Discrete Short Time Fourier Transform

Andersen Ang

Mathématique et recherche opérationnelle
UMONS, Belgium

manshun.ang@umons.ac.be   Homepage: angms.science

First draft: November 26 2020
Last update: December 14, 2020

Fourier Transform (FT)

- FT of a given a function $f(x) : \mathbb{R} \to \mathbb{R}$, is a function F defined as

$$F(\xi) := \int_{\infty}^{\infty} f(x) e^{-2\pi i x \xi} dx.$$

- $|F(\xi)|$ tells the correlation between $f(x)$ and $e^{-2\pi i x \xi}$ at ξ.

- In signal processing, $f(x)$ is interpreted as a time-domain signal and $F(\xi)$ is interpreted as the frequency-domain spectrum.

- If f is absolutely integrable, then its FT exists. In signal processing it means f has finite energy.

- If we discretize FT, the infinite-dimensional f now becomes a finite-dimensional vector $\mathbf{x}$, and we have the Discrete FT.

Discrete Fourier Transform (DFT)

- The (N-point) DFT of a given vector $\mathbf{x} \in \mathbb{R}^N$, denoted as $\mathbf{y}$, is

$$y(k) := \sum_{n=0}^{N-1} x(n) e^{-i\frac{2\pi}{N}kn}.$$

  Note: $\mathbf{x}$ has N elements but we treat $\mathbf{x}$ as $[x(0), x(1), \ldots, x(N)]$. This is just a convention.

- $\mathbf{y} \in \mathbb{C}^N$.

- Let $\mathbf{e}_k$ be vector of $e^{-i\frac{2\pi}{N}kn}$ for $k \in [0 : N-1]$, then $\mathbf{y} = \langle \mathbf{x}, \mathbf{e}_k \rangle$, where $\langle \mathbf{a}, \mathbf{b} \rangle = \mathbf{x}^\top \mathbf{b}^* = \sum_k a_k b_k^*$ and $*$ is the complex conjugate.

- By the interpretation of inner product $\langle \cdot, \cdot \rangle$, then $|y(k)|$ tells the degree of similarity between $\mathbf{x}$ and $\mathbf{e}_k$, revealing its spectra component.

Matrix implementation of DFT

$$y(k) := \sum_{n=0}^{N-1} x(n) e^{-i\frac{2\pi}{N}kn}. \tag{1}$$

- From (1), it looks tempting to implement DFT by for-loop.
- Let $\mathbf{\Psi}_N(n,k) = e^{-i\frac{2\pi}{N}kn}$ for $n \in [0 : N-1]$ and $k \in [0 : N-1]$. Then $\mathbf{y} = \mathbf{\Psi}_N \mathbf{x}$ where

$$\mathbf{\Psi}_N = \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & \sigma_N & \sigma_N^2 & \dots & \sigma_N^{N-1} \\ 1 & \sigma_N^2 & \sigma_N^4 & \dots & \sigma_N^{2(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \sigma_N^{N-1} & \sigma_N^{2(N-1)} & \dots & \sigma_N^{(N-1)(N-1)} \end{bmatrix}, \quad \sigma_N = \exp\left(-\frac{2\pi i}{N}\right).$$

The motivation of σ_N is base on the property of Nth root of unity.

Motivation of Short-time Fourier Transform (STFT)

- Inspecting the DFT equation:

$$y(k) := \sum_{n=0}^{N-1} x(n) e^{-i\frac{2\pi}{N}kn}.$$

  As we get the frequency information using the entire time domain, DFT cannot tell when does the frequency change occur.

- Gabor's idea (1946): perform DFT on a local portion of $\mathbf{x}$, then we can tell when does frequency change.

- Such idea can be realized by using a window function $\mathbf{w}$, called "analysis windows" in signal processing.

STFT

$$Y(k,m) := \sum_{n=0}^{N-1} w(n)x(n+mH)e^{-i\frac{2\pi}{N}kn}.$$

- Now $\mathbf{x} \in \mathbb{R}^L$ not $\mathbb{R}^N$, although $L = N$ is possible.
- $[x(0+mH) : x(N-1+mH)]$ is a segment of $\mathbf{x} \in \mathbb{R}^L$.
- $H \geq 0$ is called hop size, this is a shift parameter.
- $\mathbf{w} \in \mathbb{R}^N$: $[w(0), w(1), \ldots, w(N-1)]$.
- $m \in [0, M]$ is the frame index and $M = \left\lfloor \frac{L-N}{H} \right\rfloor$ is the maximal frame index .
- $k \in [0 : K]$ is the frequency bin index with $K = N-1$, and $k = \lfloor \frac{N}{2} \rfloor$ is the frequency bin corresponding to the Nyquist frequency.
- So STFT is like $\mathbb{R}^L \to \mathbb{C}^{K,M+1}$ with $K = N-1, M = \left\lfloor \frac{L-N}{H} \right\rfloor$.

Matrix implementation of STFT

$$Y(k,m) := \sum_{n=0}^{N-1} w(n)x(n+mH)e^{-i\frac{2\pi}{N}kn}.$$

- Let $\mathbf{x}_{\vec{m}}$ be a sub-sequence of $\mathbf{x}$ defined as

$$\mathbf{x}_{\vec{m}} = [x(0+mH), \ldots, x(N-1+mH)].$$

  Depends onn H, the vectors $\mathbf{x}_{\vec{m}}$ may overlap with each other.

- Suppose now $w(n) = 1$, then STFT at time frame m = DFT on $\mathbf{x}_{\vec{m}}$:

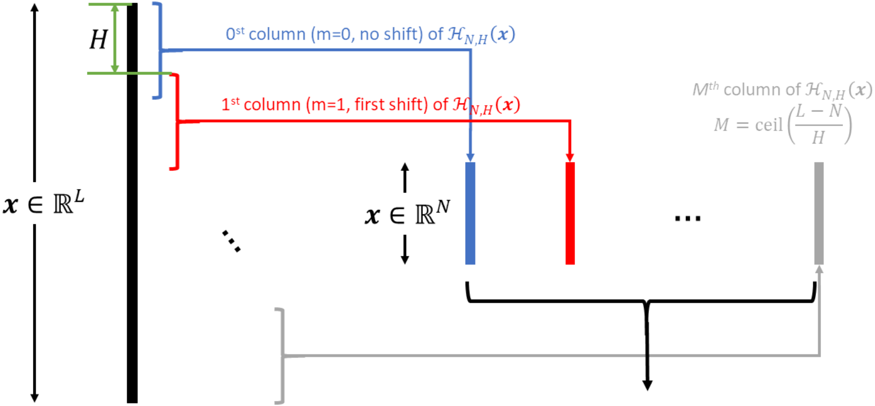
$$\mathbf{Y}(:,m) = \mathbf{\Psi}_N \mathbf{x}_{\vec{m}} \in \mathbb{C}^{K,1},$$

  and the whole matrix

$$\mathbf{Y} = \mathbf{\Psi}_N \underbrace{\left[\mathbf{x}_{\vec{0}}, \mathbf{x}_{\vec{1}}, \ldots \mathbf{x}_{\vec{M}}\right]}_{\mathbf{X}},$$

  where $\mathbf{X}$ is a Hankelization of $\mathbf{x}$ under segment parameter N and shifting parameter H.

Hankelization of $\mathbf{x}$



$\mathcal{H}_{N,H}(\boldsymbol{x}) \in \mathbb{R}^{N \times M+1}$: a “Hankel matrix” of $\boldsymbol{x}$, with shift parameter *H* and segment length *N*

Tensor representation of Hankelization

- Hankelization of a vector $\mathbf{x} \in \mathbb{R}^L$ to a matrix $\mathbf{X} \in \mathbb{R}^{N \times M+1}$ cannot be expressed using matrix operation, but it can be expressed using tensor operation.

- Let $\mathcal{H} \in \mathbb{R}^{N \times M+1 \times L}$ be a 3rd-order tensor, then the Hankelization of $\mathbf{x}$ can be expressed as a mode-3 product between $\mathcal{H}$ and $\mathbf{x}$

$$\mathbf{X} = \mathcal{H}(\mathbf{x}) = \mathcal{H} \times_3 \mathbf{x}.$$

- Example. $h : \mathbb{R}^4 \to \mathbb{R}^{2\times 2}$ as

$$\mathbf{x} = \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} \xrightarrow{h} \mathbf{X} = \begin{bmatrix} a & c \\ b & d \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}}_{\mathbf{H}_1} a + \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} b + \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} c + \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} d.$$

So $h(\mathbf{x}) = \mathbf{H}_1 x_1 + \mathbf{H}_2 x_2 + \cdots + \mathbf{H}_4 x_4$, where $\mathbf{H}_i$ are 0-1 sparse matrices that encode the location(s) of x_i. And $\mathbf{H}_i$ is the ith frontal slice of $\mathcal{H}$.

STFT: equivalent expressions

- STFT on $\mathbf{x}$ with a window $\mathbf{w}$, expressed using shifting notation is

$$\mathbf{Y} = \mathbf{\Psi}_N \mathbf{W} \underbrace{\left[\mathbf{x}_{\vec{0}}, \mathbf{x}_{\vec{1}}, \dots \mathbf{x}_{\vec{M}}\right]}_{\mathbf{X}}.$$

- Expressed using Hankelization operator

$$\mathbf{Y} = \mathbf{\Psi}_N \mathbf{W} \mathcal{H}_{N,H}(\mathbf{x}).$$

- Expressed using tensor representation of Hankelization

$$\mathbf{Y} = \mathbf{\Psi}_N \mathbf{W} \mathcal{H} \times_3 \mathbf{x}.$$

Inverse STFT (iSTFT)

- Recall that DFT: $\mathbf{y} = \mathbf{\Psi}_N \mathbf{x}$. By definition, $\mathbf{\Psi}_N$ is non-singular, so $\mathbf{x} = \mathbf{\Psi}_N^{-1} \mathbf{y}$.
- As STFT

$$\mathbf{Y}(:, m) = \mathbf{\Psi}_N \mathbf{W} \mathbf{x}_{\vec{m}},$$

hence

$$\mathbf{x}_{\vec{m}} = \mathbf{\Psi}_N^{-1} \mathbf{Y}(:, m),$$

and for all m then

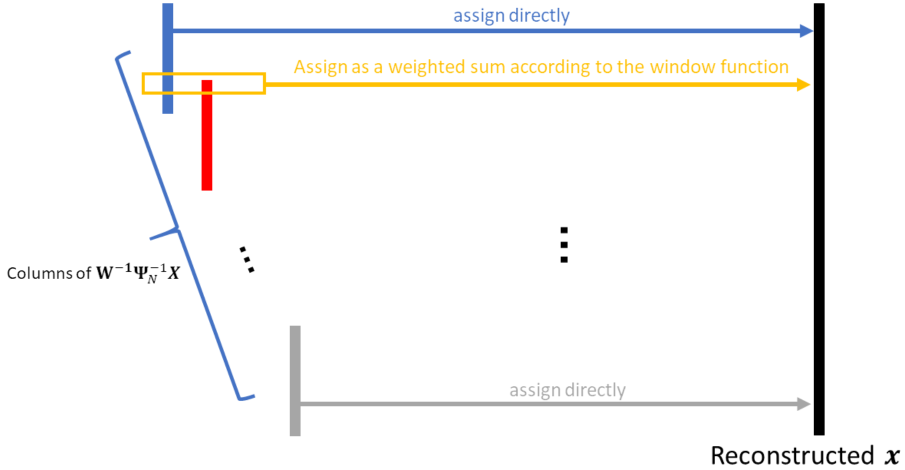
$$h_{N,H}(\mathbf{x}) = \mathbf{\Psi}_N^{-1} \mathbf{Y}.$$

- As all $\mathbf{x}_{\vec{m}}$ are portions of $\mathbf{x}$, we just need sum them back to get $\mathbf{x}$.

$$\mathbf{x} = \mathcal{H}_{N,H}(\mathbf{x})\theta = \mathbf{W}^{-1} \mathbf{\Psi}_N^{-1} \mathbf{X}^\top \theta,$$

where θ is just a weigh vector determined by the window function.

The inverse Hankelization transform of $\mathbf{W}^{-1}\mathbf{\Psi}_N^{-1}\mathbf{X}$



About implementation

- The product $\mathbf{\Psi}_N \mathbf{x}$ in DFT operation takes $\mathcal{O}(N^2)$ costs to compute. Using FFT (by making use of the structure of $\mathbf{\Psi}_N$), it goes down to $\mathcal{O}(N \log N)$.
- For window function $\mathbf{w}$, if we use the all-one window, the frequency profile will contains ripples: it is well-known that the Fourier Transform of a rectangular function with sharp change will contains all the frequency components. Hence, in practice, Gaussian window, or raised-cosine window is used.
- However, when using non-all-one window, the vector $\mathbf{x}$ is sheared/deformed.

Summary

- Review of DFT and STFT

End of document