COMPUTABILITY OF LINEAR EQUATIONS

Vasco Brattka*
Theoretische Informatik 1, FernUniversität Hagen
58084 Hagen, Germany
Vasco.Brattka@FernUni-Hagen.de

Martin Ziegler**
Heinz Nixdorf Institute, University of Paderborn
33095 Paderborn, Germany
ziegler@uni-paderborn.de

Abstract Do the solutions of linear equations depend computably on their coefficients? Implicitly, this has been one of the central questions in linear algebra since the very beginning of the subject and the famous Gauß algorithm is one of its numerical answers. Today there exists a tremendous number of algorithms which solve this problem for different types of linear equations. However, actual implementations in floating point arithmetic keep exhibiting numerical instabilities for ill-conditioned inputs. This situation raises the question which of these instabilities are intrinsic, thus caused by the very nature of the problem, and which are just side effects of specific algorithms. To approach this principle question we revisit linear equations from the rigorous point of view of computability. Therefore we apply methods of computable analysis, which is the Turing machine based theory of computable real number functions. It turns out that, given the coefficients of a system of linear equations, we can compute the space of solutions, if and only if the dimension of the solution space is known in advance. Especially, this explains why there cannot exist any stable algorithms under weaker assumptions.

Track 1: Algorithms, Complexity and Models of Computation.

Keywords: Computable Analysis, Linear Equations.

*Work partially supported by DFG Grant BR 1807/4-1
**Work partially supported by DFG Grant Me 872/7-3

R. Baeza-Yates et al. (eds.), *Foundations of Information Technology in the Era of Network and Mobile Computing* © Springer Science+Business Media New York 2002

1. Introduction

In this paper we want to study computability properties of systems of linear equations $Ax = b$, where $A \in \mathbb{R}^{m \times n}$ is a real matrix, $b \in \mathbb{R}^m$ is a real vector and $x \in \mathbb{R}^n$ is a variable. Solving systems of linear equations *effectively*, that is, by means of some sort of computing machinery, has been a particularly prominent subject in both mathematics and computer science. Gauß, when inventing his now famous algorithm, considered real numbers as entities which, in one step, can be operated on exactly. This idea has been captured, in order to analyze the complexity of other numerical algorithms as well, by Blum, Cucker, Shub, and Smale's model of real number computation [1].

In contrast, numerical mathematics takes into account properties of floating point numbers by tracing propagation of rounding errors throughout the computational process. There, the input is presumed to be *exact* (a rational number of for example type `float`) and one asks for the required accuracy (e.g., `double`) for intermediate computations in order to obtain the result with desired precision (say, `float` again). Notice that, strictly speaking, no reals but only rational numbers are involved for input, processing, and output.

Numerical analysis provides a large number of algorithms to solve linear equations, such as Gauß' elimination algorithm, Cholesky's decomposition algorithm and many others. Unfortunately, the applicability of these algorithms is limited by their numerical instability and error analysis is a non-trivial topic of research (cf. [12] as a standard text on this topic).

In the present work, we consider a truly *real* model of computation based on approximation by sequences of rational numbers. This model takes into account that, in practice, input data like $1.6 \cdot 10^{-19}$ (negative charge, in Coulomb, of an electron) in fact is *not* exact but has been obtained, for instance, by a physical measurement (here: by R.A. Millican, 1911) together with some error bound (say, $\pm 0.1 \cdot 10^{-19}$). Improved experimental technique later yielded better approximations like $(1.6022 \pm 0.0001) \cdot 10^{-19}$; a further increase in precision can be expected for the future.

This observation is captured in a model of real number computation introduced by Alan Turing [9] and Grzegorczyk [4] which nowadays forms the basis of *Computable Analysis* [8, 6, 10]: A real number $x \in \mathbb{R}$ is fed into a computer by means of an infinite sequence of rationals q_n and corresponding error bounds ε_n such that $|x - q_n| \leq \varepsilon_n \to 0$. Upon this input, it *computes* result $y = f(x)$ if it outputs two corresponding rational sequences (p_n) and (δ_n) for $y \in \mathbb{R}$.

One advantage of this approach is that it allows to unify the development of algorithms and their stability analysis in a single model. And although real numbers are represented by approximations in this model, one can still consider them as entities to operate on algorithmically: The *feasible real RAM* model, introduced in [2], characterizes the Turing machine approach in terms of real RAMs.

Since Turing's seminal work, a vast number of publications have investigated of various problems over real numbers from areas ranging from Classical Analysis [8] over Geometry, Theory of Fractals [7, 5] to Physics [8, 11]. In fact, several of them turned out to be *un*computable with respect to this model and thus explained for numerical instabilities experienced in actual software implementations.

In the present work, we address one of the very basic problems in Linear Algebra: Solving systems of linear equations $Ax = b$. The set of solutions of such equations are exactly the *affine* subsets $L \subseteq \mathbb{R}^n$ and any such subset can equivalently be characterized by an affine basis. But are these characterizations also *computably* equivalent? In other words: Can some Turing Machine effectively convert between the following representations for an affine subspace $L \subseteq \mathbb{R}^n$:

a) matrix $A \in \mathbb{R}^{m \times n}$ and vector $b \in \mathbb{R}^m$, given by rational approximations and error bounds such that $L = \{x \in \mathbb{R}^n : Ax = b\}$;

b) affine basis $x_1, ..., x_d \in \mathbb{R}^n$ and $x_0 \in \mathbb{R}^n$, given by rational approx. and error bounds, such that $L = \{x_0 + \lambda_1 x_1 + ... + \lambda_d x_d : \lambda_i \in \mathbb{R}\}$;

c) distance function $d_L : \mathbb{R}^n \to \mathbb{R}, y \mapsto \inf\{||x - y|| : x \in L\}$, given by a "program" of d_L (or, equivalently, by approximating rational Weierstraß polynomials).

Converting from a) to b) or c) means finding all solutions x to $Ax = b$ in a more ore less explicit way, that is, '*solving*' the equation. Our main result shows that this conversion is possible, if and only if the dimension of the solution space is known in advance (that is, if $\text{rank}(A, b) = \text{rank}(A)$ is given as additional input information).

Since by Church's thesis the Turing machine model characterizes those functions which are realizable on physical machines, our results imply certain intrinsic limitations of algorithmic solutions of systems linear equations: there is no general algorithm which could be performed on a physical machine and which solves systems of linear equations without knowing the dimension of the solution space in advance. And actually, our conclusions are in best conformity with the practical knowledge in numerical analysis. But since numerical analysis does not use any formal model of computation, it was not before such a theoretical study that this heuristic knowledge on principal limitations could be expressed in form of concise theorems.

In a previous paper [13] we have started to link linear algebra to computable analysis and we have investigated the question in which sense the dimension of a linear subspace can be computed. The present article continues along this line. The following section contains a short introduction to computable analysis and our previous results. Section 3 contains the technical main part of the paper and discusses how certain types of information on linear subspaces can be computably translated into each other. Finally, Section 4 applies the results to linear equations in order to study their computability properties.

2. Computable Analysis and Linear Algebra

In this section we briefly present some basic notions from computable analysis and some direct consequences of well-known facts. We will use Weihrauch's representation based approach to computable analysis, the so-called *Type-2-Theory of Effectivity*, since it allows to express computations with real numbers, continuous functions and subsets in a highly uniform way. For a precise and comprehensive reference we refer the reader to [10]. Roughly speaking, a partial real number function $f :\subseteq \mathbb{R}^n \to \mathbb{R}$ is computable, if there exists a Turing machine which transfers each sequence $p \in \Sigma^\omega$ that represents some input $x \in \mathbb{R}^n$ into some sequence $F_M(p)$ which represents the output $f(x)$. Since the set of real numbers has continuum cardinality, real numbers can only be represented by infinite sequences $p \in \Sigma^\omega$ (over some finite alphabet Σ) and thus, such a Turing machine M has to compute infinitely long, but eventually it transfers each input sequence p into an appropriate output sequence $F_M(p)$. It is reasonable to allow only one-way output tapes for infinite computations since otherwise the output after finite time would give no information on the final result (because it could possibly be replaced later by the machine). It is straightforward how this notion of computability can be generalized to other sets X with a corresponding *representation*, that is a surjective partial mapping $\delta :\subseteq \Sigma^\omega \to X$.

Definition 1 (Computable functions) Let δ, δ' be representations of X, Y, respectively. A function $f :\subseteq X \to Y$ is called *(δ, δ')–computable*, if there exists some Turing machine M such that $\delta' F_M(p) = f\delta(p)$ for all $p \in \mathrm{dom}(f\delta)$.

Here, $F_M :\subseteq \Sigma^\omega \to \Sigma^\omega$ denotes the partial function, computed by the Turing machine M. It is straightforward how to generalize this definition to functions with several inputs and it can even be generalized to multi-valued operations $f :\subseteq X \rightrightarrows Y$, where $f(x)$ is a subset of Y instead of a single value. In this case we replace the condition in the definition above by $\delta' F_M(p) \in f\delta(p)$. We can also define the notion of *(δ, δ')–continuity* by replacing F_M by a continuous function $F :\subseteq \Sigma^\omega \to \Sigma^\omega$ (with respect to the Cantor topology on Σ^ω).

Already in case of the real numbers it appears that the defined notion of computability sensitively relies on the chosen representation of the real numbers. The theory of *admissible* representations completely answers the question how to find "reasonable" representations of topological spaces [10]. We will make no formal use of admissibility, and hence the reader may ignore all corresponding information. Let us just mention that for admissible representations δ, δ' each (δ, δ')–computable function is necessarily continuous (with respect to the final topologies of δ, δ').

An example of an admissible representation of the real numbers is the so-called *Cauchy representation* $\rho :\subseteq \Sigma^\omega \to \mathbb{R}$, where roughly speaking, $\rho(p) = x$ if p is an (appropriately encoded) sequence of rational numbers $(q_i)_{i \in \mathbb{N}}$ which converges rapidly to x, i.e. $|q_i - q_k| \leq 2^{-k}$ for all $i > k$. By standard coding techniques this representation can easily be generalized to a representation of the n-dimensional Euclidean space $\rho^n :\subseteq \Sigma^\omega \to \mathbb{R}^n$ and to a representation of $m \times n$ matrices $\rho^{m \times n} :\subseteq \Sigma^\omega \to \mathbb{R}^{m \times n}$. A vector $x \in \mathbb{R}^n$ or a matrix $A \in \mathbb{R}^{m \times n}$ will be called *computable*, if it has a computable ρ^n–, *$\rho^{m \times n}$–name*, i.e. if there exists a computable $p \in \Sigma^\omega$ such that $x = \rho^n(p)$ or $A = \rho^{m \times n}(p)$, respectively. A function $f :\subseteq \mathbb{R}^n \to \mathbb{R}$ is just called *computable*, if it is (ρ^n, ρ)–computable.

If δ and δ' are admissible representations of topological spaces X and Y, respectively, then there exists a canonical representation $[\delta, \delta'] :\subseteq \Sigma^\omega \to X \times Y$ of the product space $X \times Y$ and a canonical function space representation $[\delta \to \delta'] :\subseteq \Sigma^\omega \to C(X,Y)$ of the set $C(X,Y)$ of total continuous functions $f : X \to Y$. We mention that these representations allow evaluation and type conversion. Evaluation means that the evaluation function $C(X,Y) \times X \to Y$, $(f,x) \mapsto f(x)$ is $([[\delta \to \delta'], \delta], \delta')$-computable and type conversion means that a function $f : Z \times X \to Y$ is $([\delta'', \delta], \delta')$-computable, if and only if the canonically associated function $f' : Z \to C(X,Y)$ with $f'(z)(x) := f(z,x)$ is $(\delta'', [\delta \to \delta'])$-computable. As a direct consequence we obtain that matrices $A \in \mathbb{R}^{m \times n}$ can effectively be identified with linear mappings $f \in \mathrm{Lin}(\mathbb{R}^n, \mathbb{R}^m)$, see Proposition 2.1 and 2.2 below. Especially, a matrix A is computable, if and only if the corresponding linear mapping is a computable function.

To express weaker computability properties, we will use two further representations $\rho_<, \rho_> :\subseteq \Sigma^\omega \to \mathbb{R}$. Roughly speaking, $\rho_<(p) = x$ if p is an (appropriately encoded) list of all rational numbers $q < x$. (Analogously, $\rho_>$ is defined with $q > x$.) It is known that a mapping $f :\subseteq X \to \mathbb{R}$ is (δ, ρ)-computable, if and only if it is $(\delta, \rho_<)$- and $(\delta, \rho_>)$-computable [10]. The $(\rho^n, \rho_<)$-, and the $(\rho^n, \rho_>)$-computable functions $f : \mathbb{R}^n \to \mathbb{R}$ are called *lower* and *upper semi-computable*, respectively.

Moreover, we will also need a representation of the space $\mathcal{L}^n$ of linear subspaces $V \subseteq \mathbb{R}^n$. Since all linear subspaces are non-empty closed spaces, we can use well-known representations of the hyperspace $\mathcal{A}^n$ of all closed non-empty subsets $A \subseteq \mathbb{R}^n$ (cf. [3, 10]). One way to represent such spaces is via the distance function $d_A : \mathbb{R}^n \to \mathbb{R}$, defined by $d_A(x) := \inf_{a \in A} d(x,a)$, where $d : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ denotes the Euclidean metric of $\mathbb{R}^n$. Altogether, we define three representations $\psi^n, \psi^n_<, \psi^n_> :\subseteq \Sigma^\omega \to \mathcal{A}^n$. We let $\psi^n(p) = A$, if and only if $[\rho^n \to \rho](p) = d_A$. In other words, p encodes a set A w.r.t. ψ^n, if it encodes the distance function d_A w.r.t. $[\rho^n \to \rho]$. Analogously, let $\psi^n_<(p) = A$, if and only if $[\rho^n \to \rho_>](p) = d_A$ and let $\psi^n_>(p) = A$, if and only if $[\rho^n \to \rho_<](p) = d_A$. One can prove that $\psi^n_<$ encodes "positive" information about the set A (all open rational balls $B(q,r) := \{x \in \mathbb{R}^n : d(x,q) < r\}$ which intersect A can be enumerated), and $\psi^n_>$ encodes "negative" information about A (all closed rational balls $\overline{B}(q,r)$ which do not intersect A can be enumerated). The final topology induced by ψ^n on $\mathcal{A}^n$ is the so-called *Fell topology*. It is a known fact that a mapping $f :\subseteq X \to \mathcal{A}^n$ is (δ, ψ^n)-computable, if and only if it is $(\delta, \psi^n_<)$- and $(\delta, \psi^n_>)$-computable [10].

A closed set $A \subseteq \mathbb{R}^n$ is called *r.e., co-r.e.* or *recursive*, if it is empty or if there is a computable $p \in \Sigma^\omega$ such that $A = \psi^n_<(p)$, $A = \psi^n_>(p)$, $A = \psi^n(p)$, respectively. Thus, the non-empty r.e., co-r.e. or recursive subsets $A \subseteq \mathbb{R}^n$ are exactly those with upper, lower semi-computable or computable distance function $d_A : \mathbb{R}^n \to \mathbb{R}$, respectively and a closed set is recursive, if and only if it is r.e. and co-r.e. By duality, an open subset $U \subseteq \mathbb{R}^n$ is called *r.e., co-r.e.* or *recursive*, if and only if its complement $\mathbb{R}^n \setminus U$ is co-r.e., r.e. or recursive. Given a representation δ of X, we will say more generally that a subset $U \subseteq Y \subseteq X$

is *δ–r.e. open* in Y, if $\delta^{-1}(U)$ is r.e. open in $\delta^{-1}(Y)$. Here a set $A \subseteq B \subseteq \Sigma^\omega$ is called *r.e. open* in B, if there exists some computable function $f :\subseteq \Sigma^\omega \to \Sigma^*$ with $\mathrm{dom}(f) \cap B = A$. Intuitively, a set U is δ–r.e. open in Y, if and only if there exists a Turing machine which halts for an input $x \in Y$ given w.r.t. δ, if and only if $x \in U$. It is known that a set $U \subseteq \mathbb{R}^n$ is ρ^n–r.e. open in $\mathbb{R}^n$, if and only if it is r.e. open. If a set $U \subseteq X$ is δ–r.e. open in X, then we will say for short that it is *δ–r.e. open.*

We close this section with a short survey on computability results in linear algebra which have been established in our previous paper [13]:

Proposition 2 *Consider the following canonical mappings from linear algebra:*

1. $\mathrm{Lin}(\mathbb{R}^n, \mathbb{R}^m) \to \mathbb{R}^{m\times n}$ *is* $([\rho^n \to \rho^m], \rho^{m\times n})$*-computable,*
2. $\mathbb{R}^{m\times n} \to \mathrm{Lin}(\mathbb{R}^n, \mathbb{R}^m)$ *is* $(\rho^{m\times n}, [\rho^n \to \rho^m])$*-computable,*
3. $\ker : \mathbb{R}^{m\times n} \to \mathcal{A}^n$ *is* $(\rho^{m\times n}, \psi^n_>)$*-computable,*
4. $\mathrm{span} : \mathbb{R}^{m\times n} \to \mathcal{A}^m$ *is* $(\rho^{m\times n}, \psi^m_<)$*-computable,*
   but neither $(\rho^{m\times n}, \psi^m_>)$*-computable, nor -continuous,*
5. $\det : \mathbb{R}^{n\times n} \to \mathbb{R}$ *is* $(\rho^{n\times n}, \rho)$*-computable,*
6. $\mathrm{rank} : \mathbb{R}^{m\times n} \to \mathbb{R}$ *is* $(\rho^{m\times n}, \rho_<)$*-computable,*
   but neither $(\rho^{m\times n}, \rho_>)$*-computable, nor -continuous,*
7. $\dim :\subseteq \mathcal{A}^n \to \mathbb{R}$ *is* $(\psi^n_<, \rho_<)$*- and* $(\psi^n_>, \rho_>)$*-computable.*

3. Linear Subspaces and their Dimension

Considering the computability results about linear algebra known so far from Proposition 2, what can be said about linear equations? If we consider only homogeneous equations $Ax = 0$ in the first step, then we obtain the solution space $L = \ker(A)$ and we can deduce from Proposition 2.3 that there exists a Turing machine which takes A as input with respect to $\rho^{m\times n}$ and which computes the space of solutions with respect to $\psi^n_>$. Unfortunately, this type of "negative" information about the space of solutions is not very helpful; in general it does not even suffice to find a single point of the corresponding space (cf. [10]). Thus, it is desirable to obtain the "positive" information (i.e. a $\psi^n_<$–name) about the space of solutions too. On the other hand we can deduce from $\mathrm{rank}(A) = n - \dim\ker(A)$ and Proposition 2.6 and 2.7 that $\ker : \mathbb{R}^{m\times n} \to \mathcal{A}^n$ is not $(\rho^{m\times n}, \psi^n_<)$–continuous. In other words: without any additional input information, positive information about the solution space is not available in principle.

What kind of additional information could suffice to obtain positive information about the solution space? We will show that it is sufficient to know the dimension of the solution space, i.e. $\mathrm{codim}(A) = \dim\ker(A)$ in advance. More precisely, the following theorem states that given a linear subspace $V \subseteq \mathbb{R}^n$ with respect to $\psi^n_>$ and given its dimension $\dim(V)$, we can effectively find a $\psi^n_<$–name of V. The remaining part of this section will be devoted to the proof of the following theorem, separated in several lemmas.

Theorem 3 *There exists a Turing machine which on input of a linear subspace $V \subseteq \mathbb{R}^n$ and $d = \dim(V)$ with respect to $\psi^n_>$ and ρ, respectively, outputs V with respect to $\psi^n_<$, more precisely, the function*

$$f :\subseteq \mathcal{A}^n \times \mathbb{R} \to \mathcal{A}^n, (V,d) \mapsto V$$

with $\mathrm{dom}(f) := \{(V,d) \in \mathcal{A}^n \times \mathbb{R} : V \in \mathcal{L}^n$ *and* $d = \dim(V)\}$ *is* $([\psi^n_>, \rho], \psi^n_<)$*-computable.*

The main technical tool for the proof of this theorem is given in the following definition. Here and in the following $|x| := \sqrt{\sum_{i=1}^n |x_i|^2}$ denotes the *Euclidean norm* of $x = (x_1, ..., x_n) \in \mathbb{R}^n$.

Definition 4 *Let $W \subseteq \mathbb{R}^n$ be a linear subspace and $\varepsilon > 0$. Denote by*

$$W_\varepsilon := \bigcup_{w \in W} B(w, \varepsilon|w|) = \{x \in \mathbb{R}^n : (\exists w \in W)\ |x - w| < \varepsilon|w|\}$$

the relative blow-up *of W by factor ε with respect to Euclidean norm.*

The following Figure 1 shows the blow-up W_ε of a one-dimensional subspace $W \subseteq \mathbb{R}^3$ by factor $\varepsilon = 1/4$ together with a one-dimensional subspace $V \subseteq W_\varepsilon \cup \{0\}$. The first useful property of the blow-up is given in the following lemma, which roughly speaking states that each linear subspace is contained in an arbitrarily small blow-up of a linear subspace of the same dimension but with rational basis.

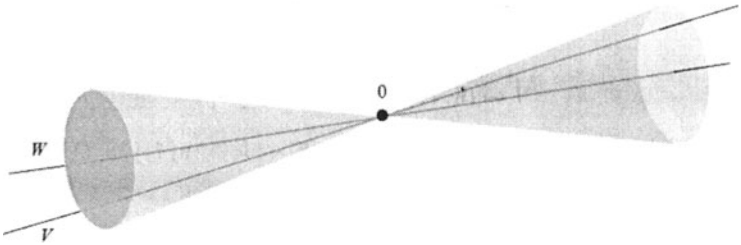

Figure 1. The blow-up W_ε of a linear subspace

Lemma 5 *Let $V \subseteq \mathbb{R}^n$ be a linear subspace of dimension d and $\varepsilon > 0$. Then there are $w_1, ..., w_d \in \mathbb{Q}^n$ such that $V \subseteq W_\varepsilon \cup \{0\}$, where $W := \mathrm{span}(w_1, ..., w_d)$.*

We leave the proof of this and the following three lemmas to the reader. Before we formulate the next property of the blow-up, we state an intermediate lemma about linear independence.

Lemma 6 *For each $n \geq 1$ there exists a constant $\Delta > 0$ such that, whenever $b_1, ..., b_d \in \mathbb{R}^n$ are pairwise orthogonal normed vectors and $x_1, ..., x_d \in \mathbb{R}^n$ with $|b_i - x_i| < \Delta$ for $i = 1, ..., d$, then $(x_1, ..., x_d)$ is linearly independent.*

From now on we assume without further mentioning that $\Delta < 1$ is a fixed rational constant as in the previous lemma (where we consider $n \geq 1$ to be arbitrary but fixed). The next lemma formulates another property of the blow-up which roughly speaking states that if a linear subspace V is contained in a sufficiently small blow-up of a linear subspace W of the same dimension, then this blow-up already approximates V quite well.

Lemma 7 *Let $V, W \subseteq \mathbb{R}^n$ be linear subspaces of equal dimension d and let $\varepsilon > 0$ with $\delta := 2\sqrt{d} \cdot \varepsilon/(1-\varepsilon) < \Delta$. If $V \subseteq W_\varepsilon \cup \{0\}$, then $B(w, \delta|w|)$ intersects V for any $w \in W \setminus \{0\}$.*

Now we formulate the last lemma of this section which states an effectivity property of the blow-up. Roughly speaking, the property $V \subseteq W_\varepsilon \cup \{0\}$ can be recognized by a Turing machine in a certain sense.

Lemma 8 *There exists a Turing machine which, on input of linear subspaces $V, W \subseteq \mathbb{R}^n$ with respect to representations $\psi^n_>$ and $\psi^n_<$ and $\varepsilon > 0$ halts, if and only if $V \subseteq W_\varepsilon \cup \{0\}$, more precisely*

$$\{(V, W, \varepsilon) \in \mathcal{A}^n \times \mathcal{A}^n \times \mathbb{R} : V \subseteq W_\varepsilon \cup \{0\} \text{ and } \varepsilon > 0\}$$

is $[\psi^n_>, \psi^n_<, \rho]$-r.e. open in $\mathcal{L}^n \times \mathcal{L}^n \times \mathbb{R}$.

Finally, we can combine Lemma 5, 7 and 8 to a proof of Theorem 3.

Proof of Theorem 3. Let $V \subseteq \mathbb{R}^n$ be a linear subspace and let $d = \dim(V) > 0$. We claim

$$\begin{aligned} B(q,r) \cap V \neq \emptyset \quad \Longleftrightarrow \quad & (\exists w_1, ..., w_d \in \mathbb{Q}^n)(\exists \lambda_1, ..., \lambda_d \in \mathbb{Q})(\exists \varepsilon > 0) \\ & \delta < \Delta,\ (w_1, ..., w_d) \text{ is linearly independent,} \\ & V \subseteq W_\varepsilon \cup \{0\} \text{ and } B(w, \delta|w|) \subsetneqq B(q,r), \\ & \text{where } W := \operatorname{span}(w_1, ..., w_d),\ w := \textstyle\sum_{i=1}^d \lambda_i w_i \neq 0 \\ & \text{and } \delta := 2\sqrt{d} \cdot \varepsilon/(1-\varepsilon) \end{aligned}$$

for all $q \in \mathbb{Q}^n$ and $r \in \mathbb{Q}$ with $r > 0$. By Lemma 7 it is clear that "$\Leftarrow$" holds. Let on the other hand $B(q,r) \cap V \neq \emptyset$ with $q \in \mathbb{Q}^n$ and $r \in \mathbb{Q}$ with $r > 0$. Then there exists some $v \in V \cap B(q,r)$, $v \neq 0$. Let $\delta(\varepsilon) := 2\sqrt{d} \cdot \varepsilon/(1-\varepsilon)$ for all $\varepsilon > 0$. Since $|q - v| < r$ there is some ε with $0 < \varepsilon < 1$ such that

$$\left(1 + \tfrac{\varepsilon + \delta(\varepsilon)}{1-\varepsilon}\right)|q - v| + \tfrac{\varepsilon + \delta(\varepsilon)}{1-\varepsilon}|q| < r.$$

Let $\delta := \delta(\varepsilon)$. By Lemma 5 there exist $w_1, ..., w_d \in \mathbb{Q}^n$ such that $V \subseteq W_\varepsilon \cup \{0\}$ with $W := \operatorname{span}(w_1, ..., w_d)$. Thus, there is some $w \in W \setminus \{0\}$ with $|v - w| < \varepsilon|w|$

and without loss of generality we can even assume that there are $\lambda_1, ..., \lambda_d \in \mathbb{Q}$ with $w = \sum_{i=1}^{d} \lambda_i w_i$. We obtain $|q - w| \leq |q - v| + |v - w| < |q - v| + \varepsilon|w|$ and $|w| \leq |q - w| + |q| \leq |q - v| + \varepsilon|w| + |q|$, and hence $|w| \leq 1/(1 - \varepsilon)(|q - v| + |q|)$ and thus

$$|q - w| + \delta|w| < |q - v| + (\varepsilon + \delta)|w| \leq \left(1 + \tfrac{\varepsilon+\delta}{1-\varepsilon}\right) |q - v| + \tfrac{\varepsilon+\delta}{1-\varepsilon}|q| < r,$$

i.e. $B(w, \delta|w|) \subsetneqq B(q, r)$. Thus, "$\Rightarrow$" holds too and the above equivalence is proved.

Thus, given V by $\psi^n_>$ and $d = \dim(V)$ by ρ, we can recursively enumerate all $q \in \mathbb{Q}^n$, $r \in \mathbb{Q}$ with $r > 0$ such that $B(q, r) \cap V \neq \emptyset$ by virtue of Lemma 8. In this way we obtain a $\psi^n_<$-name of V. □

Using Theorem 3 we can improve the statement of Corollary 1 in [13] in the following way.

Corollary 9 *The multi-valued mapping* basis $:\subseteq \mathcal{A}^n \times \mathbb{R} \rightrightarrows \mathcal{A}^n$,

$$(V, d) \mapsto \{\{b_1, ..., b_d\} \subseteq \mathbb{R}^n : (b_1, ..., b_d) \text{ is a basis of } V\}$$

with $\mathrm{dom(basis)} := \{(V, d) : d = \dim(V)\}$ *is* $([\psi^n_<, \rho], \psi^n)$*-computable and* $([\psi^n_>, \rho], \psi^n)$*-computable.*

Here, the $([\psi^n_<, \rho], \psi^n)$-computability of basis has been proved in [13] and the $([\psi^n_>, \rho], \psi^n)$-computability follows with Theorem 3. Roughly speaking, we can deduce that the following equivalences hold for different types of information about linear subspaces:

positive + dimension ≡ negative + dimension ≡ positive + negative ≡ basis

These equivalences could be made precise by defining corresponding representations of $\mathcal{L}^n$ and by proving their equivalence, but we are not going to discuss this here. Instead of that, we mention that for single linear subspaces one obtains the following less uniform corollary.

Corollary 10 *A linear subspace $V \subseteq \mathbb{R}^n$ is r.e., if and only if it is co-r.e., if and only if it is recursive, if and only if it admits a computable basis.*

Since the dimension is always a computable number, the proof of this corollary follows directly from the previous corollary and the fact that the mapping span $:\subseteq \mathbb{R}^{n \times d} \to \mathcal{A}^n$, restricted to linear independent inputs $(b_1, ..., b_d)$, is $(\rho^{n \times d}, \psi^n)$-computable, which has been proved in [13].

4. Linear Equations

In this section we want to apply the results of the previous section to solve linear equations $Ax = b$. It is a well-known and obvious fact from linear algebra that such a linear equation is solvable, if and only if $\mathrm{rank}(A) = \mathrm{rank}(A, b)$. The following theorem is the main result of this paper. It states that the solution operator of solvable linear equations is computable, provided that the rank of the linear equation is given as additional input.

Theorem 11 *There exists a Turing machine which takes a solvable linear equation $Ax = b$ together with $d = \mathrm{rank}(A, b)$ as input and which computes the space of solutions $L = \{x : Ax = b\}$. More precisely, the function*

$$\mathrm{solve} :\subseteq \mathbb{R}^{m\times n} \times \mathbb{R}^m \times \mathbb{R} \to \mathcal{A}^n, (A, b, d) \mapsto L = \{x \in \mathbb{R}^n : Ax = b\}$$

with $\mathrm{dom}(\mathrm{solve}) := \{(A, b, d) \in \mathbb{R}^{m\times n} \times \mathbb{R}^m \times \mathbb{R} : \mathrm{rank}(A) = \mathrm{rank}(A, b) = d\}$ *is* $([\rho^{m\times n}, \rho^m, \rho], \psi^n)$*-computable.*

Proof. Notice that $x \in L$, if and only if, in homogeneous coordinates, x is a solution to $(A, b) \cdot {}^t(x, -1) = 0$. We therefore may determine the kernel of $(A, b) \in \mathbb{R}^{m\times(n+1)}$ and scale the results x such that $x_{n+1} = -1$.

To realize this idea precisely, we perform several steps: let $A \in \mathbb{R}^{m\times n}$ be given by $\rho^{m\times n}$, let $b \in \mathbb{R}^m$ be given by ρ^m and let $d = \mathrm{rank}(A) = \mathrm{rank}(A, b)$ be given by ρ. First, we determine $\ker(A, b)$ w.r.t. $\psi^{n+1}_{>}$, which is possible by Proposition 2.3. Then we can use Theorem 3 and the formula $\dim\ker(A, b) = n + 1 - d$ to determine a $\psi^{n+1}_{<}$–name of $\ker(A, b)$. Especially, this name allows to find effectively a point $z = (z_1, ..., z_{n+1}) \in \ker(A, b)$ w.r.t. ρ^{n+1} such that $z_{n+1} < 0$. Let $c_i := z_i/|z_{n+1}|$ for $i = 1, ..., n$. Then $c := (c_1, ..., c_n)$ is a solution of $Ax = b$ and $L = \{x : Ax = b\} = c + \ker(A)$. Since $\dim\ker(A) = n - d$ we can compute a ψ^n–name of $\ker(A)$ by Proposition 2.3 and Theorem 3. Finally, we note that the function $\mathbb{R}^n \times \mathcal{A}^n \to \mathcal{A}^n, (x, A) \mapsto x + A := \{x + a \in \mathbb{R}^n : a \in A\}$ is $([\rho^n, \psi^n], \psi^n)$–computable. Altogether, this allows us to compute a ψ^n–name of L. □

Regarding the proof and Corollary 9 we can even conclude the following corollary, which states that given a solvable linear equation together with its rank we can effectively find a specific solution and a basis for the homogeneous equation.

Corollary 12 *The map* $s :\subseteq \mathbb{R}^{m\times n} \times \mathbb{R}^m \times \mathbb{R} \rightrightarrows \mathbb{R}^n \times \mathcal{A}^n$, $(A, b, d) \mapsto S$, *where* S *is the set*

$$\big\{(c, \{b_1, ..., b_{n-d}\}) \in \mathbb{R}^n \times \mathcal{A}^n : c + \mathrm{span}(b_1, ..., b_{n-d}) = \{x : Ax = b\}\big\},$$

and $\mathrm{dom}(s) := \{(A, b, d) \in \mathbb{R}^{m\times n} \times \mathbb{R}^m \times \mathbb{R} : \mathrm{rank}(A) = \mathrm{rank}(A, b) = d < n\}$, *is* $([\rho^{m\times n}, \rho^m, \rho], [\rho^n, \psi^n])$*-computable.*

Moreover, the previous theorem allows to deduce an immediate consequence about single linear equations.

Corollary 13 *If* $A \in \mathbb{R}^{m\times n}$ *is a computable matrix and* $b \in \mathbb{R}^m$ *a computable vector, then* $L = \{x \in \mathbb{R}^n : Ax = b\}$ *is a recursive set. If, additionally,* $Ax = b$ *has a unique solution* $x \in \mathbb{R}^n$*, then this solution is computable.*

It is interesting to note that our results also allow to handle the problem which is inverse to solving a linear equation: given an affine subspace, we can find a linear equation with this affine subspace as solution space.

Theorem 14 *There exists a Turing machine which takes an affine space L as input and computes a linear equation $Ax = b$ with $L = \{x : Ax = b\}$ as set of solutions. More precisely, the function* solve *admits a multi-valued right inverse $r :\subseteq \mathcal{A}^n \rightrightarrows \mathbb{R}^{m\times n} \times \mathbb{R}^m \times \mathbb{R}$ which is $(\psi^n, [\rho^{m\times n}, \rho^m, \rho])$-computable, for any $m \geq n$.*

Proof. Let L be given w.r.t. ψ^n. Then we can effectively find some point $c \in L$ w.r.t. ρ^n. As in the proof of Theorem 11 we can compute $L - c$ w.r.t. ψ^n. By Corollary 1 from [13] we can find a basis $(b_1, ..., b_k) \in \mathbb{R}^{n\times k}$ of $L-c$ w.r.t. $\rho^{n\times k}$. If $d := n - k = 0$, then $A = 0$ and $b = 0$ defines a linear equation with $L = \mathbb{R}^n$. Otherwise, apply the Gram-Schmidt orthogonalization process to determine an orthogonal basis $(o_1, ..., o_k)$ of $L - c$ w.r.t. $\rho^{n\times k}$, i.e.

$$o_1 := b_1,\ o_{j+1} := b_{j+1} - \sum\nolimits_{i=1}^{j} \frac{b_{j+1} \cdot o_i}{|o_i|^2} o_i$$

for $j = 1, ..., k - 1$. Then, find some vectors vectors $b_{k+1}, ..., b_n \in \mathbb{R}^n$ w.r.t. ρ^n such that $(o_1, ..., o_k, b_{k+1}, ..., b_n)$ is linear independent, which is possible by Lemma 4 in [13]. Then, apply the Gram-Schmidt orthogonalization process again to determine vectors $o_{k+1}, ..., o_n$ w.r.t. ρ^n such that $(o_1, ..., o_n)$ is an orthogonal basis of $\mathbb{R}^n$. Thus, $(o_{k+1}, ..., o_n)$ is an orthogonal basis of the orthogonal complement of $L - c$. Now, we can compute $A := {}^t(o_{k+1}, ..., o_n, 0, ..., 0) \in \mathbb{R}^{m\times n}$ w.r.t. $\rho^{m\times n}$ and $b := Ac$ w.r.t. ρ^m. Then we obtain $\ker(A) = L - c$ and $L = \{x : Ax = b\}$. Altogether, the procedure describes how to compute a right inverse r of the function solve. □

Again we can deduce a simple fact about single spaces and equations.

Corollary 15 *If $L \subseteq \mathbb{R}^n$ is a recursive non-empty affine subspace, then there exists a computable matrix $A \in \mathbb{R}^{m\times n}$ and a computable vector $b \in \mathbb{R}^m$ such that $L = \{x \in \mathbb{R}^n : Ax = b\}$ for any $m \geq n$.*

5. Conclusion

In this paper we have continued our project to investigate computability properties in linear algebra with rigorous methods from computable analysis. This project has been started with [13] and could be continued along several different lines. On the one hand, it would be interesting to extend the investigation to complexity questions. Surely, this is possible as long as one considers some non-uniform versions of our results. However, the fully uniform versions would require complexity measures for asymmetric hyperspaces which are beyond the state of the art. On the other hand, it is a promising topic to study other parts of linear algebra such as spectral theory or linear inequalities. Some steps in this direction have been presented in [14, 15].

Last but not least, our results give further ground to the hope that computable analysis can help to explain fundamental limitations of real number computations. Many practical observations of numerical analysis, e.g. the fact that numerical differentiation is much more difficult than numerical integration, already found natural explanations in computable analysis (see [10]). We have tried to extend these applications of computable analysis to linear algebra topics.

References

[1] L. Blum, F. Cucker, M. Shub, and S. Smale, *Complexity and Real Computation*, Springer, New York 1998.

[2] V. Brattka and P. Hertling, Feasible real random access machines, *J. Complexity* **14** (1998) 490–526.

[3] V. Brattka and K. Weihrauch, Computability on subsets of Euclidean space I: Closed and compact subsets, *Theoret. Comp. Sci.* **219** (1999) 65–93.

[4] A. Grzegorczyk, On the definitions of computable real continuous functions, *Fund. Math.* **44** (1957) 61–71.

[5] H. Kamo, K. Kawamura, and I. Takeuti, Computational complexity of fractal sets, *Real Analysis Exchange* **26** (2000/01) 773–793.

[6] K.-I. Ko, *Complexity Theory of Real Functions*, Birkhäuser, Boston 1991.

[7] K.-I. Ko, On the computability of fractal dimensions and Hausdorff measure, *Ann. Pure Appl. Logic* **93** (1998) 195–216.

[8] M. B. Pour-El and J. I. Richards, *Computability in Analysis and Physics*, Springer, Berlin 1989.

[9] A. M. Turing, On computable numbers, with an application to the "Entscheidungsproblem", *Proc. London Math. Soc.* **42** (1936) 230–265.

[10] K. Weihrauch, *Computable Analysis*, Springer, Berlin 2000.

[11] K. Weihrauch and N. Zhong, Turing computability of a nonlinear Schrödinger propagator, in: J. Wang (ed.), *Computing and Combinatorics*, vol. 2108 of *Lect. Not. Comp. Sci.*, Springer, Berlin 2001, 596–599.

[12] J. H. Wilkinson, *The Algebraic Eigenvalue Problem*, Oxford University Press, Oxford 1965.

[13] M. Ziegler and V. Brattka, Computing the dimension of linear subspaces, in: V. Hlaváč, K. G. Jeffery, and J. Wiedermann (eds.), *SOFSEM 2000: Theory and Practice of Informatics*, vol. 1963 of *Lect. Not. Comp. Sci.*, Springer, Berlin 2000, 450–458.

[14] M. Ziegler and V. Brattka, A computable spectral theorem, in: J. Blanck, V. Brattka, and P. Hertling (eds.), *Computability and Complexity in Analysis*, vol. 2064 of *Lect. Not. Comp. Sci.*, Springer, Berlin 2001, 378–388.

[15] M. Ziegler and V. Brattka, Turing computability of (non-)linear optimization, in: T. Biedl (ed.), *Thirteenth Canadian Conference on Computational Geometry*, University of Waterloo 2001, 181–184.