Nonequispaced Fast Fourier Transform and Applications

Simone Parisotto
simone.parisotto@tin.it

University of Verona
Faculty of Mathematical Physical and Natural Science
-
Bachelor's Degree Course in Applied Mathematics

October 15th 2010

Index

Trigonometric Polynomials

- Introduction: J. Fourier (superposition principle)

Trigonometric Polynomials

- Introduction: J. Fourier (superposition principle)

Definition (Trigonometric Polynomial of order m and 2π-periodic)

$$P_m(x) = a_0 + \sum_{k=1}^{m} (a_k \cos kx + b_k \sin kx), \quad \text{with } a_k, b_k \in \mathbb{C}$$

Trigonometric Polynomials

- Introduction: J. Fourier (superposition principle)

Definition (Trigonometric Polynomial of order m and 2π-periodic)

$$P_m(x) = a_0 + \sum_{k=1}^{m} (a_k \cos kx + b_k \sin kx), \quad \text{with } a_k, b_k \in \mathbb{C}$$

An orthogonal basis for 2π-periodic functions

Let $B = \left\{ \cos nx, \sin nx, \ n \in \mathbb{N} \right\}$ with

- an inner product $\langle u, v \rangle = \int_0^{2\pi} u(x)\overline{v(x)}\,\mathrm{d}x;$
- a norm $||u|| = \sqrt{\langle u, u \rangle}.$

Trigonometric Polynomials

- Introduction: J. Fourier (superposition principle)

Definition (Trigonometric Polynomial of order m and 2π-periodic)

$$P_m(x) = a_0 + \sum_{k=1}^{m} (a_k \cos kx + b_k \sin kx), \quad \text{with } a_k, b_k \in \mathbb{C}$$

An orthogonal basis for 2π-periodic functions

Let $B = \left\{ \cos nx, \sin nx, \;\; n \in \mathbb{N} \right\}$ with

- an inner product $\langle u, v \rangle = \int_0^{2\pi} u(x)\overline{v(x)}\,\mathrm{d}x$;
- a norm $||u|| = \sqrt{\langle u, u \rangle}$.

Proof of orthogonality

- $\langle \cos mx, \cos nx \rangle = \begin{cases} \pi & \text{if } m \neq n, \\ 2\pi & \text{if } m = n = 0, \\ 0 & \text{otherwise;} \end{cases}$
- $\langle \sin mx, \sin nx \rangle = \begin{cases} \pi & \text{if } m \neq n, \\ 0 & \text{otherwise;} \end{cases}$
- $\langle \cos mx, \sin nx \rangle = 0.$

Fourier Series for 2π-periodic and l-periodic functions

Trigonometric Fourier Series ($T = 2\pi$)

$$f(x) \approx \frac{a_0}{2} + \sum_{k=1}^{\infty} (a_k \cos kx + b_k \sin kx), \;\; x \in [0, 2\pi)$$

- $a_0 = \frac{1}{\pi} \int_0^{2\pi} f(x)\,\mathrm{d}x, \;\; a_k = \frac{1}{\pi} \int_0^{2\pi} f(x) \cos kx \,\mathrm{d}x,$
- $b_k = \frac{1}{\pi} \int_0^{2\pi} f(x) \sin kx \,\mathrm{d}x.$

Fourier Series for 2π-periodic and l-periodic functions

Trigonometric Fourier Series ($T = 2\pi$)

$$f(x) \approx \frac{a_0}{2} + \sum_{k=1}^{\infty}(a_k \cos kx + b_k \sin kx), \quad x \in [0, 2\pi)$$

- $a_0 = \frac{1}{\pi}\int_0^{2\pi} f(x)\,\mathrm{d}x, \quad a_k = \frac{1}{\pi}\int_0^{2\pi} f(x)\cos kx\,\mathrm{d}x,$
- $b_k = \frac{1}{\pi}\int_0^{2\pi} f(x)\sin kx\,\mathrm{d}x.$

Exponential Fourier Series ($T = 2\pi$)

$$f(x) \approx \sum_{k=-\infty}^{+\infty} c_k e^{ikx}, \quad x \in [0, 2\pi)$$

- $c_k = \frac{1}{2\pi}\int_0^{2\pi} f(x) e^{-ikx}\,\mathrm{d}x.$

Fourier Series for 2π-periodic and l-periodic functions

Trigonometric Fourier Series ($T = 2\pi$)

$$f(x) \approx \frac{a_0}{2} + \sum_{k=1}^{\infty} (a_k \cos kx + b_k \sin kx), \quad x \in [0, 2\pi)$$

- $a_0 = \frac{1}{\pi} \int_0^{2\pi} f(x)\,\mathrm{d}x, \quad a_k = \frac{1}{\pi} \int_0^{2\pi} f(x) \cos kx\,\mathrm{d}x,$
- $b_k = \frac{1}{\pi} \int_0^{2\pi} f(x) \sin kx\,\mathrm{d}x.$

Exponential Fourier Series ($T = 2\pi$)

$$f(x) \approx \sum_{k=-\infty}^{+\infty} c_k e^{ikx}, \quad x \in [0, 2\pi)$$

- $c_k = \frac{1}{2\pi} \int_0^{2\pi} f(x) e^{-ikx}\,\mathrm{d}x.$

Some useful identities

- $e^{i\theta} = \cos\theta + i \sin\theta;$
- $\cos x = (e^{ix} + e^{-ix})/2;$
- $\sin x = (e^{ix} - e^{-ix})/2;$
- $a_k = c_k + c_{-k};$
- $b_k = i(c_k - c_{-k});$
- $c_0 = a_0/2;$
- $c_k = (a_k - ib_k)/2;$
- $c_{-k} = (a_k + ib_k)/2.$

Fourier Series for 2π-periodic and l-periodic functions

Trigonometric Fourier Series ($T = 2\pi$)

$$f(x) \approx \frac{a_0}{2} + \sum_{k=1}^{\infty}(a_k \cos kx + b_k \sin kx), \;\; x \in [0, 2\pi)$$

- $a_0 = \frac{1}{\pi}\int_0^{2\pi} f(x)\,\mathrm{d}x, \;\; a_k = \frac{1}{\pi}\int_0^{2\pi} f(x)\cos kx\,\mathrm{d}x,$
- $b_k = \frac{1}{\pi}\int_0^{2\pi} f(x)\sin kx\,\mathrm{d}x.$

Exponential Fourier Series ($T = 2\pi$)

$$f(x) \approx \sum_{k=-\infty}^{+\infty} c_k e^{ikx}, \;\; x \in [0, 2\pi)$$

- $c_k = \frac{1}{2\pi}\int_0^{2\pi} f(x)e^{-ikx}\,\mathrm{d}x.$

Some useful identities

- $e^{i\theta} = \cos\theta + i\sin\theta;$
- $\cos x = (e^{ix} + e^{-ix})/2;$
- $\sin x = (e^{ix} - e^{-ix})/2;$
- $a_k = c_k + c_{-k};$
- $b_k = i(c_k - c_{-k});$
- $c_0 = a_0/2;$
- $c_k = (a_k - ib_k)/2;$
- $c_{-k} = (a_k + ib_k)/2.$

Trigonometric and Exponential Fourier Series ($T = l = b - a$):

- Replacing with $2\pi(x-a)/(b-a), \;\; x \in [a, b]$;
- Changing the extremes of integration with a and b;
- Replacing with $2/(b-a)$ (Trigonometric FS), with $1/(b-a)$ (Exponential FS).

Convergence Results

Convergence Results

Theorem (Some type of Convergences)

Let $f(x)$ a 2π-periodic and C^1-piecewise function on the interval $[0, 2\pi)$. Then, the Fourier Series of $f(x)$ converges

- *uniformly to $f(x)$ on every compact set which does not contain any discontnuity point;*
- *to the average of right and left limits of f to x^* if x^* is a discontinuity point;*

Let $f(x)$ a 2π-periodic function on the interval $[0, 2\pi)$, with $\int_0^{2\pi} |f(x)|^2 \mathrm{d}x < +\infty$, then

$$\int_0^{2\pi} |f(x) - f_N(x)|^2 \mathrm{d}x \leq \int_0^{2\pi} |f(x) - P_N(x)|^2$$

for any trigonometric polynomial P_N of order N (2nd order mean), and

$$\lim_{N \to +\infty} \int_0^{2\pi} |f(x) - f_N(x)|^2 \mathrm{d}x = 0.$$

Convergence Results

Theorem (Some type of Convergences)

Let $f(x)$ a 2π-periodic and C^1-piecewise function on the interval $[0, 2\pi)$. Then, the Fourier Series of $f(x)$ converges

- *uniformly to $f(x)$ on every compact set which does not contain any discontnuity point;*
- *to the average of right and left limits of f to x^* if x^* is a discontinuity point;*

Let $f(x)$ a 2π-periodic function on the interval $[0, 2\pi)$, with $\int_0^{2\pi} |f(x)|^2 \mathrm{d}x < +\infty$, then

$$\int_0^{2\pi} |f(x) - f_N(x)|^2 \mathrm{d}x \leq \int_0^{2\pi} |f(x) - P_N(x)|^2$$

for any trigonometric polynomial P_N of order N (2nd order mean), and

$$\lim_{N \to +\infty} \int_0^{2\pi} |f(x) - f_N(x)|^2 \mathrm{d}x = 0.$$

Bessel–Parseval Identity

In the same previous hypotheses: $\frac{1}{\pi} \int_0^{2\pi} |f(x)|^2 \mathrm{d}x = \frac{a_0}{2} + \sum_{n=1}^{\infty} \left(|a_k|^2 + |b_k|^2 \right) = 2 \sum_{k \in \mathbb{Z}} |c_k|^2.$

DFT: Discrete Fourier Trasform

Standard Problem

Let $f(x) : [a, b) \to \mathbb{C}$ a periodic function. We suppose that $f(x)$ can be written as:

$$f(x) = \sum_{k=-\infty}^{+\infty} c_k e^{i2\pi k(\frac{x-a}{b-a})}, \text{ subject to previous convergence results.}$$

DFT: Discrete Fourier Trasform

Standard Problem

Let $f(x) : [a, b) \to \mathbb{C}$ a periodic function. We suppose that $f(x)$ can be written as:

$$f(x) = \sum_{k=-\infty}^{+\infty} c_k e^{i2\pi k(\frac{x-a}{b-a})}, \text{ subject to previous convergence results.}$$

- For MATLAB: $\phi_k(x) = \dfrac{e^{i2\pi(k-1-N/2)(x-a)/(b-a)}}{\sqrt{b-a}}$, $\forall k \in \mathbb{Z}$, orthonormal respect to
  - $\langle \phi_j, \phi_k \rangle = \int_a^b \phi_j(x)\overline{\phi_k(x)}\mathrm{d}x;$
  - $\langle \phi_j, \phi_k \rangle_N = \dfrac{b-a}{N} \sum_{n=1}^{N} \phi_j(x_n)\overline{\phi_k(x_n)}$

DFT: Discrete Fourier Trasform

Standard Problem

Let $f(x) : [a, b) \to \mathbb{C}$ a periodic function. We suppose that $f(x)$ can be written as:

$$f(x) = \sum_{k=-\infty}^{+\infty} c_k e^{i2\pi k\left(\frac{x-a}{b-a}\right)},$$ subject to previous convergence results.

- For MATLAB: $\phi_k(x) = \dfrac{e^{i2\pi(k-1-N/2)(x-a)/(b-a)}}{\sqrt{b-a}}$, $\forall k \in \mathbb{Z}$, orthonormal respect to
  - $\langle \phi_j, \phi_k \rangle = \int_a^b \phi_j(x)\overline{\phi_k(x)}\mathrm{d}x;$
  - $\langle \phi_j, \phi_k \rangle_N = \dfrac{b-a}{N} \sum_{n=1}^{N} \phi_j(x_n)\overline{\phi_k(x_n)}$ with $-N+1 \leq j-k \leq N-1$.

DFT: Discrete Fourier Trasform

Standard Problem

Let $f(x) : [a, b) \to \mathbb{C}$ a periodic function. We suppose that $f(x)$ can be written as:

$$f(x) = \sum_{k=-\infty}^{+\infty} c_k e^{i2\pi k(\frac{x-a}{b-a})}, \text{ subject to previous convergence results.}$$

- For MATLAB: $\phi_k(x) = \dfrac{e^{i2\pi(k-1-N/2)(x-a)/(b-a)}}{\sqrt{b-a}}, \ \forall k \in \mathbb{Z}$, orthonormal respect to
  - $\langle \phi_j, \phi_k \rangle = \int_a^b \phi_j(x)\overline{\phi_k(x)}\mathrm{d}x;$
  - $\langle \phi_j, \phi_k \rangle_N = \dfrac{b-a}{N} \sum_{n=1}^{N} \phi_j(x_n)\overline{\phi_k(x_n)}$ with $-N+1 \leq j-k \leq N-1$.

How to approximate c_k and $f(x_k)$, with $k = 1, \ldots, N$

$$c_k = \int_a^b f(x)\overline{\phi_k(x)}\mathrm{d}x$$

DFT: Discrete Fourier Trasform

Standard Problem

Let $f(x) : [a, b) \to \mathbb{C}$ a periodic function. We suppose that $f(x)$ can be written as:

$$f(x) = \sum_{k=-\infty}^{+\infty} c_k e^{i2\pi k\left(\frac{x-a}{b-a}\right)}, \text{ subject to previous convergence results.}$$

- For MATLAB: $\phi_k(x) = \dfrac{e^{i2\pi(k-1-N/2)(x-a)/(b-a)}}{\sqrt{b-a}}$, $\forall k \in \mathbb{Z}$, orthonormal respect to
  - $\langle \phi_j, \phi_k \rangle = \int_a^b \phi_j(x)\overline{\phi_k(x)}\mathrm{d}x$;
  - $\langle \phi_j, \phi_k \rangle_N = \dfrac{b-a}{N} \sum_{n=1}^{N} \phi_j(x_n)\overline{\phi_k(x_n)}$ with $-N+1 \leq j-k \leq N-1$.

How to approximate c_k and $f(x_k)$, with $k = 1, \ldots, N$

$$c_k = \int_a^b f(x)\overline{\phi_k(x)}\mathrm{d}x \approx \frac{\sqrt{b-a}}{N} \sum_{n=1}^{N} \left(f(x_n)e^{iN\pi y_n}\right) e^{-i2\pi(k-1)y_n} = \hat{f}_k, \text{ (DFT).}$$

DFT: Discrete Fourier Trasform

Standard Problem

Let $f(x) : [a, b) \to \mathbb{C}$ a periodic function. We suppose that $f(x)$ can be written as:

$$f(x) = \sum_{k=-\infty}^{+\infty} c_k e^{i2\pi k(\frac{x-a}{b-a})}, \text{ subject to previous convergence results.}$$

- For MATLAB: $\phi_k(x) = \dfrac{e^{i2\pi(k-1-N/2)(x-a)/(b-a)}}{\sqrt{b-a}}$, $\forall k \in \mathbb{Z}$, orthonormal respect to
  - $\langle \phi_j, \phi_k \rangle = \int_a^b \phi_j(x)\overline{\phi_k(x)}\mathrm{d}x;$
  - $\langle \phi_j, \phi_k \rangle_N = \dfrac{b-a}{N} \sum_{n=1}^{N} \phi_j(x_n)\overline{\phi_k(x_n)}$ with $-N+1 \leq j-k \leq N-1$.

How to approximate c_k and $f(x_k)$, with $k = 1, \dots, N$

$$c_k = \int_a^b f(x)\overline{\phi_k(x)}\mathrm{d}x \approx \frac{\sqrt{b-a}}{N} \sum_{n=1}^{N} \left(f(x_n) e^{iN\pi y_n} \right) e^{-i2\pi(k-1)y_n} = \hat{f}_k, \text{ (DFT)}.$$

$$f(x_k) \approx \hat{\hat{f}}_k = \sum_{n=1}^{N} \hat{f}_n \phi_n(x_k) = \frac{N}{\sqrt{b-a}} \frac{1}{N} \left(\sum_{n=1}^{N} \hat{f}_n e^{i2\pi(n-1)y_k} \right) e^{-iN\pi y_k} \text{ (IDFT)}.$$

Error estimate

Truncation error

$$\int_a^b \left| f(x) - \sum_{j=1}^{N} c_j \phi_j(x) \right|^2 \mathrm{d}x = \sum_{k \in J} |c_k|^2 \text{ with } J = \mathbb{Z} \backslash \{1, \dots, N\}.$$

Error estimate

Truncation error

$$\int_a^b \left| f(x) - \sum_{j=1}^{N} c_j \phi_j(x) \right|^2 \mathrm{d}x = \sum_{k \in J} |c_k|^2 \text{ with } J = \mathbb{Z} \setminus \{1, \dots, N\}.$$

Estimate for c_k (integrating by parts) - Spectral convergence

$f(x) \in C^1 \Rightarrow c_k = \mathcal{O}(k^{-1})$

$f(x) \in C^2 \Rightarrow c_k = \mathcal{O}(k^{-2})$

$f(x) \in C^\infty$ decays faster than any negative power of k

Error estimate

Truncation error

$$\int_a^b \left| f(x) - \sum_{j=1}^{N} c_j \phi_j(x) \right|^2 \mathrm{d}x = \sum_{k \in J} |c_k|^2 \text{ with } J = \mathbb{Z} \backslash \{1, \ldots, N\}.$$

Estimate for c_k (integrating by parts) - Spectral convergence

$f(x) \in C^1 \Rightarrow c_k = \mathcal{O}(k^{-1})$

$f(x) \in C^2 \Rightarrow c_k = \mathcal{O}(k^{-2})$

$f(x) \in C^\infty$ decays faster than any negative power of k

Upper-bound (Boyd)

- $|f(x) - f_N(x)| \leq \sum_{k \in J} |c_k|$, where $f_N = \sum_{k=1}^{N} c_k \phi_k(x)$;
- $|f(x) - F_N(x)| \leq 2 \sum_{k \in J} |c_k|$, where $F_N = \sum_{k=1}^{N} \hat{f}_k \phi_k(x)$.

Trapezoidal quadrature formula

$$\int_a^b g(x)\mathrm{d}x \approx \frac{b-a}{2N}\left(g(x_1) + 2\sum_{k=2}^{N} g(x_k) + g(x_{N+1})\right) = \frac{b-a}{N}\sum_{k=1}^{N} g(x_k).$$

Trapezoidal quadrature formula

$$\int_a^b g(x)\mathrm{d}x \approx \frac{b-a}{2N}\left(g(x_1) + 2\sum_{k=2}^{N} g(x_k) + g(x_{N+1})\right) = \frac{b-a}{N}\sum_{k=1}^{N} g(x_k).$$

DFT is *exact* on N points for $\{\phi_k\}_{k=-N+1}^{N-1}$

- $\phi_k(x)$ is orthonormal respect to $< \cdot, \cdot >_N$;
- $F_N(x_k)$ is interpolant.

Trapezoidal quadrature formula

$$\int_a^b g(x)\mathrm{d}x \approx \frac{b-a}{2N}\left(g(x_1) + 2\sum_{k=2}^{N} g(x_k) + g(x_{N+1})\right) = \frac{b-a}{N}\sum_{k=1}^{N} g(x_k).$$

DFT is *exact* on N points for $\{\phi_k\}_{k=-N+1}^{N-1}$

- $\phi_k(x)$ is orthonormal respect to $< \cdot, \cdot >_N$;
- $F_N(x_k)$ is interpolant.

Proof of the interpolant property

$$F_N(x_k) = \sum_{n=1}^{N} \hat{f}_n \phi_n(x_k) =$$

$$= \sum_{n=1}^{N}\left(\left(\frac{\sqrt{b-a}}{N}\sum_{m=1}^{N} f(x_m)e^{iN\pi y_m}\right)e^{-i2\pi(n-1)y_m}\right)\frac{e^{i2\pi(n-1-N/2)(x_k-a)/(b-a)}}{\sqrt{b-a}} =$$

$$= \frac{1}{N}\sum_{m=1}^{N} f(x_m)e^{iN\pi(m-1)/N}e^{-iN\pi(k-1)/N}\sum_{n=1}^{N} e^{-i2\pi(n-1)(m-1)/N}e^{i2\pi(n-1)(k-1)/N} =$$

$$= \frac{1}{N}\sum_{m=1}^{N} f(x_m)e^{i(m-k)\pi}\sum_{n=1}^{N} e^{i2\pi(n-1)(k-m)/N} = \frac{1}{N}f(x_k)N = f(x_k).$$

Equispaced Matrix Fourier Transform

Given N equispaced sampling nodes and M equispaced evaluation points, how to implement DFT and IDFT in Matlab? MFT or FFT.

Equispaced Matrix Fourier Transform

Given N equispaced sampling nodes and M equispaced evaluation points, how to implement DFT and IDFT in Matlab? MFT or FFT.

1. Matrix Fourier Transform (MFT) and Inverse (IMFT) with $M = N$

Let $(F)_{jk} = e^{-i2\pi(j-1)y_{\boldsymbol{k}}}$ and $y_k = (n-1)/N,\ n = 1, \dots, N+1$:

- $\dfrac{\sqrt{b-a}}{N} \cdot F[f(x_1)e^{iN\pi y_{\boldsymbol{1}}}, \dots, f(x_N)e^{iN\pi y_{\boldsymbol{N}}}]^T$ is the MFT;
- $\dfrac{N}{\sqrt{b-a}}\left(\dfrac{F^H[\hat{f}_1, \dots, \hat{f}_N]}{N}\right) \circ [e^{-i\pi N y_{\boldsymbol{1}}}, \dots, e^{-i\pi N y_{\boldsymbol{N}}}]$ is the IMFT.

Equispaced Matrix Fourier Transform

Given N equispaced sampling nodes and M equispaced evaluation points, how to implement DFT and IDFT in Matlab? MFT or FFT.

1. Matrix Fourier Transform (MFT) and Inverse (IMFT) with $M = N$

Let $(F)_{jk} = e^{-i2\pi(j-1)y_k}$ and $y_k = (n-1)/N,\ n = 1, \dots, N+1$:

- $\dfrac{\sqrt{b-a}}{N} \cdot F[f(x_1)e^{iN\pi y_1}, \dots, f(x_N)e^{iN\pi y_N}]^T$ is the MFT;
- $\dfrac{N}{\sqrt{b-a}} \left(\dfrac{F^H[\hat{f}_1, \dots, \hat{f}_N]}{N} \right) \circ [e^{-i\pi N y_1}, \dots, e^{-i\pi N y_N}]$ is the IMFT.

1. Matrix Fourier Transform (MFT) and Inverse (IMFT) with $M \neq N$

- $M > N$: $\hat{\hat{f}}_k = \sum_{n=1}^{N} \hat{f}_n \phi_n(x_k), \quad$ with $k = 1, \dots, M$;
- $M < N$: there is no difference with the case discussed above.

Equispaced Fast Fourier Transform

(1965) FFT: Cooley (IBM), Tuckey (Princeton)

Equispaced Fast Fourier Transform

(1965) FFT: Cooley (IBM), Tuckey (Princeton)

Some key concepts:

- $N = N_1 + N_2$, ($N_1 = N_2$ with N power of 2);

Equispaced Fast Fourier Transform

(1965) FFT: Cooley (IBM), Tuckey (Princeton)

Some key concepts:

- $N = N_1 + N_2$, ($N_1 = N_2$ with N power of 2);
- $$\hat{f}_k = \underbrace{\sum_{m=0}^{N/2-1} f(x_{2m}) e^{-i2\pi \frac{m}{N/2} k}}_{\mathrm{E_k: DFT\ of\ even-indexed\ part\ of\ y_m}} + e^{-i2\pi \frac{1}{N} k} \underbrace{\sum_{m=0}^{N/2-1} f(x_{2m+1}) e^{-i2\pi \frac{m}{N/2} k}}_{\mathrm{O_k: DFT\ of\ odd-indexed\ part\ of\ y_m}}$$

Equispaced Fast Fourier Transform

(1965) FFT: Cooley (IBM), Tuckey (Princeton)

Some key concepts:

- $N = N_1 + N_2$, ($N_1 = N_2$ with N power of 2);
- $$\hat{f}_k = \underbrace{\sum_{m=0}^{N/2-1} f(x_{2m}) e^{-i2\pi \frac{m}{N/2} k}}_{\mathrm{E_k:\ DFT\ of\ even-indexed\ part\ of\ y_m}} + e^{-i2\pi \frac{1}{N} k} \underbrace{\sum_{m=0}^{N/2-1} f(x_{2m+1}) e^{-i2\pi \frac{m}{N/2} k}}_{\mathrm{O_k:\ DFT\ of\ odd-indexed\ part\ of\ y_m}}$$
- they are two DFTs of length $N/2$, so for the periodicity properties of DFT we have

$$E_{k+N/2} = E_k,$$
$$O_{k+N/2} = O_k;$$

Equispaced Fast Fourier Transform

(1965) FFT: Cooley (IBM), Tuckey (Princeton)

Some key concepts:

- $N = N_1 + N_2$, ($N_1 = N_2$ with N power of 2);
- $$\hat{f}_k = \underbrace{\sum_{m=0}^{N/2-1} f(x_{2m}) e^{-i2\pi \frac{m}{N/2} k}}_{\mathrm{E_k: DFT\ of\ even-indexed\ part\ of\ y_m}} + e^{-i2\pi \frac{1}{N} k} \underbrace{\sum_{m=0}^{N/2-1} f(x_{2m+1}) e^{-i2\pi \frac{m}{N/2} k}}_{\mathrm{O_k: DFT\ of\ odd-indexed\ part\ of\ y_m}}$$
- they are two DFTs of length $N/2$, so for the periodicity properties of DFT we have
$$E_{k+N/2} = E_k,$$
$$O_{k+N/2} = O_k;$$
- twiddle factor: $e^{-2\pi i \frac{(k+N/2)}{N}} = e^{-\pi i} e^{-2\pi i \frac{k}{N}} = -e^{-2\pi i \frac{k}{N}}$;
- $$\hat{f}_k = \begin{cases} E_k + e^{-\frac{2\pi i}{N} k} O_k & \text{if } k < N/2 \\ E_{k-N/2} - e^{-\frac{2\pi i}{N}(k-N/2)} O_{k-N/2} & \text{if } k \geq N/2. \end{cases}$$

Fastest Fourier Transform in the West: a FFT library for MATLAB

- written in C (Frigo - Johnson);
- `fft`, `ifft`, `fftshift`, `ifftshift`;
- distributed "wisdom" (size, type of transformation, etc...).

Fastest Fourier Transform in the West: a FFT library for MATLAB

- written in C (Frigo - Johnson);
- `fft`, `ifft`, `fftshift`, `ifftshift`;
- distributed "wisdom" (size, type of transformation, etc...).

2. Fast Fourier Transform (FFT) and Inverse (IFFT) with $M = N$

- $\frac{\sqrt{b-a}}{N} \cdot \texttt{fftshift}\left(\texttt{fft}\left([f(x_1), \ldots, f(x_n)]^T\right)\right)$ is the FFT;
- $\frac{N}{\sqrt{b-a}} \cdot \texttt{ifft}\left(\texttt{fftshift}\left([\hat{f}_1, \ldots, \hat{f}_N]\right)\right)$ is the IFFT.

Fastest Fourier Transform in the West: a FFT library for MATLAB

- written in C (Frigo - Johnson);
- `fft`, `ifft`, `fftshift`, `ifftshift`;
- distributed "wisdom" (size, type of transformation, etc...).

2. Fast Fourier Transform (FFT) and Inverse (IFFT) with $M = N$

- $\frac{\sqrt{b-a}}{N} \cdot \mathtt{fftshift}\Big(\mathtt{fft}\Big([f(x_1), \ldots, f(x_n)]^T\Big)\Big)$ is the FFT;
- $\frac{N}{\sqrt{b-a}} \cdot \mathtt{ifft}\Big(\mathtt{fftshift}\Big([\hat{f}_1, \ldots, \hat{f}_N]\Big)\Big)$ is the IFFT.

2. Fast Fourier Transform (FFT) and Inverse (IFFT) with $M \neq N$

- $M > N$: $\hat{f}^* = [\ \underbrace{0, \ldots, 0,}_{\text{(M−N)/2 items}}\ \ \hat{f},\ \ \underbrace{0, \ldots, 0}_{\text{(M−N)/2 items}}\ ]$;
- $M < N$: $\hat{f}^* = [\ \underbrace{\hat{f}_1, \ldots, \hat{f}_{\frac{M-N}{2}},}_{\text{items to be deleted}}\ \ \hat{f}_{\frac{M-N}{2}+1}, \ldots, \hat{f}_{\frac{M+N}{2}},\ \ \underbrace{\hat{f}_{\frac{M+N}{2}+1}, \ldots, \hat{f}_N}_{\text{items to be deleted}}\ ]$. (!)

Computational cost and comparison between MFT and FFT

Computational cost and comparison between MFT and FFT

Computational cost ($M = N$ equispaced nodes)

From space domain to frequency domain or backward the computational cost is

- MFT: $\mathcal{O}(N^2)$;
- FFT: $\mathcal{O}(N \log_2 N)$.

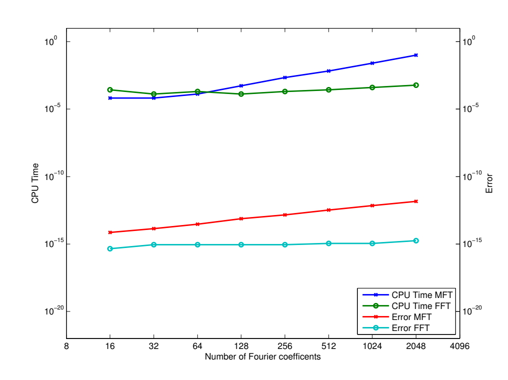
Computational cost and comparison between MFT and FFT

Computational cost ($M = N$ equispaced nodes)

From space domain to frequency domain or backward the computational cost is

- MFT: $\mathcal{O}(N^2)$;
- FFT: $\mathcal{O}(N \log_2 N)$.

Comparison MFT–FFT, test `f(x) = sin(2*pi*(x-a)/(b-a)) + 2*cos(4*2*pi*(x-a)/(b-a))`



Nonequispaced DFT

What if we have M nonequispaced evaluation nodes?

Nonequispaced DFT

What if we have M nonequispaced evaluation nodes? NDFT or NFFT!

Nonequispaced DFT

What if we have M nonequispaced evaluation nodes? NDFT or NFFT!

Nonequispaced Discrete Fourier Transform (NDFT) and Inverse (INDFT) by Kunis

- $\hat{f}(x) = \sum_{k=-N/2}^{N/2-1} f(x) e^{i2\pi kx}$ (NDFT);
- $\hat{\hat{f}}_k = \sum_{k=-N/2}^{N/2-1} \hat{f}_k e^{-i2\pi ky}$ (INDFT).

with $x, y \in [-\frac{1}{2}, \frac{1}{2})$. This works for $M = N$ and $M \neq N$.

Nonequispaced DFT

What if we have M nonequispaced evaluation nodes? NDFT or NFFT!

Nonequispaced Discrete Fourier Transform (NDFT) and Inverse (INDFT) by Kunis

- $\hat{f}(x) = \displaystyle\sum_{k=-N/2}^{N/2-1} f(x)e^{i2\pi kx}$ (NDFT);
- $\hat{\hat{f}}_k = \displaystyle\sum_{k=-N/2}^{N/2-1} \hat{f}_k e^{-i2\pi ky}$ (INDFT).

with $x, y \in [-\frac{1}{2}, \frac{1}{2})$. This works for $M = N$ and $M \neq N$.

Differences between Kunis notations and the ours: how to fix?

- In Kunis: $x \in [-\frac{1}{2}, \frac{1}{2})$ but we consider $y \in [0, 1)$.
- DFT, NDFT (and *Inverses*) presents a “minus” factor in the exponential.

Nonequispaced DFT

What if we have M nonequispaced evaluation nodes? NDFT or NFFT!

Nonequispaced Discrete Fourier Transform (NDFT) and Inverse (INDFT) by Kunis

- $\hat{f}(x) = \sum_{k=-N/2}^{N/2-1} f(x) e^{i2\pi kx}$ (NDFT);
- $\hat{\hat{f}}_k = \sum_{k=-N/2}^{N/2-1} \hat{f}_k e^{-i2\pi ky}$ (INDFT).

with $x, y \in [-\frac{1}{2}, \frac{1}{2})$. This works for $M = N$ and $M \neq N$.

Differences between Kunis notations and the ours: how to fix?

- In Kunis: $x \in [-\frac{1}{2}, \frac{1}{2})$ but we consider $y \in [0, 1)$.
- DFT, NDFT (and *Inverses*) presents a "minus" factor in the exponential.

How to fix this issue?

- $x \in [a, b) \to y = -\big((x-a)/(b-a) - 0.5\big)$;
- $y \in [-\frac{1}{2}, \frac{1}{2}] \to x = -(b-a)(y - 0.5) + a$.

Nonequispaced FFT: alghoritm and error estimation

- We consider NFFT from frequency domain to space domain on nonequispaced nodes (not forward).

Nonequispaced FFT: alghoritm and error estimation

- We consider NFFT from frequency domain to space domain on nonequispaced nodes (not forward).

What we need:

- nodes $x_j = [-\frac{1}{2}, \frac{1}{2})$ with $j = 0 \ldots, N$;
- frequencies $I_N = [-\frac{N}{2}, \frac{N}{2})$;
- an oversampling factor $\sigma > 1$, setting $n = \sigma N$.

Nonequispaced FFT: alghoritm and error estimation

- We consider NFFT from frequency domain to space domain on nonequispaced nodes (not forward).

What we need:

- nodes $x_j = [-\frac{1}{2}, \frac{1}{2})$ with $j = 0 \ldots, N$;
- frequencies $I_N = [-\frac{N}{2}, \frac{N}{2})$;
- an oversampling factor $\sigma > 1$, setting $n = \sigma N$.

IDEA: $f(x) = \sum_{k \in I_N} \hat{f}_k e^{-2\pi ikx} \approx s_1(x) = \sum_{l \in I_n} g_l \tilde{\varphi}\left(x - \frac{l}{n}\right)$, with $\tilde{\varphi}(x) = \sum_{r \in \mathbb{Z}} \varphi(x + r)$, with $\hat{f}_k$ given.

Nonequispaced FFT: alghoritm and error estimation

- We consider NFFT from frequency domain to space domain on nonequispaced nodes (not forward).

What we need:

- nodes $x_j = [-\frac{1}{2}, \frac{1}{2})$ with $j = 0 \ldots, N$;
- frequencies $I_N = [-\frac{N}{2}, \frac{N}{2})$;
- an oversampling factor $\sigma > 1$, setting $n = \sigma N$.

IDEA: $f(x) = \sum_{k \in I_N} \hat{f}_k e^{-2\pi ikx} \approx s_1(x) = \sum_{l \in I_n} g_l \tilde{\varphi}\left(x - \frac{l}{n}\right)$, with $\tilde{\varphi}(x) = \sum_{r \in \mathbb{Z}} \varphi(x + r)$, with $\hat{f}_k$ given.

1. Cut-off in frequency domain

Nonequispaced FFT: alghoritm and error estimation

- We consider NFFT from frequency domain to space domain on nonequispaced nodes (not forward).

What we need:

- nodes $x_j = [-\frac{1}{2}, \frac{1}{2})$ with $j = 0 \ldots, N$;
- frequencies $I_N = [-\frac{N}{2}, \frac{N}{2})$;
- an oversampling factor $\sigma > 1$, setting $n = \sigma N$.

IDEA: $f(x) = \sum_{k \in I_N} \hat{f}_k e^{-2\pi i k x} \approx s_1(x) = \sum_{l \in I_n} g_l \tilde{\varphi}\left(x - \frac{l}{n}\right)$, with $\tilde{\varphi}(x) = \sum_{r \in \mathbb{Z}} \varphi(x + r)$, with $\hat{f}_k$ given.

1. Cut-off in frequency domain

$$s_1(x) = \sum_{k \in \mathbb{Z}} \hat{g}_k \hat{\varphi}_k e^{-2\pi i k x} = \sum_{k \in I_n} \hat{g}_k \hat{\varphi}_k e^{-2\pi i k x} + \sum_{r \in \mathbb{Z} \setminus \{0\}} \sum_{k \in I_n} \hat{g}_k \hat{\varphi}_{k+nr} e^{-2\pi i (k+nr) x},$$

Nonequispaced FFT: alghoritm and error estimation

- We consider NFFT from frequency domain to space domain on nonequispaced nodes (not forward).

What we need:

- nodes $x_j = [-\frac{1}{2}, \frac{1}{2})$ with $j = 0 \dots, N$;
- frequencies $I_N = [-\frac{N}{2}, \frac{N}{2})$;
- an oversampling factor $\sigma > 1$, setting $n = \sigma N$.

IDEA: $f(x) = \sum_{k \in I_N} \hat{f}_k e^{-2\pi ikx} \approx s_1(x) = \sum_{l \in I_n} g_l \tilde{\varphi}\left(x - \frac{l}{n}\right)$, with $\tilde{\varphi}(x) = \sum_{r \in \mathbb{Z}} \varphi(x + r)$, with $\hat{f}_k$ given.

1. Cut-off in frequency domain

$$s_1(x) = \sum_{k \in \mathbb{Z}} \hat{g}_k \hat{\varphi}_k e^{-2\pi ikx} = \sum_{k \in I_n} \hat{g}_k \hat{\varphi}_k e^{-2\pi ikx} + \sum_{r \in \mathbb{Z} \setminus \{0\}} \sum_{k \in I_n} \hat{g}_k \hat{\varphi}_{k+nr} e^{-2\pi i(k+nr)x},$$

- $\hat{g}_k = \sum_{l \in I_n} g_l e^{2\pi ik\frac{l}{n}}$
- $\hat{g}_k = \begin{cases} \hat{f}_k / \hat{\varphi}_k & \text{if } k \in I_N \\ 0 & \text{if } k \in I_n \setminus I_N \end{cases}$
- FFT: $g_l = \frac{1}{n} \sum_{k \in I_N} \hat{g}_k e^{-2\pi ik\frac{l}{n}}$ with $l \in I_n$,

- $\hat{\varphi}_k = \int_{-1/2}^{1/2} \tilde{\varphi}(x) e^{2\pi ikx} \mathrm{d}x, \quad k \in \mathbb{Z}$.
- if $\hat{\varphi}_k$ are small enough for $k \in \mathbb{Z} \setminus I_n$ and $\hat{\varphi}_k \neq 0$ for $k \in I_N$.
- aliasing error

2. Cut-off in time/space domain

2. Cut-off in time/space domain

If φ is well localized in time domain $\psi(x) = \varphi(x)\chi_{[-\frac{m}{n}, \frac{m}{n}]}(x)$, $m \ll N$.

2. Cut-off in time/space domain

If φ is well localized in time domain $\psi(x) = \varphi(x)\chi_{[-\frac{m}{n}, \frac{m}{n}]}(x)$, $m \ll N$.

$$f(x_j) \approx s_1(x_j) \approx s(x_j) = \sum_{l \in I_{n,m}(x_j)} g_l \tilde{\psi}\left(x_j - \frac{l}{n}\right), \text{ with } \tilde{\psi} = \sum_{r \in \mathbb{Z}} \psi(x + r)$$

where $I_{n,m}(x_j) = \{l \in I_n : nx_j - m \leq l \leq nx_j + m\}$.

2. Cut-off in time/space domain

If φ is well localized in time domain $\psi(x) = \varphi(x)\chi_{[-\frac{m}{n},\frac{m}{n}]}(x)$, $m \ll N$.

$$f(x_j) \approx s_1(x_j) \approx s(x_j) = \sum_{l \in I_{n,m}(x_j)} g_l \tilde{\psi}\Big(x_j - \frac{l}{n}\Big), \text{ with } \tilde{\psi} = \sum_{r \in \mathbb{Z}} \psi(x+r)$$

where $I_{n,m}(x_j) = \{l \in I_n : nx_j - m \le l \le nx_j + m\}$.

- It contains at most $2m+1$ nonzero summands;

2. Cut-off in time/space domain

If φ is well localized in time domain $\psi(x) = \varphi(x)\chi_{[-\frac{m}{n},\frac{m}{n}]}(x)$, $m \ll N$.

$$f(x_j) \approx s_1(x_j) \approx s(x_j) = \sum_{l \in I_{n,m}(x_j)} g_l \tilde{\psi}\left(x_j - \frac{l}{n}\right), \text{ with } \tilde{\psi} = \sum_{r \in \mathbb{Z}} \psi(x + r)$$

where $I_{n,m}(x_j) = \{l \in I_n : nx_j - m \leq l \leq nx_j + m\}$.

- It contains at most $2m + 1$ nonzero summands;
- Truncation error.

2. Cut-off in time/space domain

If φ is well localized in time domain $\psi(x) = \varphi(x)\chi_{[-\frac{m}{n}, \frac{m}{n}]}(x)$, $m \ll N$.

$$f(x_j) \approx s_1(x_j) \approx s(x_j) = \sum_{l \in I_{n,m}(x_j)} g_l \tilde{\psi}\Big(x_j - \frac{l}{n}\Big), \text{ with } \tilde{\psi} = \sum_{r \in \mathbb{Z}} \psi(x + r)$$

where $I_{n,m}(x_j) = \{l \in I_n : nx_j - m \leq l \leq nx_j + m\}$.

- It contains at most $2m + 1$ nonzero summands;
- Truncation error.

The alghoritm

2. Cut-off in time/space domain

If φ is well localized in time domain $\psi(x) = \varphi(x)\chi_{[-\frac{m}{n},\frac{m}{n}]}(x)$, $m \ll N$.

$$f(x_j) \approx s_1(x_j) \approx s(x_j) = \sum_{l \in I_{n,m}(x_j)} g_l \tilde{\psi}\left(x_j - \frac{l}{n}\right), \text{ with } \tilde{\psi} = \sum_{r \in \mathbb{Z}} \psi(x+r)$$

where $I_{n,m}(x_j) = \{l \in I_n : nx_j - m \leq l \leq nx_j + m\}$.

- It contains at most $2m+1$ nonzero summands;
- Truncation error.

The alghoritm

Given $N \in \mathbb{N}$, $\sigma > 1$, $n = \sigma N$, $x_j \in [-\frac{1}{2}, \frac{1}{2})$ and $\hat{f}_k \in \mathbb{C}$:

2. Cut-off in time/space domain

If φ is well localized in time domain $\psi(x) = \varphi(x)\chi_{[-\frac{m}{n},\frac{m}{n}]}(x)$, $m \ll N$.

$$f(x_j) \approx s_1(x_j) \approx s(x_j) = \sum_{l \in I_{n,m}(x_j)} g_l \tilde{\psi}\left(x_j - \frac{l}{n}\right), \text{ with } \tilde{\psi} = \sum_{r \in \mathbb{Z}} \psi(x+r)$$

where $I_{n,m}(x_j) = \{l \in I_n : nx_j - m \leq l \leq nx_j + m\}$.

- It contains at most $2m+1$ nonzero summands;
- Truncation error.

The alghoritm

Given $N \in \mathbb{N}$, $\sigma > 1$, $n = \sigma N$, $x_j \in [-\frac{1}{2}, \frac{1}{2})$ and $\hat{f}_k \in \mathbb{C}$:

1. Precompute $\hat{\varphi}_k$, with $k \in I_N$.

2. Cut-off in time/space domain

If φ is well localized in time domain $\psi(x) = \varphi(x)\chi_{[-\frac{m}{n},\frac{m}{n}]}(x)$, $m \ll N$.

$$f(x_j) \approx s_1(x_j) \approx s(x_j) = \sum_{l \in I_{n,m}(x_j)} g_l \tilde{\psi}\left(x_j - \frac{l}{n}\right), \text{ with } \tilde{\psi} = \sum_{r \in \mathbb{Z}} \psi(x+r)$$

where $I_{n,m}(x_j) = \{l \in I_n : nx_j - m \leq l \leq nx_j + m\}$.

- It contains at most $2m+1$ nonzero summands;
- Truncation error.

The alghoritm

Given $N \in \mathbb{N}$, $\sigma > 1$, $n = \sigma N$, $x_j \in [-\frac{1}{2}, \frac{1}{2})$ and $\hat{f}_k \in \mathbb{C}$:

1. Precompute $\hat{\varphi}_k$, with $k \in I_N$.
2. Precompute $\psi(x_j - \frac{l}{n})$, with $l \in I_{n,m}(x_j)$.

2. Cut-off in time/space domain

If φ is well localized in time domain $\psi(x) = \varphi(x)\chi_{[-\frac{m}{n},\frac{m}{n}]}(x)$, $m \ll N$.

$$f(x_j) \approx s_1(x_j) \approx s(x_j) = \sum_{l \in I_{n,m}(x_j)} g_l \tilde{\psi}\left(x_j - \frac{l}{n}\right), \text{ with } \tilde{\psi} = \sum_{r \in \mathbb{Z}} \psi(x+r)$$

where $I_{n,m}(x_j) = \{l \in I_n : nx_j - m \leq l \leq nx_j + m\}$.

- It contains at most $2m+1$ nonzero summands;
- Truncation error.

The alghoritm

Given $N \in \mathbb{N}$, $\sigma > 1$, $n = \sigma N$, $x_j \in [-\frac{1}{2}, \frac{1}{2})$ and $\hat{f}_k \in \mathbb{C}$:

1. Precompute $\hat{\varphi}_k$, with $k \in I_N$.
2. Precompute $\psi(x_j - \frac{l}{n})$, with $l \in I_{n,m}(x_j)$.
3. Generate $\hat{g}_k = \hat{f}_k / \hat{\varphi}_k$, with $k \in I_N$.

2. Cut-off in time/space domain

If φ is well localized in time domain $\psi(x) = \varphi(x)\chi_{[-\frac{m}{n},\frac{m}{n}]}(x)$, $m \ll N$.

$$f(x_j) \approx s_1(x_j) \approx s(x_j) = \sum_{l \in I_{n,m}(x_j)} g_l \tilde{\psi}\left(x_j - \frac{l}{n}\right), \text{ with } \tilde{\psi} = \sum_{r \in \mathbb{Z}} \psi(x+r)$$

where $I_{n,m}(x_j) = \{l \in I_n : nx_j - m \leq l \leq nx_j + m\}$.

- It contains at most $2m+1$ nonzero summands;
- Truncation error.

The alghoritm

Given $N \in \mathbb{N}$, $\sigma > 1$, $n = \sigma N$, $x_j \in [-\frac{1}{2}, \frac{1}{2})$ and $\hat{f}_k \in \mathbb{C}$:

1. Precompute $\hat{\varphi}_k$, with $k \in I_N$.
2. Precompute $\psi(x_j - \frac{l}{n})$, with $l \in I_{n,m}(x_j)$.
3. Generate $\hat{g}_k = \hat{f}_k / \hat{\varphi}_k$, with $k \in I_N$.
4. Compute g_l using a d-variate FFT, in our case $d = 1$ (FFTW library is used).

2. Cut-off in time/space domain

If φ is well localized in time domain $\psi(x) = \varphi(x)\chi_{[-\frac{m}{n},\frac{m}{n}]}(x)$, $m \ll N$.

$$f(x_j) \approx s_1(x_j) \approx s(x_j) = \sum_{l \in I_{n,m}(x_j)} g_l \tilde{\psi}\Big(x_j - \frac{l}{n}\Big), \text{ with } \tilde{\psi} = \sum_{r \in \mathbb{Z}} \psi(x+r)$$

where $I_{n,m}(x_j) = \{l \in I_n : nx_j - m \le l \le nx_j + m\}$.

- It contains at most $2m+1$ nonzero summands;
- Truncation error.

The alghoritm

Given $N \in \mathbb{N}$, $\sigma > 1$, $n = \sigma N$, $x_j \in [-\frac{1}{2}, \frac{1}{2})$ and $\hat{f}_k \in \mathbb{C}$:

1. Precompute $\hat{\varphi}_k$, with $k \in I_N$.
2. Precompute $\psi(x_j - \frac{l}{n})$, with $l \in I_{n,m}(x_j)$.
3. Generate $\hat{g}_k = \hat{f}_k / \hat{\varphi}_k$, with $k \in I_N$.
4. Compute g_l using a d-variate FFT, in our case $d = 1$ (FFTW library is used).
5. Set $s(x_j) = \displaystyle\sum_{l \in I_{n,m}(x_j)} g_l \tilde{\psi}\Big(x_j - \frac{l}{n}\Big)$.

2. Cut-off in time/space domain

If φ is well localized in time domain $\psi(x) = \varphi(x)\chi_{[-\frac{m}{n},\frac{m}{n}]}(x)$, $m \ll N$.

$$f(x_j) \approx s_1(x_j) \approx s(x_j) = \sum_{l \in I_{n,m}(x_j)} g_l \tilde{\psi}\left(x_j - \frac{l}{n}\right), \text{ with } \tilde{\psi} = \sum_{r \in \mathbb{Z}} \psi(x + r)$$

where $I_{n,m}(x_j) = \{l \in I_n : nx_j - m \leq l \leq nx_j + m\}$.

- It contains at most $2m + 1$ nonzero summands;
- Truncation error.

The alghoritm

Given $N \in \mathbb{N}$, $\sigma > 1$, $n = \sigma N$, $x_j \in [-\frac{1}{2}, \frac{1}{2})$ and $\hat{f}_k \in \mathbb{C}$:

1. Precompute $\hat{\varphi}_k$, with $k \in I_N$.
2. Precompute $\psi(x_j - \frac{l}{n})$, with $l \in I_{n,m}(x_j)$.
3. Generate $\hat{g}_k = \hat{f}_k / \hat{\varphi}_k$, with $k \in I_N$.
4. Compute g_l using a d-variate FFT, in our case $d = 1$ (FFTW library is used).
5. Set $s(x_j) = \sum_{l \in I_{n,m}(x_j)} g_l \tilde{\psi}\left(x_j - \frac{l}{n}\right)$.

The values $s(x_j)$ approximate $f(x_j)$.

Error estimate

$$E(x_j) = |f(x_j) - s(x_j)| \leq E_a(x_j) + E_t(x_j) = C(\sigma, m)||\hat{f}||_1,$$

Error estimate

$$E(x_j) = |f(x_j) - s(x_j)| \leq E_a(x_j) + E_t(x_j) = C(\sigma, m)||\hat{f}||_1,$$

Default window function: Dilated Keiser-Bessel functions

$$\varphi(x) = \frac{1}{\pi} \begin{cases} \dfrac{\sinh(b\sqrt{m^2 - n^2x^2})}{\sqrt{m^2 - n^2x^2}} & \text{for } |x| \leq \dfrac{m}{n}, \quad \text{with } b = \pi\left(2 - \dfrac{1}{\alpha}\right), \\[2ex] \dfrac{\sinh(b\sqrt{n^2x^2 - m^2})}{\sqrt{n^2x^2 - m^2}} & \text{otherwise,} \end{cases}$$

$$\hat{\varphi}_k = \frac{1}{n} \begin{cases} I_0\left(m\sqrt{b^2 - \left(\dfrac{2\pi k}{n}\right)^2}\right) & \text{for } k = -n\left(1 - \frac{1}{2\sigma}\right), \ldots, n\left(1 - \frac{1}{2\sigma}\right), \\[2ex] 0 & \text{otherwise,} \end{cases}$$

with $C(\sigma, m) = 4\pi(\sqrt{m} + m)\sqrt[4]{1 - \frac{1}{\sigma}}\, e^{-2\pi m\sqrt{1 - \frac{1}{\sigma}}}$.

Comparison between MFT, FFT and NFFT

Comparison between MFT, FFT and NFFT

Computational Cost

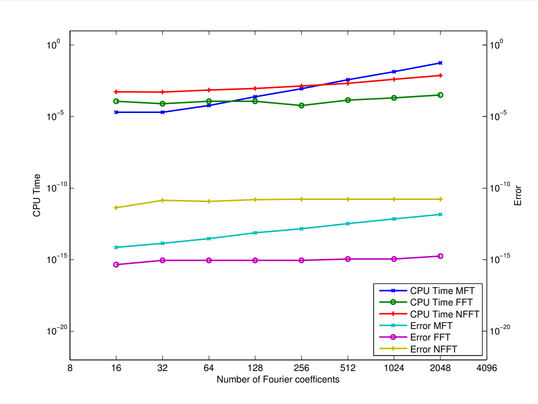
- NDFT: $\mathcal{O}(NM)$;
- NFFT: $\mathcal{O}((\sigma N)^d \log(\sigma N) + m^d M)$.

Comparison between MFT, FFT and NFFT

Computational Cost

- NDFT: $\mathcal{O}(NM)$;
- NFFT: $\mathcal{O}((\sigma N)^d \log(\sigma N) + m^d M)$.

Comparison between MFT, FFT and NFFT: from frequency domain to space domain



A standard hyperbolic PDE

A simple hyperbolic PDE

$$\begin{cases} \dfrac{\partial u}{\partial t} + a\dfrac{\partial u}{\partial x} = 0 & x \in \mathbb{R},\ a \neq 0,\ t > 0; \\ u(x,0) = u_0(x) & x \in \mathbb{R}. \end{cases}$$

A standard hyperbolic PDE

A simple hyperbolic PDE

$$\begin{cases} \dfrac{\partial u}{\partial t} + a\dfrac{\partial u}{\partial x} = 0 & x \in \mathbb{R},\ a \neq 0,\ t > 0; \\ u(x,0) = u_0(x) & x \in \mathbb{R}. \end{cases}$$

Solution: $u(x,t) = u_0(x - at)$, a wave travelling at at the speed a.

A standard hyperbolic PDE

A simple hyperbolic PDE

$$\begin{cases} \dfrac{\partial u}{\partial t} + a\dfrac{\partial u}{\partial x} = 0 & x \in \mathbb{R},\ a \neq 0,\ t > 0; \\ u(x,0) = u_0(x) & x \in \mathbb{R}. \end{cases}$$

Solution: $u(x,t) = u_0(x - at)$, a wave travelling at at the speed a.

Characteristics curves $x(t)$ on the plane (x,t)

$$x(t) \text{ solution of ODE } \begin{cases} \dfrac{\mathrm{d}x}{\mathrm{d}t} = a & t > 0 \\ x(0) = x_0, & x_0 \in \mathbb{R}. \end{cases}$$

The solution $u(x,t)$ is *constant* along them because

$$\frac{\mathrm{d}}{\mathrm{d}t}u(x,t) = \frac{\mathrm{d}t}{\mathrm{d}t}\frac{\partial u}{\partial t}(x,t) + \frac{\mathrm{d}x}{\mathrm{d}t}\frac{\partial u}{\partial x}(x,t) = \frac{\partial u}{\partial t}(x,t) + a\frac{\partial u}{\partial x}(x,t) = 0.$$

Given a source $f(x,t)$ instead of 0 and $a = a(x,t)$, the result is the same (changing the colored text).

Solution of an hyperbolic PDE* on equispaced nodes

*** = supposing $a(x, t)$, the transport coefficient, periodic, and $f(x, t) = 0$.**

Solution of an hyperbolic PDE* on equispaced nodes

*** = supposing $a(x, t)$, the transport coefficient, periodic, and $f(x, t) = 0$.**

Forward and Backward Characteristic

Solution of an hyperbolic PDE* on equispaced nodes

*** = supposing $a(x,t)$, the transport coefficient, periodic, and $f(x,t) = 0$.**

Forward and Backward Characteristic

- Forward Characteristic: no! $(u(x_n, t_f) = u_0(x_s)$, $x_n \in \{x_{ne}\}_{ne}$, $+$ NFFT $+$ IFFT$)$

Solution of an hyperbolic PDE* on equispaced nodes

* = supposing $a(x, t)$, the transport coefficient, periodic, and $f(x, t) = 0$.

Forward and Backward Characteristic

- Forward Characteristic: no! $(u(x_n, t_f) = u_0(x_s)$, $x_n \in \{x_{ne}\}_{ne}$, + NFFT + IFFT)
- Backward Characteristic:

Solution of an hyperbolic PDE* on equispaced nodes

* = supposing $a(x, t)$, the transport coefficient, periodic, and $f(x, t) = 0$.

Forward and Backward Characteristic

- Forward Characteristic: no! $(u(x_n, t_f) = u_0(x_s)$, $x_n \in \{x_{ne}\}_{ne}$, + NFFT + IFFT)
- Backward Characteristic: setting $\tilde{t} = -t$ and $\tilde{x}(\tilde{t}) = x_s$

$$x_n \in \{x_{ne}\}_{ne} \text{ solution of } \begin{cases} \dfrac{\mathrm{d}x}{\mathrm{d}t} = -a(x,t) & t \in [-t_f, -t_0] \\ x(-t_f) = x_s, & x_s \in \{x_e\}_e \text{ equispaced.} \end{cases}$$

so $u(x_s, t_f) = u_0(x_n)$.

Solution of an hyperbolic PDE* on equispaced nodes

* = supposing $a(x,t)$, the transport coefficient, periodic, and $f(x,t)=0$.

Forward and Backward Characteristic

- Forward Characteristic: no! $(u(x_n,t_f)=u_0(x_s)$, $x_n \in \{x_{ne}\}_{ne}$, + NFFT + IFFT)
- Backward Characteristic: setting $\tilde{t}=-t$ and $\tilde{x}(\tilde{t})=x_s$

$$x_n \in \{x_{ne}\}_{ne} \text{ solution of } \begin{cases} \dfrac{\mathrm{d}x}{\mathrm{d}t} = -a(x,t) & t \in [-t_f, -t_0] \\ x(-t_f) = x_s, & x_s \in \{x_e\}_e \text{ equispaced.} \end{cases}$$

so $u(x_s,t_f)=u_0(x_n)$. We can solve the ODE with rk45 (adaptative) in MATLAB.

Solution of an hyperbolic PDE* on equispaced nodes

* = supposing $a(x,t)$, the transport coefficient, periodic, and $f(x,t) = 0$.

Forward and Backward Characteristic

- Forward Characteristic: no! ($u(x_n, t_f) = u_0(x_s)$, $x_n \in \{x_{ne}\}_{ne}$, + NFFT + IFFT)
- Backward Characteristic: setting $\tilde{t} = -t$ and $\tilde{x}(\tilde{t}) = x_s$

$$x_n \in \{x_{ne}\}_{ne} \text{ solution of } \begin{cases} \dfrac{\mathrm{d}x}{\mathrm{d}t} = -a(x,t) & t \in [-t_f, -t_0] \\ x(-t_f) = x_s, & x_s \in \{x_e\}_e \text{ equispaced.} \end{cases}$$

so $u(x_s, t_f) = u_0(x_n)$. We can solve the ODE with rk45 (adaptative) in MATLAB.

The alghoritm

Solution of an hyperbolic PDE* on equispaced nodes

* = supposing $a(x, t)$, the transport coefficient, periodic, and $f(x, t) = 0$.

Forward and Backward Characteristic

- Forward Characteristic: no! $(u(x_n, t_f) = u_0(x_s)$, $x_n \in \{x_{ne}\}_{ne}$, + NFFT + IFFT)
- Backward Characteristic: setting $\tilde{t} = -t$ and $\tilde{x}(\tilde{t}) = x_s$

$$x_n \in \{x_{ne}\}_{ne} \text{ solution of } \begin{cases} \dfrac{\mathrm{d}x}{\mathrm{d}t} = -a(x, t) & t \in [-t_f, -t_0] \\ x(-t_f) = x_s, & x_s \in \{x_e\}_e \text{ equispaced.} \end{cases}$$

so $u(x_s, t_f) = u_0(x_n)$. We can solve the ODE with rk45 (adaptative) in MATLAB.

The alghoritm

1. Are given:
   - a set of x_s equispaced nodes and a time interval $[t_0, t_f]$;
   - starting values $u_0(x_s)$;
   - an hyperbolic PDE*.

Solution of an hyperbolic PDE* on equispaced nodes

* = supposing $a(x,t)$, the transport coefficient, periodic, and $f(x,t)=0$.

Forward and Backward Characteristic

- Forward Characteristic: no! $(u(x_n, t_f) = u_0(x_s)$, $x_n \in \{x_{ne}\}_{ne}$, + NFFT + IFFT)
- Backward Characteristic: setting $\tilde{t} = -t$ and $\tilde{x}(\tilde{t}) = x_s$

$$x_n \in \{x_{ne}\}_{ne} \text{ solution of } \begin{cases} \dfrac{\mathrm{d}x}{\mathrm{d}t} = -a(x,t) & t \in [-t_f, -t_0] \\ x(-t_f) = x_s, & x_s \in \{x_e\}_e \text{ equispaced.} \end{cases}$$

so $u(x_s, t_f) = u_0(x_n)$. We can solve the ODE with rk45 (adaptative) in MATLAB.

The alghoritm

1. Are given:
   - a set of x_s equispaced nodes and a time interval $[t_0, t_f]$;
   - starting values $u_0(x_s)$;
   - an hyperbolic PDE*.
2. Find x_n with *Bacward Characteristic*, with x_s as starting value.

Solution of an hyperbolic PDE* on equispaced nodes

* = supposing $a(x,t)$, the transport coefficient, periodic, and $f(x,t) = 0$.

Forward and Backward Characteristic

- Forward Characteristic: no! $(u(x_n, t_f) = u_0(x_s)$, $x_n \in \{x_{ne}\}_{ne}$, + NFFT + IFFT)
- Backward Characteristic: setting $\tilde{t} = -t$ and $\tilde{x}(\tilde{t}) = x_s$

$$x_n \in \{x_{ne}\}_{ne} \text{ solution of } \begin{cases} \dfrac{\mathrm{d}x}{\mathrm{d}t} = -a(x,t) & t \in [-t_f, -t_0] \\ x(-t_f) = x_s, & x_s \in \{x_e\}_e \text{ equispaced.} \end{cases}$$

so $u(x_s, t_f) = u_0(x_n)$. We can solve the ODE with rk45 (adaptative) in MATLAB.

The alghoritm

1. Are given:
   - a set of x_s equispaced nodes and a time interval $[t_0, t_f]$;
   - starting values $u_0(x_s)$;
   - an hyperbolic PDE*.
2. Find x_n with *Bacward Characteristic*, with x_s as starting value.
3. Compute FFT on the starting values $u_0(x_s)$.

Solution of an hyperbolic PDE* on equispaced nodes

* = supposing $a(x, t)$, the transport coefficient, periodic, and $f(x, t) = 0$.

Forward and Backward Characteristic

- Forward Characteristic: no! $(u(x_n, t_f) = u_0(x_s)$, $x_n \in \{x_{ne}\}_{ne}$, + NFFT + IFFT)
- Backward Characteristic: setting $\tilde{t} = -t$ and $\tilde{x}(\tilde{t}) = x_s$

$$x_n \in \{x_{ne}\}_{ne} \text{ solution of } \begin{cases} \dfrac{\mathrm{d}x}{\mathrm{d}t} = -a(x, t) & t \in [-t_f, -t_0] \\ x(-t_f) = x_s, & x_s \in \{x_e\}_e \text{ equispaced.} \end{cases}$$

so $u(x_s, t_f) = u_0(x_n)$. We can solve the ODE with rk45 (adaptative) in MATLAB.

The alghoritm

1. Are given:
   - a set of x_s equispaced nodes and a time interval $[t_0, t_f]$;
   - starting values $u_0(x_s)$;
   - an hyperbolic PDE*.
2. Find x_n with *Bacward Characteristic*, with x_s as starting value.
3. Compute FFT on the starting values $u_0(x_s)$.
4. Compute NFFT on the set of $x_n \in \{x_{ne}\}_{ne}$ to rebuild the starting values $u_0(x_n)$ for every $x_n \in \{x_{ne}\}_{ne}$.

Solution of an hyperbolic PDE* on equispaced nodes

* = supposing $a(x,t)$, the transport coefficient, periodic, and $f(x,t)=0$.

Forward and Backward Characteristic

- Forward Characteristic: no! $(u(x_n, t_f) = u_0(x_s)$, $x_n \in \{x_{ne}\}_{ne}$, + NFFT + IFFT)
- Backward Characteristic: setting $\tilde{t} = -t$ and $\tilde{x}(\tilde{t}) = x_s$

$$x_n \in \{x_{ne}\}_{ne} \text{ solution of } \begin{cases} \dfrac{\mathrm{d}x}{\mathrm{d}t} = -a(x,t) & t \in [-t_f, -t_0] \\ x(-t_f) = x_s, & x_s \in \{x_e\}_e \text{ equispaced.} \end{cases}$$

so $u(x_s, t_f) = u_0(x_n)$. We can solve the ODE with rk45 (adaptative) in MATLAB.

The alghoritm

1. Are given:
   - a set of x_s equispaced nodes and a time interval $[t_0, t_f]$;
   - starting values $u_0(x_s)$;
   - an hyperbolic PDE*.
2. Find x_n with *Bacward Characteristic*, with x_s as starting value.
3. Compute FFT on the starting values $u_0(x_s)$.
4. Compute NFFT on the set of $x_n \in \{x_{ne}\}_{ne}$ to rebuild the starting values $u_0(x_n)$ for every $x_n \in \{x_{ne}\}_{ne}$.
5. $u_0(x_n)$ is the solution of the hyperbolic PDE on x_s at the final time t_f (i.e. $u(x_s, t_f)$).

Example

An hyperbolic PDE with periodic transport coefficient

$$\begin{cases} \dfrac{\partial u}{\partial t} - \sin(x)\dfrac{\partial u}{\partial x} = 0 & x \in [0, 2\pi),\ \ t \in (0, 1.571] \\ u(x, 0) = \sin(x), & x \in [0, 2\pi) \text{ eq.}, \end{cases} \quad \text{with} \quad \begin{cases} \dfrac{\mathrm{d}x}{\mathrm{d}t} = \sin(x) & t \in (0, 1.571] \\ x(0) = x, & x \in [0, 2\pi) \text{ eq.} \end{cases}$$

Example

An hyperbolic PDE with periodic transport coefficient

$$\begin{cases} \frac{\partial u}{\partial t} - \sin(x)\frac{\partial u}{\partial x} = 0 & x \in [0, 2\pi),\ t \in (0, 1.571] \\ u(x, 0) = \sin(x), & x \in [0, 2\pi) \text{ eq.}, \end{cases} \quad \text{with} \quad \begin{cases} \frac{\mathrm{d}x}{\mathrm{d}t} = \sin(x) & t \in (0, 1.571] \\ x(0) = x, & x \in [0, 2\pi) \text{ eq.} \end{cases}$$

Solution: $u(x, t) = \sin\left(2\tan^{-1}\left(e^{t}\tan\frac{x}{2}\right)\right)$.

Example

An hyperbolic PDE with periodic transport coefficient

$$\begin{cases} \dfrac{\partial u}{\partial t} - \sin(x)\dfrac{\partial u}{\partial x} = 0 & x \in [0, 2\pi),\ t \in (0, 1.571] \\ u(x, 0) = \sin(x), & x \in [0, 2\pi) \text{ eq.}, \end{cases} \quad \text{with} \quad \begin{cases} \dfrac{\mathrm{d}x}{\mathrm{d}t} = \sin(x) & t \in (0, 1.571] \\ x(0) = x, & x \in [0, 2\pi) \text{ eq.} \end{cases}$$

Solution: $u(x,t) = \sin\left(2\tan^{-1}\left(e^{t}\tan\frac{x}{2}\right)\right)$.