

The Solvability Complexity Index - Computer Science and Logic Meet Scientific Computing

Jonathan Ben-Artzi
Department of Mathematics
Imperial College London
email: *J.Ben-Artzi@imperial.ac.uk*

Anders C. Hansen
DAMTP, Centre for Mathematical Sciences
University of Cambridge
email: *A.Hansen@damtp.cam.ac.uk*

Olavi Nevanlinna
Department of Mathematics and Systems Analysis
Aalto University
email: *olavi.nevanlinna@aalto.fi*

Markus Seidel
Department of Mathematics
West Saxon University of Applied Sciences, Zwickau
email: *markus.seidel@fh-zwickau.de*

Abstract

We address and establish some of the basic barriers in the theory of computations, as well as settle long standing problems such as the computational spectral problem and computability of spectra of Schrödinger operators. These barriers establish a new classification theory of problems in scientific computing that is reminiscent of the Arithmetical Hierarchy in computer science and logic. In particular, many computational problems can be solved as follows: either a solution can be computed by an algorithm in finite time, or a sequence of approximations is created by an algorithm (each element in the sequence is computed in finite time), and the solution to the problem is the limit of this sequence. These problems include the core problems in computer science as well as key problems in scientific computing such as computing eigenvalues of an $n \times n$ matrix, computing solutions to PDEs on bounded domains, etc. Thus, it may be surprising that, for several basic problems in computations (computing spectra of operators, solutions to linear inverse problems, solutions to convex optimization problems or roots of polynomials using rational maps), such a procedure based on one limit is impossible. Yet, one can compute solutions to these problems, but only by using several limits.

To analyze this phenomenon we use the Solvability Complexity Index (SCI). The SCI is the smallest number of limits needed in order to compute a desired quantity, given a set of restrictions on what the algorithm can do and process. This allows one to use the SCI to analyze problems in computation ranging from Smale's question on the existence of generally convergent algorithms for polynomial root finding, with its solution provided by McMullen and Doyle & McMullen, via computational spectral problems, quantum mechanics, inverse problems, PDEs, optimization in imaging and sampling theory to logic and the Arithmetical Hierarchy. The fact that there are many core problems in computations that have $\text{SCI} > 1$ (more than one limit is required to compute the solution) demonstrates how the set of problems in scientific computing has a very rich classification hierarchy in terms of the SCI.

1 Introduction

The classical theory of computations and complexity is based on asking the question:

- (Q.1) Given a computational problem, does there exist an algorithm, satisfying certain restrictions, that can compute the solution to the problem in finite time? If this is impossible, as in the case of computing eigenvalues of general $n \times n$ matrices, roots of polynomials, or solutions to PDEs for example, one asks: does there exist a sequence of approximations converging to the solution, where each element in the sequence can be computed in finite time?

If the answer to this question is 'yes', the next question is: how complex or difficult will it be to carry out the computation? The vast amount of research over the last century has left us with the following two main classes of problems:

- (I) The problem can be solved in finite time. This class encompasses most of the problems in computer science with its many complexity classes such as P, NP, PSPACE etc. The complexity theory of such problems contains famous results [1, 24, 40] and hosts the legendary "P versus NP" problem [23, 67].
- (II) The problem cannot be solved in finite time, however, it can be solved by computing a sequence of approximations, and the solution to the problem is the limit of the approximations. The complexity theory of such problems contains famous questions, among them Smale's 17th problem [3, 21] and a very rich mathematical theory [7, 20, 22, 58–60], which includes different models such as for example the Bit Model [18, 19, 41], the TTE model [17, 52, 66], the BSS model [6–8] and others [31, 32, 43, 53, 64]. The field of information based complexity theory [51, 62] is devoted specifically to these types of problems.

The key issue is that, as this paper establishes, there are fundamental barriers that prevent many problems at the heart of computational theory from fitting into the two classes above. In particular, there is a third class of problems.

- (III) The problem cannot be computed by passing to the limit of a sequence of approximations, however, it can be computed by passing to several limits over a (multi parameter) family of approximations. This class includes problems such as spectral problems, inverse problems, optimization problems, root finding etc., that are some of the core problems in scientific computing.

We establish Class III through the theory of the Solvability Complexity Index (SCI) and towers of algorithms. The SCI is the smallest number of limits needed in order to compute the desired quantity, given a certain type of algorithm. Moreover, the existence of the SCI and Class III imply a potentially very rich classification theory in terms of a non-collapsing hierarchy whose subclasses are described by the SCI. We refer to this as the SCI Hierarchy (see section 3.8) for which the Δ_m classes in the Arithmetical Hierarchy are special cases. The main consequence of this paper is therefore:

The field of scientific computing has a hierarchical structure that is reminiscent of the Arithmetical Hierarchy in logic and computer science. As we establish, the SCI phenomenon is independent of the computational model (although obviously the SCI differs in the different models), thus any theory aiming at establishing the foundations of scientific computing allowing core problems such as spectral problems, inverse problems, PDEs, optimization problems etc., will have to include the SCI.

Problems in Class III are typically discovered after a long time with attempts of finding a solution to Q.1 for a specific problem. One of the first rigorous examples came with Smale's question [59] on the existence of generally convergent algorithms for polynomial root finding, with its solution provided by McMullen [47, 48, 61] and Doyle & McMullen. This problem, for the quartic and the quintic, is unsolvable in one limit but solvable in three limits. Another example is the computational spectral problem that we finally settle in [4].

2 The Solvability Complexity Index and towers of algorithms

We propose a unified setup for computational problems. This setup does not require a machine (Turing, BSS or any other), however a machine certainly can be specified (see Section 3.2) if desired. The important point is that the SCI can be defined regardless of the axiomatic setup. The basic objects are:

$$\Omega \text{ is some set, called the } \textit{primary} \text{ set,} \quad (2.1a)$$

$$\Lambda \text{ is a set of complex valued functions on } \Omega, \text{ called the } \textit{evaluation} \text{ set,} \quad (2.1b)$$

$$\mathcal{M} \text{ is a metric space,} \quad (2.1c)$$

$$\Xi \text{ is a mapping } \Omega \rightarrow \mathcal{M}, \text{ called the } \textit{problem} \text{ function.} \quad (2.1d)$$

The set Ω is the set of objects that give rise to our computational problems. It can be a family of (finite or infinite) matrices, polynomials, Schrödinger, Dirac operators or another specific class of differential operators, a type of PDEs with initial conditions, inverse problems (operator plus measurements) etc. The problem function $\Xi : \Omega \rightarrow \mathcal{M}$ is what we are interested in computing. It could be the set of eigenvalues of an $n \times n$ matrix, root(s) of a polynomial, the spectrum of an operator, solution to the PDE etc. Finally, the set Λ is the collection of functions that provide us with the information we are allowed to read, say matrix elements, polynomial coefficients or pointwise values of a potential function of a Schrödinger operator, for example. This leads to the following definition.

Definition 2.1 (Computational Problem). *Given a primary set Ω , an evaluation set Λ , a metric space \mathcal{M} and a problem function $\Xi : \Omega \rightarrow \mathcal{M}$ we call the collection $\{\Xi, \Omega, \mathcal{M}, \Lambda\}$ a computational problem.*

The goal of this abstract definition is to allow most of the known computational problems into the framework. However, to make this abstract definition a little more concrete, let us consider the following example.

Example 2.2. Let $\Omega = \mathcal{B}(\mathcal{H})$, the set of all bounded linear operators on a separable Hilbert space \mathcal{H} , and the problem function Ξ be the mapping $A \mapsto \text{sp}(A)$ (the spectrum of A). Here (\mathcal{M}, d) is the set of all compact subsets of \mathbb{C} provided with the Hausdorff metric $d = d_H$. The evaluation functions in Λ could consist of the family of all functions $f_{i,j} : A \mapsto \langle Ae_j, e_i \rangle$, $i, j \in \mathbb{N}$, which provide the entries of the matrix representation of A w.r.t. an orthonormal basis $\{e_i\}_{i \in \mathbb{N}}$. Of course, Ω could be a strict subset of $\mathcal{B}(\mathcal{H})$, for example the set of self-adjoint or normal operators, and Ξ could have represented the pseudo spectrum, the essential spectrum or any other interesting information about the operator.

Our aim is to find and to study families of functions which permit to approximate the function Ξ . The main pillar of our framework is the concept of a tower of algorithms.

Definition 2.3 (General Algorithm). *Given a computational problem $\{\Xi, \Omega, \mathcal{M}, \Lambda\}$, a general algorithm is a mapping $\Gamma : \Omega \rightarrow \mathcal{M}$ such that for each $A \in \Omega$*

- (i) *there exists a finite subset of evaluations $\Lambda_\Gamma(A) \subset \Lambda$,*
- (ii) *the action of Γ on A only depends on $\{A_f\}_{f \in \Lambda_\Gamma(A)}$ where $A_f := f(A)$,*
- (iii) *for every $B \in \Omega$ such that $B_f = A_f$ for every $f \in \Lambda_\Gamma(A)$, it holds that $\Lambda_\Gamma(B) = \Lambda_\Gamma(A)$.*

A general algorithm has no restrictions on the operations allowed. The only restriction is that it can only take a finite amount of information, though it is allowed to *adaptively* choose the finite amount of information it reads depending on the input. Condition (iii) assures that the algorithm reads the information in a consistent way.

Definition 2.4 (Tower of Algorithms). Given a computational problem $\{\Xi, \Omega, \mathcal{M}, \Lambda\}$, a tower of algorithms of height k for $\{\Xi, \Omega, \mathcal{M}, \Lambda\}$ is a family of sequences of functions

$$\Gamma_{n_k} : \Omega \rightarrow \mathcal{M}, \Gamma_{n_k, n_{k-1}} : \Omega \rightarrow \mathcal{M}, \dots, \Gamma_{n_k, \dots, n_1} : \Omega \rightarrow \mathcal{M},$$

where $n_k, \dots, n_1 \in \mathbb{N}$ and the functions Γ_{n_k, \dots, n_1} at the “lowest level” of the tower are general algorithms in the sense of Definition 2.3. Moreover, for every $A \in \Omega$,

$$\Xi(A) = \lim_{n_k \rightarrow \infty} \Gamma_{n_k}(A), \quad \Gamma_{n_k, \dots, n_{j+1}}(A) = \lim_{n_j \rightarrow \infty} \Gamma_{n_k, \dots, n_j}(A) \quad j = k-1, \dots, 1.$$

In addition to a general tower of algorithms (defined above), we will focus on arithmetic towers, and, if one specifies a Turing machine, what we have called a Kleene-Shoenfield tower, see Section 3.2.

Definition 2.5 (Arithmetic Tower). Given a computational problem $\{\Xi, \Omega, \mathcal{M}, \Lambda\}$, an Arithmetic Tower of Algorithms of height k for $\{\Xi, \Omega, \mathcal{M}, \Lambda\}$ is a tower of algorithms where the lowest level functions $\Gamma = \Gamma_{n_k, \dots, n_1} : \Omega \rightarrow \mathcal{M}$ satisfy the following: For each $A \in \Omega$ the action of Γ on A consists of only finitely many arithmetic operations on $\{A_f\}_{f \in \Lambda_\Gamma(A)}$, where we remind that $A_f = f(A)$.

Given the definitions above we can now define the key concept, namely, the Solvability Complexity Index:

Definition 2.6 (Solvability Complexity Index (SCI)). A given computational problem $\{\Xi, \Omega, \mathcal{M}, \Lambda\}$ is said to have Solvability Complexity Index $\text{SCI}(\Xi, \Omega, \mathcal{M}, \Lambda)_\alpha = k$ with respect to towers of algorithms of type α if k is the smallest integer for which there exists a tower of algorithms of type α of height k . If no such tower exists then $\text{SCI}(\Xi, \Omega, \mathcal{M}, \Lambda)_\alpha = \infty$. If there exists a tower $\{\Gamma_n\}_{n \in \mathbb{N}}$ of type α and height one such that $\Xi = \Gamma_{n_1}$ for some $n_1 < \infty$, then we define $\text{SCI}(\Xi, \Omega, \mathcal{M}, \Lambda)_\alpha = 0$. The type α may be General, or Arithmetic, denoted respectively G and A . We may sometimes write $\text{SCI}(\Xi, \Omega)_\alpha$ to simplify notation when \mathcal{M} and Λ are obvious.

We will let $\text{SCI}(\Xi, \Omega)_A$ and $\text{SCI}(\Xi, \Omega)_G$ denote the SCI with respect to arithmetic towers and general towers, respectively. Note that a general tower means just a tower of algorithms as in Definition 2.4, where there are no restrictions on the mathematical operations. Thus, clearly $\text{SCI}(\Xi, \Omega)_A \geq \text{SCI}(\Xi, \Omega)_G$. The evaluation sets Λ and $\Lambda_\Gamma(A)$, given a general algorithm Γ and an element $A \in \Omega$, are crucial when determining the SCI as the following example demonstrates.

Suppose we want to compute the area of a disc given its radius. Denote the set of discs by Ω . Let f be the evaluation function which assigns to a closed disc D its radius $r = f(D)$. Let $\Lambda_1 = \{f\}$ and let Λ_2 be the union of Λ_1 and the set of all constant functions on Ω . If we allow Λ_2 then the SCI of this problem with respect to arithmetic towers is obviously zero as the formula πr^2 immediately gives the answer. However, if we only allow Λ_1 then we must have that $\text{SCI} > 0$ (π cannot be obtained in finitely many arithmetic operations from the input r).

3 The main results

3.1 Computing spectra

The affirmative answer to Q.1 for eigenvalues of most finite matrices has been known since the fifties, however, was completely unknown in the case of infinite matrices, even in the self-adjoint case, although certain special cases were known. This posed a serious problem as most operators in mathematical physics whose spectra are of interest act on infinite-dimensional spaces. Thus, the field of computational spectral theory has produced a vast amount of research on specific examples (we can only cite a tiny subset

here) [2, 10, 11, 27, 28, 34–36, 45, 54–56]. However, no algorithm that could compute the spectra of general matrices was known. A partial solution to this long standing problem was provided in [36] where an algorithm, allowing arbitrary infinite matrices, using 3 limits with arithmetic operations and radicals was constructed. We can now provide a complete solution to this problem in Theorem 3.4 including also the essential spectrum. Before we state the main theorem we need some definitions.

Definition 3.1 (Dispersion). *We say that the dispersion of $A \in \mathcal{B}(\ell^2(\mathbb{N}))$ is bounded by the function $f : \mathbb{N} \rightarrow \mathbb{N}$ if $\max\{\|(I - P_{f(m)})AP_m\|, \|P_m A(I - P_{f(m)})\|\} \rightarrow 0$ as $m \rightarrow \infty$, where P_m denotes the projection onto $\text{span}\{e_1, \dots, e_m\}$ and $\{e_j\}$ is the canonical basis. We denote the set of all operators with the dispersion bounded by f by $\mathcal{B}_f(\ell^2(\mathbb{N}))$.*

Definition 3.2 (Controlled resolvent). *Given some continuous function $g : [0, \infty) \rightarrow [0, \infty)$ that vanishes only at $x = 0$ and such that $g(x) \rightarrow \infty$ as $x \rightarrow \infty$, define $\mathcal{R}_g(\mathcal{H})$ to be the class of all closed linear operators on the Hilbert space \mathcal{H} satisfying $\|(A - zI)^{-1}\|^{-1} \geq g(\text{dist}(z, \text{sp}(A)))$ for every $z \in \mathbb{C}$, where we use the convention $\|B^{-1}\|^{-1} := 0$ if B has no bounded inverse. Note that normal (and self-adjoint) operators satisfy this condition with $g(x) = x$.*

Just as the spectrum is crucial in mathematical physics, the pseudospectrum [63] has also received a substantial amount of interest.

Definition 3.3 (Pseudospectra). *For $\epsilon > 0$, the ϵ -pseudospectrum of a closed linear operator H on a Hilbert space \mathcal{H} is defined to be the set $\text{sp}_\epsilon(H) := \text{cl}(\{z \in \mathbb{C} : \|(H - zI)^{-1}\| > 1/\epsilon\})$, where cl denotes the closure.*

We will consider variants of the computational problem suggested in Example 2.2. Consider the following classes of operators:

$$\Omega_1 = \mathcal{B}(\ell^2(\mathbb{N})), \quad \Omega_2 = \mathcal{B}_f(\ell^2(\mathbb{N})), \quad \Omega_3 = \mathcal{R}_g(\ell^2(\mathbb{N})) \cap \mathcal{B}(\ell^2(\mathbb{N})), \quad \Omega_4 = \mathcal{K}(\ell^2(\mathbb{N})),$$

the latter meaning the set of all compact operators on $\ell^2(\mathbb{N})$, and $\Omega_5 = \Omega_2 \cap \Omega_3$. Define the following problem functions: $\Xi_1(A) = \text{sp}(A)$, for $\epsilon > 0$ let $\Xi_2(A) = \text{sp}_\epsilon(A)$, as well as $\Xi_3(A) = \text{sp}_{\text{ess}}(A)$ (the essential spectrum). When considering Ω_3 then Λ contains, besides the usual evaluations $f_{i,j} : A \mapsto \langle Ae_j, e_i \rangle$ ($i, j \in \mathbb{N}$) the constant functions $g_{i,j} : A \mapsto g(i/j)$ ($i, j \in \mathbb{N}$), which provide the values of g in all positive rational numbers. When considering Ω_2 the values $f(m)$ ($m \in \mathbb{N}$) shall be available to the algorithms as constant evaluation functions. We then have

Theorem 3.4 (SCI and spectra). *Given the above setup*

$$\begin{aligned} \text{Spectrum: } & \text{SCI}(\Xi_1, \Omega_1)_G = \text{SCI}(\Xi_1, \Omega_1)_A = 3, & \text{SCI}(\Xi_1, \Omega_i)_G = \text{SCI}(\Xi_1, \Omega_i)_A = 2, & i = 2, 3, \\ & \text{SCI}(\Xi_1, \Omega_4)_G = \text{SCI}(\Xi_1, \Omega_4)_A = 1, & \text{SCI}(\Xi_1, \Omega_5)_G = \text{SCI}(\Xi_1, \Omega_5)_A = 1, \\ \text{Pseudospectrum: } & \text{SCI}(\Xi_2, \Omega_1)_G = \text{SCI}(\Xi_2, \Omega_1)_A = 2, & \text{SCI}(\Xi_2, \Omega_2)_G = \text{SCI}(\Xi_2, \Omega_2)_A = 1, \\ & \text{SCI}(\Xi_2, \Omega_4)_G = \text{SCI}(\Xi_2, \Omega_4)_A = 1, \\ \text{Ess-spectrum: } & \text{SCI}(\Xi_3, \Omega_1)_G = \text{SCI}(\Xi_3, \Omega_1)_A = 3, & \text{SCI}(\Xi_3, \Omega_5)_G = \text{SCI}(\Xi_3, \Omega_5)_A = 2. \end{aligned}$$

This means that it is impossible to compute spectra and essential spectra of infinite matrices in less than three limits. This is universal for all algorithms regardless of the operations allowed (arithmetic operation, radicals etc). The only assumption on the algorithms is that they can only read a finite number of matrix entries in each iteration step. This implies that even if there had existed an algorithm that could compute the spectrum of a finite dimensional matrix using finitely many arithmetic operations (of course no such algorithm exists), one could still not compute the spectrum of an infinite matrix in less than three limits.

However, it is possible to compute spectra and essential spectra in three limits when allowing arithmetic operations of complex numbers. The proofs are all constructive and yield implementable practical algorithms. This finally settles the computational spectral problem.

3.2 The SCI, the Arithmetical Hierarchy and computer science

When considering decision problems there is a clear connection between the SCI and the Arithmetical Hierarchy [42, 50]. In particular, due to Schoenfield's Limit Lemma, the Δ_m sets in the Arithmetical Hierarchy can equivalently be characterised in terms of the SCI.

Given a subset $A \subset \mathbb{Z}_+$ with characteristic function χ_A being definable in First-Order Arithmetic, we are interested in the SCI of deciding whether a given number $x \in \mathbb{Z}_+$ belongs to A or not. In other words, we want to determine the value of the characteristic function of A at the point x . Thus, we want to consider Towers of Algorithms for the problem function χ_A where the functions/relations at the lowest level shall be computable, and we again ask for the minimal height. More precisely, we consider the primary set $\Omega := \mathbb{Z}_+$, the evaluation set $\Lambda = \{\lambda\}$ consisting of the function $\lambda : \mathbb{Z}_+ \rightarrow \mathbb{C}$, $x \mapsto x$, the metric space $\mathcal{M} := (\{\text{Yes, No}\}, d_{discr})$, where d_{discr} denotes the discrete metric, and consider all functions $\Xi : \Omega \rightarrow \mathcal{M}$ in the Arithmetical Hierarchy. In honour of Kleene and Shoenfield we call a Tower of Algorithms that is computable (in the sense of Turing [65]) a Kleene-Shoenfield tower.

Definition 3.5 (Kleene-Shoenfield tower). *A tower of algorithms given by a family $\{\Gamma_{n_k, \dots, n_1} : \Omega \rightarrow \mathcal{M} : n_k, \dots, n_1 \in \mathbb{N}\}$ of functions at the lowest level is said to be a Kleene-Shoenfield tower, if the function*

$$\mathbb{N}^k \times \Omega \rightarrow \mathcal{M}, \quad (n_k, \dots, n_1, x) \mapsto \Gamma_{n_k, \dots, n_1}(x)$$

is computable. Given a computational problem $\{\Xi, \Omega, \mathcal{M}, \Lambda\}$ as above, we will write $\text{SCI}(\Xi, \Omega, \mathcal{M}, \Lambda)_{\text{KS}}$ to denote the SCI with respect to a Kleene-Shoenfield tower.

We can now present the main theorem, linking the SCI and the Arithmetical Hierarchy, via Shoenfield's Limit Lemma [57] as well as the corollary.

Theorem 3.6 (The SCI and the Arithmetical Hierarchy). *Let $m \in \mathbb{Z}_+$ and recall the classes $\Sigma_m, \Pi_m, \Delta_m$ from the Arithmetical Hierarchy. If Ξ is Δ_{m+1} then there exists a Kleene-Shoenfield tower of algorithms of height m . Conversely, if $\text{SCI}(\Xi, \Omega)_{\text{KS}} = m$ then Ξ is Δ_{m+1} , but not Δ_m .*

Corollary 3.7 (The SCI can become arbitrarily large). *For every $k \in \mathbb{N}$ there exists a problem function Ξ on Ω with $\text{SCI}(\Xi, \Omega)_{\text{KS}} = k$.*

Note that the Δ_m sets in the Arithmetical Hierarchy are not the only known hierarchy that is a special case of the SCI Hierarchy, this is also the case for the Arithmetical Hierarchy of Real Numbers [68]. There is also a close connection to the Borel hierarchy [49] and the Weihrauch degrees [15, 16, 39].

3.3 Quantum Mechanics

A basic problem in the foundations of computing is: given a Schrödinger operator $H = -\Delta + V$ that is defined on some appropriate domain so that the spectrum is uniquely determined by the potential, can one compute the spectrum by using arithmetic operations with the point samples $V(x)$ where $x \in \mathbb{R}^d$ and then take limits? This problem has created vast amount of research for specific cases with real potential, and we can only cite a small subset [5, 13, 14, 25, 26, 29, 46], however, the general question has been open since the launch of quantum mechanics in the thirties. We now present a solution to this problem for large classes of

potentials. Note that our results allow for complete generality and allow for non-normal operators, ensuring that the theory includes non-Hermitian quantum mechanics [38] and resonances [69].

Consider the Schrödinger operator $H = -\Delta + V$ on some appropriate domain, let Ω be some set of potential functions and define $\Xi_1(V) = \text{sp}(-\Delta + V)$ as well as, for $\epsilon > 0$, let $\Xi_2(V) = \text{sp}_\epsilon(-\Delta + V)$.

We consider two main classes of potentials: bounded and unbounded with blowup at infinity. For the bounded case the domain $\mathcal{D}(-\Delta + V) = W^{2,2}(\mathbb{R}^d)$ (the standard Sobolev space). Given that the spectra are unbounded, let (\mathcal{M}, d_{AW}) denote the set of closed subsets of \mathbb{C} equipped with the *Attouch-Wets* metric (see [4]). Also, Λ will be the set of all evaluations $f_x : V \mapsto V(x)$, $x \in \mathbb{R}^d$ and all constant functions. We consider the following potential classes: $\Omega_1 = \{V : V \in L^\infty(\mathbb{R}^d) \cap \text{BV}_\phi(\mathbb{R}^d)\}$, where $\phi : [0, \infty) \rightarrow [0, \infty)$ is some increasing function, $\text{BV}_\phi(\mathbb{R}^d) = \{f : \text{TV}(f|_{[-a,a]^d}) \leq \phi(a)\}$, $f|_{[-a,a]^d}$ meaning f restricted to the box $[-a, a]^d$ and TV being the total variation of a function in the sense of Hardy and Krause (see [4]). Also, $\Omega_2 = \Omega_1 \cap \{V : -\Delta + V \in \mathcal{R}_g(L^2(\mathbb{R}^d))\}$ (recall $\mathcal{R}_g(L^2(\mathbb{R}^d))$ from Definition 3.2). In the case of unbounded potential we assume nonnegative θ_1, θ_2 such that $\theta_1 + \theta_2 < \pi$. Define

$$\Omega_3 = \{V \in C(\mathbb{R}^d) : \forall x \arg(V(x)) \in [-\theta_2, \theta_1], |V(x)| \rightarrow \infty \text{ as } x \rightarrow \infty\}.$$

In this case we define the operator H via the minimal operator h as: $H = h^{**}$, $h = -\Delta + V$, $\mathcal{D}(h) = C_c^\infty(\mathbb{R}^d)$. We then get the following result.

Theorem 3.8 (Schrödinger operators). *Given the above setup*

$$\begin{aligned} \textbf{Spectrum:} \quad \text{SCI}(\Xi_1, \Omega_1)_G &\leq \text{SCI}(\Xi_1, \Omega_1)_A \leq 2, & \text{SCI}(\Xi_1, \Omega_2)_G &= \text{SCI}(\Xi_1, \Omega_2)_A = 1, \\ & & \text{SCI}(\Xi_1, \Omega_3)_G &= \text{SCI}(\Xi_1, \Omega_3)_A = 1, \end{aligned}$$

$$\textbf{Pseudospectrum:} \quad \text{SCI}(\Xi_2, \Omega_1)_G \leq \text{SCI}(\Xi_2, \Omega_1)_A \leq 2, \quad \text{SCI}(\Xi_2, \Omega_3)_G = \text{SCI}(\Xi_2, \Omega_3)_A = 1,$$

Note that the question whether $\text{SCI}(\Xi_1, \Omega_1)_G = 2$ or not is still open. Moreover, there are of course other classes of potentials of interests that must be classified in terms of the SCI, however, many of them can probably be covered by the proof techniques from this paper.

3.4 Polynomial root finding with rational maps

Another crucial problem in computation has been the well known fact that Newton's method fails for the problem of polynomial root finding. This problem prompted S. Smale [59] to ask whether there exists an alternative to Newton's method, namely, a purely iterative generally convergent algorithm (see below). Smale asked: "*Is there any purely iterative generally convergent algorithm for polynomial zero finding ?*" This problem was settled by C. McMullen in [47] as follows: yes, if the degree is three; no, if the degree is higher (see also [48, 61]). However, in [30] P. Doyle and C. McMullen demonstrated a striking phenomenon: this problem can be solved in the case of the quartic and the quintic using several limits. In particular, they provide a construction such that, by using several rational maps and several limits, a root of the polynomial can be computed. In particular, a problem in Class III was established (this result was quoted as one of several reasons for McMullen receiving the Fields medal [61]). We demonstrate how this result can be put into the SCI framework.

Definition 3.9. *A purely iterative algorithm [59] is a rational map (i.e. it is a rational map of the coefficients of p) $T : \mathbb{P}_d \rightarrow \text{Rat}_m$, $p \mapsto T_p$ which sends any polynomial p of degree $\leq d$ to a rational function T_p of a certain degree m .*

An important example of a purely iterative algorithm is *Newton's method*. Furthermore, a purely iterative algorithm is said to be *generally convergent* if $\lim_{n \rightarrow \infty} T_p^n(z)$ exists for (p, z) in an open dense subset of

$\mathbb{P}_d \times (\mathbb{C} \cup \{\infty\})$, and the limit is a root of p . Here $T_p^n(z)$ denotes the n th iterate $T_p^n(z) = T_p(T_p^{n-1}(z))$ of T_p . For instance, Newton's method is generally convergent *only* when $d = 2$. Smale's question led to the definition below.

Definition 3.10 (Doyle-McMullen tower). *A (Doyle-McMullen) tower of algorithms is a finite sequence of generally convergent algorithms, linked together serially, so the output of one or more can be used to compute the input to the next. The final output of the tower is a single number, computed rationally from the original input and the outputs of the intermediate generally convergent algorithms.*

It can be shown that a Doyle-McMullen tower is a general tower as in Definition 2.4 with the slight change that the convergence holds only for an open dense set, thus the following can be formulated in terms of the SCI.

Theorem 3.11 (McMullen [47]; Doyle and McMullen [30]). *For \mathbb{P}_d there exists a generally convergent algorithm only for $d \leq 3$. Towers of algorithms exist additionally for $d = 4$ and $d = 5$ but not for $d \geq 6$.*

By the proof of this theorem one gets that the height of the tower is three, and thus the previous theorem can be formulated in terms of the SCI as follows. For $d \leq 3$, the SCI = 1, for $d = 4, 5$ one has $\text{SCI} \in \{2, 3\}$ while for $d \geq 6$ there is no tower: $\text{SCI} = \infty$. See [4] for details.

3.5 Inverse Problems

There is a vast literature on computing solutions to certain infinite-dimensional inverse problems in one limit, see [9, 12, 33, 34, 44] and references therein. However, as we demonstrate, it is impossible to compute the solution to a general infinite linear system in one limit (this is universal for all algorithms regardless of the operations allowed), yet it is possible in two. In particular, we seek solutions to problems of the form $Ax = b$ where $A \in \mathcal{B}_{\text{inv}}(\ell^2(\mathbb{N}))$, the class of bounded invertible operators, and $b \in \ell^2(\mathbb{N})$. In particular, $\Xi_1(A, b) = x$. We define the classes of pairs (A, b) by

$$\Omega_1 = \mathcal{B}_{\text{inv}}(\ell^2(\mathbb{N})) \times \ell^2(\mathbb{N}), \quad \Omega_2 = \mathcal{B}_{\text{inv},sa}(\ell^2(\mathbb{N})) \times \ell^2(\mathbb{N}), \quad \Omega_3 = (\mathcal{B}_{\text{inv}}(\ell^2(\mathbb{N})) \cap \mathcal{B}_f(\ell^2(\mathbb{N}))) \times \ell^2(\mathbb{N}),$$

where $\mathcal{B}_{\text{inv},sa}$ denotes invertible and self-adjoint, and we recall $\mathcal{B}_f(\ell^2(\mathbb{N}))$ from Definition 3.1. The metric space \mathcal{M} would simply be $\ell^2(\mathbb{N})$ and Λ the collection of mappings $\{g_{i,j}\}_{i \in \mathbb{N}, j \in \mathbb{Z}_+}$ where $g_{i,j} : (A, b) \mapsto \langle Ae_j, e_i \rangle$ for $j \in \mathbb{N}$ and $g_{i,0} : (A, b) \mapsto \langle b, e_i \rangle$. When considering Ω_2 the values $f(m)$ ($m \in \mathbb{N}$) shall be available to the algorithms as constant evaluation functions.

We also consider the problem of computing the norm of the inverse. In particular, let $\Omega_4 = \mathcal{B}(\ell^2(\mathbb{N}))$, Ω_5 the subset of self-adjoint operators, Ω_6 the subset of operators with dispersion bounded by an $f : \mathbb{N} \rightarrow \mathbb{N}$, and let $\Xi_2 : A \mapsto \|A^{-1}\|$, where $\|A^{-1}\| := \infty$ if A is not invertible. The metric space \mathcal{M} is \mathbb{R}_+ equipped with the metric $d(x, y) := \frac{|x-y|}{1+|x-y|}$, and extended in the obvious way at ∞ . We now have the following.

Theorem 3.12 (Inverse problems). *Given the above setup*

$$\begin{aligned} \text{Solving } Ax = y & : \text{SCI}(\Xi_1, \Omega_1)_G = \text{SCI}(\Xi_1, \Omega_1)_A = 2, & \text{SCI}(\Xi_1, \Omega_2)_G = \text{SCI}(\Xi_1, \Omega_2)_A = 2, \\ & \text{SCI}(\Xi_1, \Omega_3)_G = \text{SCI}(\Xi_1, \Omega_3)_A = 1, \\ \text{Compute } \|A^{-1}\| & : \text{SCI}(\Xi_2, \Omega_4)_G = \text{SCI}(\Xi_2, \Omega_4)_A = 2, & \text{SCI}(\Xi_2, \Omega_5)_G = \text{SCI}(\Xi_2, \Omega_5)_A = 2, \\ & \text{SCI}(\Xi_2, \Omega_6)_G = \text{SCI}(\Xi_2, \Omega_6)_A = 1. \end{aligned}$$

3.6 Impossibility of Error Control

A key concept in computations is error control. Given a computational problem $\{\Xi, \Omega, \mathcal{M}, \Lambda\}$ where we have $\text{SCI}(\Xi, \Omega, \mathcal{M}, \Lambda)_\alpha = k$ for some tower of algorithms of type α , and a tower of algorithms of height k ,

we want to control the convergence $\Gamma_{n_k} \rightarrow \Xi, \dots, \Gamma_{n_k, \dots, n_1} \rightarrow \Gamma_{n_k, \dots, n_2}$. For $\epsilon > 0$, how big do n_k, \dots, n_1 have to be so that $d(\Gamma_{n_k, \dots, n_1}(A), \Xi(A)) \leq \epsilon$, for all $A \in \Omega$? Unfortunately, such choices of n_k, \dots, n_1 may be impossible. More precisely, problems with SCI greater than one with respect to a General tower will never have error control.

Theorem 3.13 (No global error control). *Let $\{\Xi, \Omega, \mathcal{M}, \Lambda\}$ be a computational problem where we have $\text{SCI}(\Xi, \Omega, \mathcal{M}, \Lambda)_G \geq 2$. Suppose that there is a general tower of algorithms of height k , $\Gamma_{n_k}, \dots, \Gamma_{n_k, \dots, n_1}$ for $\{\Xi, \Omega, \mathcal{M}, \Lambda\}$. Then there do NOT exist integers $n_k = n_k(m), \dots, n_1 = n_1(m)$ (depending on m) such that $d(\Gamma_{n_k, \dots, n_1}(A), \Xi(A)) \leq \frac{1}{m}$, for all $A \in \Omega$ and for all $m \in \mathbb{N}$.*

A weaker concept than global error control is local error control: $\forall A \in \Omega$ and $\forall \epsilon > 0$, $\exists n_k, \dots, n_1$ such that $d(\Gamma_{n_k, \dots, n_1}(A), \Xi(A)) < \epsilon$. Indeed, the existence of n_k, \dots, n_1 is guaranteed by the definition of a Tower of algorithms. However, the integers n_k, \dots, n_1 cannot be computed as the next theorem demonstrates, and thus local error control is also impossible.

Theorem 3.14 (Local error control cannot be computed). *Given a computational problem $\{\Xi, \Omega, \mathcal{M}, \Lambda\}$ with $\text{SCI}(\Xi, \Omega, \mathcal{M}, \Lambda)_G \geq 2$, suppose that there is a general tower of algorithms $\Gamma_{n_k}, \dots, \Gamma_{n_k, \dots, n_1}$ of height k for the computational problem. Then, there does NOT exist a sequence $\{\tilde{\Gamma}_n\}$ of general algorithms $\tilde{\Gamma}_n : \Omega \rightarrow \mathbb{N}^k$ such that for any $A \in \Omega$, $d(\Gamma_{\tilde{\Gamma}_n(A)_k, \dots, \tilde{\Gamma}_n(A)_1}(A), \Xi(A)) < \frac{1}{n}$.*

3.7 Optimization and PDEs

The field of scientific computing is full of unexplored problems when it comes to the SCI. We will give a little sneak peak into work in progress and future developments.

ℓ^p optimization: Nonlinear optimization problems are popular for solving problems in sampling theory and imaging. As many of such problems come from applications such as medical imaging, the computational model is often done over the continuum rather than the discrete. A typical problem is, given an $A \in \mathcal{B}(l^2(\mathbb{N}))$ and $y \in l^2(\mathbb{N})$, compute

$$x \in \underset{\eta \in l^p(\mathbb{N})}{\text{argmin}} \|\eta\|_{l^p} \quad \text{subject to} \quad \|A\eta - y\| \leq \delta, \quad \delta \geq 0, \quad p \in [1, \infty). \quad (3.1)$$

In this case we may consider the computational problem $\{\Xi, \Omega, \mathcal{M}, \Lambda\}$ similar to what we have in Section 3.5, however, we would first consider the closed metric space $\{\eta : \|A\eta - y\| \leq \delta\}$ and let \mathcal{M} be the metric quotient space when identifying all the minimizers. Also, $\Xi = [x]$ where x is a minimizer and $[x]$ denotes the equivalence class corresponding to x . Note that if $\Omega = \Omega_1 \times \Omega_2$ where $\Omega_1 = \{A : A = A^*, A \text{ is compact}, \|A\| \leq M\}$ for some $M > 0$ and $\Omega_2 = \{y : y \in \text{Ran}(A)\}$, where $\text{Ran}(A)$ denotes the range of A , then it can be shown [37] that $\text{SCI}(\Xi, \Omega)_G \geq 2$.

Total Variation optimization: Another problem that is popular is the Total Variation Basis Pursuit. In particular, we are interested in finding the SCI of computing

$$g \in \underset{u \in \text{BV}(\tilde{\Omega})}{\text{argmin}} \text{TV}(u) \quad \text{subject to} \quad \int_{\tilde{\Omega}} (Tu - f)^2 dx \leq \delta, \quad (3.2)$$

where $\tilde{\Omega} \subset \mathbb{R}^d$ is some appropriate domain, $T : L^2(\tilde{\Omega}) \rightarrow L^2(\mathbb{R}^d)$ is some linear operator, f is some function, possibly not even in the range of T , $\text{BV}(\tilde{\Omega})$ denotes the set of functions that have bounded variations and

$$\text{TV}(u) = \sup \left\{ \int_{\tilde{\Omega}} u \text{div} v dx : v \in C_c^1(\tilde{\Omega}, \mathbb{R}^2), \|v\|_\infty \leq 1 \right\}$$

where $C_c^1(\tilde{\Omega}, \mathbb{R}^2)$ is the set of continuously differentiable vector functions of compact support contained in $\tilde{\Omega}$. Without going into details, one can also show that the $\text{SCI} \geq 2$ [37].

PDEs on unbounded domains: Given $H = -\Delta + V$, we want to compute

$$e^{-itH}\psi, \quad \psi \in L^2(\mathbb{R}^n), \quad t > 0. \quad (3.3)$$

If we consider ψ to be known, then solving (3.3) can be done by solving a PDE on an unbounded domain. This is just an example of a PDE on an unbounded domain, however, it is a completely open question on how the classification theory for these problems regarding the SCI will turn out. This is work in progress.

3.8 Conclusion: The Solvability Complexity Index Hierarchy

As we have established, the existence of Class III and the SCI implies that the set of problems in scientific computing form hierarchies according to the SCI. In particular, given certain types of tower of algorithms (we denote this type by α) we can define the the Solvability Complexity Index Hierarchy. We deliberately use the Δ_m notation from the Arithmetic Hierarchy, as these will be special cases when considering a Kleene-Schoenfield tower.

- (i) Δ_0^α is the set of problems with $\text{SCI} = 0$.
- (ii) Δ_1^α is the set of problems with $\text{SCI} \leq 1$, moreover one has error control and knows a bound on the error committed.
- (iii) Δ_2^α is the set of problems with $\text{SCI} \leq 1$, however, error control may not be possible.
- (iv) Δ_{m+1}^α is the set of problems with $\text{SCI} \leq m$.

Obviously $\Delta_0 \subset \Delta_1 \subset \Delta_2 \subset \dots$, moreover, these hierarchies typically do not collapse, i.e. $\Delta_{m+1} \setminus \Delta_m \neq \emptyset$ is the set of problems with $\text{SCI} = m$. The formal definition of these hierarchies can be found in [37].

Determining these hierarchies completely is probably a never ending task, as there will always be sub-problems with extra structure, where its classification in the SCI Hierarchy will be unknown. However, determining as much information about the hierarchy is of great importance in order to understand the complexity of the vast world of scientific computing.

3.9 Proof techniques

Getting upper and lower bounds on the SCI is not done by a universal recipe, however, we will give a short sketch on how to get upper and lower bounds on certain spectral problems.

Short sketch of proof of $\text{SCI}(\Xi_1, \Omega_3) \leq 2$ in Section 3.1. For $A \in \Omega_3$ define, for $m, n \in \mathbb{N}$,

$$\begin{aligned} \zeta_{m,n}^A(z) &:= \min \{k/m : k \in \mathbb{N}, \\ k/m &\geq \min\{\sigma_1(P_n(A - zI)P_m|_{\text{Ran}(P_m)}), \sigma_1(P_n(A^* - \bar{z}I)P_m|_{\text{Ran}(P_m)})\}\}, \end{aligned}$$

where σ_1 denotes the smallest singular value. Let $G^\delta(K) := (K + B_\delta(0)) \cap (\delta(\mathbb{Z} + i\mathbb{Z}))$, when $K \subset \mathbb{C}$ is compact, $h_\delta(y) := \min\{k\delta : k \in \mathbb{N}, g(k\delta) > y\}$, and finally the tower

$$\Gamma_{m,n}(A) := \Upsilon_{B_m(0)}^{1/m}(\zeta_{m,n}^A), \quad \Gamma_m(A) := \lim_{n \rightarrow \infty} \Gamma_{m,n}(A) \quad (3.4)$$

where $\Upsilon_K^\delta(\zeta)$ is defined as follows: For each $z \in G^\delta(K)$ let $I_z := B_{h_\delta(\zeta(z))}(z) \cap (\delta(\mathbb{Z} + i\mathbb{Z}))$. Also, let

$$M_z = \begin{cases} \{w \in I_z : \zeta(w) \leq \zeta(v) \forall v \in I_z\} & \zeta(z) \leq 1 \\ \emptyset & \zeta(z) > 1. \end{cases}$$

Now define $\Upsilon_K^\delta(\zeta) := \bigcup_{z \in G^\delta(K)} M_z$. It turn out that this yields a height two tower for Ξ_1 . \square

Short sketch of proof of $\text{SCI}(\Xi_1, \Omega_1) \geq 3$ in Section 3.1. The idea of the proof is as follows. We first analyze a decision problem and show that the SCI of this problem is ≥ 3 with respect to a general tower of algorithms. Then we show that if the SCI of the general spectral problem was < 3 with respect to a general tower of algorithms, then the SCI of the decision problem is < 3 reaching a contradiction. When considering the decision problem we will be working with a problem of the form $\Xi : \Omega \rightarrow \mathcal{M} := \{\text{Yes}, \text{No}\}$, where \mathcal{M} is equipped with the discrete metric. In particular, let Ω denote the collection of all infinite matrices $\{a_{i,j}\}_{i,j \in \mathbb{Z}}$ with entries $a_{i,j} \in \{0, 1\}$. We then ask the following question.

$$\Xi : \{a_{i,j}\}_{i,j \in \mathbb{Z}} \mapsto \left(\exists D \forall j \left(\left(\forall i \sum_{k=-i}^i a_{k,j} < D \right) \vee \left(\forall R \exists i \sum_{k=0}^i a_{k,j} > R \wedge \sum_{k=-i}^0 a_{k,j} > R \right) \right) \right)$$

(“there is a bound D such that every column has either less than D 1s or is two-sided infinite”).

Claim: $\text{SCI}(\Xi, \Omega)_G \geq 3$. To prove this claim we assume the existence of a general tower of algorithm of height 2, namely $\Gamma_{n,m} : \Omega \rightarrow \mathcal{M}$ and $\Gamma_n : \Omega \rightarrow \mathcal{M}$ such that $\lim_{m \rightarrow \infty} \Gamma_{n,m}(A) = \Gamma_n(A)$ and $\lim_{n \rightarrow \infty} \Gamma_n(A) = \Xi(A)$ for all $A \in \Omega$ and where $\Gamma_{n,m}$ is a general algorithm. By using a Baire Category type of argument one can construct inductively an element $B \in \Omega$ where $\Xi(B) = \text{No}$ however, $\Gamma_n(B) = \text{Yes}$ for infinitely many n . Hence, a contradiction.

The rest of the proof is about converting this result to a problem of computing spectra. First observe that, without loss of generality, we may identify $\Omega_1 = \mathcal{B}(l^2(\mathbb{N}))$ with $\Omega = \mathcal{B}(X)$, where $X = \bigoplus_{n=-\infty}^{\infty} X_n$ in the l^2 -sense and where $X_n = l^2(\mathbb{Z})$. Second, we consider sequences $a = \{a_i\}_{i \in \mathbb{Z}}$ over \mathbb{Z} with $a_i \in \{0, 1\}$, and define respective operators $B_a \in \mathcal{B}(l^2(\mathbb{Z}))$ with matrix representation $B_a = \{b_{k,i}\}$ by

$$b_{k,i} := \begin{cases} 1 : & k = i \text{ and } a_k = 0 \\ 1 : & k < i \text{ and } a_k = a_i = 1 \text{ and } a_j = 0 \text{ for all } k < j < i \\ 0 : & \text{otherwise.} \end{cases}$$

Then B_a is again a shift on a certain subset of the canonical basis elements and the identity on the other basis elements, hence we have the following possible spectra:

- $\text{sp}(B_a) \subset \{0, 1\}$ if $\{a_i\}$ has finitely many 1s.
- $\text{sp}(B_a) = \mathbb{T}$, the unit circle, if there are infinitely many $i > 0$ with $a_i = 1$ and infinitely many $i < 0$ with $a_i = 1$ (we say $\{a_i\}$ is two-sided infinite).
- $\text{sp}(B_a) = \mathbb{D}$, the unit disc, if $\{a_i\}$ has infinitely many 1s, but only finitely many for $i < 0$ or finitely many for $i > 0$ (we say $\{a_i\}$ is one-sided infinite in that case).

Next for a matrix $\{a_{i,j}\}_{i,j \in \mathbb{Z}}$ we define the operator $C := \bigoplus_{k=-\infty}^{\infty} B_k$ on X , where $B_k = B_{\{a_{i,k}\}_{i \in \mathbb{Z}}}$ corresponds to the column $\{a_{i,k}\}_{i \in \mathbb{Z}}$ in the above sense. Concerning its spectrum we have $\bigcup_{k \in \mathbb{Z}} \text{sp}(B_k) \subset \text{sp}(C) \subset \mathbb{D}$ since $\|C\| = 1$. Clearly, if one of the columns is one-sided infinite then $\text{sp}(C) = \mathbb{D}$. The same holds true if for every $k \in \mathbb{N}$ there is a finite column with at least k 1s. Otherwise (that is if there is a number D such that for every column it holds that it either has less than D 1s or is two-sided infinite) the spectrum $\text{sp}(C)$ is a subset of $\{0\} \cup \mathbb{T}$. Therefore if we had a height two tower $\{\Gamma_m\}, \{\Gamma_{m,n}\}$ for the computation of the spectrum of C or its counterpart in Ω_1 then

$$\begin{aligned} \tilde{\Gamma}_{m,n}(\{a_{i,j}\}) &:= \text{Yes if and only if } \Gamma_{m,n}(C) \cap B_{1/4}(1/2) = \emptyset \\ \tilde{\Gamma}_m(\{a_{i,j}\}) &:= \text{Yes if and only if } \Gamma_m(C) \cap B_{1/4}(1/2) = \emptyset \end{aligned} \tag{3.5}$$

would provide a height two tower for Ξ , contradicting the claim above. \square

References

- [1] M. Agrawal, N. Kayal, and N. Saxena. PRIMES is in P. *Ann. of Math. (2)*, 160(2):781–793, 2004.
- [2] W. Arveson. C^* -algebras and numerical linear algebra. *J. Funct. Anal.*, 122(2):333–360, 1994.
- [3] C. Beltran, Luis, and M. Pardo. Smale’s 17th problem: average polynomial time to compute affine and projective solutions. *J. Amer. Math. Soc.*, 22(2):363–385, 2009.
- [4] J. Ben-Artzi, A. C. Hansen, O. Nevanlinna, and M. Seidel. Can everything be computed? - On the Solvability Complexity Index and Towers of Algorithms. *arXiv:1508.03280v1*, 2015.
- [5] G. Beylkin and M. Mohlenkamp. Numerical operator calculus in higher dimensions. *Proc. Natl. Acad. Sci. USA*, 99(16):10246–10251, 2002.
- [6] L. Blum. Computing over the reals: where Turing meets Newton. *Notices Amer. Math. Soc.*, 51(9):1024–1034, 2004.
- [7] L. Blum, F. Cucker, M. Shub, S. Smale, and R. M. Karp. *Complexity and real computation*. Springer, New York, Berlin, Heidelberg, 1998.
- [8] L. Blum, M. Shub, and S. Smale. On a theory of computation and complexity over the real numbers: NP-completeness, recursive functions and universal machines. *Bull. Amer. Math. Soc. (N.S.)*, 21(1):1–46, 1989.
- [9] A. Böttcher. C^* -algebras in numerical analysis. *Irish Math. Soc. Bull.*, (45):57–133, 2000.
- [10] A. Böttcher, H. Brunner, A. Iserles, and S. P. Nørsett. On the singular values and eigenvalues of the Fox-Li and related operators. *New York J. Math.*, 16:539–561, 2010.
- [11] A. Böttcher, A. V. Chithra, and M. N. N. Namboodiri. Approximation of approximation numbers by truncation. *Integral Equations Operator Theory*, 39(4):387–395, 2001.
- [12] A. Böttcher and B. Silbermann. *Introduction to Large Truncated Toeplitz Matrices*. Universitext (Berlin. Print). Springer New York, 1999.
- [13] L. Boulton. Projection methods for discrete Schrödinger operators. *Proc. London Math. Soc. (3)*, 88(2):526–544, 2004.
- [14] A. Bove, G. Fano, G. Turchetti, and A. G. Teolis. Functional integration, Padé approximants, and the Schrödinger equation. *Journal of Mathematical Physics*, 16(2), 1975.
- [15] V. Brattka. Effective Borel measurability and reducibility of functions. *Mathematical Logic Quarterly*, 51(1):19–44, 2005.
- [16] V. Brattka, M. de Brecht, and A. Pauly. Closed choice and a uniform low basis theorem. *Annals of Pure and Applied Logic*, 163(8):968–1008, 2012.
- [17] V. Brattka, P. Hertling, and K. Weihrauch. A tutorial on computable analysis. In *New computational paradigms*, pages 425–491. Springer, New York, 2008.
- [18] M. Braverman. On the complexity of real functions. *Foundations of Computer Science, 2005. FOCS 2005. 46th Annual IEEE Symposium on*, pages 155–164, 2005.

- [19] M. Braverman and S. Cook. Computing over the reals: foundations for scientific computing. *Notices Amer. Math. Soc.*, 53(3):318–329, 2006.
- [20] P. Bürgisser and F. Cucker. Exotic quantifiers, complexity classes, and complete problems. *Foundations of Computational Mathematics*, (2):135–170.
- [21] P. Bürgisser and F. Cucker. On a problem posed by Steve Smale. *Ann. of Math. (2)*, 174(3):1785–1836, 2011.
- [22] P. Bürgisser and F. Cucker. *Condition - The Geometry of Numerical Algorithms.*, volume 349 of *Grundlehren der mathematischen Wissenschaften*. Springer, 2013.
- [23] S. Cook. The P versus NP problem. In *The millennium prize problems*, pages 87–104. Clay Math. Inst., Cambridge, MA, 2006.
- [24] S. A. Cook. The complexity of theorem-proving procedures. In *Proceedings of the Third Annual ACM Symposium on Theory of Computing*, STOC '71, pages 151–158, New York, NY, USA, 1971. ACM.
- [25] J. W. Cooley. An Improved Eigenvalue Corrector Formula for Solving the Schrödinger Equation for Central Fields. *Mathematics of Computation*, 15(76):pp. 363–374, 1961.
- [26] E. B. Davies. A hierarchical method for obtaining eigenvalue enclosures. *Math. Comp.*, 69(232):1435–1455, 2000.
- [27] E. B. Davies and M. Plum. Spectral pollution. *IMA J. Numer. Anal.*, 24(3):417–438, 2004.
- [28] P. Deift, L. C. Li, and C. Tomei. Toda flows with infinitely many variables. *J. Funct. Anal.*, 64(3):358–402, 1985.
- [29] T. Digernes, V. S. Varadarajan, and S. R. S. Varadhan. Finite approximations to quantum systems. *Rev. Math. Phys.*, 6(4):621–648, 1994.
- [30] P. Doyle and C. McMullen. Solving the quintic by iteration. *Acta Math.*, 163(3-4):151–180, 1989.
- [31] S. Feferman. About and around computing over the reals. In *Computability—Turing, Gödel, Church, and beyond*, pages 55–76. MIT Press, Cambridge, MA, 2013.
- [32] H. Friedman and R. Mansfield. Algorithmic procedures. *Trans. Amer. Math. Soc.*, 332(1):297–312, 1992.
- [33] K. Gröchenig, Z. Rzesotnik, and T. Strohmer. Convergence analysis of the finite section method and banach algebras of matrices. *Integral Equations and Operator Theory*, 67(2):183–202, 2010.
- [34] R. Hagen, S. Roch, and B. Silbermann. *C*-algebras and numerical analysis*, volume 236 of *Mono-graphs and Textbooks in Pure and Applied Mathematics*. Marcel Dekker Inc., New York, 2001.
- [35] A. C. Hansen. On the approximation of spectra of linear operators on Hilbert spaces. *J. Funct. Anal.*, 254(8):2092–2126, 2008.
- [36] A. C. Hansen. On the solvability complexity index, the n -pseudospectrum and approximations of spectra of operators. *J. Amer. Math. Soc.*, 24(1):81–124, 2011.
- [37] A. C. Hansen, B. Jennings, and M. Seidel. On the Solvability Complexity Index Hierarchy and the foundations of computational mathematics. *In preparation*.

- [38] N. Hatano and D. R. Nelson. Vortex pinning and non-hermitian quantum mechanics. *Phys. Rev. B*, 56(14):8651–8673, Oct 1997.
- [39] K. Higuchi and A. Pauly. The degree-structure of Weihrauch-reducibility. *Logical Methods in Computer Science*, 9(2), 2013.
- [40] R. M. Karp. Reducibility among combinatorial problems. In R. E. Miller and J. W. Thatcher, editors, *Complexity of Computer Computations*, The IBM Research Symposia Series, pages 85–103. Plenum Press, New York, 1972.
- [41] A. Kawamura and S. Cook. Complexity theory for operators in analysis. In *STOC'10—Proceedings of the 2010 ACM International Symposium on Theory of Computing*, pages 495–502. ACM, New York, 2010.
- [42] S. C. Kleene. Recursive predicates and quantifiers. *Trans. Amer. Math. Soc.*, 53:41–73, 1943.
- [43] K.-I. Ko. *Complexity theory of real functions*. Progress in Theoretical Computer Science. Birkhäuser Boston, Inc., Boston, MA, 1991.
- [44] M. Lindner. *Infinite matrices and their finite sections*. Frontiers in Mathematics. Birkhäuser Verlag, Basel, 2006.
- [45] M. Marletta. Neumann-dirichlet maps and analysis of spectral pollution for non-self-adjoint elliptic pdes with real essential spectrum. *IMA Journal of Numerical Analysis*, 2009.
- [46] M. Marletta and S. Naboko. The finite section method for dissipative operators. *Mathematika*, 60:415–443, 7 2014.
- [47] C. McMullen. Families of rational maps and iterative root-finding algorithms. *Ann. of Math. (2)*, 125(3):467–493, 1987.
- [48] C. McMullen. Braiding of the attractor and the failure of iterative algorithms. *Invent. Math.*, 91(2):259–272, 1988.
- [49] Y. N. Moschovakis. *Descriptive Set Theory*, volume 100 of *Studies in Logic and the Foundations of Mathematics*. North-Holland, 1980.
- [50] A. Mostowski. On definable sets of positive integers. *Fund. Math.*, 34:81–112, 1947.
- [51] E. Novak, I. H. Sloan, and H. Woźniakowski. Tractability of Approximation for Weighted Korobov Spaces on Classical and Quantum Computers. *Foundations of Computational Mathematics*, 4(2):121–156, 2004.
- [52] A. Pauly. Multi-valued functions in computability theory. In *How the world computes*, volume 7318 of *Lecture Notes in Comput. Sci.*, pages 571–580. Springer, Heidelberg, 2012.
- [53] M. B. Pour-El and J. I. Richards. *Computability in analysis and physics*. Perspectives in Mathematical Logic. Springer-Verlag, Berlin, 1989.
- [54] M. Seidel. On (N, ϵ) -pseudospectra of operators on Banach spaces. *J. Funct. Anal.*, 262(11):4916–4927, 2012.
- [55] M. Seidel and B. Silbermann. Finite sections of band-dominated operators – norms, condition numbers and pseudospectra. In *Operator theory, pseudo-differential equations, and mathematical physics*, volume 228 of *Oper. Theory Adv. Appl.*, pages 375–390. Birkhäuser/Springer Basel AG, Basel, 2013.

- [56] E. Shargorodsky. On the limit behaviour of second order relative spectra of self-adjoint operators. *J. Spectr. Theory*, 2013.
- [57] J. R. Shoenfield. On degrees of unsolvability. *Ann. of Math. (2)*, 69:644–653, 1959.
- [58] M. Shub and S. Smale. Complexity of Bézout’s theorem. I. Geometric aspects. *J. Amer. Math. Soc.*, 6(2):459–501, 1993.
- [59] S. Smale. The fundamental theorem of algebra and complexity theory. *Bull. Amer. Math. Soc. (N.S.)*, 4(1):1–36, 1981.
- [60] S. Smale. Complexity theory and numerical analysis. In *Acta numerica, 1997*, volume 6 of *Acta Numer.*, pages 523–551. Cambridge Univ. Press, Cambridge, 1997.
- [61] S. Smale. The work of Curtis T McMullen. In *Proceedings of the International Congress of Mathematicians I, Berlin*, Doc. Math. J. DMV, pages 127–132. 1998.
- [62] J. Traub, G. Wasilkowski, and H. Woźniakowski. *Information-based Complexity*. Academic Press, 1988.
- [63] L. N. Trefethen and M. Embree. *Spectra and pseudospectra*. Princeton University Press, Princeton, NJ, 2005. The behavior of nonnormal matrices and operators.
- [64] J. V. Tucker and J. I. Zucker. Computable functions and semicomputable sets on many-sorted algebras. In *Handbook of logic in computer science, Vol. 5*, volume 5 of *Handb. Log. Comput. Sci.*, pages 317–523. Oxford Univ. Press, New York, 2000.
- [65] A. M. Turing. On Computable Numbers, with an Application to the Entscheidungsproblem. *Proc. London Math. Soc.*, S2-42(1):230.
- [66] K. Weihrauch. *Computable analysis*. Texts in Theoretical Computer Science. An EATCS Series. Springer-Verlag, Berlin, 2000. An introduction.
- [67] A. Wigderson. P, NP and mathematics – a computational complexity perspective. *Proceedings of the International Congress of Mathematicians, Madrid, I*, 2006.
- [68] X. Zheng and K. Weihrauch. The arithmetical hierarchy of real numbers. In *Mathematical Foundations of Computer Science 1999*, volume 1672 of *Lecture Notes in Computer Science*, pages 23–33. Springer Berlin Heidelberg, 1999.
- [69] M. Zworski. Resonances in physics and geometry. *Notices Amer. Math. Soc.*, 46(3):319–328, 1999.