Fractional Fourier Transform

From papers:

FOURIER NEURAL OPERATOR FOR PARAMETRIC PARTIAL DIFFERENTIAL EQUATIONS (2020)

Multiwavelet-based Operator Learning for Differential Equations (2021)

Optimal Filtering in Fractional Fourier Domains (1997)

Fractional Fourier transform as a signal processing tool: An overview of recent developments (2011)

Digital Computation of the Fractional Fourier Transform (1996)

A Convolution an Product Theorem for the Fractional Fourier Transform (1998)

Suwei Yang 2023.Nov.22

Outline

- Motivation
- Fractional Fourier Transform → dark
  imp
- Works

Motivation-from FNO

Definition 2 (Kernel integral operator $\mathcal{K}$) *Define the kernel integral operator mapping in (2) by*

$$(\mathcal{K}(a;\phi)v_t)(x) := \int_D \kappa(x,y,a(x),a(y);\phi)v_t(y)\mathrm{d}y, \qquad \forall x \in D \tag{3}$$

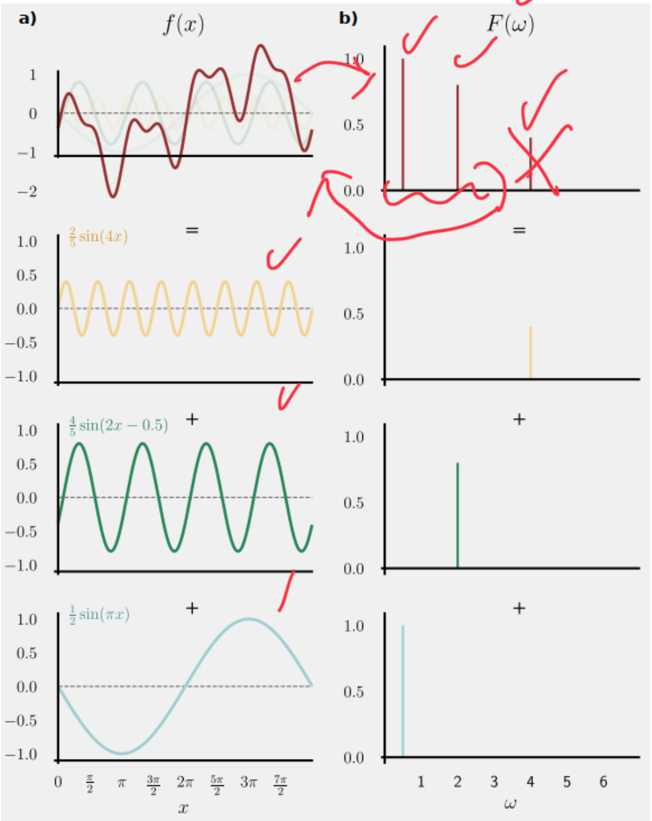
If we remove the dependence on the function a and impose $\kappa_\phi(x,y) = \kappa_\phi(x-y)$

Definition 3 (Fourier integral operator $\mathcal{K}$) *Define the Fourier integral operator*

$$(\mathcal{K}(\phi)v_t)(x) = \mathcal{F}^{-1}\Big(R_\phi \cdot (\mathcal{F}v_t)\Big)(x) \qquad \forall x \in D \tag{4}$$

where R_ϕ is the Fourier transform of a periodic function $\kappa : \bar{D} \to \mathbb{R}^{d_v \times d_v}$ parameterized by $\phi \in \Theta_{\mathcal{K}}$. An illustration is given in Figure 2 (b).

high freq



Motivation-from wavelet transform

$$Ta(x) = \int_D K(x,y)a(y)dy.$$

time
freq

Multi Resolution Analysis: We begin by defining the space of piecewise polynomial functions, for $k \in \mathbb{N}$ and $n \in \mathbb{Z}^+ \cup \{0\}$ as, $\mathbf{V}_n^k = \bigcup_{l=0}^{2^n-1}\{f|\deg(f) < k \text{ for } x \in (2^{-n}l, 2^{-n}(l+1)) \wedge 0, \text{ elsewhere}\}$. Clearly, $\dim(\mathbf{V}_n^k) = 2^n k$, and for subsequent n, each subspace is contained in another as shown by the following relation:

$$\mathbf{V}_0^k \subset \mathbf{V}_1^k \ldots \subset \mathbf{V}_{n-1}^k \subset \mathbf{V}_n^k \subset \ldots. \tag{2}$$

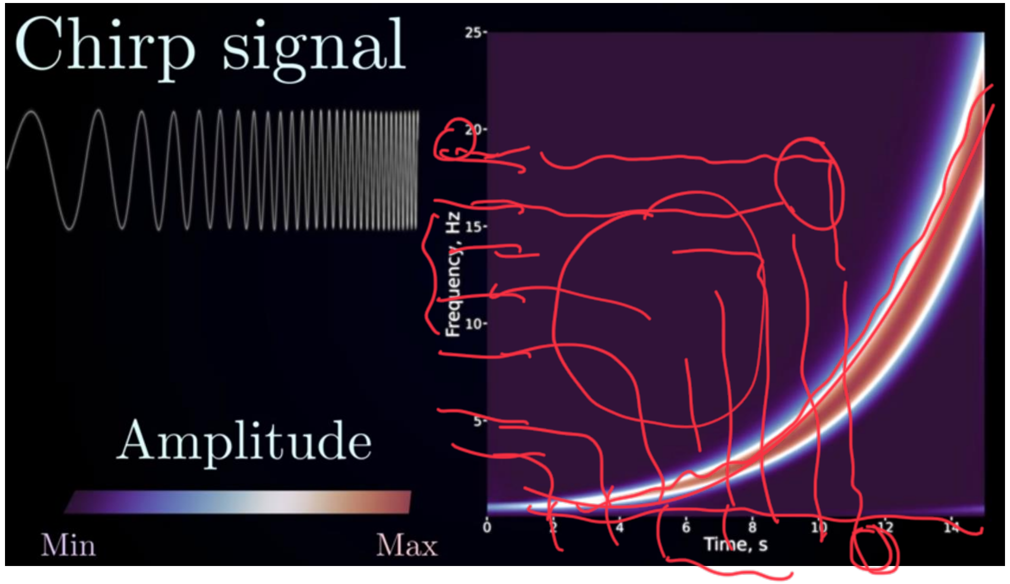
Similarly, we define the sequence of measures $\mu_0, \mu_1, \ldots$ such that $f \in \mathbf{V}_n^k$ is measurable w.r.t. μ_n and the norm of f is taken as $||f|| = \langle f, f\rangle_{\mu_n}^{1/2}$. Next, since $\mathbf{V}_{n-1}^k \subset \mathbf{V}_n^k$, we define the multiwavelet subspace as $\mathbf{W}_n^k$ for $n \in \mathbb{Z}^+ \cup \{0\}$, such that

$$\mathbf{V}_{n+1}^k = \mathbf{V}_n^k \bigoplus \mathbf{W}_n^k, \quad \mathbf{V}_n^k \perp \mathbf{W}_n^k. \tag{3}$$

Non-Standard Form: The multiwavelet representation of the operator kernel $K(x,y)$ can be obtained by an appropriate tensor product of the multiscale and multiwavelet basis. One issue, however, in this approach, is that the basis at various scales are *coupled* because of the tensor product. To untangle the basis at various scales, we use a trick as proposed in [13] called the non-standard wavelet representation. The extra mathematical price paid for the non-standard representation, actually serves as a ground for reducing the proposed model complexity (see Section 2.3), thus, providing data efficiency. For the operator under consideration T with integral kernel $K(x,y)$, let us denote T_n as the projection of T on V_n^k, which essentially is obtained by projecting the kernel K onto basis ϕ_{jl}^n w.r.t. measure μ_n. If P_n is the projection operator such that $P_n f = \sum_{j,l}\langle f, \phi_{jl}^n\rangle_{\mu_n}\phi_{jl}^n$, then $T_n = P_n T P_n$. Using telescopic sum, T_n is expanded as

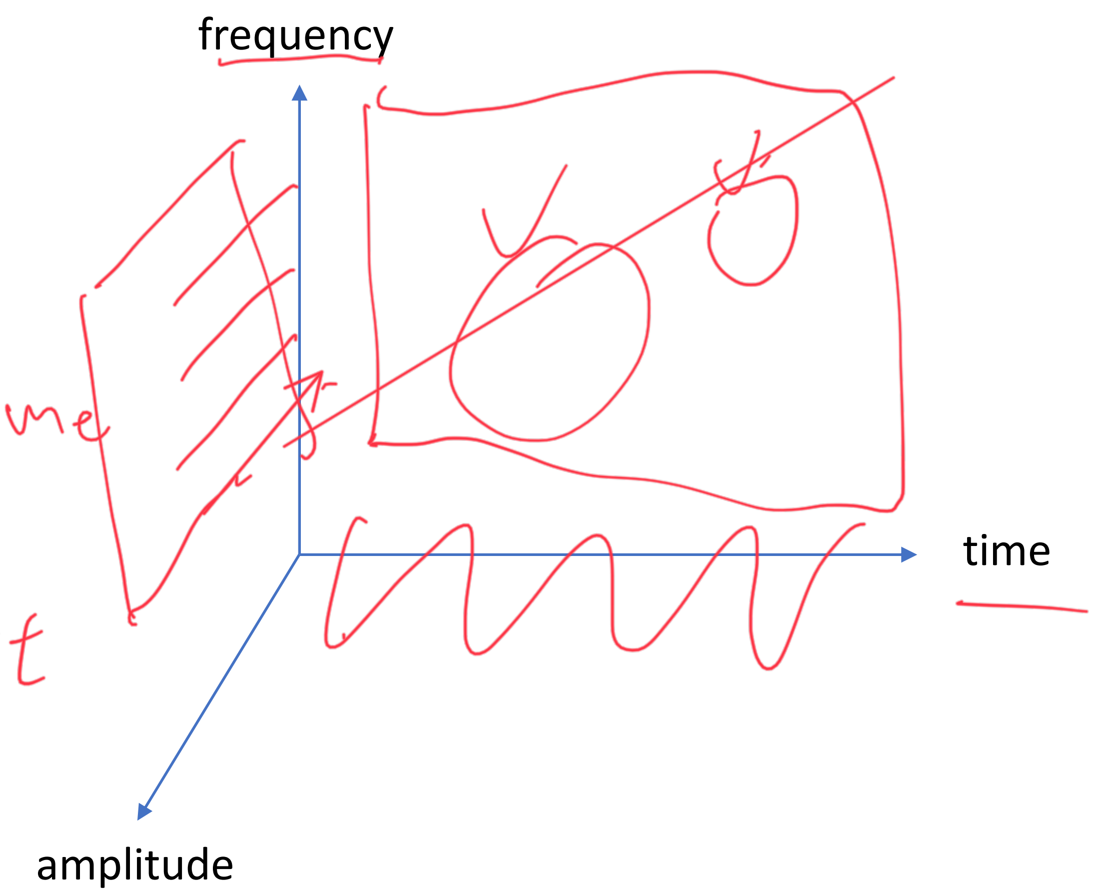
$$T_n = \sum_{i=L+1}^{n}(Q_iTQ_i + Q_iTP_{i-1} + P_{i-1}TQ_i) + P_LTP_L, \tag{10}$$

where, $Q_i = P_i - P_{i-1}$ and L is the coarsest scale under consideration ($L \geqslant 0$). From eq. (3), it is apparent that Q_i is the multiwavelet operator. Next, we denote $A_i = Q_iTQ_i, B_i = Q_iTP_{i-1}, C_i = P_{i-1}TQ_i$, and $\bar{T} = P_LTP_L$. In Figure 1, we show the non-standard multiwavelet transform for a given kernel $K(x,y)$. The

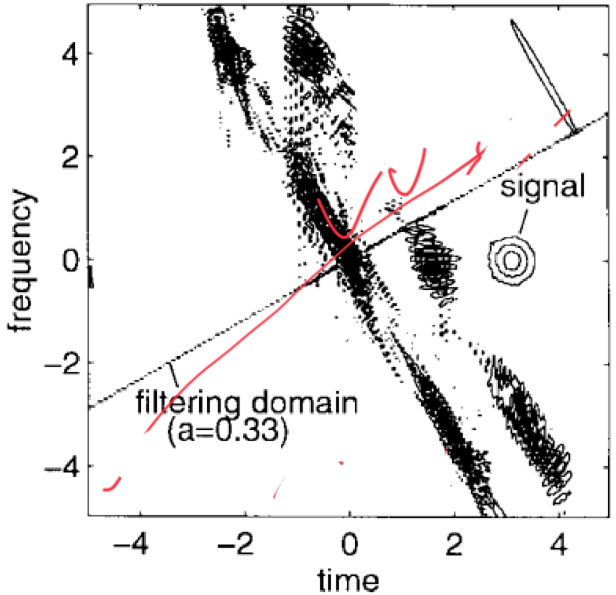


Motivation

- FNO:
  - Simple
  - Lack of time information fre
- MWTNO:
  - Provide time information fre time
  - Complicated and slow
- Fractional Fourier Transform frft
  - No complicated as wavelet
  - Keep time information



Fractional Fourier Transform



(b)

2. The fractional Fourier transform

The fractional Fourier transform (FRFT) is a linear operator defined as [10–13]

$$X_\alpha(u) = \mathcal{F}_\alpha(x(t)) = \int_{-\infty}^{+\infty} x(t) K_\alpha(t,u)\, dt \tag{1}$$

with $K_\alpha(t,u)$ representing the kernel function defined as

$$K_\alpha(t,u) = \begin{cases} \sqrt{\dfrac{1-j\cot\alpha}{2\pi}} \times e^{j(u^2/2)\cot\alpha} e^{j(t^2/2)\cot\alpha - jut\csc\alpha} & \text{if } \alpha \text{ is not multiple of } \pi \\ \delta(t-u) & \text{if } \alpha \text{ is a multiple of } 2\pi \\ \delta(t+u) & \text{if } \alpha+\pi \text{ is a multiple of } 2\pi \end{cases} \tag{2}$$

and $\delta(t)$ representing the Dirac function. Throughout the paper we use $\mathcal{F}_\alpha$ to denote the operator associated with the FRFT. It should be noted that we adopted notation for

In this approach, we assume $a \in [-1, 1]$. Manipulating (1), we can write

$$f_a(x) = \exp\left[-i\pi x^2 \tan(\phi/2)\right] g'(x), \tag{14}$$

$$g'(x) = A_\phi \int_{-\infty}^{\infty} \exp\left[i\pi\beta(x-x')^2\right] g(x')\, dx', \tag{15}$$

$$g(x) = \exp\left[-i\pi x^2 \tan(\phi/2)\right] f(x) \tag{16}$$

where $g(x)$ and $g'(x)$ represent intermediate results, and $\beta = \csc\phi$.

$$A_\phi \equiv \frac{\exp\left(-i\pi\,\mathrm{sgn}(\sin\phi)/4 + i\phi/2\right)}{|\sin\phi|^{1/2}} \tag{1}$$

where

$$\phi = \frac{a\pi}{2} \tag{2}$$

$\alpha = 1$

$\alpha = 0, 1$

$\exp(-i\pi x^2)$

Works

Assumption:

FNO: $\int k(x,y)v(y)dy \xRightarrow{impose} \int k(x-y)v(y)dy = (k*v)(x)$

Mine: $\int k(x,y)v(y)dy \xRightarrow{impose} (k*v)(x)$

ft x ft

See Fig. 1 for a realization of the convolution operation $\star$.

Now we state and prove our convolution theorem.

Theorem 1: Let $h(x) = (f \star g)(x)$ and $F_\alpha, G_\alpha, H_\alpha$ denote the FRFT of f, g and h, respectively. Then

$$H_\alpha(u) = F_\alpha(u) G_\alpha(u) e^{-ja(\alpha)u^2}. \quad (6)$$

$\alpha \times \alpha$

$\alpha = \frac{\pi}{2}$ ft

α

time

$\alpha \neq \frac{\pi}{2} \rightarrow \alpha$ domain

Works

A. Problem Statement

Our signal observation model can be written as

$$y(t) = \int_{-\infty}^{\infty} h(t,t')x(t')\,dt' + n(t) \qquad (1)$$

where $h(t,t')$ is the kernel of the degradation model, and $n(t)$ is the additive noise term. We will assume that as a prior

is the time-bandwidth product of the signals. In this paper, we restrict our estimate so that it corresponds to a multiplication with a filter function in the ath fractional Fourier domain. This estimate can be written in operator notation as

$$\hat{\boldsymbol{x}} = \mathcal{F}^{-a}(\boldsymbol{g} \cdot \mathcal{F}^{a}(\boldsymbol{y})) \qquad (6)$$

where $\mathcal{F}^a$ is the ath-order fractional Fourier transformation operator, and $\boldsymbol{g}$ is the multiplicative filter. We note that for $a = 1$, this estimate corresponds to filtering in the conventional Fourier domain. With this form of estimation operator, the

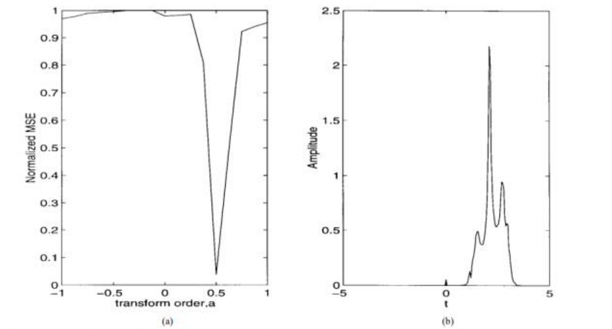


Fig. 10. [Example 4] (a) Normalized MSE for different values of transform order a. (b) Optimum filter function $g_o(t_{0.5})$ in the $a_{opt} = 0.5$th domain (absolute value).

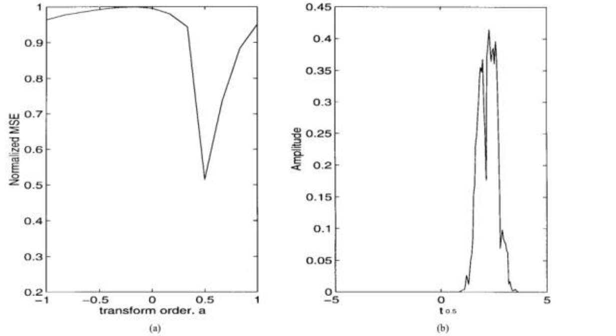


Fig. 13. [Example 5] (a) Normalized MSE for different values of transform order a. (b) Optimum filter function $g_o(t_{0.5})$ in the $a_{opt} = 0.5$th domain (absolute value).

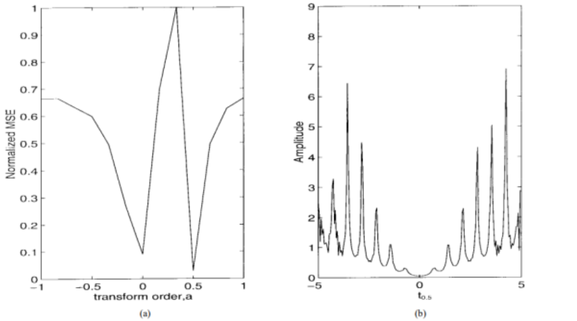
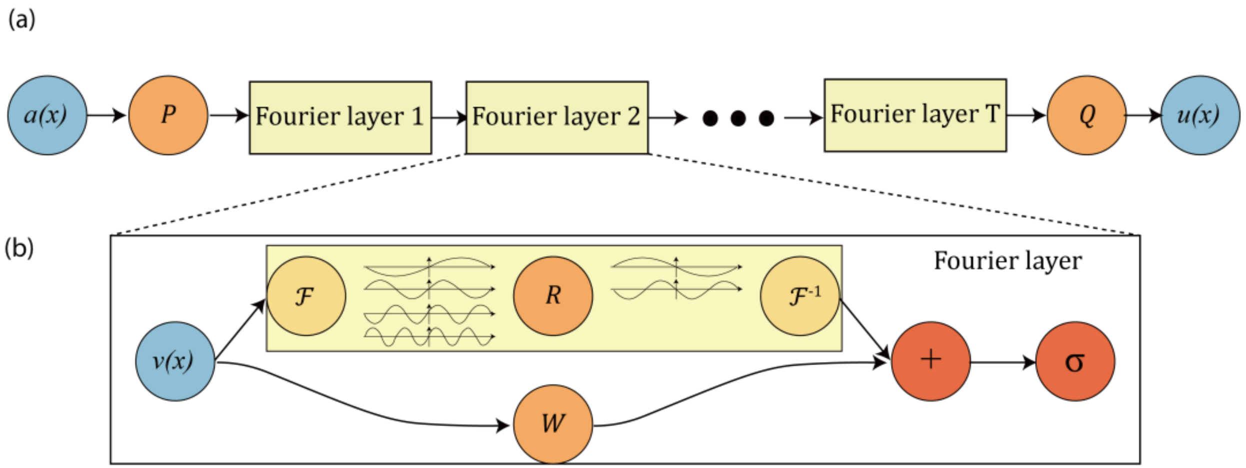


Fig. 2. [Example 1] (a) Normalized MSE for different values of transform order a. (b) Optimum filter function $g_o(t_{0.5})$ in the $a_{opt} = 0.5$th domain (absolute value).

alpha: hyperparameters

g: NN does it

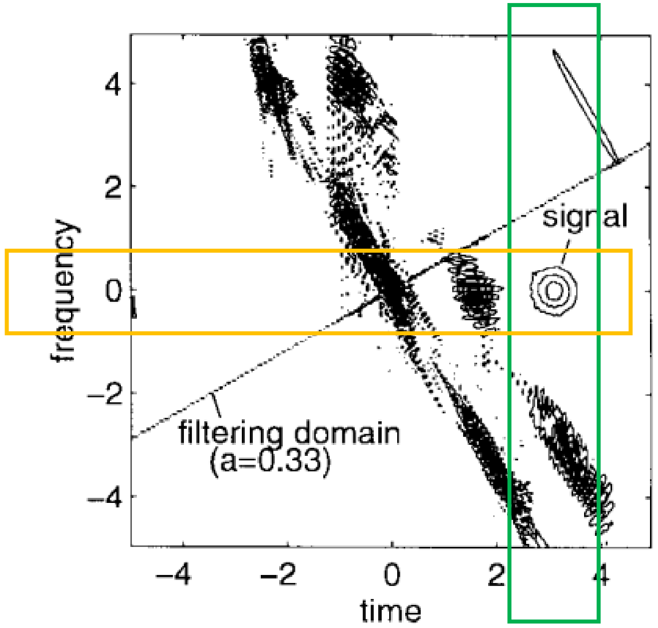
Why FNO works?



(a) The full architecture of neural operator: start from input a. 1. Lift to a higher dimension channel space by a neural network P. 2. Apply four layers of integral operators and activation functions. 3. Project back to the target dimension by a neural network Q. Output u. **(b) Fourier layers**: Start from input v. On top: apply the Fourier transform $\mathcal{F}$; a linear transform R on the lower Fourier modes and filters out the higher modes; then apply the inverse Fourier transform $\mathcal{F}^{-1}$. On the bottom: apply a local linear transform W.

Figure 2: **top:** The architecture of the neural operators; **bottom:** Fourier layer.

FNO layer



(b)

P layer
W layer

Experiments