Mathematisches Forschungsinstitut Oberwolfach

Report No. 46/2024

DOI: 10.4171/OWR/2024/46

Arbeitsgemeinschaft: Quantum Signal Processing and Nonlinear Fourier Analysis

Organized by
András Gilyén, Budapest
Lin Lin, Berkeley
Christoph Thiele, Bonn

6 October – 11 October 2024

Abstract. A pivotal algorithm in quantum signal processing represents a real-valued function as a specific entry in a product of $SU(2)$ matrices, where an argument matrix alternates with a sequence of coefficient matrices. Recently, this framework has been identified as a nonlinear Fourier series, a concept studied across various mathematical disciplines. This Arbeitsgemeinschaft explored the intersection of quantum computing and harmonic analysis, emphasizing the potential of this connection.

Mathematics Subject Classification (2020): 68Q12, 81P68, 34L25, 42C99.

Introduction by the Organizers

The goal of this Arbeitsgemeinschaft was to investigate the interplay between quantum computing and harmonic analysis, inspired by the recent discovery that a key algorithm in quantum signal processing can be understood as a nonlinear Fourier series. This insight not only improved algorithms in quantum signal processing but also sparked new questions in the field of nonlinear Fourier analysis.

The Arbeitsgemeinschaft primarily gathered young researchers from a wide array of disciplines, including physics and computer science with an emphasis on quantum computing, as well as classical mathematical fields such as harmonic analysis and scattering theory. A predefined list of topics was distributed among the participants, with each individual assigned as either the primary speaker or a backup speaker for one of the topics.

The first day featured a review of the fundamental theory of quantum computing, with Shor's algorithm serving as a highlight. On the second day, the focus shifted to an overview of nonlinear Fourier analysis, emphasizing the foundational $SU(2)$ and $SU(1,1)$ models. Particular attention was given to the mapping properties of these Fourier series, especially the nuanced L^2 theory governed by a Plancherel identity. This segment concluded with the presentation "QSP and NLFT," which elaborated the connections between quantum signal processing and nonlinear Fourier analysis.

The second half of the week delved deeper into advanced quantum signal processing algorithms and concluded with a survey of classical mathematical topics related to nonlinear Fourier analysis. These included Schur's algorithm, orthogonal polynomials, and the application of the inverse scattering method to solve integrable systems.

The program also included a problem session after dinner, during which participants proposed open problems. Even during the Arbeitsgemeinschaft, some groups began to form around new research questions in the field.

Overall, the participants found the Arbeitsgemeinschaft highly stimulating, thanks in part to the inspiring atmosphere at Oberwolfach and the outstanding support provided by the staff.

Acknowledgement: The MFO and the workshop organizers would like to thank the National Science Foundation for supporting the participation of junior researchers in the workshop by the grant DMS-1641185, “US Junior Oberwolfach Fellows”.

Arbeitsgemeinschaft: Quantum Signal Processing and Nonlinear Fourier Analysis

Table of Contents

Abstracts

A variational nonlinear Hausdorff-Young inequality in the discrete setting

Michel Alexis

Following [1], we show a variational Hausdorff-Young inequality for the $SU(1,1)$-valued nonlinear Fourier transform (NLFT).

Recall that given a sequence F in the complex unit disk $\mathbb{D}$, we (informally) define the NLFT (a,b) of F by

$$\begin{pmatrix} a(z) & b(z) \\ \overline{b(z)} & \overline{a(z)} \end{pmatrix} := \prod_k \frac{1}{\sqrt{1-|F_k|^2}} \begin{pmatrix} 1 & F_k z^k \\ \overline{F_k} z^{-k} & 1 \end{pmatrix} . \tag{1}$$

This is a map sending points z in the unit circle $\mathbb{T}$ to elements of the matrix group

$$SU(1,1) := \left\{ \begin{pmatrix} \alpha & \beta \\ \overline{\beta} & \overline{\alpha} \end{pmatrix} \ : \ |\alpha|^2 - |\beta|^2 = 1 \right\} ,$$

whose elements have operator norm given by

$$\left\| \begin{pmatrix} \alpha & \beta \\ \overline{\beta} & \overline{\alpha} \end{pmatrix} \right\|_{\mathrm{op}} = |\alpha| + |\beta| .$$

For each $z \in \mathbb{T}$, we visualize truncations of the above NLFT product as a curve γ in $SU(1,1)$ beginning at the identity matrix and terminating at the NLFT of F. Namely, for each integer time N, define

$$\gamma(N;z) := \prod_{k \leq N} \frac{1}{\sqrt{1-|F_k|^2}} \begin{pmatrix} 1 & F_k z^k \\ \overline{F_k} z^{-k} & 1 \end{pmatrix} ,$$

and

$$\gamma(-\infty;z) := \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} , \qquad \gamma(+\infty;z) := \begin{pmatrix} a(z) & b(z) \\ b^*(z) & a^*(z) \end{pmatrix} .$$

Given $1 \leq r < \infty$, define the r-variation of such a curve γ in $SU(1,1)$ as

$$\mathcal{V}_r(\gamma)(z) := \sup_K \sup_{N_1 < \ldots < N_K} \left(\sum_{j=1}^{K-1} d(\gamma(N_j), \gamma(N_{j+1}))^r \right)^{\frac{1}{r}} ,$$

where the metric d given by

$$d(X,Y) := \log\bigl(1 + \|X^{-1}Y - 1\|_{\mathrm{op}}\bigr) .$$

We present the main theorem of [1], a variational Hausdorff-Young inequality for the nonlinear Fourier transform.

Theorem 1 ([1]). *Let $1 \le p < r < 2$. There exists a constant $C_{p,r} < \infty$ such that*

$$\|\mathcal{V}_r(\gamma)\|_{L^{p'}(\mathbb{S})} + \|\mathcal{V}_r(\gamma)\|^{\frac{1}{r}}_{L^{\frac{p'}{r}}(\mathbb{S})} \le C_{p,r} \left\| \log\left(\frac{1+|F_n|}{1-|F_n|}\right)\right\|_{\ell^p(\mathbb{Z})}, \tag{2}$$

where $\mathbb{S}$ is the set of points $z \in \mathbb{T}$ with small variation, i.e., $\mathcal{V}_r(\gamma(z)) \le 1$.

Compare this theorem to its linear analogue

$$\|\mathcal{V}_r(\sigma)\|_{L^{p'}(\mathbb{T})} \le C_{p,r} \|F\|_{\ell^p(\mathbb{Z})}, \tag{3}$$

where the curve

$$\sigma(N;z) := \begin{pmatrix} 0 & \sum\limits_{k\le N} F_k z^k \\ \sum\limits_{k\le N} \overline{F_k} z^{-k} & 0 \end{pmatrix},$$

is an embedding of the truncated linear Fourier series in the Lie algebra $\mathfrak{su}(1,1)$ of $SU(1,1)$, and its r-variation is given by

$$\mathcal{V}_r(\sigma)(z) := \sup_K \sup_{N_1<\ldots<N_K} \left(\sum_{j=1}^{K-1} \|\sigma(N_{j+1};z) - \sigma(N_j;z)\|^r\right)^{\frac{1}{r}}.$$

Recalling the nonlinear Fourier transform can be viewed as an exponentiation of the linear Fourier series, we point out the following perturbative heuristic: the nonlinear theorem should morally follow from the linear theorem when the potential F is "small" in an appropriate sense. And [1] proves (2) for *all* potentials using precisely this philosphy: indeed, [1] is able to deduce (2) from (3) by showing the nonlinear variation is controlled by the linear variation and a "big jumps"-term, i.e.,

$$\mathcal{V}_r(\gamma)(z) \le \mathcal{V}_r(\sigma)(z) + C_r\left(\min\{\mathcal{V}_r(\sigma)(z)^r, \mathcal{V}_r(\sigma)(z)^2\} + \|F\|^{r-1}_{\ell^r(\mathbb{Z})}\left\|\log\frac{1+|F_n|}{1-|F_n|}\right\|_{\ell^r(\mathbb{Z}}\right).$$

The proof of this estimate is broken into 3 steps below.

Step 1: The proof of [1] begins by making rigorous the perturbative heurtistic by showing that when the potential F has small linear variation, then the logarithm of the nonlinear series and the linear series are are essentially equal. More precisely, they show there exist constants $C_r \ge 1$ and $\delta > 0$ such that whenever $\mathcal{V}_r\left(\sigma_{[M,N]}(z)\right) \le \delta$, then

$$\|\log\left(\gamma_{[M,N]}(z)\right) - \sigma_{[M,N]}(z)\|_{\mathrm{op}} \le C_r \mathcal{V}_r\left(\sigma_{[M,N]}(z)\right)^2, \tag{4}$$

where $\gamma_{[M,N]}$ and $\sigma_{[M,N]}$ denote the nonlinear and linear curves associated to the truncated potential $(F_n \mathbf{1}_{\{M\le n\le N\}})_n$. To show (4), the author first proves it when the linear variation is smaller than some δ depending on the individual potential F (and depending monotonically on $|N-M|$). But by then breaking up the potential into pieces where the variation is less than δ_F (and dealing with big jumps separately), [1] bootstraps the initial potential-dependent estimate to show that the estimate actually holds for some δ *independent* of the potential F.

Step 2: Using (4), the author then shows that the nonlinear and linear variations are equal up to first order in the perturbative case, i.e.,

$$\mathcal{V}_r(\gamma) = \mathcal{V}_r(\sigma) + O(\mathcal{V}_r(\sigma)^2)$$

when $\mathcal{V}_r(\sigma) \leq \delta$. They prove this using (4) and Taylor expansions of the logarithms and exponentials of matrices to get good second order estimates.

Step 3: This last step is done via divide and conquer. Because the case of small variations is handled by Step 2, the author only considers large variations. In this case, the author breaks up the curve γ into pieces consisting of big jump singletons, and pieces with no big jumps but with small variation. The contribution of any individual big jump singleton is directly estimated by

$$\| \log \frac{1 + |F_n|}{1 - |F_n|} \|_{\ell^r} ,$$

and the number of these jumps is estimated by $\|F\|_{\ell^r}^{r-1}$ using Chebyshev's inequality. As for the pieces without big jumps but with small variation, we apply Step 2, while keeping track that the number of these pieces is also controlled by the linear variation.

References

[1] E Silva, D. Oliveira, *A variational nonlinear Hausdorff-Young inequality in the discrete setting*, Preprint (2017) `ArXiv:1704.00688` .

Orthogonal polynomials and Geronimus's theorem

Cade Ballew

This talk is based on [1].

A Schur function $f(z)$ is an analytic function defined in the open unit disk $\mathbb{D} = \{z \in \mathbb{C} : |z| < 1\}$ such that $|f(z)| \leq 1$ for all $z \in \mathbb{D}$. For a given Schur function, Schur's algorithm generates a sequence of Schur functions and so-called Schur parameters. It is defined by the recurrence

$$f_0 = f, \quad f_{n+1} = \frac{f_n(z) - f_n(0)}{z(1 - \overline{f_n(0)} f_n(z))}, \quad n \in \mathbb{N}.$$

Provided that f is not a finite Blaschke product, this algorithm generates an infinite sequence of Schur functions $\{f_n\}_{n=0}^{\infty}$ and Schur parameters $\{\gamma_n\}_{n=0}^{\infty}$ where $\gamma_n = f_n(0)$ satisfies $|\gamma_n| < 1$. It turns out that given any sequence $\{\gamma_n\}_{n=0}^{\infty} \subset \mathbb{C}$ such that $|\gamma_n| < 1$, there exists a unique Schur function with these Schur parameters.

A Carathéodory function F is an analytic function defined in the open unit disk $\mathbb{D} = \{z \in \mathbb{C} : |z| < 1\}$ such that $\mathrm{Re} F(z) \geq 0$ for all $z \in \mathbb{D}$. Given a Schur function f with Schur parameters $\{\gamma_n\}_{n=0}^{\infty}$, define

$$F(z) = \frac{1 + zf(z)}{1 - zf(z)}. \tag{1}$$

For $z \in \mathbb{D}$

$$\mathrm{Re}F(z) = \frac{1 - |zf(z)|^2}{|1 - zf(z)|^2} > 0,$$

and $F(z)$ is analytic, so F defines an associated Carathéodory function. By the Herglotz representation theorem, for any Carathéodory function F such that $F(0) = 1$, there exists some Borel probability measure $\frac{1}{2\pi}\mathrm{d}\sigma$ defined on $[0, 2\pi)$ such that

$$F(z) = \frac{1}{2\pi} \int_0^{2\pi} \frac{\mathrm{e}^{\mathrm{i}\theta} + z}{\mathrm{e}^{\mathrm{i}\theta} - z} \mathrm{d}\sigma(\theta). \tag{2}$$

For a Carathéodory function F associated to a Schur function f, The support of $\mathrm{d}\sigma$ is infinite except in the case where f is a finite Blaschke product. Conversely, given any Borel probability measure $\frac{1}{2\pi}\mathrm{d}\sigma$ defined on $[0, 2\pi)$, there exists some Carathéodory function F satisfying (2) and therefore an associated Schur function f satisfying (1).

On the other hand, orthogonal polynomials on the unit circle can be defined for a Borel probability measure $\frac{1}{2\pi}\mathrm{d}\sigma$ on $[0, 2\pi)$. To ensure that an infinite sequence of orthogonal polynomials exists, we assume that the support of $\mathrm{d}\sigma$ is an infinite set. Consider the inner product

$$\langle f, g \rangle_\sigma = \int_0^{2\pi} f\left(\mathrm{e}^{\mathrm{i}\theta}\right) \overline{g\left(\mathrm{e}^{\mathrm{i}\theta}\right)} \mathrm{d}\sigma(\theta),$$

defined for functions of the unit circle $\mathbb{U} = \{z \in \mathbb{C} : |z| = 1\}$. Given the restriction that $\varphi_n(z) = \chi_n z^n + \ldots$ where $\chi_n > 0$, there exists a unique system of orthonormal polynomials $\{\varphi_n\}_{n=0}^\infty$ on $\mathbb{U} = \{z \in \mathbb{C} : |z| = 1\}$ such that

$$\langle \varphi_n, \varphi_m \rangle_\sigma = \delta_{n,m},$$

for all $n, m \in \mathbb{N}$, where $\delta_{n,m}$ is the Kronecker delta. Such polynomials can be generated through, say, the Gram–Schmidt algorithm applied to the set $\{\diamond^n\}_{n=0}^\infty$, starting of course from $n = 0$. The monic orthogonal polynomials $\{\Phi_n\}_{n=0}^\infty$ are defined by normalizing the orthonormal polynomials to have leading coefficient 1. That is,

$$\Phi_n(z) = \frac{1}{\chi_n} \varphi_n(z) = z^n + \ldots, \quad n \in \mathbb{N}.$$

The reverse polynomials (and more broadly the * operation) are defined by reversing and conjugating the orthogonal polynomial coefficients. That is,

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

The monic orthogonal polynomials and their reverse satisfy a the following pair of recurrence formulae:

$$\begin{aligned} \Phi_{n+1}(z) &= z\Phi_n(z) - \overline{a}_n \Phi_n^*(z), \quad n \in \mathbb{N}, \\ \Phi_{n+1}^*(z) &= \Phi_n^*(z) - a_n z \Phi_n(z), \quad n \in \mathbb{N}, \end{aligned} \tag{3}$$

where $a_n = -\overline{\Phi_{n+1}(0)}$. The parameters $\{a_n\}_{n=0}^\infty$ are known as Verblunsky coefficients, and $|a_n| < 1$ for all $n \in \mathbb{N}$. Note that $\Phi_0 = \Phi_0^* = 1$, so the monic orthogonal polynomials and their reverse are uniquely generated by their Verblunsky coefficients. It turns out that this construction goes both ways. Favard's theorem says that given any sequence $\{a_n\}_{n=0}^{\infty} \subset \mathbb{C}$ such that $|a_n| < 1$ for all $n \in \mathbb{N}$, there exists a Borel probability measure $\frac{1}{2\pi}\mathrm{d}\sigma$ on $[0, 2\pi)$ such that the corresponding system of orthogonal polynomials $\{\Phi_n\}_{n=0}^{\infty}$ satifies (3) and $a_n = -\overline{\Phi_{n+1}(0)}$ for all $n \in \mathbb{N}$.

The connection between Schur functions and orthogonal polynomials on the unit circle is the following:

Theorem 1 (Geronimus). *It holds that $a_n = \gamma_n$ for all $n \in \mathbb{N}$.*

We will prove this theorem. In a sense, it says that Schur functions and orthogonal polynomials on the unit circle are equivalent. Through the theory presented above, both Schur functions and orthogonal polynomials on the unit circle are associated with a set of parameters and a probability measure which can each be derived from the other. Geronimus's theorem tells us that when the measures agree, so do the parameters and vice versa. This allows us to appeal to the theory of orthogonal polynomials to derive properties of Schur functions, yielding theorems that guarantee decay rates of Schur parameters given smoothness properties of their associated Schur functions and vice versa. We will discuss (and prove if time permits) the following theorems via orthogonal polynomial theory.

Theorem 2. *Let f be a Schur function with boundary values $f(\mathrm{e}^{\mathrm{i}\theta})$. It is said to be regular if its boundary values are continuous and $\sup_{\theta\in\mathbb{R}} |f(\mathrm{e}^{\mathrm{i}\theta})| < 1$. If f is regular and*

$$\sum_{n=1}^{\infty} \frac{1}{\sqrt{n}} \sup_{0<\tau<\frac{1}{n}} \sup_{\theta\in\mathbb{R}} |f(\mathrm{e}^{\mathrm{i}(\theta+\tau)}) - f(\mathrm{e}^{\mathrm{i}\theta})| < \infty,$$

then its associated Schur parameters satisfy

$$\sum_{n=0}^{\infty} |\gamma_n| < \infty.$$

Conversely, if $\{\gamma_n\}_{n=0}^{\infty}$ are absolutely summable, then their associated Schur function is regular.

Theorem 3. *Schur coefficients $\{\gamma_n\}_{n=0}^{\infty}$ satisfy*

$$\limsup_{n\to\infty} |\gamma_n|^{1/n} < 1,$$

if and only if their associated Schur function is analytic in a region containing the closed unit disk $\overline{\mathbb{D}}$ and $\sup_{z\in\overline{\mathbb{D}}} |f(z)| < 1$.

References

[1] L. B. Golinskii. Schur Functions, Schur Parameters and Orthogonal Polynomials on the Unit Circle *Zeitschrift für Analysis und ihre Anwendungen*, 12(3):457–469, 1993.

The postulates of quantum computing

Tiklung Chan

We discuss the basics of quantum computing based on chapter 1 of Ronald de Wolf's notes [1]. Classical computing is based on classical physics, including the important notions of locality and that systems can only exist in one state at a time, and bits, the basic objects of study, behave accordingly by taking on one of two states (0 or 1) and only changing when acted upon by a classical operation. On the other hand, quantum computing is based on quantum physics, which allows for nonlocal operations and superpositions of states, and qubits reflect these differences by taking on a superposition of states (0 and 1) and can be operated on by more complex operations. This leads to a richer theory of computing - in particular, quantum computing algorithms can work much faster and accomplish more complicated tasks than classical algorithms, for example Shor's algorithm for integer factorization. With this being said, much of the work on quantum computing is still theoretical as there are still many serious obstacles to effectively constructing physical quantum computers.

First, we introduce the notation and basic ideas of quantum mechanics as they relate to quantum computing. We use Dirac's bra-ket notation where a "bra" $\langle\cdot|$ represents a $1 \times n$ row vector and a "ket" $|\cdot\rangle$ represents an $n \times 1$ column vector. The point is that a "braket" ("bracket") correctly represents the inner product of two vectors. We will be intentionally ambiguous about exactly what we put in place of the $\cdot$'s as we will often have to deal with complicated states and may need to abuse notation to simplify computations.

Typically we will use kets to represent states - for example, if we have N states we may denote them by $|0\rangle$, $|1\rangle$, ..., and $|N-1\rangle$ which are orthonormal basis vectors in some N-complex-dimensional Hilbert space. Clasically, a system would only exist in one of these states at a given time but as we allow for superpositions in quantum mechanics, the state of a quantum system would be represented by:

$$|\Phi\rangle = \sum_{n=0}^{N-1} \alpha_n |n\rangle$$

Per the rules of quantum mechanics, the complex amplitudes $\alpha_n \in \mathbb{C}$ must represent a probability distribution in the sense that $\sum_{n=0}^{N-1} |\alpha_n|^2 = 1$. When we have multiple systems, we represent the state by the tensor product of these vectors. In particular, recall that if $\{|0\rangle, ..., |N-1\rangle\}$ is an orthonormal basis of the Hilbert space $\mathcal{H}_A$ and $\{|0\rangle, ..., |M-1\rangle\}$ is an orthonormal basis of the Hilbert space $\mathcal{H}_B$ then $\{|0\rangle \otimes |0\rangle, ..., |N-1\rangle \otimes |M-1\rangle\}$ is an orthonormal basis of the NM-dimensional Hilbert space $\mathcal{H}_A \otimes \mathcal{H}_B$.

In quantum mechanics there are two basic operations that can be applied to a state. We can either measure the state, which yields a single classical state $|n\rangle$ with probability $|\alpha_n|^2$, or we can unitarily evolve the state, which yields a new quantum state. By the laws of quantum mechanics, only linear operations are allowed, i.e. matrix multiplication, and the probability must be preserved, i.e. unitary matrix multiplication.

For the purposes of computing, we will focus on systems of qubits where the state of each qubit is represented by a vector in a 2-complex-dimensional Hilbert space where the basis vectors are represented by $|0\rangle$ and $|1\rangle$. For shorthand, we will represent the tensor product of n of these basis vectors by:

$$|b_1\rangle \otimes |b_2\rangle ... \otimes |b_n\rangle = |b_1\rangle |b_2\rangle ... |b_n\rangle = |b_1 b_2 ... b_n\rangle$$

depending on the context, where $b_i \in \{0, 1\}$. It can also be useful to instead represent each of these basis vectors by an integer in $\{0, ..., 2^n - 1\}$. A key phenomenon in quantum computing is quantum entanglement, where the probabilities of each qubit being in state 0 or 1 are entangled with each other in the sense that as soon as one qubit is measured and it collapses into an observable classical state, the state of another qubit is immediately known as well. An example of this is an Einstein-Podolsky-Rosen (EPR) pair:

$$|\Phi\rangle = \frac{1}{\sqrt{2}} |00\rangle + \frac{1}{\sqrt{2}} |11\rangle$$

Here, if the first qubit is measured and found to be in the 0-state then the second qubit is immediately known to be in the 0-state as well (and similarly for the 1-state). These pairs are named after Einstein, Podolsky, and Rosen who studied their properties extensively. Formally, the state of a 2-qubit system is "entangled" if it cannot be writen as a tensor product $|\Phi_A\rangle \otimes |\Phi_B\rangle$.

We can then define and use gates, which are named in analogy with the same notion in classical computing. We will focus on gates used for systems of a small number of qubits (say 2 or 3 qubits). Each gate is a unitary matrix which acts upon quantum systems in particular ways. For example, the NOT gate which negates the state of a 1-qubit system is represented by the matrix:

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

There are many interesting gates which we will discuss in more detail.

Finally, we explore some of the possibilities and impossibilities of quantum computing. First, we observe that qubits cannot be cloned nor can they be deleted through these unitary operations:

Theorem 1 (No cloning). *There is no unitary U which maps any $|\phi\rangle$:*

$$U : |\phi\rangle |0\rangle \mapsto |\phi\rangle |\phi\rangle$$

Theorem 2 (No deleting). *There is no unitary U which maps any $|\phi\rangle$:*

$$U : |\phi\rangle \mapsto |0\rangle$$

We also note (but do not prove) Holevo's theorem which states that by sending one qubit, only one bit of information can be sent. However, through clever usage of quantum entanglement, we describe the processes of quantum teleportation - which allows for the information of a qubit to be sent across space via two classical bits - and superdense coding - which allows for the information of two classical bits to be sent across space via one qubit. These combine many of the basic ideas and principles of quantum computing described above, demonstrating some of the many fascinating new possibilities of quantum computing as compared to classical computing.

References

[1] R. de Wolf, *Quantum Computing: Lecture Notes*, https://arxiv.org/abs/1907.09415

The fast and the faster quantum Fourier transform

Jaume de Dios Pont

1. The Classical Discrete Fourier Transform

The discrete Fourier transform, or the Fourier transform in $\mathbb{Z}/N\mathbb{Z}$ is the natural discretization of the Fourier transform on $\mathbb{R}$. It is a fundamental tool used in various fields of classical computing, such as signal processing, data compression, and complexity theory. It is an operator acting on functions from $\mathbb{Z}/N\mathbb{Z} \to \mathbb{C}$, or equivalently, a matrix F_N acting on $N-$dimensional vectors. We will take this later perspective.

Let $\omega_N = e^{2\pi i/N}$ be the N-th root of unity, which satisfies $\omega_N^N = 1$. The (j,k)-th entry of the matrix F_N is given by:

$$(F_N)_{j,k} = \frac{1}{\sqrt{N}} \omega_N^{jk}, \quad j,k \in \{0,\dots,N-1\}.$$

The matrix F_N is unitary, which means its columns are orthogonal and have a norm of 1:

$$\frac{1}{N} \sum_{j=0}^{N} (F_N)_{j,k} \overline{(F_N)_{j,l}} = \frac{1}{N} \sum_{j=0}^{N} w_N^{(j-l)k} = \delta_{k,l} \tag{1}$$

This property ensures that the inverse of F_N is its conjugate transpose, $F_N^{-1} = F_N^*$, differing only in the sign of the exponent. Given a vector $v \in \mathbb{R}^N$, its Fourier transform $\hat{v}$ is computed as:

$$\hat{v}_j = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \omega_N^{jk} v_k.$$

4.2 The Fast Fourier Transform

The naive way of computing the Fourier transform of a vector $v \in \mathbb{R}^N$ using matrix multiplication takes $O(N^2)$ operations. However, there is a more efficient algorithm called the Fast Fourier Transform (FFT), developed by Cooley and Tukey in 1965 [1], which reduces the time complexity to $O(N \log N)$, whenever N is a power of 2.

The key idea behind the FFT is to divide the computation of the Fourier transform into two smaller Fourier transforms: one for the even-indexed elements and one for the odd-indexed elements. For $N = 2^n$, we can express the Fourier transform $\hat{v}_j$ as:

$$\hat{v}_j = \frac{1}{\sqrt{N}} \left(\sum_{\text{even } k} \omega_N^{jk} v_k + \omega_N^j \sum_{\text{odd } k} \omega_N^{j(k-1)} v_k \right).$$

This recursion continues by dividing the problem into smaller Fourier transforms of size $N/2$, allowing the computation to be done in $O(N \log N)$ steps. The last insight is that these sums are periodic

$$\frac{1}{\sqrt{N}} \sum_{\text{even } k} \omega_N^{jk} v_k = \frac{1}{\sqrt{N}} \sum_{\text{even } k} \omega_N^{(j+N/2)k} v_k$$

and just have to be completed half of the time. In fact, there are a $N/2$-dimensional Fourier transform. Define the following operations for v a $N = 2^n$-dimensional vector:

$$\begin{cases} \text{ODD}_n(v) = [v_1, v_3, \ldots, v_{2^n-1}] \\ \text{EVEN}_n(v) = [v_0, v_2, \ldots, v_{2^n-2}] \\ \text{COPY}_n(v) = [v_0, v_1 \ldots, v_{2^n-1}, v_0, v_1 \ldots, v_{2^n-1}] \end{cases} \tag{2}$$

Then we can rewrite the formula above as

$$\begin{aligned} \text{FFT}_n(v) = & \qquad \text{COPY}_{n-1}(\text{FFT}_{n-1}(\text{EVEN})(v)) \\ & + (w_N^j)_{j=0}^{2^n-1} \odot \text{COPY}_{n-1}(\text{FFT}_n(\text{ODD})(v)) \end{aligned}$$

This allows one to define the FFT recursively.

Complexity: The cost of this recursive computation FFT_n is $\approx 2^n {*} n = N \log_2 N$: The recursion goes on for n steps. At step k we have split v into 2^k pieces using EVEN and ODD k times, and each piece has length 2^{n-k}. For each piece we will have to do operations (addition/multiplication) that take 2^{n-k} time, for a total of $2^k \cdot 2^{n-k} = 2^n$ operations per recursion step. For n steps, we do $n2^n$ operations.

2. The Quantum Discrete Fourier Transform

Let H_n be the 2^n-dimensional Hilbert space, or the Hilbert space on n qubits. When $n = 1$, it contains the qbits: $|0\rangle, |1\rangle$. For $n > 1$, and $0 \leq k < 2^n$ with binary expansion $([k]_{n-1}, \ldots, [k]_1)$ (so that $k = \sum_{j=1}^{n} [k]_j 2^{jj}$), we denote by $|k\rangle = |[k]_1\rangle \otimes \cdots \otimes |[k]_1\rangle$. The discrete Fourier transform for vectors of length 2^{2^n}, $v_k \mapsto \hat{v}_k$ is unitary, and thus the transformation

$$\sum_{k \in [2^n]} v_k |k\rangle \mapsto \sum_{k \in [2^n]} \hat{v}_k |k\rangle \tag{3}$$

is a unitary operation in $Q-$bits. This operation is called the QFT. Unlike its quantum counterpart, it can be executed in $n = \log N$ steps. Using a recursive decomposition as before, we see that

$$\begin{aligned} F_N|k\rangle =& \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} e^{2\pi i jk/N}|j\rangle \\ =& \frac{1}{\sqrt{2^n}} \sum_{j\in\{0,1\}^n} e^{2\pi i\left(\sum_{\ell=1}^{n} j_\ell 2^{-\ell}\right)k}|j_1\dots j_n\rangle \\ =& \bigotimes_{\ell=1}^{n} \frac{1}{\sqrt{2}}\left(|0\rangle + e^{2\pi i k/2^\ell}|1\rangle\right). \end{aligned}$$

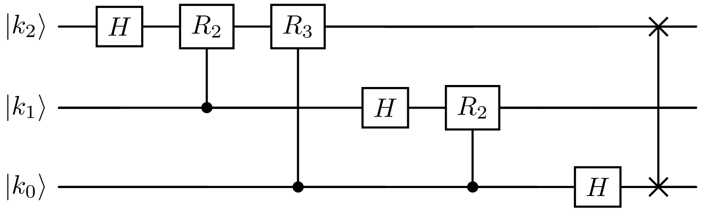
As an explicit example,

$$F_8|k\rangle = (|0\rangle + e^{2\pi i 0.k_0}|1\rangle) \otimes (|0\rangle + e^{2\pi i 0.k_1k_0}|1\rangle) \otimes (|0\rangle + e^{2\pi i 0.k_2k_1k_0}|1\rangle).$$

The first term of this product ($l = 1$) only depends on whether k is even or odd, i.e. it only depends on the least significant bit of k. The second term depends on the two least significant bits, and so on. From here, the state $F_8|k\rangle$ can be build in stages as

$$\begin{array}{ccccc} |k_0\rangle & \otimes & |k_1\rangle & \otimes & |k_2\rangle \\ |k_0\rangle & \otimes & |k_1\rangle & \otimes & (|0\rangle + e^{2\pi i 0.k_2}|1\rangle) \\ |k_0\rangle & \otimes & |k_1\rangle & \otimes & (|0\rangle + e^{2\pi i 0.k_2k_1k_0}|1\rangle) \\ |k_0\rangle & \otimes & (|0\rangle + e^{2\pi i 0.k_1}|1\rangle) & \otimes & (|0\rangle + e^{2\pi i 0.k_2k_1k_0}|1\rangle) \\ |k_0\rangle & \otimes & (|0\rangle + e^{2\pi i 0.k_1k_0}|1\rangle) & \otimes & (|0\rangle + e^{2\pi i 0.k_2k_1k_0}|1\rangle) \\ (|0\rangle + e^{2\pi i 0.k_0}|1\rangle) & \otimes & (|0\rangle + e^{2\pi i 0.k_1k_0}|1\rangle) & \otimes & (|0\rangle + e^{2\pi i 0.k_2k_1k_0}|1\rangle) \end{array}$$

which becomes a quantum circuit of the form



In general, this procedure requires $n = \log N$ Hadamard gates, and $\frac{n^2-n}{2}$ controlled rotation gates. An approximated version (which drops the small rotations) can be built with n gates and approximation error n^{-1}

References

[1] J. W. Cooley and J. W. Tukey, "An algorithm for the machine calculation of complex Fourier series," *Mathematics of Computation*, vol. 19, no. 90, pp. 297–301, 1965.

Quantum Signal Processing

João F. Doriguello

Quantum signal processing (QSP) [10, 7] is an algorithmic primitive that allows to efficiently transform the eigenvalues of a Hermitian matrix using some given polynomial $P \in \mathbb{C}[x]$. More specifically, QSP is a framework that allows the construction of an operator $\mathsf{S}_P = \sum_\theta P(\mathrm{e}^{\mathrm{i}\theta})|\theta\rangle\langle\theta|$ from a simpler unitary $\mathsf{U} = \mathrm{e}^{\mathrm{i}\theta}|\theta\rangle\langle\theta|$. The transformation requires one or two ancillary qubits and applies a sequence of control-U and control-$\mathsf{U}^\dagger$ interspersed by single-qubit gates on the control a number of times approximately equal to the degree of P. QSP thus provides an efficiently implementable recipe to transform $\mathrm{e}^{\mathrm{i}\theta} \mapsto P(\mathrm{e}^{\mathrm{i}\theta})$ in superposition, which can also be used for a general function f with an approximating polynomial with small degree. QSP produces gate-efficient quantum algorithms for several important problems [4], e.g., Hamiltonian simulation [3, 1, 2, 5] and linear system of equations [6, 8, 4].

There are a few different but equivalent approaches to QSP. In the following, let X, Y, Z be the usual Pauli matrices. Following [9], let $x \in [-1, 1]$ and consider the rotation matrix through an angle $\arccos x$,

$$\mathsf{O}(x) = \mathsf{U}(x)\mathsf{Z} = \begin{pmatrix} x & -\sqrt{1-x^2} \\ \sqrt{1-x^2} & x \end{pmatrix}, \quad \text{where } \mathsf{U}(x) = \begin{pmatrix} x & \sqrt{1-x^2} \\ \sqrt{1-x^2} & -x \end{pmatrix}$$

is the one-qubit Hermitian block encoding of x. The main idea of QSP is to find angles $\Phi = (\phi_0, \phi_1, \dots, \phi_d) \in \mathbb{R}^{d+1}$ such that the representation

$$\text{(1)} \qquad \mathsf{S}_\Phi(x) = \mathrm{e}^{\mathrm{i}\phi_0\mathsf{Z}}\mathsf{O}(x)\mathrm{e}^{\mathrm{i}\phi_1\mathsf{Z}}\mathsf{O}(x)\cdots\mathrm{e}^{\mathrm{i}\phi_{d-1}\mathsf{Z}}\mathsf{O}(x)\mathrm{e}^{\mathrm{i}\phi_d\mathsf{Z}} = \mathrm{e}^{\mathrm{i}\phi_0\mathsf{Z}}\prod_{k=1}^{d}\mathsf{O}(x)\mathrm{e}^{\mathrm{i}\phi_k\mathsf{Z}}$$

reconstruct a polynomial $P \in \mathbb{C}[x]$ of interest in one of its entries. That such a procedure is possible is proven in Theorem 1 below. We briefly mention that it is possible to consider different representations, e.g., Gilyén et al. [4, Theorem 4] consider the representation

$$\text{(2)} \qquad \mathsf{S}_{\widetilde{\Phi}}(x) = \mathrm{e}^{\mathrm{i}\widetilde{\phi}_0\mathsf{Z}}\mathsf{W}(x)\mathrm{e}^{\mathrm{i}\widetilde{\phi}_1\mathsf{Z}}\mathsf{W}(x)\cdots\mathrm{e}^{\mathrm{i}\widetilde{\phi}_{d-1}\mathsf{Z}}\mathsf{W}(x)\mathrm{e}^{\mathrm{i}\widetilde{\phi}_d\mathsf{Z}} = \mathrm{e}^{\mathrm{i}\widetilde{\phi}_0\mathsf{Z}}\prod_{k=1}^{d}\mathsf{W}(x)\mathrm{e}^{\mathrm{i}\widetilde{\phi}_k\mathsf{Z}},$$

where

$$\mathsf{W}(x) = \mathrm{e}^{\mathrm{i}\mathsf{X}\arccos x} = \begin{pmatrix} x & \mathrm{i}\sqrt{1-x^2} \\ \mathrm{i}\sqrt{1-x^2} & x \end{pmatrix}.$$

Eq. (1) is usually called *O-representation*, while Eq. (2) is called *W-representation.* Since $\mathsf{W}(x) = \mathrm{e}^{-\mathrm{i}\frac{\pi}{4}\mathsf{Z}}\mathsf{O}(x)\mathrm{e}^{\mathrm{i}\frac{\pi}{4}\mathsf{Z}}$, it is not hard to see that both representations can be used interchangeably:

$$\mathsf{S}_{\widetilde{\Phi}}(x) = \mathrm{e}^{\mathrm{i}\widetilde{\phi}_0\mathsf{Z}}\prod_{k=1}^{d}\mathsf{W}(x)\mathrm{e}^{\mathrm{i}\widetilde{\phi}_k\mathsf{Z}} = \mathrm{e}^{-\mathrm{i}\frac{\pi}{4}\mathsf{Z}}\mathrm{e}^{\mathrm{i}\widetilde{\phi}_0\mathsf{Z}}\left(\prod_{k=1}^{d}\mathsf{W}(x)\mathrm{e}^{\mathrm{i}\widetilde{\phi}_k\mathsf{Z}}\right)\mathrm{e}^{\mathrm{i}\frac{\pi}{4}\mathsf{Z}} = \mathsf{S}_\Phi(x),$$

where $\phi_0 = \widetilde{\phi}_0 - \frac{\pi}{4}$, $\phi_d = \widetilde{\phi}_d + \frac{\pi}{4}$, and $\phi_k = \widetilde{\phi}_k$ for $k = 1, \dots, d-1$.

Theorem 1 (Quantum signal processing)**.** *There is a set of phases $\Phi := (\phi_0, \dots, \phi_d) \in \mathbb{R}^{d+1}$ such that*

$$\mathsf{S}_\Phi(x) = \mathrm{e}^{\mathrm{i}\phi_0 \mathsf{Z}} \prod_{k=1}^{d} \mathsf{O}(x)\mathrm{e}^{\mathrm{i}\phi_k \mathsf{Z}} = \begin{pmatrix} P(x) & -Q(x)\sqrt{1-x^2} \\ Q^*(x)\sqrt{1-x^2} & P^*(x) \end{pmatrix} \tag{3}$$

if and only if $P, Q \in \mathbb{C}[x]$ satisfy

(1) $\deg(P) \le d$ *and* $\deg(Q) \le d-1$ *(where* $\deg(Q) = -1$ *means* $Q = 0$*);*
(2) P *has parity* $d \pmod 2$ *and* Q *has parity* $d-1 \pmod 2$*;*
(3) $|P(x)|^2 + (1-x^2)|Q(x)|^2 = 1$ *for all* $x \in [-1,1]$.

Proof. We start with the “ $\implies$ ” direction. From the unitarity of $\mathsf{S}_\Phi(x)$,

$$\mathsf{S}_\Phi(x)\mathsf{S}_\Phi(x)^\dagger = \mathsf{I} \implies |P(x)|^2 + (1-x^2)|Q(x)|^2 = 1,$$

so Condition 3 is satisfied. When $d = 0$, $\mathsf{S}_\Phi(x) = \mathrm{e}^{\mathrm{i}\phi_0 \mathsf{Z}}$, which yields $P(x) = \mathrm{e}^{\mathrm{i}\phi_0}$ and $Q = 0$, thus satisfying Conditions 1 and 2. For $d > 0$, suppose by induction that $\mathsf{S}_{(\phi_0,\dots,\phi_{d-1})}(x)$ takes the form in Eq. (3) with degree $d-1$ and its corresponding $P, Q \in \mathbb{C}[x]$ satisfy Conditions 1 and 2. Hence, for all $\phi \in \mathbb{R}$,

$$\begin{aligned}
&\mathsf{S}_{(\phi_0,\dots,\phi_{d-1},\phi)}(x) = \mathsf{S}_{(\phi_0,\dots,\phi_{d-1})}(x)\mathsf{O}(x)\mathrm{e}^{\mathrm{i}\phi\mathsf{Z}} \\
&= \begin{pmatrix} P(x) & -Q(x)\sqrt{1-x^2} \\ Q^*(x)\sqrt{1-x^2} & P^*(x) \end{pmatrix} \begin{pmatrix} x & -\sqrt{1-x^2} \\ \sqrt{1-x^2} & x \end{pmatrix} \begin{pmatrix} \mathrm{e}^{\mathrm{i}\phi} & 0 \\ 0 & \mathrm{e}^{-\mathrm{i}\phi} \end{pmatrix} \\
&= \begin{pmatrix} xP(x) - (1-x^2)Q(x) & -\sqrt{1-x^2}(P(x) + xQ(x)) \\ \sqrt{1-x^2}(P^*(x) + xQ^*(x)) & xP^*(x) - (1-x^2)Q^*(x) \end{pmatrix} \begin{pmatrix} \mathrm{e}^{\mathrm{i}\phi} & 0 \\ 0 & \mathrm{e}^{-\mathrm{i}\phi} \end{pmatrix} \\
&= \begin{pmatrix} \mathrm{e}^{\mathrm{i}\phi}\big(xP(x) - (1-x^2)Q(x)\big) & \mathrm{e}^{-\mathrm{i}\phi}\big(-\sqrt{1-x^2}(P(x) + xQ(x))\big) \\ \mathrm{e}^{\mathrm{i}\phi}\big(\sqrt{1-x^2}(P^*(x) + xQ^*(x))\big) & \mathrm{e}^{-\mathrm{i}\phi}\big(xP^*(x) - (1-x^2)Q^*(x)\big) \end{pmatrix} \\
&= \begin{pmatrix} \widetilde{P}(x) & -\widetilde{Q}(x)\sqrt{1-x^2} \\ \widetilde{Q}^*(x)\sqrt{1-x^2} & \widetilde{P}^*(x) \end{pmatrix}.
\end{aligned}$$

Thus $\widetilde{P}(x) = \mathrm{e}^{\mathrm{i}\phi}(xP(x) - (1-x^2)Q(x))$ and $\widetilde{Q}(x) = \mathrm{e}^{-\mathrm{i}\phi}(P(x) + xQ(x))$ have degree at most d and $d-1$, respectively, since $\deg(P) \le d-1$ and $\deg(Q) \le d-2$ by induction hypothesis. Similarly, $\widetilde{P}$ and $\widetilde{Q}$ have parity $d \pmod 2$ and $d-1 \pmod 2$, respectively, since P has parity $d-1 \pmod 2$ and Q has parity $d-2 \pmod 2$. This completes the “ $\implies$ ” direction.

We now consider the “ $\impliedby$ ” direction. When $d = 0$, then $Q = 0$ and P is such that $|P(x)| = 1$ and $\deg(P) = 0$. The only possibility is thus $P(x) = \mathrm{e}^{\mathrm{i}\phi_0}$ and $Q = 0$, which satisfies Eq. (3). Another possibility is $d > 0$ even while $\deg(P) = 0$, meaning that $P(x) = \mathrm{e}^{\mathrm{i}\phi_0}$ and $Q = 0$. In this case, note that

$$\mathsf{O}(x)^{-1} = \mathsf{O}(x)^\dagger = \begin{pmatrix} x & \sqrt{1-x^2} \\ -\sqrt{1-x^2} & x \end{pmatrix} = \mathrm{e}^{-\mathrm{i}\frac{\pi}{2}\mathsf{Z}}\mathsf{O}(x)\mathrm{e}^{\mathrm{i}\frac{\pi}{2}\mathsf{Z}},$$

and so, if we take $\phi_k = (-1)^k \frac{\pi}{2}$ for $k = 1, \dots, d$, then

$$\mathrm{e}^{\mathrm{i}\phi_0 \mathsf{Z}} \prod_{k=1}^{d} \mathsf{O}(x) \mathrm{e}^{\mathrm{i}\phi_k \mathsf{Z}} = \mathrm{e}^{\mathrm{i}\phi_0 \mathsf{Z}} (\mathsf{O}(x) \mathrm{e}^{-\mathrm{i}\frac{\pi}{2}\mathsf{Z}} \mathsf{O}(x) \mathrm{e}^{\mathrm{i}\frac{\pi}{2}\mathsf{Z}})^{\frac{d}{2}} = \mathrm{e}^{\mathrm{i}\phi_0 \mathsf{Z}} (\mathsf{O}(x)\mathsf{O}(x)^\dagger)^{\frac{d}{2}} = \mathrm{e}^{\mathrm{i}\phi_0 \mathsf{Z}}.$$

This means that Eq. (3) holds.

For the remaining cases, the proof is by induction on d. Assume Eq. (3) holds for $d-1$. Consider P, Q satisfying Conditions 1 to 3 with $\deg(P) = d > 0$. Therefore, $\deg(|P(x)|^2) = 2d > 0$, and by Condition 3, we must have that $\deg(|Q(x)|^2) = 2d - 2 \implies \deg(Q) = d - 1$. Expand P, Q as

$$P(x) = \sum_{k=0}^{d} \alpha_k x^k \quad \text{and} \quad Q(x) = \sum_{k=0}^{d-1} \beta_k x^k.$$

The leading term of $|P(x)|^2 + (1-x^2)|Q(x)|^2$ is $(|\alpha_d|^2 - |\beta_{d-1}|^2)x^{2d}$, which is zero according to Condition 3. Thus $|\alpha_d| = |\beta_{d-1}|$. On the other hand, for any $\phi \in \mathbb{R}$,

$$\begin{aligned}
(4) \quad & \begin{pmatrix} P(x) & -Q(x)\sqrt{1-x^2} \\ Q^*(x)\sqrt{1-x^2} & P^*(x) \end{pmatrix} \mathrm{e}^{-\mathrm{i}\phi \mathsf{Z}} \mathsf{O}(x)^\dagger \\
&= \begin{pmatrix} P(x) & -Q(x)\sqrt{1-x^2} \\ Q^*(x)\sqrt{1-x^2} & P^*(x) \end{pmatrix} \begin{pmatrix} \mathrm{e}^{-\mathrm{i}\phi} & 0 \\ 0 & \mathrm{e}^{\mathrm{i}\phi} \end{pmatrix} \begin{pmatrix} x & \sqrt{1-x^2} \\ -\sqrt{1-x^2} & x \end{pmatrix} \\
&= \begin{pmatrix} \mathrm{e}^{-\mathrm{i}\phi} x P(x) + (1-x^2)\mathrm{e}^{\mathrm{i}\phi} Q(x) & \sqrt{1-x^2}(\mathrm{e}^{-\mathrm{i}\phi} P(x) - \mathrm{e}^{\mathrm{i}\phi} x Q(x)) \\ \sqrt{1-x^2}(\mathrm{e}^{-\mathrm{i}\phi} x Q^*(x) - \mathrm{e}^{\mathrm{i}\phi} P^*(x)) & \mathrm{e}^{\mathrm{i}\phi} x P^*(x) + \mathrm{e}^{-\mathrm{i}\phi}(1-x^2) Q^*(x) \end{pmatrix} \\
&= \begin{pmatrix} \widetilde{P}(x) & -\widetilde{Q}(x)\sqrt{1-x^2} \\ \widetilde{Q}^*(x)\sqrt{1-x^2} & \widetilde{P}^*(x) \end{pmatrix}.
\end{aligned}$$

By a proper choice of $\phi \in \mathbb{R}$, it is possible to obtain $\deg(\widetilde{P}) = d - 1$. Indeed, let $\mathrm{e}^{2\mathrm{i}\phi} = \alpha_d / \beta_{d-1}$ (since $|\alpha_d|/|\beta_{d-1}| = 1$). Then the coefficient of the x^{d+1} term in $\widetilde{P}$ is $\mathrm{e}^{-\mathrm{i}\phi}\alpha_d - \mathrm{e}^{\mathrm{i}\phi}\beta_{d-1} = \mathrm{e}^{-\mathrm{i}\phi}(\alpha_d - \mathrm{e}^{2\mathrm{i}\phi}\beta_{d-1}) = 0$. By the same token, the coefficient of the x^d term in $\widetilde{Q}$ is $-\mathrm{e}^{-\mathrm{i}\phi}\alpha_d + \mathrm{e}^{\mathrm{i}\phi}\beta_{d-1} = 0$. The coefficient of the x^d term in $\widetilde{P}$ and the coefficient of the x^{d-1} term in $\widetilde{Q}$ must also be zero, this time by the parity condition from Condition 2. In summary,

(1) $\deg(\widetilde{P}) \leq d - 1$ and $\deg(\widetilde{Q}) \leq d - 2$;
(2) $\widetilde{P}$ has parity $d - 1 \pmod 2$ and $\widetilde{Q}$ has parity $d - 2 \pmod 2$;
(3) $|\widetilde{P}(x)|^2 + (1-x^2)|\widetilde{Q}(x)|^2 = |P(x)|^2 + (1-x^2)|Q(x)|^2 = 1$ for all $x \in [-1, 1]$, by unitarity.

By the induction hypothesis, Eq. (3) holds for phases $(\phi_0, \dots, \phi_{d-1}) \in \mathbb{R}^d$ and polynomials $\widetilde{P}, \widetilde{Q}$. Therefore, from Eq. (4) it means that Eq. (3) holds for phases $(\phi_0, \dots, \phi_{d-1}, \phi) \in \mathbb{R}^{d+1}$ and polynomials P, Q. This completes the proof. □

References

[1] Dominic W. Berry, Andrew M. Childs, Richard Cleve, Robin Kothari, and Rolando D. Somma. Exponential improvement in precision for simulating sparse hamiltonians. In *Proceedings of the Forty-Sixth Annual ACM Symposium on Theory of Computing*, STOC '14, page 283–292, New York, NY, USA, 2014. Association for Computing Machinery.

[2] Andrew M. Childs, Dmitri Maslov, Yunseong Nam, Neil J. Ross, and Yuan Su. Toward the first quantum simulation with quantum speedup. *Proceedings of the National Academy of Sciences*, 115(38):9456–9461, 2018

[3] Andrew M. Childs and Nathan Wiebe. Hamiltonian simulation using linear com- binations of unitary operations. *Quantum Info. Comput.*, 12(11–12):901–924, Nov 2012.

[4] András Gilyén, Yuan Su, Guang Hao Low, and Nathan Wiebe. Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2019, page 193–204, New York, NY, USA, 2019. Association for Computing Machinery.

[5] Jeongwan Haah, Matthew Hastings, Robin Kothari, and Guang Hao Low. Quantum algorithm for simulating real time evolution of lattice Hamiltonians. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 350–360, 2018.

[6] Aram W. Harrow, Avinatan Hassidim, and Seth Lloyd. Quantum algorithm for linear systems of equations. *Phys. Rev. Lett.*, 103:150502, Oct 2009.

[7] Guang Hao Low and Isaac L. Chuang. Optimal Hamiltonian simulation by quantum signal processing. *Phys. Rev. Lett.*, 118:010501, Jan 2017.

[8] Guang Hao Low and Isaac L. Chuang. Hamiltonian Simulation by Qubitization. *Quantum*, 3:163, July 2019.

[9] Lin Lin. Lecture notes on quantum algorithms for scientific computation. *arXiv preprint arXiv:2201.08309*, 2022.

[10] Guang Hao Low, Theodore J. Yoder, and Isaac L. Chuang. Methodology of resonant equiangular composite quantum gates. *Phys. Rev. X*, 6:041067, Dec 2016.

The modified Korteweg-de Vries equation and inverse scattering

Nicolas Faross

In this talk, we consider the modified Korteweg-de Vries equation, a nonlinear partial differential equation used to model water waves. Based on the previous talks on the nonlinear Fourier transform, we sketch how this equation can be solved using the inverse scattering method. Additionally, we compare this approach to the method of solving the Airy equation using the linear Fourier transform. Our main references are the lecture notes [1, Lecture 6.1] and the article [2].

1. The modified Korteweg-de Vries equation

The modified Korteweg-de Vries (mKdV) equation for a function $F(t,x)$ is given by

$$F_t + F_{xxx} + 6F^2F_x = 0,$$

where the lower indices denote partial derivatives with respect to the position x and the time t. Physically, $F(t,x)$ describes the height of water relative to its surface and the equation models the propagation of waves in shallow water. A main feature of the equation is the existence of soliton solutions given by wave packets that travel indefinitely while preserving their shape. Further, the unmodified equation has been the main motivation for the development of the inverse scattering method in [3]. For further information, we refer to [4, Section 9.1.1] or [5, Chapter 7].

2. The linear Fourier method

Following the lecture [1, Lecture 6.1], we first consider the Airy equation and show how it can be solved using the linear Fourier transform. Consider the Cauchy-problem for the Airy equation given by

$$F_t = F_{xxx}, \quad F(0,x) = F_0(x)$$

that is obtained by dropping the nonlinear term $6F^2F_x$ of the mKdV equation. By applying the linear Fourier transform on $\mathbb{R}$ defined by

$$\widehat{F}(t,k) = \int_{\mathbb{R}} F(t,x)e^{2ikx}\,\mathrm{d}x,$$

we can transform the Airy equation into the new equation

$$\widehat{F_t} = (-2ik)^3\widehat{F} = 8ik^3\widehat{F}.$$

For fixed k, this yields an ordinary differential equation that can be solved explicitly by

$$\widehat{F}(t,k) = e^{8ik^3t}\widehat{F_0}(k).$$

Thus, we have constructed a solution to the Airy equation modulo to applying a Fourier transform to the initial data F_0 and an inverse Fourier transform to $\widehat{F}$.

3. The nonlinear Fourier transform

Next, we want to apply the nonlinear Fourier transform to solve the Cauchy-problem for the mKdV equation of the form

$$F_t = F_{xxx} + 6F^2F_x, \quad F(0,x) = F_0(x).$$

However, we first have to pass from the nonlinear Fourier transform on $\mathbb{Z}$ in the previous talks to the nonlinear Fourier transform on $\mathbb{R}$. This can be done via a limit procedure that replaces $(a_n(t,z), a_n(t,z))$ index by a discrete $n \in \mathbb{Z}$ with $(a(t,k,x), b(t,k,x))$ index by a continuous parameter $k \in \mathbb{R}$. Further, the recursion relation for $(a_n(t,z), b_n(t,z))$ becomes the differential equation

$$\frac{\partial}{\partial x}(a(t,k,x), b(t,k,x)) = (a(t,k,x), b(t,k,x))(0, F(t,x)e^{2ikx}).$$

In the following, we assume for simplicity that F has only compact support. Then the nonlinear Fourier transform of F is given by

$$\overbrace{F}(t,k) = (a(t,k,\infty), b(t,k,\infty))$$

with initial values $(a(t,k,-\infty), b(t,k,-\infty)) = (1,0)$ and $x = \pm\infty$ denoting a sufficiently large/small point outside the support of F. In the next section, we outline how applying the Fourier transform to the mKdV equation yields

$$\overbrace{F}(t,k) = (a(t,k,\infty), e^{8ik^3t}b(t,k,\infty)),$$

where $(a(0,k,\infty), b(0,k,\infty))$ is obtained from the Fourier transform of initial data

$$\widehat{F_0}\,(k) = (a(0,k,\infty), b(0,k,\infty)).$$

Thus, we obtain a solution of the mKdV equation modulo the nonlinear Fourier transform similarly to the linear case of the Airy equation.

4. Lax pair formulation

To prove the previous statement, one can use the function $F(t,x)$ to construct a Lax pair [6] consisting of operators $L(t)$ and $P(t)$ satisfying the equation

$$\frac{\mathrm{d}}{\mathrm{dt}} L(t) = [L(t), P(t)],$$

where $[L,P]$ denotes the commutator of L and P. This equation implies that eigenvectors $\phi(t)$ of $L(t)$ are preserved under the flow of P defined by

$$\frac{\mathrm{d}}{\mathrm{dt}} \phi(t) = P(t)\phi(t).$$

By the construction of L (see [1]), its eigenvectors $\phi(t)$ are determined by the functions $a(t,k,x)$ and $b(t,k,x)$ appearing in the definition of the nonlinear Fourier transform of $F(t,k)$. Thus, given the nonlinear Fourier transform

$$\widehat{F_0}\,(k) = (a(0,k,\infty), b(0,k,\infty))$$

of the initial data F_0, we can construct initial eigenvectors $\phi(0)$ and use the flow equation for P to obtain an eigenvector $\phi(t)$ at any time $t > 0$. From this eigenvector $\phi(t)$, we reconstruct again the nonlinear Fourier transform given by

$$\widehat{F}\,(t,k) = (a(t,k,\infty), b(t,k,\infty)).$$

Since the point $x = \infty$ lies outside the support of F, the operator P does not depend on F in this region. Thus, the flow equation can be solved explicitly, which yields the statement

$$\widehat{F}\,(t,k) = (a(0,k,\infty), e^{8ik^3 t} b(0,k,\infty)).$$

5. Soliton solutions

Note that in the previous section, we did not consider the initial mKdV equation but the sign adjusted version used in [1, Lecture 6.1]. A solution of the initial mKdV equation

$$F_t + F_{xxx} + 6F^2 F_x = 0$$

is constructed in [2] using a similar method. However, in this case, the eigenvectors of the operator $L(t)$ no longer correspond to the nonlinear Fourier transform with respect to $SU(1,1)$ but they are given to the nonlinear Fourier transform with respect to $SU(2)$. Moreover, this version of the mKdV equation now admits soliton solutions. See again [2] or [4, Chapter 9.2.1] for explicit descriptions of these solutions.

References

[1] T. Tao and C. Thiele, *Nonlinear Fourier analysis*, arXiv:1201.5129 (2012).

[2] S. Tanaka, *Some remarks on the modified Korteweg-de Vries equations*, Publications of the Research Institute for Mathematical Sciences **8** (1972), 429–437.

[3] C. Gardner, J. Greene, M. Kruskal and R. Miura, *Method for solving the Korteweg-deVries equation*, Physical Review Letters **19** (1967), 1095–1097.

[4] A. Polyanin and V. Zaitsev, *Handbook of nonlinear partial differential equations*, Chapman and Hall/CRC (2003).

[5] F. Linares and G. Ponce, *Introduction to nonlinear dispersive equations*, Springer (2015).

[6] P. Lax, *Integrals of nonlinear equations of evolution and solitary waves*, Communications on Pure and Applied Mathematics **21** (1968), 467–490.

Jacobi matrices and Schrödinger operators

Rubén de la Fuente Fernández

We study an application of Schur's algorithm to prove spectral properties of Jacobi matrices that can be used to show results about the spectrum of Schrödinger operators on the discrete half-line.

In particular, we will focus on the paper [DK04], which is devoted to studying spectral properties of half-line discrete Schrödinger operators with a Dirichlet boundary condition at the origin. Those operators take the form

$$[h_V\psi](n) = \psi(n+1) + \psi(n-1) + V(n)\psi(n) \tag{1}$$

with $\psi \in l^2(\mathbb{Z}^+)$, $\mathbb{Z}^+ = \{1, 2, ...\}$ and $\psi(0)$=0.

In the free case ($V = 0$), the spectrum of h_0 is $[-2, 2]$, and it is purely absolutely continuous. In [DK04], the authors give two results regarding the spectrum of h_V.

Theorem 1. [DK04, Theorem 1] *A discrete half-line Schrödinger operator h_V with spectrum contained in $[-2, 2]$ has purely absolutely continuous spectrum.*

Theorem 2. [DK04, Theorem 3] *If a discrete half-line Schrödinger operator h_V has only finitely many eigenvalues outside $[-2, 2]$, then it has purely absolutely continuous spectrum on $[-2, 2]$.*

To prove these theorems, the authors mainly use one result about the spectrum of Jacobi operators [DK04, Theorem 5], which will be the object of our attention. The proof of this is strongly based on Schur's algorithm and orthogonal polynomials to give a relation between the coefficients defining a Jacobi operator and Schur coefficients. They also prove natural analogs to Theorems 1 and 2 for the continuous setting, but we will focus on the discrete case.

Jacobi matrices and Schur coefficients

A Jacobi Matrix acting on the disrete half-line is an operator of the form

$$[J\psi](n) = a_n\psi(n+1) + a_{n-1}\psi(n-1) + b_n\psi(n)$$

with $\psi \in l^2(\mathbb{Z}^+)$, a_n positive and b_n real. Both coefficient sequences are assumed to be bounded; therefore, J defines a bounded self-adjoint operator. Note that in the case $a_n = 1$, $b_n = V(n)$, this is just a Schrödinger operator of the form (1).

The result we will focus on, which corresponds to Section 2 in [DK04], is the following

Theorem 3. [DK04, Theorem 5] *A Jacobi matrix with coefficients a_n and b_n has spectrum $\sigma(J) \subseteq [-2,2]$ if and only if there is a sequence $\gamma_n \in (-1,1)$, $n \in \{0,1,...\}$, that obeys*

$$b_{n+1} = (1-\gamma_{2n-1})\gamma_{2n} - (1+\gamma_{2n-1})\gamma_{2n-2} \tag{2}$$

$$a_{n+1}^2 = (1-\gamma_{2n-1})(1-\gamma_{2n})^2(1+\gamma_{2n+1}) \tag{3}$$

(Here $\gamma_{-1} = -1$, and the value of γ_{-2} is irrelevant since it is multiplied by zero.)

This sequence γ_n comprises exactly the coefficients of a Schur algorithm starting from a function constructed using a spectral measure of J. The authors of [DK04] devote Sections 3, 4, and 5 to prove bounds for the Schur coefficients associated with Jacobi matrices and use them together with Theorem 3 to show properties of the spectrum of Schrödinger operators and, in particular, Theorems 1 and 2.

Sketch of the proof of Theorem 3

The strategy of the proof starts with constructing the spectral measure of J associated with the vector $\delta_{1,n} \in l^2(\mathbb{Z}^+)$, the Kronecker delta function at 1, which we will denote $d\mu$. This vector is cyclic with respect to J, so the support of $d\mu$ coincides with the spectrum of J. From cyclicity, we can also derive that J is unitarily equivalent to $g(x) \mapsto xg(x)$ in $L^2(d\mu)$. As $l^2(\mathbb{Z}^+)$ is infinite dimensional, $L^2(d\mu)$ is as well, therefore $d\mu$ must be supported in an infinite set.

We can also write the m-function of J using $d\mu$

$$m_0(z) = \int \frac{1}{t-z} d\mu(t),$$

that uniquely determines J. Now, we assume $\sigma(J) \subseteq [-2,2]$. Then, we can use $d\mu$ to define a measure on $\mathbf{S}^1$

$$\int g(t)d\mu(t) = \int g(\zeta + \zeta^{-1})d\rho(\zeta).$$

The next step is to construct a Schur function f_0 using the Carathéodory function F_0 associated with ρ

$$F_0(\xi) = \int \frac{\zeta+\xi}{\zeta-\xi} d\rho(\zeta) = (\xi - \xi^{-1})m_0(\xi+\xi^{-1}) \quad ; \quad f_0 = \frac{1}{\xi}\frac{F_0(\xi)-1}{F_0(\xi)+1}.$$

The fact that $d\mu$ is supported on an infinite set implies that f_0 cannot be written as a Blaschke product, allowing us to set up Schur's algorithm to construct the Schur coefficients for f_0

$$f_{n+1}(\xi) = \frac{1}{\xi}\frac{f_n(\xi) - \gamma_n}{1 - \bar{\gamma}_n f_n(\xi)},$$

with $\gamma_n = f_n(0)$ and $|\gamma_n| < 1$. The key observation is that the step in which we defined f_0 using $d\rho$ can be inverted, so we can define a different measure on $\mathbf{S}^1$ for each f_n we produce. These measures are then used to construct Carathéodory functions F_n, m-functions m_n and Jacobi matrices J_n. Relations (2) and (3) are obtained by computing the coefficients of J_{2n}.

For the converse direction of Theorem 3, we assume that the coefficients of J fulfill (2) and (3). Then we can construct a Schur function with these coefficients γ_n. With this function we construct a probability measure $d\tilde{\rho}$ on $\mathbf{S}^1$, which induces a probability measure on $[-2, 2]$, and therefore a Jacobi matrix $\tilde{J}$ with spectrum contained in $[-2, 2]$. But as the coefficients of $\tilde{J}$ are also determined by (2) and (3), then $\tilde{J} = J$ and $\sigma(J) = \sigma(\tilde{J}) \subseteq [-2, 2]$.

Theorem 3 is not new to [DK04]; it already appeared in [G54]. However, the authors of [DK04] present a short, clear and self-contained proof that explicitly relies on the Schur algorithm and the shape of Jacobi operators.

References

[DK04] D. Damanik and R. Killip, *Half line Schrödinger operators with no bound states.* Acta Math., 193(1):31–72, 2004.

[G54] Ya. L. Geronimus, *Polynomials Orthogonal on a Circle and Their Applications.* Amer. Math. Soc. Translation **104**, AMS, Providence, RI, 1954.

[S05] B. Simon, *Orthogonal Polynomials on the Unit Circle. Part 2. Spectral theory.* American Mathematical Society Colloquium Publications. **54**, AMS, Providence, RI, 2005.

[T00] G. Teschl, *Jacobi Operators and Completely Integrable Nonlinear Lattices.* Math. Surveys Monogr. **72**, AMS, Providence, RI, 2000.

The Nonlinear Fourier Transform on $\ell^2(\mathbf{Z})$

Max Giessler

In this talk we extend the definition of the NLFT to square summable sequences on the full line and discuss some of its properties, following lecture 3 of Tao's and Thiele's lecture notes [2]. In particular, we are concerned with its invertibility properties.

For a sequence $F \in \ell^2(\mathbf{Z}_{\leq -1}, D)$ supported on the negative integers and taking values in the complex disc D we define its NLFT to be

$$\overbrace{F}(z) := (a^*(z^{-1}), b(z^{-1})), \quad z \in \mathbf{T},$$

where (a, b) is the NLFT of the reflected sequence in $\ell^2(\mathbf{Z}_{\geq 1}, D)$ as defined in the previous talk. Recall that the NLFT is a homeomorphism from $\ell^2(\mathbf{Z}_{\geq 0}, D)$ to $\mathbf{H}$, defined to be the space of all $SU(1, 1)$-valued functions (a, b) such that a satisfies an outerness condition and a normalization condition at ∞, and b/a^* satisfies a holomorphicity condition (see [1] for details on the underlying complex analysis). Analogously, we define $\mathbf{H}_0^*$, replacing the holomorphicity condition with one for b/a and additionally requiring that $b(\infty) = 0$. By the shifting property (Lemma 1 (5) in [2]) one can deduce that the NLFT extends to a homeomorphism from $\ell^2(\mathbf{Z}_{\leq -1}, D)$ to $\mathbf{H}_0^*$. Intuitively, the NLFT is a bijection on both half-lines.

Now we define the NLFT of a sequence $F \in \ell^2(\mathbf{Z}, D)$ to be the $SU(1,1)$-valued measurable function on $\mathbf{T}$ given by the matrix product

$$\overbrace{F}(z) := \overbrace{F^{\leq -1}}(z)\,\overbrace{F^{\geq 0}}(z), \tag{1}$$

where $F^{\leq -1} \in \ell^2(\mathbf{Z}_{\leq -1}, D)$ and $F^{\geq 0} \in \ell^2(\mathbf{Z}_{\geq 0}, D)$ denote the truncations of the sequence F to its negative and non-negative entries, respectively. In line with the earlier notation, we also write $(a, b) = (a_- b_-)(a_+ b_+)$ for (1). Observe that we modified the definition of the NLFT on the half-line in accordance with Lemma 1 in [2] to gain a definition for the full-line. Therefore, the NLFT on the full-line still satisfies Lemma 1 and is consistent with the definition of the NLFT on $\ell^p(\mathbf{Z})$ for $1 \leq p < 2$. In a way, until now we have only harvested the fruit from our work on the half-line.

Further investigation shows that the NLFT maps $\ell^2(\mathbf{Z}, D)$ into the space $\mathbf{L}$ and is continuous. The Plancherel identity

$$\int_{\mathbf{T}} \log|a(z)| = -\frac{1}{2} \sum_{z \in \mathbf{Z}} \log(1 - |F_n|^2)$$

carries over. However, bijectivity is lost: Indeed, the NLFT in *not* injective on $\ell^2(\mathbf{Z}, D)$.

This insight leads the way to the inverse problem with which we shall concern ourselves for the remainder of the talk. We want to find a (not necessarily unique) preimage for a given function $(a, b) \in \mathbf{L}$, i.e. a sequence $F \in \ell^2(\mathbf{Z}, D)$ with NLFT (a, b). By the half-line theory, this boils down to finding a matrix factorization

$$(a, b) = (a_- b_-)(a_+ b_+) \quad \text{with } (a_- b_-) \in \mathbf{H}_0^* \text{ and } (a_+ b_+) \in \mathbf{H}. \tag{2}$$

That is because any such factorization is the NLFT of uniquely determined truncations $F^{\leq -1}$ and $F^{\geq 0}$ of a sequence $F \in \ell^2(\mathbf{Z}, D)$ as the NLFT is a bijection on the half-lines. Keeping in mind this equivalence between finding a preimage for the NLFT and finding a factorization as in (2) is worth it.

The factorization problem (2) of a matrix-valued function on $\mathbf{T}$ is called *Riemann–Hilbert problem.* It is possible to rewrite it as a product of functions on D and D^*, respectively, obtaining the classical formulation of the R-H problem modulo outerness, normalization, and holomorphicity constraints (this is done in [2]).

We can recover injectivity of the NLFT or, equivalently, prove the uniqueness of any R-H factorization if we additionally assume a to be bounded:

Theorem 1 ([2], Lemma 18)**.** *For a function $(a, b) \in \mathbf{L}$ where a is bounded there is a* unique *$F \in \ell^2(\mathbf{Z}, D)$ such that $\overbrace{F} = (a, b)$.*

The proof relies upon the Banach fixed-point theorem. W.l.o.g. we can assume that a_+, b_+^*, a_-, b_- lie in the Hardy space of square-integrable functions $H^2(D^*)$. We can show that the factor (a_+, b_+) in any R-H factorization must be a fixed point of the map

$$(A, B) \mapsto \big(c + P_{D^*}(Bb^*/a^*), P_D(Ab/a)\big) \tag{3}$$

mapping $L^2(\mathbf{T}) \times L^2(\mathbf{T})$ into itself. Here, P_D and P_{D^*} are the orthogonal projections from the Hilbert space $L^2(\mathbf{T})$ to $H^2(D)$ and $H^2_0(D^*)$, respectively, and c is a uniquely determined constant. Crucially, to prove that (3) is a contraction we need that a is bounded. Then we obtain as the unique fixed point (a_+, b_+) and the corresponding factor (a_-, b_-) by rewriting (2):

$$(a_-, b_-) = (a, b)(a_+^*, -b_+). \tag{4}$$

This shows the uniqueness of the R-H factorization and therefore of the inverse.

The general case of unbounded a is more complicated. In particular, we cannot prove the uniqueness of the preimage anymore. Let us fix a function $(a, b) \in \mathbf{L}$. Instead of using the Banach fixed-point theorem, we apply the Riesz representation theorem to the functional $\lambda : (A, B) \mapsto \operatorname{Re}[A(\infty)]$ on certain Hilbert spaces H_{min} and H_{max}. These are nested in between

$$H^2(D^*) \times H^2(D) \subseteq H_{min} \subsetneq H_{max} \subseteq aH^2(D^*) \times a^* H^2(D)$$

and equipped with the scalar product $\langle (A', B'), (A, B) \rangle := \int_{\mathbf{T}} \operatorname{Re}[A'(A^* - \frac{b}{a}B^*) + (B')^*(B - \frac{b}{a}A)]$. Let (A_{min}, B_{min}) be the unique element in H_{min} which represents λ in this scalar product and likewise (A_{max}, B_{max}). In this setting, we can prove the following general result for the inverse problem:

Theorem 2 ([2], Theorem 7)**.** *Let $(a, b) \in \mathbf{L}$. Then there exists a R-H factorization (2). Two possible choices are given by*

$$(a_+, b_+) = (A_{min}, B_{min}) A_{min}(\infty)^{-1/2} \quad \textit{and}$$
$$(a_+, b_+) = (A_{max}, B_{max}) A_{max}(\infty)^{-1/2}$$

where in each case (a_-, b_-) is determined as in (4).

While we approached the proof of Theorem 1 from the point of view of the R-H problem, here we take the perspective of finding a preimage for the NLFT. We sketch the proof for H_{min}. Let H_n for all $n \in \mathbf{Z}$ be a particular family of Hilbert spaces such that $H_0 = H_{min}$ and $H_{n+1} \subseteq H_n$ (see [2] for details). For each integer n let (A_n, B_n) be the Riesz representer of the functional λ in H_n. Then we can deduce the relation

$$(A_{n+1}, B_{n+1}) = (A_n, B_n) - F_n(B_n^* z^n, A_n^* z^n)$$

for uniquely determined complex numbers F_n in the unit disc D. It remains to verify that the sequence $F := (F_n)_{n \in \mathbf{Z}}$ is indeed in $\ell^2(\mathbf{Z}, D)$ and that its NLFT is (a, b). Finally, the NLFT of its truncation $\widehat{F^{\geq 0}} = (a_+, b_+)$ is of the required form. Note that we can imitate this proof for H_{max} and thus obtain another, different sequence $\tilde{F}$ with NLFT (a, b).

Taking the point of view of the R-H problem again the underlying reason for the loss of uniqueness in Theorem 2 turns out the be the following:

Theorem 3 ([2], Theorem 8)**.** *Let* $(a,b) \in \mathbf{L}$*. Then there exists a* unique *factorization*

$$(a,b) = (a_{--}, b_{--})(a_o, b_o)(a_{++}, b_{++})$$

with $(a_{--}, b_{--}) \in \mathbf{H}_0^*$*,* $(a_o, b_o) \in \mathbf{H}_0^* \cap \mathbf{H}$*,* $(a_{++}, b_{++}) \in \mathbf{H}$*. Furthermore,* (a_{--}, b_{--}) *and* (a_{++}, b_{++}) *only admit the trivial R-H factorizations* $(a_{--}, b_{--}) = (a_{--}, b_{--})(1,0)$ *and* $(a_{++}, b_{++}) = (1,0)(a_{++}, b_{++})$*.*

Because the factor (a_o, b_o) is in both the spaces $\mathbf{H}_0^* \cap \mathbf{H}$ we can multiply it with either factor (a_{--}, b_{--}) or (a_{++}, b_{++}) and stay within the spaces $\mathbf{H}_0^*$ or $\mathbf{H}$ by the group structure of $SU(1,1)$. We can even find new R-H factorizations of the middle factor (a_o, b_o) which we merge again with the factors (a_{--}, b_{--}) and (a_{++}, b_{++}). In this way, we obtain different R-H factorizations and consequently different preimages for (a,b).

References

[1] J.B. Garnett, *Bounded Analytic Functions*, Springer New York (2007).

[2] T. Tao and C. Thiele, *Nonlinear Fourier Analysis*, arXiv: 1201.5129 [math.CA] (2012).

Stability of Schur's iterates and fast solution of the discrete integrable NLS

Kaiyi Huang

1. Preliminaries

Let $\mathbb{D} = \{z \in \mathbb{C} : |z| < 1\}$ and $\mathbb{T} = \{z \in \mathbb{C} : |z| = 1\}$. Let $\mathcal{S}$ denote the Schur class of holomorphic functions $F : \mathbb{D} \longrightarrow \mathbb{D}$. For $1 \le p \le \infty$, $l^p(\mathbb{Z}, \mathbb{D})$ denotes the set of l^p sequences $\{q_n\}_{n\in\mathbb{Z}} \subset \mathbb{D}$. We may simply write l^p. Let $\mathbb{Z}_+ = \mathbb{Z} \cap [0, \infty)$. Let m denote the normalized Lebesgue measure on $\mathbb{T}$. For $r \in (0,1)$, define the $L^2(r\mathbb{T})$-norm by

$$\|F\|_{L^2(r\mathbb{T})} = \left(\int_{\mathbb{T}} |F(r\xi)|^2 dm(\xi) \right)^{\frac{1}{2}}.$$

1.1. **Nonlinear Fourier transform (NLFT) on** $SU(1,1)$**.** Let $q \in l^p(\mathbb{Z}, \mathbb{D})$ for some $1 \le p \le \infty$. Define $a, b : \mathbb{T} \longrightarrow \mathbb{C}$ by

$$\begin{pmatrix} a & b \\ \bar{b} & \bar{a} \end{pmatrix} = \prod_{k\in\mathbb{Z}} \frac{1}{\sqrt{1 - |q(k)|^2}} \begin{pmatrix} 1 & \overline{q(k)} z^{-k} \\ q(k) z^k & 1 \end{pmatrix},$$

where the product of matrices T_k is defined by

$$\prod_{k\in\mathbb{Z}} T_k = \lim_{n \longrightarrow +\infty} T_n T_{-n+1} \cdots T_{n-1} T_n.$$

The *reflection coefficient* of q is defined as $\mathbf{r}_q = \frac{b}{a}$. The scattering map $\mathcal{F}_{sc} : q \mapsto \mathbf{r}_q$ is called the *nonlinear Fourier transform* (NLFT).

Define

$$X = \{h \in L^\infty(\mathbb{T}) : \|h\|_{L^\infty(\mathbb{T})} \leq 1, \log(1 - |h|^2) \in L^1(\mathbb{T})\},$$

equipped with the Sylvester–Winebrenner metric[2][1]

$$\rho_s(h_1, h_2) = \sqrt{-\int_{\mathbb{T}} \log\left(1 - \left|\frac{h_1 - h_2}{1 - \overline{h_1} h_2}\right|^2\right) dm}.$$

For $\delta \in [0, 1)$, denote $B[\delta] = \{h \in l^\infty : \|h\|_{L^\infty(\mathbb{T})} \leq \delta\}$. Then $B[\delta] \subset X$. Define $\mathcal{G}[\delta] = \mathcal{F}_{sc}^{-1}(B[\delta]) \cap l^2$.

Theorem 1 (Properties of NLFT)**.**

(1) If $\mathbf{r}_q$ exists, then $|a|^2 - |b|^2 \equiv 1$.

(2) If $p = 1$, then $\mathbf{r}_q$ converges uniformly on $\mathbb{T}$; if $p = 2$, then $\mathbf{r}_q$ is known to converge in Lebesgue measure.

(3) $\mathcal{F}_{sc} : l^1 \longrightarrow L^\infty(\mathbb{T})$ is injective, and extends uniquely to a continuous map $\mathcal{F}_{sc} : l^2 \longrightarrow X$, which is surjective, but not injective.

(4) For every $\delta \in (0, 1)$, $\mathcal{F}_{sc} : \mathcal{G}[\delta] \longrightarrow B[\delta]$ is a homeomorphism.

(5) If $q = q(t, n)$ solves (1) with the initial data $q_0 \in \mathcal{G}[\delta]$ for some $\delta \in (0, 1)$, then $q(t, \cdot) = \mathcal{F}_{sc}^{-1}(e^{it(z+\frac{1}{z})}\mathcal{F}_{sc}(q_0)) \in \mathcal{G}[\delta]$ for all $t \in \mathbb{R}$.

(6) For every $q \in l^2$ and $n \in \mathbb{Z}$, $\mathcal{F}_{sc}(q(\cdot - n)) = z^{-n}\mathcal{F}_{sc}(q)$.

1.2. **Schur's algorithm.** Let $F \in \mathcal{S}$. Schur's algorithm runs as follows:

$$F_0 = F, \qquad zF_{n+1} = \frac{F_n - F_n(0)}{1 - \overline{F_n(0)} F_n}, \quad n \geq 0.$$

$F_n \in \mathcal{S}$ for all $n \geq 0$. $\{F_n(0)\}_{n \geq 0}$ are called the *recurrence coefficients* of F. We further define $\eta(F) = \prod_{k=0}^\infty (1 - |F_n(0)|^2)$.

Theorem 2 (Properties of Schur's algorithm)**.**

(1) (Stability) Let $F, G \in \mathcal{S}$. Suppose $\eta(F), \eta(G) \geq \eta > 0$. Then for every $r \in (0, 1), n \in \mathbb{Z}_+$

$$\|F_n - G_n\|_{L^2(r\mathbb{T})} \leq C(\eta, r) r^{-n} \|F - G\|_{L^2(r\mathbb{T})},$$

where $C(\eta, r) = \exp\{[2 + (1 - \sqrt{1 - \eta})^{-1}][4(1 - r)^{-2} + 1] \log(\eta^{-1})\}$.

(2) (Connections with NLFT) Suppose q is an l^2 sequence supported on $\mathbb{Z}_+$. Then $\mathbf{f}_q := \frac{\bar{b}}{a} \in \mathcal{S}$, and its recurrent coefficients coincide with q.

[1] Someone claims that in [2] there is a mistake in the proof of ρ_s being a metric.

2. Ablowitz–Ladik (AL) equation

We study the defocusing *Ablowitz–Ladik equation*,

$$
\left\{ \begin{array}{rcl} \frac{\partial}{\partial t} q(t,n) & = & i(1-|q(t,n)|^2)(q(t,n-1)+q(t,n+1)), \\ q(0,n) & = & q_0(n), \quad (t,n) \in \mathbb{R} \times \mathbb{Z}. \end{array} \right. \tag{1}
$$

This is a discretization of the defocusing cubic nonlinear Schrödinger equation (NLS),

$$
i\frac{\partial}{\partial t}u(t,x) = -\frac{\partial^2}{\partial x^2}u(t,x) + 2|u(t,x)|^2 u(t,x), \quad (t,x) \in \mathbb{R}^2.
$$

There are many ways to discretize the NLS equation so that it is integrable, but (1) gives an evolution $q(t,\cdot) = e^{it(z+\frac{1}{z})}q_0$ that will be useful.

3. The inverse scattering theory (IST)

For $q_0 \in l^1$, one can solve (1) by the following algorithm:

Step 1. Given initial data $q_0 \in l^1(\mathbb{Z}, \mathbb{D})$, compute the reflection coefficient $\mathbf{r}_{q_0}$.
Step 2. Find $q(t,\cdot) : \mathbb{Z} \longrightarrow \mathbb{D}$ such that $\mathbf{r}_{q(t,\cdot)} = e^{it(z+\frac{1}{z})}\mathbf{r}_{q_0}$ on $\mathbb{T}$, where $e^{it(z+\frac{1}{z})}$ is called the *inverse scattering multiplier.*

The algorithm doesn't work for $q_0 \in l^2$ because $\mathcal{F}_{sc}$ is not injective by Theorem 1[3, 4].

4. A fast algorithm for l^2 initial data

The following algorithm approximates the solution to (1) at $t > 0, n_0 \in \mathbb{Z}$ fast and accurately when $q_0 \in l^2$ and $\eta(q_0) = \prod_{n\in\mathbb{Z}}(1-|q_0(n)|^2) \geq \eta > 0$.

Step 1. Truncate q_0 by taking $\tilde{q}_0 = q_0 \mathcal{X}_{[n_0-N, n_0+N]}$ for some large N.
Step 2. Shift $\tilde{q}_0$ so that it's supported on $\mathbb{Z}_+ \cap [0, 2N]$.
Step 3. Truncate the Fourier series of $e^{it(z+\frac{1}{z})}$. Let $P_{n,t}$ be the partial Fourier sum of $e^{it(z+\frac{1}{z})}$ up to order n. Define

$$
G_{n,t} = (1-\delta_{n,t})z^n P_{n,t}, \qquad \delta_{n,t} = \frac{t^n e^t}{n!}, n > ct, c > e.
$$

Step 4. Run Schur's algorithm. Set $\mathbf{f}_{q_{0,N}} = \frac{\bar{b}}{a}, F_n = G_{n,t}\mathbf{f}_{q_{0,N}}$. Then $F_{n,0} \in \mathcal{S}$. Let $\{\alpha_{n,k}\}_{k\geq 0}$ denote the corresponding recurrence coefficients. Define

$$
\tilde{q}_n(t,j) = \begin{cases} \alpha_{n,n+j}, & j \geq -n, \\ 0, & j < -n. \end{cases}
$$

Step 5. Consider $q_0(-\cdot)$, and repeat steps 1-4.

Remark 3. *One can obtain results for $t < 0$ by symmetry. One can consider only $\mathbb{Z}_+$ by Theorem 1.6.*

The following theorems guarantee that $\lim_n \tilde{q}_n(t, n_0) = q(t, n_0)$.

Theorem 4. *Let $q_0 \in l^2$ and $\eta(q_0) \geq \eta > 0$. Let q be the solution to* (1) *with initial data q_0. Define $q_{0,N} = \mathcal{X}_{[-N,N]}q_0$. Let q_N be the corresponding solution to* (1)*. Then, for $|j| \leq N, t > 0, r \in (0,1)$,*

$$|q(t,j) - q_N(t,j)| \leq \frac{4e^{\frac{t}{r}}C(\eta(\eta,r))}{1-r}r^{N-|j|}.$$

Remark 5. *$l^2(\mathbb{Z},\mathbb{D})$ is nonlinear, so the density argument doesnt' apply. Stability of Schur's algorithm is key to the proof. Since Schur's algorithm goes only in one direction, one needs to run the algorithm in both directions of support $\mathbb{Z}$ of q_0.*

Theorem 6. *Let $t > 0$ and $q_0 \in l^2$ be compactly supported on $\mathbb{Z}_+$. Assume $\eta(q_0) \geq \eta > 0$. Then for $n \in \mathbb{Z}_+, j \in \mathbb{Z}, t > 0$ with $n + j \geq 0, n > t$, and $\delta_{n,t} < 1$,*

$$|q(t,j) - \tilde{q}_n(t,j)| \leq 2^j C(\eta,\frac{1}{2})\frac{12e^{5t}}{\sqrt{2\pi n}}\left(\frac{2et}{n}\right)^n$$

5. Accuracy and complexity

For $t > 0, n_0 \in \mathbb{Z}$, taking $N = 5+4et+\log_2 \frac{C(\eta,\frac{1}{2})}{\varepsilon}, n = 2N$, we can approximate q by $\tilde{q}_n$ in $[n_0 - \frac{N}{2}, n_0 + \frac{N}{2}]$ with an absolute error $\mathcal{O}(\varepsilon)$. The algorithm requires $\mathcal{O}(\mathbf{n}\log^2\mathbf{n})$ operations, where $\mathbf{n} = t + \log\frac{1}{\varepsilon}$.

References

[1] Bessonov, R.V. and Gubkin, P.V., *Stability of Schur's iterates and fast solution of the discrete integrable NLS.* arXiv preprint arXiv:2402.02434 (2024)

[2] Sylvester, J. and Winebrenner, D.P., *Linear and nonlinear inverse scattering.* SIAM Journal on Applied Mathematics, 59(2), 669-699 (1998)

[3] Tao, T. and Thiele, C., *Nonlinear Fourier analysis.* arXiv preprint arXiv:1201.5129 (2012)

[4] Volberg, A. and Yuditskii, P., *On the Inverse Scattering Problem for Jacobi Matrices with the Spectrum on an Interval, a Finite System of Intervals or a Cantor Set of Positive Length.* Communications in mathematical physics, 226(3), 567-605 (2002)

The quantum circuit model

Massimiliano Incudini

In quantum computing, the de facto standard model of computation is the circuit model, which can be derived by generalizing the classical circuit one.

1. Circuit model

A Boolean circuit is a directed acyclic graph where the vertices represent inputs, outputs, or computational nodes that correspond to logical gates like AND, OR, and NOT. These circuits can be evaluated to compute some Boolean function f, and indeed, any Boolean function can be computed using circuits composed of these basic gates.

Let $L \subseteq \{0,1\}^*$ be a decision problem; here, each binary string $x \in \{0,1\}^*$ is an instance of the problem. We can solve decision problems using the circuit framework. A *circuit family* $\mathcal{C} = \{C_n\}_{n\in\mathbb{N}}$ is a sequence of circuits, one for each input size n. The class of problems admitting a circuit family computing the characteristic function of L is large and includes some problems that are uncomputable, too. The root of this issue is that each input size could potentially have its own unique circuit, leading to an infinite set of circuits that cannot be practically implemented. However, the issue is addressed by requiring there is a single finite program that generates the circuit for all input sizes. A circuit family $\mathcal{C} = \{C_n\}_{n\in\mathbb{N}}$ is *uniformly polynomial* if there exists a Turing machine T that, given input n, can generate a description of C_n using space bounded by $\mathcal{O}(\operatorname{poly}\log n)$. A decision problem L belongs to the complexity class P if there is a uniformly polynomial circuit family $\mathcal{C} = \{C_n\}_{n\in\mathbb{N}}$ such that for every $x \in L$ of size n, $C_n(x) = 1$, and for every $x \notin L$ of size n, $C_n(x) = 0$.

A randomized circuit family $\mathcal{C} = \{C_n\}_{n\in\mathbb{N}}$ is a sequence of circuits, one for each input size n, where each circuit is given $\mathcal{O}(\operatorname{poly} n)$ random bits in addition to the input. A decision problem L is in the complexity class BPP if there exists a uniformly polynomial randomized $\mathcal{C} = \{C_n\}$ such that for all $x \in \{0,1\}^n$, the probability that $Q_n(x) = 1$ is at least $\frac{2}{3}$ if $x \in L$, and at most $\frac{1}{3}$ if $x \notin L$. Furthermore, it is possible to improve these probability thresholds repeating the computation and selecting the output via majority voting.

In the quantum setting, a *quantum circuit family* $\mathcal{C} = \{Q_n\}$ consists of circuits where each gate corresponds to a unitary transformation on a set of qubits. Notable quantum gates include:

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \qquad Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \qquad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

$$H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \qquad S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} \qquad T = \begin{pmatrix} 1 & 0 \\ 0 & e^{-i\pi/4} \end{pmatrix}$$

$$\text{CNOT} = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \otimes \mathbb{I} + \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \otimes X$$

The set $\{H, T, \text{CNOT}\}$ is universal for quantum computation, meaning the composition of these gates can approximately implement any arbitrary large unitary. A decision problem L is in the class BQP if there exists a uniformly polynomial family of quantum circuits $\mathcal{C} = \{Q_n\}$ such that for all $x \in \{0,1\}^n$, the probability that $Q_n(x) = 1$ is at least $\frac{2}{3}$ if $x \in L$, and at most $\frac{1}{3}$ if $x \notin L$.

2. Query model

The query model differs substantially from the circuit model: we do not begin with a quantum state that depends on the input. The input $x \in \{0,1\}^N$, with $N = 2^n$, is provided as a black-box oracle O_x, whose action is $O_x |i\rangle |b\rangle = |i\rangle |b \oplus x_i\rangle$. Each use or call of the oracle is referred to as a *query*. The quantum circuit interleaves the queries to the oracle with standard, non-query operations. This model is used because it is difficult to prove lower bounds on the complexity of computing some functions over explicit input data. In contrast, it is often possible to demonstrate that a certain number of queries are required to compute some given function of the black-box input. The input $x \in \{0,1\}^N$, with $N = 2^n$, is provided as a black-box *phase oracle* $O_x^{\pm}$ if $O_x^{\pm} |i\rangle = (-1)^{x_i} |i\rangle$. These two formats are equivalent, and the circuit implementing the phase oracle given a traditional oracle is $O_x^{\pm} = (\mathbb{I} \otimes XH) O_x (\mathbb{I} \otimes HX)$.

Two problems show a significant separation between classical (deterministic and randomized) and quantum computational capabilities. The first is:

BALANCED-OR-CONSTANT problem
Input: oracle access O_f to a Boolean function $f : \{0,1\}^n \to \{0,1\}$.
Promise: f is either constant or balanced ($|f^{-1}(0)| = |f^{-1}(1)|$).
Output: 0 if f is constant and 1 if f is balanced.

The classical (deterministic) query complexity for the BALANCED-OR-CONSTANT problem is $N/2+1$. Conversely, the quantum query complexity for the BALANCED-OR-CONSTANT problem is 1, achieved using the DEUTSCH-JOZSA algorithm [1]. This involves a system starting at $|0\rangle^{\otimes n}$ and evolving according to the unitary $H^{\otimes n} O_f^{\pm} H^{\otimes n}$, followed by a measurement in the computational basis. Specifically,

$$\begin{aligned} |\psi_0\rangle &= |0^n\rangle , \\ |\psi_1\rangle &= 2^{-n/2} \sum_{i \in \{0,1\}^n} |i\rangle , \\ |\psi_2\rangle &= 2^{-n/2} \sum_{i \in \{0,1\}^n} (-1)^{f(i)} |i\rangle , \\ |\psi_3\rangle &= 2^{-n} \sum_{i \in \{0,1\}^n} (-1)^{f(i)} \sum_{j \in \{0,1\}^n} (-1)^{i \cdot j} |j\rangle . \end{aligned}$$

Notably, the amplitude of the state $|0\rangle$ encodes the solution: the measurement will always yield the all-zero string for a constant function and will always be anything other than the all-zero string for a balanced function. Finally, if we allow a small constant probability of error, the BALANCED-OR-CONSTANT problem can be solved efficiently with a constant number of randomized queries.

The second problem is:

SECRET-STRING problem
Input: oracle access O_f to a Boolean function $f : \{0,1\}^n \to \{0,1\}$.
Promise: $f(x) = (s \cdot x) \bmod 2$.
Output: the n-bit string s.

Note that we have phrased the problem as a search and not a decision problem for simplicity (although a decision variant can be defined). In this case, we need at least n deterministic or randomized queries, since each query returns only one bit of information. However, the quantum query complexity is 1, as demonstrated by the BERNSTEIN-VAZIRANI algorithm. The quantum circuit used is identical to that of the DEUTSCH-JOZSA algorithm. At the penultimate step, we have

$$|\psi_2\rangle = O_f^{\pm} H^{\otimes n} |0^n\rangle = 2^{-n/2} \sum_{x \in \{0,1\}^n} (-1)^{x \cdot s} |x\rangle .$$

Now, recall that $H^{\otimes n} |s\rangle = 2^{-n/2} \sum_{y \in \{0,1\}^n} (-1)^{s \cdot y} |y\rangle$. To recover s, we just need to apply the inverse of $H^{\otimes n}$, which is $H^{\otimes n}$ itself. After this, we obtain $|s\rangle$ in the computational basis with probability 1.

References

[1] D. Deutsch and R. Jozsa *Rapid solution of problems by quantum computation*, Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences **439.1907** (1992), 553–558.

[2] E. Bernstein and U. Vazirani, *Quantum complexity theory*, SIAM Journal on Computing, 26.5 (1997), 1411–1473.

Quantum period finding and Shor's algorithm

Ágoston Kaposi

1. Summary

Shor's algorithm is a fast quantum computing algorithm that factors a natural number into a nontrivial product of two nontrivial natural numbers. If a sufficiently large quantum computer is ever built to factor large numbers, it would compromise many existing cryptographic protocols, especially those relying on the difficulty of factoring, such as RSA, leading to the need for quantum-resistant cryptographic systems. This note essentially covers the same content as Chapter 5 of [2] and wants to examine the main ideas of the algorithm.

2. Reduction from Factoring to Period-Finding

The crucial observation of Shor was that there is an efficient quantum algorithm for the problem of period-finding and that factoring can be reduced to this.

Suppose we want to find factors of the composite number $N > 1$. We may assume N is odd and not a prime power, since those cases can easily be filtered out by a classical algorithm. Consider for a randomly chosen integer $x \in \{2, \ldots, N-1\}$ which is coprime to N. Consider the sequence of $\{1 = x^0 \pmod N),\ x^1 \pmod N),\ x^2 \pmod N),\ \ldots\}$. Let r be the period of this sequence. Assuming N is odd and not a prime power (those cases are easy to factor anyway), it can be shown that with probability $\geq 1/2$, the period r is even and $x^{r/2}+1$ and $x^{r/2}-1$ are not multiples of N[3]. Hence, we have that $x^r \equiv 1(\mod N) \Leftrightarrow (x^{r/2}+1)(x^{r/2}-1) = kN$ for some k. Note that $k > 0$ because both $x^{r/2}+1 > 0$ and $x^{r/2}-1 > 0$. Hence $x^{r/2}+1$ or $x^{r/2}-1$ will share a factor with N with high probability. Thus, the problem of factoring reduces to the period-finding problem.

The period-finding problem: We are given some function $f : \mathbb{N} \to \{0, \ldots, N-1\}$ with the property that there is some unknown $r \in \{0, \ldots, N-1\}$ such that $f(a) = f(b)$ iff $a = b \mod r$. The goal is to find r.

At first glance, this problem could seem to be easy, it is generally believed that classical computers cannot solve period-finding efficiently.

We will show below how we can solve this problem efficiently on a quantum computer, using only $O(\log\log N)$ evaluations of f and $O(\log\log N)$ quantum Fourier transforms. An evaluation of f can be viewed as analogous to the application of a query in the algorithms of the previous chapters.

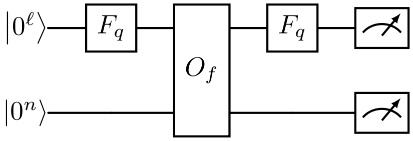
Shor's algorithm finds a factor of N using an expected number of

$$O((\log N)^2(\log\log N)^2 \log\log\log N)$$

gates, which is only slightly worse than quadratic in the input length.

3. Shor's period-finding algorithm

Now we will show how Shor's algorithm finds the period r of the function f , given a ???black-box??? that maps $|a\rangle\,|0^n\rangle \to |a\rangle\,|f(a)\rangle$. Let $q = 2^\ell$ such that $N^2 < q \leq 2N^2$. Then we can implement the Fourier transform F_q using $O((\log N)^2)$ gates. Let O_f denote the unitary that maps $|a\rangle\,|0^n\rangle \to |a\rangle\,|f(a)\rangle$, where the first register consists of ℓ qubits, and the second of $n = \lceil d \log N \rceil$ qubits.



Shor's period-finding algorithm is illustrated in the circuit above. Apply the QFT (or just ℓ Hadamard gates) to the first register to build the uniform superposition $\frac{1}{\sqrt{q}}\sum_{a=0}^{q-1} |a\rangle\,|0^n\rangle$. Now use the ???black-box??? to compute $f(a)$ in quantum parallel $\frac{1}{\sqrt{q}}\sum_{a=0}^{q-1} |a\rangle\,|f(a)\rangle$ Observing the second register gives some value $f(s)$, with $s < r$. Let m be the number of elements of $0, \ldots, q-1$ that map to the observed value $f(s)$ which are in the form of $jr + s$ $(0 \leq j < m$. The second register is ignored, since it collapses to the classical state $|f(s)\rangle$. In the first we have $\frac{1}{\sqrt{m}}\sum_{j=0}^{m-1} |jr + s\rangle$. Applying the QFT again gives

$$\text{(1)} \qquad \frac{1}{\sqrt{m}}\sum_{j=0}^{m-1}\frac{1}{\sqrt{q}}\sum_{b=0}^{q-1} e^{2\pi i \frac{(jr+s)b}{q}}\,|b\rangle = \frac{1}{\sqrt{mq}}\sum_{b=0}^{q-1} e^{2\pi i \frac{sb}{q}}\left(\sum_{j=0}^{m-1} e^{2\pi i \frac{jrb}{q}}\right)|b\rangle$$

We want to see which $|b\rangle$ have amplitudes with large squared absolute value. Those are the b we are likely to see if we now measure. Using the sum of geometric series, we compute:

$$\text{(2)} \qquad \sum_{j=0}^{m-1} e^{2\pi i \frac{jrb}{q}} = \sum_{j=0}^{m-1}\left(e^{2\pi i \frac{rb}{q}}\right)^j = \begin{cases} m & \text{if } e^{2\pi i \frac{rb}{q}} = 1 \\ \frac{1-e^{2\pi i \frac{mrb}{q}}}{1-e^{2\pi i \frac{rb}{q}}} & \text{if } e^{2\pi i \frac{rb}{q}} \neq 1 \end{cases}$$

Easy case: r divides q. Suppose r divides $q = 2^\ell$. For the first case of Eq.(2), note that $e^{2\pi irb/q} = 1$ iff $rb/q \in \mathrm{N}$ iff b is a multiple of q/r. Such b will have squared amplitude equal to $(m/\sqrt{mq})^2 = m/q = 1/r$. Since there are exactly r such basis states b, together they have all the amplitude: the sum of squares of those amplitudes is 1, so the amplitudes of b that are not integer multiples of q/r must all be 0. Thus we are left with a superposition where only the b that are integer multiples of q/r have nonzero amplitude. Observing this final superposition gives some random multiple $b = cq/r$, with c a uniformly random number in $\{0, \ldots, r-1\}$. Thus we get a b such that $b/q = c/r$, where b and q

are known to the algorithm, and c and r are not. There are $\phi(r) \in \Omega(r/\log\log r)$ numbers smaller than r that are coprime to r, so c will be coprime to r with probability $\Omega(1/\log\log r) \geq \Omega(1/\log\log N)$. Accordingly, an expected number of $O(\log\log N)$ repetitions of the procedure of this section suffices to obtain a $b = cq/r$ with c coprime to r. Once we have such a b, we can obtain r as the denominator by writing b/q in lowest terms. Of course, our algorithm does not actually know whether c and r are coprime in some particular run of the algorithm, but it can be efficiently checked classically.

Hard case: r does not divide q. It is actually quite likely that r does not divide q. However, the same algorithm will still yield with high probability a b which is close to a multiple of q/r. Note that q/r is no longer an integer, and $m = \lfloor q/r \rfloor$, possibly $+1$. All calculations up to and including Eq. (2) are still valid. Using $|1 - e^{i\theta}| = 2|\sin(\theta/2)|$, we can rewrite the absolute value of the second case of Eq. (2) to

$$\frac{|1 - e^{2\pi i \frac{mrb}{q}}|}{|1 - e^{2\pi i \frac{rb}{q}}|} = \frac{|\sin(\pi mrb/q)|}{|\sin(\pi rb/q)|} \tag{3}$$

The right-hand side is the ratio of two sine-functions of b, where the numerator oscillates much faster than the denominator because of the additional factor of m. Note that the denominator is close to 0 (making the ratio large) iff b is close to an integer multiple of q/r. For most of those b, the numerator will not be close to 0. Hence, roughly speaking, the ratio will be small if b is far from an integer multiple of q/r, and large for most b that are close to a multiple of q/r. Doing the calculation precisely, one can show that with high probability the measurement yields a b such that $|b/q - c/r| \leq 1/(2q)$ for a random $c \in \{0, \ldots, r-1\}$. Equivalently, $|b - cq/r| \leq 1/2$, so the measurement outcome b will be an integer multiple of q/r rounded up or down to an integer. As in the easy case, b and q are known to us while c and r are unknown.

Because the known ratio b/q is now not exactly equal to the unknown ratio q/r, we cannot just try to find r by writing b/q in lowest terms. However, two distinct fractions, each with denominator $\leq N$, must be at least $1/N^2 > 1/q$ apart. Therefore c/r is the only fraction with denominator $\leq N$ at distance $\leq 1/2q$ from the known ratio b/q. Applying a classical method called ???continued-fraction expansion??? to b/q efficiently gives us the fraction with denominator $\leq N$ that is closest to b/q (see [1]). This fraction must be c/r. Again, c and r will be coprime with probability $\Omega(1/\log\log r)$, in which case writing c/r in lowest terms gives r.

Continued-fraction expansion of a real number x

$$\begin{aligned} a_0 &:= \lfloor x \rfloor, & x_1 &:= 1/(x - a_0) \text{ and for } n > 0: \\ a_n &:= \lfloor x_n \rfloor, & x_{n+1} &:= 1/(x_n - a_n)\ldots \end{aligned}$$

The convergents of the expansion approximate x as follows

$$\text{If } x = [a_0, \ldots, a_n] = \frac{p_n}{q_n} \text{ then } \left| x - \frac{p_n}{q_n} \right| < \frac{1}{q_n^2}.$$

Recall that q_n increases exponentially with n, so this convergence is quite fast. Moreover, p_n/q_n provides the best approximation of x among all fractions with denominator $\leq q_n$:

$$\text{If } n > 1,\ q \leq q_n,\ p/q \neq p_n/q_n,\ \text{then } \left| x - \frac{p_n}{q_n} \right| < \left| x - \frac{p}{q} \right|.$$

References

[1] GH. Hardy, *An introduction to the theory of numbers*, Oxford Science Publication, 1979

[2] Ronald de Wolf, *Quantum Computing: Lecture Notes*, arXiv, https://arxiv.org/abs/1907.09415

[3] Nielsen, Michael A., and Isaac L. Chuang., *Quantum computation and quantum information*, Cambridge university press, 2010.

Simon's algorithm

Miriam Kosik

1. Introduction

Simon's algorithm, invented by Daniel Simon in 1994, was the first quantum algorithm to show an exponential speed-up over the best classical algorithm for a given problem (in this case - for Simon's problem). Its importance also stems from the fact that it served as direct inspiration for Peter Shor to create his famous quantum factoring algorithm.

2. Simon's problem statement

Let us start by presenting the problem considered by Simon. A black box (oracle) is given which implements a function from n-bit binary strings into n-bit binary strings, i.e. $f : \{0,1\}^n \to \{0,1\}^n$ such that for all $x, y \in \{0,1\}^n$:

$$f(x) = f(y) \quad \text{iff} \quad x = y \ \text{ or } \ x = y \oplus s. \tag{1}$$

Here, s denotes an n-bit binary string which is considered to be a hidden property of the oracle and $\oplus$ denotes addition modulo 2.

Note that f can either be 2-to-1 (if $s \neq 000...0$) or 1-to-1 (if $s = 000...0$). The essence of Simon's problem is to determine whether $s = 000...0$ is the only solution that fulfils condition (1), querying the oracle as few times as possible. If the trivial all-zero solution is not the only one, the goal is also to find a non-trivial solution s.

Input: x	Output: $f(x)$
000	000
001	001
010	010
011	011
100	001
101	000
110	011
111	010

Input: $\|x\rangle$	Output: $U_f\,\|x\rangle$
$\|000\,000\rangle$	$\|000\,000\rangle$
$\|001\,000\rangle$	$\|001\,001\rangle$
$\|010\,000\rangle$	$\|010\,010\rangle$
$\|011\,000\rangle$	$\|011\,011\rangle$
$\|100\,000\rangle$	$\|100\,001\rangle$
$\|101\,000\rangle$	$\|101\,000\rangle$
$\|110\,000\rangle$	$\|110\,011\rangle$
$\|111\,000\rangle$	$\|111\,010\rangle$

Table 1. On the left: the action of a classical Simon's oracle $f(x)$ with the hidden string $s = 101$. On the right: the action of a quantum counterpart of f, denoted as U_f.

2.1. **Example of Simon's oracle with $s = 101$.** Let us look at an example of a function $f(x)$ which satisfies Eq. (1). It is presented as a table of values on the left side in Table 1.

Let us consider how one could create a quantum oracle that implements $f(x)$. We need to keep in mind one important fact - quantum operations must be reversible. To ensure this, we make the quantum oracle act on one input register (denoted $|x_i\rangle$) but store the output in a separate register (denoted $|x_o\rangle$):

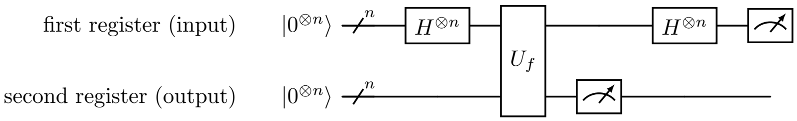
$$f(x) = y \quad \longrightarrow \quad U_f\,|x_i\rangle\,|x_o\rangle = |x_i\rangle\,|x_o \oplus f(x_i)\rangle\,. \tag{2}$$

In this way, the input qubits are never changed and we are guaranteed to get different outputs for different input values. On the right in Table 1, we can see the action of the quantum counterpart of f on a selected subset of all possible inputs. In general, the action on U_f is defined for all possible binary strings as input values in the second register but for simplicity we will always assume that the second register is initialized in an all-zero state.

3. Solving Simon's problem

3.1. **Classical approach.** How would one approach the problem in a classical setting? A simple idea is to query the oracle by providing it with random strings as input, store the input-output pairs and repeat this procedure until you find a repeating output. This is analogous to the *birthday problem* since finding any pair of matching outputs is enough to determine the answer. Hence, on average $\Omega(\sqrt{2^n})$ queries are needed to recover s. One can also show that this is the best classically achievable complexity for this problem, including *randomized* algorithms (see [1]).

3.2. **Quantum algorithm (Simon's algorithm).** The quantum circuit to solve Simon's problem is presented below.



The initial state of the system consists of all qubits in state $|0\rangle$. First, a Hadamard transform is applied to all qubits in the first register:

$$\left|0^{\otimes n}\right\rangle\left|0^{\otimes n}\right\rangle \xrightarrow{H^{\otimes n}} \frac{1}{\sqrt{2^n}}(|0\rangle+|1\rangle)^{\otimes n}\left|0^{\otimes n}\right\rangle = \frac{1}{\sqrt{2^n}}\sum_{x\in\{0,1\}^n}|x\rangle\left|0^{\otimes n}\right\rangle \tag{3}$$

Next, the query to the oracle is made, which causes the second register to change:

$$\frac{1}{\sqrt{2^n}}\sum_{x\in\{0,1\}^n}|x\rangle\left|0^{\otimes n}\right\rangle \xrightarrow{U_f} \frac{1}{\sqrt{2^n}}\sum_{x\in\{0,1\}^n}|x\rangle|f(x)\rangle \tag{4}$$

Next, we measure the second register. This will yield some particular value of $f(x)$ as a result. Let us denote the possible arguments that yield that value x_m and $x_m \oplus s$, and the measured result is then $f(x_m)$.

$$\frac{1}{\sqrt{2^n}}\sum_{x\in\{0,1\}^n}|x\rangle|f(x)\rangle \xrightarrow[\text{2nd register}]{\text{measurement of}} \frac{1}{\sqrt{2}}(|x_m\rangle+|x_m\oplus s\rangle)|f(x_m)\rangle \tag{5}$$

From now on we will ignore the second register and again apply Hadamard transforms to the first n qubits:

$$\begin{aligned}
&\frac{1}{\sqrt{2}}(|x_m\rangle+|x_m\oplus s\rangle)|f(x_m)\rangle \xrightarrow{H^{\otimes n}}\\
&\qquad\frac{1}{\sqrt{2^{n+1}}}\left(\sum_{j\in\{0,1\}^n}(-1)^{x_m\cdot j}|j\rangle+\sum_{j\in\{0,1\}^n}(-1)^{(x_m\oplus s)\cdot j}|j\rangle\right)|f(x_m)\rangle=\\
&\qquad=\frac{1}{\sqrt{2^{n+1}}}\sum_{j\in\{0,1\}^n}(-1)^{x_m\cdot j}\left(1+(-1)^{s\cdot j}\right)|j\rangle|f(x_m)\rangle\,.
\end{aligned}\tag{6}$$

As a result, on the first register we obtain a superposition of states labelled $|j\rangle$, which have a non-zero amplitude if and only if $s\cdot j \bmod 2 = 0$. Hence, measuring state $|j\rangle$ and receiving as a result the some value j gives information about s. Precisely, it gives a random element from the set $\{j \mid s\cdot j \bmod 2 = 0\}$ - we get a linear equation that involves s.

We repeat the quantum part multiple times until we get a system of $n-1$ linear equations. This system will have either one or two solutions. If $f(x)$ is 1-to-1, the only solution will be the all-zero string 000....0. If it is 2-to-1, there will be another non-trivial for solution s. Such a system of linear equations can be solved efficiently using a classical algorithm, e.g. Gaussian elimination, which has runtime O(n^3) for an $n\times n$ system of equations. Hence, the overall complexity of the quantum algorithm is $O(n)$ oracles queries and polynomially many other operations.

References

[1] R. de Wolf, *Quantum Computing: Lecture Notes* [v5], arXiv:1907.09415v5, 2023.

[2] M. A. Nielsen, I. L. Chuang, *Quantum Computation and Quantum Information: 10th Anniversary Edition*, Cambridge University Press, 2011.

Appendix A. Appendix - Simon's oracle implementation

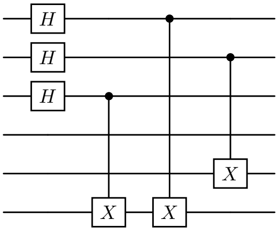
In the oracle computation model, an oracle is an operation that has some property which is hidden from the rest of the world. The term *black box* is also often used equivalently. It is an object that takes some input, returns an output but you have no access or information about what is happening inside. The goal of most quantum algorithms is to query the black box, analyze its outputs and in this way - retrieve the hidden information.

In reality, we cannot look into the oracle. However, to simulate quantum algorithms, one needs to implement a model of the oracle. Below, the implementation steps, an exemplary circuit and an a Qiskit implementation of Simon's oracle are given.

Implementation steps:

- copy the contents of first register onto the second one (using CNOT gates),
- if s is not all equal zero, find the smallest index j for which $s_j = 1$. If $x_j = 0$, then XOR the second register with s. Otherwise, do not do anything.

Circuit for the Simon's oracle hiding string $s = 101$.



Qiskit implementation of the Simon's oracle hiding string s.

```
from qiskit import QuantumCircuit

def simon_oracle(s):

    qc = QuantumCircuit(2*len(s))
    least_1_found = False
    for idx, i in enumerate(s):

        # find the smallest index j for which s_j is 1
        if i == "1" and not least_1_found:
            least_1_idx = idx
            least_1_found = True
        # copy contents of 1st register onto second
        qc.cx(idx, idx + len(s))

    for idx, i in enumerate(s):
        if i == "1" and least_1_found:
            qc.cx(least_1_idx, idx + len(s))

    return qc
```

Quantum Singular Value Transformation

James Berkeley Larsen

1. Introduction

In this report, we introduce the quantum singular value transformation (QSVT). We mainly base the content on these notes by András Gilyén and §3.1-3.3 of [1].

The basic idea of QSVT is to combine methods from quantum signal processing (QSP) with block-encoding to construct a quantum circuit that can perform polynomial transformations of the singular values of arbitrary rectangular matrices. This seemingly abstract task has been shown to provide a unifying framework for all major quantum algorithms, somehow capturing the speedups present in search, phase estimation, and Hamiltonian simulation [2].

2. The Heart of the QSVT

Let $U \in \mathbb{C}^{N\times N}$ be a block-encoding of A as follows:

$$U = \begin{pmatrix} A & B \\ C & D \end{pmatrix},$$

where A and D may be rectangular with different dimensions. Let Π and Π' be the orthogonal projectors such that $A = \Pi' U \Pi$ and $D = (I - \Pi')U(I - \Pi)$. Let us also define the following operators:

$$(1) \quad Z_{\Pi}(\phi) := e^{i\phi}\Pi + e^{-i\phi}(I - \Pi),$$

$$(2) \qquad U_{\Phi} := Z_{\Pi^{(\prime)}}(\phi_d) \cdots U^{\dagger} \cdot Z_{\Pi'}(\phi_3) \cdot U \cdot Z_{\Pi}(\phi_2) \cdot U^{\dagger} \cdot Z_{\Pi'}(\phi_1) \cdot U \cdot Z_{\Pi}(\phi_0),$$

where $\Phi = (\phi_0, \phi_1, \phi_2, \ldots, \phi_d)$. Note that Eq. (1) has a straightforward implementation on a quantum computer using Z rotation gates on an ancillary qubit. The heart of the QSVT is the fact that Eq. (2) applies a polynomial transformation to the singular values of A and D. More concretely,

$$U_\Phi = \begin{cases} \begin{pmatrix} AP(A^\dagger A) & BQ^*(D^\dagger D) \\ CQ(A^\dagger A) & DP^*(D^\dagger D) \end{pmatrix} & \text{for } d \text{ odd} \\ \begin{pmatrix} P(A^\dagger A) & C^\dagger DQ^*(D^\dagger D) \\ B^\dagger AQ(A^\dagger A) & P^*(D^\dagger D) \end{pmatrix} & \text{for } d \text{ even,} \end{cases} \tag{3}$$

for some $P, Q \in \mathbb{C}[x]$ with $\deg(P) \le \frac{d}{2}$ and $\deg(Q) \le \frac{d-1}{2}$. In §3, we will provide an inductive derivation of Eq. (3) to reveal a recurrence relation that defines the polynomials P and Q. We will then conclude in §4 by showing how to choose the angles Φ given some polynomial P.

3. Derivation of Polynomial Recurrence Relations

For the base case of our inductive derivation, notice that when $d = 0$,

$$U_{(\phi_0)} = Z_\Pi(\phi_0) = \begin{pmatrix} e^{i\phi_0} I & 0 \\ 0 & e^{-i\phi_0} I \end{pmatrix},$$

i.e., $P \equiv e^{i\phi_0}$ and $Q \equiv 0$. These (constant) polynomials will also serve as the base case for our recurrence relations.

We let $[P(x)|x = M^{(SV)}]$ denote the application of a polynomial P to the singular values of a matrix $M = \sum_i \sigma_i |\phi_i\rangle\langle\psi_i|$, i.e., $P(M^\dagger M) = [P(x^2)|x = M^{(SV)}]$ and $MP(M^\dagger M) = [xP(x^2)|x = M^{(SV)}]$. We only treat the case for even d, the odd case can be derived using the same steps. Let $P_\Phi(x) := P(x^2)$ and $Q_\Phi(x) := xQ(x^2)$. By the assumed unitarity of U,

$$I = UU^\dagger = \begin{pmatrix} A & B \\ C & D \end{pmatrix} \cdot \begin{pmatrix} A^\dagger & C^\dagger \\ B^\dagger & D^\dagger \end{pmatrix} = \begin{pmatrix} AA^\dagger + BB^\dagger & AC^\dagger + BD^\dagger \\ CA^\dagger + DB^\dagger & CC^\dagger + DD^\dagger \end{pmatrix}. \tag{4}$$

We then can derive that

$$Z_{\Pi'}(-\phi_{d+1}) \cdot U_{(\phi_0,\phi_1,\cdots,\phi_d,\phi_{d+1})} = U \cdot U_\Phi$$

$$= \begin{pmatrix} A & B \\ C & D \end{pmatrix} \cdot \begin{pmatrix} [P_\Phi(x)|x = A^{(SV)}] & C^\dagger \cdot [Q^*_\Phi(x)|x = D^{(SV)}] \\ B^\dagger \cdot [Q_\Phi(x)|x = A^{(SV)}] & [P^*_\Phi(x)|x = D^{(SV)}] \end{pmatrix} \tag{5}$$

$$= \begin{pmatrix} [xP_\Phi(x) + (1-x^2)Q_\Phi(x)|x = A^{(SV)}] & B \cdot [P^*_\Phi(x) - xQ^*_\Phi(x)|x = D^{(SV)}] \\ C \cdot [P_\Phi(x) - xQ_\Phi(x)|x = A^{(SV)}] & [xP^*_\Phi(x) + (1-x^2)Q^*_\Phi(x)|x = D^{(SV)}] \end{pmatrix}, \tag{6}$$

where $\Phi = (\phi_0, \cdots, \phi_d)$, Eq. (5) invokes the inductive hypothesis, and Eq. (6) uses the four matrix identities provided by Eq. (4).

We have thus arrived at the following recurrence relations for the polynomials P_Φ and Q_Φ:

$$\begin{aligned} d = 0: \quad & P_{(\phi_0)} = e^{i\phi_0},\ Q_{(\phi_0)} = 0, \\ d \to d+1: \quad & \\ & \begin{cases} P_{(\phi_0,\phi_1,\cdots,\phi_d,\phi_{d+1})} = e^{i\phi_{d+1}}\left[xP_{(\phi_0,\phi_1,\cdots,\phi_d)}(x) + (1-x^2)Q_{(\phi_0,\phi_1,\cdots,\phi_d)}(x)\right], \\ Q_{(\phi_0,\phi_1,\cdots,\phi_d,\phi_{d+1})} = e^{-i\phi_{d+1}}\left[P_{(\phi_0,\phi_1,\cdots,\phi_d)}(x) - xQ_{(\phi_0,\phi_1,\cdots,\phi_d)}(x)\right]. \end{cases} \end{aligned}$$

Note that P_Φ and Q_Φ have opposite but well-defined parity. Note that for even d, the original P and Q polynomials from Eq. (3) satisfy $P(x) = P_\Phi(\sqrt{x})$ and $Q(x) = \frac{1}{\sqrt{x}}Q_\Phi(\sqrt{x})$ (the case for odd d is similar).

4. Choosing Angles for a Polynomial

In §3, we derived what polynomial transforms would be accomplished by Eq. (3) given some angles Φ. It is important to note that the angles determine the polynomials independent of the choice or dimensions of sub-blocks of U. Specifically, QSP considers the simple case when U is 2×2. For example, if $U = \begin{pmatrix} x & \sqrt{1-x^2} \\ \sqrt{1-x^2} & -x \end{pmatrix}$, we have that $U_\Phi = \begin{pmatrix} P_\Phi(x) & \sqrt{1-x^2}Q_\Phi^*(-x) \\ \sqrt{1-x^2}Q_\Phi(x) & P_\Phi^*(-x) \end{pmatrix}$. Enforcing unitarity of U_Φ in this case gives us an additional requirement that

$$|P_\Phi(x)|^2 + (1-x^2)|Q_\Phi(x)|^2 = 1 \quad \forall x \in [-1,1]. \tag{7}$$

We can now try to reverse the process and derive the angles given two polynomials P and Q with $\deg(P) = \deg(Q) + 1$. Given the recurrence relations P_Φ and Q_Φ must eventually satisfy, the leading coefficients p_d and q_{d-1} must have the same magnitude, so let $\phi_d := \frac{1}{2i}(\ln p_d - \ln q_{d-1})$ (i.e. $e^{2i\phi_d} = p_d/q_{d-1}$). Next, let $\tilde{P}(x)$ and $\tilde{Q}(x)$ be defined as follows:

$$\begin{pmatrix} \tilde{P}(x) \\ \sqrt{1-x^2}\tilde{Q}(x) \end{pmatrix} = \begin{pmatrix} x & \sqrt{1-x^2} \\ \sqrt{1-x^2} & -x \end{pmatrix} \cdot \begin{pmatrix} e^{-i\phi_d} & 0 \\ 0 & e^{i\phi_d} \end{pmatrix} \cdot \begin{pmatrix} P(x) \\ \sqrt{1-x^2}Q(x) \end{pmatrix}.$$

A straightforward matrix computation reveals that the leading coefficients cancel out, resulting in $\deg(\tilde{P}) = \deg(P) - 1$ and $\deg(\tilde{Q}) = \deg(Q) - 1$. Therefore, the process can be iterated d times to find $\phi_{d-1}, ..., \phi_0$, with each iteration decreasing the degree of the target polynomials.

Often, a practitioner only cares about applying a real polynomial transformation $P \in \mathbb{R}[x]$ to the singular values of the A block of U. In this case, one can find a corresponding $\hat{P}_\Phi$ with $\mathcal{R}(\hat{P}_\Phi) = P_\Phi$ and $\hat{Q}_\Phi$ satisfying the stringent form of the recurrence relations if and only if:

$$|P_\Phi(x)| \leq 1 \quad \forall x \in [-1,1]. \tag{8}$$

Therefore, one need only check the condition from Eq. (8) to guarantee that suitable angles can be found for the desired transformation.

References

[1] A. Gilyén, Y. Su, G. Low, and N. Wiebe. Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, STOC '19. ACM, June 2019.

[2] J. Martyn, Z. Rossi, A. Tan, and I. Chuang. Grand unification of quantum algorithms. *PRX Quantum*, 2(4), December 2021.

The $SU(2)$ nonlinear Fourier transform on $\ell^2(\mathbb{Z}\cap[0,\infty))$

Shao Liu

We start by defining the $SU(2)$ nonlinear Fourier transform of sequences in $\ell^2(\mathbb{Z}\cap[0,\infty))$. We then show that the $SU(2)$ nonlinear Fourier transform maps $\ell^2(\mathbb{Z}\cap[0,\infty))$ bijectively to some suitable space. We finally highlight some crucial differences compared to the $SU(1,1)$ case.

We mainly follow [1].

1. Definition of $SU(2)$ nonlinear Fourier transform on $\ell^2(\mathbb{Z}\cap[0,\infty))$

1.1. **Sequences with finite support.** For a sequence $F\colon \mathbb{Z}\to\mathbb{C}$ with finite support, define the matrix valued function G on $\mathbb{C}\cup\{\infty\}$ by the recursive equation

$$G_k(z) = G_{k-1}(z)\frac{1}{\sqrt{1+|F_k|^2}}\begin{pmatrix} 1 & F_k z^k \\ -\overline{F_k} z^{-k} & 1 \end{pmatrix} \tag{1}$$

with the initial condition

$$\lim_{k\to-\infty} G_k(z) = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix},$$

and define the $SU(2)$ nonlinear Fourier transform

$$G(z) = \lim_{k\to\infty} G_k(z) = \begin{pmatrix} a(z) & b(z) \\ -b^*(z) & a^*(z) \end{pmatrix}. \tag{2}$$

The matrix factors in (1) are in $SU(2)$ on $\mathbb{T}$ and hence so is G. In particular,

$$aa^* + bb^* = 1.$$

We write the $SU(2)$ nonlinear Fourier transform of the sequence F on $\mathbb{Z}$ as

$$\overbrace{F} := G = (a,b).$$

Here we identify the row vector (a,b) with the matrix function as in (2). We write $(a_k,b_k) := G_k$. From the multilinear expansion of a, we have $a\in H^2(\mathbb{D}^*)$ and

$$a(\infty) = \prod_{n\in\mathbb{Z}} \left(1+|F_n|^2\right)^{-\frac{1}{2}}. \tag{3}$$

1.2. **One sided square summable sequences.** We use (3) to extend the definition. Let $\mathbf{L}$ be the set of pairs of measurable functions (a,b) on $\mathbb{T}$ such that

$$aa^* + bb^* = 1$$

almost everywhere on $\mathbb{T}$ and a is in $H^2(\mathbb{D}^*)$ with $a(\infty) > 0$. We introduce the following metric on $\mathbf{L}$:

$$\rho((a,b),(c,d)) = \left(\int_{\mathbb{T}} |a-c|^2\right)^{\frac{1}{2}} + \left(\int_{\mathbb{T}} |b-d|^2\right)^{\frac{1}{2}} + |\log(a(\infty)) - \log(c(\infty))| \,.$$

Let $\overline{\mathbf{H}}$ be the set of functions in $\mathbf{L}$ such that b is in $H^2(\mathbb{D})$. $\overline{\mathbf{H}}$ is complete.

Theorem 1. *Let F be a sequence in $\ell^2(\mathbb{Z})$ with support in $[0,\infty)$. The sequence (a_k, b_k) converges in $\mathbf{L}$ to an element (a,b) in $\overline{\mathbf{H}}$. We have*

$$a(\infty) = \prod_{n\geq 0} \left(1 + |F_n|^2\right)^{-\frac{1}{2}} .$$

We call the limit (a,b) in this theorem the nonlinear Fourier transform of F.

2. Properties of the $SU(2)$ nonlinear Fourier transform

2.1. Layer stripping method.

Theorem 2. *Let $(a,b) \in \overline{\mathbf{H}}$. There is a unique $y \in \mathbb{C}$ such that (c,d) is in $\overline{\mathbf{H}}$ where*

$$(c(z), d(z)z) := \left(1 + |y|^2\right)^{-\frac{1}{2}} (1, -y)\,(a(z), b(z))$$

for almost all $z \in \mathbb{T}$. Using this statement, define the functions (a_n, b_n) recursively for $n \geq 0$ by

$$(a_0, b_0) = (a, b)\,,$$
$$(a_{n+1}(z), b_{n+1}(z)z) = \left(1 + |F_n|^2\right)^{-\frac{1}{2}} (1, -F_n)\,(a_n(z), b_n(z))\,,$$

where F_n is the unique number such that $(a_{n+1}, b_{n+1}) \in \overline{\mathbf{H}}$. Then the sequence $\{F_n\}$ is square summable and

$$a(\infty) \leq \prod_{n\geq 0} \left(1 + |F_n|^2\right)^{-\frac{1}{2}} . \tag{4}$$

The sequence produced in this theorem is called the layer stripping sequence of (a,b).

2.2. **Injectivity.** We can construct the inverse nonlinear Fourier transform by means of the “layer stripping method”. Therefore, we have the following theorem.

Theorem 3. *The $SU(2)$ nonlinear Fourier transform maps $\ell^2(\mathbb{Z} \cap [0,\infty))$ injectively to $\overline{\mathbf{H}}$.*

2.3. **Surjectivity.** Concerning surjectivity, it turns out that $\overline{\mathbf{H}}$ is not a suitable space. This is due to the existence of some common inner factor (see below for the definition). More precisely, if $(a, b) \in \overline{\mathbf{H}}$ has a common inner factor, then we have strict inequality in (4). As a result, we need to define a new space. Let $\mathbf{H}$ be the set of functions in $\overline{\mathbf{H}}$ such that a^* and b have no common inner factor in the sense that if a^*g^{-1} and bg^{-1} are in $H^2(\mathbb{D})$ for some inner function g on $\mathbb{T}$, then g is constant. Note that the bar in $\overline{\mathbf{H}}$ has a meaning of a closure rather than a complex conjugation.

Theorem 4. *The $SU(2)$ nonlinear Fourier transform maps $\ell^2(\mathbb{Z} \cap [0, \infty))$ bijectively to $\mathbf{H}$.*

3. Difference compared to the $SU(1,1)$ case

The main difference is that unlike the $SU(1,1)$ case, a^* is not necessarily outer in the $SU(2)$ case. This fact leads to several different behaviours. For example, in the $SU(1,1)$ case, we have the nonlinear Plancherel identity, that is,

$$\int_{\mathbb{T}} \log |a(z)| = -\frac{1}{2} \sum_n \log\left(1 - |F_n|^2\right),$$

while in the $SU(2)$ case, one has the nonlinear Plancherel inequality

$$-\int_{\mathbb{T}} \log |a(z)| \leq \frac{1}{2} \sum_n \log\left(1 + |F_n|^2\right).$$

Moreover, the arguments for the surjectivity are also fundamentally different in these two cases. All of these phenomena are closely related to the existence of the soliton $(g^*, 0)$ in the $SU(2)$ case, where g is some inner function such that $g(0) > 0$.

References

[1] M. Alexis, G. Mnatsakanyan, C. Thiele, *Quantum signal processing and nonlinear Fourier analysis*, Rev Mat Complut **37** (2024), 655–694.

[2] T. Tao, C. Thiele, Y. Tsai, *The nonlinear Fourier transform*, Lecture Note (2012), 1–196.

The SU(1,1) nonlinear Fourier series, square integrable on half line

Ricardo Motta

We present the extension of the nonlinear Fourier transform (NLFT) for $\ell^2(\mathbb{Z}_{\geq 0}, \mathbb{D})$ and prove that it defines a homeomorphism between this space and another space $\mathbf{H}$ that can be explicitly described.

1. Introduction

As established in [1, Lecture 1], given $F = \{F_n\}_{n\in\mathbb{Z}}$ a finite sequence, we recursively define $\begin{pmatrix} a_n & b_n \end{pmatrix} := \begin{pmatrix} a_{n-1} & b_{n-1} \end{pmatrix} T_n$, where

$$T_n := \frac{1}{\sqrt{1-|F_n|^2}} \begin{pmatrix} 1 & F_n z^n \\ \overline{F_n} z^{-n} & 1 \end{pmatrix} \tag{1}$$

and[1] $\begin{pmatrix} a_{-\infty} & b_{-\infty} \end{pmatrix} := \begin{pmatrix} 1 & 0 \end{pmatrix}$. The nonlinear Fourier transform of this sequence F is the pair of functions $(a(z), b(z))$ in the parameter $z \in \mathbb{T}$, where a_n and b_n are equal to a and b for sufficiently large positive n. We denote the NLFT of F as

$$\overbrace{F}(z) = (a(z), b(z)).$$

The goal of this lecture in [1, Lecture 2] is to extend the NLFT for square summable sequences supported on the right half-line, with values in $\mathbb{D}$. The challenge arises from the fact that this transform for finite sequences is an infinite product of transfer matrices (1), whose pointwise convergence is not guaranteed here. To overcome this, we proceed by approximation arguments using Cauchy sequences and some tools of complex analysis.

2. Extension to half-line square summable sequences

The authors' approach draws parallels with classical Fourier analysis, and the first ingredient is the extension of Plancherel's theorem to this context.

Lemma 1. *Let $F = \{F_n\}_{n\in\mathbb{Z}}$ be a finite sequence of elements in the unit disc. Consider the nonlinear transform of F, denoted by $\overbrace{F} = (a, b)$. Then,*

$$\frac{1}{2}\int_{\mathbb{T}} \log\bigl(1 + |b(z)|^2\bigr)\, dz = \int_{\mathbb{T}} \log|a(z)|\, dz = -\frac{1}{2}\sum_n \log\bigl(1 - |F_n|^2\bigr). \tag{2}$$

Similar to the linear Fourier Transform in L^2, the Plancherel identity in (2) will be a fundamental key to define the NLFT of $F \in \ell^2(\mathbb{Z}_{\geq 0}, \mathbb{D})$ as a limit of a Cauchy sequence in some appropriate space $\mathbf{H}$.

To construct this space, we first consider $\mathbf{K}$ to be the space of all measurable functions $(a(z), b(z)) \in SU(1,1)$ on the circle with $\log|a(z)| \in L^1(\mathbb{T})$. We can embed this space into the space $L^1(\mathbb{T}) \times L^2(\mathbb{T}) \times L^2(\mathbb{T})$ by mapping the function (a, b) to the function $(\log|a|, b/|a|, a/|a|)$ and this embedding is indeed injective. We endow the space $\mathbf{K}$ with the inherited metric such that

$$\int_{\mathbb{T}} |\log|a| - \log|a'|| + \left(\int_{\mathbb{T}} |b/|a| - b'/|a'||^2\right)^{1/2} + \left(\int_{\mathbb{T}} |a/|a| - a'/|a'||^2\right)^{1/2}$$

is equal to $\operatorname{dist}((a,b),(a',b'))$, and this makes $\mathbf{K}$ a complete metric space.

Furthermore, we also need to construct $\mathbf{H}$ and $\mathbf{L}$, two subspaces of $\mathbf{K}$. To achieve this, we must introduce the concept of an outer function.

[1] Here $a_{-\infty}$ and $b_{-\infty}$ need to be interpreted as a_n and b_n for sufficiently small n.

Definition 2. *We say g is an outer function in $\mathbb{D}$ if g belongs to the Nevanlinna class*[2] *and there exists a function $G : \mathbb{T} \to [0, \infty)$, with $G \in L^1(\mathbb{T})$, such that*

$$g(z) = \exp\left(\int_0^{2\pi} \frac{e^{i\theta} + z}{e^{i\theta} - z} G(e^{i\theta})\, d\theta\right)$$

for $z \in \mathbb{D}$.

Let $\mathbf{L} \subset \mathbf{K}$ be the subspace of pairs (a, b) where a is the boundary value of an outer function a on the disk $\mathbb{D}^*$ that is positive at ∞, and $\mathbf{H}$ be the subspace of $\mathbf{L}$ where b/a^* is the boundary value of an analytic function in the Hardy space $H^2(\mathbb{D})$. The space $\mathbf{H}$ is closed in $\mathbf{L}$, and $\mathbf{L}$ is closed in $\mathbf{K}$. Moreover, if F is a finite sequence supported on $\mathbb{Z}_{\geq 0}$, then $(a, b) \in \mathbf{H}$.

Lemma 3. *Let F be a sequence in $\ell^2(\mathbb{Z}_{\geq 0}, \mathbb{D})$ and let $F^{(\leq N)}$ denote the truncations to $[0, N]$. Then $(a_N, b_N) = \widehat{F^{(\leq N)}}$ is a Cauchy sequence in $\mathbf{H}$.*

Since $\widehat{F^{(\leq N)}}$ forms a Cauchy sequence in $\mathbf{H}$, we define $\widehat{F}$ as its limit. The distance between the NLFT of these truncations and the full sequence converges to 0, ensuring that (2) holds on all of $\ell^2(\mathbb{Z}_{\geq 0}, \mathbb{D})$.

The theorem below follows from several technical results in [1, Lecture 2].

Theorem 4. *The NLFT is a homeomorphism from $\ell^2(\mathbb{Z}_{\geq 0}, \mathbb{D})$ to $\mathbf{H}$.*

Sketch of Proof. The most challenging aspect of the proof is establishing the bijection, that draws upon key concepts such as the layer-stripping method, the Schur algorithm, the properties of the Möbius transform in the disk, and the Plancherel identity in this context.

For the continuity of the NLFT, the strategy relies on truncating the sequences and controlling the error introduced by this truncation. One term of this approximation is controlled by using the equivalence of norms in finite-dimensional spaces. Therefore, this proof does not guarantee uniform continuity due to the lack of control over the sequence's length when switching from ℓ^2 to ℓ^1 norms.

On the other hand, the proof shows that each term F_n in the inverse NLFT depends continuously on (a, b). This is true for F_0 by the mean formula, and using induction for higher-order terms, continuity is shown via Möbius transformation, which induces a Lipschitz distortion that depends on F_0. To finish, we combine this with an approximation argument and the Plancherel identity. □

Finally, there are higher-order identities of Plancherel type, which arise from calculating higher derivatives of $\log(a^*)$ at 0 using the Cauchy integral formula.

[2] We define the Nevanlinna class as

$$N = \left\{ f \in \mathrm{Hol}(\mathbb{D}) \mid \sup_{r<1} \int_{\mathbb{T}} \log_+ |f(rs)|\, ds < \infty \right\}.$$

Lemma 5. *For $F = \{F_n\}_n$ in $\ell^2(\mathbb{Z}_{\geq 0}, \mathbb{D})$, we have*

$$2\int_{\mathbb{T}} z^{-1} \log|a|(z) = \sum_n \overline{F_n} F_{n+1}$$

and

$$4\int_{\mathbb{T}} z^{-2} \log|a|(z) = -\sum_n (\overline{F_n} F_{n+1})^2 + 2\sum_n \overline{F_n}(1-|F_{n+1}|^2)F_{n+2}.$$

References

[1] T. Tao and C. Thiele, *Nonlinear Fourier Analysis*, arXiv:1201.5129 [math.CA] (2012)

Algorithms for finding QSP angles and infinite quantum signal processing

Hongkang Ni

Let

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \; Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \; W(x) = e^{\mathrm{i}\arccos(x) X} = \begin{pmatrix} x & \mathrm{i}\sqrt{1-x^2} \\ \mathrm{i}\sqrt{1-x^2} & x \end{pmatrix}.$$

Question: Given an even (or odd) polynomial $f(x)$ of degree n that satisfies $\max_{x\in[-1,1]} |f(x)| \leq 1$, how to find the phase factor $\Phi = (\phi_0, \ldots, \phi_n)$ such that

$$\begin{aligned} U(x,\Phi) &:= e^{\mathrm{i}\phi_0 Z} W(x) e^{\mathrm{i}\phi_1 Z} W(x) \cdots e^{\mathrm{i}\phi_{n-1} Z} W(x) e^{\mathrm{i}\phi_n Z} \\ &= \begin{pmatrix} P(x) & Q(x)\sqrt{1-x^2} \\ Q^*(x)\sqrt{1-x^2} & P^*(x) \end{pmatrix}, \end{aligned}$$

satisfy $\mathrm{Im}[P] = f$?

In the previous talk, direct methods for finding QSP phase factors were introduced. In this talk, we will present several iterative approaches and methods based on NLFT. For a more comprehensive overview of current methods, we refer the audience to [6].

1. Iterative methods for phase factor finding

We introduce three iterative methods for finding phase factors: an optimization-based method (1), fixed-point iteration (FPI) (3), and Newton's iteration (4).

In general, the phase factors are not unique. This can be understood by comparing the degrees of freedom–there are $\lfloor \frac{n+2}{2} \rfloor$ degrees of freedom versus $n+1$ parameters in the phase factors. To address this mismatch, we can impose a symmetry constraint on the phase factors, such that

$$(\phi_0, \ldots, \phi_n) = (\psi_d, \ldots, \psi_0, \ldots, \psi_d),$$

where $d = \lceil \frac{n}{2} \rceil$. We call $\Psi = (\psi_0, \ldots, \psi_d)$ the *reduced phase factors.* With a slight abuse of notation, we will write $U(x,\Phi) = U(x,\Psi)$ when Φ is symmetric and corresponds to the reduced phase factors Ψ.

Under this symmetry constraint, the existence of phase factors still holds, and there are only a finite number of phase factor sets that correspond to a given $f(x)$.

Ref. [1] suggests solving the following optimization problem:

$$L(\Psi)=\frac{1}{d+1}\sum_{j=1}^{d+1}\left|\operatorname{Im}\left[P\left(x_j,\Psi\right)\right]-f\left(x_j\right)\right|^2. \tag{1}$$

where $x_j=\cos\left(\frac{(2j-1)\pi}{4(d+1)}\right)$ are the Chebyshev nodes.

Theorem 1 ([2, Cor.7])**.** *If the target polynomial satisfies $\|f\|_\infty\le\frac{\sqrt{3}}{20\pi(d+1)}$, then the optimization based method will converge to Ψ^* with initial guess*

$$\Psi^0=(0,0,\ldots,0). \tag{2}$$

Our next goal is to find a method (and a proper norm of f) that the convergence result can get rid of this d-reliance.

We begin by constructing a map from Ψ to the Chebyshev coefficients of $\operatorname{Im}[P(x,\Psi)]$. We call it as

$$F:\mathbb{R}^{d+1}\to\mathbb{R}^{d+1}:\Psi\mapsto c=(c_0,c_1,\ldots,c_d),$$

where c is defined by the coefficients of the Chebyshev expansion

$$\operatorname{Im}[P(x,\Psi)]=\sum_{j=0}^{d}c_jT_{2j}(x)$$

where $T_{2j}(x)$ is the $2j$-th Chebyshev polynomial. Once the degrees of freedom are balanced, the problem can be reformulated as solving a non-linear equation. Specifically, given the Chebyshev coefficients $\hat{f}$ of f, where $f(x)=\sum_{j=0}^{d}\hat{f}_jT_{2j}(x)$, we aim to find a Ψ such that

$$F(\Psi)=\hat{f}.$$

We remark that by Plancherel's identity, we can also formulate (1) as

$$\underset{\Psi}{\text{minimize}}\ \left\|F(\Psi)-\hat{f}\right\|^2.$$

Notice that F is invariant under padding, which is $F(c_0,\ldots,c_d,0)=(\Psi,0)\in\mathbb{R}^{d+2}$, we can naturally extend F to $\mathbb{R}^\infty$.

Next, we introduce a fixed point iteration method for finding phase factors.

Theorem 2 ([3, Thm.6] FPI iteration method)**.** *For f such that $\|\hat{f}\|_1\le 0.861$, there exists a Ψ such that $F(\Psi)=c$, and the fixed point iteration*

$$\Psi^0=0,\quad \Psi^{t+1}=\Psi^t-\frac{1}{2}\left(F\left(\Psi^t\right)-\hat{f}\right) \tag{3}$$

converges because it is a contraction mapping under the ℓ^1 norm.

The convergence result is obtained by investigating the Jacobian matrix of this map. The straightforward evaluation of F requires $O(d^2)$ time, as it involves computing $\operatorname{Im}[P]$ at all Chebyshev nodes followed by a fast Fourier transform

(FFT). However, recent work in [6] shows that this process can be accelerated to $O(d\log^2 d)$ by a divide-and-conquer approach.

Similarly, we can establish the convergence of the Newton iteration:

$$\Psi^0 = 0, \quad \Psi^{t+1} = \Psi^t - DF(\Psi^t)^{-1}\left(F\left(\Psi^t\right) - \hat{f}\right), \tag{4}$$

which converges for all $\hat{f}$ such that $\|\hat{f}\|_1 \leq$ some constant.

As demonstrated in [4], the Jacobian matrix $DF(\Psi)$ can be computed efficiently in $O(d^2 \log d)$ time, significantly improving upon the naive $O(d^3)$ scaling. Although Newton's method requires solving a dense linear system at each step, its key advantage is that it generally converges in far fewer iterations compared to FPI or the optimization-based method. Furthermore, [4] observes that while FPI may fail for some fully coherent functions f, where $\|f\|_\infty$ approaches 1, Newton's iteration remains robust in these cases.

2. Infinit QSP and the decaying properties of phase factors

Using that $\mathbb{R}^\infty$ is dense in ℓ^1 space and the smoothness of F, we can further extend F as a map $\overline{F} : \ell^1 \to \ell^1$, and it is invertible near the origin. For $\|c\|_1 < 0.902$, there exists a $\Psi \in \ell^1$ such that $\overline{F}(\Psi) = c$.

Theorem 3 ([3, Thm.4] Decay properties of reduced phase factors)**.** *Given a target function f with $\|c\|_1 < 0.902$, and $\Psi^\star := \overline{F}^{-1}(c) = (\psi_0, \psi_1, \ldots) \in \ell^1$, then there exists constants C, C' such that for any n,*

$$C' \sum_{k>n} |c_k| \leq \sum_{k>n} |\psi_k| \leq C \sum_{k>n} |c_k|. \tag{5}$$

3. NLFT-based algorithms for phase factor finding

The NLFT-based method for phase factor finding is the first class of algorithms that is provably stable for any target function. The Riemann-Hilbert-Weiss algorithm proposed in [5] is summarized as follows.

Step 1: Construct the pair (a, b) from the given f, and calculate $\frac{b}{a}$. It turns out that $\frac{b}{a}$ can be expand as Laurent series

$$\frac{b(z)}{a(z)} = \sum_{k=-\infty}^{d} c_k z^k, \tag{6}$$

with pure imaginary coefficients c_k. The coefficients $c_0, c_1, \ldots, c_d$ can be computed using the Weiss algorithm [5, Algorithm 2].

Step 2: The Riemann-Hilbert factorization procedure can be translated as the following algorithm to recover F_k, and hence the phase factors ψ_k. First, construct a Hankel matrix Ξ_k of size $(d+1-k) \times (d+1-k)$ with $(c_k, \ldots, c_d)^T$ as its first column and $(c_d, 0, \ldots, 0)$ as its last row. Then solve the linear system

$$\begin{pmatrix} I & -\Xi_k \\ -\Xi_k & I \end{pmatrix} \begin{pmatrix} \mathbf{a}_k \\ \mathbf{b}_k \end{pmatrix} = \begin{pmatrix} \mathbf{e}_0 \\ \mathbf{0} \end{pmatrix}$$

for $\mathbf{a}_k$ and $\mathbf{b}_k$, where $\mathbf{e}_0$ is the first column of the identity matrix. Finally, compute $\psi_k = \arctan\left(-\mathrm{i}\frac{b_{k,0}}{a_{k,0}}\right)$, where $a_{k,0}$ and $b_{k,0}$ are the first entries of $\mathbf{a}_k$ and $\mathbf{b}_k$, respectively.

Note that this algorithm requires solving a linear system for each ψ_k, making the process computationally expensive when recovering the entire set of phase factors. Its computational cost is $\widetilde{O}(d\eta^{-1}\log\epsilon^{-1} + d^4)$, where $\eta = 1 - \|f\|_\infty$.

A more efficient method is developed in [6]. It is observed that by rearranging the linear system, the vectors $\mathbf{a}_k$ and $\mathbf{b}_k$ can be read from the permuted LU factorization of

$$M := \begin{pmatrix} I & -\Xi_0 \\ -\Xi_0 & I \end{pmatrix}.$$

Moreover, the matrix M has some special structure that admits fast LU factorization with only $O(d^2)$ cost. Therefore, the complexity of the algorithm can be improved to $\widetilde{O}(d\eta^{-1}\log\epsilon^{-1} + d^2)$, while still maintaining stability for any target function f.

References

[1] Yulong Dong, Xiang Meng, K Birgitta Whaley, and Lin Lin, *Efficient phase factor evaluation in quantum signal processing*, Phys. Rev. A **103** (2021), 042419. arXiv:2002.11649

[2] Jiasu Wang, Yulong Dong, and Lin Lin, *On the energy landscape of symmetric quantum signal processing*, arXiv preprint arXiv:2110.04993 (2021).

[3] Yulong Dong, Lin Lin, Hongkang Ni, and Jiasu Wang, *Infinite quantum signal processing*, arXiv preprint arXiv:2209.10162 (2022).

[4] Yulong Dong, Lin Lin, Hongkang Ni, and Jiasu Wang, *Robust iterative method for symmetric quantum signal processing in all parameter regimes*, arXiv preprint arXiv:2307.12468 (2023).

[5] Michel Alexis, Lin Lin, Gevorg Mnatsakanyan, Christoph Thiele, and Jiasu Wang, *Infinite quantum signal processing for arbitrary szegő functions*, arXiv preprint arXiv:2407.05634 (2024).

[6] Hongkang Ni and Lexing Ying, *Fast phase factor finding for quantum signal processing*, arXiv preprint arXiv:2410.06409 (2024).

Riemann-Hilbert problem for rational functions

Kristina Oganesyan

1. Definitions

Let us define the following classes:

$$\begin{aligned}\mathbf{L} &:= \{SU(1,1) - \text{valued measurable } (a,b) :\ a \text{ has outer extension to } \mathbb{D}^*,\\ &\qquad a(\infty) > 0\},\\ \mathbf{H} &:= \{(a,b) \in \mathbf{L} :\ b/a^* \text{ has a holomorphic extension to } \mathbb{D} \text{ which is in } H^2(\mathbb{D})\},\\ \mathbf{H}^* &:= \{(a,b) \in \mathbf{L} :\ b/a \text{ has a holomorphic extension to } \mathbb{D}^* \text{ which is in } H^2(\mathbb{D}^*)\},\end{aligned}$$

and let $\mathbf{H}_0 := \{(a,b) \in \mathbf{H} :\ b(0) = 0\}$, $\mathbf{H}_0^* := \{(a,b) \in \mathbf{H}^* :\ b(\infty) = 0\}$. Note that $\mathbf{L}$ is exactly the class of nonlinear Fourier transforms of ℓ^2 sequences.

We will call a pair (a, b) *rational* if both a and b are rational functions. We also say that a rational function g is *subordinate* to a rational function f on a certain domain if for all points z in the domain with $ord(g, z) > 0$ we have $ord(f, z) \geq ord(g, z)$.

For $(a, b) \in \mathbf{L}$, the factorization

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

is called *a Riemann-Hilbert factorization* if $(a_-, b_-) \in \mathbf{H}_0^*$ and $(a_+, b_+) \in \mathbf{H}$.

2. Properties of rational pairs

First of all, we establish the following important properties of rational pairs (a, b) and their factorizations.

Lemma 1 (Parametrization by b). *a) For a rational function b, there is a unique rational function a such that $aa^* = 1 + bb^*$, a has no zeros and poles in $\mathbb{D}^*$, and $a(\infty) > 0$. This is the unique function a such that $(a, b) \in \mathbf{L}$.*

b) For rational $(a, b) \in \mathbf{L}$, we have $(a, b) \in \mathbf{H}$ if and only if b has no poles in $\mathbb{D}$. (Similarly, we have $(a, b) \in \mathbf{H}^*$ if and only if b has no poles in $\mathbb{D}^*$).

Lemma 2 (Preservation of the class of rational functions). *Let $(a, b) \in \mathbf{L}$ be rational. Given a Riemann-Hilbert factorization*

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

we have that (a_-, b_-) and (a_+, b_+) are also rational.

Lemma 3 (Subordination). *Let $(a, b) \in \mathbf{L}$ be rational. Then a is subordinate to bb^*. For a Riemann-Hilbert factorization*

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

the functions b_- and b_+ are subordinate to b.

In light of the results above, we can reduce the Riemann-Hilbert problem for rational functions to a finite dimensional algebraic problem.

3. Existence and uniqueness of the Riemann-Hilbert factorization for rational functions

With Lemmas 1-3 in hand, we are able to prove our main result.

Theorem 4. *Let $(a, b) \in \mathbf{L}$ be rational. There exists a unique Riemann-Hilbert factorization*

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

with either a) b_+ or b) b_- having no poles on $\mathbb{T}$.

Recall that, according to the triple factorization theorem, for any $(a, b) \in \mathbf{L}$, there is a unique factorization

$$(a, b) = (a_{--}, b_{--})(a_0, b_0)(a_{++}, b_{++})$$

such that $(a_{--}, b_{--}) \in \mathbf{H}_0^*$, $(a_0, b_0) \in \mathbf{H}_0^* \cap \mathbf{H}$, $(a_{++}, b_{++}) \in \mathbf{H}$, and (a_{--}, b_{--}) and (a_{++}, b_{++}) do not have nontrivial Riemann-Hilbert factorizations. Moreover, one can show that for a pair $(a, b) \in \mathbf{H}$, the factor (a_-, b_-) in its Riemann-Hilbert factorization also belongs to $\mathbf{H}$. It therefore follows that the factor (a_+, b_+) in part a) of Theorem 4 coincides with (a_{++}, b_{++}) (and similarly, the factor (a_-, b_-) in part b) of Theorem 4 coincides with (a_{--}, b_{--})).

4. Factorization of the middle factor in the triple factorization for rational functions

For the sake of completeness, we comment (without proof) also on the description of the factorizations of the middle term in the triple factorization.

Theorem 5. *Consider a rational* $(a, b) \in \mathbf{H} \cap \mathbf{H}_0^*$. *Let* $z_j \in \mathbb{T}$, $j = 1, ..., N$ *be the distinct poles of* a *and let* n_j *be the order of the pole* z_j. *Let the nonnegative numbers* $n_j^+, n_j^- \leq n_j$, $j = 1, ..., N$ *be such that for each* j *either*

$$n_j^+ + n_j^- = n_j \quad (\textit{split case})$$

or

$$n_j^+ + n_j^- - 1 = n_j \quad (\textit{shared case}).$$

Assume also that for each j *in the shared case we are given positive numbers* μ_j^+, μ_j^- *such that*

$$\mu_j^+ \mu_j^- = \mu_j,$$

where μ_j *is defined by*

$$\frac{1}{aa^*}(\zeta) = \mu_j (\zeta - z_j)^{n_j} \Big(\frac{1}{\zeta} - \frac{1}{z_j}\Big)^{n_j} + \mathcal{O}(\zeta - z_j)^{2n_j+1}.$$

Then there exists a unique Riemann-Hilbert factorization

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

satisfying

$$ord(a_+, z_j) = n_j^+, \quad ord(a_-, z_j) = n_j^-,$$

and, for j *in the shared case,*

$$A_+ := \frac{a_-^*}{a_+ a^*} = -\mu_j^+ z_j (\zeta - z_j)^{n_j^+ - 1} \Big(\frac{1}{\zeta} - \frac{1}{z}\Big)^{n_j^+} + \mathcal{O}((\zeta - z_j)^{2n_j^+}),$$

$$A_- := \frac{a_+}{a_-^* a} = -\mu_j^- z_j^* (\zeta - z_j)^{n_j^-} \Big(\frac{1}{\zeta} - \frac{1}{z}\Big)^{n_j^- - 1} + \mathcal{O}((\zeta - z_j)^{2n_j^-}).$$

Moreover, all Riemann-Hilbert factorizations are obtained in this way.

References

[1] T. Tao, C. Thiele, *Nonlinear Fourier analysis*, arXiv:1201.5129.

The $SU(2)$ Nonlinear Fourier Transform on $\ell^2(\mathbb{Z})$

Lorenzo Pompili

Recall the following:

- If $g \in H^p(\mathbb{D})$ has modulus 1 a.e. on $\mathbb{T}$, then g is called an inner function.
- Two functions $a, b \in H^2(\mathbb{D})$ are said to have no common inner factor if for every inner function g, both a/g and b/g are $H^2(\mathbb{D})$ functions if and only if g is constant.
- A function $g \in L^\infty(\mathbb{T})$ is called outer if $\log|g| \in L^1(\mathbb{T})$ and $g = e^G$ where $G = \log|g| + i\mathrm{H}(\log|g|)$. Note that if g is outer, then G has an analytic extension to $\mathbb{D}$ with real part bounded above, and hence $g \in H^\infty(\mathbb{D})$.

1. Introduction

We follow [2].

From the previous talk (after using the symmetries of the NLFT), we consider the spaces $\mathbf{H}_{\geq k}$, $\mathbf{H}_{\leq k}$ so that the NLFT maps $\ell^2(\mathbb{Z} \cap [k, \infty))$ to $\mathbf{H}_{\geq k}$ and $\ell^2(\mathbb{Z} \cap (-\infty, k])$ to $\mathbf{H}_{\leq k}$ bijectively.

Analogously to the $SU(1,1)$ case, we can split $F \in \ell^2(\mathbb{Z})$ as $F_- + F_+$, where F_- is supported in $(-\infty, -1]$ and F_+ is supported in $[0, \infty)$, and define the nonlinear Fourier transform of F as

$$(a, b) := (a_-, b_-)(a_+, b_+), \tag{1}$$

where $(a_-, b_-) \in \mathbf{H}_{\leq -1}$, $(a_+, b_+) \in \mathbf{H}_{\geq 0}$ denote the NLFT of $F_\pm$. As in the $SU(1,1)$ case, finding a preimage of (a, b) is equivalent to finding a factorization (1) with $(a_-, b_-) \in H_{\leq -1}$, $(a_+, b_+) \in \mathbf{H}_{\geq 0}$.

For the reader's convenience, we summarize the main findings on the $SU(2)$ NLFT on $\ell^2(\mathbb{Z})$ before diving more precisely into the factorization problem.

- The $SU(2)$ NLFT maps $\ell^2(\mathbb{Z})$ to $\mathbf{L}$.
- The NLFT $\widehat{\ F\ } = (a, b)$ of a sequence $F \in \ell^2(\mathbb{Z})$ satisfies the condition

  $$a^*(0) = \prod_j (1 + |F_j|^2)^{-\frac{1}{2}} > 0$$

  and the nonlinear Plancherel inequality[1]

  $$\sum_n (1 + |F_n|^2) \geq -\int_{\mathbb{T}} \log\big(1 - |b(z)|^2\big)\,,$$

[1] Note that it holds

$$\sum_n (1 + |F_n|^2) = -2\log|a^*(0)|, \qquad \int_{\mathbb{T}} \log\big(1 - |b(z)|^2\big) = 2\int_{\mathbb{T}} \log|a(z)|\,.$$

where equality holds if and only if a^* is outer. In particular, it holds the Szegő condition

$$\int_{\mathbb{T}} \log\bigl(1 - |b(z)|^2\bigr) > -\infty \,. \tag{2}$$

- Unlike the $SU(1,1)$ case, the coefficient a^* is not necessarily outer (in particular, it can have zeros in $\mathbb{D}$).
- If we restrict ourselves to the class $\mathbf{B}$ of $SU(2)$-valued function (a,b) such that a is outer[2] with $a^*(0) > 0$, we have existence of a unique R-H factorization, hence the NLFT is bijective on the set of F such that a^* is outer.
- For any given $b \in L^\infty(\mathbb{T})$ with $\|b\|_{L^\infty} \leq 1$, there exists a unique a such that a^* is outer and (a,b) is a NLFT [2, Theorem 4]. There exist NLFT with same b but different a ($b \equiv 0 \to$ solitons).

2. The factorization problem

Let $\mathbf{B}$ be the set of pairs of measurable functions (a,b) on $\mathbb{T}$ for which a^* is outer with $a^*(0) > 0$, and

$$aa^* + bb^* = 1$$

almost everywhere on $\mathbb{T}$.

Theorem 1 ([2, Theorem 5], Riemann-Hilbert factorization)**.** *Let $(a,b) \in \mathbf{B}$. Then for each $k \in \mathbb{Z}$, there exists a unique factorization*

$$(a,b) = (a_{<k}, b_{<k})(a_{\geq k}, b_{\geq k})$$

with $(a_{<k}, b_{<k}) \in \mathbf{H}_{\leq k-1}$ and $(a_{\geq k}, b_{\geq k}) \in \mathbf{H}_{\geq k}$.

Fix $(a,b) \in \mathbf{B}$ from now on. It is enough to look at the case $k = 0$, and we use the notation $(a_\pm, b_\pm)$ as above. We aim at finding (a_+, b_+). Similarly to the $SU(1,1)$ case, after clever rewriting, one first argues that for any such pair (a_+, b_+) we have that

$$\begin{pmatrix} A \\ B \end{pmatrix} := a_+(\infty) \begin{pmatrix} a_+ \\ b_+ \end{pmatrix} \tag{3}$$

has to solve the equation

$$(\mathrm{Id} + M) \begin{pmatrix} A \\ B \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \tag{4}$$

where

$$M := \begin{pmatrix} 0 & P_{\mathbb{D}^*} \frac{b^*}{a^*} \\ -P_{\mathbb{D}} \frac{b}{a} & 0 \end{pmatrix} .$$

and where $P_{\mathbb{D}}$, $P_{\mathbb{D}^*}$ are the projections from $L^2(\mathbb{T})$ to $H^2(\mathbb{D})$, $H^2(\mathbb{D}^*)$. The equation makes sense if $\inf |a| > 0$, since then M is bounded on the Hilbert space

$$\mathcal{H} := H^2(\mathbb{D}^*) \times H^2(\mathbb{D}) \subset L^2(\mathbb{T}) \times L^2(\mathbb{T}).$$

[2] The outerness of a^* and the relation $aa^* + bb^* = 1$ directly imply the Szegő condition (2).

In that case, it can be shown that finding a solution (A,B) to (4) is equivalent to finding (a_+,b_+) that solves the R-H factorization problem through equation (3)(see the steps in the proof of [1, Theorem 11]). The case of general $(a,b)\in \mathbf{B}$ is more complicated due to the unboundedness of $1/a$, and the above reduction is formal for now.

In [1] it was shown that if $\|b\|_{L^\infty} < 1/\sqrt{2}$, M has norm less than 1, and so $(\mathrm{Id}+M)$ can be inverted as a Neumann series. For general $(a,b)\in\mathbf{B}$, M is not bounded. The main idea of [2] is show that M is skew-adjoint with an appropriate dense domain in $\mathcal{H}$, so that $(\mathrm{Id}+M)^{-1}$ is well-defined, bounded and injective.

Let

$$\mathcal{P}_{\mathcal{H}} = \begin{pmatrix} P_{\mathbb{D}} & 0 \\ 0 & P_{\mathbb{D}^*} \end{pmatrix}$$

denote the $L^2\times L^2$ orthogonal projection onto $\mathcal{H}$. Within the Hilbert space $\mathcal{H}$, define $\mathcal{E}$ to be the subspace of elements $f\in\mathcal{H}$ for which

$$\mathcal{P}_{\mathcal{H}} \begin{pmatrix} 0 & \frac{b^*}{a^*_\eta} \\ -\frac{b}{a_\eta} & 0 \end{pmatrix} f$$

converges weakly in $\mathcal{H}$ as $\eta\to 0$, where a^*_η is defined to be the unique outer function on $\mathbb{D}$ whose boundary values on $\mathbb{T}$ satisfy

$$\log|a_\eta| := \mathbf{1}_{\{|a|>\eta\}}\log|a|\,.$$

It is possible to show that for $(a,b)\in\mathbf{B}$, $\mathcal{E}$ is dense in $\mathcal{H}$ as it contains the dense subspace $\mathcal{D} := aH^2(\mathbb{D}^*)\times a^*H^2(\mathbb{D})$. We define the unbounded operator $M:\mathcal{E}\to\mathcal{H}$ by

$$Mf := \lim_{\eta\to 0}\mathcal{P}_{\mathcal{H}} \begin{pmatrix} 0 & \frac{b^*}{a^*_\eta} \\ -\frac{b}{a_\eta} & 0 \end{pmatrix} f\,,$$

where the above is a weak limit.

Lemma 2. *Let $(a,b)\in\mathbf{B}$. The unbounded operator iM is self-adjoint and hence has real spectrum. In particular, $(\mathrm{Id}+M)^{-1}$ is bounded on $\mathcal{H}$.*

Additional technical work is needed to make the above formal reduction rigorous and prove Theorem 1.

References

[1] M. Alexis, G. Mnatsakanyan, C. Thiele. Quantum signal processing and nonlinear Fourier analysis. *Rev Mat Complut* **37**, 655-694 (2024).

[2] M. Alexis, L. Lin, G. Mnatsakanyan, C. Thiele, J. Wang. Infinite quantum signal processing for arbitrary Szegö functions. Preprint: `arXiv:2407.05634` (2024).

Alternative and multivariable quantum signal processing

Zane Marius Rossi
(joint work with Isaac Chuang)

1. Overview

Quantum algorithms remain difficult to design and interpret; correspondingly, great effort has been spent to not only generate algorithms but formalize the motifs of quantum advantage. These desires have been partially addressed with quantum signal processing (QSP) [1, 2], which allows one to transform the spectrum of large linear operators by tunable polynomial functions using a simple alternating ansatz, in turn unifying and simplifying most known quantum algorithms.

QSP's success follows from a thorough understanding of the permitted maps of type $\mathbb{T} \to \mathrm{SU}(2)$ (from the complex unit circle to the two-dimensional special unitary group) affiliated with QSP's ansatz. A natural extension promotes this study to the multivariable setting, i.e., maps of type $\mathbb{T}^{\otimes n} \to \mathrm{SU}(2)$. Physically, here the computing parity is allowed access to not just one but multiple independent oracles, between which one is allowed to intersperse their own unitaries.

The work of [3] considers the simplest instance of this extension—two commuting, single-qubit oracles—showing that the necessary and sufficient conditions under which a given *multivariable* polynomial transform is achievable are far from obvious, and entangled with results in functional analysis and analytic geometry.

This talk centers on [3] but is steered by insights accumulated over the two years since its publication. Here we briefly place this work in the context of a larger research program on extensions to QSP/QSVT:

(a) **Tethering circuit ansätze and function classes:** The standard map between QSP circuit parameterizations and polynomial transforms is degenerate and awkward. Later work has removed unnecessary d.o.f's, allowing performant phase-finding algorithms. Do similar techniques extend to the multivariable setting? *More broadly, how are constraints on circuit parameterizations taken to constraints on achieved polynomials?*

(b) **Novel constructive and non-constructive theorems for the existence of good circuit parameterizations:** QSP ansatz specifications rely on constructive, inductive proof methods to show the achieveability of specific classes of polynomials.[1] As such, modifying the ansatz, i.e., moving to the multivariable setting, requires overhauling the constructive proof. *Do there exist non-constructive methods to only* ***indirectly*** *show density of ansätze in wider function classes?*

(c) **An *algorithmic resource theory* built around the block-encoding data type:** One way to interpret the incomplete results of [3] is that a given multivariable polynomial transformation of block encodings may require a certain ciruit depth, width, and query-complexity. With the advent of 'generalized polynomial methods' for matrix functions [18], *can we*

[1] While in practice QSP phases are found by iterative, optimization-based methods.

provide tighter upper and lower bounds for general algebraic manipulation of commuting/non-commuting block encodings?

It stands that 'solutions' to the problems raised by multivariable variants of QSP can take multiple forms. Of greatest benefit would be a better understanding of how precisely various naïve extensions of QSP fail, and more diverse techniques for expressing structured products of unitaries. In this way, recent work on QSP as a form of nonlinear Fourier analysis is exciting progress: useful analytic properties of the QSP ansatz related to iterative phase-finding methods are connected to at-first-glance un-physical properties of the analytic extension of the induced polynomial transforms at infinity!

2. Main statements

Multivariable quantum signal processing (M-QSP) as introduced considered in [3] allows the use (in any order) of two possible signal unitaries $W(x_1), W(x_2)$: rotations about a fixed axis on the Bloch sphere by different, unknown angles. As such, while suppressing some underlying complexity discussed afterwards, M-QSP's characterization theorem looks superficially similar to that of standard QSP.

Theorem 1 (M-QSP in the Laurent picture)**.** *Let* $\Phi = \{\phi_0, \ldots, \phi_k\} \in \mathbb{R}^{k+1}$ *and* $s = \{s_1, \ldots, s_k\} \in \{0,1\}^k$. *Then the M-QSP unitary for* (Φ, s) *has form*

$$\begin{aligned} &U_{M\text{-}QSP}(x_1, x_2; \Phi, s) \equiv \Phi[W(x_1), W(x_2)] \\ &= e^{i\phi_0\sigma_z} \prod_{j=1}^{k} W(x_1)^{s_j} W(x_2)^{1-s_j} e^{i\phi_j\sigma_z} = \begin{pmatrix} P & Q \\ -Q^* & P^* \end{pmatrix} \end{aligned} \tag{1}$$

where $W(x) = (1/2)(x + x^{-1})I + (1/2)(x - x^{-1})\sigma_x$ *for* $(x_1, x_2) \in \mathbb{T}^2$ iff $P, Q \in \mathbb{C}[x_1, x_2]$ *are Laurent polynomials in* x_1 *and* x_2 *satisfying the following conditions:*

(1) deg(P) $\preccurlyeq (m, n-m)$ *and deg(Q)* $\preccurlyeq (m, n-m)$ *with* $n = |s|$*, the Hamming weight of s.*

(2) P has even parity under $(x_1, x_2) \mapsto (x_1^{-1}, x_2^{-1})$ *and Q has odd parity under* $(x_1, x_2) \mapsto (x_1^{-1}, x_2^{-1})$.

(3) P has parity m mod 2 under $x_1 \mapsto -x_1$ *and parity* $(m-n)$ *mod 2 under* $x_2 \mapsto -x_2$. *Q has parity m mod 2 under* $x_1 \mapsto -x_1$ *and parity* $(m-n)$ *mod 2 under* $x_2 \mapsto -x_2$.

(4) For all $(x_1, x_2) \in \mathbb{T}^2$, *we have* $|P|^2 + |Q|^2 = 1$.

(5) A statement of equivalent strength to the FRT = QSP condition,[2] *given in* [3]*, holds.*

The difficulty in the above statement is the last condition, namely recovering the inductive property that allows any M-QSP unitary to be written as the product of two unitaries—one with the form $W(x_k)e^{i\phi_j\sigma_z}$ for some k, ϕ_j, and the other an M-QSP unitary with strictly lower degree.

[2] A counterexample to the original conjecture has since been given in [4]. Think of this as a statement guaranteeing that polynomials with unitary extensions permit, at each degree, the inductive step required to iteratively compute QSP phases.

The difficulty in M-QSP comes from two places: (1) ensuring that a given polynomial as a matrix element can be suitably ‘completed’ and embedded in a unitary satisfying the first four conditions of the theorem, and (2) ensuring completions satisfying the first four conditions always permit decompositions into products of oracles and σ_z-generated rotations.

Problem (1) can be answered, albeit opaquely, by appealing to *multivariable Fejér-Riesz theorems* (FRTs) [19]; such theorems specify when positive (or non-negative) multivariable trigonometric polynomials can be expressed as squares.[3] Problem (2), however, originally left up to ‘FRT = QSP’ conjecture in [3], has proven more obstinate; currently there is not even a non-trivial *sufficient* condition for when such unitary completions permit factorizations into products of only oracles and SU(2) rotations.

Against these difficulties, we can either (a) give more abilities to the computing party in an attempt to broaden the set of achievable functions, or (b) provide sufficient conditions such that a given polynomial function permits both (1) unitary completion and (2) phase read-off automatically. Approaches toward this are more specifically enumerated in Sec. 4.

3. A brief guide to related works

While [3] posed initial questions on multivariable QSP variants, a greater impact of its publication manifests in companion papers which address its limitations, examine extensions, and push the theory of QSP/QSVT in new directions. We break these papers into categories for a new reader.

(a) **Restricted and extended ansätze:** Outside of the multivariable setting, numerous works investigate modifications to the QSP circuit, either by restricting the ansatz to improve numerical properties [14], investigating infinite-dimensional variants [13], or allowing larger gate sets to relax certain parity requirements [10].

(b) **Multivariable variants:** Insightful papers have since followed [3] providing counterexamples to the conjecture provided [4, 5] (along with alternative ansätze), as well as LCU-based variants using additional space to achieve similar block encodings [8].

(c) **General, modular block-encoding manipulation:** By relaxing resource models, multivariable polynomials in block-encoded operators can be achieved with incomparable complexities, e.g., through LCU-methods [8] mentioned, or black-box composition of QSP-protocols (first described in [11, 9]) or special supersets of the QSP ansatz (*gadgets*) [6].

(d) **QSP and NLFA:** Finally, recent works [16, 17] have recast QSP as a form of *nonlinear Fourier analysis*, wherein suitably modified results from standard Fourier analysis can be recovered for group-valued functions which, suitably discretized, allow clean proofs of convergence for iterative phase-finding algorithms for wide classes (e.g., Szegő) of target polynomials.

[3] The statement of these theorems (and variants) is involved but, inspired by the univariate case, require that a Toeplitz matrix of Fourier coefficients of the intended function has low rank.

4. Discussion and open problems

Having reviewed some of the main statements of the original instantiation of M-QSP, as well as works since published addressing, extending, or circumventing its methods, we enumerate some open problems/promising avenues.

(a) **Adding abilities to the computing party:** To enable the recovery of the inductive step used for finding M-QSP phases, we could permit additional space, intervening measurement, disparate oracle types, etc. We know LCU and QSP-based techniques can, with additional space and query complexity, achieve arbitrary bounded multivariable matrix polynomials. *Can the required query complexity be lower bounded in terms of desired function class?*

(b) **Identifying a restricted sub-class of achievable functions:** just as symmetrized QSP usefully restricts the class of achievable functions to improve QSP's numerical properties, we can imagine identifying a more simply-describable, but non-trivial, subset of M-QSP-achievable functions. *We know that Chebyshev polynomials and certain algebraic relations among these polynomials are achievable without additional space; can the dictionary of permitted algebraic operations be diversified?*

(c) **Identifying new iterative phase-finding algorithms and nonconstructive existence theorems:** Work in QSP as nonlinear Fourier analysis, as well as symmetrized QSP in general, has yielded exciting algorithms for QSP phase finding of new character, relying on solving structured linear systems. *Do similarly well-performing algorithms exist in the multivariable case, and do they suggest classes of functions in which a given ansatz is dense?*

Ultimately, in this author's opinion, the unifying character of QSP/QSVT seems less a statement of these algorithms generality, and more a suggestion that the quantum algorithmist's toolkit is narrow. The success of recent works rests on their ability to systematically break QSP's basic assumptions while still recovering QSP-like guarantees, in turn building broad families of analytically well-understood parameterized ansätze from well-understood techniques in functional analysis.

References

[1] G. H. Low and I. L. Chuang. *Optimal Hamiltonian simulation by quantum signal processing.* Phys. Rev. Lett., 118:010501, 2017.

[2] G. H. Low and I. L. Chuang. *Hamiltonian simulation by qubitization.* Quantum, 3:163, 2019.

[3] Z. M. Rossi, I. L. Chuang, *Multivariable quantum signal processing (M-QSP): prophecies of the two-headed oracle*, *Quantum* 6, 2022, 811.

[4] Balázs Németh and Blanka Kövér and Boglárka Kulcsár and Roland Botond Miklósi and András Gilyén, *On variants of multivariate quantum signal processing and their characterizations*, arXiv preprint (arxiv:2312.09072), 2023.

[5] Hitomi Mori, Keisuke Fujii, Kaoru Mizuta, *Comment on "Multivariable quantum signal processing (M-QSP): prophecies of the two-headed oracle"*. ArXiv preprint (2310.00918), 2023.

[6] Z. M. Rossi and J. L. Ceroni and I. L. Chuang, *Modular quantum signal processing in many variables*, arXiv preprint (2309.16665), 2023.

[7] András Gilyén, Yuan Su, Guang Hao Low, and Nathan Wiebe. *Quantum singular value transformation and be- yond: exponential improvements for quantum matrix arithmetics.* Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, 2019.

[8] Yonah Borns-Weil, Tahsin Saffat, and Zachary Stier. *A quantum algorithm for functions of multiple commuting Hermitian matrices.* arXiv preprint, (2302.11139), 2023.

[9] Zane M. Rossi and Isaac L. Chuang. *Semantic embedding for quantum algorithms.* Journal of Mathematical Physics, 64(12):122202, 12 2023.

[10] Danial Motlagh and Nathan Wiebe. *Generalized Quantum Signal Processing.* PRX Quantum 5, 020368, 2024.

[11] Kaoru Mizuta and Keisuke Fujii. *Recursive quantum eigenvalue and singular-value transformation: Analytic construction of matrix sign function by Newton iteration.* Phys. Rev. Research 6, 2024.

[12] Dong, Yulong and Lin, Lin and Ni, Hongkang and Wang, Jiasu. *Robust Iterative Method for Symmetric Quantum Signal Processing in All Parameter Regimes.* SIAM Journal on Scientific Computing 46 (5), 2024.

[13] Yulong Dong, Lin Lin, Hongkang Ni, and Jiasu Wang. *Infinite quantum signal processing.* arXiv preprint (2209.10162), 2022.

[14] Jiasu Wang, Yulong Dong, and Lin Lin. *On the energy landscape of symmetric quantum signal processing.* Quantum, 6:850, 2022.

[15] Yulong Dong, Xiang Meng, K. Birgitta Whaley, and Lin Lin. *Efficient phase-factor evaluation in quantum signal processing.* Phys. Rev. A, 103(4), 2021.

[16] Michel Alexis and Gevorg Mnatsakanyan and Christoph Thiele. *Quantum signal processing and nonlinear Fourier analysis.* ArXiv preprint, (2310.12683), 2024.

[17] Michel Alexis, Lin Lin, Gevorg Mnatsakanyan, Christoph Thiele, Jiasu Wang. *Infinite quantum signal processing for arbitrary Szegőfunctions.* ArXiv preprint (2407.05634), 2024.

[18] Ashley Montanaro, Changpeng Shao. *Quantum and classical query complexities of functions of matrices.* ArXiv preprint (2311.06999), 2023

[19] J. Geronimo and Hugo Woerdeman. *Positive extensions, Fejér-Riesz factorization and autoregressive filters in two variables.* Ann. Math. 160, 839–906 (2004).

Block encoding and qubitization

Rahul Sarkar

The material of this section follows the lecture notes in [1, Sections 6, 7.1]. The Hilbert space associated with n-qubits is $\mathbb{C}^N$, where $N = 2^n$. Suppose that we are given a matrix $A \in \mathbb{C}^{N \times N}$, and we will assume that $\|A\|_{\max} := \max_{ij} |A_{ij}| \leq 1$. We will also assume that we are given an oracle O_A to access the entries of A, which achieves the transformation $O_A |0\rangle |i\rangle |j\rangle = (A_{ij} |0\rangle + \sqrt{1 - |A_{ij}|^2} |1\rangle) |i\rangle |j\rangle$. The identity matrix will always be denoted as I and its shape can be deduced from context; however, in cases where confusion may arise, we will use I_k to denote the $k \times k$ identity matrix.

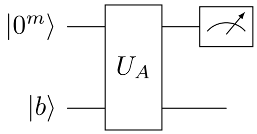
Block encoding. The goal of block encoding is as follows: *can we find $M \geq N$ and a unitary $U_A \in \mathbb{C}^{M \times M}$ with $M \geq N$, such that $U_A = \left(\begin{smallmatrix} A & * \\ * & * \end{smallmatrix}\right)$?* The answer to this question is yes, if and only if $\|A\| \leq 1$, where $\|\cdot\|$ is the matrix 2-norm. For example, suppose we have the full SVD (singular value decomposition) of A given by $A = U\Sigma V^*$. Then there exists a block encoding for $M = 2N$, given by

$$U_A = \begin{pmatrix} U & 0 \\ 0 & I \end{pmatrix} \begin{pmatrix} \Sigma & \sqrt{I-\Sigma^2} \\ \sqrt{I-\Sigma^2} & -\Sigma \end{pmatrix} \begin{pmatrix} V^* & 0 \\ 0 & I \end{pmatrix} = \begin{pmatrix} A & U\sqrt{I-\Sigma^2} \\ \sqrt{I-\Sigma^2}V^* & -\Sigma \end{pmatrix}. \tag{1}$$

In general, for a fixed N (not necessarily a power of 2), this value $M = 2N$ is both necessary and sufficient to block encode any $A \in \mathbb{C}^{N\times N}$ satisfying $\|A\| \le 1$. However, for specific choices of A, one can improve this to $N + \operatorname{rank}(I - A^*A)$, which is a tight lower bound for M. Specializing to the case of qubits, where both N, M are powers of 2, we now introduce the following definition:

Definition 1. *Given an n-qubit matrix A, if we can find $\alpha, \epsilon > 0$, and an $(m+n)$-qubit unitary matrix U_A such that $\|A - \alpha(\langle 0^m| \otimes I_N)U_A(|0^m\rangle \otimes I_N)\| \le \epsilon$, then U_A is called an (α, m, ϵ)-block-encoding of A. When the block encoding is exact with $\epsilon = 0$, the U_A is called an (α, m)-block-encoding of A. The set of all (α, m, ϵ)-block-encoding of A is denoted by $BE_{\alpha,m}(A, \epsilon)$, and we define $BE_{\alpha,m}(A) := BE_{\alpha,m}(A, 0)$.*

Given an $(m + n)$-qubit block encoding U_A of a n-qubit matrix A, let us next consider the following circuit



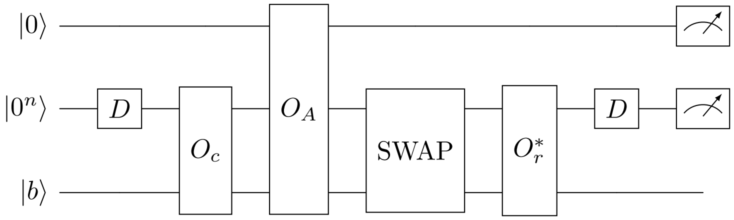
where the top wire represents the m ancilla qubits needed for block encoding, the bottom wire represents n-qubits, and $|b\rangle$ represents any normalized state. A simple calculation shows that the quantum state before the measurement of the top wire is $|0^m\rangle A|b\rangle + |\perp\rangle$, where $|\perp\rangle$ is orthogonal to any state of the form $|0^m\rangle |x\rangle$. If the measurement of the top wire yields the state $|0^m\rangle$, then by the postulates of quantum measurements, the final state obtained is $|0^m\rangle A|b\rangle / \|A|b\rangle\|$, with probability $\|A|b\rangle\|^2$. This probability is upper bounded by $\|A\|^2$, while a lower bound is given by $\sigma_{\min}(A)^2$, where $\sigma_{\min}(A)$ is the smallest singular value of A. As an example, consider the block encoding of a diagonal matrix A, with a simplified oracle O_A that gives access to the diagonal entries of A via the transformation $O_A |0\rangle |i\rangle = (A_{ii} |0\rangle + \sqrt{1 - |A_{ii}|^2} |1\rangle) |i\rangle$. Then we may easily verify that $O_A \in \mathrm{BE}_{1,1}(A)$.

Block encoding of k-sparse matrices. We now discuss two different oracles for block encoding of k-sparse matrices, starting from the oracle for the general case, and then proceeding to a simpler oracle that is valid under additional assumptions on the sparsity structure of the matrix. A matrix $A \in \mathbb{C}^{N\times N}$ being k sparse means that each of its rows or columns have at most k non-zero entries. We will assume here that $k = 2^s$ (otherwise we can simply regard some of the zero entries as non-zeros).

The general oracle. Here we assume that we have access to the following two $2n$-qubit unitary oracles O_r and O_c that achieve the transformations

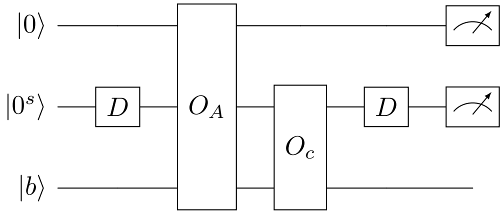
$$O_r \left|\ell\right\rangle \left|i\right\rangle = \left|r(i,\ell)\right\rangle \left|i\right\rangle , \; O_c \left|\ell\right\rangle \left|j\right\rangle = \left|c(j,\ell)\right\rangle \left|j\right\rangle , \tag{2}$$

where $r(i,\ell)$ gives the column index of the ℓ-th non-zero entry in the i-th row, and $c(j,\ell)$ gives the row index of the ℓ-th non-zero entry in the j-th column respectively. It is important to note here that even though $\ell \in [k]$, for this oracle to work, we need to express ℓ using the full n-bit binary string. We also need a diffusion operator D satisfying $D\left|0^n\right\rangle = \frac{1}{\sqrt{k}} \sum_{\ell\in[k]} \left|\ell\right\rangle$, which is realized as $D = I_{2^{n-s}} \otimes H^{\otimes s}$. We assume that we have access to the query oracle O_A mentioned at the beginning of this summary. Then consider the following circuit, which leads to Theorem 2 (the proof is a simple calculation):



Theorem 2. *The above circuit defines an unitary $U_A \in BE_{k,n+1}(A)$.*

The simpler oracle. Under an extra assumption, the circuit above for the general case of sparse matrices, can be simplified significantly, which we now discuss. Let us still assume that $A \in \mathbb{C}^{N\times N}$ is k-sparse. Here we will assume that $c_{j,\ell} := c(j,\ell)$ is the row index of the ℓ-th non-zero entry in the j-th column. We also assume the existence of an unitary O_c that gives access to the locations of elements of A via $O_c \left|\ell\right\rangle \left|j\right\rangle = \left|\ell\right\rangle \left|c(\ell,j)\right\rangle$. The first register here is now only s qubits, unlike in the previous case. Since O_c is unitary, we also need that the map $(\ell,j) \mapsto (\ell, c(\ell,j))$ be injective. We also need the diffusion operator as in the general case $D = H^{\otimes s}$ which achieves the transformation $D\left|0^s\right\rangle = \frac{1}{\sqrt{k}} \sum_{\ell\in[k]} \left|\ell\right\rangle$. Then the following circuit leads to Theorem 3:



Theorem 3. *The above circuit defines an unitary $U_A \in BE_{k,s+1}(A)$.*

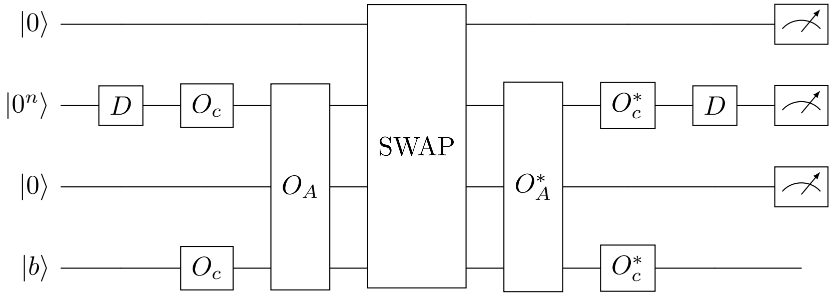
Hermitian block encoding. When A is Hermitian, it is possible to create block-encodings of A that are also Hermitian. This fact leads to the following definition:

Definition 4. *Let U_A be an (α, m, ϵ)-block-encoding of A. If U_A is also Hermitian, then it is called an (α, m, ϵ)-Hermitian-block-encoding of A. When $\epsilon = 0$, it is called an (α, m)-Hermitian-block-encoding. The set of all (α, m, ϵ)-Hermitian-block-encoding of A is denoted by $HBE_{\alpha,m}(A, \epsilon)$, and we define $HBE_{\alpha,m}(A) = HBE_{\alpha,m}(A, 0)$.*

We now provide an oracle to construct Hermitian block encodings of a Hermitian matrix A that is k-sparse. We may exploit the fact that A is Hermitian, and thus we will need only the O_c oracle from the general case above in Eq. (2). We will use the same diffusion operator D as in the general case. The new element in the Hermitian case, that is different from the general (non-Hermitian) case is that we will need two signal qubits (instead of one), and a computational basis state takes the form $|a\rangle\,|i\rangle\,|b\rangle\,|j\rangle\,|a\rangle\,|i\rangle\,|b\rangle\,|j\rangle$, where $a, b \in 0, 1$ and $i, j \in [N]$. We also need a SWAP gate that acts on a computational basis state as $\text{SWAP}\,|a\rangle\,|i\rangle\,|b\rangle\,|j\rangle = |b\rangle\,|j\rangle\,|a\rangle\,|i\rangle$. The oracle O_A to query matrix entries will achieve the following transformation (it does not change the first signal qubit)

$$O_A\,|i\rangle\,|0\rangle\,|j\rangle = |i\rangle\left(\sqrt{A_{ij}}\,|0\rangle + \sqrt{1 - |A_{ij}|}\,|1\rangle\right)|j\rangle\,. \tag{3}$$

With this we may now consider the following circuit that produces a Hermitian block encoding of A leading to Theorem 5:



Theorem 5. *The above circuit defines an unitary $U_A \in HBE_{k,n+2}(A)$.*

Qubitization. Hermitian block encoding serves as an entry point to discuss qubitization. Here we discuss the idea very briefly. Suppose that $U_A \in \text{HBE}_{k,m}(A)$. Then the eigenvalues of U_A are either 1 or -1. Let us denote the associated eigenspaces as Ξ_+ and Ξ_-, and thus we have the orthogonal direct sum representation $V := \mathbb{C}^{2^{n+m}} = \Xi_+ \oplus \Xi_-$ (recall that U_A is a linear operator on $\mathbb{C}^{2^{n+m}}$). Since U_A is Hermitian and unitary, it implies that $U_A^2 = I$, and thus every Krylov subspace of U_A has dimension at most 2. Now take any $x \in V$, and then note that $x = x_+ + x_-$, where x_+ and x_- are the projections of x onto Ξ_+ and Ξ_- respectively. The Krylov subspace $\mathcal{K}(U_A, x) := \text{span}\{x, U_A x, U_A^2 x, \dots\}$ either has dimension 1 or 2, corresponding to the cases whether x is an eigenvector of U_A or not. It also easily follows that $\mathcal{K}(U_A, x) = \text{span}\{x_+, x_-\}$.

Next, denote as W the subspace of V spanned by all vectors $|0^m\rangle\,|x\rangle$, where $|x\rangle \in \mathbb{C}^{2^n}$, and let $W_\perp$ be the orthogonal complement; that is, we have the orthogonal direct sum $V = W \oplus W^\perp$. By the block encoding construction, A is an Hermitian operator acting on W. Pick an orthonormal set $\{v_1, v_2, \dots, v_N\}$ (here $N = 2^n$) of eigenvectors of A than span W, and we split these eigenvectors into disjoint two sets $\mathcal{V}_1$ and $\mathcal{V}_2$ as follows: for every eigenvector in $\mathcal{V}_1$ (resp. $\mathcal{V}_2$), the corresponding eigenvalue has absolute value strictly less than 1 (resp. equal to 1). Denote the spans of $\mathcal{V}_1$ and $\mathcal{V}_2$ as V_1 and V_2 respectively. Then we again have the orthonormal direct sum $W = V_1 \oplus V_2$.

Now take take any $x \in W$, and apply U_A to it to get $U_A x = y_W + y_W^\perp$, where y_W and $y_W^\perp$ are the projections of $U_A x$ on W and $W^\perp$ respectively. Since, U_A is a block encoding, by definition we have $y_W = Ax$, and thus we have $U_A x = Ax + y_W^\perp$. It then follows by taking norms that $\|x\|^2 = \|U_A x\|^2 = \|Ax\|^2 + \left\|y_W^\perp\right\|^2$. Let us first consider the uninteresting case: take any $v \in \mathcal{V}_2$. Then $\|Av\| = \|v\|$, which implies $U_A v = Av = \pm v$, and thus $\operatorname{span}\{v\}$ is an invariant subspace of U_A. The more interesting case is when we take $v \in \mathcal{V}_1$, and suppose $Av = \lambda v$ with $|\lambda| < 1$, and in this case we have $\|Av\| < \|v\|$. Thus it follows that $U_A v = Av + y_W^\perp$ (again $y_W^\perp$ is the projection of $U_A v$ onto $W^\perp$), where $y_W^\perp \neq 0$. Thus $\mathcal{K}(U_A, v)$ has dimension 2 and moreover we have the following:

$$\mathcal{K}(U_A, v) = \operatorname{span}\{v_+, v_-\} = \operatorname{span}\{v, y_W^\perp\}. \tag{4}$$

In this way, every eigenvector $v \in \mathcal{V}_1$ of A, gets paired with an element of W and together they span a 2-dimensional subspace of V that is invariant under the action of U_A. This process is called “qubitization”. The end result is that if $\mathcal{V}_1 = \{v'_1, \dots, v'_r\}$, where all the eigenvectors are orthonormal, then there exists a set of orthonormal elements $\{y'_1, \dots, y'_r\} \subseteq W^\perp$, such that $\operatorname{span}\{v'_i, y'_i\}$ are invariant subspaces of U_A, for every $i = 1, 2, \dots, r$.

References

[1] L. Lin, *Lecture notes on quantum algorithms for scientific computation*, arXiv:2201.08309 (2022).

[2] A. Gilyén, Y. Su, G.H. Low, N. Wiebe, *Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics*, Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing (2019), 193–204.

QSP and NLFT

Miquel Saucedo

1. Connection between QSP and NLFT

1.1. Definition of QSP.

Proposition 1 (Pauli matrices and some of their properties)**.** *Set*

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \text{ and } Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}.$$

The following properties hold:

(1) The matrix

$$M = 2^{-\frac{1}{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

diagonalizes X and satisfies $M = M^{-1}$, that is,

$$MXM = Z.$$

(2) For $\theta \in \mathbb{R}$,

$$e^{i\theta X} = \begin{pmatrix} \cos\theta & i\sin\theta \\ i\sin\theta & \cos\theta \end{pmatrix} \text{ and } e^{i\theta Z} = \begin{pmatrix} e^{i\theta} & 0 \\ 0 & e^{-i\theta} \end{pmatrix}.$$

Let $\Psi = (\psi_k)_{k\in\mathbb{N}}$ with $\psi_k \in (-\frac{\pi}{2}, \frac{\pi}{2})$. We now define the QSP for Ψ.

Definition 2. *Set $x = \cos(\theta)$ with $\theta \in [0, \frac{\pi}{2}]$ and define recursively the symmetric QSP*

$$U_0(\Psi, x) = e^{i\psi_0 Z}, \quad U_d(\Psi, x) = e^{i\psi_d Z} e^{i\theta X} U_{d-1}(\Psi, x) e^{i\theta X} e^{i\psi_d Z}. \tag{1}$$

The QSP of Ψ is the limit imaginary part of the upper-left entry of U_d as $d \to \infty$.

1.2. **Definition of SU(2) NLFT and connection with QSP.** First we recall the definition of NLFT. Let $F = (F_n)_{n\in\mathbb{Z}}$ with $F_n \in \mathbb{C}$ and finite support (extensions to larger spaces have been discussed in previous lectures).

Definition 3 (NLFT)**.** *Define recursively*

$$G_k(z) = G_{k-1}(z) T_k(F_k, z)$$

with the initial condition $G_{-\infty} = I$, and where

$$T_k(F_k, z) = \frac{1}{\sqrt{1 + |F_k|^2}} \begin{pmatrix} 1 & F_k z^k \\ -\overline{F_k} z^{-k} & 1 \end{pmatrix}.$$

We write $\overbrace{F} = (a, b)$, where

$$G_\infty(z) = \begin{pmatrix} a(z) & b(z) \\ -b^*(z) & a^*(z) \end{pmatrix}.$$

We now describe the connection between QSP and NLFT.

Theorem 4. *Let $F = (F_n)_{n\in\mathbb{Z}} = (i\tan\psi_{|n|})_{n\in\mathbb{Z}}$. Set $x = \cos(\theta)$ and $z = e^{2i\theta}$. Then, for $n \in \mathbb{N}$*

$$G_n(F, z) = e^{-in\theta Z} M U_n(\Psi, x) M e^{-in\theta Z}.$$

Therefore $\overbrace{F}(z) = (a(z), b(z))$ and , because of the symmetries of F_n,

$$b(z) = -b^*(z),$$

and

$$(a(z), b(z)) = (a^*(z^{-1}), -b^*(z^{-1})).$$

Hence, b is purely imaginary, even and $\lim_n \operatorname{Im}(U_n(\Psi, x)_{1,1}) = b(z)$.

Proof. Observe that

$$T_k(F_k,z) = \begin{pmatrix} \cos\psi_k & i\sin\psi_k e^{2ki\theta} \\ i\sin\psi_k e^{-2ki\theta} & \cos\psi_k \end{pmatrix} = e^{ik\theta Z}e^{i\psi_k X}e^{-ik\theta Z}.$$

Hence,

$$\begin{aligned} G_n(F,z) &= e^{-in\theta Z}e^{i\psi_n X}e^{in\theta Z}\cdots e^{-i2\theta Z}e^{i\psi_2 X}e^{i2\theta Z}e^{-i\theta Z}e^{i\psi_1 X}e^{i\theta Z}\times \\ &\times\ e^{i\psi_0 X}e^{i\theta Z}e^{i\psi_1 X}e^{-i\theta Z}e^{i2\theta Z}e^{i\psi_2 X}e^{-i2\theta Z}\cdots e^{in\theta Z}e^{i\psi_n X}e^{-in\theta Z} \\ &= e^{-in\theta Z}e^{i\psi_n X}e^{i\theta Z}\cdots e^{i\psi_0 X}e^{i\theta Z}e^{i\psi_1 X}e^{i\theta Z}e^{i\psi_2 X}e^{i\theta Z}\cdots e^{i\theta Z}e^{i\psi_n X}e^{-in\theta Z} \\ &= e^{-in\theta Z}MU_d(\Psi,x)Me^{-in\theta Z}. \end{aligned}$$

□

Hence, the problem of finding angles which encode f becomes finding F such that $\overbrace{F} = (a,b)$ for a given even, imaginary b and some a.

2. Given b, find NLF coefficients

Recall that, by previous lectures, given *suitable* (a,b) (more precisely $|a^*|^2+|b|^2 = 1$ on $\mathbb{T}$, $a^* \in H^2(\mathbb{D})$, $\inf_{z\in\mathbb{D}}|a^*(z)|^2 > \frac{1}{2}$ and $a^*(0) > 0$) we can find the NLF coefficients by applying layer-stripping to the Riemann Hilbert factorization. The problem is now for a given b to find such an a.

2.1. Finding a from b.

Theorem 5. *Let b be a measurable function on $\mathbb{T}$ with $\sup b^2 < \frac{1}{2}$. Then there exists an $a^* \in H^2(\mathbb{D})$ such that $a^*(0) > 0$, $|a^*|^2+|b|^2 = 1$ on $\mathbb{T}$ and $\inf_{z\in\mathbb{D}}|a^*(z)|^2 > \frac{1}{2}$.*

Proof. Let, for $z \in \mathbb{T}$,

$$M(z) := \frac{1}{2}\log\bigl(1-|b(z)|^2\bigr).$$

Since M is real,

$$N(z) := M(z) + iH(M)(z) \sim \hat{M}(0) + \sum_{n=1}^{\infty} 2\hat{M}(n)z^n \in L^2(\mathbb{T})$$

where the Hilbert transform $H(M)$ is also real. Thus, N can be extended to a holomorphic function in the disc via the formula (equivalently by convolution with the Poisson kernel)

$$N(re^{i\theta}) = P_r * N(\theta) = \hat{M}(0) + \sum_{n=1}^{\infty} 2\hat{M}(n)(re^{i\theta})^n.$$

Set

$$a^*(z) = \exp(N(z)),$$

it is a holomorphic function with radial limits at $z \in \mathbb{T}$ satisfying $|a^*(z)|^2 = 1-|b(z)|^2$ almost everywhere and $a^*(0) = e^{\hat{M}(0)} > 0$. It is outer because

$$\log|a^*(re^{i\theta})| = \operatorname{Re} N(re^{i\theta}) = P_r * M(\theta) = P_r * \log|a^*(\theta)|.$$

Finally, by Jensen's inequality, for any real λ,

$$|a^*(re^{i\theta})|^\lambda \leq P_r * |a^*(\theta)|^\lambda.$$

Since on the boundary $\frac{1}{2}+\varepsilon \leq |a^*|^2 \leq 1$, we have $a^* \in H^2(\mathbb{D})$ and $\inf_{z\in\mathbb{D}} |a^*(z)|^2 > \frac{1}{2}$.

□

Remark 6. *We have*

$$\begin{aligned} -\frac{1}{2}\sum_{n\in\mathbb{Z}} \log\left(1+\tan^2(\psi_{|n|})\right) &= \log(a^*(0)) = \frac{1}{2}\int_{\mathbb{T}} \log\bigl(1-|b(\theta)|^2\bigr)d\theta \\ &= \frac{1}{\pi}\int_0^\pi \log\bigl(1-f(x)^2\bigr)\frac{dx}{\sqrt{1-x^2}}. \end{aligned}$$

Hence the QSP of the truncated series converges in the

$$\int_0^\pi \log\bigl(1-f(x)^2\bigr)\frac{dx}{\sqrt{1-x^2}}$$

sense.

3. Summmary: how to find QSP angles

Given $f : [0,1] \to \mathbb{R}$ to find ψ: extend it to an even function, let $b(z) = if(x)$ with $x = \cos\theta$, $\theta \in [0,\pi]$ and $z = e^{2i\theta}$. Find the outer a as in Theorem 2.1. Let F be the NLF coefficients of (a,b), since b is imaginary and even, F_n is even and imaginary. The angles of the QSP are $\psi_n = \arctan(-iF_n)$.

References

[1] M. Alexis, G. Mnatsakanyan and C. Thiele, *Quantum signal processing and nonlinear Fourier analysis*, Rev. Mat. Complut (2024).

Direct methods for finding QSP angles

Philipp Schleich

1. Existence of phase factors

The QSP theorem says that given a target polynomial $f \in \mathbb{R}[x]$ that satisfies $\deg(f) = n$, definite parity$(f) = n \mod 2$, and $\|f\|_\infty < 1$, there exists a sequence of *phase factors* $\Phi = (\phi_0, \phi_1, \ldots, \phi_n) \in [-\pi,\pi)^{n+1}$ so that

$$f(x) = g(x;\Phi) := \mathrm{Re}\bigl(\langle 0|U(x;\Phi)|0\rangle\bigr), \quad x \in [-1;1] \quad \text{with} \tag{1}$$

$$U(x;\Phi) := e^{\mathrm{i}\phi_0 Z} e^{\mathrm{i}\arccos(x) X} e^{\mathrm{i}\phi_1 Z} e^{\mathrm{i}\arccos(x) X} \cdots e^{\mathrm{i}\phi_{n-1} Z} e^{\mathrm{i}\arccos(x) X} e^{\mathrm{i}\phi_n Z} \tag{2}$$

$$\simeq \begin{pmatrix} P(x) & \mathrm{i}Q(x)\sqrt{1-x^2} \\ \mathrm{i}Q^*(x)\sqrt{1-x^2} & P^*(x) \end{pmatrix}, \quad P,Q \in \mathbb{C}[x]. \tag{3}$$

The proof of this result, i.e., the existence of such phase factors, is often done in a constructive manner; e.g., [2, 1]. This means it naturally provides an algorithms to find a set of phase factors. In the following, we motivate this. Instead of providing the formal proof, we repeat the illustrative sketch from [3].

It is convenient to think about the following liftings: $t \in [-\pi, \pi] : f(t) := f(x = \cos t)$, $z \in U(1) : f(z = e^{\mathrm{i}t}) := f(t)$. Throughout this report we will to the variables x, t, z according to these definitions. Then, we can rewrite (3) as

$$(4) \qquad \begin{pmatrix} P(x) & \mathrm{i}Q(x)\sqrt{1-x^2} \\ \mathrm{i}Q^*(x)\sqrt{1-x^2} & P^*(x) \end{pmatrix} \overset{x\mapsto t}{\simeq} \begin{pmatrix} p(t) & q(t) \\ q^*(t) & p^*(t) \end{pmatrix} \overset{t\mapsto z}{\simeq} \begin{pmatrix} p(z) & q(z) \\ q^*(z) & p^*(z) \end{pmatrix}.$$

Next, consider $p(z), q(z)$ in the with $\deg(p) = n, \deg(q) = n-1$ in a monomial basis as Laurent polynomials as $p(z) = \sum_j p_j z^j$ and $q(z) = \sum_j q_j z^j$. This expansion is equivalent to a Fourier decomposition in the t-variable, namely $p(t) = p_n e^{\mathrm{i}nt} + \ldots$, and similarly for q.

An initial choice is given by the ratio of leading coefficients $e^{2\mathrm{i}\phi_n} = p_n/q_{n-1}$. Considering $\left(p^{(1)}(z) \; q^{(1)}(z) \right) = \left(p(z) \; q(z) \right) e^{-\mathrm{i}\phi_n Z} e^{-\mathrm{i}tX}$, the previous leading-order term cancels and in $\left(p^{(1)}(z) \; q^{(1)}(z) \right)$ is now the former $n-1$st, successively decreasing the degree. Intuitively, we "split off" factors from the right in (2). Then, set $e^{2\mathrm{i}\phi_{n-1}} = p_{n-1}/q_{n-2}$, and so forth, until we arrive at $\left(p^{(n)}(z) \; q^{(n)}(z) \right)$ to retrieve ϕ_0. There are some minor subtleties regarding even vs. odd degree at the least iteration, which we will skip here and refer to [2].

While this construction seems quite straightforward, we *do not* a priori know $p(z), q(z)$, but only the target $f = \mathrm{Re}(p)$. Ref. [4] calls this the "completion" step, and the above paragraph "decomposition".

2. Factorization approaches: Finding complementary polynomials

Following the notation in [1, 3], we introduce the real Laurent polynomials

$$(5) \quad a(z) = \mathrm{Re}\big(p(z)\big), \quad b(z) = \mathrm{Re}\big(q(z)\big), \quad c(z) = \mathrm{Im}\big(p(z)\big), \quad d(z) = \mathrm{Im}\big(q(z)\big).$$

To successfully carry out the completion step of finding the QSP angles, we need to find expressions for $b(z), c(z), d(z)$, having only prior information about $a(z)$.

This set of polynomials is constrained by normalization, $a(z)^2 + b(z)^2 + c(z)^2 + d(z)^2 = 1$, and parity. Factorizing the Laurent polynomial $F(z) = 1 - a(z)^2 - b(z)^2 = c(z)^2 + d(z)^2$ is sufficient to fully determine a, b, c, d and thus find Φ. Restricting to symmetric phase factors as done in [2] implies that $b(z) = 0$ and $Q \in \mathbb{R}[x]$ in (3).

The polynomial $F(z)$ has $2n$ roots (at most, for this talk let us assume there are) taking into account multiplicity, and we know that they appear in pairs (r_j, r_j^{-1}) so that for each r_j inside the unit disc, there is a r_j^{-1} outside the unit disc. This means we can write $F(z) = \alpha \prod_{j=1}^{2n} (r_j - z)(r_j - z^{-1})$.

Then, we look at a factor of $F(z)$ of the following form, $e(z) := z^{-n} \prod_{|r_j|<1} (z - r_j)$, with the goal to find a factorization so that all of the corresponding roots come from within the unit disc. This allows us to define the "complementary polynomials" $b(z), c(z)$ via

$$(6) \qquad c(z) = \sqrt{\alpha}\frac{e(z)+e(1/z)}{2}, \quad d(z) = \sqrt{\alpha}\frac{e(z)-e(1/z)}{2\mathrm{i}}, \quad \alpha = \frac{1-a(z)^2-b(z)^2}{e(z)e(1/z)} > 0.$$

2.1. **Factorization via root-finding.** The most straightforward way to construct the complementary polynomials is applying a root-finding algorithm to $F(z)$, with complexity at most cubic in n [1, 4]. However, there is a draw-back that such root-finding approaches of high-degree polynomials tend to be numerically unstable. The algorithm in [1] is stable only assuming high-precision arithmetic with $\Omega(\operatorname{poly} n \log(n/\varepsilon))$ bits of precision. The reason for this lies in the way that the decomposition step is carried out: The leading order coefficients (in our notation above that is p_n, q_{n-1}) are much smaller in norm compared to the lower-order ones and high precision overall needed so that the numerical error remains bounded. In comparison, [4] introduce a different way to approach the decomposition step they call “halfing”; while not theoretically shown, numerically they fare well with usual machine precision.

2.2. **Alternative factorization via Prony's method.** Ying [3] has proposed an alternative way to factorize $F(z)$ using Prony's method. In order to apply Prony's method, we need to first make the following observation: The roots of $F(z)$ are the poles of its reciprocal, $g(z) := \frac{1}{1-a(z)^2-b(z)^2}$. Then it is possible to identify the Fourier transform through a contour integral, with k a negative integer and γ the counter-clockwise boundary of the unit disk,

$$\tag{7} \tfrac{1}{2\pi \mathrm{i}} \int_\gamma \tfrac{g(z)}{z^k} \tfrac{\mathrm{d}z}{z} = \tfrac{1}{2\pi \mathrm{i}} \int_0^{2\pi} g(t) e^{-\mathrm{i}kt} \mathrm{i}\mathrm{d}t = \tfrac{1}{2\pi} \int_0^{2\pi} g(t) e^{-\mathrm{i}kt} \mathrm{d}t = \hat{g}_k.$$

This relies on $g(z)$ being meromorphic and thus $g(z) = \sum_{r_j} \frac{w_j}{r_j - z} + \text{const.}$

Now, Prony's method can be used on the Fourier coefficients $\hat{g}_k$ in order to recover $e(z)$. To that end, consider the semi-infinite vector

$$\hat{g}_- := \begin{pmatrix} \hat{g}_{-1}, & \hat{g}_{-1}, & \cdots \end{pmatrix}^T = \begin{pmatrix} -\sum_{|r_j|<1} w_j r_j^0, & -\sum_{|r_j|<1} w_j r_j^1, & \cdots \end{pmatrix}^T.$$

Then, let S be a shift operator so that for any $|r_j| < 1, \quad (S - r_j)\left(r_j^0, r_j^1, \ldots \right)^T = 0.$ and we can carry this over to $\prod_{|r_j|<1} (S - r_j) \hat{g}_- = 0$. To express $g(z)$, the choice of $b(z)$ is a degree of freedom. Ref. [3] uses $b(z) = b_n \sin(nt) + b_{n-2} \sin((n-2)t) + \ldots$, while the coefficients b_j are chosen randomly. This guarantees that the roots of $\prod_{|r_j|<1} (z - r_j)$ are disjoint almost surely and $\deg(e(z)) = 2n$. Further, we can write $\prod_{|r_j|<1} (z - r_j) = m(z) = \sum_{j=0}^{2n} m_j z^j$. That means that $\prod_{|r_j|<1} (S - r_j) \hat{g}_- = 0 \Leftrightarrow \sum_{j=0}^{2n} (S^j \hat{g}_-) m_j$. This makes up a linear system of equations, where the coefficients $\{m_j\}_j$ describing the sought after polynomial are a non-trivial solution. Solving for $m(z)$ implies we know $e(z) = z^{-n} m(z)$ and we can assemble the complementary polynomials as outlined above. Although we started off with a semi-infinite $\hat{g}_-$, imposing a $m(z)$ of fixed degree also fixes the size of the system.

Furthermore, [3] points out that for a numerically stable algorithm it is important that the leading-order coefficient of $b(z)$ is chosen to be much larger than the lower-order ones to ensure the system has full numerical rank. A strength of this algorithm compared to [1] seems that the issue of small leading-order coefficient can be circumvented by a smart choice and thereby avoiding high-precision arithmetics.

3. Finding angles by optimization and maximal solution

An alternative to carry out factorization by decomposition and completion is to formulate an optimization problem over symmetric QSP (i.e., $b(z) = 0, Q \in \mathbb{R}[x]$ and Φ symmetric), namely

$$\Phi^* = \arg\min_\Phi \tfrac{1}{\tilde{n}} \textstyle\sum_{k=1}^{\tilde{n}} |g(x_k, \Phi) - f(x_k)|^2, \tag{8}$$

where $\{x_k\}$ are chosen to be the Chebychev nodes and $\tilde{n} = \lceil (n+1)/2 \rceil$ is the degree of freedom after normalization and parity constraint. As for the factorization approaches, the set of admissible factorizations and conversely the number of global minimizers is not unique. However, [2] did find one solution that stands out, which they call *maximal solution*. Namely, using a specific initial guess $\Phi_0 = (\frac{\pi}{4}, 0, \cdots, 0, \frac{\pi}{4})$ of length $n+1$, they show that in a neighbourhood of this initial point, there exists a global minimizer and the optimization landscape is strongly convex assuming that $\|f\|_\infty = O(1/n)$ for $n = \deg(f)$. In fact, this solution cannot be found by unmodified earlier approaches such as [1] and similarly in [3] as $b(z) \neq 0$ by construction in the latter.

References

[1] J. Haah, *Product Decomposition of Periodic Functions in Quantum Signal Processing*, Quantum **3** (2019), 190.

[2] J. Wang, Y. Dong, L. Lin, *On the energy landscape of symmetric quantum signal processing*, Quantum **6** (2022), 850.

[3] L. Ying, *Stable factorization for phase factors of quantum signal processing*, Quantum **6** (2022), 842.

[4] R. Chao, D. Ding, A. Gilyen, C. Huang, M. Szegedy, *Finding angles for quantum signal processing with machine precision*, arXiv:2003.02831 (2020).

Schur's Algorithm for Bounded Holomorphic Functions

Alberto Takase

Let $f : \mathbb{D}(1) \to \overline{\mathbb{D}(1)}$ be a function and assume f is holomorphic. Here $\mathbb{D}(1)$ is the unit disk $\mathbb{D}(1) := \{z \in \mathbb{C} : |z| < 1\}$. Boyd [1] recalls that, by Schur's Algorithm (1918), given f there exist polynomials A_n, Q_n such that $A_n/Q_n \to f$. Boyd then announces an improvement that if $\|f\|_\infty < 1$, i.e., $f : \mathbb{D}(1) \to \mathbb{D}(1)$, then there exist polynomials A_n, Q_n such that $A_n \to a$ for some holomorphic function a and $Q_n \to q$ for some holomorphic function q, and $f = a/q$. This result was discovered while generalizing Schur's Algorithm to cases involving functions with a finite number of poles; see the 40-page paper written in French by Chamfy (1958) which Boyd cites.

Define

$$f_0(z) := f(z) \quad \text{and} \quad \gamma_0 := f_0(0).$$

Observe $|\gamma_0| < 1$. Indeed, $\|f\|_\infty < 1$. Define

$$f_{n+1}(z) := \frac{f_n(z) - \gamma_n}{z(1 - \overline{\gamma_n} f_n(z))} \quad \text{and} \quad \gamma_n := f_n(0).$$

Observe $|\gamma_n| < 1$ provided f is not of the form $\varepsilon z^s \overline{Q}(z^{-1})/Q(z)$, where Q is a polynomial of degree s and ε is a constant with $|\varepsilon| = 1$. This nontrivial observation is claimed by Boyd in his `section 1`.

Observe

$$f = \frac{(A_0 + zQ_0^* f_1)}{(Q_0 + zA_0^* f_1)},$$

$$A_0^*(z) =: z^0\overline{A_0(z^{-1})} = \overline{A_0}, \quad Q_0^*(z) =: z^0\overline{Q_0(z^{-1})} = \overline{Q_0},$$

where A_0, Q_0 are polynomials of degree at most 0. Indeed, $A_0 = \gamma_0$ and $Q_0 = 1$. Observe

$$f = \frac{(A_n + zQ_n^* f_{n+1})}{(Q_n + zA_n^* f_{n+1})},$$

$$A_n^*(z) =: z^n\overline{A_n(z^{-1})}, \quad Q_n^*(z) =: z^n\overline{Q_n(z^{-1})},$$

where A_n, Q_n are polynomials of degree at most n. For example, $A_1 = \gamma_0 + \gamma_1 z$ and $Q_1 = 1 + \overline{\gamma_0}\gamma_1 z$. For example, $A_2 = \gamma_0 + (\gamma_1 + \gamma_0\overline{\gamma_1}\gamma_2)z + \gamma_2 z^2$ and $Q_2 = 1 + (\overline{\gamma_0}\gamma_1 + \overline{\gamma_1}\gamma_2)z + \overline{\gamma_0}\gamma_2 z^2$. Furthermore,

$$\begin{bmatrix} Q_n & A_n \\ A_n^* & Q_n^* \end{bmatrix} = \begin{bmatrix} 1 & \gamma_n z \\ \overline{\gamma_n} z & z \end{bmatrix} \begin{bmatrix} Q_{n-1} & A_{n-1} \\ A_{n-1}^* & Q_{n-1}^* \end{bmatrix}.$$

This nontrivial observation is claimed by Boyd in his `section 1`.

It follows that

$$\begin{bmatrix} Q_n & zA_n \\ A_n^* & zQ_n^* \end{bmatrix} = \begin{bmatrix} Q_n & A_n \\ A_n^* & Q_n^* \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & z \end{bmatrix} = \begin{bmatrix} 1 & \gamma_n z \\ \overline{\gamma_n} & z \end{bmatrix} \cdots \begin{bmatrix} 1 & \gamma_0 z \\ \overline{\gamma_0} & z \end{bmatrix}$$

and

$$Q_n Q_n^* - A_n A_n^* = \omega_n z^n, \quad \omega_n =: (1 - |\gamma_n|^2) \cdots (1 - |\gamma_0|^2).$$

Observe

$$|A_n(z)| < |Q_n(z)| \text{ on } |z| = 1.$$

Indeed, $0 < \omega_n < 1$ and $Q_n Q_n^* - A_n A_n^* = \omega_n z^n$. Observe Q_n has no zeros in $|z| < 1$. Furthermore, given $f = \sum u_k z^k$,

$$\frac{A_n(z)}{Q_n(z)} = u_0 + \cdots + u_n z^n + s_{n+1} z^{n+1} + \cdots$$

and

$$|u_{n+1} - s_{n+1}| \leq \omega_n.$$

This nontrivial observation is claimed in Section 1 of [1].

We now briefly summarize Section 2 and Section 3 of [1], which introduce a notation and a lemma. Let g be a measurable function and assume $\log^+ |g(e^{it})|$ is integrable on $[0, 2\pi)$. Let $|z| < 1$. Define the outer function

$$\mathcal{G}(g; z) := \exp\left(\frac{1}{2\pi}\int_{[0,2\pi]} \frac{e^{it} + z}{e^{it} - z} \log|g(e^{it})|\, dt\right),$$

$$\mathcal{G}(g) := \mathcal{G}(g; 0).$$

Here if $\log^- |g(e^{it})|$ is not integrable, then $\mathcal{G}(g, z) \equiv 0$. Boyd lists facts about the outer function, but we forgo listing them here. The lemma is the following.

Lemma. Given f as above

$$\omega(f) := \lim_{n\to\infty} \omega_n(f) = \mathcal{G}(1 - |f|^2).$$

Furthermore, $|f|_\infty < 1$ if and only if $0 < \omega(f)$.

We now summarize Section 4 of [1], which states the theorem and provides a proof.

Theorem. Let $f : \mathbb{D}(1) \to \overline{\mathbb{D}(1)}$ be a function and assume f is holomorphic. If $|f|_\infty < 1$, i.e., $f : \mathbb{D}(1) \to \mathbb{D}(1)$, then there exist polynomials A_n, Q_n such that $A_n \to a$ for some holomorphic function a and $Q_n \to q$ for some holomorphic function q and (i) q is an outer function with $q(0) = 1$ and (ii) $f(z) = a(z)/q(z)$ on $\mathbb{D}(1)$ and (iii) for almost all z with $|z| = 1, |q(z)|^2 - |a(z)|^2 = \omega(f)$ and (iv) both $A_n^* \to 0$ and $Q_n^* \to 0$. Here the convergence is "uniformly on compact subsets of $\mathbb{D}(1)$."

References

[1] D. W. Boyd, *Schur's Algorithm for Bounded Holomorphic Functions*, Bull. London Math Society **11** (1979), 145–150.

Nonlinear Fourier series for better than square summable

Mitchell Taylor

The content of this talk is based on Lecture 1 of [1].

1. Review of the (linear) Fourier transform

Before introducing the nonlinear Fourier transform, we set some conventions.

Given a sequence $F = (F_n)_{n\in\mathbb{Z}}$ of complex numbers, the *Fourier transform* of F is defined formally as:

$$\widehat{F}(\theta) = \sum_{n\in\mathbb{Z}} F_n e^{-2\pi i\theta n}.$$

As is well-known, the Fourier inversion formula

$$F_n = \int_0^1 \widehat{F}(\theta) e^{2\pi i\theta n} d\theta$$

yields a correspondence between $F \in \ell^2(\mathbb{Z})$ and $\widehat{F} \in L^2(\mathbb{T})$.[1]

[1] Here, we are identifying 1-periodic functions in θ with functions in the variable $z \in \mathbb{T}$ via $z = e^{-2\pi i\theta}$.

2. The (discrete) nonlinear Fourier transform

The discrete nonlinear Fourier transform acts on sequences $F = (F_n)_{n\in\mathbb{Z}}$, where each F_n is in the unit disc D. The definition of this transform will be given in stages. As a first step, we assume that F is a finitely supported sequence, so that there exists $N \in \mathbb{N}$ such that $F_n = 0$ whenever $|n| \geq N$.

For a complex parameter z, we consider the formal infinite recursion:

$$\begin{bmatrix} a_n & b_n \end{bmatrix} = \frac{1}{\sqrt{1-|F_n|^2}} \begin{bmatrix} a_{n-1} & b_{n-1} \end{bmatrix} \begin{bmatrix} 1 & F_n z^n \\ \overline{F}_n z^{-n} & 1 \end{bmatrix}$$

with the initialization

$$a_{-\infty} = 1, \;\; b_{-\infty} = 0. \tag{1}$$

Note that by the assumption that $(F_n)_{n\in\mathbb{Z}}$ is compactly supported, the *transfer matrix*

$$\frac{1}{\sqrt{1-|F_n|^2}} \begin{bmatrix} 1 & F_n z^n \\ \overline{F}_n z^{-n} & 1 \end{bmatrix}$$

is the identity matrix when $|n|$ is sufficiently large. For this reason, the initialization (1) can be interpreted as the condition that $a_n = 1$ and $b_n = 0$ for sufficiently negative n.

We define the *nonlinear Fourier transform* of the sequence $F = (F_n)_{n\in\mathbb{Z}}$ as the pair of functions (a_∞, b_∞) in the parameter $z \in \mathbb{T}$, which is again well-defined by the assumption that $(F_n)_{n\in\mathbb{Z}}$ is compactly supported. We will use the notation

$$\overbrace{F}(z) = (a_\infty(z), b_\infty(z))$$

to denote this function.

Remark: Note that $\overbrace{F}$ is a finite Laurent polynomial in z, so may be defined everywhere on the complex plane except at the origin. However, we will view the nonlinear Fourier transform as a function on the unit circle $\mathbb{T}$, as this will be necessary when we extend the definition of $\overbrace{F}$ to non-compactly supported F.

2.1. Interpretation as a map into a group.

Recall that the group $SU(1,1)$ consists of all complex matrices of the form

$$\begin{bmatrix} a & b \\ \overline{b} & \overline{a} \end{bmatrix}$$

which have determinant one. Note that for each $z \in \mathbb{T}$, the transfer matrix is in this group. Moreover, for $z \in \mathbb{T}$, it is easy to see that the functions (a_n, b_n) can be equivalently defined via the recursion

$$\begin{bmatrix} a_n & b_n \\ \overline{b}_n & \overline{a}_n \end{bmatrix} = \frac{1}{\sqrt{1-|F_n|^2}} \begin{bmatrix} a_{n-1} & b_{n-1} \\ \overline{b}_{n-1} & \overline{a}_{n-1} \end{bmatrix} \begin{bmatrix} 1 & F_n z^n \\ \overline{F}_n z^{-n} & 1 \end{bmatrix}$$

with

$$\begin{bmatrix} a_{-\infty} & b_{-\infty} \\ \overline{b}_{-\infty} & \overline{a}_{-\infty} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

Hence, one may easily check that each matrix

$$\begin{bmatrix} a_n & b_n \\ \overline{b}_n & \overline{a}_n \end{bmatrix}$$

is in $SU(1,1)$; in particular, $|a_n|^2 = 1 + |b_n|^2$ and we may consider the nonlinear Fourier transform as a map

$$\ell_0(\mathbb{Z}; D) \to C(\mathbb{T}; SU(1,1)),$$

where $\ell_0(\mathbb{Z}; D)$ denotes the set of all finitely supported integer sequences with values in the open unit disc D. We will abuse notation and write

$$(a, b)(c, d) = (ac + b\overline{d}, ad + b\overline{c}),$$

which is consistent with the group law.

2.2. **Properties of the nonlinear Fourier transform.** For small values of F_n, the nonlinear Fourier transform can be approximated by the linear (inverse) Fourier transform. Indeed, this can be seen by linearizing in F. By Taylor expansion, we have $(1 - |F_n|^2)^{-1/2} \approx 1$ for F_n small. Otherwise, the remaining formula for (a_∞, b_∞) is polynomial in the variables F and $\overline{F}$. Collecting only the constant and linear terms, we have

$$(a_\infty, b_\infty) = (1, \sum_{n\in\mathbb{Z}} F_n z^n).$$

Thus, to leading order, $a_\infty = 1$ and b_∞ is the usual discrete Fourier transform. The following theorem summarizes various properties of the nonlinear Fourier transform. In it, for a function c defined on an open set E in the Riemann sphere, we use the notation $c^*(z) := \overline{c(\overline{z}^{-1})}$ which is defined on $E^* = \{z : \overline{z}^{-1} \in E\}$.

Theorem: The following properties hold.

(1) If $F_n = 0$ for $n \neq m$, then

$$\overbrace{(F_n)} = (1 - |F_m|^2)^{-\frac{1}{2}}(1, F_m z^m).$$

(2) If $\overbrace{(F_n)} = (a, b)$, then for the shifted sequence with n-th entry F_{n+1}, we have

$$\overbrace{(F_{n+1})} = (a, bz^{-1}).$$

(3) If the support of F is entirely to the left of the support of G, then

$$\overbrace{(F + G)} = \overbrace{F}\overbrace{G}.$$

(4) If $|c| = 1$ then

$$\overbrace{(cF_n)} = (a, cb).$$

(5) For the reflected sequence whose n-th entry is F_{-n}, we have

$$\overbrace{(F_{-n})}(z) = (a^*(z^{-1}), b(z^{-1})).$$

(6) For the complex conjugate sequence, we have

$$\widehat{(\overline{F}_n)} = (a^*(z^{-1}), b^*(z^{-1})).$$

(7) The nonlinear Fourier transform is a bijection from $\ell_0(\mathbb{Z}; D)$ into the space of all pairs (a, b) with b an arbitrary Laurent polynomial and a the unique Laurent polynomial satisfying $aa^* = 1 + bb^*$, $a(\infty) > 0$, and a has no zeros in D^*.

Remark: Observe that statements (2)-(6) are consistent with the linearization $a \sim 1$ and $b \sim \sum F_n z^n$. Statement (7) is by far the most delicate to prove; it will be relevant later for identifying the mapping properties of the nonlinear Fourier transform on Hilbert spaces.

2.3. **The definition of the nonlinear Fourier transform, summabel sequences.** As with the classical Fourier transform, the extension of the nonlinear Fourier transform to absolutely summable sequences is relatively straightforward. To see this, we define a metric on the space $SU(1,1)$ by

$$\operatorname{dist}(G, G') = \|G - G'\|_{op}.$$

Since $SU(1,1)$ is closed in $\mathbb{C}^4$, $SU(1,1)$ is a complete metric space. We define $L^\infty(\mathbb{T}; SU(1,1))$ to be the metric space of all essentially bounded functions $G : \mathbb{T} \to SU(1,1)$ with distance

$$\operatorname{dist}(G, G') = \sup_z \operatorname{dist}(G(z), G'(z)).$$

We also make $\ell^1(\mathbb{Z}; D)$ into a complete metric space by defining

$$\operatorname{dist}(F, F') = \sum_n \|T_n - T_n'\|_{op},$$

where T_n and T_n' are the associated transfer matrices. We first observe that for every $\epsilon > 0$, the above metric is bi-Lipschitz equivalent to the usual ℓ^1 metric

$$\operatorname{dist}'(F, F') = \sum_n |F_n - F_n'|$$

on $B_\epsilon = \{F_n : \sup_n |F_n| < 1 - \epsilon\}$ and $\cup_\epsilon (B_\epsilon \cap \ell^1(\mathbb{Z}; D)) = \ell^1(\mathbb{Z}; D)$. In particular, finitely supported sequences will be dense in $\ell^1(\mathbb{Z}; D)$. With this in mind, we have the following lemma.

Lemma: With the above metrics, the NLFT on $\ell_0(\mathbb{Z}; D)$ extends uniquely to a locally Lipschitz map from $\ell^1(\mathbb{Z}; D)$ to $L^\infty(\mathbb{T}; SU(1,1))$. Moreover, the NLFT of an $\ell^1(\mathbb{Z}; D)$ sequence can be written as the convergent infinite ordered product of the transfer matrices.

The proof is an application of Trotter's formula to obtain a Lipschitz estimate on bounded sets for finite sequences together with a standard approximation argument.

Remark: By somewhat more sophisticated arguments, the NLFT can be extended to ℓ^p sequences in a rather explicit fashion when $1 \leq p < 2$. However, the extension to ℓ^2 is more delicate as certain multilinear expansions in the above definition of the NLFT will fail to converge. This will be discussed in the next lecture.

References

[1] T. Tao and C. Thiele, *Nonlinear Fourier Analysis*, Lect. Notes IAS Park City Summer School, July 2003.

Reporter: Christoph Thiele

Participants

Dr. Michel Alexis
Mathematisches Institut
Universität Bonn
Endenicher Allee 60
53115 Bonn
GERMANY

Dr. Elie Alhajjar
RAND
Arlington, VA 22202
UNITED STATES

Cade Ballew
Department of Applied Mathematics
University of Washington
201 Lewis Hall
P.O. Box 353925
Seattle, WA 98195
UNITED STATES

Lars Becker
Mathematisches Institut
Universität Bonn
Endenicher Allee 60
53115 Bonn
GERMANY

Tiklung Chan
Department of Mathematics
University of California, San Diego
9500 Gilman Drive
La Jolla, CA 92093-0112
UNITED STATES

Dr. Mateus Costa de Sousa
BCAM
Basque Center for Applied Mathematics
Alameda de Mazarredo 14
48009 Bilbao, Bizkaia
SPAIN

Dr. Jaume de Dios Pont
Forschungsinstitut für Mathematik
ETH-Zürich
ETH Zentrum
Rämistr. 101
8092 Zürich
SWITZERLAND

Rubén de la Fuente Fernández
BCAM
Basque Center for Applied Mathematics
Gran Via, 35-2
48009 Bilbao, Bizkaia
SPAIN

Dr. Joao Doriguello
Alfred Renyi Institute of Mathematics
Hungarian Academy of Sciences
P.O. Box 127
1364 Budapest
HUNGARY

Nicolas Faroß
Fachrichtung Mathematik
Universität des Saarlandes
Campus
66123 Saarbrücken
GERMANY

Max Gießler
Mathematisches Institut
Universität Bonn
Endenicher Allee 60
53115 Bonn
GERMANY

Dr. András Gilyén
HUN-REN Alfréd Rényi Institute of
Mathematics
Reáltanoda utca 13-15
1053 Budapest
HUNGARY

Prof. Dr. Martin Hairer
EPFL
Department of Mathematics
1015 Lausanne
SWITZERLAND

Dr. Minh Ha Quang
RIKEN Center for Advanced Intelligence Project
1-4-1 Nihonbashi, 15F, Chuo-ku
Tokyo 103-0027
JAPAN

Alexander Win Hsu
Department of Applied Mathematics
University of Washington
201 Lewis Hall
Box 353925
Seattle WA 98195-4350
UNITED STATES

Kaiyi Huang
Department of Mathematics
University of Wisconsin-Madison
480 Lincoln Drive
Madison, WI 53706-1388
UNITED STATES

Massimiliano Incudini
Dipartimento di Informatica
Universita di Verona
Ca'Vignal 2, Strada Le Grazie 15
37134 Verona
ITALY

Prof. Dr. Asgar Jamneshan
Department of Mathematics
Koc University
Rumeli Feneri Yolu
34450 İstanbul
TURKEY

Agoston Kaposi
Department of Programming Languages and Compilers
Faculty of Informatics
Eötvös Loránd University
Pázmány P. sny 1/C.
1117 Budapest
HUNGARY

Dr. Miriam Kosik
60687 Poznań
POLAND

Dr. Lasse Björn Kristensen
Datalogisk Institute
University of Kopenhagen
Universitetsparken 1
2100 København
DENMARK

James Larsen
Department of Mathematics
University of Michigan
East Hall, 530 Church St
Ann Arbor, MI 48109
UNITED STATES

Chong-Wei Liang
Department of Mathematics
Macquarie University
Sydney NSW 2109
AUSTRALIA

Fred Lin
Hausdorff Center for Mathematics
Universität Bonn
Villa Maria
Endenicher Allee 62
53115 Bonn
GERMANY

Prof. Dr. Lin Lin
Department of Mathematics
University of California, Berkeley
1083 Evans Hall
Berkeley CA 94270
UNITED STATES

Shao Liu
Mathematisches Institut
Universität Bonn
Endenicher Allee 60
53115 Bonn
GERMANY

Dr. József Mák
Wigner Research Centre for Physics,
Budapest
P.O. Box 49
1525 Budapest 114
HUNGARY

Gevorg Mnatsakanyan
Department of Mathematics
University of Wisconsin-Madison
480 Lincoln Drive
Madison, WI 53706-1388
UNITED STATES

Ricardo Motta
Basque Center for Applied Mathematics
Alameda de Mazarredo 14
48009 Bilbao, Bizkaia
SPAIN

Dr. Giuseppe Negro
Instituto Superior Técnico
Universidade de Lisboa
Avenida Rovisco Pais, 1
Lisboa 1049-001
PORTUGAL

Hongkang Ni
Institute for computational and
mathematical engineering
Stanford University
Stanford, CA 94305-2125
UNITED STATES

Dr. Kristina Oganesyan
Faculty of Mechanics and Mathematics
Lomonosov Moscow State University
Leninskie gory 1
119991 Moscow
RUSSIAN FEDERATION

Dr. Itamar Oliveira
School of Mathematics and Statistics
The University of Birmingham
Edgbaston
Birmingham B15 2TT
UNITED KINGDOM

Prof. Dr. Diogo Oliveira e Silva
Departamento de Matemática
Instituto Superior Técnico
Lisboa 1049-001
PORTUGAL

Joseph Peetz
Department of Physics and Astronomy
University of California
Los Angeles, CA 90095
UNITED STATES

Lorenzo Pompili
Mathematisches Institut
Universität Bonn
Platanenweg 29
53225 Bonn
GERMANY

Prof. Dr. Luz Roncal
Basque Center for Applied Mathematics
Alameda de Mazarredo 14
48009 Bilbao, Bizkaia
SPAIN

Zane Marius Rossi
Department of Physics
Massachusetts Institute of
Technology
77 Massachusetts Avenue
Cambridge MA 02139-4307
UNITED STATES

Rahul Sarkar
Institute for Computational and
Mathematical Engineering
Stanford University
Stanford, CA 94305-2125
UNITED STATES

Miquel Saucedo
Centre de Recerca Matemàtica
Campus de Bellaterra Edifici C
08193 Bellaterra, Barcelona
SPAIN

Philipp Schleich
Department of Computer Science
University of Toronto
10 Kings College Road
Toronto ON M5S 1A4
CANADA

Anne Schreuder
Department of Pure Mathematics and
Mathematical Statistics
Centre for Mathematical Sciences
Wilberforce Road
Cambridge CB3 0WB
UNITED KINGDOM

Dr. Rajula Srivastava
Mathematical Institute
University of Bonn and Max Planck
Institute for Mathematics
53111 Bonn
GERMANY

Giovanni Taffarello
Mathematisches Institut
Universität Bonn
Endenicher Allee 60
53115 Bonn
GERMANY

Dr. Alberto Takase
Department of Mathematics
Rice University
P.O. Box 1892
Houston, TX 77005-1892
UNITED STATES

Dr. Mitchell Taylor
Departement Mathematik
ETH-Zentrum
Rämistr. 101
8092 Zürich
SWITZERLAND

Prof. Dr. Christoph Thiele
Mathematisches Institut
Universität Bonn
Endenicher Allee 60
53115 Bonn
GERMANY

Dr. Gennady Uraltsev
Department of Mathematics
University of Arkansas
Fayetteville AR 72701
UNITED STATES

Jianghao Zhang
Mathematisches Institut
Universität Bonn
Endenicher Allee 60
53115 Bonn
GERMANY