How to represent a function in a quantum computer?

Michel Alexis

University of Bonn

June 25, ACOTCA

How to represent a function in a quantum computer?

How to represent a function in a quantum computer?

More precisely:

How to represent a function in a quantum computer?

More precisely: the Quantum signal processing algorithm (QSP) takes a sequence of *phase factors* $\Psi = \{\psi_k\}_{k=0}^{\infty}$ in $(-\frac{\pi}{2}, \frac{\pi}{2})$ and an input variable $x \in [0, 1]$ and acts on a qubit (i.e., a unit vector in $\mathbb{C}^2$) by the infinite 2×2 matrix product $U_\infty(x, \Psi) :=$

How to represent a function in a quantum computer?

More precisely: the Quantum signal processing algorithm (QSP) takes a sequence of *phase factors* $\Psi = \{\psi_k\}_{k=0}^{\infty}$ in $(-\frac{\pi}{2}, \frac{\pi}{2})$ and an input variable $x \in [0, 1]$ and acts on a qubit (i.e., a unit vector in $\mathbb{C}^2$) by the infinite 2×2 matrix product $U_\infty(x, \Psi) :=$

$$\dots e^{i\psi_d Z} e^{i\theta X} \dots e^{i\theta X} e^{i\psi_1 Z} e^{i\theta X} e^{i\psi_0 Z} e^{i\theta X} e^{i\psi_1 Z} e^{i\theta X} \dots e^{i\theta X} e^{i\psi_d Z} \dots ,$$

How to represent a function in a quantum computer?

More precisely: the Quantum signal processing algorithm (QSP) takes a sequence of *phase factors* $\Psi = \{\psi_k\}_{k=0}^{\infty}$ in $(-\frac{\pi}{2}, \frac{\pi}{2})$ and an input variable $x \in [0, 1]$ and acts on a qubit (i.e., a unit vector in $\mathbb{C}^2$) by the infinite 2×2 matrix product $U_\infty(x, \Psi) :=$

$$\dots e^{i\psi_d Z} e^{i\theta X} \dots e^{i\theta X} e^{i\psi_1 Z} e^{i\theta X} e^{i\psi_0 Z} e^{i\theta X} e^{i\psi_1 Z} e^{i\theta X} \dots e^{i\theta X} e^{i\psi_d Z} \dots,$$

where $\theta = \arccos(x)$,

How to represent a function in a quantum computer?

More precisely: the Quantum signal processing algorithm (QSP) takes a sequence of *phase factors* $\Psi = \{\psi_k\}_{k=0}^{\infty}$ in $(-\frac{\pi}{2}, \frac{\pi}{2})$ and an input variable $x \in [0,1]$ and acts on a qubit (i.e., a unit vector in $\mathbb{C}^2$) by the infinite 2×2 matrix product $U_\infty(x, \Psi) :=$

$$\dots e^{i\psi_d Z} e^{i\theta X} \dots e^{i\theta X} e^{i\psi_1 Z} e^{i\theta X} e^{i\psi_0 Z} e^{i\theta X} e^{i\psi_1 Z} e^{i\theta X} \dots e^{i\theta X} e^{i\psi_d Z} \dots ,$$

where $\theta = \arccos(x)$,
and $X := \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ and $Z := \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$ are the Pauli matrices.

How to represent a function in a quantum computer?

More precisely: the Quantum signal processing algorithm (QSP) takes a sequence of *phase factors* $\Psi = \{\psi_k\}_{k=0}^{\infty}$ in $(-\frac{\pi}{2}, \frac{\pi}{2})$ and an input variable $x \in [0,1]$ and acts on a qubit (i.e., a unit vector in $\mathbb{C}^2$) by the infinite 2×2 matrix product $U_\infty(x, \Psi) :=$

$$\dots e^{i\psi_d Z} e^{i\theta X} \dots e^{i\theta X} e^{i\psi_1 Z} e^{i\theta X} e^{i\psi_0 Z} e^{i\theta X} e^{i\psi_1 Z} e^{i\theta X} \dots e^{i\theta X} e^{i\psi_d Z} \dots,$$

where $\theta = \arccos(x)$,
and $X := \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ and $Z := \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$ are the Pauli matrices.
Question: Given a function $f : [0,1] \to [-1,1]$, does there exist an infinite phase factor sequence $\Psi = \{\psi_k\}_{k=0}^{\infty}$ for which $f(x)$ is the imaginary part of the upper left entry of $U_\infty(x, \Psi)$?

How to represent a function in a quantum computer?

More precisely: the Quantum signal processing algorithm (QSP) takes a sequence of *phase factors* $\Psi = \{\psi_k\}_{k=0}^{\infty}$ in $(-\frac{\pi}{2}, \frac{\pi}{2})$ and an input variable $x \in [0,1]$ and acts on a qubit (i.e., a unit vector in $\mathbb{C}^2$) by the infinite 2×2 matrix product $U_\infty(x, \Psi) :=$

$$\dots e^{i\psi_d Z} e^{i\theta X} \dots e^{i\theta X} e^{i\psi_1 Z} e^{i\theta X} e^{i\psi_0 Z} e^{i\theta X} e^{i\psi_1 Z} e^{i\theta X} \dots e^{i\theta X} e^{i\psi_d Z} \dots ,$$

where $\theta = \arccos(x)$,
and $X := \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ and $Z := \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$ are the Pauli matrices.
Question: Given a function $f : [0,1] \to [-1,1]$, does there exist an infinite phase factor sequence $\Psi = \{\psi_k\}_{k=0}^{\infty}$ for which $f(x)$ is the imaginary part of the upper left entry of $U_\infty(x, \Psi)$?
Why should we care? QSP is apparently a very simple and physically intuitive quantum algorithm.

Making sense of of the matrix product

Consider the truncated product

$$U_d(x,\Psi) := e^{i\psi_d Z} e^{i\theta X} e^{i\psi_{d-1} Z} e^{i\theta X} \dots e^{i\psi_0 Z} \dots e^{i\theta X} e^{i\psi_{d-1} Z} e^{i\theta X} e^{i\psi_d Z},$$

where $\theta = \arccos(x)$, $X := \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ and $Z := \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$.

Making sense of of the matrix product

Consider the truncated product

$$U_d(x,\Psi) := e^{i\psi_d Z} e^{i\theta X} e^{i\psi_{d-1} Z} e^{i\theta X} \dots e^{i\psi_0 Z} \dots e^{i\theta X} e^{i\psi_{d-1} Z} e^{i\theta X} e^{i\psi_d Z},$$

where $\theta = \arccos(x)$, $X := \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ and $Z := \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$.

Note $e^{i\theta X} = \begin{pmatrix} \cos\theta & i\sin\theta \\ i\sin\theta & \cos\theta \end{pmatrix}$

Making sense of of the matrix product

Consider the truncated product

$$U_d(x,\Psi) := e^{i\psi_d Z} e^{i\theta X} e^{i\psi_{d-1} Z} e^{i\theta X} \dots e^{i\psi_0 Z} \dots e^{i\theta X} e^{i\psi_{d-1} Z} e^{i\theta X} e^{i\psi_d Z},$$

where $\theta = \arccos(x)$, $X := \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ and $Z := \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$.

Note $e^{i\theta X} = \begin{pmatrix} \cos\theta & i\sin\theta \\ i\sin\theta & \cos\theta \end{pmatrix} = \begin{pmatrix} x & i\sqrt{1-x^2} \\ i\sqrt{1-x^2} & x \end{pmatrix}$

Making sense of of the matrix product

Consider the truncated product

$$U_d(x,\Psi) := e^{i\psi_d Z} e^{i\theta X} e^{i\psi_{d-1} Z} e^{i\theta X} \dots e^{i\psi_0 Z} \dots e^{i\theta X} e^{i\psi_{d-1} Z} e^{i\theta X} e^{i\psi_d Z},$$

where $\theta = \arccos(x)$, $X := \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ and $Z := \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$.

Note $e^{i\theta X} = \begin{pmatrix} \cos\theta & i\sin\theta \\ i\sin\theta & \cos\theta \end{pmatrix} = \begin{pmatrix} x & i\sqrt{1-x^2} \\ i\sqrt{1-x^2} & x \end{pmatrix}$ and $e^{i\psi_j Z} = \begin{pmatrix} e^{i\psi_j} & 0 \\ 0 & e^{-i\psi_j} \end{pmatrix}$.

Making sense of of the matrix product

Consider the truncated product

$$U_d(x,\Psi) := e^{i\psi_d Z} e^{i\theta X} e^{i\psi_{d-1} Z} e^{i\theta X} \dots e^{i\psi_0 Z} \dots e^{i\theta X} e^{i\psi_{d-1} Z} e^{i\theta X} e^{i\psi_d Z},$$

where $\theta = \arccos(x)$, $X := \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ and $Z := \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$.

Note $e^{i\theta X} = \begin{pmatrix} \cos\theta & i\sin\theta \\ i\sin\theta & \cos\theta \end{pmatrix} = \begin{pmatrix} x & i\sqrt{1-x^2} \\ i\sqrt{1-x^2} & x \end{pmatrix}$ and

$e^{i\psi_j Z} = \begin{pmatrix} e^{i\psi_j} & 0 \\ 0 & e^{-i\psi_j} \end{pmatrix}$. Then

$$e^{i\theta X} e^{i\psi_j Z} = \begin{pmatrix} poly(x) & poly(x)\sqrt{1-x^2} \\ poly(x)\sqrt{1-x^2} & poly(x) \end{pmatrix} \begin{pmatrix} e^{i\psi_j} & 0 \\ 0 & e^{-i\psi_j} \end{pmatrix}$$

Making sense of of the matrix product

Consider the truncated product

$$U_d(x,\Psi) := e^{i\psi_d Z} e^{i\theta X} e^{i\psi_{d-1} Z} e^{i\theta X} \dots e^{i\psi_0 Z} \dots e^{i\theta X} e^{i\psi_{d-1} Z} e^{i\theta X} e^{i\psi_d Z},$$

where $\theta = \arccos(x)$, $X := \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ and $Z := \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$.

Note $e^{i\theta X} = \begin{pmatrix} \cos\theta & i\sin\theta \\ i\sin\theta & \cos\theta \end{pmatrix} = \begin{pmatrix} x & i\sqrt{1-x^2} \\ i\sqrt{1-x^2} & x \end{pmatrix}$ and

$e^{i\psi_j Z} = \begin{pmatrix} e^{i\psi_j} & 0 \\ 0 & e^{-i\psi_j} \end{pmatrix}$. Then

$$e^{i\theta X} e^{i\psi_j Z} = \begin{pmatrix} poly(x) & poly(x)\sqrt{1-x^2} \\ poly(x)\sqrt{1-x^2} & poly(x) \end{pmatrix}$$

Making sense of of the matrix product

Consider the truncated product

$$U_d(x,\Psi) := e^{i\psi_d Z} e^{i\theta X} e^{i\psi_{d-1} Z} e^{i\theta X} \dots e^{i\psi_0 Z} \dots e^{i\theta X} e^{i\psi_{d-1} Z} e^{i\theta X} e^{i\psi_d Z},$$

where $\theta = \arccos(x)$, $X := \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ and $Z := \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$.

Note $e^{i\theta X} = \begin{pmatrix} \cos\theta & i\sin\theta \\ i\sin\theta & \cos\theta \end{pmatrix} = \begin{pmatrix} x & i\sqrt{1-x^2} \\ i\sqrt{1-x^2} & x \end{pmatrix}$ and

$e^{i\psi_j Z} = \begin{pmatrix} e^{i\psi_j} & 0 \\ 0 & e^{-i\psi_j} \end{pmatrix}$. Then

$e^{i\theta X} e^{i\psi_j Z} = \begin{pmatrix} poly(x) & poly(x)\sqrt{1-x^2} \\ poly(x)\sqrt{1-x^2} & poly(x) \end{pmatrix}$

Mutliplying two such matrices together yields another such matrix. Indeed

$$\begin{pmatrix} poly(x) & poly(x)\sqrt{1-x^2} \\ poly(x)\sqrt{1-x^2} & poly(x) \end{pmatrix} \begin{pmatrix} poly(x) & poly(x)\sqrt{1-x^2} \\ poly(x)\sqrt{1-x^2} & poly(x) \end{pmatrix}$$

Making sense of of the matrix product

Consider the truncated product

$$U_d(x,\Psi) := e^{i\psi_d Z} e^{i\theta X} e^{i\psi_{d-1} Z} e^{i\theta X} \dots e^{i\psi_0 Z} \dots e^{i\theta X} e^{i\psi_{d-1} Z} e^{i\theta X} e^{i\psi_d Z},$$

where $\theta = \arccos(x)$, $X := \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ and $Z := \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$.

Note $e^{i\theta X} = \begin{pmatrix} \cos\theta & i\sin\theta \\ i\sin\theta & \cos\theta \end{pmatrix} = \begin{pmatrix} x & i\sqrt{1-x^2} \\ i\sqrt{1-x^2} & x \end{pmatrix}$ and $e^{i\psi_j Z} = \begin{pmatrix} e^{i\psi_j} & 0 \\ 0 & e^{-i\psi_j} \end{pmatrix}$. Then

$$e^{i\theta X} e^{i\psi_j Z} = \begin{pmatrix} poly(x) & poly(x)\sqrt{1-x^2} \\ poly(x)\sqrt{1-x^2} & poly(x) \end{pmatrix}$$

Mutliplying two such matrices together yields another such matrix. Indeed

$$\begin{pmatrix} poly(x) & poly(x)\sqrt{1-x^2} \\ poly(x)\sqrt{1-x^2} & poly(x) \end{pmatrix} \begin{pmatrix} poly(x) & poly(x)\sqrt{1-x^2} \\ poly(x)\sqrt{1-x^2} & poly(x) \end{pmatrix}$$
$$= \begin{pmatrix} poly(x) & poly(x)\sqrt{1-x^2} \\ poly(x)\sqrt{1-x^2} & poly(x) \end{pmatrix}.$$

Making sense of of the matrix product

Consider the truncated product

$$U_d(x,\Psi) := e^{i\psi_d Z} e^{i\theta X} e^{i\psi_{d-1} Z} e^{i\theta X} \dots e^{i\psi_0 Z} \dots e^{i\theta X} e^{i\psi_{d-1} Z} e^{i\theta X} e^{i\psi_d Z},$$

where $\theta = \arccos(x)$, $X := \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ and $Z := \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$.

Note $e^{i\theta X} = \begin{pmatrix} \cos\theta & i\sin\theta \\ i\sin\theta & \cos\theta \end{pmatrix} = \begin{pmatrix} x & i\sqrt{1-x^2} \\ i\sqrt{1-x^2} & x \end{pmatrix}$ and

$e^{i\psi_j Z} = \begin{pmatrix} e^{i\psi_j} & 0 \\ 0 & e^{-i\psi_j} \end{pmatrix}$. Then

$e^{i\theta X} e^{i\psi_j Z} = \begin{pmatrix} poly(x) & poly(x)\sqrt{1-x^2} \\ poly(x)\sqrt{1-x^2} & poly(x) \end{pmatrix}$

Mutliplying two such matrices together yields another such matrix.

Making sense of of the matrix product

Consider the truncated product

$$U_d(x,\Psi) := e^{i\psi_d Z} e^{i\theta X} e^{i\psi_{d-1} Z} e^{i\theta X} \dots e^{i\psi_0 Z} \dots e^{i\theta X} e^{i\psi_{d-1} Z} e^{i\theta X} e^{i\psi_d Z},$$

where $\theta = \arccos(x)$, $X := \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ and $Z := \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$.

Note $e^{i\theta X} = \begin{pmatrix} \cos\theta & i\sin\theta \\ i\sin\theta & \cos\theta \end{pmatrix} = \begin{pmatrix} x & i\sqrt{1-x^2} \\ i\sqrt{1-x^2} & x \end{pmatrix}$ and

$e^{i\psi_j Z} = \begin{pmatrix} e^{i\psi_j} & 0 \\ 0 & e^{-i\psi_j} \end{pmatrix}$. Then

$e^{i\theta X} e^{i\psi_j Z} = \begin{pmatrix} poly(x) & poly(x)\sqrt{1-x^2} \\ poly(x)\sqrt{1-x^2} & poly(x) \end{pmatrix}$

Mutliplying two such matrices together yields another such matrix.

So $U_d(x,\Psi) = \begin{pmatrix} poly(x) & poly(x)\sqrt{1-x^2} \\ poly(x)\sqrt{1-x^2} & poly(x) \end{pmatrix}$.

Main Theorem

Main Theorem

Because the truncated matrices $U_d(x, \Psi)$ have upper left entries which are $poly(x)$, the QSP problem is actually about approximating f by polynomials generated in this fashion.

Main Theorem

Because the truncated matrices $U_d(x, \Psi)$ have upper left entries which are *poly*(x), the QSP problem is actually about approximating f by polynomials generated in this fashion.

Theorem (A.-Mnatsakanyan-Thiele, 2023)

For each f with $\|f\|_\infty < \frac{1}{\sqrt{2}}$, there exists a unique phase factor sequence Ψ such that the imaginary parts of the upper left entries of $U_d(x, \Psi)$ converge to f in $L^2\left(\frac{dx}{\sqrt{1-x^2}}\right)$, and we have the nonlinear Plancherel identity

$$\sum_k \log(1 + \tan^2 \psi_{|k|}) = -\frac{2}{\pi} \int_0^1 \log|1 - f(x)^2| \frac{dx}{\sqrt{1-x^2}} .$$

Main Theorem

Because the truncated matrices $U_d(x, \Psi)$ have upper left entries which are *poly*(x), the QSP problem is actually about approximating f by polynomials generated in this fashion.

Theorem (A.-Mnatsakanyan-Thiele, 2023)

For each f with $\|f\|_\infty < \frac{1}{\sqrt{2}}$, there exists a unique phase factor sequence Ψ such that the imaginary parts of the upper left entries of $U_d(x, \Psi)$ converge to f in $L^2\left(\frac{dx}{\sqrt{1-x^2}}\right)$, and we have the nonlinear Plancherel identity

$$\sum_k \log(1 + \tan^2 \psi_{|k|}) = -\frac{2}{\pi} \int_0^1 \log|1 - f(x)^2| \frac{dx}{\sqrt{1-x^2}} .$$

The key idea here is to use nonlinear Fourier analysis!

Some quick basics

Some quick basics

Given f a function on the unit disc $\mathbb{D} \subset \mathbb{C}$, define its reflection

$$f^*(z) := \overline{f\left(\frac{1}{\overline{z}}\right)}.$$

Some quick basics

Given f a function on the unit disc $\mathbb{D} \subset \mathbb{C}$, define its reflection

$$f^*(z) := \overline{f\left(\frac{1}{\overline{z}}\right)}.$$

If f is holomorphic on $\mathbb{D}$, then f^* is then holomorphic on the reflected disk $\mathbb{D}^*$ at ∞.

Some quick basics

Given f a function on the unit disc $\mathbb{D} \subset \mathbb{C}$, define its reflection

$$f^*(z) := \overline{f\left(\frac{1}{\overline{z}}\right)}.$$

If f is holomorphic on $\mathbb{D}$, then f^* is then holomorphic on the reflected disk $\mathbb{D}^*$ at ∞.
Given a Laurent series $f(z) = \sum_k a_k z^k$, the $*$ map reflects its Fourier support about the origin.

Some quick basics

Given f a function on the unit disc $\mathbb{D} \subset \mathbb{C}$, define its reflection

$$f^*(z) := \overline{f\left(\frac{1}{\overline{z}}\right)}.$$

If f is holomorphic on $\mathbb{D}$, then f^* is then holomorphic on the reflected disk $\mathbb{D}^*$ at ∞.
Given a Laurent series $f(z) = \sum_k a_k z^k$, the $*$ map reflects its Fourier support about the origin.
Recall the 2-dimensional unitary group $SU(2)$ consists of matrices $\begin{pmatrix} \alpha & \beta \\ -\overline{\beta} & \overline{\alpha} \end{pmatrix}$ satisfying the determinant condition $|\alpha|^2 + |\beta|^2 = 1$.

The $SU(2)$-nonlinear Fourier transform (NLFT)

The $SU(2)$-nonlinear Fourier transform (NLFT)

Fourier transform : $(F_k)_k \mapsto$ Laurent series $\sum_k F_k z^k$.

The $SU(2)$-nonlinear Fourier transform (NLFT)

Fourier transform : $(F_k)_k \mapsto$ Laurent series $\sum_k F_k z^k$.

nonlinear Fourier transform: $(F_k)_k \mapsto$ a matrix of Laurent series.

The $SU(2)$-nonlinear Fourier transform (NLFT)

Fourier transform : $(F_k)_k \mapsto$ Laurent series $\sum_k F_k z^k$.

nonlinear Fourier transform: $(F_k)_k \mapsto$ a matrix of Laurent series.
Define the nonlinear Fourier transform of a sequence $(F_k)_k$ as the $SU(2)$-valued function

$$\begin{pmatrix} a(z) & b(z) \\ -b^*(z) & a^*(z) \end{pmatrix} = \prod_{j=-\infty}^{\infty} \frac{1}{\sqrt{1+|F_j|^2}} \begin{pmatrix} 1 & F_j z^j \\ -\overline{F_j} z^{-j} & 1 \end{pmatrix}$$

where the matrix product should be read left to right as j increases.

The $SU(2)$-nonlinear Fourier transform (NLFT)

Fourier transform : $(F_k)_k \mapsto$ Laurent series $\sum_k F_k z^k$.

nonlinear Fourier transform: $(F_k)_k \mapsto$ a matrix of Laurent series.
Define the nonlinear Fourier transform of a sequence $(F_k)_k$ as the $SU(2)$-valued function

$$\begin{pmatrix} a(z) & b(z) \\ -b^*(z) & a^*(z) \end{pmatrix} = \prod_{j=-\infty}^{\infty} \frac{1}{\sqrt{1+|F_j|^2}} \begin{pmatrix} 1 & F_j z^j \\ -\overline{F_j} z^{-j} & 1 \end{pmatrix}$$

where the matrix product should be read left to right as j increases.
Note this is well-defined for compactly supported $\{F_k\}$, since only finitely many matrices above are not the identity.

The $SU(2)$-nonlinear Fourier transform (NLFT)

Fourier transform : $(F_k)_k \mapsto$ Laurent series $\sum_k F_k z^k$.

nonlinear Fourier transform: $(F_k)_k \mapsto$ a matrix of Laurent series.

Define the nonlinear Fourier transform of a sequence $(F_k)_k$ as the $SU(2)$-valued function

$$\begin{pmatrix} a(z) & b(z) \\ -b^*(z) & a^*(z) \end{pmatrix} = \prod_{j=-\infty}^{\infty} \frac{1}{\sqrt{1+|F_j|^2}} \begin{pmatrix} 1 & F_j z^j \\ -\overline{F_j} z^{-j} & 1 \end{pmatrix}$$

where the matrix product should be read left to right as j increases.

Note this is well-defined for compactly supported $\{F_k\}$, since only finitely many matrices above are not the identity.

We abbreviate this $SU(2)$ matrix as the pair (a, b).

The $SU(2)$-nonlinear Fourier transform (NLFT)

Fourier transform : $(F_k)_k \mapsto$ Laurent series $\sum_k F_k z^k$.

nonlinear Fourier transform: $(F_k)_k \mapsto$ a matrix of Laurent series.
Define the nonlinear Fourier transform of a sequence $(F_k)_k$ as the $SU(2)$-valued function

$$\begin{pmatrix} a(z) & b(z) \\ -b^*(z) & a^*(z) \end{pmatrix} = \prod_{j=-\infty}^{\infty} \frac{1}{\sqrt{1+|F_j|^2}} \begin{pmatrix} 1 & F_j z^j \\ -\overline{F_j} z^{-j} & 1 \end{pmatrix}$$

where the matrix product should be read left to right as j increases.
Note this is well-defined for compactly supported $\{F_k\}$, since only finitely many matrices above are not the identity.
We abbreviate this $SU(2)$ matrix as the pair (a, b).
Note (a, b) then satisfies the determinant condition $|a|^2 + |b|^2 = 1$ for $z \in \mathbb{T}$.

Why call this a Fourier transform?

Why call this a Fourier transform?

We do an informal computation, assuming F is "small."

Why call this a Fourier transform?

We do an informal computation, assuming F is "small."

$$\begin{pmatrix} a & b \\ -b^* & a^* \end{pmatrix} = \left(\prod_j (1+|F_j|^2)^{-\frac{1}{2}}\right) \prod_j \begin{pmatrix} 1 & F_j z^j \\ -\overline{F_j} z^{-j} & 1 \end{pmatrix}$$

Why call this a Fourier transform?

We do an informal computation, assuming F is "small."

$$\begin{pmatrix} a & b \\ -b^* & a^* \end{pmatrix} \approx \prod_j \begin{pmatrix} 1 & F_j z^j \\ -\overline{F_j} z^{-j} & 1 \end{pmatrix}$$

Why call this a Fourier transform?

We do an informal computation, assuming F is “small.”

$$\begin{pmatrix} a & b \\ -b^* & a^* \end{pmatrix} \approx \prod_j \left(\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + \begin{pmatrix} 0 & F_j z^j \\ -\overline{F_j} z^{-j} & 0 \end{pmatrix} \right)$$

Why call this a Fourier transform?

We do an informal computation, assuming F is "small."

$$\begin{pmatrix} a & b \\ -b^* & a^* \end{pmatrix} \approx \prod_j \left(\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + \begin{pmatrix} 0 & F_j z^j \\ -\overline{F_j} z^{-j} & 0 \end{pmatrix} \right)$$

$$= \sum_{n \geqslant 0} \sum_{j_1 < \ldots < j_n} \begin{pmatrix} 0 & F_{j_1} z^{j_1} \\ -\overline{F_{j_1}} z^{-j_1} & 0 \end{pmatrix} \cdots \begin{pmatrix} 0 & F_{j_n} z^{j_n} \\ -\overline{F_{j_n}} z^{-j_n} & 0 \end{pmatrix} .$$

Why call this a Fourier transform?

We do an informal computation, assuming F is "small."

$$\begin{pmatrix} a & b \\ -b^* & a^* \end{pmatrix} \approx \prod_j \left(\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + \begin{pmatrix} 0 & F_j z^j \\ -\overline{F_j} z^{-j} & 0 \end{pmatrix} \right)$$

$$= \sum_{n \geqslant 0} \sum_{j_1 < \ldots < j_n} \begin{pmatrix} 0 & F_{j_1} z^{j_1} \\ -\overline{F_{j_1}} z^{-j_1} & 0 \end{pmatrix} \cdots \begin{pmatrix} 0 & F_{j_n} z^{j_n} \\ -\overline{F_{j_n}} z^{-j_n} & 0 \end{pmatrix} .$$

$n = 0$: $\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$	$n = 1$:
$n = 2$:	$n = 3$:
n even :	n odd :
$a =$	$b =$

Why call this a Fourier transform?

We do an informal computation, assuming F is "small."

$$\begin{pmatrix} a & b \\ -b^* & a^* \end{pmatrix} \approx \prod_j \left(\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + \begin{pmatrix} 0 & F_j z^j \\ -\overline{F_j} z^{-j} & 0 \end{pmatrix} \right)$$

$$= \sum_{n \geqslant 0} \sum_{j_1 < \ldots < j_n} \begin{pmatrix} 0 & F_{j_1} z^{j_1} \\ -\overline{F_{j_1}} z^{-j_1} & 0 \end{pmatrix} \cdots \begin{pmatrix} 0 & F_{j_n} z^{j_n} \\ -\overline{F_{j_n}} z^{-j_n} & 0 \end{pmatrix} .$$

$n = 0$: $\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$	$n = 1$: $\sum_j \begin{pmatrix} 0 & F_j z^j \\ -\overline{F_j} z^{-j} & 0 \end{pmatrix}$
$n = 2$:	$n = 3$:
n even :	n odd :
$a =$	$b =$

Why call this a Fourier transform?

We do an informal computation, assuming F is "small."

$$\begin{pmatrix} a & b \\ -b^* & a^* \end{pmatrix} \approx \prod_j \left(\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + \begin{pmatrix} 0 & F_j z^j \\ -\overline{F_j} z^{-j} & 0 \end{pmatrix} \right)$$

$$= \sum_{n \geqslant 0} \sum_{j_1 < \ldots < j_n} \begin{pmatrix} 0 & F_{j_1} z^{j_1} \\ -\overline{F_{j_1}} z^{-j_1} & 0 \end{pmatrix} \cdots \begin{pmatrix} 0 & F_{j_n} z^{j_n} \\ -\overline{F_{j_n}} z^{-j_n} & 0 \end{pmatrix} .$$

$n = 0 :$ $\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$	$n = 1 :$ $\begin{pmatrix} 0 & \sum_j F_j z^j \\ -\sum_j \overline{F_j} z^{-j} & 0 \end{pmatrix}$
$n = 2 :$	$n = 3 :$
n even :	n odd :
$a =$	$b =$

Why call this a Fourier transform?

We do an informal computation, assuming F is "small."

$$\begin{pmatrix} a & b \\ -b^* & a^* \end{pmatrix} \approx \prod_j \left(\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + \begin{pmatrix} 0 & F_j z^j \\ -\overline{F_j} z^{-j} & 0 \end{pmatrix} \right)$$

$$= \sum_{n \geqslant 0} \sum_{j_1 < \ldots < j_n} \begin{pmatrix} 0 & F_{j_1} z^{j_1} \\ -\overline{F_{j_1}} z^{-j_1} & 0 \end{pmatrix} \cdots \begin{pmatrix} 0 & F_{j_n} z^{j_n} \\ -\overline{F_{j_n}} z^{-j_n} & 0 \end{pmatrix} .$$

$n = 0:$	$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$	$n = 1:$	$\begin{pmatrix} 0 & \sum_j F_j z^j \\ -\sum_j \overline{F_j} z^{-j} & 0 \end{pmatrix}$
$n = 2:$	$\sum_{j_1, j_2} \begin{pmatrix} 0 & F_{j_1} z^{j_1} \\ -\overline{F_{j_1}} z^{-j_1} & 0 \end{pmatrix} \begin{pmatrix} 0 & F_{j_2} z^{j_2} \\ -\overline{F_{j_2}} z^{-j_2} & 0 \end{pmatrix}$	$n = 3:$	
n even :		n odd :	
$a =$		$b =$	

Why call this a Fourier transform?

We do an informal computation, assuming F is "small."

$$\begin{pmatrix} a & b \\ -b^* & a^* \end{pmatrix} \approx \prod_j \left(\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + \begin{pmatrix} 0 & F_j z^j \\ -\overline{F_j} z^{-j} & 0 \end{pmatrix} \right)$$

$$= \sum_{n \geqslant 0} \sum_{j_1 < \ldots < j_n} \begin{pmatrix} 0 & F_{j_1} z^{j_1} \\ -\overline{F_{j_1}} z^{-j_1} & 0 \end{pmatrix} \cdots \begin{pmatrix} 0 & F_{j_n} z^{j_n} \\ -\overline{F_{j_n}} z^{-j_n} & 0 \end{pmatrix} .$$

$n = 0$: $\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$	$n = 1$: $\begin{pmatrix} 0 & \sum_j F_j z^j \\ -\sum_j \overline{F_j} z^{-j} & 0 \end{pmatrix}$
$n = 2$: $\sum_{j_1, j_2} \begin{pmatrix} -F_{j_1} \overline{F_{j_2}} z^{j_1 - j_2} & 0 \\ 0 & -\overline{F_{j_1}} F_{j_2} z^{j_2 - j_1} \end{pmatrix}$	$n = 3$:
n even :	n odd :
$a =$	$b =$

Why call this a Fourier transform?

We do an informal computation, assuming F is "small."

$$\begin{pmatrix} a & b \\ -b^* & a^* \end{pmatrix} \approx \prod_j \left(\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + \begin{pmatrix} 0 & F_j z^j \\ -\overline{F_j} z^{-j} & 0 \end{pmatrix} \right)$$

$$= \sum_{n \geqslant 0} \sum_{j_1 < \ldots < j_n} \begin{pmatrix} 0 & F_{j_1} z^{j_1} \\ -\overline{F_{j_1}} z^{-j_1} & 0 \end{pmatrix} \cdots \begin{pmatrix} 0 & F_{j_n} z^{j_n} \\ -\overline{F_{j_n}} z^{-j_n} & 0 \end{pmatrix} .$$

$n = 0:$	$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$	$n = 1:$	$\begin{pmatrix} 0 & \sum_j F_j z^j \\ -\sum_j \overline{F_j} z^{-j} & 0 \end{pmatrix}$
$n = 2:$	$\sum_{j_1, j_2} \begin{pmatrix} O(F)^2 & 0 \\ 0 & O(F)^2 \end{pmatrix}$	$n = 3:$	
n even :		n odd :	
$a =$		$b =$	

Why call this a Fourier transform?

We do an informal computation, assuming F is "small."

$$\begin{pmatrix} a & b \\ -b^* & a^* \end{pmatrix} \approx \prod_j \left(\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + \begin{pmatrix} 0 & F_j z^j \\ -\overline{F_j} z^{-j} & 0 \end{pmatrix} \right)$$

$$= \sum_{n \geqslant 0} \sum_{j_1 < \ldots < j_n} \begin{pmatrix} 0 & F_{j_1} z^{j_1} \\ -\overline{F_{j_1}} z^{-j_1} & 0 \end{pmatrix} \cdots \begin{pmatrix} 0 & F_{j_n} z^{j_n} \\ -\overline{F_{j_n}} z^{-j_n} & 0 \end{pmatrix} .$$

$n = 0 :$ $\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$

$n = 2 :$ $\sum_{j_1, j_2} \begin{pmatrix} O(F)^2 & 0 \\ 0 & O(F)^2 \end{pmatrix}$

n even :

$a =$

$n = 1 :$ $\begin{pmatrix} 0 & \sum_j F_j z^j \\ -\sum_j \overline{F_j} z^{-j} & 0 \end{pmatrix}$

$n = 3 :$ $\sum_{j_1 < j_2 < j_3} \begin{pmatrix} O(F)^2 & 0 \\ 0 & O(F)^2 \end{pmatrix} \begin{pmatrix} 0 & F_{j_3} z^{j_3} \\ -\overline{F_{j_3}} z^{-j_3} & 0 \end{pmatrix}$

n odd :

$b =$

Why call this a Fourier transform?

We do an informal computation, assuming F is "small."

$$\begin{pmatrix} a & b \\ -b^* & a^* \end{pmatrix} \approx \prod_j \left(\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + \begin{pmatrix} 0 & F_j z^j \\ -\overline{F_j} z^{-j} & 0 \end{pmatrix} \right)$$

$$= \sum_{n \geqslant 0} \sum_{j_1 < \ldots < j_n} \begin{pmatrix} 0 & F_{j_1} z^{j_1} \\ -\overline{F_{j_1}} z^{-j_1} & 0 \end{pmatrix} \cdots \begin{pmatrix} 0 & F_{j_n} z^{j_n} \\ -\overline{F_{j_n}} z^{-j_n} & 0 \end{pmatrix} .$$

$n = 0 :$ $\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$

$n = 2 :$ $\sum_{j_1, j_2} \begin{pmatrix} O(F)^2 & 0 \\ 0 & O(F)^2 \end{pmatrix}$

n even :

$a =$

$n = 1 :$ $\begin{pmatrix} 0 & \sum_j F_j z^j \\ -\sum_j \overline{F_j} z^{-j} & 0 \end{pmatrix}$

$n = 3 :$ $\sum_{j_1 < j_2 < j_3} \begin{pmatrix} 0 & O(F)^3 \\ O(F)^3 & 0 \end{pmatrix}$

n odd :

$b =$

Why call this a Fourier transform?

We do an informal computation, assuming F is "small."

$$\begin{pmatrix} a & b \\ -b^* & a^* \end{pmatrix} \approx \prod_j \left(\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + \begin{pmatrix} 0 & F_j z^j \\ -\overline{F_j} z^{-j} & 0 \end{pmatrix} \right)$$

$$= \sum_{n \geqslant 0} \sum_{j_1 < \ldots < j_n} \begin{pmatrix} 0 & F_{j_1} z^{j_1} \\ -\overline{F_{j_1}} z^{-j_1} & 0 \end{pmatrix} \cdots \begin{pmatrix} 0 & F_{j_n} z^{j_n} \\ -\overline{F_{j_n}} z^{-j_n} & 0 \end{pmatrix} .$$

$n = 0:$	$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$	$n = 1:$	$\begin{pmatrix} 0 & \sum_j F_j z^j \\ -\sum_j \overline{F_j} z^{-j} & 0 \end{pmatrix}$
$n = 2:$	$\sum_{j_1, j_2} \begin{pmatrix} O(F)^2 & 0 \\ 0 & O(F)^2 \end{pmatrix}$	$n = 3:$	$\sum_{j_1 < j_2 < j_3} \begin{pmatrix} 0 & O(F)^3 \\ O(F)^3 & 0 \end{pmatrix}$
n even :	$\sum_{j_1 < \ldots < j_n} \begin{pmatrix} O(F)^n & 0 \\ 0 & O(F)^n \end{pmatrix}$	n odd :	
$a =$		$b =$	

Why call this a Fourier transform?

We do an informal computation, assuming F is "small."

$$\begin{pmatrix} a & b \\ -b^* & a^* \end{pmatrix} \approx \prod_j \left(\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + \begin{pmatrix} 0 & F_j z^j \\ -\overline{F_j} z^{-j} & 0 \end{pmatrix} \right)$$

$$= \sum_{n \geqslant 0} \sum_{j_1 < \ldots < j_n} \begin{pmatrix} 0 & F_{j_1} z^{j_1} \\ -\overline{F_{j_1}} z^{-j_1} & 0 \end{pmatrix} \cdots \begin{pmatrix} 0 & F_{j_n} z^{j_n} \\ -\overline{F_{j_n}} z^{-j_n} & 0 \end{pmatrix} .$$

$n = 0$: $\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$

$n = 2$: $\sum_{j_1, j_2} \begin{pmatrix} O(F)^2 & 0 \\ 0 & O(F)^2 \end{pmatrix}$

n even : $\sum_{j_1 < \ldots < j_n} \begin{pmatrix} O(F)^n & 0 \\ 0 & O(F)^n \end{pmatrix}$

$a =$

$n = 1$: $\begin{pmatrix} 0 & \sum_j F_j z^j \\ -\sum_j \overline{F_j} z^{-j} & 0 \end{pmatrix}$

$n = 3$: $\sum_{j_1 < j_2 < j_3} \begin{pmatrix} 0 & O(F)^3 \\ O(F)^3 & 0 \end{pmatrix}$

n odd : $\sum_{j_1 < \ldots < j_n} \begin{pmatrix} 0 & O(F)^n \\ O(F)^n & 0 \end{pmatrix}$

$b =$

Why call this a Fourier transform?

We do an informal computation, assuming F is "small."

$$\begin{pmatrix} a & b \\ -b^* & a^* \end{pmatrix} \approx \prod_j \left(\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + \begin{pmatrix} 0 & F_j z^j \\ -\overline{F_j} z^{-j} & 0 \end{pmatrix} \right)$$

$$= \sum_{n \geqslant 0} \sum_{j_1 < \ldots < j_n} \begin{pmatrix} 0 & F_{j_1} z^{j_1} \\ -\overline{F_{j_1}} z^{-j_1} & 0 \end{pmatrix} \cdots \begin{pmatrix} 0 & F_{j_n} z^{j_n} \\ -\overline{F_{j_n}} z^{-j_n} & 0 \end{pmatrix} .$$

$n = 0:$	$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$	$n = 1:$	$\begin{pmatrix} 0 & \sum_j F_j z^j \\ -\sum_j \overline{F_j} z^{-j} & 0 \end{pmatrix}$
$n = 2:$	$\sum_{j_1, j_2} \begin{pmatrix} O(F)^2 & 0 \\ 0 & O(F)^2 \end{pmatrix}$	$n = 3:$	$\sum_{j_1 < j_2 < j_3} \begin{pmatrix} 0 & O(F)^3 \\ O(F)^3 & 0 \end{pmatrix}$
n even :	$\sum_{j_1 < \ldots < j_n} \begin{pmatrix} O(F)^n & 0 \\ 0 & O(F)^n \end{pmatrix}$	n odd :	$\sum_{j_1 < \ldots < j_n} \begin{pmatrix} 0 & O(F)^n \\ O(F)^n & 0 \end{pmatrix}$
$a =$	$1+$ higher order terms	$b =$	

Why call this a Fourier transform?

We do an informal computation, assuming F is "small."

$$\begin{pmatrix} a & b \\ -b^* & a^* \end{pmatrix} \approx \prod_j \left(\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + \begin{pmatrix} 0 & F_j z^j \\ -\overline{F_j} z^{-j} & 0 \end{pmatrix} \right)$$

$$= \sum_{n\geqslant 0} \sum_{j_1<\ldots<j_n} \begin{pmatrix} 0 & F_{j_1} z^{j_1} \\ -\overline{F_{j_1}} z^{-j_1} & 0 \end{pmatrix} \cdots \begin{pmatrix} 0 & F_{j_n} z^{j_n} \\ -\overline{F_{j_n}} z^{-j_n} & 0 \end{pmatrix} .$$

$n = 0:$	$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$	$n = 1:$	$\begin{pmatrix} 0 & \sum_j F_j z^j \\ -\sum_j \overline{F_j} z^{-j} & 0 \end{pmatrix}$
$n = 2:$	$\sum_{j_1, j_2} \begin{pmatrix} O(F)^2 & 0 \\ 0 & O(F)^2 \end{pmatrix}$	$n = 3:$	$\sum_{j_1<j_2<j_3} \begin{pmatrix} 0 & O(F)^3 \\ O(F)^3 & 0 \end{pmatrix}$
n even :	$\sum_{j_1<\ldots<j_n} \begin{pmatrix} O(F)^n & 0 \\ 0 & O(F)^n \end{pmatrix}$	n odd :	$\sum_{j_1<\ldots<j_n} \begin{pmatrix} 0 & O(F)^n \\ O(F)^n & 0 \end{pmatrix}$
$a =$	$1+$ higher order terms	$b =$	$\sum_j F_j z^j$ + higher order terms

The L^2 theory for nonlinear Fourier transforms

A contour integral yields

$$-\sum_{n\in\mathbb{Z}} \log(1+|F_n|^2) = \int_{\mathbb{T}} \log(1-|b|^2) + 2\sum_k \log|z_k| \, .$$

where $\{z_k\}$ are the zeros of a^* in $\mathbb{D}$.

The L^2 theory for nonlinear Fourier transforms

A contour integral yields

$$-\sum_{n\in\mathbb{Z}} \log(1+|F_n|^2) = \int_{\mathbb{T}} \log(1-|b|^2) + 2\sum_k \log|z_k|\,.$$

where $\{z_k\}$ are the zeros of a^* in $\mathbb{D}$. This is known as nonlinear Plancherel.

The L^2 theory for nonlinear Fourier transforms

A contour integral yields

$$-\sum_{n\in\mathbb{Z}} \log(1+|F_n|^2) = \int_{\mathbb{T}} \log(1-|b|^2) + 2\sum_k \log|z_k| \,.$$

where $\{z_k\}$ are the zeros of a^* in $\mathbb{D}$. This is known as nonlinear Plancherel. Using nonlinear Plancherel, one can show the NLFT extends to a bijection from $\ell^2([0,\infty))$ onto the set of pairs $(a,b) \in H^2(\mathbb{D}^*) \times H^2(\mathbb{D})$ for which

- $a^*(0) > 0$.
- $aa^* + bb^* = 1$ a.e. on $\mathbb{T}$.
- a^* and b share no common inner factor.

The L^2 theory for nonlinear Fourier transforms

A contour integral yields

$$-\sum_{n\in\mathbb{Z}} \log(1+|F_n|^2) = \int_{\mathbb{T}} \log(1-|b|^2) + 2\sum_k \log|z_k|\,.$$

where $\{z_k\}$ are the zeros of a^* in $\mathbb{D}$. This is known as nonlinear Plancherel. Using nonlinear Plancherel, one can show the NLFT extends to a bijection from $\ell^2([0,\infty))$ onto the set of pairs $(a,b) \in H^2(\mathbb{D}^*) \times H^2(\mathbb{D})$ for which

- $a^*(0) > 0$.
- $aa^* + bb^* = 1$ a.e. on $\mathbb{T}$.
- a^* and b share no common inner factor.

Similarly, the NLFT extends to a bijection from $\ell^2((-\infty,-1])$ to pairs (a,b) for which (a,b^*) is like above and b^* has a zero of order 1 at $z=0$.

The L^2 theory for nonlinear Fourier transforms

A contour integral yields

$$-\sum_{n\in\mathbb{Z}} \log(1+|F_n|^2) = \int_{\mathbb{T}} \log(1-|b|^2) + 2\sum_k \log|z_k| .$$

where $\{z_k\}$ are the zeros of a^* in $\mathbb{D}$. This is known as nonlinear Plancherel.
Using nonlinear Plancherel, one can show the NLFT extends to a bijection from $\ell^2([0,\infty))$ onto the set of pairs $(a,b) \in H^2(\mathbb{D}^*) \times H^2(\mathbb{D})$ for which

- $a^*(0) > 0$.
- $aa^* + bb^* = 1$ a.e. on $\mathbb{T}$.
- a^* and b share no common inner factor.

Similarly, the NLFT extends to a bijection from $\ell^2((-\infty,-1])$ to pairs (a,b) for which (a,b^*) is like above and b^* has a zero of order 1 at $z=0$.
Given a sequence $F \in \ell^2(\mathbb{Z})$, define its NLFT by

$$(a,b) := (a_-,b_-)(a_+,b_+)$$

where (a_-,b_-) is the NLFT of $(F_k \mathbf{1}_{k<0})_{k\in\mathbb{Z}}$, and (a_+,b_+) is the NLFT of $(F_k \mathbf{1}_{k\geqslant 0})_{k\in\mathbb{Z}}$.

The L^2 theory for nonlinear Fourier transforms

A contour integral yields

$$-\sum_{n\in\mathbb{Z}} \log(1+|F_n|^2) = \int_{\mathbb{T}} \log(1-|b|^2) + 2\sum_k \log|z_k| \, .$$

where $\{z_k\}$ are the zeros of a^* in $\mathbb{D}$. This is known as nonlinear Plancherel. Using nonlinear Plancherel, one can show the NLFT extends to a bijection from $\ell^2([0,\infty))$ onto the set of pairs $(a,b)\in H^2(\mathbb{D}^*)\times H^2(\mathbb{D})$ for which

- $a^*(0)>0$.
- $aa^*+bb^*=1$ a.e. on $\mathbb{T}$.
- a^* and b share no common inner factor.

Similarly, the NLFT extends to a bijection from $\ell^2((-\infty,-1])$ to pairs (a,b) for which (a,b^*) is like above and b^* has a zero of order 1 at $z=0$. Given a sequence $F\in\ell^2(\mathbb{Z})$, define its NLFT by

$$(a,b) := (a_-,b_-)(a_+,b_+)$$

where (a_-,b_-) is the NLFT of $(F_k\mathbf{1}_{k<0})_{k\in\mathbb{Z}}$, and (a_+,b_+) is the NLFT of $(F_k\mathbf{1}_{k\geqslant 0})_{k\in\mathbb{Z}}$. Unknown whether the NLFT is injective on $\ell^2(\mathbb{Z})$ however.

Solving the QSP problem, proof of Theorem

Initial problem: given f with $\|f\|_\infty < \frac{1}{\sqrt{2}}$, find a phase factor sequence Ψ such that the imaginary parts of the upper left entries of $U_d(x, \Psi)$ converge to f in $L^2\left(\frac{dx}{\sqrt{1-x^2}}\right)$.

Solving the QSP problem, proof of Theorem

Initial problem: given f with $\|f\|_\infty < \frac{1}{\sqrt{2}}$, find a phase factor sequence Ψ such that the imaginary parts of the upper left entries of $U_d(x, \Psi)$ converge to f in $L^2\left(\frac{dx}{\sqrt{1-x^2}}\right)$.

First step: do a change of variable.

Solving the QSP problem, proof of Theorem

Initial problem: given f with $\|f\|_\infty < \frac{1}{\sqrt{2}}$, find a phase factor sequence Ψ such that the imaginary parts of the upper left entries of $U_d(x, \Psi)$ converge to f in $L^2\left(\frac{dx}{\sqrt{1-x^2}}\right)$.

First step: do a change of variable. Namely, set $z = e^{2i\theta}$ where $\theta = \arccos(x)$, and define an even sequence F by $F_k := i \tan \psi_{|k|}$.

Solving the QSP problem, proof of Theorem

Initial problem: given f with $\|f\|_\infty < \frac{1}{\sqrt{2}}$, find a phase factor sequence Ψ such that the imaginary parts of the upper left entries of $U_d(x, \Psi)$ converge to f in $L^2\left(\frac{dx}{\sqrt{1-x^2}}\right)$.

First step: do a change of variable. Namely, set $z = e^{2i\theta}$ where $\theta = \arccos(x)$, and define an even sequence F by $F_k := i \tan \psi_{|k|}$. Then for the matrix $M = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$ we have

$$MU_d(x, \Psi)M^{-1} := \begin{pmatrix} z^{d/2} & 0 \\ 0 & z^{-d/2} \end{pmatrix} \begin{pmatrix} a_d & b_d \\ -b_d^* & a_d^* \end{pmatrix} \begin{pmatrix} z^{d/2} & 0 \\ 0 & z^{-d/2} \end{pmatrix}$$

Solving the QSP problem, proof of Theorem

Initial problem: given f with $\|f\|_\infty < \frac{1}{\sqrt{2}}$, find a phase factor sequence Ψ such that the imaginary parts of the upper left entries of $U_d(x,\Psi)$ converge to f in $L^2\left(\frac{dx}{\sqrt{1-x^2}}\right)$.

First step: do a change of variable. Namely, set $z = e^{2i\theta}$ where $\theta = \arccos(x)$, and define an even sequence F by $F_k := i\tan\psi_{|k|}$. Then for the matrix $M = \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}$ we have

$$MU_d(x,\Psi)M^{-1} := \begin{pmatrix} z^{d/2} & 0 \\ 0 & z^{-d/2}\end{pmatrix}\begin{pmatrix} a_d & b_d \\ -b_d^* & a_d^*\end{pmatrix}\begin{pmatrix} z^{d/2} & 0 \\ 0 & z^{-d/2}\end{pmatrix}$$

where (a_d, b_d) is the NLFT of the truncated sequence $(F_k \mathbf{1}_{-d\leqslant k\leqslant d})_k$.

Solving the QSP problem, proof of Theorem

Initial problem: given f with $\|f\|_\infty < \frac{1}{\sqrt{2}}$, find a phase factor sequence Ψ such that the imaginary parts of the upper left entries of $U_d(x, \Psi)$ converge to f in $L^2\left(\frac{dx}{\sqrt{1-x^2}}\right)$.

First step: do a change of variable. Namely, set $z = e^{2i\theta}$ where $\theta = \arccos(x)$, and define an even sequence F by $F_k := i \tan \psi_{|k|}$. Then for the matrix $M = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$ we have

$$MU_d(x, \Psi)M^{-1} := \begin{pmatrix} z^{d/2} & 0 \\ 0 & z^{-d/2} \end{pmatrix}\begin{pmatrix} a_d & b_d \\ -b_d^* & a_d^* \end{pmatrix}\begin{pmatrix} z^{d/2} & 0 \\ 0 & z^{-d/2} \end{pmatrix}$$

where (a_d, b_d) is the NLFT of the truncated sequence $(F_k \mathbf{1}_{-d \leqslant k \leqslant d})_k$. In particular, one gets $-ib_d(z)$ equals the imaginary part of of the upper left entry of $U_d(x, \Psi)$.

Solving the QSP problem, proof of Theorem

Initial problem: given f with $\|f\|_\infty < \frac{1}{\sqrt{2}}$, find a phase factor sequence Ψ such that the imaginary parts of the upper left entries of $U_d(x, \Psi)$ converge to f in $L^2\left(\frac{dx}{\sqrt{1-x^2}}\right)$.

First step: do a change of variable. Namely, set $z = e^{2i\theta}$ where $\theta = \arccos(x)$, and define an even sequence F by $F_k := i \tan \psi_{|k|}$. Then for the matrix $M = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$ we have

$$MU_d(x, \Psi)M^{-1} := \begin{pmatrix} z^{d/2} & 0 \\ 0 & z^{-d/2} \end{pmatrix} \begin{pmatrix} a_d & b_d \\ -b_d^* & a_d^* \end{pmatrix} \begin{pmatrix} z^{d/2} & 0 \\ 0 & z^{-d/2} \end{pmatrix}$$

where (a_d, b_d) is the NLFT of the truncated sequence $(F_k \mathbf{1}_{-d \leqslant k \leqslant d})_k$. In particular, one gets $-ib_d(z)$ equals the imaginary part of of the upper left entry of $U_d(x, \Psi)$.

New problem: given b "with symmetry" on $\mathbb{T}$ with $\|b\|_\infty < \frac{1}{\sqrt{2}}$, find an even sequence of pure imaginary nonlinear Fourier coefficients F whose truncated NLFTs (a_d, b_d) satisfy $b_d \to b$.

Solving the new problem

New problem: given b "with symmetry" on $\mathbb{T}$ with $\|b\|_\infty < \frac{1}{\sqrt{2}}$, find an even sequence of pure imaginary nonlinear Fourier coefficients F whose truncated NLFTs (a_d, b_d) satisfy $b_d \to b$.

Solving the new problem

New problem: given b "with symmetry" on $\mathbb{T}$ with $\|b\|_\infty < \frac{1}{\sqrt{2}}$, find an even sequence of pure imaginary nonlinear Fourier coefficients F whose truncated NLFTs (a_d, b_d) satisfy $b_d \to b$.

Key idea: we're only given b, meaning we can choose any a we want. Let's choose a^* outer on $\mathbb{D}$ so that $|a|^2 + |b|^2 = 1$ on $\mathbb{T}$.

Solving the new problem

New problem: given b "with symmetry" on $\mathbb{T}$ with $\|b\|_\infty < \frac{1}{\sqrt{2}}$, find an even sequence of pure imaginary nonlinear Fourier coefficients F whose truncated NLFTs (a_d, b_d) satisfy $b_d \to b$.
Key idea: we're only given b, meaning we can choose any a we want. Let's choose a^* outer on $\mathbb{D}$ so that $|a|^2 + |b|^2 = 1$ on $\mathbb{T}$.
Reduced problem: Given $(a, b) \in H^2(\mathbb{D}^*) \times L^\infty(\mathbb{T})$ such that a^* outer, $\|b\|_\infty < \frac{1}{\sqrt{2}}$, and b "has symmetry", find an even sequence of pure imaginary nonlinear Fourier coefficients F whose truncated NLFTs (a_d, b_d) converge to (a, b).

Solving the new problem

New problem: given b "with symmetry" on $\mathbb{T}$ with $\|b\|_\infty < \frac{1}{\sqrt{2}}$, find an even sequence of pure imaginary nonlinear Fourier coefficients F whose truncated NLFTs (a_d, b_d) satisfy $b_d \to b$.

Key idea: we're only given b, meaning we can choose any a we want. Let's choose a^* outer on $\mathbb{D}$ so that $|a|^2 + |b|^2 = 1$ on $\mathbb{T}$.

Reduced problem: Given $(a, b) \in H^2(\mathbb{D}^*) \times L^\infty(\mathbb{T})$ such that a^* outer, $\|b\|_\infty < \frac{1}{\sqrt{2}}$, and b "has symmetry", find an even sequence of pure imaginary nonlinear Fourier coefficients F whose truncated NLFTs (a_d, b_d) converge to (a, b).

Soln: Use first assumption to set up a fixed point mapping to solve for the Fourier coefficients. The second assumption implies the fixed point mapping is a contraction mapping, so there exist unique Fourier coefficients $F \in \ell^2(\mathbb{Z})$ corresponding to (a, b).

Solving the new problem

New problem: given b "with symmetry" on $\mathbb{T}$ with $\|b\|_\infty < \frac{1}{\sqrt{2}}$, find an even sequence of pure imaginary nonlinear Fourier coefficients F whose truncated NLFTs (a_d, b_d) satisfy $b_d \to b$.
Key idea: we're only given b, meaning we can choose any a we want. Let's choose a^* outer on $\mathbb{D}$ so that $|a|^2 + |b|^2 = 1$ on $\mathbb{T}$.
Reduced problem: Given $(a, b) \in H^2(\mathbb{D}^*) \times L^\infty(\mathbb{T})$ such that a^* outer, $\|b\|_\infty < \frac{1}{\sqrt{2}}$, and b "has symmetry", find an even sequence of pure imaginary nonlinear Fourier coefficients F whose truncated NLFTs (a_d, b_d) converge to (a, b).
Soln: Use first assumption to set up a fixed point mapping to solve for the Fourier coefficients. The second assumption implies the fixed point mapping is a contraction mapping, so there exist unique Fourier coefficients $F \in \ell^2(\mathbb{Z})$ corresponding to (a, b).
Then argue F is even and pure imaginary because b is "symmetric."

Thank you for Listening!