Computability of the Fourier Transform and ZFC

Holger Boche and Ullrich J. Mönich

Technische Universität München
Lehrstuhl für Theoretische Informationstechnik

SampTA
July 12, 2019

Motivation

Fourier Transform:

$$\widehat{f}(\omega) = (\mathcal{F}f)(\omega) = \int_{-\infty}^{\infty} f(t)\, e^{-i\omega t} \, dt$$

- The Fourier transform is an important operation in signal processing, physics, and mathematics.
- Only for very simple functions there exists a closed form solution of the Fourier transform.
- Hence, computer algorithms / digital computers are used to compute the Fourier transform.

Motivation

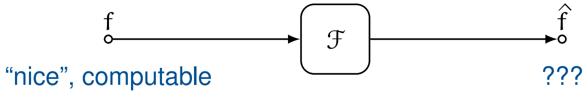
Fourier Transform:

$$\hat{f}(\omega) = (\mathcal{F}f)(\omega) = \int_{-\infty}^{\infty} f(t)\, e^{-i\omega t}\, dt$$

- The Fourier transform is an important operation in signal processing, physics, and mathematics.
- Only for very simple functions there exists a closed form solution of the Fourier transform.
- Hence, computer algorithms / digital computers are used to compute the Fourier transform.

Question:

How does the Fourier transform alter the properties of a function?



Computability

- Theory of computability is a well-established field in computer sciences.
- Computability theory is different from complexity theory.
- Complexity theory analyzes and classifies the computable problems with respect to their complexity.
- Computability theory studies the theoretically feasible.
  - No restriction on memory, computing time
  - Tool: Turing machines

General Problem

Many practical problems are continuous:

- Capacity of channels
- Fourier transform
- Maxwell's equations

A Turing machine can solve arbitrary complex discrete problems.

Question

- When can we approximate a practical analog problem by a discrete problem with a controlled approximation error?
- Only if we can control the approximation error, the solution of the discrete problem, which can be solved on a Turing machine, gives useful information about for the continuous problem.

Notation

- $L^p(\Omega)$, $1 \leqslant p < \infty$: space of all measurable, pth-power Lebesgue integrable functions on Ω
  Norm: $\|f\|_p = \left(\int_\Omega |f(t)|^p \, dt\right)^{1/p}$
- $L^\infty(\Omega)$: space of all functions for which the essential supremum norm $\|\cdot\|_\infty$ is finite

Definition (Bernstein Space)

Let $\mathcal{B}_\sigma$ be the set of all entire functions f with the property that for all $\epsilon > 0$ there exists a constant $C(\epsilon)$ with $|f(z)| \leqslant C(\epsilon) \exp\big((\sigma + \epsilon)|z|\big)$ for all $z \in \mathbb{C}$.

The Bernstein space $\mathcal{B}_\sigma^p$ consists of all functions in $\mathcal{B}_\sigma$, whose restriction to the real line is in $L^p(\mathbb{R})$, $1 \leqslant p \leqslant \infty$. The norm for $\mathcal{B}_\sigma^p$ is given by the L^p-norm on the real line.

- A function in $\mathcal{B}_\sigma^p$ is called bandlimited to σ.
- We have $\mathcal{B}_\sigma^p \subset \mathcal{B}_\sigma^r$ for all $1 \leqslant p \leqslant r \leqslant \infty$.
- $\mathcal{B}_\sigma^2$ is the space of bandlimited functions with finite energy.

Partial Recursive Functions

- Partial recursive functions, mapping from $\mathbb{N}$ to $\mathbb{N}$, are exactly those functions that can be algorithmically computed with a Turing machine.
- Partial function on $\mathbb{N}$: function $f(n)$ that may not be defined for all $n \in \mathbb{N}$.

Computable Real Numbers

Key idea: effective approximation

A real number x is said to be computable if there exists a computable sequence of rational numbers $\{r_n\}_{n\in\mathbb{N}}$ and a recursive function $\xi\colon \mathbb{N} \to \mathbb{N}$ such that

$$|x - r_{\xi(n)}| < 2^{-n}$$

for all $n \in \mathbb{N}$.

- $\mathbb{R}_c$ denotes the set of computable real numbers.
- $\mathbb{C}_c = \mathbb{R}_c + i\mathbb{R}_c$ denotes the set of computable complex numbers.
- Commonly used constants like e and π are computable.

Computable Functions

Several definitions of computable functions:

- Turing computable
- Markov computable
- Banach–Mazur computable

A functions that is computable with respect to any of the above definitions, has the property that it maps computable numbers into computable numbers.

$\rightarrow$ This property is a necessary condition for computability.

- Usual functions like sin, sinc, log, and exp are Turing computable, and finite sums of computable functions are Turing computable.

Computable Bandlimited Functions I

We call a function f elementary computable if there exists a natural number N and a sequence of computable numbers $\{\alpha_k\}_{k=-N}^{N}$ such that

$$f(t) = \sum_{k=-N}^{N} \alpha_k \frac{\sin(\pi(t-k))}{\pi(t-k)}.$$

- Every elementary computable function f is a finite sum of Turing computable functions and hence Turing computable.
- For every $t \in \mathbb{R}_c$ the number $f(t)$ is computable.
- For every elementary computable function f, the norm $\|f\|_{\mathcal{B}_\pi^p}$ is computable.

Approximation by Elementary Functions

Fact

Let $f \in \mathcal{B}_\pi^p$, $1 < p < \infty$. *For every* $\epsilon > 0$ *there exists an* $N \in \mathbb{N}$ *and numbers* $\{c_k\}_{k=-N}^N$ *such that*

$$\left\| f - \sum_{k=-N}^{N} c_k \frac{\sin(\pi(\,\cdot - k))}{\pi(\,\cdot - k)} \right\|_{\mathcal{B}_\pi^p} < \epsilon.$$

Classical approximation of bandlimited functions by elementary computable functions.

Computable Bandlimited Functions II

A function in $f \in \mathcal{B}_\pi^p$, $1 \leqslant p < \infty$, is computable in $\mathcal{B}_\pi^p$ if there exists a computable sequence of elementary computable functions $\{f_n\}_{n\in\mathbb{N}}$ and a recursive function $\xi\colon \mathbb{N} \to \mathbb{N}$ such that

$$\|f - f_{\xi(n)}\|_{\mathcal{B}_\pi^p} \leqslant 2^{-n}$$

for all $n \in \mathbb{N}$.

- We can approximate every function that is computable in $f \in \mathcal{B}_\pi^p$ by an elementary computable function, where we have an "effective" control of the approximation error.

Bit Strings

- Σ^*: set of all finite sequences of 0's and 1's (finite bit string)
- $|u|$: length of u

We can define a total order $<_{\Sigma^*}$ for the set Σ^* by putting $u <_{\Sigma^*} v$ if

1. $|u| < |v|$, or
2. $|u| = |v|$ and u lexicographically precedes v.

$$0 <_{\Sigma^*} 1 <_{\Sigma^*} 00 <_{\Sigma^*} 01 <_{\Sigma^*} 10 <_{\Sigma^*} 11 <_{\Sigma^*} 000 <_{\Sigma^*} \dots$$

Bit Strings

- Σ^*: set of all finite sequences of 0's and 1's (finite bit string)
- $|u|$: length of u

We can define a total order $<_{\Sigma^*}$ for the set Σ^* by putting $u <_{\Sigma^*} v$ if

1. $|u| < |v|$, or
2. $|u| = |v|$ and u lexicographically precedes v.

$$0 <_{\Sigma^*} 1 <_{\Sigma^*} 00 <_{\Sigma^*} 01 <_{\Sigma^*} 10 <_{\Sigma^*} 11 <_{\Sigma^*} 000 <_{\Sigma^*} \dots$$

- This ordering provides a numbering of Σ^*, and thus a bijection between $\mathbb{N}$ and Σ^*.

Any partial recursive function $\psi \colon \mathbb{N} \to \mathbb{N}$ can be interpreted as a mapping from Σ^* into Σ^*.

Prefix-Free Code

- $u \frown v$: concatenation of u and v

Definition (Prefix)

A bit string $u \in \Sigma^*$ is a prefix of a bit string $v \in \Sigma^*$ if $v = u \frown r$ for some $r \in \Sigma^*$.

Definition (Prefix-Free Code)

$A \subset \Sigma^*$ is called prefix-free code, if for arbitrary $u, v \in A$ with the property that u is a prefix of v, we have $u = v$.

Prefix-Free Code

- $u \frown v$: concatenation of u and v

Definition (Prefix)

A bit string $u \in \Sigma^*$ is a prefix of a bit string $v \in \Sigma^*$ if $v = u \frown r$ for some $r \in \Sigma^*$.

Definition (Prefix-Free Code)

$A \subset \Sigma^*$ is called prefix-free code, if for arbitrary $u, v \in A$ with the property that u is a prefix of v, we have $u = v$.

For a prefix-free code $A \subset \Sigma^*$ we have the Kraft–Chaitin inequality

$$\sum_{u \in A} \frac{1}{2^{|u|}} \leqslant 1.$$

Chaitin Function

Definition (Chaitin Function)

We call a partial recursive function $\psi\colon \Sigma^* \supset A \to \Sigma^*$ a Chaitin function if its domain $\text{dom}(\psi)$ is a prefix-free code.

- ψ: Chaitin function
- $A = \text{dom}(\psi) \subset \Sigma^*$ (prefix-fee code)
- $\phi_A : \mathbb{N} \to A$: recursive enumeration of the elements of A (created by the total order $<_{\Sigma^*}$)

We set

$$\Omega_A := \sum_{N=1}^{\infty} \frac{1}{2^{|\phi_A(N)|}}.$$

Remarks about Ω_A

From the Kraft–Chaitin inequality it follows that

$$\Omega_A = \sum_{N=1}^{\infty} \frac{1}{2^{|\phi_A(N)|}} \leqslant 1.$$

The partial sums

$$x_l = \sum_{N=1}^{l} \frac{1}{2^{|\phi_{A_*}(N)|}}$$

define a monotonically increasing and bounded sequence $\{x_l\}_{l \in \mathbb{N}}$ of dyadic rational numbers.

$\Rightarrow$ The limit $\Omega_A = \sum_{N=1}^{\infty} \frac{1}{2^{|\phi_A(N)|}} = \lim_{l \to \infty} x_l$ exists and is unique.

ZFC

- The Zermelo–Fraenkel set theory with the axiom of choice included (ZFC) is the common and accepted foundation of mathematics.
- Almost all mathematical statements can be formulated in a way that provable statements can be derived from ZFC.

We call ZFC arithmetically sound if any sentence of arithmetic which is a theorem of ZFC is true in the standard model of Peano arithmetic (PA).

Binary Expansions

A rational number $x \in (0, 1)$ is called dyadic rational if we have $x = m/2^N$ for some $m, N \in \mathbb{N}$. (We can assume that m and 2^N are coprime).

Binary Expansion: For every number $x \in (0, 1)$ that is not dyadic rational we have the unique representation

$$x = \sum_{n=1}^{\infty} a_n(x) \frac{1}{2^n},$$

where $a_n(x) \in \{0, 1\}$, $n \in \mathbb{N}$.

- We call $a_n(x)$ the n-th binary digit of x.

Solovay's Theorem

Theorem (Solovay)

There exists a Chaitin function ψ_**, such that ZFC, if arithmetically sound, can determine no single binary digit of* Ω_{A_*}*, where* $A_* = \mathrm{dom}(\psi_*)$*.*

Solovay's Theorem

Theorem (Solovay)

There exists a Chaitin function ψ_*, such that ZFC, if arithmetically sound, can determine no single binary digit of Ω_{A_*}, where $A_* = \mathrm{dom}(\psi_*)$.

- We use Ω_{A_*} to construct a "nice" function $f_* \in \mathcal{B}^1_{2\pi}$ such that

$$\hat{f}_*(0) = \Omega_{A_*} = \sum_{N=1}^{\infty} \frac{1}{2^{|\phi_{A_*}(N)|}}$$

- Hence, ZFC, if arithmetically sound, cannot determine a single binary digit of $\hat{f}_*(0)$.

Solovay's Theorem

Theorem (Solovay)

There exists a Chaitin function ψ_*, such that ZFC, if arithmetically sound, can determine no single binary digit of Ω_{A_*}, where $A_* = \text{dom}(\psi_*)$.

- We use Ω_{A_*} to construct a "nice" function $f_* \in \mathcal{B}^1_{2\pi}$ such that

$$\hat{f}_*(0) = \Omega_{A_*} = \sum_{N=1}^{\infty} \frac{1}{2^{|\phi_{A_*}(N)|}}$$

- Hence, ZFC, if arithmetically sound, cannot determine a single binary digit of $\hat{f}_*(0)$.

> ZFC, if arithmetically sound, cannot determine whether $\hat{f}_*(0) \in (0, 1/2)$ or $\hat{f}_*(0) \in (1/2, 1)$.

Fourier Transform and ZFC

Theorem

We construct a function $f_* \in \mathcal{B}^1_{2\pi}$ *such that:*

1. f_* *is computable as an element of* $\mathcal{B}^p_{2\pi}$ *for all* $1 < p < \infty, p \in \mathbb{R}_c$,
2. f_* *has a continuous Fourier transform* $\widehat{f}_*$,
3. $\widehat{f}_*(\omega) \in \mathbb{C}_c$ *for all* $\omega \in \mathbb{R}_c \setminus \{0\}$,
4. *ZFC, if arithmetically sound, cannot determine a single binary digit of* $\widehat{f}_*(0)$.

Maximum of the Fourier Transform

The function f_* has also been constructed such that

$$|\hat{f}_*(\omega)| \leqslant \hat{f}_*(0), \qquad \omega \in \mathbb{R}.$$

Corollary

ZFC, if arithmetically sound, cannot determine a single binary digit of $\|\hat{f}_*\|_\infty$.

Interesting because:
$\hat{f}_*(\omega) \in \mathbb{C}_c$ for all $\omega \in \mathbb{R}_c \setminus \{0\}$ and $\lim_{\omega \to 0} \hat{f}_*(\omega) = \hat{f}_*(0) = \|\hat{f}_*\|_\infty$.

Consequences

Theorem

There exists a natural number M_0 such that ZFC, if arithmetically sound, cannot prove the statement $|\widehat{f}_*(0) - \lambda| < 2^{-M_0}$ for any $\lambda \in \mathbb{Q} \cap (0, 1)$.

- $\widehat{f}_*(0)$ cannot be effectively approximated by rational numbers.
- The statement $|\widehat{f}_*(0) - \lambda| < 2^{-M_0}$ is true for a countably infinite subset of $\mathbb{Q} \cap (0, 1)$. But it cannot be proved for a single of these rational numbers.

Turing Computability

- For any number that is Turing computable, ZFC can determine every binary digit of the binary expansion.

Corollary

If ZFC is arithmetically sound, then $\widehat{f}_*$ is not Turing computable, because $\widehat{f}_*(0)$ is not Turing computable.

- The Fourier transform is not computable on a digital computer, because we have no way of effectively controlling the approximation error.

Conclusions

- There exists a "nice" function f_* such that its Fourier transform $\hat{f}_*$ is not Turing computable.
- ZFC (if arithmetically sound) cannot determine a single bit of $\hat{f}_*(0)$.
- Similar non-computability results can be shown for other problems:
  - bandlimited interpolation
  - discrete Fourier transform [BM19]
  - spectral factorization [BP19]

[BM19] H. Boche and U. J. Mönich, "On the Fourier representation of computable continuous signals," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '19)*, May 2019, pp. 5013–5017

[BP19] H. Boche and V. Pohl, "On the algorithmic solvability of the spectral factorization and the calculation of the Wiener filter on Turing machines," in *Proceedings of the 2019 IEEE International Symposium on Information Theory*, 2019, accepted

Thank you!