the Parseval & Plancherel equalities holds.)[2]
- We now have the following relationship.

$$\hat{f}(k/A) = \sqrt{A}\alpha_k \approx \frac{A}{\sqrt{N}} F_k.$$

- *The N-point inverse DFT* is defined, as you can imagine, as follows.

$$f_\ell := \frac{1}{\sqrt{N}} \sum_{k=-N/2+1}^{N/2} F_k \, \mathrm{e}^{2\pi \mathrm{i} k\ell/N}, \quad \ell = -\frac{N}{2}+1, \ldots, \frac{N}{2}.$$

  The proof of this formula gets easier when we use the vector-matrix notation later in this lecture.

[2] Note that the definition used in the standard book [2] uses the factor $1/N$ instead, which makes DFT non-unitary. You need to be careful about various definitions of DFT!

Outline

The Reciprocity Relations

- Let $\Delta\xi$ be a sampling rate in the frequency domain, i.e., $\Delta\xi = 1/A$. Since we know $\Delta x = A/N$, and $k/A = \Omega/2$ at $k = N/2$ (highest frequency in consideration), we have the following fundamental relations:

$$\Delta x \Delta\xi = \frac{1}{N}, \quad A\Omega = N.$$

- Interpretation of these relations is very important! For example:
  - *For fixed N, $A \uparrow \Longrightarrow \Delta x \uparrow \ \Omega \downarrow \ \Delta\xi \downarrow$ (coarser space sampling leads to finer frequency sampling, but the frequency bandwidth also decreases).*
  - *For fixed A, $N \uparrow \Longrightarrow \Delta x \downarrow \ \Omega \uparrow \ \Delta\xi = \text{const.} = 1/A$ (finer space sampling leads to the increase of the frequency bandwidth).*

The Reciprocity Relations

- Let $\Delta\xi$ be a sampling rate in the frequency domain, i.e., $\Delta\xi = 1/A$. Since we know $\Delta x = A/N$, and $k/A = \Omega/2$ at $k = N/2$ (highest frequency in consideration), we have the following fundamental relations:

$$\Delta x \Delta\xi = \frac{1}{N}, \quad A\Omega = N.$$

- Interpretation of these relations is very important! For example:
  - *For fixed N, $A \uparrow \Longrightarrow \Delta x \uparrow\ \Omega \downarrow\ \Delta\xi \downarrow$ (coarser space sampling leads to finer frequency sampling, but the frequency bandwidth also decreases).*
  - *For fixed A, $N \uparrow \Longrightarrow \Delta x \downarrow\ \Omega \uparrow\ \Delta\xi \equiv const. = 1/A$ (finer space sampling leads to the increase of the frequency bandwidth).*

The Reciprocity Relations

- Let $\Delta\xi$ be a sampling rate in the frequency domain, i.e., $\Delta\xi = 1/A$. Since we know $\Delta x = A/N$, and $k/A = \Omega/2$ at $k = N/2$ (highest frequency in consideration), we have the following fundamental relations:

$$\Delta x \Delta\xi = \frac{1}{N}, \quad A\Omega = N.$$

- Interpretation of these relations is very important! For example:
  - *For fixed* N, $A \uparrow \Longrightarrow \Delta x \uparrow \; \Omega \downarrow \; \Delta\xi \downarrow$ *(coarser space sampling leads to finer frequency sampling, but the frequency bandwidth also decreases).*
  - *For fixed* A, $N \uparrow \Longrightarrow \Delta x \downarrow \; \Omega \uparrow \; \Delta\xi \equiv const. = 1/A$ *(finer space sampling leads to the increase of the frequency bandwidth).*

The Reciprocity Relations

- Let $\Delta\xi$ be a sampling rate in the frequency domain, i.e., $\Delta\xi = 1/A$. Since we know $\Delta x = A/N$, and $k/A = \Omega/2$ at $k = N/2$ (highest frequency in consideration), we have the following fundamental relations:

$$\Delta x \Delta\xi = \frac{1}{N}, \quad A\Omega = N.$$

- Interpretation of these relations is very important! For example:
  - *For fixed N, $A \uparrow \Longrightarrow \Delta x \uparrow \; \Omega \downarrow \; \Delta\xi \downarrow$ (coarser space sampling leads to finer frequency sampling, but the frequency bandwidth also decreases).*
  - *For fixed A, $N \uparrow \Longrightarrow \Delta x \downarrow \; \Omega \uparrow \; \Delta\xi \equiv \text{const.} = 1/A$ (finer space sampling leads to the increase of the frequency bandwidth).*

Outline

The vector-matrix notation of DFT

- We can gain great insights by expressing DFT using the *vector-matrix notation*. To do this, we need to define a couple of things.
- Let $\omega_N := \mathrm{e}^{2\pi\mathrm{i}/N}$, i.e., the *Nth root of unity*.
- Note that $\overline{\omega}_N = \omega_N^{-1}$; $\omega_N^0 = \omega_N^N = 1$; $\omega_N^{N/2} = -1$; and $\omega_N^{k+N} = \omega_N^k$ for any $k \in \mathbb{Z}$.
- Then, define a column vector:

$$\boldsymbol{w}_N^k := \frac{1}{\sqrt{N}}\left(\omega_N^{k\cdot 0}, \omega_N^{k\cdot 1}, \ldots, \omega_N^{k\cdot\frac{N}{2}}, \ldots, \omega_N^{k\cdot(N-1)}\right)^{\mathsf{T}}, \quad k = 0, \ldots, N-1.$$

- We also define another column vector:

$$\widetilde{\boldsymbol{w}}_N^k := \frac{1}{\sqrt{N}}\left(\omega_N^{k\cdot(-\frac{N}{2}+1)}, \omega_N^{k\cdot(-\frac{N}{2}+2)}, \ldots, \omega_N^{k\cdot 0}, \ldots, \omega_N^{k\cdot\frac{N}{2}}\right)^{\mathsf{T}}, \quad k = -\frac{N}{2}+1, \ldots, \frac{N}{2}.$$

The vector-matrix notation of DFT

- We can gain great insights by expressing DFT using the *vector-matrix notation*. To do this, we need to define a couple of things.
- Let $\omega_N := \mathrm{e}^{2\pi \mathrm{i}/N}$, i.e., the *$N$th root of unity*.
- Note that $\overline{\omega}_N = \omega_N^{-1}$; $\omega_N^0 = \omega_N^N = 1$; $\omega_N^{N/2} = -1$; and $\omega_N^{k+N} = \omega_N^k$ for any $k \in \mathbb{Z}$.
- Then, define a column vector:

$$\boldsymbol{w}_N^k := \frac{1}{\sqrt{N}} \left(\omega_N^{k\cdot 0}, \omega_N^{k\cdot 1}, \ldots, \omega_N^{k\cdot \frac{N}{2}}, \ldots, \omega_N^{k\cdot (N-1)} \right)^{\mathsf{T}}, \quad k = 0, \ldots, N-1.$$

- We also define another column vector:

$$\widetilde{\boldsymbol{w}}_N^k := \frac{1}{\sqrt{N}} \left(\omega_N^{k\cdot (-\frac{N}{2}+1)}, \omega_N^{k\cdot (-\frac{N}{2}+2)}, \ldots, \omega_N^{k\cdot 0}, \ldots, \omega_N^{k\cdot \frac{N}{2}} \right)^{\mathsf{T}}, \quad k = -\frac{N}{2}+1, \ldots, \frac{N}{2}.$$

The vector-matrix notation of DFT

- We can gain great insights by expressing DFT using the *vector-matrix notation*. To do this, we need to define a couple of things.
- Let $\omega_N := \mathrm{e}^{2\pi \mathrm{i}/N}$, i.e., the *$N$th root of unity*.
- Note that $\overline{\omega}_N = \omega_N^{-1}$; $\omega_N^0 = \omega_N^N = 1$; $\omega_N^{N/2} = -1$; and $\omega_N^{k+N} = \omega_N^k$ for any $k \in \mathbb{Z}$.
- Then, define a column vector:

$$\boldsymbol{w}_N^k := \frac{1}{\sqrt{N}} \left(\omega_N^{k\cdot 0}, \omega_N^{k\cdot 1}, \ldots, \omega_N^{k\cdot \frac{N}{2}}, \ldots, \omega_N^{k\cdot (N-1)} \right)^{\mathsf{T}}, \quad k = 0, \ldots, N-1.$$

- We also define another column vector:

$$\widetilde{\boldsymbol{w}}_N^k := \frac{1}{\sqrt{N}} \left(\omega_N^{k\cdot(-\frac{N}{2}+1)}, \omega_N^{k\cdot(-\frac{N}{2}+2)}, \ldots, \omega_N^{k\cdot 0}, \ldots, \omega_N^{k\cdot \frac{N}{2}} \right)^{\mathsf{T}}, \quad k = -\frac{N}{2}+1, \ldots, \frac{N}{2}.$$

The vector-matrix notation of DFT

- We can gain great insights by expressing DFT using the *vector-matrix notation*. To do this, we need to define a couple of things.
- Let $\omega_N := \mathrm{e}^{2\pi \mathrm{i}/N}$, i.e., the *$N$th root of unity*.
- Note that $\overline{\omega}_N = \omega_N^{-1}$; $\omega_N^0 = \omega_N^N = 1$; $\omega_N^{N/2} = -1$; and $\omega_N^{k+N} = \omega_N^k$ for any $k \in \mathbb{Z}$.
- Then, define a column vector:

$$\boldsymbol{w}_N^k := \frac{1}{\sqrt{N}} \left(\omega_N^{k\cdot 0}, \omega_N^{k\cdot 1}, \ldots, \omega_N^{k\cdot \frac{N}{2}}, \ldots, \omega_N^{k\cdot (N-1)} \right)^{\mathsf{T}}, \quad k = 0, \ldots, N-1.$$

- We also define another column vector:

$$\widetilde{\boldsymbol{w}}_N^k := \frac{1}{\sqrt{N}} \left(\omega_N^{k\cdot(-\frac{N}{2}+1)}, \omega_N^{k\cdot(-\frac{N}{2}+2)}, \ldots, \omega_N^{k\cdot 0}, \ldots, \omega_N^{k\cdot \frac{N}{2}} \right)^{\mathsf{T}}, \quad k = -\frac{N}{2}+1, \ldots, \frac{N}{2}.$$

The vector-matrix notation of DFT

- We can gain great insights by expressing DFT using the *vector-matrix notation*. To do this, we need to define a couple of things.
- Let $\omega_N := \mathrm{e}^{2\pi\mathrm{i}/N}$, i.e., the *$N$th root of unity*.
- Note that $\overline{\omega}_N = \omega_N^{-1}$; $\omega_N^0 = \omega_N^N = 1$; $\omega_N^{N/2} = -1$; and $\omega_N^{k+N} = \omega_N^k$ for any $k \in \mathbb{Z}$.
- Then, define a column vector:

$$\boldsymbol{w}_N^k := \frac{1}{\sqrt{N}} \left(\omega_N^{k\cdot 0}, \omega_N^{k\cdot 1}, \ldots, \omega_N^{k\cdot \frac{N}{2}}, \ldots, \omega_N^{k\cdot(N-1)} \right)^{\mathsf{T}}, \quad k = 0, \ldots, N-1.$$

- We also define another column vector:

$$\widetilde{\boldsymbol{w}}_N^k := \frac{1}{\sqrt{N}} \left(\omega_N^{k\cdot(-\frac{N}{2}+1)}, \omega_N^{k\cdot(-\frac{N}{2}+2)}, \ldots, \omega_N^{k\cdot 0}, \ldots, \omega_N^{k\cdot \frac{N}{2}} \right)^{\mathsf{T}}, \quad k = -\frac{N}{2}+1, \ldots, \frac{N}{2}.$$

- Using the properties of ω_N listed above, one can easily show that

$$\tilde{\boldsymbol{w}}_N^k = P_N \boldsymbol{w}_N^k, \quad P_N := T_N^{-1} S_N,$$

- T_N shifts vector entries *circularly* in one step "down", i.e., $T_N(a_1, \ldots, a_N)^{\mathsf{T}} = (a_N, a_1, \ldots, a_{N-1})^{\mathsf{T}}$.
- Note that $T_N^{-1} = T_N^{\mathsf{T}}$ represents the "up" circular shift operation.
- In MATLAB, $T_N^m \boldsymbol{a}$ corresponds to `circshift(a,m)` where m is an arbitrary integer (positive or negative).
- On the other hand, S_N is equivalent to `fftshift` in MATLAB:

$$S_N := \begin{bmatrix} O_{\frac{N}{2}} & I_{\frac{N}{2}} \\ I_{\frac{N}{2}} & O_{\frac{N}{2}} \end{bmatrix},$$

  i.e., $S_N(a_1, \ldots, a_N)^{\mathsf{T}} = \left(a_{\frac{N}{2}+1}, \ldots, a_N, a_1, \ldots, a_{\frac{N}{2}}\right)^{\mathsf{T}}$.
- Note that $S_N^{\mathsf{T}} = S_N^{-1} = S_N$.

- Using the properties of ω_N listed above, one can easily show that

$$\widetilde{\boldsymbol{w}}_N^k = P_N \boldsymbol{w}_N^k, \quad P_N := T_N^{-1} S_N,$$

- T_N shifts vector entries *circularly* in one step "down", i.e., $T_N(a_1, \ldots, a_N)^\mathsf{T} = (a_N, a_1, \ldots, a_{N-1})^\mathsf{T}$.
- Note that $T_N^{-1} = T_N^\mathsf{T}$ represents the "up" circular shift operation.
- In MATLAB, $T_N^m \boldsymbol{a}$ corresponds to `circshift(a,m)` where m is an arbitrary integer (positive or negative).
- On the other hand, S_N is equivalent to `fftshift` in MATLAB:

$$S_N := \begin{bmatrix} O_{\frac{N}{2}} & I_{\frac{N}{2}} \\ I_{\frac{N}{2}} & O_{\frac{N}{2}} \end{bmatrix},$$

i.e., $S_N(a_1, \ldots, a_N)^\mathsf{T} = \left(a_{\frac{N}{2}+1}, \ldots, a_N, a_1, \ldots, a_{\frac{N}{2}}\right)^\mathsf{T}$.

- Note that $S_N^\mathsf{T} = S_N^{-1} = S_N$.

- Using the properties of ω_N listed above, one can easily show that

$$\tilde{\boldsymbol{w}}_N^k = P_N \boldsymbol{w}_N^k, \quad P_N := T_N^{-1} S_N,$$

- T_N shifts vector entries *circularly* in one step "down", i.e., $T_N(a_1, \dots, a_N)^\mathsf{T} = (a_N, a_1, \dots, a_{N-1})^\mathsf{T}$.
- Note that $T_N^{-1} = T_N^\mathsf{T}$ represents the "up" circular shift operation.
- In MATLAB, $T_N^m \boldsymbol{a}$ corresponds to `circshift(a,m)` where m is an arbitrary integer (positive or negative).
- On the other hand, S_N is equivalent to `fftshift` in MATLAB:

$$S_N := \begin{bmatrix} O_{\frac{N}{2}} & I_{\frac{N}{2}} \\ I_{\frac{N}{2}} & O_{\frac{N}{2}} \end{bmatrix},$$

  i.e., $S_N(a_1, \dots, a_N)^\mathsf{T} = \left(a_{\frac{N}{2}+1}, \dots, a_N, a_1, \dots, a_{\frac{N}{2}}\right)^\mathsf{T}$.
- Note that $S_N^\mathsf{T} = S_N^{-1} = S_N$.

- Using the properties of ω_N listed above, one can easily show that

$$\widetilde{\boldsymbol{w}}_N^k = P_N \boldsymbol{w}_N^k, \quad P_N := T_N^{-1} S_N,$$

- T_N shifts vector entries *circularly* in one step "down", i.e., $T_N(a_1,\ldots,a_N)^\mathsf{T} = (a_N, a_1, \ldots, a_{N-1})^\mathsf{T}$.
- Note that $T_N^{-1} = T_N^\mathsf{T}$ represents the "up" circular shift operation.
- In MATLAB, $T_N^m \boldsymbol{a}$ corresponds to `circshift(a,m)` where m is an arbitrary integer (positive or negative).
- On the other hand, S_N is equivalent to `fftshift` in MATLAB:

$$S_N := \begin{bmatrix} O_{\frac{N}{2}} & I_{\frac{N}{2}} \\ I_{\frac{N}{2}} & O_{\frac{N}{2}} \end{bmatrix},$$

i.e., $S_N(a_1,\ldots,a_N)^\mathsf{T} = \left(a_{\frac{N}{2}+1},\ldots,a_N,a_1,\ldots,a_{\frac{N}{2}}\right)^\mathsf{T}$.
- Note that $S_N^\mathsf{T} = S_N^{-1} = S_N$.

- Using the properties of ω_N listed above, one can easily show that

$$\widetilde{\boldsymbol{w}}_N^k = P_N \boldsymbol{w}_N^k, \quad P_N := T_N^{-1} S_N,$$

- T_N shifts vector entries *circularly* in one step "down", i.e., $T_N(a_1, \ldots, a_N)^\mathsf{T} = (a_N, a_1, \ldots, a_{N-1})^\mathsf{T}$.
- Note that $T_N^{-1} = T_N^\mathsf{T}$ represents the "up" circular shift operation.
- In MATLAB, $T_N^m \boldsymbol{a}$ corresponds to `circshift(a,m)` where m is an arbitrary integer (positive or negative).
- On the other hand, S_N is equivalent to `fftshift` in MATLAB:

$$S_N := \begin{bmatrix} O_{\frac{N}{2}} & I_{\frac{N}{2}} \\ I_{\frac{N}{2}} & O_{\frac{N}{2}} \end{bmatrix},$$

i.e., $S_N(a_1, \ldots, a_N)^\mathsf{T} = \left(a_{\frac{N}{2}+1}, \ldots, a_N, a_1, \ldots, a_{\frac{N}{2}}\right)^\mathsf{T}$.

- Note that $S_N^\mathsf{T} = S_N^{-1} = S_N$.

- Using the properties of ω_N listed above, one can easily show that

$$\widetilde{\boldsymbol{w}}_N^k = P_N \boldsymbol{w}_N^k, \quad P_N := T_N^{-1} S_N,$$

- T_N shifts vector entries *circularly* in one step "down", i.e., $T_N(a_1, \ldots, a_N)^{\mathsf{T}} = (a_N, a_1, \ldots, a_{N-1})^{\mathsf{T}}$.
- Note that $T_N^{-1} = T_N^{\mathsf{T}}$ represents the "up" circular shift operation.
- In MATLAB, $T_N^m \boldsymbol{a}$ corresponds to `circshift(a,m)` where m is an arbitrary integer (positive or negative).
- On the other hand, S_N is equivalent to `fftshift` in MATLAB:

$$S_N := \begin{bmatrix} O_{\frac{N}{2}} & I_{\frac{N}{2}} \\ I_{\frac{N}{2}} & O_{\frac{N}{2}} \end{bmatrix},$$

  i.e., $S_N(a_1, \ldots, a_N)^{\mathsf{T}} = \left(a_{\frac{N}{2}+1}, \ldots, a_N, a_1, \ldots, a_{\frac{N}{2}}\right)^{\mathsf{T}}$.
- Note that $S_N^{\mathsf{T}} = S_N^{-1} = S_N$.

Just in case, the matrix representations of T_N and P_N are:

$$T_N := \begin{bmatrix} 0 & \cdots & \cdots & 0 & 1 \\ 1 & 0 & \cdots & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 & 0 \\ 0 & \cdots & 0 & 1 & 0 \end{bmatrix} \in \mathbb{R}^{N \times N};$$

$$P_N := \left[\begin{array}{cccc|cccc} 0 & \cdots & \cdots & 0 & 0 & 1 & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & \vdots & \vdots & \ddots & \ddots & 1 \\ 1 & 0 & \cdots & 0 & 0 & \cdots & \cdots & 0 \\ \hline 0 & 1 & \cdots & 0 & 0 & \cdots & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots & \vdots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 1 & 0 & \ddots & \ddots & \vdots \\ 0 & \cdots & \cdots & 0 & 1 & 0 & \cdots & 0 \end{array}\right] \in \mathbb{R}^{N \times N}.$$

- Let $\boldsymbol{f} = \left(f_{-\frac{N}{2}+1}, \ldots, f_{\frac{N}{2}}\right)^{\mathsf{T}}$ be a vector of sampled points $f_\ell = f(\ell \Delta x)$.
- Now DFT can be written as follows:

$$F_k = \left\langle \boldsymbol{f}, \widetilde{\boldsymbol{w}}_N^k \right\rangle, \quad k = -\frac{N}{2}+1, \ldots, \frac{N}{2}.$$

- Finally, define an *N-point DFT matrix* commonly used in the literature:

$$W_N := \left[\boldsymbol{w}_N^0 \;\middle|\; \boldsymbol{w}_N^1 \;\middle|\; \cdots \;\middle|\; \boldsymbol{w}_N^{N-1} \right]$$

- On the other hand, we define the following matrix compliant with our definition of DFT in Eq. (2):

$$\widetilde{W}_N := \left[\widetilde{\boldsymbol{w}}_N^{-\frac{N}{2}+1} \;\middle|\; \widetilde{\boldsymbol{w}}_N^{-\frac{N}{2}+2} \;\middle|\; \cdots \;\middle|\; \widetilde{\boldsymbol{w}}_N^{\frac{N}{2}} \right] = P_N W_N P_N^{\mathsf{T}}.$$

- Let $\boldsymbol{f} = \left(f_{-\frac{N}{2}+1}, \ldots, f_{\frac{N}{2}}\right)^{\mathsf{T}}$ be a vector of sampled points $f_\ell = f(\ell \Delta x)$.
- Now DFT can be written as follows:

$$F_k = \left\langle \boldsymbol{f}, \widetilde{\boldsymbol{w}}_N^k \right\rangle, \quad k = -\frac{N}{2}+1, \ldots, \frac{N}{2}.$$

- Finally, define an *N-point DFT matrix* commonly used in the literature:

$$W_N := \left[\boldsymbol{w}_N^0 \;\middle|\; \boldsymbol{w}_N^1 \;\middle|\; \cdots \;\middle|\; \boldsymbol{w}_N^{N-1} \right]$$

- On the other hand, we define the following matrix compliant with our definition of DFT in Eq. (2):

$$\widetilde{W}_N := \left[\widetilde{\boldsymbol{w}}_N^{-\frac{N}{2}+1} \;\middle|\; \widetilde{\boldsymbol{w}}_N^{-\frac{N}{2}+2} \;\middle|\; \cdots \;\middle|\; \widetilde{\boldsymbol{w}}_N^{\frac{N}{2}} \right] = P_N W_N P_N^{\mathsf{T}}.$$

- Let $\boldsymbol{f} = \left(f_{-\frac{N}{2}+1}, \ldots, f_{\frac{N}{2}}\right)^{\mathsf{T}}$ be a vector of sampled points $f_\ell = f(\ell \Delta x)$.
- Now DFT can be written as follows:

$$F_k = \left\langle \boldsymbol{f}, \widetilde{\boldsymbol{w}}_N^k \right\rangle, \quad k = -\frac{N}{2}+1, \ldots, \frac{N}{2}.$$

- Finally, define an *N-point DFT matrix* commonly used in the literature:

$$W_N := \left[\boldsymbol{w}_N^0 \;\middle|\; \boldsymbol{w}_N^1 \;\middle|\; \cdots \;\middle|\; \boldsymbol{w}_N^{N-1} \right]$$

- On the other hand, we define the following matrix compliant with our definition of DFT in Eq. (2):

$$\widetilde{W}_N := \left[\widetilde{\boldsymbol{w}}_N^{-\frac{N}{2}+1} \;\middle|\; \widetilde{\boldsymbol{w}}_N^{-\frac{N}{2}+2} \;\middle|\; \cdots \;\middle|\; \widetilde{\boldsymbol{w}}_N^{\frac{N}{2}} \right] = P_N W_N P_N^{\mathsf{T}}.$$

- Let $\boldsymbol{f} = \left(f_{-\frac{N}{2}+1}, \ldots, f_{\frac{N}{2}}\right)^{\mathsf{T}}$ be a vector of sampled points $f_\ell = f(\ell \Delta x)$.
- Now DFT can be written as follows:

$$F_k = \left\langle \boldsymbol{f}, \widetilde{\boldsymbol{w}}_N^k \right\rangle, \quad k = -\frac{N}{2}+1, \ldots, \frac{N}{2}.$$

- Finally, define an *N-point DFT matrix* commonly used in the literature:

$$W_N := \left[\boldsymbol{w}_N^0 \;\middle|\; \boldsymbol{w}_N^1 \;\middle|\; \cdots \;\middle|\; \boldsymbol{w}_N^{N-1} \right]$$

- On the other hand, we define the following matrix compliant with our definition of DFT in Eq. (2):

$$\widetilde{W}_N := \left[\widetilde{\boldsymbol{w}}_N^{-\frac{N}{2}+1} \;\middle|\; \widetilde{\boldsymbol{w}}_N^{-\frac{N}{2}+2} \;\middle|\; \cdots \;\middle|\; \widetilde{\boldsymbol{w}}_N^{\frac{N}{2}} \right] = P_N W_N P_N^{\mathsf{T}}.$$

- Let $\boldsymbol{F} = \left(F_{-\frac{N}{2}+1}, \ldots, F_{\frac{N}{2}}\right)^{\mathsf{T}} \in \mathbb{C}^N$.
- Then, the N-point DFT/IDFT can be conveniently written as:

$$\boldsymbol{F} = \widetilde{W}_N^* \boldsymbol{f}, \quad \boldsymbol{f} = \widetilde{W}_N \boldsymbol{F},$$

  where $\widetilde{W}_N^*$ is an hermitian conjugate (transposition followed by element-wise complex conjugation) of $\widetilde{W}_N$, and also often written as $\widetilde{W}_N^{\mathsf{H}}$ in literature.
- In fact, $\widetilde{W}_N^* = \left(P_N W_N P_N^{\mathsf{T}}\right)^* = P_N W_N^* P_N^{\mathsf{T}}$.
- We also denote $\mathscr{D}_N[\boldsymbol{f}] := \widetilde{W}_N^* \boldsymbol{f}$.

Theorem

Both W_N and $\widetilde{W}_N$ are N-by-N unitary matrix. In other words, both $\{\boldsymbol{w}_N^k\}_{k=0}^{N-1}$ and $\{\widetilde{\boldsymbol{w}}_N^k\}_{k=-\frac{N}{2}+1}^{\frac{N}{2}}$ are orthonormal bases of $\mathbb{C}^N$.

(Proof) Exercise. A main thing is to prove $\langle \boldsymbol{w}_N^k, \boldsymbol{w}_N^\ell \rangle = \delta_{k,\ell}$.

- Let $\boldsymbol{F} = \left(F_{-\frac{N}{2}+1}, \ldots, F_{\frac{N}{2}}\right)^{\mathsf{T}} \in \mathbb{C}^N$.
- Then, the N-point DFT/IDFT can be conveniently written as:

$$\boldsymbol{F} = \widetilde{W}_N^* \boldsymbol{f}, \quad \boldsymbol{f} = \widetilde{W}_N \boldsymbol{F},$$

where $\widetilde{W}_N^*$ is an hermitian conjugate (transposition followed by element-wise complex conjugation) of $\widetilde{W}_N$, and also often written as $\widetilde{W}_N^{\mathsf{H}}$ in literature.

- In fact, $\widetilde{W}_N^* = \left(P_N W_N P_N^{\mathsf{T}}\right)^* = P_N W_N^* P_N^{\mathsf{T}}$.
- We also denote $\mathscr{D}_N[\boldsymbol{f}] := \widetilde{W}_N^* \boldsymbol{f}$.

Theorem

Both W_N and $\widetilde{W}_N$ are N-by-N unitary matrix. In other words, both $\{\boldsymbol{w}_N^k\}_{k=0}^{N-1}$ and $\{\widetilde{\boldsymbol{w}}_N^k\}_{k=-\frac{N}{2}+1}^{\frac{N}{2}}$ are orthonormal bases of $\mathbb{C}^N$.

(Proof) Exercise. A main thing is to prove $\langle \boldsymbol{w}_N^k, \boldsymbol{w}_N^\ell \rangle = \delta_{k,\ell}$.

- Let $\boldsymbol{F} = \left(F_{-\frac{N}{2}+1}, \ldots, F_{\frac{N}{2}}\right)^{\mathsf{T}} \in \mathbb{C}^N$.
- Then, the N-point DFT/IDFT can be conveniently written as:

$$\boldsymbol{F} = \widetilde{W}_N^* \boldsymbol{f}, \quad \boldsymbol{f} = \widetilde{W}_N \boldsymbol{F},$$

  where $\widetilde{W}_N^*$ is an hermitian conjugate (transposition followed by element-wise complex conjugation) of $\widetilde{W}_N$, and also often written as $\widetilde{W}_N^{\mathsf{H}}$ in literature.
- In fact, $\widetilde{W}_N^* = \left(P_N W_N P_N^{\mathsf{T}}\right)^* = P_N W_N^* P_N^{\mathsf{T}}$.
- We also denote $\mathscr{D}_N[\boldsymbol{f}] := \widetilde{W}_N^* \boldsymbol{f}$.

Theorem

Both W_N and $\widetilde{W}_N$ are N-by-N unitary matrix. In other words, both $\{\boldsymbol{w}_N^k\}_{k=0}^{N-1}$ and $\{\widetilde{\boldsymbol{w}}_N^k\}_{k=-\frac{N}{2}+1}^{\frac{N}{2}}$ are orthonormal bases of $\mathbb{C}^N$.

(Proof) Exercise. A main thing is to prove $\langle \boldsymbol{w}_N^k, \boldsymbol{w}_N^\ell \rangle = \delta_{k,\ell}$.

- Let $\boldsymbol{F} = \left(F_{-\frac{N}{2}+1}, \ldots, F_{\frac{N}{2}}\right)^{\mathsf{T}} \in \mathbb{C}^N$.
- Then, the N-point DFT/IDFT can be conveniently written as:

$$\boldsymbol{F} = \widetilde{W}_N^* \boldsymbol{f}, \quad \boldsymbol{f} = \widetilde{W}_N \boldsymbol{F},$$

where $\widetilde{W}_N^*$ is an hermitian conjugate (transposition followed by element-wise complex conjugation) of $\widetilde{W}_N$, and also often written as $\widetilde{W}_N^{\mathsf{H}}$ in literature.

- In fact, $\widetilde{W}_N^* = \left(P_N W_N P_N^{\mathsf{T}}\right)^* = P_N W_N^* P_N^{\mathsf{T}}$.
- We also denote $\mathscr{D}_N[\boldsymbol{f}] := \widetilde{W}_N^* \boldsymbol{f}$.

Theorem

Both W_N and $\widetilde{W}_N$ are N-by-N unitary matrix. In other words, both $\{\boldsymbol{w}_N^k\}_{k=0}^{N-1}$ and $\{\widetilde{\boldsymbol{w}}_N^k\}_{k=-\frac{N}{2}+1}^{\frac{N}{2}}$ are orthonormal bases of $\mathbb{C}^N$.

(Proof) Exercise. A main thing is to prove $\langle \boldsymbol{w}_N^k, \boldsymbol{w}_N^\ell \rangle = \delta_{k,\ell}$.

- Let $\boldsymbol{F} = \left(F_{-\frac{N}{2}+1}, \ldots, F_{\frac{N}{2}}\right)^{\mathsf{T}} \in \mathbb{C}^N$.
- Then, the N-point DFT/IDFT can be conveniently written as:

$$\boldsymbol{F} = \widetilde{W}_N^* \boldsymbol{f}, \quad \boldsymbol{f} = \widetilde{W}_N \boldsymbol{F},$$

where $\widetilde{W}_N^*$ is an hermitian conjugate (transposition followed by element-wise complex conjugation) of $\widetilde{W}_N$, and also often written as $\widetilde{W}_N^{\mathsf{H}}$ in literature.

- In fact, $\widetilde{W}_N^* = \left(P_N W_N P_N^{\mathsf{T}}\right)^* = P_N W_N^* P_N^{\mathsf{T}}$.
- We also denote $\mathscr{D}_N[\boldsymbol{f}] := \widetilde{W}_N^* \boldsymbol{f}$.

Theorem

Both W_N and $\widetilde{W}_N$ are N-by-N unitary matrix. In other words, both $\{\boldsymbol{w}_N^k\}_{k=0}^{N-1}$ and $\{\widetilde{\boldsymbol{w}}_N^k\}_{k=-\frac{N}{2}+1}^{\frac{N}{2}}$ are orthonormal bases of $\mathbb{C}^N$.

(Proof) Exercise. A main thing is to prove $\langle \boldsymbol{w}_N^k, \boldsymbol{w}_N^\ell \rangle = \delta_{k,\ell}$.

- Let $\boldsymbol{F} = \left(F_{-\frac{N}{2}+1}, \ldots, F_{\frac{N}{2}}\right)^{\mathsf{T}} \in \mathbb{C}^N$.
- Then, the N-point DFT/IDFT can be conveniently written as:

$$\boldsymbol{F} = \widetilde{W}_N^* \boldsymbol{f}, \quad \boldsymbol{f} = \widetilde{W}_N \boldsymbol{F},$$

where $\widetilde{W}_N^*$ is an hermitian conjugate (transposition followed by element-wise complex conjugation) of $\widetilde{W}_N$, and also often written as $\widetilde{W}_N^{\mathsf{H}}$ in literature.

- In fact, $\widetilde{W}_N^* = \left(P_N W_N P_N^{\mathsf{T}}\right)^* = P_N W_N^* P_N^{\mathsf{T}}$.
- We also denote $\mathscr{D}_N[\boldsymbol{f}] := \widetilde{W}_N^* \boldsymbol{f}$.

Theorem

Both W_N and $\widetilde{W}_N$ are N-by-N unitary matrix. In other words, both $\{\boldsymbol{w}_N^k\}_{k=0}^{N-1}$ and $\{\widetilde{\boldsymbol{w}}_N^k\}_{k=-\frac{N}{2}+1}^{\frac{N}{2}}$ are orthonormal bases of $\mathbb{C}^N$.

(Proof) Exercise. A main thing is to prove $\langle \boldsymbol{w}_N^k, \boldsymbol{w}_N^\ell \rangle = \delta_{k,\ell}$.

Theorem

All the eigenvalues of W_N and $\widetilde{W}_N$ are $1, -1, \mathrm{i}, -\mathrm{i}$.

(Proof) See e.g., [1, 2, 3]. Note that from this theorem we have $W_N^4 = \widetilde{W}_N^4 = I_N$.

Theorem (McClellan-Parks [3])

The multiplicities of the eigenvalues of W_N are summarized as:

N	mult(1)	mult(−1)	mult(i)	mult(−i)
$4m$	$m+1$	m	m	$m-1$
$4m+1$	$m+1$	m	m	m
$4m+2$	$m+1$	$m+1$	m	m
$4m+3$	$m+1$	$m+1$	$m+1$	m

Research Opportunity: W_N and $\widetilde{W}_N$ are already the ONBs of $\mathbb{C}^N$. *What is the use of their eigenvectors?*

Theorem

All the eigenvalues of W_N *and* $\widetilde{W}_N$ *are* $1, -1, \mathrm{i}, -\mathrm{i}$.

(Proof) See e.g., [1, 2, 3]. Note that from this theorem we have $W_N^4 = \widetilde{W}_N^4 = I_N$.

Theorem (McClellan-Parks [3])

The multiplicities of the eigenvalues of W_N *are summarized as:*

N	mult(1)	mult(−1)	mult(i)	mult(−i)
$4m$	$m+1$	m	m	$m-1$
$4m+1$	$m+1$	m	m	m
$4m+2$	$m+1$	$m+1$	m	m
$4m+3$	$m+1$	$m+1$	$m+1$	m

Research Opportunity: W_N and $\widetilde{W}_N$ are already the ONBs of $\mathbb{C}^N$. *What is the use of their eigenvectors?*

Theorem

All the eigenvalues of W_N and $\widetilde{W}_N$ are $1, -1, \mathrm{i}, -\mathrm{i}$.

(Proof) See e.g., [1, 2, 3]. Note that from this theorem we have $W_N^4 = \widetilde{W}_N^4 = I_N$.

Theorem (McClellan-Parks [3])

The multiplicities of the eigenvalues of W_N are summarized as:

N	mult(1)	mult(−1)	mult(i)	mult(−i)
$4m$	$m+1$	m	m	$m-1$
$4m+1$	$m+1$	m	m	m
$4m+2$	$m+1$	$m+1$	m	m
$4m+3$	$m+1$	$m+1$	$m+1$	m

Research Opportunity: W_N and $\widetilde{W}_N$ are already the ONBs of $\mathbb{C}^N$. *What is the use of their eigenvectors?*

Theorem

All the eigenvalues of W_N *and* $\widetilde{W}_N$ *are* $1, -1, \mathrm{i}, -\mathrm{i}$.

(Proof) See e.g., [1, 2, 3]. Note that from this theorem we have $W_N^4 = \widetilde{W}_N^4 = I_N$.

Theorem (McClellan-Parks [3])

The multiplicities of the eigenvalues of W_N *are summarized as:*

N	mult(1)	mult(−1)	mult(i)	mult(−i)
$4m$	$m+1$	m	m	$m-1$
$4m+1$	$m+1$	m	m	m
$4m+2$	$m+1$	$m+1$	m	m
$4m+3$	$m+1$	$m+1$	$m+1$	m

Research Opportunity: W_N and $\widetilde{W}_N$ are already the ONBs of $\mathbb{C}^N$. *What is the use of their eigenvectors?*

Outline

Using the properties of ω_N, in particular the periodicity with period N, we have:

$$W_N^* = \begin{bmatrix} (\boldsymbol{w}_N^0)^* \\ (\boldsymbol{w}_N^1)^* \\ (\boldsymbol{w}_N^2)^* \\ \vdots \\ (\boldsymbol{w}_N^{N/2})^* \\ \vdots \\ (\boldsymbol{w}_N^{N-1})^* \end{bmatrix} = \frac{1}{\sqrt{N}} \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & \overline{\omega}_N^1 & \overline{\omega}_N^2 & \cdots & \overline{\omega}_N^{N-1} \\ 1 & \overline{\omega}_N^2 & \overline{\omega}_N^4 & \cdots & \overline{\omega}_N^{2(N-1)} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 1 & \overline{\omega}_N^{N/2} & \overline{\omega}_N^{2N/2} & \cdots & \overline{\omega}_N^{(N-1)N/2} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 1 & \overline{\omega}_N^{N-1} & \overline{\omega}_N^{2(N-1)} & \cdots & \overline{\omega}_N^{(N-1)(N-1)} \end{bmatrix}$$

$$= \frac{1}{\sqrt{N}} \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & \omega_N^{-1} & \omega_N^{-2} & \cdots & \omega_N^{-(N-1)} \\ 1 & \omega_N^{-2} & \omega_N^{-4} & \cdots & \omega_N^{-2(N-1)} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 1 & \omega_N^{-N/2+1} & \omega_N^{2(-N/2+1)} & \cdots & \omega_N^{(N-1)(-N/2+1)} \\ 1 & \omega_N^{-N/2} & \omega_N^{-2N/2} & \cdots & \omega_N^{-(N-1)N/2} \\ 1 & \omega_N^{N/2-1} & \omega_N^{2(N/2-1)} & \cdots & \omega_N^{(N-1)(N/2-1)} \\ 1 & \omega_N^{N/2-2} & \omega_N^{2(N/2-2)} & \cdots & \omega_N^{(N-1)(N/2-2)} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 1 & \omega_N^{1} & \omega_N^{2} & \cdots & \omega_N^{N-1} \end{bmatrix}.$$

Using the properties of ω_N, in particular the periodicity with period N, we have:

$$
W_N^* = \begin{bmatrix} (\boldsymbol{w}_N^0)^* \\ (\boldsymbol{w}_N^1)^* \\ (\boldsymbol{w}_N^2)^* \\ \vdots \\ (\boldsymbol{w}_N^{N/2})^* \\ \vdots \\ (\boldsymbol{w}_N^{N-1})^* \end{bmatrix} = \frac{1}{\sqrt{N}} \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & \overline{\omega}_N^1 & \overline{\omega}_N^2 & \cdots & \overline{\omega}_N^{N-1} \\ 1 & \overline{\omega}_N^2 & \overline{\omega}_N^4 & \cdots & \overline{\omega}_N^{2(N-1)} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 1 & \overline{\omega}_N^{N/2} & \overline{\omega}_N^{2N/2} & \cdots & \overline{\omega}_N^{(N-1)N/2} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 1 & \overline{\omega}_N^{N-1} & \overline{\omega}_N^{2(N-1)} & \cdots & \overline{\omega}_N^{(N-1)(N-1)} \end{bmatrix}
$$

$$
= \frac{1}{\sqrt{N}} \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & \omega_N^{-1} & \omega_N^{-2} & \cdots & \omega_N^{-(N-1)} \\ 1 & \omega_N^{-2} & \omega_N^{-4} & \cdots & \omega_N^{-2(N-1)} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 1 & \omega_N^{-N/2+1} & \omega_N^{2(-N/2+1)} & \cdots & \omega_N^{(N-1)(-N/2+1)} \\ 1 & \omega_N^{-N/2} & \omega_N^{-2N/2} & \cdots & \omega_N^{-(N-1)N/2} \\ 1 & \omega_N^{N/2-1} & \omega_N^{2(N/2-1)} & \cdots & \omega_N^{(N-1)(N/2-1)} \\ 1 & \omega_N^{N/2-2} & \omega_N^{2(N/2-2)} & \cdots & \omega_N^{(N-1)(N/2-2)} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 1 & \omega_N^{1} & \omega_N^{2} & \cdots & \omega_N^{N-1} \end{bmatrix}.
$$

Using the properties of ω_N, in particular the periodicity with period N, we have:

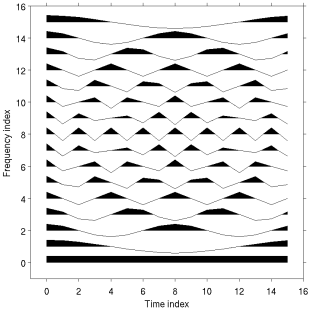
$$W_N^* = \begin{bmatrix} (\boldsymbol{w}_N^0)^* \\ (\boldsymbol{w}_N^1)^* \\ (\boldsymbol{w}_N^2)^* \\ \vdots \\ (\boldsymbol{w}_N^{N/2})^* \\ \vdots \\ (\boldsymbol{w}_N^{N-1})^* \end{bmatrix} = \frac{1}{\sqrt{N}} \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & \overline{\omega}_N^1 & \overline{\omega}_N^2 & \dots & \overline{\omega}_N^{N-1} \\ 1 & \overline{\omega}_N^2 & \overline{\omega}_N^4 & \dots & \overline{\omega}_N^{2(N-1)} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 1 & \overline{\omega}_N^{N/2} & \overline{\omega}_N^{2N/2} & \dots & \overline{\omega}_N^{(N-1)N/2} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 1 & \overline{\omega}_N^{N-1} & \overline{\omega}_N^{2(N-1)} & \dots & \overline{\omega}_N^{(N-1)(N-1)} \end{bmatrix}$$

$$= \frac{1}{\sqrt{N}} \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & \omega_N^{-1} & \omega_N^{-2} & \dots & \omega_N^{-(N-1)} \\ 1 & \omega_N^{-2} & \omega_N^{-4} & \dots & \omega_N^{-2(N-1)} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 1 & \omega_N^{-N/2+1} & \omega_N^{2(-N/2+1)} & \dots & \omega_N^{(N-1)(-N/2+1)} \\ 1 & \omega_N^{-N/2} & \omega_N^{-2N/2} & \dots & \omega_N^{-(N-1)N/2} \\ 1 & \omega_N^{N/2-1} & \omega_N^{2(N/2-1)} & \dots & \omega_N^{(N-1)(N/2-1)} \\ 1 & \omega_N^{N/2-2} & \omega_N^{2(N/2-2)} & \dots & \omega_N^{(N-1)(N/2-2)} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 1 & \omega_N^1 & \omega_N^2 & \dots & \omega_N^{N-1} \end{bmatrix}.$$

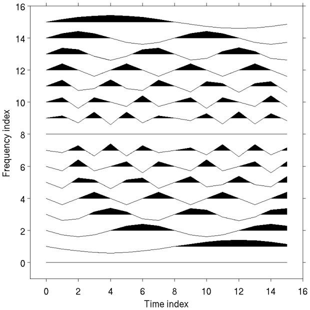
The following figures show the matrix W_N^* with $N = 16$ as waveforms.

Note that the first row vector of a matrix is displayed in the bottom while the last row in the top in each figure.

The following figures show the matrix W_N^* with $N = 16$ as waveforms.



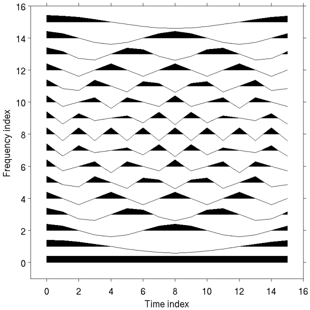
(a) $\mathrm{Re}(W^*)$



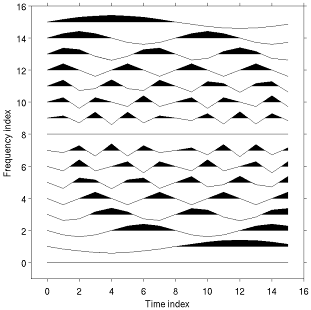
(b) $\mathrm{Im}(W^*)$

Note that the first row vector of a matrix is displayed in the bottom while the last row in the top in each figure.

The following figures show the matrix W_N^* with $N = 16$ as waveforms.



(a) $\mathrm{Re}(W^*)$



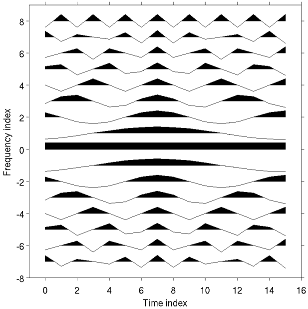
(b) $\mathrm{Im}(W^*)$

Note that the first row vector of a matrix is displayed in the bottom while the last row in the top in each figure.

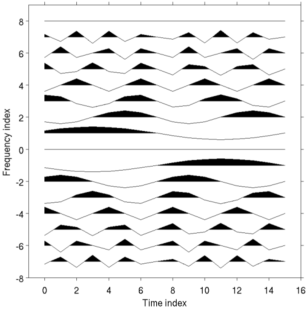
Now, how about $\widetilde{W}_N^*$?

Note the change of the locations of the basis vectors as well as symmetry $(W_N^*)^{\mathsf{T}} = W_N^*$, $(\widetilde{W}_N^*)^{\mathsf{T}} = \widetilde{W}_N^*$.

Now, how about $\widetilde{W}_N^*$?



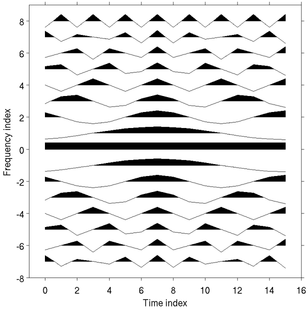
(a) $\mathrm{Re}(\widetilde{W}^*)$



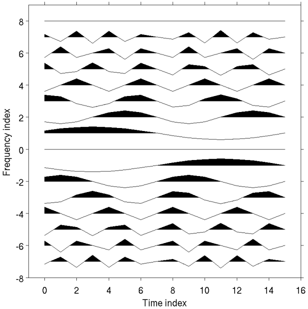
(b) $\mathrm{Im}((\widetilde{W}^*)$

Note the change of the locations of the basis vectors as well as symmetry $(W_N^*)^\mathsf{T} = W_N^*$, $(\widetilde{W}_N^*)^\mathsf{T} = \widetilde{W}_N^*$.

Now, how about $\widetilde{W}_N^*$?



(a) $\mathrm{Re}(\widetilde{W}^*)$



(b) $\mathrm{Im}((\widetilde{W}^*)$

Note the change of the locations of the basis vectors as well as symmetry $(W_N^*)^{\mathsf{T}} = W_N^*$, $(\widetilde{W}_N^*)^{\mathsf{T}} = \widetilde{W}_N^*$.

Outline

- It is amazing to know that the definition of DFT varies with the software systems one uses. We should always be careful what is the exact definition of the DFT for each software system.

MATLAB, R, S-Plus: $F_k = \sum_{\ell=1}^{N} f_\ell \, \mathrm{e}^{-2\pi \mathrm{i}(k-1)(\ell-1)/N}$ for $k = 1:N$.

Mathematica: $F_k = \frac{1}{\sqrt{N}} \sum_{\ell=1}^{N} f_\ell \, \mathrm{e}^{2\pi \mathrm{i}(k-1)(\ell-1)/N}$ for $k = 1:N$.

Maple: $F_{k+1} = \frac{1}{N} \sum_{\ell=0}^{N-1} f_\ell \, \mathrm{e}^{-2\pi \mathrm{i} k\ell/N}$ for $k = 0:(N-1)$.

MathCad: $F_k = \frac{1}{\sqrt{N}} \sum_{\ell=0}^{N-1} f_\ell \, \mathrm{e}^{2\pi \mathrm{i} k\ell/N}$ for $k = 0:(N-1)$.

- Hence, the DFT we defined in this lecture, i.e., $\boldsymbol{F} = \widetilde{W}_N^* \boldsymbol{f}$, can be realized by the following MATLAB command (assuming that `f` is a 1D vector):

```
F=circshift(fftshift(fft(fftshift(circshift(f,1)))),-1)/sqrt(length(f));
```

- It is amazing to know that the definition of DFT varies with the software systems one uses. We should always be careful what is the exact definition of the DFT for each software system.

MATLAB, R, S-Plus: $F_k = \sum_{\ell=1}^{N} f_\ell \, \mathrm{e}^{-2\pi \mathrm{i}(k-1)(\ell-1)/N}$ for $k = 1:N$.

Mathematica: $F_k = \frac{1}{\sqrt{N}} \sum_{\ell=1}^{N} f_\ell \, \mathrm{e}^{2\pi \mathrm{i}(k-1)(\ell-1)/N}$ for $k = 1:N$.

Maple: $F_{k+1} = \frac{1}{N} \sum_{\ell=0}^{N-1} f_\ell \, \mathrm{e}^{-2\pi \mathrm{i} k\ell/N}$ for $k = 0:(N-1)$.

MathCad: $F_k = \frac{1}{\sqrt{N}} \sum_{\ell=0}^{N-1} f_\ell \, \mathrm{e}^{2\pi \mathrm{i} k\ell/N}$ for $k = 0:(N-1)$.

- Hence, the DFT we defined in this lecture, i.e., $\boldsymbol{F} = \widetilde{W}_N^* \boldsymbol{f}$, can be realized by the following MATLAB command (assuming that `f` is a 1D vector):

```
F=circshift(fftshift(fft(fftshift(circshift(f,1)))),-1)/sqrt(length(f));
```

- It is amazing to know that the definition of DFT varies with the software systems one uses. We should always be careful what is the exact definition of the DFT for each software system.

MATLAB, R, S-Plus: $F_k = \sum_{\ell=1}^{N} f_\ell \, \mathrm{e}^{-2\pi \mathrm{i}(k-1)(\ell-1)/N}$ for $k = 1:N$.

Mathematica: $F_k = \frac{1}{\sqrt{N}} \sum_{\ell=1}^{N} f_\ell \, \mathrm{e}^{2\pi \mathrm{i}(k-1)(\ell-1)/N}$ for $k = 1:N$.

Maple: $F_{k+1} = \frac{1}{N} \sum_{\ell=0}^{N-1} f_\ell \, \mathrm{e}^{-2\pi \mathrm{i} k\ell/N}$ for $k = 0:(N-1)$.

MathCad: $F_k = \frac{1}{\sqrt{N}} \sum_{\ell=0}^{N-1} f_\ell \, \mathrm{e}^{2\pi \mathrm{i} k\ell/N}$ for $k = 0:(N-1)$.

- Hence, the DFT we defined in this lecture, i.e., $\boldsymbol{F} = \widetilde{W}_N^* \boldsymbol{f}$, can be realized by the following MATLAB command (assuming that f is a 1D vector):

```
F=circshift(fftshift(fft(fftshift(circshift(f,1)))),-1)/sqrt(length(f));
```

- It is amazing to know that the definition of DFT varies with the software systems one uses. We should always be careful what is the exact definition of the DFT for each software system.

MATLAB, R, S-Plus: $F_k = \sum_{\ell=1}^{N} f_\ell \, \mathrm{e}^{-2\pi \mathrm{i}(k-1)(\ell-1)/N}$ for $k = 1:N$.

Mathematica: $F_k = \frac{1}{\sqrt{N}} \sum_{\ell=1}^{N} f_\ell \, \mathrm{e}^{2\pi \mathrm{i}(k-1)(\ell-1)/N}$ for $k = 1:N$.

Maple: $F_{k+1} = \frac{1}{N} \sum_{\ell=0}^{N-1} f_\ell \, \mathrm{e}^{-2\pi \mathrm{i} k\ell/N}$ for $k = 0:(N-1)$.

MathCad: $F_k = \frac{1}{\sqrt{N}} \sum_{\ell=0}^{N-1} f_\ell \, \mathrm{e}^{2\pi \mathrm{i} k\ell/N}$ for $k = 0:(N-1)$.

- Hence, the DFT we defined in this lecture, i.e., $\boldsymbol{F} = \widetilde{W}_N^* \boldsymbol{f}$, can be realized by the following MATLAB command (assuming that `f` is a 1D vector):

```
F=circshift(fftshift(fft(fftshift(circshift(f,1)))),-1)/sqrt(length(f));
```

- It is amazing to know that the definition of DFT varies with the software systems one uses. We should always be careful what is the exact definition of the DFT for each software system.

MATLAB, R, S-Plus: $F_k = \sum_{\ell=1}^{N} f_\ell \, \mathrm{e}^{-2\pi \mathrm{i}(k-1)(\ell-1)/N}$ for $k = 1:N$.

Mathematica: $F_k = \frac{1}{\sqrt{N}} \sum_{\ell=1}^{N} f_\ell \, \mathrm{e}^{2\pi \mathrm{i}(k-1)(\ell-1)/N}$ for $k = 1:N$.

Maple: $F_{k+1} = \frac{1}{N} \sum_{\ell=0}^{N-1} f_\ell \, \mathrm{e}^{-2\pi \mathrm{i} k\ell/N}$ for $k = 0:(N-1)$.

MathCad: $F_k = \frac{1}{\sqrt{N}} \sum_{\ell=0}^{N-1} f_\ell \, \mathrm{e}^{2\pi \mathrm{i} k\ell/N}$ for $k = 0:(N-1)$.

- Hence, the DFT we defined in this lecture, i.e., $\boldsymbol{F} = \widetilde{W}_N^* \boldsymbol{f}$, can be realized by the following MATLAB command (assuming that `f` is a 1D vector):

```
F=circshift(fftshift(fft(fftshift(circshift(f,1)))),-1)/sqrt(length(f));
```

- It is amazing to know that the definition of DFT varies with the software systems one uses. We should always be careful what is the exact definition of the DFT for each software system.

MATLAB, R, S-Plus: $F_k = \sum_{\ell=1}^{N} f_\ell \, \mathrm{e}^{-2\pi\mathrm{i}(k-1)(\ell-1)/N}$ for $k = 1:N$.

Mathematica: $F_k = \frac{1}{\sqrt{N}} \sum_{\ell=1}^{N} f_\ell \, \mathrm{e}^{2\pi\mathrm{i}(k-1)(\ell-1)/N}$ for $k = 1:N$.

Maple: $F_{k+1} = \frac{1}{N} \sum_{\ell=0}^{N-1} f_\ell \, \mathrm{e}^{-2\pi\mathrm{i}k\ell/N}$ for $k = 0:(N-1)$.

MathCad: $F_k = \frac{1}{\sqrt{N}} \sum_{\ell=0}^{N-1} f_\ell \, \mathrm{e}^{2\pi\mathrm{i}k\ell/N}$ for $k = 0:(N-1)$.

- Hence, the DFT we defined in this lecture, i.e., $\boldsymbol{F} = \widetilde{W}_N^* \boldsymbol{f}$, can be realized by the following MATLAB command (assuming that `f` is a 1D vector):

```
F=circshift(fftshift(fft(fftshift(circshift(f,1)))),-1)/sqrt(length(f));
```

- It is amazing to know that the definition of DFT varies with the software systems one uses. We should always be careful what is the exact definition of the DFT for each software system.

MATLAB, R, S-Plus: $F_k = \sum_{\ell=1}^{N} f_\ell \, e^{-2\pi i (k-1)(\ell-1)/N}$ for $k = 1 : N$.

Mathematica: $F_k = \frac{1}{\sqrt{N}} \sum_{\ell=1}^{N} f_\ell \, e^{2\pi i (k-1)(\ell-1)/N}$ for $k = 1 : N$.

Maple: $F_{k+1} = \frac{1}{N} \sum_{\ell=0}^{N-1} f_\ell \, e^{-2\pi i k\ell/N}$ for $k = 0 : (N-1)$.

MathCad: $F_k = \frac{1}{\sqrt{N}} \sum_{\ell=0}^{N-1} f_\ell \, e^{2\pi i k\ell/N}$ for $k = 0 : (N-1)$.

- Hence, the DFT we defined in this lecture, i.e., $\boldsymbol{F} = \widetilde{W}_N^* \boldsymbol{f}$, can be realized by the following MATLAB command (assuming that `f` is a 1D vector):

```
F=circshift(fftshift(fft(fftshift(circshift(f,1)))),-1)/sqrt(length(f));
```

Outline

References

For more information about the DFT including higher-dimensional versions, see [2].
Also, the DFT matrix has more *profound* properties. See the challenging and deep paper by [1].

[1] L. Auslander and R. Tolimieri, *Is computing with the finite Fourier transform pure or applied mathematics?*, Bull. Amer. Math. Soc., 1 (1979), pp. 847–897.

[2] W. L. Briggs and V. E. Henson, *The DFT: An Owner's Manual for the Discrete Fourier Transform*, SIAM, Philadelphia, PA, 1995.

[3] J. H. McClellan and T. W. Parks, *Eigenvalue and eigenvector decomposition of the discrete Fourier transform*, IEEE Trans. Audio Electacoust., AU-20 (1972), pp. 66–65.