

On the Arithmetic Complexity of the Bandwidth of Bandlimited Signals

Holger Boche¹, Yannik N. Böck², Ullrich J. Mönich²

Abstract

The bandwidth of a signal is an important physical property that is of relevance in many signal- and information-theoretic applications. In this paper we study questions related to the computability of the bandwidth of computable bandlimited signals. To this end we employ the concept of Turing computability, which exactly describes what is theoretically feasible and can be computed on a digital computer. Recently, it has been shown that there exist computable bandlimited signals with finite energy, the actual bandwidth of which is not a computable number, and hence cannot be computed on a digital computer. In this work, we consider the most general class of band-limited signals, together with different computable representations thereof. Among other things, our analysis includes a characterization of the arithmetic complexity of the bandwidth of such signals and yields a negative answer to the question of whether it is at least possible to compute non-trivial upper or lower bounds for the bandwidth of a bandlimited signal. Furthermore, we relate the problem of bandwidth computation to the theory of oracle machines. In particular, we consider halting and totality oracles, which belong to the most frequently investigated oracle machines in the theory of computation.

Keywords: Bandlimited Signal, Shannon Sampling, Bandwidth, Turing Machine.

¹Holger Boche is with the Technische Universität München, Lehrstuhl für Theoretische Informationstechnik, 80290 Munich, Germany, and the Munich Center for Quantum Science and Technology (MCQST), Schellingstr. 4, 80799 Munich, Germany. e-mail: boche@tum.de.

²Yannik N. Böck is with the Technische Universität München, Lehrstuhl für Theoretische Informationstechnik, 80290 Munich, Germany. e-mail: yannik.boeck@tum.de.

³Ullrich J. Mönich is with the Technische Universität München, Lehrstuhl für Theoretische Informationstechnik, 80290 Munich, Germany. e-mail: moenich@tum.de.

1. Introduction

The applications of bandlimited signals are a prominent field of research within the community of information theory. While most real-world physical systems are analog and continuous in time, the actual processing of information is often done on digital devices that operate in discrete-time computational cycles. Hence, the conversion of signals from the analog to the digital domain and vice versa is indispensable for modern technology [1, 2]. The link between both domains is established by various sampling theorems [3, 4], including signal recovery in the presence of noise [5], estimates for the error arising from finite-length sampling-series approximations [6], and refined theories for the treatment of multi-band signals [7, 8]. All of the mentioned applications include the bandwidth of the involved signals as an essential parameter. The processing of discrete-time representations of bandlimited signals has been studied as well, see e.g. [9, 10]. There, the authors consider the replication of time-continuous LTI systems in the discrete-time domain. Again, the bandwidth of the involved signals appears as a crucial quantity. Furthermore, bandlimited signals play a significant role in wireless communication systems, where the spectrum of the transmit signal has to be controlled in order to not interfere with other systems [11, 12].

For a bandlimited signal f , we refer to the smallest number σ , such that f is bandlimited with bandwidth σ , as the actual bandwidth of $B(f)$. According to Shannon's sampling theorem, a bandlimited signal f with finite energy is uniquely determined by the sequence of samples $(f(k/r))_{k \in \mathbb{Z}}$ if $r \geq r_{\min} = B(f)/\pi$ holds true, in which case it may be reconstructed by means of the Shannon sampling series. Hence, the actual bandwidth of a bandlimited signal is a relevant quantity.

In this paper we will study questions related to the computability of the actual bandwidth $B(f)$ of computable bandlimited signals. Our analysis is based on the theory of Turing computability, which characterizes the fundamental limits of digital computation. There exists a variety of problems that have been

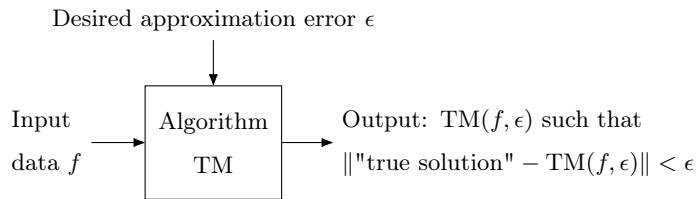


Figure 1: We adopt the computability model considered in [16]: The algorithm TM gets two inputs: the data f and the desired approximation error ϵ . The computed output $\text{TM}(f, \epsilon)$ is guaranteed to be ϵ -close to the true solution.

shown to be uncomputable on a digital computer, e.g., the computation of the Fourier transform for certain signals [13, 14] or the spectral factorization [15]. That is, all of these problems lack a way to control the approximation error involved in the computation.

Today, simulations in science and engineering rarely treat the involved approximation error explicitly, which, for example, can be observed in the absence of error bars in plots. In those cases, the computer computes some rational number, which serves as an approximation of the solution, without any quality guarantees. A framework which establishes such quality guarantees, i.e., an algorithmic control of the approximation error, is provided by the concept of computability. There, the algorithm obtains the desired approximation error ϵ , which could be the maximum tolerable error, along with the actual data f as an input, and returns a solution that satisfies the error specification. This algorithmic control of the error is illustrated in Fig. 1.

In [16], it has been shown that there exist computable bandlimited signals f for which the actual bandwidth $B(f)$ is not a computable number. This means that there cannot exist any algorithm for the computation of $B(f)$ with an effective control of the approximation error. To the best of our knowledge, [16] is the first work on computable signals that satisfy essential properties of high practical relevance, like finite energy, continuity, etc., and at the same time defy computability with respect to a fundamental signal parameter.

In this work we aim to further develop the theory of computability of ban-

dlimited signals. Several open questions and conjectures were posed in [16] with regard to the class of computable entire functions of exponential type \mathcal{CE}_π , which corresponds to the most general class of bandlimited signals:

1. For specific classes of computable bandlimited signals, the bandwidth $B(f)$ is the limit value of a monotonically non-decreasing computable sequence of rational numbers. Does this hold true for the class \mathcal{CE}_π as well? It was conjectured that the answer is no.
2. For specific classes of computable bandlimited signals, the subset of signals that satisfy $B(f) > \lambda$ for a given λ is semi-decidable. Does this hold true for the class \mathcal{CE}_π as well? Again, it was conjectured that the answer is no.
3. Even if it not possible to compute $B(f)$ for problematic signals in \mathcal{CE}_π , it might still be possible to compute meaningful bounds for $B(f)$. Hence, the question is: Do there exist Turing machines $\underline{\text{TM}}_{\text{BW}}$ and $\overline{\text{TM}}_{\text{BW}}$ such that for all signals f in \mathcal{CE}_π that satisfy $B(f) \leq \pi$, we have

$$\underline{\text{TM}}_{\text{BW}}(f) \leq B(f) \leq \overline{\text{TM}}_{\text{BW}}(f)?$$

It was conjectured that the only Turing machines that satisfy this requirement yield trivial values, i.e., $\underline{\text{TM}}_{\text{BW}}(f) = 0$ and $\overline{\text{TM}}_{\text{BW}}(f) = \pi$, for all signals in \mathcal{CE}_π .

In this work, we provide a comprehensive study of the computability properties of the bandwidth $B(f)$ of signals f in the class \mathcal{CE}_π , proving all of the above conjectures correct. As indicated above, the class \mathcal{CE}_π respects the common general definition of bandlimited signals, which is a broad extension of the class of admissible signals compared to [16]. In particular, we present a sharp characterization of the “degree” of non-computability of the number $B(f)$ in the following sense. On the one hand, the arithmetic complexity of the number $B(f)$ can never exceed the class Π_2 (which will subsequently be introduced in a formal manner). On the other hand, for every number $x \in \Pi_2 \cap [0, \pi]$, there exists a signal $f \in \mathcal{CE}_\pi$ with $B(f)$ equal to x . This characterization is analogous to the notion of achievability and converse in information theory.

Our analysis leads to interesting insights about the general limits of computability. In recent years, attempts have been made to push the boundaries of computing through various approaches, including analog, neuromorphic and quantum hardware. At least in theory, the computational capabilities of these technologies go beyond those of digital computers. For the question of whether any of them may also yield practical advantages at some point in the future, it is essential to precisely understand the phenomena underlying the limitations of digital computing, especially in the context of mathematical models that describe practically relevant problems from engineering and science.

Furthermore, our analysis uncovers interesting property of bandlimited signals, concerning the relation between the arithmetic complexity of $B(f)$ and the structural properties of f in the time-domain. To the authors' knowledge, no distinguished physical quantity other than the bandwidth of bandlimited signals has yet been identified to show a comparable behavior. The class \mathcal{CE}_π includes the subclass \mathcal{CB}_π^∞ of bandlimited signals with finite L^∞ -norm, which, in contrast to \mathcal{CE}_π , exhibits a Banach space structure. In particular, this structure restricts the decay of the corresponding signals on the time-axis. While all signals in \mathcal{CE}_π are computationally well-behaved in the time domain, we observe a jump in the arithmetic complexity of the bandwidth when extending the class of feasible signals from \mathcal{CB}_π^∞ to \mathcal{CE}_π . Hence, the computability properties of the bandwidth are coupled directly with the presence or absence of a Banach space structure in the time domain. Although not previously observed, the authors believe that such characteristics may occur for a variety of mathematical models used in the applied sciences.

The remainder of the paper is structured as follows. Sections 2 to 6 are dedicated to preliminaries. There, we introduce the basic concepts of bandlimited signals, Turing machines and A -computable functions, as well as the arithmetical hierarchy of real numbers, which provides the theoretical framework for classifying different degrees of uncomputability. Our main results are presented in Sections 7 to 11. We consider different computable representations of bandlimited signals and derive the answers to the above-mentioned con-

tures. Furthermore, we characterize the problem of computing the bandwidth of bandlimited signals by means of the arithmetical hierarchy of real numbers. Last but not least, we relate the problem of computing the bandwidth of bandlimited signals to a class of oracle computation machines. The paper closes in Section 12, with a discussion on our results and their implications.

2. Bandlimited Signals

Our analysis of bandlimited signals is based on commonly used definitions, several of which have already been employed in [16]. For the sake of self-containedness, we introduce all of the relevant definitions in the following.

By $\mathbb{N}_+ := \{1, 2, 3, \dots\}$, we denote the set of (positive) natural numbers. By $\mathbb{N} := \mathbb{N}_+ \cup \{0\}$ the set of natural numbers including zero. For $\Omega \subseteq \mathbb{R}$, let $L^p(\Omega)$, $1 \leq p < \infty$, be the space of all measurable, p -th power Lebesgue integrable functions on Ω , with the usual norm $\|\cdot\|_p$. A function f is said to be entire if it is defined and holomorphic on all of \mathbb{C} .

Definition 1. *An entire function f is called bandlimited if there exists $\sigma \geq 0$ such that for all $\epsilon > 0$ there exists a constant $C(\epsilon)$ with*

$$|f(z)| \leq C(\epsilon) e^{(\sigma+\epsilon)|z|} \tag{1}$$

for all $z \in \mathbb{C}$ [18, 4]. By \mathcal{E}_σ we denote the set of all entire functions that are bandlimited with bandwidth σ .

In particular, we will consider signals that are bandlimited with bandwidth π .

According to the definition above, $f \in \mathcal{E}_{\sigma_1}$ implies $f \in \mathcal{E}_{\sigma_2}$ whenever σ_1 satisfies $\sigma_1 \leq \sigma_2$. That is, a signal that is bandlimited with bandwidth σ_1 is also bandlimited with any bandwidth σ_2 larger than σ_1 . For a given bandlimited signal f , we denote by

$$B(f) = \min\{\sigma \geq 0: f \in \mathcal{E}_\sigma\} \tag{2}$$

the *actual* bandwidth of the signal.

Remark 1. For an entire function f that satisfies (1) for some $\sigma \in \mathbb{R}$, the minimum in (2) does exist. For details, see [16, Appendix B, p. 15].

In the following, we introduce further signal spaces with practically relevant properties. If we additionally restrict the set \mathcal{E}_π to signals with finite L^1 -norm in the time domain, i.e.,

$$\int_{-\infty}^{\infty} |f(t)| dt < \infty,$$

we obtain the Bernstein space \mathcal{B}_π^1 . On the other hand, if we restrict the set \mathcal{E}_π to signals with well-defined Fourier transform in L^2 , we obtain the Bernstein space \mathcal{B}_π^2 . According to Plancherel's theorem, these signals also have a finite L^2 -norm in the time domain. The Bernstein space \mathcal{B}_π^2 is the frequently used space of bandlimited signals with finite energy. According to the Paley–Wiener theorem [4, Theorem 7.2, p. 68], the support of the Fourier transform \hat{f} of a signal $f \in \mathcal{B}_\pi^2$ is contained in $[-\pi, \pi]$, and we have

$$f(t) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \hat{f}(\omega) e^{i\omega t} d\omega.$$

Hence, for the space \mathcal{B}_π^2 we have a further, different characterization of the actual bandwidth. For $f \in \mathcal{B}_\pi^2$, $B(f)$ is the smallest number $\sigma > 0$ such that

$$f(t) = \frac{1}{2\pi} \int_{-\sigma}^{\sigma} \hat{f}(\omega) e^{i\omega t} d\omega$$

holds true for all $t \in \mathbb{R}$. According to Plancherel's identity, this is also the smallest $\sigma > 0$ such that

$$\int_{-\infty}^{\infty} |f(t)|^2 dt = \frac{1}{2\pi} \int_{-\sigma}^{\sigma} |\hat{f}(\omega)|^2 d\omega.$$

is satisfied. The actual bandwidth $B(f)$ of a bandlimited signal f is a distinguished quantity, because it determines the minimum sampling rate that is required so that the samples uniquely determine f .

The general definition of the Bernstein spaces is as follows.

Definition 2. The Bernstein space \mathcal{B}_π^p , $1 \leq p \leq \infty$, consists of all functions in \mathcal{E}_π , whose restriction to the real line is in $L^p(\mathbb{R})$ [4, p. 49]. The norm for \mathcal{B}_π^p is given by the L^p -norm on the real line.

Remark 2. We have $\mathcal{B}_\pi^r \subsetneq \mathcal{B}_\pi^s \subsetneq \mathcal{E}_\pi$ for all $1 \leq r < s \leq \infty$.

We will discuss properties of signals $f \in \mathcal{E}_\pi$ next. Since every signal $f \in \mathcal{E}_\pi$ is entire, it can be represented as a power series that converges uniformly on all compact subsets of \mathbb{C} . Denote the n -th derivative of f by $f^{(n)}$. Then, defining $a_n := f^{(n)}(0)$ for all $n \in \mathbb{N}$, we have

$$f(z) = \sum_{n=0}^{\infty} \frac{a_n}{n!} z^n = \sum_{n=0}^{\infty} \frac{f^{(n)}(0)}{n!} z^n, \quad z \in \mathbb{C}. \quad (3)$$

The actual bandwidth $B(f)$ can be determined directly from the family of coefficients $(a_n)_{n \in \mathbb{N}}$ according to

$$B(f) = \limsup_{n \rightarrow \infty} \sqrt[n]{|a_n|}. \quad (4)$$

For details, see [19, pp. 356] or [18, Theorem 3, p. 6].

3. Preliminaries on Turing Machines and Recursive Functions

The theory of Turing machines, recursive functions and computable analysis are well-established fields in theoretical computer science. Nevertheless, in order to establish a self-contained work, we introduce all definitions and results that will be required subsequently, even if they have already been given in [16]. A comprehensive treatment of the topic may be found in [20, 21, 22, 23, 24, 25].

Turing machines, as introduced by Turing in [26, 27], are a mathematical model of what we intuitively understand as computation machines. In this sense, they yield an abstract idealization of today's real-world computers. Even though the model is relatively simple in structure, any algorithm that can be executed by a real-world computer can be simulated by a Turing machine. In contrast to real-world computers, however, Turing machines are not subject to any restrictions regarding energy consumption, computation time or memory size. All computation steps on a Turing machine are furthermore assumed to be executed with zero chance of error. Thus, computability in the sense of Turing is the exact characterization of what can be achieved by digital hardware, e.g., central processing units (CPUs), digital signal processors (DSPs), or field

programmable gate arrays (FPGAs), if practical limitations, such as energy constraints, computing errors, and hardware restrictions, are disregarded.

In formal terms, a Turing machine consists of a formal language over a finite alphabet, together with a list of transformation rules for the associated words. The transformation rules can be seen as an “algorithm”, where the words represent the “data” being processed. Since formal languages exhibit a number of intuitive encodings into the set of natural numbers (cf. Remark 4), each Turing machine may be characterized by some element of the set

$$\mathcal{N} := \bigcup_{n=1}^{\infty} \{g : \mathbb{N}^n \leftrightarrow \mathbb{N}\}, \quad (5)$$

where we use the symbol " \leftrightarrow " to denote a *partial mapping*.

Recursive functions, more specifically referred to as *μ -recursive functions*, characterize the notion of computability by means of different approach and were, amongst others, considered by Kleene in [28]. According to (5), the set \mathcal{N} contains all possible functions $g : \mathbb{N}^n \leftrightarrow \mathbb{N}$ for all $n \in \mathbb{N}_+$, and is thus uncountably infinite in cardinality. The set of those functions $g \in \mathcal{N}$ that correspond to our intuitive understanding of computability in the sense that they can be fully described by a finite sequence of fundamental arithmetic-logic operations, must necessarily be of countable cardinality, and thus be a proper subset of the set \mathcal{N} . Contrary to Turing machines, the notion of recursive functions tries to characterize this subset directly by defining a set of fundamental computable operations on the natural numbers, rather than starting from formal languages. Yet, Turing machines and recursive functions turned out to be equivalent in the following sense: the class of functions characterized by the concept of Turing machines coincides with the set of recursive functions [29]. Hence, a function $g \in \mathcal{N}$ can be computed on some Turing machine if and only if it is a recursive function.

In the following, we will look further into the properties of recursive functions. For $A \subseteq \mathbb{N}$, denote by $\mathcal{C}(A) \subsetneq \mathcal{N}$ the set which consists of the *indicator function* $\mathbb{1}_A$ of A , the *successor function*, and all *constant* and *identity functions* on tuples of natural numbers [24, Definition 2.1, p. 8]. By $\mathcal{C}^*(A)$, denote

the closure of $\mathcal{C}(A)$ with respect to *composition*, *primitive recursion* and *unbounded search* [24, Definition 2.1, p. 8, Definition 2.2, p. 10]. Then, the set $\mathcal{C}^*(A)$ is referred to as the set of A -computable functions. In particular, the set $\mathcal{C}^*(\emptyset)$ is the set of recursive functions. For brevity, we write \mathcal{C}^* instead of $\mathcal{C}^*(\emptyset)$. Furthermore, for $n \in \mathbb{N}_+$, we denote by $\mathcal{C}_n^*(A)$ and \mathcal{C}_n^* the set of A -computable functions in n -variables and the set of recursive functions in n -variables, respectively. That is, we have

$$\begin{aligned}\mathcal{C}_n^*(A) &:= \mathcal{C}^*(A) \cap \{g : \mathbb{N}^n \hookrightarrow \mathbb{N}\}, \\ \mathcal{C}^*(A) &= \bigcup_{n=1}^{\infty} \mathcal{C}_n^*(A)\end{aligned}$$

for all $n \in \mathbb{N}_+$ and all $A \subseteq \mathbb{N}$.

Definition 3. *A set $A \subseteq \mathbb{N}$ is said to be recursively enumerable if there exists a recursive function $g : \mathbb{N} \hookrightarrow \mathbb{N}$ with domain $D(g)$ equal to A .*

In the context of Turing machines, the domain $D(g) \subseteq \mathbb{N}$ of a function $g \in \mathcal{C}_1^*$ has a dedicated interpretation. Consider a Turing machine TM_g that computes the function g . Then, given an input $m \in \mathbb{N}$, the Turing machine TM_g reaches its halting state after a finite number of computational steps if and only if $m \in D(g)$ is satisfied. In contrast, if $m \in \mathbb{N} \setminus D(g)$, the Turing machine TM_g runs forever.

Definition 4. *A set $G \subseteq \mathbb{N}$ is said to be recursive if the corresponding indicator function $\mathbb{1}_G : \mathbb{N} \rightarrow \{0, 1\}$ is a recursive function.*

Remark 3. *A set $G \subseteq \mathbb{N}$ is recursive if and only if both G and $G^c := \mathbb{N} \setminus G$ are recursively enumerable sets. Furthermore, for a set $G \subseteq \mathbb{N}$, we have $\mathcal{C}^*(G) = \mathcal{C}^*$ if and only if G is a recursive set.*

The set of A -computable functions in one variable, $\mathcal{C}_1^*(A)$, which will be of special significance in the following, is *recursively enumerable* itself. In this context, recursive enumerability refers to the existence of a *universal* A -computable function $\Phi^A \in \mathcal{C}_2^*(A)$ such that for all A -computable functions $g \in \mathcal{C}_1^*(A)$, there

exists a number $n \in \mathbb{N}$ such that

$$\begin{aligned} \forall m \in D(g) : \Phi^A(n, m) &= g(m), \\ \forall m \notin D(g) : (n, m) &\notin D(\Phi^A) \end{aligned} \tag{6}$$

hold true [25, Theorem 1.5.3, p. 11]. In short, we say that g satisfies $\Phi^A(n, m) = g(m)$ for all $m \in \mathbb{N}$, implicitly (and with some abuse of notation) including the case of $\Phi^A(n, m)$ and $g(m)$ being undefined for some $m \in \mathbb{N}$. For all $n, m \in \mathbb{N}$, define $\varphi_n^A(m) := \Phi^A(n, m)$. Then, the family $(\varphi_n^A)_{n \in \mathbb{N}}$ is a *recursive enumeration* of the set of all A -computable functions in one variable. A Turing machine $\text{TM}_{\Phi^\emptyset}$ that computes the function Φ^\emptyset is referred to as a *universal Turing machine*.

The universal function Φ^A is not unique, and hence, neither is the recursive enumeration $(\varphi_n^A)_{n \in \mathbb{N}}$. Thus, we consider an arbitrary but fixed recursive enumeration $(\varphi_n^A)_{n \in \mathbb{N}}$ for the rest of this work. For the sake of simplicity, we write $(\varphi_n)_{n \in \mathbb{N}}$ instead of $(\varphi_n^\emptyset)_{n \in \mathbb{N}}$ in the special case of $A = \emptyset$. Within the scope of this work, we will consider this case most of the time.

The case of $A \subsetneq \mathbb{N}$ being some non-recursive set leads to the idea of oracle-computations, which we will investigate in Section 11. From the equivalence of Turing machines and recursive functions, we deduce the existence of a recursive *runtime function*, which will be essential in the context of oracle computations. Intuitively speaking, given a universal Turing machine $\text{TM}_{\Phi^\emptyset}$, we count the number of steps of calculation (that is, the number of successive applications of the specified transformation rules) that are required for $\text{TM}_{\Phi^\emptyset}$, given an input $(n, m) \in \mathbb{N}^2$, to reach its halting state. If $m \in D(\varphi_n)^c$, the counting continues for an infinite amount of time. Expressed in a formal way, there exists a (total) recursive function $\Psi : \mathbb{N}^3 \rightarrow \{0, 1\}$ such that the following holds true:

- For all $n, m, k \in \mathbb{N}$ that satisfy $\Psi(n, m, k) = 1$ we have $\Psi(n, m, k + 1) = 1$.
- For all $n, m \in \mathbb{N}$ that satisfy $m \in D(\varphi_n)$, there exists $k \in \mathbb{N}$ such that we have $\Psi(n, m, k) = 1$.
- For all $n, m \in \mathbb{N}$ that satisfy $m \notin D(\varphi_n)$, we have $\Psi(n, m, k) = 0$ for all $k \in \mathbb{N}$.

Even for a fixed enumeration $(\varphi_n)_{n \in \mathbb{N}}$ of \mathcal{C}_1^* , the runtime function Ψ is not unique. Hence, we again consider an arbitrary but fixed runtime function Ψ for the remainder of this work.

Definition 5. *For a universal recursive function Φ with runtime function Ψ , we denote by*

$$A_T^\Phi := \{n \in \mathbb{N} : \forall m \in \mathbb{N} : \exists k \in \mathbb{N} : \Psi(n, m, k) = 1\}$$

the totality set of Φ .

From the properties of Ψ , it follows that the totality set contains exactly the indices of all total functions in $(\varphi_n)_{n \in \mathbb{N}}$. Hence, we have

$$A_T^\Phi := \{n \in \mathbb{N} : D(\varphi_n) = \mathbb{N}\},$$

which yields a direct relation to the family $(D(\varphi_n))_{n \in \mathbb{N}}$, which enumerates the set of recursively enumerable sets according to the universal function Φ . In this context, we will reconsider the runtime function Ψ in Lemma 3 and Lemma 4 at the end of Section 6. Ultimately, the set $\mathcal{C}^*(A_T^\Phi)$ of A_T^Φ -computable functions and the set $\mathcal{C}^*(D(g))$ of $D(g)$ -computable functions for $g \in \mathcal{C}^*$ will play a fundamental role in Section 11 in the scope of oracle computations.

4. Encoding Abstract Structures into the Natural Numbers

Throughout this work, we consider computations on different abstract structures, like, for example, real numbers and sequences thereof. By the term “abstract structure”, we refer to sets whose elements are no natural numbers. In this section, we will prepare the formalization of computability on abstract structures, following the presentations given in [20, 22, 24, 25].

In the previous section, we have stated that Turing machines are characterized by the set \mathcal{C}^* , which is a subset of the set \mathcal{N} . In other words, Turing machines characterize algorithms that operate on the natural numbers. Hence, the elements of an abstract structure are not directly accessible to Turing machines and thus need to be represented in a suitable manner. In particular, we

want to represent each element of the abstract structure in question by at least one natural number. In formal terms, for an abstract structure \mathcal{A} , we consider partial surjective mappings $\nu_{\mathcal{A}} : \mathbb{N} \hookrightarrow \mathcal{A}$. We refer to a mapping of this kind as *notation*.

Remark 4. In [20, Definition 2.3.1, p. 33], Weihrauch employs the word “notation” as a name for the concept of describing abstract objects by words of a formal language. For example, every definable object (to be precise: every definable set) in ZFC set theory can, by definition, be described by a formula in first-order predicate logic, see [30, Definition 2.8, p. 26] for details. A formal language may be encoded into the natural numbers, as was done by Turing in [29] in order to prove the equivalence of Turing machines and recursive functions, or earlier by Gödel in [31] in the context of his work on incompleteness theorems. On the other hand, natural numbers may be denoted by words of a formal language, as is the case for the usual representation of natural numbers by means of the Arabic numerals. Ultimately, both the use of formal languages and the use of natural numbers as “fundamental” structure lead to the same notion of computability on abstract structures.

As indicated at the end of the previous section, we will mostly concern ourselves with recursive functions regarding questions of computability, since they characterize the capabilities of real-world computers. Thus, we will restrict ourselves to considering the set \mathcal{C}^* within this section. In principle, all of the following considerations apply to general A -computable functions in the same manner.

The set \mathcal{C}_1^* of recursive functions in one variable yields a direct way to characterize recursive sets. For every recursive indicator function $\mathbb{1}_G$, there exists an $n \in \mathbb{N}$ such that $\mathbb{1}_G = \varphi_n$ holds true. Accordingly, we define the notation

$$n \mapsto G \quad :\Leftrightarrow \quad \varphi_n = \mathbb{1}_G \text{ for } G \subseteq \mathbb{N}. \quad (7)$$

This notation is (truly) partial in the following sense: a function φ_n satisfies $\varphi_n = \mathbb{1}_G$ for some recursive set $G \subseteq \mathbb{N}$ if and only if it is total and attains no

values other than 0 and 1. Clearly, there exists $n \in \mathbb{N}$ such that φ_n does not satisfy these requirements. Hence, only a proper subset of the natural numbers actually represents recursive sets with respect to the notation defined in (7).

In the following, we consider n -tuples $\mathbf{m} = (m_j)_{j=1}^n = (m_1, m_2, \dots, m_n)$ of natural numbers and, for $l \leq k \leq n$, the projection $[\mathbf{m}]_l^k = (m_j)_{j=l}^k$ on the subtuple $(m_j)_{j=l}^k = (m_l, \dots, m_k)$ consisting of those components of \mathbf{m} with index between l and k . In particular, for $l = k$, we write $[\mathbf{m}]_k = m_k$. Given tuples $\mathbf{m} \in \mathbb{N}^l$ and $\mathbf{s} \in \mathbb{N}^k$ for $l, k \in \mathbb{N}$, we define $\mathbf{m} \circ \mathbf{s} := (m_1, \dots, m_l, s_1, \dots, s_k) \in \mathbb{N}^{l+k}$.

Remark 5. For $k, n \in \mathbb{N}_+$ with $k \leq n$, the function $[\cdot]_k : \mathbb{N}^n \rightarrow \mathbb{N}$ is an element of the set of identity functions, which we have previously used to define the set \mathcal{C}^* of \emptyset -computable functions. Hence, the function $[\cdot]_k : \mathbb{N}^n \rightarrow \mathbb{N}$ is recursive by definition for all $k, n \in \mathbb{N}_+$ with $k \leq n$.

In order to extend the idea of notations to structures that involve tuples of natural numbers, we make use of the *Cantor pairing function* $\langle \cdot \rangle_2 : \mathbb{N}^2 \rightarrow \mathbb{N}$,

$$(m_1, m_2) \mapsto m_2 + \frac{1}{2}(m_1 + m_2)(m_1 + m_2 + 1),$$

which maps the set \mathbb{N}^2 bijectively to the set \mathbb{N} . For $n > 2$, the n -th extension $\langle \cdot \rangle_n : \mathbb{N}^n \rightarrow \mathbb{N}$ of the Cantor pairing function is defined inductively by

$$\begin{aligned} \langle \mathbf{m} \rangle_n &= \langle m_1, m_2, \dots, m_n \rangle_n \\ &:= \langle m_1, \langle [\mathbf{m}]_2^n \rangle_{n-1} \rangle_2 \\ &= \langle m_1, \langle m_2, m_3, \dots, m_n \rangle_{n-1} \rangle_2. \end{aligned}$$

For the sake of completeness, we also define the trivial Cantor “pairing” $\langle \cdot \rangle_1 : \mathbb{N} \rightarrow \mathbb{N}$, $m \mapsto m$. For all $n \in \mathbb{N}_+$, the function $\langle \cdot \rangle_n$ is total and for all $m \in \mathbb{N}$, there exists exactly one $\mathbf{m} \in \mathbb{N}^n$ such that $\langle \mathbf{m} \rangle_n = m$ holds true. Hence, the inverse Cantor pairing function $\Pi_n : \mathbb{N} \rightarrow \mathbb{N}^n$ is well-defined. Furthermore, $\langle \cdot \rangle_n$ is recursive, as is $[\Pi_n(\cdot)]_k$ for all $n, k \in \mathbb{N}_+$ that satisfy $k \geq n$. Using the inverse Cantor pairing function for $n = 3$, we can specify a notation for the set

of rational numbers \mathbb{Q} by defining

$$m \mapsto q := (-1)^{[\Pi_3(m)]_1} \frac{[\Pi_3(m)]_2}{1 + [\Pi_3(m)]_3}$$

for $m \in \mathbb{N}$.

We define $\varpi_1(\cdot) := [\Pi_2(\cdot)]_1$ and $\varpi_2(\cdot) := [\Pi_2(\cdot)]_2$ for the special case of $n = 2$. Then, inverse Cantor pairing function also yields a notation for the set of finite tuples of natural numbers $\bigcup_{n=1}^{\infty} \mathbb{N}^n$ by setting

$$m \mapsto \Pi_{\varpi_1(m)}(\varpi_2(m)),$$

which is particularly useful whenever we want to define a notation for the set of finite tuples of elements of some abstract structure \mathcal{A} . If \mathcal{A} admits a notation itself, a notation for $\bigcup_{n=1}^{\infty} \mathcal{A}^n$ can then be defined by means of composition.

For the sake of readability, we write $\Pi_k \varphi_n(m)$ instead of $\Pi_k(\varphi_n(m))$ in the following.

Definition 6. *An n -fold sequence of rational numbers $(r_{\mathbf{m}})_{\mathbf{m} \in \mathbb{N}^n}$ is said to be computable if there exist exists a number $k \in \mathbb{N}$ such that*

$$r_{\mathbf{m}} = (-1)^{[\Pi_3 \varphi_k(\langle \mathbf{m} \rangle_n)]_1} \frac{[\Pi_3 \varphi_k(\langle \mathbf{m} \rangle_n)]_2}{1 + [\Pi_3 \varphi_k(\langle \mathbf{m} \rangle_n)]_3}$$

holds true for all $\mathbf{m} \in \mathbb{N}^n$.

In general, real numbers defy exact computability by Turing machines due to their irrational and hence infinite nature. Practically relevant functions, like exp, sin and cos are not computable exactly, even when their domain is restricted to the rational numbers. Hence, a shift from the domain of exact computability, which we have considered so far, to the domain of approximate computability is necessary. In order for approximate computations to be meaningful, it is necessary to incorporate a procedure for estimating the approximation error.

Definition 7. *A sequence $(x_m)_{m \in \mathbb{N}}$ of real numbers is said to converge effectively towards a number $x_* \in \mathbb{R}$ if there exists a number $c \in \mathbb{N}$ such that $\xi := \varphi_c$ is a total recursive function and*

$$|x_* - x_m| < 1/2^M \tag{8}$$

holds true for all $m, M \in \mathbb{N}$ that satisfy $m \geq \xi(M)$.

The function $\xi : \mathbb{N} \rightarrow \mathbb{N}, M \mapsto \xi(M)$ is referred to as (recursive) *modulus of convergence* for the sequence $(x_m)_{m \in \mathbb{N}}$.

Definition 8. A real number x is said to be *computable* if there exists a *computable sequence of rational numbers that converges effectively towards x* .

We denote the set of computable real numbers by \mathbb{R}_c , by \mathbb{R}_c^{+0} the non-negative numbers in \mathbb{R}_c , and by $\mathbb{C}_c = \mathbb{R}_c + i\mathbb{R}_c$ the set of computable complex numbers. Prominent examples of computable, irrational numbers are $\sqrt{2}$, e , and π . Given a computable real number x , a pair $((r_m)_{m \in \mathbb{N}}, \xi)$ consisting of a computable sequence $(r_m)_{m \in \mathbb{N}}$ of rational numbers that satisfies $\lim_{m \rightarrow \infty} r_m = x$ and a corresponding recursive modulus of convergence ξ such that (8) holds true is called a *standard description* of the number x .

Remark 6. On the set of standard descriptions of computable real numbers, the computable real numbers induce an equivalence relation as follows: Two standard descriptions $((r_m)_{m \in \mathbb{N}}, \xi)$ and $((r'_m)_{m \in \mathbb{N}}, \xi')$ are equivalent, denoted by

$$((r_m)_{m \in \mathbb{N}}, \xi) \sim ((r'_m)_{m \in \mathbb{N}}, \xi'),$$

if they represent the same number $x \in \mathbb{R}_c$. By $((r_m)_{m \in \mathbb{N}}, \xi) \rightrightarrows x$, we indicate that $((r_m)_{m \in \mathbb{N}}, \xi)$ is a representative of $x \in \mathbb{R}_c$. In general, we will employ the symbol ' \rightrightarrows ' in the context of computation on abstract sets whenever we want to indicate that some abstract object is represented by some "less abstract" object which is accessible to Turing machines.

By Definition 6, the family $(\varphi_k)_{k \in \mathbb{N}}$ induces for all $n \in \mathbb{N}_+$ a notation for the set of n -fold computable sequences of rational numbers. Since Definition 6 implicitly requires the function φ_k to satisfy $D(\varphi_k) = \mathbb{N}$, the notation is partial. Furthermore, the mapping

$$l \mapsto (\varpi_1(l), \varpi_2(l)) =: (k, c) \mapsto (\varphi_k, \varphi_c), \quad (9)$$

provides a notation for the set \mathbb{R}_c . For every standard description $((r_m)_{m \in \mathbb{N}}, \xi)$ of some computable number x , there exist $k, c \in \mathbb{N}$, such that φ_k characterizes the sequence $(r_m)_{m \in \mathbb{N}}$ in the sense of Definition 6 and φ_c characterizes the function ξ in the sense of Definition 7.

Definition 9. An n -fold sequence $(x_{\mathbf{m}})_{\mathbf{m} \in \mathbb{N}^n}$ of computable real numbers is called *computable* if there exists an $(n + 1)$ -fold computable sequence $(r_{\mathbf{m} \circ s})_{\mathbf{m} \circ s \in \mathbb{N}^{n+1}}$ of rational numbers as well as a number $c \in \mathbb{N}$ such that $\xi := \varphi_c$ is a total recursive function and

$$|x_{\mathbf{m}} - r_{\mathbf{m} \circ s}| < \frac{1}{2^M}$$

holds true for all $s \in \mathbb{N}$, $\mathbf{m} \in \mathbb{N}^n$, $M \in \mathbb{N}$ that satisfy $s \geq \xi(\langle \mathbf{m} \circ M \rangle_{n+1})$.

The pair $((r_{\mathbf{m} \circ s})_{\mathbf{m} \circ s \in \mathbb{N}^{n+1}}, \xi)$ is referred to as a *standard description* of the sequence $(x_{\mathbf{m}})_{\mathbf{m} \in \mathbb{N}^n}$. Again, the mapping $l \mapsto (\varphi_k, \varphi_c)$ defined in (9) provides a notation for set of n -fold computable sequences of computable numbers. Every standard description $((r_{\mathbf{m} \circ s})_{\mathbf{m} \circ s \in \mathbb{N}^{n+1}}, \xi)$ of some sequence $(x_{\mathbf{m}})_{\mathbf{m} \in \mathbb{N}^n}$ may be characterized by a pair (φ_k, φ_c) according to Definitions 6 and 7, analogous to the notation for the set \mathbb{R}_c defined above.

Finally, returning to the domain of exact computation, we introduce for all $n \in \mathbb{N}_+$ a notation for the set \mathcal{C}_n^* of recursive functions in n variables by defining

$$m \mapsto \varphi_m(\langle \cdot \rangle_n). \quad (10)$$

Clearly, $\varphi_m(\langle \cdot \rangle_n) : \mathbb{N}^n \leftrightarrow \mathbb{N}$ is a recursive function for all $m \in \mathbb{N}$. On the other hand, consider an arbitrary recursive function $g : \mathbb{N}^n \leftrightarrow \mathbb{N}$. Then, $g(\Pi_n(\cdot))$ is a recursive function in one variable. Consequently, there exists an $m \in \mathbb{N}$ such that $\varphi_m(\cdot) = g(\Pi_n(\cdot))$ holds true and we have

$$\varphi_m(\langle \cdot \rangle_n) = g(\Pi_n(\langle \cdot \rangle_n)) = g(\cdot).$$

Hence, (10) yields for all $n \in \mathbb{N}_+$ a notation for the set \mathcal{C}_n^* . By setting

$$m \mapsto \varphi_{\varpi_1(m)}(\langle \cdot \rangle_{\varpi_2(m)}),$$

we extend this notation to a notation for the set \mathcal{C}^* of all recursive functions.

5. Computations and Algorithms on Abstract Structures

In the previous section, we have introduced the notion of notations. In this section, we will employ notations to formally define the idea of *computability on a Turing machine* for abstract structures.

Let \mathcal{A} and \mathcal{A}' be two abstract structures with fixed notations $\nu_{\mathcal{A}} : \mathbb{N} \hookrightarrow \mathcal{A}$ and $\nu_{\mathcal{A}'} : \mathbb{N} \hookrightarrow \mathcal{A}'$. That is, $\nu_{\mathcal{A}}$ and $\nu_{\mathcal{A}'}$ are (not necessarily total) surjections from \mathbb{N} onto \mathcal{A} and \mathcal{A}' , respectively. Furthermore, consider a mapping $\mathcal{G} : \mathcal{A} \hookrightarrow \mathcal{A}'$. If there exists a recursive function $g : \mathbb{N} \hookrightarrow \mathbb{N}$ such that for all pairs $(a, n) \in \mathcal{A} \times \mathbb{N}$, we have

$$a \in D(\mathcal{G}) \wedge \nu_{\mathcal{A}}(n) = a \quad \Rightarrow \quad \nu_{\mathcal{A}'}(g(n)) = \mathcal{G}(a),$$

then we say that *there exists a Turing machine $TM_{\mathcal{G}}$ that returns $a' := \mathcal{G}(a)$ for input a* . This Turing machine computes the mapping $a \mapsto a' := \mathcal{G}(a)$ in the following sense:

- Whenever $TM_{\mathcal{G}}$ is presented with a number n that *denotes* the object $a = \nu_{\mathcal{A}}(n) \in D(\mathcal{G})$ with respect to the notation $\nu_{\mathcal{A}}$, it returns a number $g(n)$ that *denotes* the object $a' = \nu_{\mathcal{A}'}(g(n)) \in \mathcal{A}'$ with respect to the notation $\nu_{\mathcal{A}'}$.
- If \mathcal{G} is undefined at the point $a = \nu_{\mathcal{A}}(n) \in \mathcal{A}$, then either $TM_{\mathcal{G}}$ does not halt in a finite number of steps for input n , or $g(n)$ is *not* an element of $D(\nu_{\mathcal{A}'})$.
- In particular, $TM_{\mathcal{G}}$ maps the set of numbers that denote the object $a \in \mathcal{A}$ with respect to $\nu_{\mathcal{A}}$ to a subset of the set of numbers that denote the object $\mathcal{G}(a) \in \mathcal{A}'$ with respect to $\nu_{\mathcal{A}'}$, i.e., for all $a \in D(\mathcal{G})$, we have

$$g(\{n : \nu_{\mathcal{A}}(n) = a\}) \subseteq \{n : \nu_{\mathcal{A}'}(n) = \mathcal{G}(a)\}.$$

With some abuse of notation, we also write $TM_{\mathcal{G}} : \mathcal{A} \rightarrow \mathcal{A}'$, $a \mapsto TM_{\mathcal{G}}(a) = a'$, despite the fact that $TM_{\mathcal{G}}$ actually computes a mapping on natural numbers rather than a mapping on the abstract structures \mathcal{A} and \mathcal{A}' .

Remark 7. *If both \mathcal{A} and \mathcal{A}' are either the set of computable reals \mathbb{R}_c or the set of computable complex numbers \mathbb{C}_c , then a function $f : \mathcal{A} \hookrightarrow \mathcal{A}'$ that satisfies the above notion of computability is referred to as Markov computable. Markov computability is not the strongest notion of computability on \mathbb{R}_c and \mathbb{C}_c that has been used in the literature: Turing computability, which is applicable to functions on all of \mathbb{R} and \mathbb{C} , implies Markov computability when restricted to \mathbb{R}_c and \mathbb{C}_c . For a comparison, see [23, Appendix 2.9, p. 21].*

The majority of notations introduced in Section 4 is induced by the enumeration $(\varphi_n)_{n \in \mathbb{N}}$. If the notations of two abstract structures \mathcal{A} and \mathcal{A}' are induced by a recursive enumeration of \mathcal{C}_1^* , then any mapping $\mathcal{G} : \mathcal{A} \hookrightarrow \mathcal{A}'$ is, in principle, a transformation of recursive functions. In the following, we will introduce a consequence of the *s-m-n Theorem* [24, Theorem 3.5, p. 16], which captures a fundamental structural property of the set \mathcal{C}^* . In particular, we will characterize the conditions under which a transformation of recursive functions is computable on a Turing machine.

Consider the following case: for a tuple $\mathbf{m} \in \mathbb{N}^n$, the abstract structure \mathcal{A} equals the set $\mathcal{C}_{m_1}^* \times \cdots \times \mathcal{C}_{m_n}^*$, while the abstract structure \mathcal{A}' equals the set \mathcal{C}^* . For all $j \in \{1, \dots, n\}$, the set $\mathcal{C}_{m_j}^*$ is equipped with an individual notation. We can extend the individual notations to a joint notation $\nu_{\mathcal{A}}$ for the set $\mathcal{C}_{m_1}^* \times \cdots \times \mathcal{C}_{m_n}^*$ by first enumerating all n -tuples of natural numbers and then applying the individual notations to the respective component. Furthermore, for the set \mathcal{C}^* , a notation $\nu_{\mathcal{A}'}$ was introduced at the end of Section 4. Now, consider a mapping

$$\begin{aligned} \mathcal{G} & : \mathcal{C}_{m_1}^* \times \cdots \times \mathcal{C}_{m_n}^* & \rightarrow & \mathcal{C}^*, \\ & (g_1, g_2, \dots, g_n) & \mapsto & h, \end{aligned}$$

which, for some number $k \in \mathbb{N}_+$, is of the form

$$\begin{aligned} (g_j)_{j=1}^n & \quad \quad \quad =: \mathbf{h}_0 & \mapsto & h_1, \\ (g_j)_{j=1}^n \circ (h_j)_{j=1}^1 & =: \mathbf{h}_1 & \mapsto & h_2, \\ & & & \vdots \\ (g_j)_{j=1}^n \circ (h_j)_{j=1}^{k-1} & =: \mathbf{h}_{k-1} & \mapsto & h_k =: h, \end{aligned}$$

such that for all $j \in \{1, \dots, k\}$, the function h_j is either an element of $\mathcal{C}(\emptyset)$ or emerges from \mathbf{h}_{j-1} through concatenation, primitive recursion or unbounded search. Then, there exists a mapping $g' \in \mathcal{C}_1^*$ such that for all $l \in \mathbb{N}$ and all $(g_j)_{j=1}^n \in \mathcal{C}_{m_1}^* \times \dots \times \mathcal{C}_{m_n}^*$ that satisfy $\nu_{\mathcal{A}}(l) = (g_j)_{j=1}^n$, we have $\nu_{\mathcal{A}'}(g'(l)) = h$. That is, there exists a Turing machine $\text{TM}_{\mathcal{G}}$ that computes the mapping $(g_1, g_2, \dots, g_n) \mapsto h$.

As a rule of thumb, if we can implement a mapping $\mathcal{G} : \mathcal{C}_{m_1}^* \times \dots \times \mathcal{C}_{m_n}^* \rightarrow \mathcal{C}^*$ on a real world computer, it is computable by a Turing machine, which applies to all common arithmetic and logic operations. Throughout the rest of this work, we make implicit use of this principle on several occasions.

6. The Arithmetical Hierarchy of Real Numbers

Since Turing published his work on the theory of computation, it has been known that almost all real numbers are uncomputable. In an attempt to characterize different degrees of (un)computability, *Zheng and Weihrauch* introduced the arithmetical hierarchy of real numbers [32], which is strongly related to the *Kleene–Mostowski hierarchy* of subsets of natural numbers [33, 34]. The hierarchical level of a real number solely depends on the logical structure that is used to define the number.

Definition 10 (Kleene–Mostowski Hierarchy, cf. [33, 34]). *For $n \in \mathbb{N}_+$, consider $\mathbf{m} = (m_1, \dots, m_n) \in \mathbb{N}^n$. Then, the sets $\Sigma_n^0 \subsetneq 2^{\mathbb{N}}$, $\Pi_n^0 \subsetneq 2^{\mathbb{N}}$ and $\Delta_n^0 \subsetneq 2^{\mathbb{N}}$ are defined as follows:*

- A set $A \subseteq \mathbb{N}$ satisfies $A \in \Sigma_n^0$ if there exists a recursive set $G \subseteq \mathbb{N}$ such that for all $j \in \mathbb{N}$, we have $j \in A$ if and only if

$$\exists m_1 \forall m_2 \exists m_3 \dots \mathbf{Q} m_n (\langle \mathbf{m} \circ j \rangle_{n+1} \in G)$$

holds true, where “Q” is replaced by “ \forall ” if n is even and by “ \exists ” if n is odd.

- A set $A \subseteq \mathbb{N}$ satisfies $A \in \Pi_n^0$ if there exists a recursive set $G \subseteq \mathbb{N}$ such

that for all $j \in \mathbb{N}$, we have $j \in A$ if and only if

$$\forall m_1 \exists m_2 \forall m_3 \dots \mathbf{Q} m_n (\langle \mathbf{m} \circ j \rangle_{n+1} \in G)$$

holds true, where “ \mathbf{Q} ” is replaced by “ \exists ” if n is even and by “ \forall ” if n is odd.

- A set $A \subseteq \mathbb{N}$ satisfies $A \in \Delta_n^0$ if satisfies both $A \in \Sigma_n^0$ and $A \in \Pi_n^0$. Hence, we have $\Delta_n^0 = \Sigma_n^0 \cap \Pi_n^0$.

Definition 11 (Zheng-Weihrauch Hierarchy, cf. [32]). For $n \in \mathbb{N}_+$, consider $\mathbf{m} = (m_1, \dots, m_n) \in \mathbb{N}^n$. Then, the sets $\Sigma_n \subsetneq \mathbb{R}$, $\Pi_n \subsetneq \mathbb{R}$ and $\Delta_n \subsetneq \mathbb{R}$ are defined as follows:

- A number $x_* \in \mathbb{R}$ satisfies $x_* \in \Sigma_n$ if there exists an n -fold computable sequence $(r_{\mathbf{m}})_{\mathbf{m} \in \mathbb{N}^n}$ of rational numbers such that

$$x_* = \sup_{m_1 \in \mathbb{N}} \inf_{m_2 \in \mathbb{N}} \sup_{m_3 \in \mathbb{N}} \dots \Theta_{m_n \in \mathbb{N}} (r_{\mathbf{m}})$$

holds true, where “ Θ ” is replaced by “ \inf ” if n is even and by “ \sup ” if n is odd.

- A number $x_* \in \mathbb{R}$ satisfies $x_* \in \Pi_n$ if there exists an n -fold computable sequence $(r_{\mathbf{m}})_{\mathbf{m} \in \mathbb{N}^n}$ of rational numbers such that

$$x_* = \inf_{m_1 \in \mathbb{N}} \sup_{m_2 \in \mathbb{N}} \inf_{m_3 \in \mathbb{N}} \dots \Theta_{m_n \in \mathbb{N}} (r_{\mathbf{m}})$$

holds true, where “ Θ ” is replaced by “ \sup ” if n is even and by “ \inf ” if n is odd.

- A number $x_* \in \mathbb{R}$ satisfies $x_* \in \Delta_n$ if satisfies both $x_* \in \Sigma_n$ and $x_* \in \Pi_n$. Hence, we have $\Delta_n = \Sigma_n \cap \Pi_n$.

Remark 8. A real number x is computable if and only if it satisfies both $x \in \Pi_1$ and $x \in \Sigma_1$. Hence, we have $\mathbb{R}_c = \Delta_1$.

Given a set $A \subseteq \mathbb{N}$, we denote $x[A] := \sum_{j \in A} 1/2^{(j+1)}$. Zheng and Weihrauch showed that for all $n \in \mathbb{N}_+$, if A satisfies $A \in \Sigma_n^0$, $A \in \Pi_n^0$ or $A \in \Delta_n^0$, then $x[A]$ satisfies $x[A] \in \Sigma_n$, $x[A] \in \Pi_n$ or $x[A] \in \Delta_n$, respectively.

Assuming the finiteness of the respective suprema and infima, an n -fold computable sequence $(x_{\mathbf{m}})_{\mathbf{m} \in \mathbb{N}^n}$ of computable numbers is referred to as n -th order lower Zheng-Weihrauch (ZW) description of the real number

$$x_* = \sup_{m_1 \in \mathbb{N}} \inf_{m_2 \in \mathbb{N}} \sup_{m_3 \in \mathbb{N}} \dots \Theta_{m_n \in \mathbb{N}} (x_{\mathbf{m}}),$$

where “ Θ ” is replaced by “inf” if n is even and by “sup” if n is odd.

Likewise, again assuming the finiteness of the respective suprema and infima, an n -fold computable sequence $(x_{\mathbf{m}})_{\mathbf{m} \in \mathbb{N}^n}$ of computable numbers is referred to as n -th order upper Zheng-Weihrauch (ZW) description of the real number

$$x_* = \inf_{m_1 \in \mathbb{N}} \sup_{m_2 \in \mathbb{N}} \inf_{m_3 \in \mathbb{N}} \dots \Theta_{m_n \in \mathbb{N}} (x_{\mathbf{m}}),$$

where “ Θ ” is replaced by “sup” if n is even and by “inf” if n is odd.

Lemma 1. *If there exists an n -th order upper ZW description for a number $x_* \in \mathbb{R}$, then $x_* \in \Pi_n$ is satisfied.*

Proof. Consider the standard description $((r'_s)_{s \in \mathbb{N}^{n+1}}, \xi)$ of an n -th order upper ZW description $(x_{\mathbf{m}})_{\mathbf{m} \in \mathbb{N}^n}$ of x_* and define

$$g'(\mathbf{m}) := (\varpi_1(m_1), m_2, \dots, m_n),$$

$$g(\mathbf{m}) := g'(\mathbf{m}) \circ \xi(g'(\mathbf{m}), \varpi_2(m_1)),$$

$$r_{\mathbf{m}} := r'_{g(\mathbf{m})} + \frac{1}{2^{\varpi_2(m_1)}},$$

for all $\mathbf{m} \in \mathbb{N}^n$. Then, $(r_{\mathbf{m}})_{\mathbf{m} \in \mathbb{N}^n}$ is an n -fold computable sequence of rational numbers that satisfies $r_{\mathbf{m}} \geq x_{g'(\mathbf{m})}$ for all $\mathbf{m} \in \mathbb{N}^n$, and hence

$$\begin{aligned} x_* &\leq \sup_{m_2 \in \mathbb{N}} \inf_{m_3 \in \mathbb{N}} \dots \Theta_{m_n \in \mathbb{N}} (x_{g'(\mathbf{m})}) \\ &\leq \sup_{m_2 \in \mathbb{N}} \inf_{m_3 \in \mathbb{N}} \dots \Theta_{m_n \in \mathbb{N}} (r_{\mathbf{m}}) \end{aligned} \quad (11)$$

for all $m_1 \in \mathbb{N}$ as well. Since $((r'_s)_{s \in \mathbb{N}^{n+1}}, \xi)$ is a standard description of $(x_{\mathbf{m}})_{\mathbf{m} \in \mathbb{N}}$, we have

$$\begin{aligned} &\left| \sup_{m_2 \in \mathbb{N}} \inf_{m_3 \in \mathbb{N}} \dots \Theta_{m_n \in \mathbb{N}} (x_{\mathbf{m}}) \dots \right. \\ &\quad \left. \dots - \sup_{m_2 \in \mathbb{N}} \inf_{m_3 \in \mathbb{N}} \dots \Theta_{m_n \in \mathbb{N}} \left(r'_{\mathbf{m} \circ \xi((\mathbf{m}), M)} \right) \right| < \frac{1}{2^M} \end{aligned}$$

for all $m_1, M \in \mathbb{N}$. Since furthermore, $(x_{\mathbf{m}})_{\mathbf{m} \in \mathbb{N}}$ is an upper ZW description of x_* , we conclude that for $\epsilon > 0$ arbitrary, there exist $l, k \in \mathbb{N}$ such that for $\mathbf{m}' := (l, m_2, \dots, m_n) \in \mathbb{N}^n$,

$$x_* + \epsilon \geq \sup_{m_2 \in \mathbb{N}} \inf_{m_3 \in \mathbb{N}} \dots \Theta_{m_n \in \mathbb{N}} \left(r'_{\mathbf{m}' \circ \xi((\mathbf{m}'), k)} \right) - \frac{1}{2^k}$$

holds true. The mapping $\langle \cdot \rangle_2^{-1} : \mathbb{N} \rightarrow \mathbb{N}^2$ is bijective. Hence, for all $l, k \in \mathbb{N}$, there exists $m_1 \in \mathbb{N}$ such that $\langle m_1 \rangle_2^{-1} = (\varpi_1(m_1), \varpi_2(m_1)) = (l, k)$ holds true. Consequently, there exists $m_1 \in \mathbb{N}$ such that

$$x_* + \epsilon \geq \sup_{m_2 \in \mathbb{N}} \inf_{m_3 \in \mathbb{N}} \dots \Theta_{m_n \in \mathbb{N}} (r_{\mathbf{m}}) \quad (12)$$

holds true. Joining (11) and (12), we conclude that

$$x_* + \epsilon \geq \inf_{m_1 \in \mathbb{N}} \sup_{m_2 \in \mathbb{N}} \sup_{m_3 \in \mathbb{N}} \dots \Theta_{m_n \in \mathbb{N}} (r_{\mathbf{m}}) \geq x_*$$

is satisfied. Since $\epsilon > 0$ was chosen arbitrarily, the claim follows. \square

Remark 9. *From the definition of computable sequences of computable numbers, it is imminent that every computable sequence of rational numbers is also a computable sequence of computable numbers. Hence, a converse to Lemma 1 is straightforward to prove, and we conclude that if a number $x_* \in \mathbb{R}$ satisfies $x_* \in \Pi_n$, there exists an n -th order upper ZW description for x_* . Likewise, if a number $y_* \in \mathbb{R}$ satisfies $y_* \in \Sigma_n$, there exists an n -th order lower ZW description for y_* . Furthermore, the line of reasoning presented in the proof of Lemma 1 applies analogously to n -th order lower ZW description of some number $y_* \in \mathbb{R}$. In summary, we arrive at the following: a number $x_* \in \mathbb{R}$ satisfies $x_* \in \Pi_n$ if and only if there exists an n -th order upper ZW description for x_* , while a number $y_* \in \mathbb{R}$ satisfies $y_* \in \Sigma_n$ if and only if there exists an n -th order lower ZW description for y_* .*

The following lemma was stated in a slightly different form by Zheng and Weihrauch. A close examination shows that the corresponding proof is constructive and exclusively employs operations that can be computed on a Turing machine. Hence, the derivation provided by Zheng and Weihrauch is sufficient to prove the subsequent version of the lemma.

Lemma 2 ([32, Lemma 3.2, p. 55]). *Let $(r_{\mathbf{m}})_{\mathbf{m} \in \mathbb{N}^2}$ be a computable double sequence of rational numbers such that $\inf_{m_1 \in \mathbb{N}} \sup_{m_2 \in \mathbb{N}} (r_{\mathbf{m}})$ exists. There exists a Turing machine which computes a mapping $(r_{\mathbf{m}})_{\mathbf{m} \in \mathbb{N}^2} \mapsto (r'_m)_{m \in \mathbb{N}}$ such that $(r'_m)_{m \in \mathbb{N}}$ is a computable sequence of rational numbers and $\limsup_{m \rightarrow \infty} (r'_m) = \inf_{m_1 \in \mathbb{N}} \sup_{m_2 \in \mathbb{N}} (r_{\mathbf{m}})$ is satisfied. \square*

In Section 3, we introduced the totality set $A_{\mathbb{T}}^{\Phi}$ and its relation to the family $(D(\varphi_n))_{n \in \mathbb{N}}$. As indicated, these sets will play a role in the context of oracle computations. In the following, we will characterize $A_{\mathbb{T}}^{\Phi}$ and $(D(\varphi_n))_{n \in \mathbb{N}}$ with respect to the Kleene–Mostowski hierarchy.

Lemma 3. *For all $n \in \mathbb{N}$, we have $D(\varphi_n) \in \Sigma_1^0$.*

Proof. Define $G_n^{\Psi} := \{m \in \mathbb{N} : \Psi(n, \varpi_2(m), \varpi_1(m)) = 1\}$. Since Ψ is a total recursive function, the set G_n^{Ψ} is recursive. Furthermore, incorporating the definition of Ψ , we have

$$\begin{aligned} D(\varphi_n) &= \{j \in \mathbb{N} : \exists k \in \mathbb{N} : \Psi(n, j, k) = 1\} \\ &= \{j \in \mathbb{N} : \exists k \in \mathbb{N} : \langle k, j \rangle_2 \in G_n^{\Psi}\}. \end{aligned}$$

The claim then follows from Definition 10. \square

Lemma 4. *The totality set $A_{\mathbb{T}}^{\Phi}$ satisfies $A_{\mathbb{T}}^{\Phi} \in \Pi_2^0$.*

Proof. Again, the claim follows from the existence of a suitable recursive set, which is induced by the runtime function Ψ . Define

$$G^{\Psi} := \{m \in \mathbb{N} : \Psi([\Pi_3(m)]_3, [\Pi_3(m)]_1, [\Pi_3(m)]_2) = 1\}.$$

Since Ψ is a total recursive function, the set G^{Ψ} is recursive. Furthermore, by Definition 5, we have

$$\begin{aligned} A_{\mathbb{T}}^{\Phi} &= \{n \in \mathbb{N} : \forall j \in \mathbb{N} : \exists k \in \mathbb{N} : \Psi(n, j, k) = 1\} \\ &= \{n \in \mathbb{N} : \forall j \in \mathbb{N} : \exists k \in \mathbb{N} : \langle j, k, n \rangle_3 \in G^{\Psi}\}. \end{aligned}$$

The claim then follows from Definition 10. \square

7. Computable Bandlimited Signals

Having introduced a framework for computable analysis, we are now equipped to formalize the concept of computable bandlimited signals. Again, for the sake of self-containedness, we repeat some of the definitions found in [16].

Definition 12. *We call a signal f elementary computable if there exists a natural number L and a sequence of computable numbers $(c_k)_{k=-L}^L$ that satisfy*

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

The building blocks of an elementary computable signal are sinc functions. Hence, elementary computable signals are exactly those functions that can be represented by a finite Shannon sampling series with computable coefficients $(c_k)_{k=-L}^L$. Note that every elementary computable signal f is a finite sum of computable continuous functions and hence a computable continuous function. As a consequence, for every $t \in \mathbb{R}_c$ the number $f(t)$ is computable. Further, the sum of finitely many elementary computable signals is elementary computable, as well as the product of an elementary computable signal with a computable number.

Definition 13. *A signal in $f \in \mathcal{B}_\pi^p$, $p \in (1, \infty) \cap \mathbb{R}_c$, is called computable in \mathcal{B}_π^p if there exists a computable sequence of elementary computable signals $(f_m)_{m \in \mathbb{N}}$ and a recursive function $\xi: \mathbb{N} \rightarrow \mathbb{N}$ such that for all $M \in \mathbb{N}$ we have*

$$\|f - f_m\|_{\mathcal{B}_\pi^p} \leq \frac{1}{2^M}$$

for all $m \geq \xi(M)$. By \mathcal{CB}_π^p , $p \in (1, \infty) \cap \mathbb{R}_c$, we denote the set of all signals in \mathcal{B}_π^p that are computable in \mathcal{B}_π^p .

According to this definition we can approximate any signal $f \in \mathcal{CB}_\pi^p$, $p \in (1, \infty) \cap \mathbb{R}_c$, by an elementary computable signal, where we have an “effective”, i.e. computable control of the approximation error. For every prescribed approximation error $\epsilon > 0$, $\epsilon \in \mathbb{R}_c$, we can compute a number $M \in \mathbb{N}$ such that

$M \geq -\log_2(\epsilon)$ holds true. Hence, the approximation error $\|f - f_m\|_{\mathcal{B}_\pi^p}$ is less than or equal to ϵ for all $m \geq \xi(M)$.

We finally give the definition of computability for bandlimited signals in \mathcal{E}_π .

Definition 14. We call a signal $f \in \mathcal{E}_\pi$ a *computable bandlimited signal* if the coefficients $(a_n)_{n \in \mathbb{N}}$ of the Taylor series (3) form a computable sequence of computable numbers. By \mathcal{CE}_π we denote the set of all signals in \mathcal{E}_π that are computable.

Note that a signal $f \in \mathcal{CE}_\pi$ is completely determined by the computable sequence $(a_n)_{n \in \mathbb{N}}$ of computable numbers according to the representation (3). Hence, a program for $(a_n)_{n \in \mathbb{N}}$ gives a complete description of f .

For $f_1, f_2 \in \mathcal{CE}_\pi$ and $\alpha_1, \alpha_2 \in \mathbb{C}_c$, we have $\alpha_1 f_1 + \alpha_2 f_2 \in \mathcal{CE}_\pi$, i.e., \mathcal{CE}_π has a linear structure.

Remark 10. We have $\mathcal{CB}_\pi^1 \subsetneq \mathcal{CB}_\pi^2 \subsetneq \mathcal{CE}_\pi$, which shows that \mathcal{CE}_π is the largest of these three sets.

Remark 11. For $f \in \mathcal{B}_\pi^2$, it follows from the properties of the Fourier transform \hat{f} that the family $(a_n)_{n \in \mathbb{N}}$ of Taylor coefficients satisfies

$$a_n = \frac{1}{2\pi} \int_{-\pi}^{\pi} \hat{f}(\omega) (i\omega)^n d\omega$$

for all $n \in \mathbb{N}$. If f furthermore satisfies $f \in \mathcal{CB}_\pi^2$, the Fourier transform \hat{f} is a computable signal in $L^2[-\pi, \pi]$ and the family of coefficients $(a_n)_{n \in \mathbb{N}}$ forms a computable sequence of computable numbers.

Definition 15. Let f satisfy $f \in \mathcal{E}_\pi$. If there exists a computable double sequence $(p_{\mathbf{m}})_{\mathbf{m} \in \mathbb{N}^2}$ of rational polynomials as well as a recursive function $\xi : \mathbb{N}^2 \rightarrow \mathbb{N}$ such that

$$|f(z) - p_{m_1, m_2}(z)| < \frac{1}{2^M}$$

holds true for all $z \in \mathbb{C}, m_1, m_2, M \in \mathbb{N}$ that satisfy $|z| \leq m_2$ and $m_1 \geq \xi(M, m_2)$, then f is referred to as *Weierstrass effective*. We denote \mathcal{WE}_π the set of Weierstrass effective π -bandlimited entire functions.

Theorem 1. *The sets \mathcal{CE}_π and \mathcal{WE}_π coincide.*

Proof. We start by proving that $f \in \mathcal{CE}_\pi \Rightarrow f \in \mathcal{WE}_\pi$ holds true for all $f \in \mathcal{CE}_\pi$. Let $f \in \mathcal{CE}_\pi$ satisfy (3), where $(a_n)_{n \in \mathbb{N}}$ is a computable sequence of computable numbers with standard description $((r_{n,m})_{n,m \in \mathbb{N}}, \xi)$. Without loss of generality, assume that ξ is the identity function, i.e., $(r_{n,m})_{n,m \in \mathbb{N}}$ and $(a_n)_{n \in \mathbb{N}}$ satisfy

$$|a_n - r_{n,m}| < \frac{1}{2^m}$$

for all $n, m \in \mathbb{N}$. Since $f \in \mathcal{CE}_\pi$ holds true, there exists $L \in \mathbb{N}$ such that

$$\sqrt[n]{|a_n|} \leq L$$

is satisfied for all $n \in \mathbb{N}$. For $K, m \in \mathbb{N}, z \in \mathbb{C}$, define

$$\begin{aligned} \bar{f}_{K,m}(z) &:= f(z) - \sum_{n=0}^K \frac{r_{n,m}}{n!} t^n \\ &= \left(\sum_{n=0}^K \frac{a_n - r_{n,m}}{n!} t^n \right) + \left(\sum_{n=K+1}^{\infty} \frac{a_n}{n!} t^n \right). \end{aligned}$$

The function $\bar{f}_{K,m}$ characterizes the error that arises when approximating the signal f by a computable polynomial based on the standard description $((r_{n,m})_{n,m \in \mathbb{N}}, \xi)$ of $(a_n)_{n \in \mathbb{N}}$. We will prove the first part of the theorem by deriving a computable upper bound on $\bar{f}_{K,m}$. Set $\bar{a}_{n,m} := a_n - r_{n,m}$ for all $n, m \in \mathbb{N}$. We have

$$\begin{aligned} \max_{|z| \leq J} |\bar{f}_{K,m}(z)| &\leq \left(\sum_{n=0}^K \frac{|\bar{a}_{n,m}|}{n!} J^n \right) + \left(\sum_{n=K+1}^{\infty} \frac{L^n}{n!} J^n \right) \\ &\leq \left(\sum_{n=0}^K \frac{1/2^m}{n!} J^n \right) + \left(\sum_{n=K+1}^{\infty} \frac{L^n}{n!} J^n \right) \quad (13) \\ &\leq \frac{1}{2^m} 4^J + \sum_{n=K+1}^{\infty} \left(\frac{LJ}{(1/4)n} \right)^n \\ &= 2^{2J-m} + \sum_{n=K+1}^{\infty} \left(\frac{4LJ}{n} \right)^n \\ &\leq 2^{2J-m} + \sum_{n=K+1}^{\infty} \underbrace{\left(\frac{4LJ}{K+1} \right)^n}_{=: x(L,J)}, \end{aligned}$$

where the bound on the first sum in (13) follows from the series expansion of the function $x \mapsto e^x$ and the bound on the second sum in (13) follows from the inequality $n! \geq (1/4 \cdot n)^n$, which in turn follows from *Stirling's approximation* for factorials. The remaining sum converges if $x(L, J)$ is sufficiently close to 0. In particular,

$$\sum_{n=K+1}^{\infty} x^n = x^{K+1} \frac{1}{1-x}$$

holds true for all $x \in [0, 1)$. Hence, consider $K \in \mathbb{N}$ such that $K + 1 > 8LJ$ is satisfied. Then, we have

$$\max_{|z| \leq J} \left| \bar{f}_{K,m}(z) \right| \leq 2^{2J-m} + 2^{-K}.$$

For $M, J \in \mathbb{N}, z \in \mathbb{C}$, we define

$$\begin{aligned} N(M, J) &:= 2J + M + 2, \\ K(M, J) &:= \max\{8LJ, M + 2\}, \\ p_{M,J}(z) &:= \sum_{n=0}^{K(M,J)} \frac{r_{n,N(M,J)}}{n!} z^n. \end{aligned}$$

Then, $(p_{M,J})_{M,J \in \mathbb{N}}$ is a computable family of polynomials such that

$$|f(z) - p_{M,J}(z)| < \frac{1}{2^M}$$

holds true for all $M, J \in \mathbb{N}, z \in \mathbb{C}$ that satisfy $|z| \leq J$. Hence, f satisfies $f \in \mathcal{WE}_\pi$.

It remains to show that $f \in \mathcal{WE}_\pi \Rightarrow f \in \mathcal{CE}_\pi$ holds true for all $f \in \mathcal{WE}_\pi$. Since f is an entire function, there exists a unique sequence $(a_n)_{n \in \mathbb{N}}$ of real numbers such that (3) holds true. Hence, we can prove the second part of the theorem by showing that $(a_n)_{n \in \mathbb{N}}$ is indeed a computable sequence of computable numbers, whenever f satisfies $f \in \mathcal{WE}_\pi$. Observe that by *Cauchy's integral formula*, we have

$$\frac{a_n}{n!} = \frac{1}{2\pi i} \oint_{|z|=1} \frac{f(z)}{z^{n+1}} dz.$$

Assume the pair $((p_{\mathbf{m}})_{\mathbf{m} \in \mathbb{N}^2}, \xi)$ satisfies the requirements of Definition 15. Furthermore, assume without loss of generality that $\xi(m_1, m_2) = m_1$ holds true for all $m_1, m_2 \in \mathbb{N}$, i.e., $(p_{\mathbf{m}})_{\mathbf{m} \in \mathbb{N}^2}$ satisfies

$$|f(z) - p_{m_1, m_2}(z)| < \frac{1}{2^{m_1}}$$

for all $m_1, m_2 \in \mathbb{N}, z \in \mathbb{C}$ that satisfy $|z| \leq m_2$. In the following, for $m \in \mathbb{N}$, consider the family $(c_k^m)_{k=0}^{K(m)}$ of coefficients of $p_{m,1}$. That is, we have

$$p_{m,1}(z) = \sum_{k=0}^{K(m)} c_k^m \cdot z^k$$

for all $z \in \mathbb{C}, m \in \mathbb{N}$. Then, for $m \in \mathbb{N}, z \in \mathbb{C}$, we define

$$r_{n,m} := \frac{n!}{2\pi i} \oint_{|z|=1} \frac{p_{m,1}(z)}{z^{n+1}} dz = c_n^m.$$

We now show that the sequence $(r_{n,m})_{m \in \mathbb{N}}$ converges effectively towards a_n for all $n \in \mathbb{N}$, with $|a_n - r_{n,m}| \leq n! \cdot 1/2^m$ satisfied for all $m \in \mathbb{N}$. We have

$$\begin{aligned} |a_n - r_{n,m}| &\leq n! \int_{-1/2}^{1/2} \left| \frac{f(e^{i2\pi\phi}) - p_{m,1}(e^{i2\pi\phi})}{e^{i2\pi\phi(n+1)}} \right| d\phi \\ &< n! \int_{-1/2}^{1/2} 1/2^m d\phi \\ &= \frac{n!}{2^m}, \end{aligned}$$

which proves the assertion made above. Hence, $(a_n)_{n \in \mathbb{N}}$ is a computable sequence of computable numbers, making f and element of \mathcal{CE}_π . \square

Corollary 1. *If f satisfies $f \in \mathcal{CE}_\pi$, then f is Markov computable.*

In other words, if f satisfies $f \in \mathcal{CE}_\pi$, there exists a Turing machine TM_f that computes the mapping $x \mapsto f(x)$ for $x \in \mathbb{R}_c$. Hence, for all $x \in \mathbb{R}_c$, we have $f(x) \in \mathbb{R}_c$.

While $\mathcal{CE}_\pi = \mathcal{WE}_\pi$ holds true in set-theoretic terms, both \mathcal{CE}_π and \mathcal{WE}_π have and individual structure in the sense of computability, which we will discuss in the following.

From the proof of Theorem 1, we conclude that there exists a Turing machine TM which, given any description $((p_{\mathbf{s}})_{\mathbf{s} \in \mathbb{N}^2}, \xi') \rightrightarrows f$, computes a mapping

$$((p_{\mathbf{s}})_{\mathbf{s} \in \mathbb{N}^2}, \xi') \mapsto (a_n)_{n \in \mathbb{N}},$$

such that $(a_n)_{n \in \mathbb{N}} \rightrightarrows f$ holds true.

The remainder of this section is dedicated to proving that the converse is not satisfied. That is, there does not exist a Turing machine that transforms any description $(a_n)_{n \in \mathbb{N}} \rightrightarrows f$ into a corresponding description $((p_{\mathbf{s}})_{\mathbf{s} \in \mathbb{N}^2}, \xi') \rightrightarrows f$. Note that since f is bandlimited, there exists a number $L \in \mathbb{N}$ such that

$$\sqrt[n]{|a_n|} \leq L$$

holds true for all $n \in \mathbb{N}$. However, no such L can be determined algorithmically from the sequence $(a_n)_{n \in \mathbb{N}}$. This observation will play a crucial role in the subsequent lemma.

Lemma 5. *There exists a sequence $(f_m)_{m \in \mathbb{N}}$ of signals in \mathcal{E}_π that simultaneously satisfies the following:*

- (a) *For all $m \in \mathbb{N}$, we have $f_m \in \mathcal{CE}_\pi$ (and, consequently, $f_m \in \mathcal{WE}_\pi$ as well).*
- (b) *There exists a computable double sequence $(a_{n,m})_{n,m \in \mathbb{N}}$ of computable numbers such that $(a_{n,m})_{n \in \mathbb{N}} \rightrightarrows f_m$ in the sense of Definition 14 holds true for all $m \in \mathbb{N}$.*
- (c) *There does not exist a pair $((p_{m,\mathbf{s}})_{m \in \mathbb{N}, \mathbf{s} \in \mathbb{N}^2}, \xi'(m, \cdot, \cdot))$ such that $((p_{m,\mathbf{s}})_{\mathbf{s} \in \mathbb{N}^2}, \xi'(m, \cdot, \cdot)) \rightrightarrows f_m$ in the sense of Definition 15 holds true for all $m \in \mathbb{N}$.*

Proof. Let $A \subset \mathbb{N}$ be a non-recursive set such that there exists $g \in \mathcal{C}_1^*$ that satisfies $D(g) = A$, i.e., A is recursively enumerable. Fix any $n \in \mathbb{N}$ such that $D(\varphi_n) = A$ holds true and consider the recursive function $h : \mathbb{N}^2 \rightarrow \mathbb{N}$ that satisfies

$$h(m, k) = \begin{cases} \min\{l : \Phi(n, m, l) = 1\}, & \text{if } \Phi(n, m, k) = 1, \\ k, & \text{otherwise.} \end{cases}$$

Using this function, define the computable triple sequence $(r_{\mathbf{m}})_{\mathbf{m} \in \mathbb{N}^3}$ of rational numbers by

$$r_{\mathbf{m}} := \begin{cases} 2^{-h(m_2, m_3)}, & \text{if } 0 \leq m_1 \leq 2^{h(m_2, m_3)} - 1, \\ 0, & \text{otherwise,} \end{cases}$$

for all $\mathbf{m} \in \mathbb{N}^3$. Next, we introduce the sequence $(a_{\mathbf{m}})_{\mathbf{m} \in \mathbb{N}^2}$ of real numbers defined by

$$a_{m_1, m_2} := \begin{cases} 2^{-k}, & \text{if } \sup_{m_3 \in \mathbb{N}} h(m_2, m_3) = k \\ & \text{and } 0 \leq m_1 \leq 2^k - 1, \\ 0, & \text{otherwise,} \end{cases}$$

which we will show to be the effective limit of $(r_{\mathbf{m}})_{\mathbf{m} \in \mathbb{N}^3}$ in m_3 for all $m_1, m_2 \in \mathbb{N}$. In other words, we want to show that $(a_{\mathbf{m}})_{\mathbf{m} \in \mathbb{N}^2}$ is a computable double sequence of computable numbers such that for some recursive function $\xi : \mathbb{N}^3 \rightarrow \mathbb{N}$, the pair $((r_{\mathbf{m}})_{\mathbf{m} \in \mathbb{N}^3}, \xi)$ is a standard description of $(a_{\mathbf{m}})_{\mathbf{m} \in \mathbb{N}^2}$. The fact that

$$\lim_{m_3 \rightarrow \infty} r_{m_1, m_2, m_3} = a_{m_1, m_2}$$

holds true for all $m_3 \in \mathbb{N}$ follows directly from the definitions of $(r_{\mathbf{m}})_{\mathbf{m} \in \mathbb{N}^3}$ and $(a_{\mathbf{m}})_{\mathbf{m} \in \mathbb{N}^2}$. It thus only remains to show that the convergence is effective for all $m_1, m_2 \in \mathbb{N}$. In particular, we want to show that the function $\xi : (m_1, m_2, M) \mapsto M + 1$ satisfies

$$\left| \left(\lim_{m_3 \rightarrow \infty} r_{m_1, m_2, m_3} \right) - r_{m_1, m_2, m_3} \right| < \frac{1}{2^M} \quad (14)$$

for all $m_1, m_2, m_3 \in \mathbb{N}$ that satisfy $m_3 \geq \xi(m_1, m_2, M)$. Consider the following case distinction:

1. Fix $m_1, m_2 \in \mathbb{N}$ and assume that there exists $k \in \mathbb{N}$ such that $\sup_{m_3 \in \mathbb{N}} h(m_2, m_3) = k$ and $2^k - 1 < m_1$ hold true. Then, $r_{m_1, m_2, m_3} = 0$ holds true for all $m_3 \in \mathbb{N}$. Hence, (14) is satisfied trivially.
2. Fix $m_1, m_2 \in \mathbb{N}$ and assume that there exists $k \in \mathbb{N}$ such that $\sup_{m_3 \in \mathbb{N}} h(m_2, m_3) = k$ and $2^k - 1 \geq m_1$ hold true. Then, for all $m_3 \in \mathbb{N}_+$

that satisfy $m_3 < \log_2(m_1 + 1)$, we have

$$|a_{m_1, m_2} - r_{m_1, m_2, m_3}| = |0 - 2^{-k}| \leq 2^{-(m_3-1)}.$$

For all $m_3 \in \mathbb{N}_+$ that satisfy $\log_2(m_1 + 1) \leq m_3 \leq k$, we have

$$|a_{m_1, m_2} - r_{m_1, m_2, m_3}| = |2^{-m_3} - 2^{-k}| < 2^{-(m_3-1)}.$$

For all $m_3 \in \mathbb{N}_+$ that satisfy $k < m_3$, we have

$$|a_{m_1, m_2} - r_{m_1, m_2, m_3}| = |2^{-k} - 2^{-k}| < 2^{-(m_3-1)}.$$

Hence, (14) is satisfied for all $m_3 \in \mathbb{N}_+$.

3. Fix $m_1, m_2 \in \mathbb{N}$ and assume that there does not exist $k \in \mathbb{N}$ such that $\sup_{m_3 \in \mathbb{N}} h(m_2, m_3) = k$ holds true. Then, for all $m_3 \in \mathbb{N}_+$ that satisfy $m_3 < \log_2(m_1 + 1)$, we have

$$|a_{m_1, m_2} - r_{m_1, m_2, m_3}| = |0 - 0| < 2^{-(m_3-1)}.$$

For all $m_3 \in \mathbb{N}_+$ that satisfy $\log_2(m_1 + 1) \leq m_3$, we have

$$|a_{m_1, m_2} - r_{m_1, m_2, m_3}| = |2^{-m_3} - 0| < 2^{-(m_3-1)}.$$

Hence, (14) is satisfied for all $m_3 \in \mathbb{N}_+$.

We conclude that (14) is satisfied for all $m_1, m_2, m_3 \in \mathbb{N}$ that satisfy $m_3 \geq M = \xi(m_1, m_2, M)$. Hence, the pair $((r_{\mathbf{m}})_{\mathbf{m} \in \mathbb{N}^3}, \xi)$ is a standard description of the computable double sequence $(a_{\mathbf{m}})_{\mathbf{m} \in \mathbb{N}^2}$ of computable numbers. Consider now the sequence $(a'_{\mathbf{m}})_{\mathbf{m} \in \mathbb{N}^2}$ defined by

$$a'_{m_1, m_2} := m_1! \cdot a_{m_1, m_2}$$

for all $m_1, m_2 \in \mathbb{N}$. Since $(a_{\mathbf{m}})_{\mathbf{m} \in \mathbb{N}^2}$ is a computable double sequence of computable numbers, so is $(a'_{\mathbf{m}})_{\mathbf{m} \in \mathbb{N}^2}$. Hence, the sequence $(f_m)_{m \in \mathbb{N}}$ defined by

$$f_m(z) := \sum_{n=0}^{\infty} \frac{a'_{n, m}}{n!} z^n = \sum_{n=0}^{\infty} a_{n, m} z^n, \quad z \in \mathbb{C},$$

is a computable sequence of entire functions in the sense of Definition 14, i.e., $(a'_m)_{m \in \mathbb{N}^2} \rightrightarrows (f_m)_{m \in \mathbb{N}}$. Furthermore, we have $B(f_m) = 0$ for all $m \in \mathbb{N}$. Thus, $(f_m)_{m \in \mathbb{N}}$ satisfies $f_m \in \mathcal{CE}_\pi$ for all $m \in \mathbb{N}$. Defining $\sup_{l \in \mathbb{N}} h(m, l) =: k(m)$, we have

$$f_m(z) = \sum_{n=0}^{2^{k(m)}-1} \frac{1}{2^{k(m)}} z^n, \quad z \in \mathbb{C} \quad (15)$$

for all $m \in \mathbb{N}$ that satisfy $m \in A$, while for all $m \in \mathbb{N}$ that satisfy $m \notin A$, we have

$$f_m(z) = 0, \quad z \in \mathbb{C}.$$

In particular, as follows from (15) by direct calculation, we obtain that

$$f_m(1) = \mathbb{1}_A(m) \quad (16)$$

holds true for all $m \in \mathbb{N}$. We now show that $(f_m)_{m \in \mathbb{N}}$ is not a computable sequence in the sense of Definition 15. That is, there does not exist a pair $((p_{m,s})_{m \in \mathbb{N}, s \in \mathbb{N}^2}, \xi'(m, \cdot, \cdot))$ consisting of a computable triple sequence $(p_{m,s})_{m \in \mathbb{N}, s \in \mathbb{N}^2}$ of rational polynomials and a recursive function $\xi : \mathbb{N}^3 \rightarrow \mathbb{N}$ such that $((p_{m,s})_{m \in \mathbb{N}, s \in \mathbb{N}^2}, \xi'(m, \cdot, \cdot)) \rightrightarrows (f_m)_{m \in \mathbb{N}}$ holds true. We show this claim by contradiction: assume there exists a pair $((p_{m,s})_{m \in \mathbb{N}, s \in \mathbb{N}^2}, \xi'(m, \cdot, \cdot))$ such that $((p_{m,s})_{m \in \mathbb{N}, s \in \mathbb{N}^2}, \xi'(m, \cdot, \cdot)) \rightrightarrows (f_m)_{m \in \mathbb{N}}$ holds true. Since $(p_{m,s})_{m \in \mathbb{N}, s \in \mathbb{N}^2}$ is a computable sequence of polynomials, there exists a Turing machine TM that computes the mapping

$$m \mapsto p_{m, \xi'(m, 1/2, 1)}(1).$$

Then, for all $m \in \mathbb{N}$, this Turing machine satisfies

$$|f_m(1) - \text{TM}(m)| < \frac{1}{2}. \quad (17)$$

Since TM furthermore satisfies $\text{TM}(m) \in \mathbb{Q}$ for all $m \in \mathbb{N}$, there exists a recursive function $g : \mathbb{N} \rightarrow \{0, 1\}$ that satisfies

$$g(m) = \begin{cases} 1, & \text{if } 1/2 < \text{TM}(m), \\ 0, & \text{otherwise.} \end{cases}$$

Incorporating (16) and (17), we obtain $g = \mathbb{1}_A$, which, since g is a recursive function, is a contradiction to the non-recursivity of A . \square

Theorem 2. *There does not exist a Turing machine that computes a mapping $(a_n)_{n \in \mathbb{N}} \mapsto ((p_m)_{m \in \mathbb{N}^2}, \xi)$, such that $((p_m)_{m \in \mathbb{N}^2}, \xi) \rightrightarrows f$ holds true for every signal $f \in \mathcal{CE}_\pi$ and every description $(a_n)_{n \in \mathbb{N}} \rightrightarrows f$.*

Proof. The claim is a direct consequence of Lemma 5 by contradiction: assume there exists a Turing machine TM that computes a mapping $(a_n)_{n \in \mathbb{N}} \mapsto ((p_m)_{m \in \mathbb{N}^2}, \xi)$ for $(a_n)_{n \in \mathbb{N}} \rightrightarrows f \in \mathcal{CE}_\pi$, such that $((p_m)_{m \in \mathbb{N}^2}, \xi) \rightrightarrows f$ holds true. Consider any pair $((f_m)_{m \in \mathbb{N}}, (a_{n,m})_{n,m \in \mathbb{N}})$ consisting of a sequence $(f_m)_{m \in \mathbb{N}}$ of signals in \mathcal{E}_π and a computable double sequence $(a_{n,m})_{n,m \in \mathbb{N}}$ of computable numbers such that $f_m \rightrightarrows (a_{n,m})_{n \in \mathbb{N}}$ holds true for all $m \in \mathbb{N}$. Then, setting

$$((p_{m,s})_{s \in \mathbb{N}^2}, \xi'(m, \cdot, \cdot)) := \text{TM}((a_{n,m})_{n \in \mathbb{N}})$$

for all $m \in \mathbb{N}$ yields a pair $((p_{m,s})_{m \in \mathbb{N}, s \in \mathbb{N}^2}, \xi'(m, \cdot, \cdot))$ such that $((p_{m,s})_{s \in \mathbb{N}^2}, \xi'(m, \cdot, \cdot)) \rightrightarrows f_m$ holds true for all $m \in \mathbb{N}$. Hence, if the Turing machine TM existed, any sequence $(f_m)_{m \in \mathbb{N}}$ of signals in \mathcal{E}_π that satisfies conditions (a) and (b) of Lemma 5 would necessarily violate condition (c). \square

In Section 8 and Section 9, we will analyze whether $B(f)$ can be computed for $f \in \mathcal{CE}_\pi$, or, if that is not possible, whether we can at least algorithmically determine upper and lower bounds for $B(f)$. In Section 10, we will characterize the bandwidth of computable bandlimited signals with respect to the Zheng-Weihsrauch hierarchy of real numbers. Furthermore, in Section 11, we will relate the bandwidth of computable bandlimited signals with specific properties to a number of common computational oracles.

8. Computability of Upper and Lower Bounds

Let $f \in \mathcal{CE}_\pi$. Then we have $B(f) \leq \pi$. As indicated in the introduction, it was posed as an open question in [16] whether it is possible to compute meaningful bounds for the number $B(f)$, given that f satisfies $f \in \mathcal{CE}_\pi$. Hence, we study the following questions:

1. Let $\overline{\text{TM}}_{\text{BW}}$ be a Turing machine with $\overline{\text{TM}}_{\text{BW}}: \mathcal{CE}_\pi \rightarrow \mathbb{R}_c^{+0}$ and $\overline{\text{TM}}_{\text{BW}}(f) \geq B(f)$ for all $f \in \mathcal{CE}_\pi$. What is the output behavior of this Turing machine?
2. Let $\underline{\text{TM}}_{\text{BW}}$ be a Turing machine with $\underline{\text{TM}}_{\text{BW}}: \mathcal{CE}_\pi \rightarrow \mathbb{R}_c^{+0}$ and $\underline{\text{TM}}_{\text{BW}}(f) \leq B(f)$ for all $f \in \mathcal{CE}_\pi$. What is the output behavior of this Turing machine?

Question 1 is concerned with finding Turing machines that compute an upper bound for the actual bandwidth of the input signal. Similarly, Question 2 is concerned with finding Turing machines that compute a lower bound for the actual bandwidth of the input signal.

In [17] we have constructed a real-valued signal $f_1 \in \mathcal{CB}_\pi^1$ such that $B(f_1) \notin \mathbb{R}_c$, i.e. the actual bandwidth $B(f_1)$ is not computable. However, for some practical applications, it might be sufficient to know a non-trivial lower or upper bound for the number $B(f_1)$. In light of this, Questions 1 and 2 are of high importance. We show in the following that Turing machines that satisfy the above requirements and yield non-trivial bounds on the number $B(f)$ for at least some feasible signals $f \in \mathcal{CE}_\pi$ cannot exist, proving the corresponding conjecture from [16] correct.

Theorem 3. *Let $\overline{\text{TM}}_{\text{BW}}$ be an arbitrary Turing machine such that for all $f \in \mathcal{CE}_\pi$ and all $(a_n)_{n \in \mathbb{N}} \rightrightarrows f$, $\overline{\text{TM}}_{\text{BW}}$ computes a mapping*

$$(a_n)_{n \in \mathbb{N}} \mapsto ((r_n)_{n \in \mathbb{N}}, \xi) \rightrightarrows \overline{\text{TM}}_{\text{BW}}(f) \in \mathbb{R}_c$$

that satisfies $\overline{\text{TM}}_{\text{BW}}(f) \geq B(f)$ for all $f \in \mathcal{CE}_\pi$. Then we have $\overline{\text{TM}}_{\text{BW}}(f) = \pi$ for all $f \in \mathcal{CE}_\pi$.

Theorem 3 provides an answer to Question 1. Any Turing machine that computes an upper bound of $B(f)$ for all signals $f \in \mathcal{CE}_\pi$ is necessarily trivial in the sense that it returns the value π for all signals $f \in \mathcal{CE}_\pi$.

Proof of Theorem 3. We do a proof by contradiction and assume that there exists a Turing machine $\overline{\text{TM}}_{\text{BW}}$ with properties as in the theorem and a signal

$f \in \mathcal{CE}_\pi$ such that $\overline{\text{TM}}_{\text{BW}}(f) < \pi$. Since $\overline{\text{TM}}_{\text{BW}}(f) \geq B(f)$, we also have $B(f) < \pi$. For $\lambda \in [0, 1] \cap \mathbb{R}_c$, let

$$f_\lambda(z) = (1 - \lambda)f(z) + \lambda \frac{\sin(\pi z)}{\pi z}.$$

We have $B(f_\lambda) = \pi$ for all $\lambda \in (0, 1] \cap \mathbb{R}_c$ as well as $f_\lambda \in \mathcal{CE}_\pi$ for all $\lambda \in [0, 1] \cap \mathbb{R}_c$, the latter being due to the linear structure of \mathcal{CE}_π . Furthermore, observe that the mapping $\lambda \mapsto f_\lambda(z)$ is computable, i.e., there exists a Turing machine which, for all $\lambda \in [0, 1] \cap \mathbb{R}_c$ and all $((r'_n)_{n \in \mathbb{N}}, \xi') \rightrightarrows \lambda$ computes a mapping $((r'_n)_{n \in \mathbb{N}}, \xi') \mapsto (a_n)_{n \in \mathbb{N}}$, such that $(a_n)_{n \in \mathbb{N}} \rightrightarrows f_\lambda$ is satisfied. Hence, by concatenation, we obtain a computable mapping

$$((r'_n)_{n \in \mathbb{N}}, \xi') \mapsto (a_n)_{n \in \mathbb{N}} \mapsto ((r_n)_{n \in \mathbb{N}}, \xi),$$

where $((r_n)_{n \in \mathbb{N}}, \xi)$ is a standard description of $\overline{\text{TM}}_{\text{BW}}(f) \in \mathbb{R}_c$, such that

$$\lim_{n \rightarrow \infty} r'_n = 0 \Leftrightarrow \lim_{n \rightarrow \infty} r_n < \pi$$

holds true. Since π satisfies $\pi \in \mathbb{R}_c$, there exists a non-negative, monotonically non-decreasing, computable sequence $(s_n)_{n \in \mathbb{N}}$ of rational numbers that satisfies $\lim_{n \rightarrow \infty} s_n = \pi$. For all $n \in \mathbb{N}$ define

$$\bar{r}_n := \min \{r_{\xi(m)} + 1/2^m : m \leq n\} - s_n.$$

Then, $(\bar{r}_n)_{n \in \mathbb{N}}$ is a monotonically non-increasing, computable sequence of rational numbers that satisfies

$$\lim_{n \rightarrow \infty} \bar{r}_n < \pi \Leftrightarrow \exists n \in \mathbb{N} : \bar{r}_n < 0.$$

Likewise, for $n \in \mathbb{N}$ define

$$\underline{r}_n := \max \{r'_{\xi'(m)} - 1/2^m : m \leq n\}.$$

Then, $(\underline{r}_n)_{n \in \mathbb{N}}$ is a monotonically non-decreasing, computable sequence of rational numbers that satisfies

$$\lim_{n \rightarrow \infty} \underline{r}_n = 0 \Leftrightarrow \forall n \in \mathbb{N} : \underline{r}_n \leq 0.$$

In summary, we observe that both

$$\lambda = 0 \Leftrightarrow \exists n \in \mathbb{N} : \bar{r}_n < 0,$$

$$\lambda > 0 \Leftrightarrow \exists n \in \mathbb{N} : \underline{r}_n > 0$$

are satisfied. Define

$$n_0 := \min\{n \in \mathbb{N} : \bar{r}_n < 0 \vee \underline{r}_n > 0\}.$$

Then, the mapping $((r'_n)_{n \in \mathbb{N}}, \xi') \mapsto n_0$ is recursive and the Turing machine

$$\text{TM}_0(((r'_n)_{n \in \mathbb{N}}, \xi')) := \begin{cases} 1 & \text{if } \underline{r}_{n_0} > 0 \\ 0 & \text{if } \bar{r}_{n_0} < 0 \end{cases}$$

is well-defined. But TM_0 satisfies $\text{TM}_0(((r'_n)_{n \in \mathbb{N}}, \xi')) = 0$ if and only if $((r'_n)_{n \in \mathbb{N}}, \xi') \Rightarrow 0$ holds true, which is a contradiction to [21, Proposition 0, p. 14]. \square

Remark 12. *For the proof of Theorem 3, the linear structure of \mathcal{CE}_π is essential. For all $p \in (1, +\infty)$, the set \mathcal{CB}_π^p exhibits a linear structure in the same manner. Furthermore, observe that f_λ satisfies $f_\lambda \in \mathcal{CB}_\pi^p$ for all $\lambda \in [0, 1]$, provided that f satisfies $f \in \mathcal{CB}_\pi^p$. Indeed, we can always find a suitable $f \in \mathcal{CB}_\pi^p$, such that the proof of Theorem 3 can be carried out for the set \mathcal{CB}_π^p instead of \mathcal{CE}_π . Hence, for all $p \in (1, +\infty)$, a statement analogous to Theorem 3 holds true.*

Theorem 4. *Let $\underline{\text{TM}}_{BW}$ be an arbitrary Turing machine such that for all $f \in \mathcal{CE}_\pi$ and all $(a_n)_{n \in \mathbb{N}} \Rightarrow f$, $\underline{\text{TM}}_{BW}$ computes a mapping*

$$(a_n)_{n \in \mathbb{N}} \mapsto ((r_n)_{n \in \mathbb{N}}, \xi) \Rightarrow \underline{\text{TM}}_{BW}(f) \in \mathbb{R}_c$$

that satisfies $\underline{\text{TM}}_{BW}(f) \leq B(f)$ for all $f \in \mathcal{CE}_\pi$. Then we have $\underline{\text{TM}}_{BW}(f) = 0$ for all $f \in \mathcal{CE}_\pi$.

Since zero is the trivial lower bound for $B(f)$, Theorem 4 shows that for \mathcal{CE}_π , it is not possible to construct a Turing machine that computes a non-trivial lower bound for $B(f)$ for all $f \in \mathcal{CE}_\pi$. Hence, Theorem 4 provides an answer to Question 2

Proof of Theorem 4. Assume that there exists a Turing machine $\underline{\text{TM}}_{\text{BW}}$ with properties as in the theorem and a signal $f \in \mathcal{CE}_\pi$ such that $\underline{\text{TM}}_{\text{BW}}(f) > 0$. Since $\underline{\text{TM}}_{\text{BW}}(f) \leq B(f)$, we also have $B(f) > 0$. Let

$$f(z) = \sum_{n=0}^{\infty} \frac{a_n}{n!} z^n, \quad z \in \mathbb{C},$$

be the power series representation of f . Then the sequence $(a_n)_{n \in \mathbb{N}}$ is a computable sequence of computable numbers, and we have

$$\limsup_{n \rightarrow \infty} \sqrt[n]{|a_n|} \leq \pi.$$

Let

$$a_n(m) = \begin{cases} a_n, & 0 \leq n \leq m, \\ 0, & n > m. \end{cases}$$

The sequence $(a_n(m))_{n \in \mathbb{N}, m \in \mathbb{N}}$ is a computable double sequence of computable numbers.

Let $\mathcal{A} \subsetneq \mathbb{N}$ be a recursively enumerable non-recursive set, and $\phi_{\mathcal{A}}: \mathbb{N} \rightarrow \mathcal{A}$ a recursive enumeration of the elements of \mathcal{A} , where $\phi_{\mathcal{A}}$ is a one-to-one function, i.e., for every element $m \in \mathcal{A}$ there exists exactly one $\hat{k} \in \mathbb{N}$ with $\phi_{\mathcal{A}}(\hat{k}) = m$.

Let

$$a_n(m, k) = \begin{cases} a_n(\hat{k}), & \text{if } m \in \{\phi_{\mathcal{A}}(1), \dots, \phi_{\mathcal{A}}(k)\}, \\ a_n(k), & \text{otherwise.} \end{cases}$$

Then $(a_n(m, k))_{n \in \mathbb{N}, m \in \mathbb{N}, k \in \mathbb{N}}$ is a computable triple sequence of computable numbers. For $n, m \in \mathbb{N}$, we will analyze the behavior of the sequence $(a_n(m, k))_{k \in \mathbb{N}}$ next. For $m \in \mathcal{A}$, we have

$$\lim_{k \rightarrow \infty} a_n(m, k) = a_n(\hat{k}),$$

where $\phi_{\mathcal{A}}(\hat{k}) = m$. For $m \notin \mathcal{A}$, we have

$$\lim_{k \rightarrow \infty} a_n(m, k) = a_n.$$

For $n \in \mathbb{N}$, we set

$$\hat{a}_n(m) = \begin{cases} a_n(\hat{k}), & \text{if } m \in \mathcal{A}, \\ a_n, & \text{if } m \notin \mathcal{A}, \end{cases}$$

where $\phi_{\mathcal{A}}(\hat{k}) = m$.

Next, we show that there exists a recursive function $\xi: \mathbb{N}^3 \rightarrow \mathbb{N}$ such that for all $M \in \mathbb{N}$, $n \in \mathbb{N}$, and $m \in \mathbb{N}$, we have

$$|\hat{a}_n(m) - a_n(m, k)| \leq 2^{-M}$$

for all $k \geq \xi(M, n, m)$.

Let $n \in \mathbb{N}$ and $m \notin \mathcal{A}$ be arbitrary but fixed. Then, for $k \geq n$, we have

$$a_n(m, k) = a_n(k) = a_n = \hat{a}_n(m).$$

It follows that for $M \in \mathbb{N}$ we have

$$|\hat{a}_n(m) - a_n(m, k)| = 0 \leq 2^{-M},$$

for all $k \geq \xi(M, m, n) = n$. Let $n \in \mathbb{N}$ and $m \in \mathcal{A}$ be arbitrary but fixed. Then, for $k \geq n$, we have: If $m \in \{\phi_{\mathcal{A}}(1), \dots, \phi_{\mathcal{A}}(k)\}$, i.e., if $m = \phi_{\mathcal{A}}(\hat{k})$ for $\hat{k} \leq k$, then we have

$$a_n(m, k) = a_n(\hat{k}) = \hat{a}_n(m),$$

and it follows that

$$|\hat{a}_n(m) - a_n(m, k)| = 0 \leq 2^{-M}$$

for all $M \in \mathbb{N}$. If $m \notin \{\phi_{\mathcal{A}}(1), \dots, \phi_{\mathcal{A}}(k)\}$, i.e., if $\hat{k} > k$, then we have

$$a_n(m, k) = a_n(k) = a_n = a_n(\hat{k}) = \hat{a}_n(m),$$

where we used that $\hat{k} > k \geq n$ in the second to last equality. Hence, for $M \in \mathbb{N}$ we have

$$|\hat{a}_n(m) - a_n(m, k)| = 0 \leq 2^{-M}$$

for all $k \geq \xi(M, m, n) = n$.

Combining all partial results, we see that, for all $M \in \mathbb{N}$, $m \in \mathbb{N}$, $n \in \mathbb{N}$, and $k \geq \xi(M, m, n)$ we have

$$|\hat{a}_n(m) - a_n(m, k)| \leq 2^{-M}.$$

This shows that $(\hat{a}_n(m))_{n \in \mathbb{N}, m \in \mathbb{N}}$ is a computable double sequence of computable numbers.

Let

$$g_m(z) = \sum_{n=0}^{\infty} \frac{\hat{a}_n(m)}{n!} z^n, \quad z \in \mathbb{C}.$$

Since

$$\limsup_{n \rightarrow \infty} \sqrt[n]{|\hat{a}_n(m)|} \leq \limsup_{n \rightarrow \infty} \sqrt[n]{|a_n|} \leq \pi,$$

we see that $(g_m)_{m \in \mathbb{N}}$ is a computable sequence of computable signals in \mathcal{CE}_π .

We now construct a Turing machine TM that consists of two Turing machines. The first Turing machine, which we denote by $\text{TM}_1: \mathbb{N} \rightarrow \{\text{TM stops, TM runs forever}\}$ stops for an input $m \in \mathbb{N}$ if and only if $m \in \mathcal{A}$. The second Turing machine $\text{TM}_2: \mathbb{N} \rightarrow \{\text{TM stops, TM runs forever}\}$ works as follows. For an input $m \in \mathbb{N}$, it first computes $g_m \in \mathcal{CE}_\pi$. Then it computes $\underline{\text{TM}}_{\text{BW}}(g_m)$. We use the fact that there exists a Turing machine $\text{TM}_>$ that, given an input $\lambda \in \mathbb{R}_c$, stops if and only if $\lambda > 0$ [21, Proposition 0, p. 14]. Our Turing machine TM_2 now starts $\text{TM}_>$ with $\underline{\text{TM}}_{\text{BW}}(g_m)$ as an input. Hence, TM_2 will stop if and only if $\underline{\text{TM}}_{\text{BW}}(g_m) > 0$. For $m \in \mathcal{A}$, g_m is a polynomial, for which we have $B(g_m) = 0$, according to (4). Hence, for $m \in \mathcal{A}$, we have $\underline{\text{TM}}_{\text{BW}}(g_m) = 0$. Consequently, TM_2 stops if and only if $m \notin \mathcal{A}$. Thus, TM, which runs TM_1 and TM_2 in parallel, is a Turing machine that, for any input $m \in \mathbb{N}$, can decide whether $m \in \mathcal{A}$ or $m \notin \mathcal{A}$. This implies that the set \mathcal{A} is recursive, which is a contradiction. Hence, $\underline{\text{TM}}_{\text{BW}}$ cannot exist. \square

9. Semi-Decidability

On a related note, one may ask whether, for a given value $\sigma \in (0, \pi) \cap \mathbb{R}_c$, it is possible to algorithmically detect if a signal $f \in \mathcal{CE}_\pi$ satisfies the

bandwidth condition $B(f) > \sigma$. Ideally, we would like to have a Turing machine $\text{TM}_\sigma: \mathcal{CE}_\pi \rightarrow \{\text{true}, \text{false}\}$ that satisfies $\text{TM}_\sigma(f) = \text{true}$ if and only if $B(f) > \sigma$ holds true. However, such a Turing machine cannot exist: relabeling the output-values of TM_σ by " σ " for "true" and "0" for "false", we immediately obtain a Turing machine that computes a non-trivial lower bound on $B(f)$ for some $f \in \mathcal{CE}_\pi$, thereby contradicting Theorem 4.

As discussed in the introduction, another open question was posed in [16], concerning a weakening of the above problem: Does there exist an algorithm $\text{TM}_{\text{BW}>\sigma}$ that, upon being presented with an input $f \in \mathcal{CE}_\pi$, halts its computation in a finite number of steps if $B(f) > \sigma$ holds true, and computes forever otherwise?

To answer this question we introduce the concept of semi-decidability. We call a set $\mathcal{M} \subset \mathcal{CE}_\pi$ semi-decidable if there exists a Turing machine $\text{TM}: \mathcal{CE}_\pi \rightarrow \{\text{TM stops}, \text{TM runs forever}\}$ that, given an input $f \in \mathcal{CE}_\pi$, stops if and only if $f \in \mathcal{M}$. Consequently, the question about the existence of $\text{TM}_{\text{BW}>\sigma}$ for some value $\sigma \in (0, \pi)$ is equivalent to the question about the semi-decidability of the set $\{f \in \mathcal{CE}_\pi: B(f) > \sigma\}$.

In [17], this question was studied for signals in \mathcal{B}_π^1 , and it was proved that the set $\{f \in \mathcal{B}_\pi^1: B(f) > \sigma\}$ is semi-decidable. Next, we show that $\{f \in \mathcal{CE}_\pi: B(f) > \sigma\}$ is not semi-decidable.

Theorem 5. *For all $\sigma \in (0, \pi) \cap \mathbb{R}_c$, the set*

$$\{f \in \mathcal{CE}_\pi: B(f) > \sigma\} \tag{18}$$

is not semi-decidable.

Proof. We do a proof by contradiction and assume that there exists $\sigma \in (0, \pi) \cap \mathbb{R}_c$ and a Turing machine $\text{TM}_{\text{BW}>\sigma}$ that accepts exactly the set $\mathcal{CE}_\pi^>(\sigma)$. Let $f_* \in \mathcal{CE}_\pi$ with $B(f_*) > \sigma$ be arbitrary. Analogous to the proof of Theorem 4, we construct a computable sequence $(g_m)_{m \in \mathbb{N}_+}$ of functions in \mathcal{CE}_π that satisfies

$$B(g_m) = \begin{cases} B(f_*), & \text{if } m \in \mathbb{N}_+ \setminus \mathcal{A}, \\ 0, & \text{otherwise,} \end{cases}$$

for some recursively enumerable, non-recursive set $\mathcal{A} \subsetneq \mathbb{N}_+$. Then, the Turing machine $\text{TM}_{\text{BW} > \sigma}$ provides an algorithm that semi-decides the set $\mathbb{N}_+ \setminus \mathcal{A}$, which contradicts the assumption. \square

In the broadest sense, the Theorems 3, 4 and 5 yield a fundamental limit to the capabilities of computer-aided design. Algorithms for automated system design commonly feature an *exit flag* functionality, that, upon termination of the algorithm, indicates whether the computation was successful or not.

In our context, a hypothetical computer-aided design tool might want to sample a signal with a given sampling rate σ/π . It would return the sampled signal together with a boolean exit flag that signals whether we had a problematic input signal, having a bandwidth that exceeds σ . However, as Theorem 5 shows, such an exit flag functionality cannot be implemented for signals in \mathcal{CE}_π .

10. Structural Properties of the Bandwidth of Computable Signals

The bandwidth of signals in \mathcal{CB}_π^1 is not always a computable number, as was shown in [16]. Hence, the question arises whether it is possible to characterize the degree of uncomputability of the elements of $B[\mathcal{CB}_\pi^p]$ for $1 \leq p \leq +\infty$ and, more generally, $B[\mathcal{CE}_\pi]$. That is, we want to classify the elements of $B[\mathcal{CB}_\pi^p]$ for $1 \leq p \leq +\infty$ and $B[\mathcal{CE}_\pi]$ with respect to the arithmetical hierarchy of real numbers.

The majority of mathematical models in science and engineering are, in the final analysis, supposed to yield quantitative results. Despite not having received much attention in the past, a classification of such models regarding their arithmetic complexity is highly desirable. Not only does it yield essential insights concerning the mathematical structure of the model itself, but also answers whether, given a certain type of computation machine, we can expect the model to produce meaningful quantitative results in the first place. To the best of our knowledge, the present work is the first one to yield such a characterization for a problem from engineering.

As mentioned above, the bandwidth of signals in \mathcal{CB}_π^1 is generally not a computable number. The proof presented in [16] relies on the construction of a signal $f \in \mathcal{CB}_\pi^1$ which exhibits this property. Implicitly, the following assertion was proven as well (c.f. [16, Theorem 2, p. 7]):

Lemma 6. *Let $(r_m)_{m \in \mathbb{N}}$ be a bounded, monotonically non-decreasing, computable sequence of positive rational numbers. Then, there exists a signal $f \in \mathcal{CB}_\sigma^1$ that satisfies*

$$\sigma := B(f) = \lim_{m \rightarrow \infty} r_m. \quad (19)$$

Proof. Consider the indicator function $\mathbf{1}_{[-1/2, 1/2]} : \mathbb{R} \mapsto \{0, 1\}$ of the real interval $[-1/2, 1/2]$ in the frequency domain and set $\hat{f}^a := \mathbf{1}_{[-1/2, 1/2]}$. Then, we have

$$f^a(t) = \frac{\sin(t/2)}{\pi t}.$$

Consider furthermore the convolution $\hat{f}^b := \hat{f}^a * \hat{f}^a$ of \hat{f}^a with itself. We have

$$\hat{f}^b(\omega) = \begin{cases} 1 - |\omega|, & \text{if } |\omega| \leq 1, \\ 0, & \text{otherwise,} \end{cases}$$

$$f^b(t) = \left(\frac{\sin(t/2)}{\pi t} \right)^2.$$

Hence, f^b satisfies $f^b(t) \geq 0$ for all $t \in \mathbb{R}$ and we have

$$\|f^b\|_{L^1} = \int_{-\infty}^{+\infty} |f^b(t)| dt = \int_{-\infty}^{+\infty} f^b(t) dt = \hat{f}^b(0) = 1.$$

We conclude that f^b satisfies $f^b \in \mathcal{B}_1^1$. It can be shown that f^b is also computable in \mathcal{B}_1^1 [16, Appendix E, p. 16], i.e., $f^b \in \mathcal{CB}_1^1$. We will now consider signals f_m^c that emerge from f^b by shortening and shifting in the frequency domain. For $m \in \mathbb{N}$, define

$$\hat{f}_m^c(\omega) := \hat{f}^b \left(\frac{2}{r_{m+1} - r_m} \left(\omega - \frac{r_{m+1} + r_m}{2} \right) \right),$$

$$f_m^c(t) = e^{i \frac{r_{m+1} + r_m}{2} t} \left(\frac{r_{m+1} - r_m}{2} f^b \left(\frac{(r_{m+1} - r_m)t}{2} \right) \right),$$

whenever $r_m < r_{m+1}$ holds true. If $r_m = r_{m+1}$ holds true instead, set $\hat{f}_m^c = f_m^c = \mathbf{0}$ (here, $\mathbf{0}$ refers to the trivial signal in L_1). We observe that f_m^c satisfies

$$\|f_m^c\|_{L^1} = \|f_m^b\|_{L^1} = 1$$

for all $m \in \mathbb{N}$ that satisfy $r_m < r_{m+1}$ and \hat{f}_m^c satisfies

$$\text{esssup } \hat{f}_m^c = [r_m, r_{m+1}]$$

for all $m \in \mathbb{N}$ that satisfy $r_m < r_{m+1}$. Hence, since r_m is a rational number for all $m \in \mathbb{N}$, we have $f_m^c \in \mathcal{CB}_{r_{m+1}}^1$ and $B(f_m^c) = r_{m+1}$ for all $m \in \mathbb{N}$ that satisfy $r_m < r_{m+1}$. We now define

$$f_k(t) := \sum_{m=1}^k \frac{1}{m^2} \cdot f_m^c(t),$$

$$f(t) := \lim_{k \rightarrow \infty} f_k(t),$$

and, in the following, show that f is indeed a signal $f \in \mathcal{CB}_\sigma^1$ that satisfies (19).

Define $\mathcal{M} := \{m \in \mathbb{N} : r_m < r_{m+1}\}$. First, observe that

$$\|f\|_{L^1} \leq \sum_{m=1}^{\infty} \left\| \frac{1}{m^2} \cdot f_m^c \right\|_{L^1} = \sum_{m \in \mathcal{M}} \frac{1}{m^2} \leq \frac{\pi^2}{6}$$

is satisfied. Since $\hat{f}_m^c(\omega) \geq 0$ holds true for all $\omega \in \mathbb{R}$, we have

$$\begin{aligned} B(f) &= \sup \bigcup_{m \in \mathcal{M}} \text{esssup } \hat{f}_m^c \\ &= \sup \bigcup_{m \in \mathcal{M}} [r_m, r_{m+1}] \\ &= \lim_{m \rightarrow \infty} r_{m+1} \end{aligned}$$

It remains to show that f is computable in \mathcal{CB}_σ^1 . Since f_m^c is computable in \mathcal{CB}_σ^1 for all $m \in \mathbb{N}$, so is f_k for all $k \in \mathbb{N}$. Hence, it is sufficient to prove that f_k converges effectively towards f for $k \rightarrow \infty$. We have

$$\begin{aligned} \|f - f_k\|_{L^1} &\leq \sum_{m=k+1}^{\infty} \frac{1}{m^2} \cdot \|f_m^c\|_{L^1} \\ &\leq \sum_{m=k+1}^{\infty} \frac{1}{m^2} \end{aligned}$$

$$= \frac{\pi^2}{6} - \sum_{m=1}^k \frac{1}{m^2},$$

which is a computable sequence of computable numbers in k that converges monotonically decreasingly towards 0. Hence, the convergence of f_k towards f is effective and we have $f \in \mathcal{CB}_\sigma^1$ with f satisfying (19). \square

For the full characterization of $B[\mathcal{CB}_\pi^p]$ for $1 \leq p \leq +\infty$ we first need another lemma. Again, the statement was proven in a related form in [16, Theorem 5, p. 10]: for all $\sigma \in (0, \pi) \cap \mathbb{R}_c$ and all $1 \leq p \leq +\infty$, the set $\mathcal{CB}_\pi^p \setminus \mathcal{CB}_\sigma^p$ is semi-decidable with respect to \mathcal{CB}_π^p . That is, for all $\sigma \in (0, \pi) \cap \mathbb{R}_c$ and all $1 \leq p \leq +\infty$, there exists a Turing machine TM that computes a partial mapping $f \mapsto m \in \mathbb{N}$, where $D(\text{TM})$ equals $\mathcal{CB}_\pi^p \setminus \mathcal{CB}_\sigma^p$. Such a Turing machine exists for all $\sigma \in (0, \pi) \cap \mathbb{R}_c$ and all $1 \leq p \leq +\infty$, and takes a description of $f \in \mathcal{CB}_\pi^p$. In order to establish the hierarchical characterization of $B[\mathcal{CB}_\pi^p]$, we require the analogous statement for input $\sigma \in (0, \pi) \cap \mathbb{R}_c$ instead of $f \in \mathcal{CB}_\pi^p$. That is, we want to show that for all $f \in \mathcal{CB}_\pi^p$ and all $1 \leq p \leq +\infty$, there exists a Turing machine that semi-decides the set $\{\sigma \in (0, \pi) \cap \mathbb{R}_c : \sigma < B(f)\}$ for input $\sigma \in (0, \pi) \cap \mathbb{R}_c$.

Lemma 7. *For all $f \in \mathcal{CB}_\pi^p$ and all $1 \leq p \leq +\infty$, the set $\{\sigma \in (0, \pi) \cap \mathbb{R}_c : \sigma < B(f)\}$ is semi-decidable with respect to the set $(0, \pi) \cap \mathbb{R}_c$.*

Proof. If f satisfies $f \in \mathcal{CB}_\pi^p$ for some $1 < p \leq +\infty$, then there exists a signal $f' \in \mathcal{CB}_\pi^1$ that satisfies $B(f) = B(f')$ [16, Proof of Theorem 5, p. 10]. Hence, without loss of generality, we can assume that $f \in \mathcal{CB}_\pi^1$ holds true. Since f satisfies $f \in \mathcal{CB}_\pi^1$, the Fourier transform \hat{f} of f is Turing computable on the interval $(-\pi, \pi]$. In particular, this implies the existence of a Turing machine TM that computes a mapping

$$\sigma \mapsto ((r_m)_{m \in \mathbb{N}}, \xi)$$

for $\sigma \in (0, \pi) \cap \mathbb{R}_c$, where $((r_m)_{m \in \mathbb{N}}, \xi)$ is a standard description of the number

$$x(\sigma) := \max \{ \hat{f}(\omega) : \omega \in [-\pi, -\sigma) \cup (\sigma, \pi] \} \in \mathbb{R}_c.$$

Then, $x(\sigma)$ satisfies $x(\sigma) > 0$ if and only if $\sigma < B(f)$ holds true. Consider the computable sequence $(r'_m)_{m \in \mathbb{N}}$ of rational numbers that satisfies

$$r'_m := r_m - 2^{-\xi(m)}.$$

We have $\lim_{m \rightarrow \infty} r'_m = x(\sigma)$ as well as $r'_m \leq x(\sigma)$ for all $m \in \mathbb{N}$. Hence, there exists an $m \in \mathbb{N}$ such that $r'_m > 0$ holds true if and only if σ satisfies $\sigma < B(f)$. Finally, we conclude the existence a Turing machine TM' that computes the mapping

$$\sigma \mapsto \min\{m \in \mathbb{N} : r'_m > 0\}$$

for input $\sigma \in (0, \pi) \cap \mathbb{R}_c$. The Turing machine TM' then satisfies $D(\text{TM}') = \{\sigma \in (0, \pi) \cap \mathbb{R}_c : \sigma < B(f)\}$, which is the required property. \square

We will now use the above results to prove the full characterization of $B[\mathcal{CB}_\pi^p]$ for $1 \leq p \leq +\infty$ with respect to the arithmetical hierarchy of real numbers.

Theorem 6.

1. If f satisfies $f \in \mathcal{CB}_\pi^p$ for $1 \leq p \leq +\infty$, then $B(f)$ satisfies $B(f) \in \Sigma_1$.
2. If x satisfies $x \in \Sigma_1 \cap [0, \pi]$ then there exists $f \in \mathcal{CB}_\pi^1$ such that $B(f) = x$ holds true.

Hence, the set $B[\mathcal{CB}_\pi^p]$ coincides with the set $\Sigma_1 \cap [0, \pi]$ for all $1 \leq p \leq +\infty$.

Proof.

1. From Lemma 7, we know that there exists a Turing machine TM that computes a mapping $\sigma \mapsto m \in \mathbb{N}$ for input $\sigma \in (0, \pi) \cap \mathbb{R}_c$, such that

$$D(\text{TM}) = \{\sigma \in (0, \pi) \cap \mathbb{R}_c : \sigma < B(f)\}$$

holds true. In the following, we consider for all $\mathbf{m} \in \mathbb{N}^3$ the rational number

$$q(\mathbf{m}) := (-1)^{m_3} \frac{m_1}{m_2}.$$

Since \mathbb{Q} is a subset of \mathbb{R}_c , Lemma 7 implies the existence of a recursive function $g'' : \mathbb{N}^3 \rightarrow \mathbb{N}$, such that

$$\begin{aligned} D(g'') \cap \left\{ \mathbf{m} \in \mathbb{N}^3 : q(\mathbf{m}) \in (0, \pi) \right\} \\ = \left\{ \mathbf{m} \in \mathbb{N}^3 : q(\mathbf{m}) \in D(\text{TM}) \right\} \end{aligned}$$

holds true. Since both 0 and π are computable numbers, the set $(0, \pi) \cap \mathbb{Q}$ is recursive in the following sense: there exists a total recursive mapping $g' : \mathbb{N} \rightarrow \mathbb{N}$ such that

$$q(\Pi_3 g'(n)) \in (0, \pi)$$

holds true for all $n \in \mathbb{N}$ and for all $\sigma \in (0, \pi) \cap \mathbb{Q}$, there exists an $n \in \mathbb{N}$ such that

$$q(\Pi_3 g'(n)) = \sigma$$

holds true. Define $g(n) := g''(\Pi_3 g'(n))$ for all $n \in \mathbb{N}$. Then, g is a recursive function that satisfies

$$\begin{aligned} D(g) &= \{n \in \mathbb{N} : q(\Pi_3 g'(n)) \in (0, B(f))\}, \\ g[\mathbb{N}] &= (0, B(f)) \cap \mathbb{Q}. \end{aligned}$$

We have $\sup(0, B(f)) \cap \mathbb{Q} = \sup\{q \in \mathbb{Q} : q < B(f)\}$, where $\{q \in \mathbb{Q} : q < B(f)\}$ is the Dedekind cut of $B(f)$. Hence, g satisfies $\sup g[\mathbb{N}] = B(f)$. Now consider $k \in \mathbb{N}$ such that $g = \varphi_k$ and define the set

$$\begin{aligned} \mathcal{Q}(l) := \left\{ q(\Pi_3 g'(\varpi_1(m))) : m \in \{1, \dots, l\}, \right. \\ \left. \Psi(k \circ \Pi_2(m)) = 1 \right\}. \end{aligned}$$

Then, we have $\mathcal{Q}(l) \subseteq \mathcal{Q}(l+1)$ for all $l \in \mathbb{N}$ as well as

$$\bigcup_{l=1}^{\infty} \mathcal{Q}(l) = g[\mathbb{N}] = (0, B(f)) \cap \mathbb{Q},$$

and the monotonically non-decreasing, computable sequence $(r_m)_{m \in \mathbb{N}}$ of rational numbers, defined by

$$r_m := \begin{cases} \max \mathcal{Q}(m), & \text{if } \mathcal{Q}(m) \neq \emptyset, \\ 0, & \text{otherwise,} \end{cases}$$

for all $m \in \mathbb{N}$, satisfies $\sup_{m \in \mathbb{N}} r_m = B(f)$.

2. If x satisfies $x \in \Sigma_1$, then, per definition, there exists a computable sequence $(r'_m)_{m \in \mathbb{N}}$ of rational numbers that satisfies

$$\sup_{m \in \mathbb{N}} (r'_m) = x.$$

Now, for all $m \in \mathbb{N}$, define

$$r_m := \max(\{r_k : k \leq m\} \cup \{0\}).$$

Then, $(r_m)_{m \in \mathbb{N}}$ is a non-negative, computable sequence of rational numbers that converges monotonically non-decreasingly towards x . Hence, the claim follows from Lemma 6.

□

As Theorem 6 shows, the computability properties of $B[\mathcal{CB}_\pi^p]$ do (for $p \geq 1$) not depend on the actual value of p . In other words, the computability behavior of signals in \mathcal{CB}_π^∞ is not getting any more benevolent if the requirements on the signal decay for $t \rightarrow \infty$ in the time domain are strengthened. However, a transition between hierarchical levels does occur when the class of allowed signals is expanded from \mathcal{CB}_π^∞ to \mathcal{CE}_π , as we will see in the following.

Theorem 7.

1. If f satisfies $f \in \mathcal{CE}_\pi$, then $B(f)$ satisfies $B(f) \in \Pi_2$.
2. If x satisfies $x \in \Pi_2 \cap [0, \pi]$ then there exists $f \in \mathcal{CE}_\pi$ such that $B(f) = x$ holds true.

Hence, the set $B[\mathcal{CE}_\pi]$ coincides with the set $\Pi_2 \cap [0, \pi]$.

Proof.

1. By (4), there exists a computable sequence $(a_m)_{m \in \mathbb{N}}$ of computable numbers such that $B(f) = \limsup_{m \rightarrow \infty} \sqrt[m]{|a_m|}$ holds true. Define $b_m := \sqrt[m]{|a_m|}$ for all $m \in \mathbb{N}$. Then, $(b_m)_{m \in \mathbb{N}}$ is a computable sequence of computable numbers and

$$B(f) = \limsup_{m \rightarrow \infty} (b_m)$$

holds true. In the following, we employ a technique applied by Zheng and Weihrauch. By definition, we have

$$\begin{aligned} \limsup_{m \rightarrow \infty} (b_m) &= \lim_{m_1 \rightarrow \infty} \left(\sup \{ b_{m_2} : m_2 \in \mathbb{N}, m_1 \leq m_2 \} \right) \\ &= \lim_{m_1 \rightarrow \infty} \left(\sup \{ b_{m_1+m_2} : m_2 \in \mathbb{N} \} \right) \end{aligned}$$

Observe that $\{b_{m_1+1+m_2} : m_2 \in \mathbb{N}\} \subseteq \{b_{m_1+m_2} : m_2 \in \mathbb{N}\}$ is satisfied for all $m_1 \in \mathbb{N}$, and hence $\sup \{b_{m_1+m_2} : m_2 \in \mathbb{N}\}$ is monotonically non-increasing in m_1 . Thus,

$$\begin{aligned} \limsup_{m \rightarrow \infty} (b_m) &= \lim_{m_1 \rightarrow \infty} \left(\sup \{ b_{m_1+m_2} : m_2 \in \mathbb{N} \} \right) \\ &= \inf_{m_1 \in \mathbb{N}} \left(\sup \{ b_{m_1+m_2} : m_2 \in \mathbb{N} \} \right) \\ &= \inf_{m_1 \in \mathbb{N}} \sup_{m_2 \in \mathbb{N}} (b_{m_1+m_2}) \end{aligned}$$

holds true. For all $m_1, m_2 \in \mathbb{N}$, define $b'_{m_1, m_2} := b_{m_1+m_2}$. Then, $(b'_m)_{m \in \mathbb{N}^2}$ is a computable double sequence of computable numbers that satisfies

$$B(f) = \inf_{m_1 \in \mathbb{N}} \sup_{m_2 \in \mathbb{N}} (b'_m).$$

That is, $(b'_m)_{m \in \mathbb{N}^2}$ is a second order upper ZW description of $B(f)$. The claim then follows by Lemma 1.

2. Consider a computable sequence $(r_m)_{m \in \mathbb{N}^2}$ of rational numbers that satisfies $\inf_{m_1 \in \mathbb{N}} \sup_{m_2 \in \mathbb{N}} (r_m) = x$. Without loss of generality, we can assume

that $r_{\mathbf{m}}$ is non-negative for all $\mathbf{m} \in \mathbb{N}^2$. Following Lemma 2, there exists a Turing machine that computes a mapping $(r_{\mathbf{m}})_{\mathbf{m} \in \mathbb{N}^2} \mapsto (r'_m)_{m \in \mathbb{N}}$ such that $(r'_m)_{m \in \mathbb{N}}$ is a non-negative computable sequence of rational numbers that satisfies $\limsup_{m \rightarrow \infty} (r'_m) = x$. We define $a_m := (r'_m)^m$ for all $m \in \mathbb{N}$. Then $(a_m)_{m \in \mathbb{N}}$ is a computable sequence of rational numbers, and hence a computable sequence of computable numbers as well. By (4), the signal

$$f(z) = \sum_{m=0}^{\infty} \frac{a_m}{m!} z^m = \sum_{m=0}^{\infty} \frac{(r'_m z)^m}{m!}, \quad z \in \mathbb{C},$$

then satisfies $B(f) = x$.

□

Remark 13. *Observe that the proof of the second statement of Theorem 7 is constructive and all involved operations can, in principle, be computed by a Turing machine. Hence if we have a hypothetical Taylor signal generator available, i.e., a machine that maps a non-negative, π -bounded, computable sequence of rational numbers $(r'_m)_{m \in \mathbb{N}}$ to the “analog” signal $\sum_{m=1}^{\infty} (r'_m z)^m / m! = f(z)$, we can in principle build an apparatus that receives a second order upper ZW description of a number x as an input and returns an analog signal f that satisfies $B(f) = x$.*

For $1 \leq p \leq +\infty$ and $f \in \mathcal{CB}_\pi^p$, there exists a monotonically non-decreasing, computable sequence of rational numbers $(r_n)_{n \in \mathbb{N}}$, such that $\lim_{n \rightarrow \infty} r_n = B(f)$ is satisfied. As Theorem 7 shows, this does not hold true for signals $f \in \mathcal{CE}_\pi$, even if the monotonicity requirement is made void. Assume for some signal $f \in \mathcal{CE}_\pi$, there exists a computable sequence $(r_n)_{n \in \mathbb{N}}$ of rational numbers such that $\lim_{n \rightarrow \infty} r_n = B(f)$ holds true. For all $m_1, m_2 \in \mathbb{N}$, define

$$r'_{m_1, m_2} := r_{m_1 + m_2}.$$

Then, $(r'_{m_1, m_2})_{m_1, m_2 \in \mathbb{N}}$ is a computable double sequence of rational numbers that satisfies

$$\inf_{m_1 \in \mathbb{N}} \sup_{m_2 \in \mathbb{N}} (r'_{m_1, m_2}) = \sup_{m_1 \in \mathbb{N}} \inf_{m_2 \in \mathbb{N}} (r'_{m_1, m_2}) = B(f).$$

Hence, we have $B(f) \in \Pi_2 \cap \Sigma_2 = \Delta_2$. But, since $B[\mathcal{CE}_\pi] = \Pi_2 \cap [0, \pi]$ and $(\Pi_2 \cap [0, \pi]) \setminus \Sigma_2 \neq \emptyset$ hold true, there exist signals $f \in \mathcal{CE}_\pi$ that satisfy $B(f) \notin \Delta_2$. Hence, for these signals, there cannot exist computable sequences $(r_n)_{n \in \mathbb{N}}$ of rational numbers that satisfy $\lim_{n \rightarrow \infty} r_n = B(f)$.

11. Bandwidth Computation and Oracle Machines

In this section, we consider bandlimited signals in the context of oracle computations. A machine OM that computes a function $g \in \mathcal{C}^*(A)$ for an arbitrary, non-recursive set $A \subsetneq \mathbb{N}$ is referred to as an *oracle machine*. In particular, we refer to the oracle machine

$$\text{OM}_T : n \mapsto \mathbb{1}_{A_T^\#}(n)$$

as *totality oracle*, and to the members

$$\text{OM}_{H,n} : m \mapsto \mathbb{1}_{D(\varphi_n)}(m)$$

of the family $(\text{OM}_{H,n})_{n \in \mathbb{N}}$ as *halting oracles*.

The properties of oracle computations and their relation to non-recursive sets have been intensively studied in the relevant literature; for a comprehensive introduction, we refer to [24]. Hence, any relation to a real world problem from science or engineering is interesting for the field of theoretical computer science. Furthermore, it yields another perspective on the *degree* of uncomputability present in the problem.

In the following, we want to relate the problem of computing the bandwidth of a bandlimited signal to these oracles. Therefore, we consider two hypotheticalal devices that operate on the set \mathcal{CE}_π :

- The device $\mathcal{O}_\sigma^\lessdot$ evaluates the mapping

$$f \mapsto \begin{cases} 1, & \text{if } B(f) < 1/2, \\ 0, & \text{otherwise.} \end{cases}$$

In other words, the device $\mathcal{O}_\sigma^\lessdot$ decides, for a signal $f \in \mathcal{CE}_\pi$, whether $B(f)$ is smaller than $1/2$ or not. Hence, $\mathcal{O}_\sigma^\lessdot$ yields a non-trivial bound on $B(f)$.

- For $x \in (0, 4]$, we denote by $A[x] \subseteq \mathbb{N}$ the unique countably infinite set that satisfies

$$x = 4 \cdot \sum_{m \in A[x]} \frac{1}{2^m}.$$

Furthermore, we set $A_n[x] := \{m \in A[x] : m \leq n + 2\}$. The device \mathcal{O}^\approx then evaluates the mapping

$$(f, n) \mapsto x[A_n[B(f)]].$$

Hence, the device \mathcal{O}^\approx yields an approximation of the number $B(f)$ which is accurate up to n -binary places.

Additionally, we consider a device \mathcal{G} which, for $(a_n)_{n \in \mathbb{N}} \rightrightarrows f$, evaluates the mapping $(a_n)_{n \in \mathbb{N}} \mapsto f$. That is, \mathcal{G} maps the description $(a_n)_{n \in \mathbb{N}}$ of f to the actual signal $f \in \mathcal{CE}_\pi$ according to (3).

In the following, we will investigate the computational strength of the oracle machines $\mathcal{O}_\sigma^<$ and \mathcal{O}^\approx . Given the fact that $B[\mathcal{CE}_\pi] = \Pi_2 \cap [0, \pi]$ and $A_T^\Phi \in \Pi_2^0$ hold true, it may not be surprising that $\mathcal{O}_\sigma^<$ and \mathcal{O}^\approx are at least as strong as the totality oracle OM_T .

Theorem 8. *There exists a Turing machine TM which, for $n \in \mathbb{N}$, computes a mapping $n \mapsto (a_n)_{n \in \mathbb{N}}$, such that*

$$\mathcal{O}_\sigma^<(\mathcal{G}(TM(n))) = OM_T(n)$$

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

Proof. Consider the runtime function $\Psi : \mathbb{N}^3 \rightarrow \{0, 1\}$. For $n \in \mathbb{N}$ fixed, define the computable double sequence $(r''_{n,\mathbf{m}})_{\mathbf{m} \in \mathbb{N}^2}$ of rational numbers by setting

$$r''_{n,\mathbf{m}} := \Psi(n, m_1, m_2)$$

for all $\mathbf{m} \in \mathbb{N}^2$. Then, the sequence $(r''_{n,\mathbf{m}})_{\mathbf{m} \in \mathbb{N}^2}$ satisfies

$$\sup_{m_2 \in \mathbb{N}} (r''_{n,\mathbf{m}}) = \begin{cases} 1, & \text{if } m_1 \in D(g_n), \\ 0, & \text{otherwise,} \end{cases}$$

for all $n, m_1 \in \mathbb{N}$, and hence

$$\inf_{m_1 \in \mathbb{N}} \sup_{m_2 \in \mathbb{N}} (r''_{n,m}) = \begin{cases} 1, & \text{if } D(g_n) = \mathbb{N}, \\ 0, & \text{otherwise,} \end{cases}$$

for all $n \in \mathbb{N}$. Following Lemma 2, there exists a Turing machine which computes a mapping $(r''_{n,m})_{m \in \mathbb{N}^2} \mapsto (r'_{n,m})_{m \in \mathbb{N}}$, such that $(r'_{n,m})_{m \in \mathbb{N}}$ is a computable sequence of rational numbers that satisfies

$$\limsup_{m \rightarrow \infty} (r'_{n,m}) = \inf_{m_1 \in \mathbb{N}} \sup_{m_2 \in \mathbb{N}} (r''_{n,m}).$$

Consider the mapping $\xi : M \mapsto 1$ and define the computable double sequence $(r_{n,m})_{m \in \mathbb{N}^2}$ of rational numbers via

$$r_{n,m_1,m_2} := (r'_{n,m_1})^{m_2}$$

for all $n, m_1, m_2 \in \mathbb{N}$. The pair $((r_{n,m})_{m \in \mathbb{N}^2}, \xi)$ is the standard description of a computable sequence $(a_{n,m})_{m \in \mathbb{N}}$ of computable numbers that satisfies

$$\limsup_{m \rightarrow \infty} \sqrt[m]{|a_{n,m}|} = \limsup_{m \rightarrow \infty} (r'_{n,m}) = \begin{cases} 1, & \text{if } D(g_n) = \mathbb{N}, \\ 0, & \text{otherwise.} \end{cases}$$

We now define $\text{TM} : n \mapsto \text{TM}(n) := (a_{n,m})_{m \in \mathbb{N}}$. Then, by (4), the signal $f(n) := \mathcal{G}(\text{TM}(n))$ satisfies

$$B(f) = \begin{cases} 1, & \text{if } D(g_n) = \mathbb{N}, \\ 0, & \text{otherwise.} \end{cases}$$

Consequently, for all $n \in \mathbb{N}$, we have $1/2 < B(f)$ if and only if φ_n is a total function. Hence, $\mathcal{O}_\sigma^<(\mathcal{G}(\text{TM}(n))) = \text{OM}_T(n)$ holds true for all $n \in \mathbb{N}$. \square

Theorem 9. *There exist a $f \in \mathcal{CE}_\pi$ as well as a Turing machine TM which, for $n \in \mathbb{N}$ and $x \in \mathbb{Q}$, computes a mapping $(x, n) \mapsto m \in \{0, 1\}$, such that*

$$\text{TM}(\mathcal{O}^\approx(f, n), n) = \text{OM}_T(n)$$

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

Proof. We have $x[A_T^\Phi] \in \Pi_2$, c.f. Lemma 4. Hence, by Theorem 7, there exists a signal $f \in \mathcal{CE}_\pi$ such that $B(f) = x[A_T^\Phi]$ holds true. Observe that for all $n \in \mathbb{N}$, we have $D(\varphi_n) = \mathbb{N}$ if and only if

$$n \in A_n[x[A_T^\Phi]] = A_n[B(f)]$$

is satisfied. In particular, $D(\varphi_n) = \mathbb{N}$ holds true if and only if the n th binary place of $4 \cdot x[A_n[B(f)]]$ equals one. Hence, with $\mathcal{O}^\approx(f, n) = x[A_n[B(f)]]$, we have $B(f) = x[A_T^\Phi]$ if and only if

$$\underbrace{\mathcal{O}^\approx(f, n) \cdot 2^{n+2} - \lfloor \mathcal{O}^\approx(f, n) \cdot 2^{n+1} \rfloor \cdot 2}_{=: g(\mathcal{O}^\approx(f, n), n)} = 1.$$

Defining

$$\text{TM}(\mathcal{O}^\approx(f, n), n) := \begin{cases} 1, & \text{if } g(\mathcal{O}^\approx(f, n), n) = 1, \\ 0, & \text{otherwise,} \end{cases}$$

yields the required Turing machine. \square

The remainder of this section will be dedicated to relating the set \mathcal{CB}_π^∞ to the family of halting oracles $(\text{OM}_{\text{H},n})_{n \in \mathbb{N}}$ as *halting oracles*.

Theorem 10. *Let f satisfy $f \in \mathcal{CB}_\pi^\infty$. There exists a Turing machine TM that computes a mapping*

$$((\text{OM}_{\text{H},n})_{n \in \mathbb{N}}, f, m) \mapsto q \in \mathbb{Q},$$

such that $|B(f) - q| < 2^{-m}$ holds true for all $m \in \mathbb{N}$.

Proof. The proof again employs the dyadic expansion of the number $B(f)$. Without loss of generality we assume $B(f) \in [0, 1] \cap \Sigma_1$. There exists a Turing machine TM' that computes a mapping $f \mapsto (r_l)_{l \in \mathbb{N}}$, such that $(r_l)_{l \in \mathbb{N}}$ is a monotonically non-decreasing, computable sequence of rational numbers, such that $\lim_{l \rightarrow \infty} r_l = B(f)$ holds true, c.f. [17] for details. Denote by A_l the largest subset of \mathbb{N} such that $x[A_l] \leq B(f)$ holds true and define

$$A := \bigcup_{l \in \mathbb{N}} A_l.$$

Then, we have $B(f) = x[A]$ as well as $A_l \subseteq A_{l+1}$ for all $l \in \mathbb{N}$. Furthermore, the set A is recursively enumerable and the mapping $(r_l)_{l \in \mathbb{N}} \mapsto g$, where g is a recursive function that satisfies $D(g) = A$, can be computed by a Turing machine. Using the family $(OM_{H,n})_{n \in \mathbb{N}}$ of halting oracles, we can now compute the set

$$B_m := \{k \in \mathbb{N} : k \leq m + 1, k \in D(g)\}.$$

For all $m \in \mathbb{N}$, we have

$$B_m = \{k \in A : k \leq m + 1\}.$$

Since A contains the dyadic expansion of $B(f)$, we have $|B(f) - x[B_m]| < 2^{-m}$. \square

Remark 14. For $f \in \mathcal{CE}_\pi$ such that $B(f) \in \Delta_2$ holds true, the family $(OM_{H,n})_{n \in \mathbb{N}}$ allows for the computation of $B(f)$ as well, c.f. [32, Lemma 5.4, p. 58]. However, the dependency on f is non recursive, since the description $(a_n)_{n \in \mathbb{N}}$ in the sense of Definition 14 is not a feasible input.

12. Conclusion

In the previous chapters, we have considered different descriptions for the signal classes \mathcal{CB}_π^p for $1 \leq p \leq +\infty$ and \mathcal{CE}_π , and showed that $B(\mathcal{CB}_\pi^p) = \Sigma_1 \cap [0, \pi]$ and $B(\mathcal{CE}_\pi) \cap [0, \pi]$ hold true for all p that satisfy $1 \leq p \leq +\infty$, which is a full characterization of $B(\mathcal{CB}_\pi^p)$ and $B(\mathcal{CE}_\pi)$ in terms of the arithmetical hierarchy of real numbers. In the scope of our analysis, we were able to confirm all conjectures posed in [17]:

1. There exist signals $f \in \mathcal{CE}_\pi$ such that *no* computable sequence $(r_n)_{n \in \mathbb{N}}$ of rational numbers satisfies $\lim_{n \rightarrow \infty} r_n = B(f)$.
2. For signals $f \in \mathcal{CE}_\pi$ and numbers $\sigma \in (0, \pi) \cap \mathbb{R}_c$, the set $\{f \in \mathcal{CE}_\pi : B(f) > \sigma\}$ is not semi-decidable.

3. If $\underline{\text{TM}}_{\text{BW}}$ and $\overline{\text{TM}}_{\text{BW}}$ are Turing machines such that for all signals f in \mathcal{CE}_π that satisfy $B(f) \leq \pi$, we have

$$\underline{\text{TM}}_{\text{BW}}(f) \leq B(f) \leq \overline{\text{TM}}_{\text{BW}}(f),$$

then $\underline{\text{TM}}_{\text{BW}}$ and $\overline{\text{TM}}_{\text{BW}}$ necessarily return trivial values, i.e., $\underline{\text{TM}}_{\text{BW}}(f) = 0$ and $\overline{\text{TM}}_{\text{BW}}(f) = \pi$ holds true for all feasible signals f .

Last but not least, we showed that the problem of computing the bandwidth of a signal $f \in \mathcal{CE}_\pi$ is at least as hard as computing a totality oracle for the set of recursive functions. Computing the bandwidth of a signal $f \in \mathcal{CB}_\pi^p$, for $1 \leq p \leq +\infty$, on the other hand, can be reduced to computing a halting oracle.

It is interesting to note that some of the questions can be positively answered for \mathcal{CB}_π^2 . For example, the set $\{f \in \mathcal{CB}_\pi^2: B(f) > \sigma\}$ is semi-decidable. Further, for $f \in \mathcal{CB}_\pi^2$, we can find a computable monotonically increasing sequence of lower bounds for $B(f)$ that converges to $B(f)$. The restrictions that we have in the time and frequency domain for signals in \mathcal{CB}_π^1 and \mathcal{CB}_π^2 make it possible that important signal processing problems, such as those above, can be algorithmically solved. In contrast, signals in \mathcal{CE}_π have—except for a simple growth condition—no such restrictions.

Questions of computability have not received much attention in control theory and signal processing so far. Recent results have shown that there are important problems that cannot be solved on a digital computer. For example, problems can occur in the computation of the Fourier transform [13], the Fourier series [35], and the spectral factorization [36], as well as in downsampling and the computation of the bandlimited interpolation [37].

In [38], conditions were analyzed under which the computability of a discrete-time signal implies the computability of the corresponding continuous-time signal. The computability of the actual bandwidth was studied in [17] and [16]. In [17] it was shown that there exist signals $f \in \mathcal{CB}_\pi^2$ for which the actual bandwidth $B(f)$ is not computable. In contrast to [17], we study the most general class of bandlimited signals in this paper and the question whether upper and lower bounds can be algorithmically determined.

The problem of computing the period of a periodic computable continuous function is similar to the problem studied in this paper. In Shor’s famous algorithms for factorizing natural numbers and computing the discrete logarithm [39], the core task—and the only part that has to be implemented on a universal quantum computer—is the computation of the period of certain computable continuous functions. Interestingly, all candidates for a “post-quantum cryptography” that already failed, were broken by quantum algorithms that compute the period of certain functions. It seems as if finding periods of functions is the only class of well-investigated mathematical problems for which it was possible to develop quantum algorithms that have a substantial complexity advantage over the best known classical algorithms.

As indicated in the introduction, our results contribute to the problem of understanding the differences between different approaches for computation. Since the problem of sampling analog signals inherently involves analog technology, the question arises if, for example, an analog or neuromorphic computer could solve the bandwidth estimation problem up to any accuracy. To answer questions of this kind, it is important to understand the mathematical structure behind the different approaches for computation. The authors believe that this topic will gain further importance in the future.

References

- [1] R. Isermann, *Digital Control Systems*, volume 1, Springer-Verlag Berlin Heidelberg, 1989.
- [2] K. Ogata, *Discrete-Time Control Systems*, Pearson, 1995.
- [3] C. E. Shannon, Communication in the presence of noise, in: *Proceedings of the IRE*, volume 37, 1949, pp. 10–21.
- [4] J. R. Higgins, *Sampling Theory in Fourier and Signal Analysis – Foundations*, Oxford University Press, 1996.

- [5] M. Pawlak, U. Stadtmüller, Recovering band-limited signals under noise, *IEEE Transactions on Information Theory* 42 (1996) 1425–1438. doi:10.1109/18.532883.
- [6] S. Cambanis, E. Masry, Truncation error bounds for the cardinal sampling expansion of band-limited signals, *IEEE Transactions on Information Theory* 28 (1982) 605–612. doi:10.1109/TIT.1982.1056527.
- [7] H. J. Landau, Sampling, data transmission, and the Nyquist rate, *Proceedings of the IEEE* 55 (1967) 1701–1706.
- [8] C. Herley, P. W. Wong, Minimum rate sampling and reconstruction of signals with arbitrary frequency support, *IEEE Transactions on Information Theory* 45 (1999) 1555–1564. doi:10.1109/18.771158.
- [9] M. K. Habib, Digital representations of operators on band-limited random signals, *IEEE Transactions on Information Theory* 47 (2001) 173–177. doi:10.1109/18.904520.
- [10] U. J. Mönich, H. Boche, A two channel system approximation for bandlimited functions, *IEEE Transactions on Information Theory* 63 (2017) 5496–5505. doi:10.1109/TIT.2017.2724550.
- [11] S. B. Weinstein, P. M. Ebert, Data transmission by frequency-division multiplexing using the discrete Fourier transform, *IEEE Transactions on Communication Technology* 19 (1971) 628–634. doi:10.1109/TCOM.1971.1090705.
- [12] J. A. C. Bingham, Multicarrier modulation for data transmission: an idea whose time has come, *IEEE Communications Magazine* 28 (1990) 5–14. doi:10.1109/35.54342.
- [13] H. Boche, U. J. Mönich, Turing computability of the Fourier transform of bandlimited functions, in: *Proceedings of the 2019 IEEE International Symposium on Information Theory*, 2019, pp. 380–384. doi:10.1109/ISIT.2019.8849462.
- [14] H. Boche, U. J. Mönich, Computability of the Fourier transform and ZFC, in: *2019 International Conference on Sampling Theory and Applications (SampTA)*, 2019, pp. 1–4. doi:10.1109/SampTA45681.2019.9030870.

- [15] H. Boche, 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, pp. 2459–2463. doi:10.1109/ISIT.2019.8849557.
- [16] H. Boche, U. J. Mönich, Algorithmic computability of the signal bandwidth, IEEE Transactions on Information Theory 67 (2021) 2450–2471. doi:10.1109/TIT.2021.3057672.
- [17] H. Boche, U. J. Mönich, Optimal sampling rate and bandwidth of bandlimited signals—an algorithmic perspective, in: Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '20), 2020, pp. 5905–5909. doi:10.1109/ICASSP40776.2020.9053158.
- [18] B. Y. Levin, Lectures on Entire Functions, AMS, Providence, Rhode Island, 1996.
- [19] G. Doetsch, Handbuch der Laplace-Transformationen – Band I – Theorie der Laplace-Transformationen, Verlag Birkhäuser Basel, 1950.
- [20] K. Weihrauch, Computable Analysis: An Introduction, Springer-Verlag Berlin Heidelberg, 2000.
- [21] M. B. Pour-El, J. I. Richards, Computability in Analysis and Physics, Springer-Verlag Berlin Heidelberg, 1989.
- [22] G. S. Boolos, J. P. Burgess, R. C. Jeffrey, Computability and Logic, Cambridge University Press, 2002.
- [23] J. Avigad, V. Brattka, Computability and analysis: the legacy of Alan Turing, in: R. Downey (Ed.), Turing’s Legacy: Developments from Turing’s Ideas in Logic, Cambridge University Press, 2014.
- [24] R. I. Soare, Recursively Enumerable Sets and Degrees, Perspectives in Mathematical Logic, Springer-Verlag Berlin Heidelberg, 1987.
- [25] R. I. Soare, Turing Computability, Springer-Verlag Berlin Heidelberg, 2016.
- [26] A. M. Turing, On computable numbers, with an application to the Entscheidungsproblem, Proceedings of the London Mathematical Society s2-42 (1936) 230–265.

- [27] A. M. Turing, On computable numbers, with an application to the Entscheidungsproblem. A correction, *Proceedings of the London Mathematical Society* s2-43 (1937) 544–546.
- [28] S. C. Kleene, General recursive functions of natural numbers, *Mathematische Annalen* 112 (1936) 727–742. URL: <https://doi.org/10.1007/BF01565439>. doi:10.1007/BF01565439.
- [29] A. M. Turing, Computability and λ -definability, *Journal of Symbolic Logic* 2 (1937) 153–163. doi:10.2307/2268280.
- [30] Y. I. Manin, *A Course in Mathematical Logic for Mathematicians*, Graduate Texts in Mathematics, Springer-Verlag New York, 2010.
- [31] K. Gödel, Über formal unentscheidbare sätze der principia mathematica und verwandter systeme i, *Monatshefte für Mathematik und Physik* 38 (1931) 173–198. doi:10.1007/BF01700692.
- [32] X. Zheng, K. Weihrauch, The arithmetical hierarchy of real numbers, *Mathematical Logic Quarterly* 47 (2001) 51–65.
- [33] S. C. Kleene, Recursive predicates and quantifiers, *Transactions of the American Mathematical Society* 53 (1943) 41–73.
- [34] A. Mostowski, On definable sets of positive integers, *Fundamenta Mathematicae* 34 (1947) 81–112.
- [35] H. Boche, 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)*, 2019, pp. 5013–5017. doi:10.1109/ICASSP.2019.8683074.
- [36] H. Boche, V. Pohl, On the algorithmic solvability of spectral factorization and applications, *IEEE Transactions on Information Theory* 66 (2020) 4574–4592. doi:10.1109/TIT.2020.2968028.
- [37] H. Boche, U. J. Mönich, Downsampling of bounded bandlimited signals and the bandlimited interpolation: Analytic properties and computability, *IEEE Transactions on Signal Processing* 67 (2019) 6424–6439. doi:10.1109/TSP.2019.2954972.

- [38] H. Boche, U. J. Mönich, Effective approximation of bandlimited signals and their samples, in: Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '20), 2020, pp. 5590–5594. doi:10.1109/ICASSP40776.2020.9053196.
- [39] P. W. Shor, Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer, SIAM Journal on Computing 26 (1997) 1484–1509. doi:10.1137/S0097539795293172.