Turing Computability of the Fourier Transform of Bandlimited Functions

Holger Boche and Ullrich J. Mönich

Technische Universität München
Lehrstuhl für Theoretische Informationstechnik

July 8, 2019 | 14:50–15:10 | Odéon, Level 3 (Session: MO3.R4)
2019 IEEE International Symposium on Information Theory

Motivation

Fourier Transform:

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

- The Fourier transform is an essential operation in information sciences.
- Only for very simple functions there exist closed form solutions of the Fourier transform.
- Hence, computer algorithms / digital computers are used to compute the Fourier transform.

We study the computability of the Fourier transform.

$$\text{increase in computational power} \xrightarrow{?} \text{increase in accuracy}$$

Turing Machine

Turing Machine:
Abstract device that manipulates symbols on a strip of tape according to certain rules.

- Turing machines are an idealized computing model.
- No limitations on computing time or memory, no computation errors.
- Although the concept is very simple, Turing machines are capable of simulating any given algorithm.

Turing Machine

Turing machines are suited to study the limitations in performance of a digital computer:

Anything that is not Turing computable cannot be computed on a real digital computer, regardless how powerful it may be.

- Alan Turing introduced the concept of a computable real number in 1936, and demonstrated some principal limitations of computability [Turing36/37].
- In 1949 a computable monotonically increasing sequence which converges to a real non-computable number was constructed [Specker49].

[Turing36] A. M. Turing, “On computable numbers, with an application to the Entscheidungsproblem,” *Proceedings of the London Mathematical Society*, vol. s2-42, no. 1, pp. 230–265, Nov. 1936

[Turing37] A. M. Turing, “On computable numbers, with an application to the Entscheidungsproblem. A correction,” *Proceedings of the London Mathematical Society*, vol. s2-43, no. 1, pp. 544–546, Jan. 1937

[Specker49] E. Specker, “Nicht konstruktiv beweisbare Sätze der Analysis,” *The Journal of Symbolic Logic*, vol. 14, no. 3, pp. 145–158, Sep. 1949

Notation

- $\ell^p(\mathbb{Z})$, $1 \leqslant p < \infty$: spaces of p-th power summable sequences $x = \{x(k)\}_{k\in\mathbb{Z}}$
  Norm: $\|x\|_{\ell^p} = (\sum_{k=-\infty}^{\infty} |x(k)|^p)^{1/p}$
- C: space of all continuous functions on $\mathbb{R}$
  Norm: $\|f\|_C = \max_{t\in\mathbb{R}} |f(t)|$
- $L^p(\Omega)$, $1 \leqslant p < \infty$: 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

Bandlimited Functions

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 $1 \leqslant p \leqslant r \leqslant \infty$.
- $\mathcal{B}_{\sigma,0}^\infty$ denotes the space of all functions in $\mathcal{B}_\sigma^\infty$ that vanish at infinity.
- $\mathcal{B}_\sigma^2$ is the space of bandlimited functions with finite energy.

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.

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.

Computable Sequences of Rationals

A sequence of rational numbers $\{r_n\}_{n\in\mathbb{N}}$ is called computable sequence if there exist recursive functions a, b, s from $\mathbb{N}$ to $\mathbb{N}$ such that $b(n) \neq 0$ for all $n \in \mathbb{N}$ and

$$r_n = (-1)^{s(n)} \frac{a(n)}{b(n)}, \qquad n \in \mathbb{N}.$$

- A recursive function is a function, mapping natural numbers into natural numbers, that is built of simple computable functions and recursions. Recursive functions are computable by a Turing machine.

Computable Real Numbers

First example of an 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.
- $\mathbb{R}_c$ is a field, i.e., finite sums, differences, products, and quotients of computable numbers are computable.
- Commonly used constants like e and π are computable.

Computability in ℓ^p

A sequence $\{x(k)\}_{k \in \mathbb{Z}}$ in ℓ^p, $p \in [1, \infty) \cap \mathbb{R}_c$ is called computable in ℓ^p if every number $x(k)$, $k \in \mathbb{Z}$, is computable and there exist a computable sequence $\{x_N\}_{N \in \mathbb{N}} \subset \ell^p$, where each x_N has only finitely many non-zero elements, all of which are computable as real numbers, and a recursive function $\xi \colon \mathbb{N} \to \mathbb{N}$, such that

$$\|x - x_{\xi(n)}\|_{\ell^p} \leqslant 2^{-n}$$

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

- $\mathcal{C}\ell^p$ denotes the set of all sequences that are computable in ℓ^p.

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.
- The sum of finitely many elementary computable functions is computable.
- The product of an elementary computable function with a computable number.
- For every elementary computable function f, the norm $\|f\|_{\mathcal{B}_\pi^p}$ is computable.

Computable Bandlimited Functions II

A function in $f \in \mathcal{B}^p_\pi$, $1 \leqslant p < \infty$, is computable in $\mathcal{B}^p_\pi$ 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

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

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

- $\mathcal{CB}^p_\pi$ denotes the set of all functions that are computable in $\mathcal{B}^p_\pi$.
- $\mathcal{CB}^p_\pi$ has a linear structure.
- We can approximate every function $f \in \mathcal{CB}^p_\pi$ by an elementary computable function, where we have an “effective” control of the approximation error.
- Similar definition for $\mathcal{CB}^\infty_{\pi,0}$.

Computable Bandlimited Functions III

The norm $\|f\|_{\mathcal{B}^\infty_{\pi,0}}$, i.e., the maximum of f, is computable for all $f \in \mathcal{C}\mathcal{B}^\infty_{\pi,0}$, because:

- we have

$$\left| \|f\|_{\mathcal{B}^\infty_{\pi,0}} - \|f_n\|_{\mathcal{B}^\infty_{\pi,0}} \right| \leqslant \|f - f_n\|_{\mathcal{B}^\infty_{\pi,0}},$$

  and

- for every elementary computable function f_n, the norm $\|f_n\|_{\mathcal{B}^\infty_{\pi,0}}$ is computable.

Approximation by Elementary Functions

Fact

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

$$\left\| f - \sum_{k=-N}^{N} c_k \frac{\sin(\pi(\cdot - k))}{\pi(\cdot - k)} \right\|_{\mathcal{B}_{\pi,0}^{\infty}} < \epsilon.$$

Classical approximation of bandlimited functions by elementary computable functions.
$\rightarrow$ The above definition of a computable function in $\mathcal{B}_{\pi,0}^{\infty}$ is sensible.

Computability of the Fourier Transform

We construct function f_* such that:

- f_* is very nice / well-behaved
  - $f_* \in \mathcal{B}^1_{2\pi}$: bandlimited and continuous
  - f_* is computable as an element of $\mathcal{B}^p_{2\pi}$ for all $1 < p < \infty$, $p \in \mathbb{R}_c$
- The Fourier transform $\widehat{f}_*$ is a continuous function
- The Fourier transform $\widehat{f}_*$ is not computable
  - $\widehat{f}_*(0) \notin \mathbb{R}_c$

Theorem

We construct a function $f_* \in \mathcal{B}^1_{2\pi}$ such that f_* is computable as an element of $\mathcal{B}^p_{2\pi}$ for all $1 < p \leqslant \infty$, $p \in \mathbb{R}_c$, and f_* has a continuous Fourier transform $\widehat{f}_*$ that is not computable in C, because $\widehat{f}_*(0) \notin \mathbb{R}_c$. Further, the function f_* is constructed such that $\widehat{f}_*(\omega) \in \mathbb{C}_c$ for all $\omega \in \mathbb{R}_c \setminus \{0\}$.

- $f_*(t)$ is computable for all computable t, but $\widehat{f}_*(0)$ is not computable.
- The $L^p(\mathbb{R})$-norms of f_* are computable for all computable $1 < p \leqslant \infty$, and, in particular, the energy, i.e., the $L^2(\mathbb{R})$-norm is computable.

Consequences

- The Fourier transform is not computable on a digital computer, because we have no way of effectively controlling the approximation error.
- This result has consequences for algorithms that use the Fourier transform of bandlimited function, e.g., the computation of the convolution via a multiplication in the Fourier domain.

Computability of the Maximum

- $\widehat{f}_*(0)$ is the maximum of the function $|\widehat{f}_*|$.
- Since $\widehat{f}_*(0)$ is not computable, it follows that

$$\|\widehat{f}_*\|_C = \max_{\omega\in\mathbb{R}}|\widehat{f}_*(\omega)| \text{ is not computable.}$$

Interesting because:

$\widehat{f}_*(\omega) \in \mathbb{C}_c$ for all $\omega \in \mathbb{R}_c \setminus \{0\}$ and $\lim_{\omega\to 0} \widehat{f}_*(\omega) = \widehat{f}_*(0)$

Analog Implementations / Analog Computers

- Fourier optics is a well-established discipline in physics and optics.
  (Older than Turing's theory of computability and digital computers.)
- The 2f architecture in Fourier optics is an optical setup, in which a lens is used to perform the Fourier transform.
- Can be seen as an analog machine for computing the Fourier transform of a bandlimited function.
- In theory: the Fourier transform computed by such a system is perfect.
  In practice: imperfections, such as misalignment and noise, that limit the precision with which the Fourier transform can be computed.

The idealized analog machine is capable of computing the Fourier transform while the Fourier transform is not Turing computable.
$\rightarrow$ An idealized analog machine can be more powerful than an idealized digital machine.

- So far it is unclear whether this theoretical superiority can be translated into practice.

Conclusions

- We studied the Fourier transform with respect to computability.
- We constructed a “nice” bandlimited function whose Fourier transform is not Turing computable.
- The Fourier transform is theoretically computable on an analog computer.
- Similar 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!