*Int. J. Appl. Math. Comput. Sci., 2022, Vol. 32, No. 1, 65–80*
DOI: 10.34768/amcs-2022-0006

# NON–STANDARD ANALYSIS REVISITED: AN EASY AXIOMATIC PRESENTATION ORIENTED TOWARDS NUMERICAL APPLICATIONS

VIERI BENCI [a], MARCO COCOCCIONI [b], LORENZO FIASCHI [b,*]

[a]Department of Mathematics
University of Pisa
Via Filippo Buonarroti, 1/c, Pisa, 56122, Italy
e-mail: benci@dma.unipi.it

[b]Department of Information Engineering
University of Pisa
L.go Lucio Lazzarino, 1, Pisa, 56122, Italy
e-mail: marco.cococcioni@unipi.it, lorenzo.fiaschi@phd.unipi.it

Alpha-Theory was introduced in 1995 to provide a simplified version of Robinson's non-standard analysis which overcomes the technicalities of symbolic logic. The theory has been improved over the years, and recently it has been used also to solve practical problems in a pure numerical way, thanks to the introduction of algorithmic numbers. In this paper, we introduce Alpha-Theory using a novel axiomatic approach oriented towards real-world applications, to avoid the need to master mathematical logic and model theory. To corroborate the strong link of this Alpha-Theory axiomatization and scientific computations, we report numerical illustrative applications never carried out by means of non-standard numbers within a computer, i.e., the computation of the eigenvalues of a non-Archimedean matrix, some computations related to non-Archimedean Markov chains, and the Cholesky factorization of a non-Archimedean matrix. We also highlight the differences between our numerical routines and pure symbolic approaches: as expected, the former scales better when the dimension of the problem increases.

**Keywords:** Alpha-Theory, non-standard analysis, non-Archimedean analysis, algorithmic numbers, non-Archimedean scientific computing.

## 1. Introduction

The purpose of this paper is to make a further contribution to the applications of non-Archimedean mathematics via numerical computations. In mathematics, *non-Archimedean* refers to an ordered field which does not satisfy the *axiom of Archimedes* (or, equivalently, that lacks the *Archimedean property*). The latter states (Deveau and Teismann, 2014) what follows.

**Axiom of Archimedes**. *Let $F$ be any totally ordered field. Then, $\forall\, x, y \in F$, $0 < x < y$, $\exists\, n \in \mathbb{N} : y < nx$.*

The axiom was called *Archimedean* by Otto Stolz, since it appears as Axiom V of Archimedes' *On the Sphere and Cylinder*.

*Corresponding author

Non-Archimedean analysis is the branch of mathematics which deals with fields lacking Archimedes' property; examples of non-Archimedean ordered fields are the Levi-Civita field (Levi-Civita, 1892), hyperreal numbers (Robinson, 1996), surreal numbers (Conway, 2000), or the Dehn field (Dehn, 1900). A very important branch of non-Archimedean analysis is non-standard analysis (NSA), originally proposed in 1961 by Robinson, who published a milestone book about it (Robinson, 1996). By now, there are many books on NSA; we refer the interested reader to those by Benci and Di Nasso (2018) or Keisler (1976) which are the books closer to our approach.

Actually, our approach is based on **Alpha-Theory** (AT) (Benci *et al*., 2006; Benci and Di Nasso, 2018) and the theory of **numerosity** (Benci and Di Nasso, 2003):

- AT is an introduction to non-standard analysis based on the notion of an $\alpha$-limit. The notion of the $\alpha$-limit is a version of the transfer principle easier to be used by practitioners; in fact, roughly speaking, it could be enunciated as follows: "*every relation between sequences is preserved by the limit.*"
- The theory of numerosity is strictly related to AT. It is useful to give a meaning to some infinite numbers such as the number $\alpha$ which is the numerosity of the set of positive natural numbers and it can be useful in some applications (e.g., see Benci *et al.*, 2018).

To improve the accessibility to the topic, in Section 2 we present Alpha-Theory in an axiomatic way, along with some basic examples. Then, in Section 3 we give a glimpse at the differences existing between field theory and its actual implementation within a computer, while in Section 4 we propose a way to numerically encode hyperreal numbers in a machine, similarly to what has been done for the reals with IEEE floating point numbers. Finally, in Section 5 we present some possible computations using hyperreal numbers. Section 6 and 7 contains some related works and concluding remarks.

## 2. Alpha-Theory

The essence of any axiomatic approach to non-standard fields relies on two points: (i) the existence of at least one infinite (or infinitesimal) number and its algebraic properties; (ii) the transfer principle. Actually, they correspond to Axioms 2 and 3 shown later in the section, respectively.

**Axiom 1.** *There exists an ordered field of $\mathbb{E} \supseteq \mathbb{R}$ whose numbers are called $\alpha$-Euclidean numbers.*

In the following we will refer to $\mathbb{E}$ as to the $\alpha$-Euclidean line.

Before introducing Axiom 2 and for a better reasoning, the following definition is needed. It introduces a partitioning of the set $\mathbb{E}$ into three categories: infinite, finite and infinitesimal numbers.

**Definition 1.** If $\xi \in \mathbb{E}$, then

- $\xi$ is infinite $\iff \forall\, n \in \mathbb{N}, |\xi| > n$,
- $\xi$ is finite $\iff \exists n \in \mathbb{N}\,, \frac{1}{n} < |\xi| < n$,
- $\xi$ is infinitesimal $\iff \forall\, n \in \mathbb{N}, |\xi| < \frac{1}{n}$.

Let $V(\mathbb{N})$ denote the superstructure on $\mathbb{N}$, namely,

$$V(\mathbb{N}) = \bigcup_{i=0}^{\infty} V_i(\mathbb{N}),$$

where

$$V_0(\mathbb{N}) = \mathbb{N}$$

and

$$V_{i+1}(\mathbb{N}) = V_i(\mathbb{N}) \cup \mathcal{P}(V_i(\mathbb{N}));$$

we set

$$\mathcal{U} := \{X \in V(\mathbb{N}) \,|X \text{ is countable}\}.$$

**Axiom 2.** *There exists a function* $\mathrm{num} : \mathcal{U} \to \mathbb{E}$ *which satisfies the following properties:*

- *if $A$ is finite,* $\mathrm{num}(A) = |A|$ *( here $|\cdot|$ denotes the cardinality of a set),*
- $\mathrm{num}(A) < \mathrm{num}(B)$ *if* $A \subset B$,
- $\mathrm{num}(A \cup B) = \mathrm{num}(A) + \mathrm{num}(B) - \mathrm{num}(A \cap B)$,
- $\mathrm{num}(A \times B) = \mathrm{num}(A) \cdot \mathrm{num}(B)$,
- $\alpha = \mathrm{num}(\mathbb{N})$.

Axiom 2 introduces the infinite number $\alpha$ and states that it can be manipulated as any other finite real number using field rules such as commutativity, associativity, etc. As an example, the following relations hold true within AT:

$$0 < \frac{1}{\alpha} = \alpha^{-1} < \alpha^0 = 1 < \alpha^1 = \alpha < (\alpha + 1),$$

$$\alpha \cdot (\alpha + 2) = \alpha^2 + 2\alpha,$$

$$\frac{-10.0\alpha^2 + 16.0 + 42.0\eta^2}{5.0\alpha^2 + 7.0} = -2.0 + 6.0\eta^2,$$

where we have set

$$\eta := \frac{1}{\alpha}.$$

Hence $\eta$ is an infinitesimal, being the reciprocal of $\alpha$. Now we will introduce axiomatically the notion of the $\alpha$-limit:

**Axiom 3.** *Every sequence $\varphi : \mathbb{N} \to \mathbb{R}$ has a unique $\alpha$-limit, denoted by $\lim_{n\uparrow\alpha} \varphi(n)$, which satisfies the following properties:*

1. *if $\xi \in \mathbb{E}$, then there exists a sequence $\varphi : \mathbb{N} \to \mathbb{R}$ such that*
$$\xi = \lim_{n\uparrow\alpha} \varphi(n);$$
2. *if $\varphi(n) = n$, then*
$$\lim_{n\uparrow\alpha} \varphi(n) = \alpha;$$
3. *if, eventually, $\varphi(n) \geq \psi(n)$ (namely, $\exists n_0 \in \mathbb{N}$ such that $\forall n \geq n_0,\ \varphi(n) \geq \psi(n)$), then*
$$\lim_{n\uparrow\alpha} \varphi(n) \geq \lim_{n\uparrow\alpha} \psi(n)$$

*4. for any sequences $\varphi$ and $\psi$*

$$\lim_{n\uparrow\alpha}\varphi(n)+\lim_{n\uparrow\alpha}\psi(n)=\lim_{n\uparrow\alpha}(\varphi(n)+\psi(n)),$$
$$\lim_{n\uparrow\alpha}\varphi(n)\cdot\lim_{n\uparrow\alpha}\psi(n)=\lim_{n\uparrow\alpha}(\varphi(n)\cdot\psi(n)).$$

Notice that in order to distinguish the usual limit of a sequence (which we will call the *Cauchy limit*) from the $\alpha$-limit, we use the symbols "$n\to\alpha$" and "$n\uparrow\alpha$", respectively. Points 1–3 are not surprising since we expect them to be satisfied by any notion of limit provided the target space is equipped with a reasonable topology. In Axiom 3, the new (and, maybe for someone, surprising) fact is that every net has an $\alpha$-limit. Nevertheless, Axiom 3 is not contradictory and a model for it can be constructed in ZFC (cf. Benci and Di Nasso, 2018).

The first consequence of the $\alpha$-limit is that every real function $f:\mathbb{R}\to\mathbb{R}$ can be extended to a function $f^*:\mathbb{E}\to\mathbb{E}$ by setting

$$f^*\left(\lim_{n\uparrow\alpha}\varphi(n)\right)=\lim_{n\uparrow\alpha}f(\varphi(n)).$$

It is not difficult to prove that this is a good definition, that is, $f^*(\xi)$ does not depend on the sequence $\varphi(n)$ which defines $\xi$. In what follows, when no ambiguity is possible, we will omit the symbol "$*$" and, therefore, $f$ and $f^*$ will be denoted by the same symbol.

As we remarked in the introduction, Axiom 3 can be seen as a weak form of the *Transfer Principle*. For example, suppose that you want to transfer the following property of trigonometric functions:

$$\sin(2x)=2\cos x\cdot\sin x$$

to their extension over $\mathbb{E}$. If we take a generic point $\xi\in\mathbb{E}$, by Axiom 3 there exists a sequence $\varphi$ such that

$$\xi=\lim_{n\uparrow\alpha}\varphi(n).$$

Since $\varphi(n)\in\mathbb{R}$, we have that

$$\sin(2\varphi(n))=2\cos\varphi(n)\cdot\sin\varphi(n).$$

We can take the $\alpha$-limit of both the sides

$$\lim_{n\uparrow\alpha}\sin(2\varphi(n))=\lim_{n\uparrow\alpha}[2\cos\varphi(n)\cdot\sin\varphi(n)]$$

and use the properties of the $\alpha$-limit to get

$$\lim_{n\uparrow\alpha}\sin(2\xi)=\lim_{n\uparrow\alpha}[2\cos\xi\cdot\sin\xi].$$

Now, let us see a basic theorem in non-Archimedean mathematics. The theorem states that any non-infinite hyper-real number is infinitely close to only one real number, namely its *standard part*, and there exists a function which associates the former to the latter. Such a function is defined from all the non-infinite values of $\mathbb{E}$ (denoted $\mathbb{E}_{\text{fin}}$) to $\mathbb{R}$ and it is denoted by $st$.

**Theorem 1.** (Standard part) *There exists a function $st:\mathbb{E}_{fin}\to\mathbb{R}$ satisfying*

- $x\sim y\Rightarrow st(x)=st(y)$,
- $st(st(x))=st(x)$,

*where $\sim$ stands for "is infinitely close to", i.e., $x-y$ is an infinitesimal number.*

*Proof.* See, for example, the work of Keisler (1976) or Benci and Di Nasso (2018). □

As an example, the following equations hold true:

$$st(2.1-5\eta)=2.1,\qquad st(-3)=-3,$$
$$st(-\sqrt{2}\eta^2+\pi\eta^3)=0,\qquad st(\alpha)=\nexists.$$

Put in another way, the standard part function maps any number into its closest real.

The standard part of a number is related to the Cauchy limit in the following way. If a sequence $\varphi(n)$ admits the Cauchy limit, the relation with the $\alpha$-limit is given by the following identity:

$$\lim_{n\to\alpha}\varphi(n)=st\left(\lim_{n\uparrow\alpha}\varphi(n)\right).\tag{1}$$

Another important relation between the two limits is the following: if

$$\lim_{n\uparrow\alpha}\varphi(n)=\xi\text{ is not infinite},$$

then there exists a subsequence $\varphi(n_k)$ of $\varphi(n)$ such that

$$\lim_{k\to\infty}\varphi(n_k)=st(\xi).$$

For the consistence of the axioms and the proofs of the facts claimed in this section, we refer to Benci and Di Nasso (2018).

### 2.1. Is $\alpha$ even or odd? And other similar questions.

Until now, we said almost nothing about the number $\alpha$ except that it is a positive infinite number. Therefore, it is reasonable that one may wonder about additional properties of $\alpha$, e.g., whether it is even or odd. The answer can be quite surprising at first sight, but it is perfectly consistent: it can be either one or the other indifferently, namely this statement cannot be deduced by Axioms 1–3. In fact, there are many models which satisfy them. Most of these questions are irrelevant for the applications and, in particular, for the applications of this paper. However, sometimes it can be useful to add some other axioms which enrich the theory. For example, Benci and Di Nasso (2018), it is proved that the theory can be implemented to get the following result:

**Theorem 2.** *The number $\alpha$ satisfies the following properties:*

- DIVISIBILITY PROPERTY: *For every $k \in \mathbb{N}$, the number $\alpha$ is a multiple of $k$ and the numerosity of the set of multiples of $k$:*

$$\mathfrak{num}(\{k, 2k, 3k, \dots, nk, \dots\}) = \frac{\alpha}{k}.$$

- ROOT PROPERTY: *For every $k \in \mathbb{N}$, the number $\alpha$ is a $k$-th power and the numerosity of the set of $k$-th powers:*

$$\mathfrak{num}(\{1^k, 2^k, 3^k, \dots, n^k, \dots\}) = \sqrt[k]{\alpha}.$$

- POWER PROPERTY: *If we set $\mathcal{P}_{fin}(A) = \{F \in \mathcal{P}(A) \mid F \text{ is a finite set}\}$, then*

$$\mathfrak{num}(\mathcal{P}_{fin}(\mathbb{N}^+)) = 2^\alpha.$$

- INTEGER NUMBERS PROPERTY:

$$\mathfrak{num}(\mathbb{Z}) = 2\alpha + 1.$$

- RATIONAL NUMBERS PROPERTY: *For every $q \in \mathbb{Q}$*

$$\mathfrak{num}(\mathbb{Q}) = 2\alpha^2 + 1$$

*and*

$$\mathfrak{num}((q, q+1] \cap \mathbb{Q}) = \mathfrak{num}((0, 1] \cap \mathbb{Q}) = \alpha.$$

*Proof.* See the work of Benci and Di Nasso (2018, Sections 16.6 and 16.7). □

## 3. Concept of the algorithmic field

$\mathbb{R}$ is a field and it is the reference number set for the majority of the theories about data science, machine learning, market analysis, etc. However, when considering problems from a numerical perspective, one immediately realizes that $\mathbb{R}$ might not be suitable for computation. Roughly speaking, it is "too rich" to be entirely managed by a finite machine. In fact, any computation between real numbers is carried out using floating point numbers, either in single or double precision. Floating point numbers are a finite-dimension encoding of real numbers and are defined in the IEEE 754 standard. They provide a very accurate approximation of a subset of $\mathbb{R}$ which is used in the majority of practical studies.

The set of all the numbers which can be represented exactly within a machine will be improperly referred to as the *algorithmic field*, stressing that those are the only numbers which are actually being crunched by algorithms and hardware accelerators. Thus they are a denumerable subset of real numbers or hyperreal numbers. Notice that even if the algorithmic field is not closed with respect to any algebraic operation, even if it is a discrete set, and even if computations doing on it may suffer by numerical instabilities, it turns out to work well for a large class of practical problems. Table 1 provides the mathematical fields and the algorithmic fields considered in this work.

### 3.1. Importance of fixed-length representations.

In symbolic computations, such as those performed by Wolfram Mathematica®, the inner representation of numbers is often variable-size, which typically slows the computations a lot. Even worse, when using iterative schemes at each iteration the inner representation of numbers typically grows in length, which makes the program slower and slower as it runs. On the contrary, working with fixed-length representations typically leads to faster code, because from the beginning to the end of the execution each number has always the same fixed length (which means it always requires the same time to be processed, it avoids checking its length at run-time in order to execute correct and in-bounds operations, etc). In addition, fixed-length representations are the only ones suitable for building hardware accelerators.

As an example, the floating point representation of real numbers in our computers (IEEE 754 standard recalled above) enjoys the hardware speedup of its operations thanks to the FPU co-processor most CPUs are equipped with (to be precise, modern CPUs have more than one FPU accelerator on each core). In the same spirit, in the next section we present the concept of algorithmic numbers (Benci and Cococcioni, 2020), a fixed-length representation for non-Archimedean numbers. In the near future, hardware accelerators for speeding up operations between them could be easily built, since there are no technical difficulties preventing it.

## 4. Algorithmic numbers

Algorithmic numbers (ANs) were introduced by Benci and Cococcioni (2020). They consist of a subset of the numbers of $\mathbb{E}$ which can be better standardized, and, therefore, easily manipulated on a computer. The definition of the AN, which follows, is inspired by the work of Levi-Civita (1892) and leverages on the concept of *monosemium*, which can be identified with a number of the form $r\alpha^p$, where $r \in \mathbb{R}$ and $p \in \mathbb{Q}$.

**Definition 2.** (*Algorithmic number*) A number $\xi \in \mathbb{E}$ is called *algorithmic* if it can be represented as a finite sum of monosemia, namely,

$$\xi = \sum_{k=0}^{\ell} r_k \alpha^{s_k}; \quad r_k \in \mathbb{R}, \quad s_k \in \mathbb{Q}; \quad s_k > s_{k+1}. \tag{2}$$

Moreover, one can always represent it in the following form, called "normal form":

$$\xi = \alpha^p P\left(\eta^{\frac{1}{m}}\right),$$

where

$$\eta := \alpha^{-1},$$

$p \in \mathbb{Q}$, $m \in \mathbb{N}$ and $P(x)$ is a polynomial with real coefficients such that $P(0) \neq 0$. A parallelism with the

Table 1. Mathematical fields vs. algorithmic fields.

| | Mathematical fields | Algorithmic fields |
|---|---|---|
| Real numbers | $\mathbb{R}$ | $\hat{\mathbb{R}}$ (approximated, not always associative, etc.) |
| Non-standard numbers | $\mathbb{E}$ | $\hat{\mathbb{E}}$ (approximated, not always associative, etc.) |

scientific notation for real numbers may help in suggesting the uniqueness of the representation. As an example, consider the number $1.3675e3$: the term $e3$ means $10^3$ and plays the same role as $\alpha^p$, while the number 1.3675 can be represented by a polynomial of non-positive powers of 10, i.e., $1 \cdot 10^0 + 3 \cdot 10^{-1} + 6 \cdot 10^{-2} + \dots$. Since a polynomial in $\eta$ is a polynomial in non-positive powers of $\alpha$, the parallelism is completed.

In particular, two issues rise when one tries to deal with ANs within a computer:

- the inverse of an AN is not always an AN, e.g., $(\alpha + 1)^{-1}$ is not an AN;
- they have a variable length coding, implying the problems discussed in Section 3.

To overcome these drawbacks, a notion of truncation is needed.

The truncation of a generic AN $\xi$ affects both the number of monosomic used to build it and the fixed length representation of the coefficients $r_i$, $i = 0, \dots, \ell$. Nevertheless, the second topic is negligible since there is abundant literature about it, and a very efficient mechanism to handle it already exists. Concerning the first one, it reduces to the truncation of the polynomial $P(\cdot)$ of its normal form. To tackle it, it is enough to define the following truncation function applied to a generic polynomial $P(x) = p_0 x^{z_0} + \dots + p_m x^{z_m}$, $z_{i-1} < z_i$, $i = 1, \dots, m$:

$$\mathfrak{tr}_n\left[P\left(x\right)\right] = \begin{cases} P(x), & n \geq m, \\ p_0 x^{z_0} + \dots + p_n x^{z_n}, & n < m. \end{cases}$$

The encoding of an AN truncated at $n$ is referred to as AN$n$. For example, if the machine precision allows one to set $n$ at most to 3, then the encoding used for representing ANs in an experiment on that machine is AN3.

**4.1. Bounded ANs: A special case of ANs.** There is a particular subset of ANs which deserves particular attention: bounded algorithmic numbers (BANs). A BAN is defined by the normal form $\alpha^p P(\eta)$, where $p \in \mathbb{Z}$ and $P(0) \neq 0$. Below, we report the approximated algebraic operations between truncated approximations of two BANs, namely $\xi = \alpha^p P(\eta)$ and $\zeta = \alpha^q Q(\eta)$. Notice that the order of truncation $n$ is an arbitrary natural value specified at compile time. In general, the user tunes it taking into consideration several aspects, e.g., the required precision of the computations, the computation time available, the architectural properties of the machine and so on.

**Sum** (assuming $p \geq q$):

$$\begin{aligned} \xi + \zeta &= \alpha^p P\left(\eta\right) + \alpha^p \left(Q(\eta)\eta^{p-q}\right) \\ &= \alpha^p \left(P(\eta) + \mathfrak{tr}_n\left[Q(\eta)\eta^{p-q}\right]\right) \end{aligned}$$

**Product**:

$$\xi\zeta = \alpha^{p+q}\mathfrak{tr}_n\left[P(\eta) \cdot Q(\eta)\right]$$

**Division**: Rewriting $\zeta$ as

$$\zeta = \alpha^q \left(q_0 + \sum_{k=1}^{n} q_k \eta^k\right) = q_0 \alpha^q \left(1 - \varepsilon\right),$$

where

$$\varepsilon = -\sum_{k=1}^{n} \frac{q_k}{q_0}\eta^k,$$

we get the definition of the division

$$\begin{aligned} \frac{\xi}{\zeta} &= \alpha^{p-q}\mathfrak{tr}_n\left[\frac{P\left(\eta\right)}{q_0}\left(1 + \varepsilon + \varepsilon^2 + \dots + \varepsilon^n\right)\right] \\ &= \alpha^{p-q}\left(\frac{P\left(\eta\right)}{q_0} + \mathfrak{tr}_n\left[\varepsilon\frac{P\left(\eta\right)}{q_0}\right] + \dots \right. \\ &\quad \left. + \mathfrak{tr}_n\left[\varepsilon^n\frac{P_n\left(\eta\right)}{q_0}\right]\right). \end{aligned}$$

Further details about these and other operations, such as the square root of a generic AN, can be found in the original work (Benci and Cococcioni, 2020), while in the remainder of this section we focus more on some delicate and technical aspects of evaluating transcendental functions of ANs.

**4.2. How to compute trigonometric functions on ANs.** The main ingredients to compute trigonometric functions involving ANs are Taylor series and trigonometric identities. For instance, consider trigonometric functions of monosemia such as $3\eta$. If we work in AN$n$, one can immediately make use of the Taylor expansion of the cosine in a neighborhood of zero truncated to the order $n$. For example, if $n = 3$, we have that

$$\begin{aligned} \cos(3\eta) &= 1 - \frac{1}{2}\left(3\eta\right)^2 + O(\eta^4) \\ &\simeq 1 - \frac{1}{2}\left(3\eta\right)^2 = 1 - \frac{9}{2}\eta^2. \end{aligned}$$

The issue to deal with multiple monosomic rather than just one comes quite straightforwardly with the help of simple trigonometric identities, such as the sine of the sum of two angles. They allow us to rewrite trigonometric functions of generic ANs as sums and products of trigonometric functions of monosemia. As an example, consider computation of the sine of $5 - 3\eta$. By means of the trigonometric identity $\sin(x + y) = \sin x \cos y - \sin y \cos x$, we can rewrite the computation as

$$\sin(5 - 3\eta) = \sin 5 \cos(3\eta) - \sin(3\eta) \cos 5$$

and then apply the Taylor series approximation as done before, obtaining

$$\begin{aligned}\sin(5 - 3\eta) &\simeq \sin 5 \left(1 - \frac{1}{2}(3\eta)^2\right) \\ &\quad - \left(3\eta - \frac{1}{3!}(3\eta)^3\right) \cos 5 \\ &= \sin 5 - 3\eta \cos 5 \\ &\quad - \frac{9}{2}\eta^2 \sin 5 + \frac{9}{2}\eta^3 \cos 5,\end{aligned}$$

where "$\simeq$" means that the numbers are approximated up to the third order. Finally, we remark that it is not possible to compute $\sin(\alpha)$ or $\cos(\alpha)$ since their values are not determined by Axioms 1–3. Consequently, we can start the same discussion as in Section 2.1.

**4.3. How to compute exponential and logarithmic functions.** In the case of exponential and logarithmic functions, one can say something more than about trigonometric functions. As one can easily imagine, the Taylor approximation is still a viable approach when it converges. For instance, the following approximations are true:

$$\begin{aligned}e^{1-\eta} &\simeq \frac{8}{3} - \frac{5}{2}\eta + \eta^2 - \frac{1}{6}\eta^3, \\ \ln(1 + 2\eta) &\simeq 2\eta - 2\eta^2 + \frac{8}{3}\eta^3.\end{aligned}$$

Now, let us examine the numerical representation of infinite values such as $e^{2\alpha+3}$ or $\ln \eta$. To do that, one needs to exit from the set of ANs and to work in a wider one, indicated with $\hat{\mathbb{V}}$.

A rigorous discussion can be found in the work of Benci and Cococcioni (2020), while for our practical purposes less formalism is needed.

The idea is to represent such transcendental infinite numbers as sum of monosemia, similarly to Eqn. (2) except that the exponents are taken not in $\mathbb{Q}$ but in a vector space $\mathbb{V}$ over $\mathbb{Q}$. To do this, one only requires the definition of a proper basis $V$. For the case $e^{2\alpha+3}$ it is enough to set

$$V = \{\beta_0, \beta_1\} = \left\{1, \frac{\alpha}{\ln \alpha}\right\} \tag{3}$$

and to define $\mathbb{V}$ as the set of all the non-standard numbers having the powers of $\alpha$ in the space spanned by $V$. Indeed, we have

$$\begin{aligned}e^{2\alpha+3} &= e^3 e^{2\alpha} = e^3 \alpha^{\log_\alpha e^{2\alpha}} = e^3 \alpha^{2\alpha \log_\alpha e} \\ &= e^3 \alpha^{2\alpha \frac{\ln e}{\ln \alpha}} = e^3 \alpha^{2\frac{\alpha}{\ln \alpha}} = e^3 \alpha^{2\beta_1}\end{aligned}$$

and more generally,

$$e^{2\alpha+3} + 5\alpha = e^3 e^{2\alpha} + 5\alpha = e^3 \alpha^{2\beta_1} + 5\alpha^{\beta_0},$$

$$2^\alpha = e^{\ln 2^\alpha} = e^{\ln(2)\alpha} = \alpha^{\ln(2)\beta_1}.$$

The case of $\ln \eta$ is quite similar but not straightforward, since it requires to pass through $\ln \alpha$. To numerically embed it, one can use the basis

$$V = \{\beta_{-1}, \beta_0\} = \left\{\frac{\ln \ln \alpha}{\ln \alpha}, 1\right\} \tag{4}$$

obtaining the identity

$$\ln \alpha = \alpha^{\log_\alpha \ln \alpha} = \alpha^{\frac{\ln \ln \alpha}{\ln \alpha}} = \alpha^{\beta_{-1}}.$$

Since $\ln \eta$ is the negative of $\ln \alpha$, it follows that

$$\ln \eta = -\ln \alpha = -\alpha^{\beta_{-1}}.$$

It is only right to say that such transcendental ANs are not very suitable for numerical computations, since actually they require a more complex structure, and even their basic algebraic operations are not easy to manage. Furthermore, they do not have a polynomial form, as opposed to BANs, and the choice of the transcendental basis $V$ is arbitrary and problem-dependent. For this reason, we omit a detailed description of their possible encoding. However, they can be useful for some particular problems, and so they deserved a mention in this pragmatical presentation of the topic.

## 5. Several applications

**5.1. Eigenvalues of a non-standard square matrix.** In this subsection we solve the problem of estimating the eigenvalue with largest absolute value of a non-standard matrix, i.e., a matrix filled with non-standard numbers. For the sake of simplicity, hereinafter we assume to work with BANs.

First of all we have implemented a procedure to create random square matrices with predefined non-standard eigenvalues. The procedure as given in Algorithm 1. Then, we implemented the non-standard version of the well-known power iteration method (Mises and Pollaczek-Geiringer, 1929) to find the highest-absolute-value eigenvalue of a matrix; its pseudocode is reported in Algorithm 2. It is worth noticing that it mainly resembles the standard version

**Algorithm 1.** Forming a random square matrix with predefined eigenvalues.

**Procedure** Rand_from_Eigen($e$)
1: /* *Get the matrix dimension* */
2: n = `length`(e)
3: /* *Generate a n × n normally distributed matrix* */
4: M = `randn`(n)
5: /* *Get the Q-matrix of the* `qr` *factorization of* M */
6: [ Q, ∼ ] = `qr`(M)
7: /* *Generate a diagonal matrix with the eigenvalues in e* */
8: D = `diag`($e$)
9: /* *Rotate* D *according to* Q */
10: **return** $Q^T DQ$

**Algorithm 2.** Finding the highest-absolute-value eigenvalue of a matrix.

**Procedure** Find_Max_Abs_Eigenv(A, $\varepsilon$)
1: /* *A: matrix to study;* $\varepsilon$ *: real valued tolerance* */
2: v = **1**
3: $\lambda_r = v^T Av$
4: $v = \frac{Av}{||Av||}$
5: $\lambda = v^T Av$
6: **while not** `all_components`$(|\lambda - \lambda_r|) < \varepsilon$ **do**
7: $\lambda_r = \lambda$
8: $v = \frac{Av}{||Av||}$
9: $\lambda = v^T Av$
10: **end while**
11: **return** $\lambda$

of the procedure, differing in a tiny but crucial aspect: the stopping criterion. As can be seen in Line 6, to guarantee that the algorithm convergence at *all* the components of the sought eigenvalue, one must consider the tolerance threshold, at each of them separately. Indeed, neither a non-standard threshold nor a threshold on the smallest-magnitude component can accomplish the task. The reason is that a non-standard threshold would prefer approximating solution manifesting insignificant improvements in higher order monosemia even at the expenses of enormous errors on the lower order ones; by contrast, a threshold on the smallest magnitude manosemium would not guarantee convergence on the higher order ones. Later in this section, a numerical example which illustrates the effectiveness of the approach is proposed.

For the purposes of this study, we used the ANs encoding BAN3, and we applied Algorithm 2 to the matrix in Eqn. (6). The latter was constructed in accordance to Algorithm 1, rotating the reference system by

$$Q = \begin{bmatrix} -0.53 & -0.846 & -0.055 \\ 0.829 & -0.531 & 0.175 \\ -0.177 & 0.047 & 0.983 \end{bmatrix} \tag{5}$$

and setting its eigenvalues to $\alpha + 5 - 4\eta$, 1, and $\eta$. The algorithm iterations are reported in Table 2, where $v_k$ is the approximated eigenvector, $\lambda_k$ is the approximated eigenvalue and $|r_k| = |\lambda_k - \lambda_r|$ is the optimality residual. For the sake of readability, the BANs encoding is reported in a human readable form. To better understand the inner representation of BANs, we provide in Table 5, in Appendix, the same results but this time with BANs shown in normal form, i.e., as they are output on screen by our software library.

From Table 2 we can can see a remarkable thing: even if the starting eigenvector is assumed to be finite, the approximated eigenvalue in the iterative scheme is immediately infinite, making the maximum eigenvalue infinite. The next iterations just help to refine its components (both the infinite, the finite and the infinitesimal one). More precisely, the algorithm is infinitely close to the optimal value just after a few iterations. This fact is present also in the other experiments with the power iteration method, suggesting that even in a non-standard context the high convergence rate of the algorithm is preserved; this property will be investigated in a future work. There is a phenomenon which may deserve further investigation in a future work. During the algorithm iterations, the very early approximations of an infinite eigenvalue may happen to be finite, probably due to rounding and other computation techniques. This may be of interest since it is the very first time that an approximating procedure which works with continuous non-standard quantities "jumps" from finite to infinite values, and a deeper study of it may reveal prospective research directions.

As is known, assuming the matrix A non-singular, the smallest eigenvalues can be computed as well applying the power iteration method to the matrix inverse and inverting the procedure output. However, the condition number plays an even more critical role in the non-standard world than in the standard one. Indeed, the condition number of non-standard matrices can be non-standard as well, meaning that the computation errors can be out-of-scale with respect to the real output, i.e., it can be infinitely bigger than the norm of the matrix itself. It is the case of the matrix in Eqn. (6) whose condition number is

$$\kappa(A) = \frac{\alpha + 5 - 4\eta}{\eta} = \alpha^2 + 5\alpha - 4,$$

while the maximum eigenvalue of $A^{-1}$ is known to be just $\alpha$.

Actually, computing the inverse as any common linear algebra suite does, i.e., solving the system $AA^{-1} = I$, one gets the result in Eqn. (7), while the correct answer should have been the matrix reported in Eqn. (8). Notice that the ill-conditioning of A does not affect only the magnitude of the components digits, which are

Table 2. Non-standard power iteration method applied step-by-step to Eqn. (6).

| $\mathbf{v}_k$ | $\lambda_k$ | $\lvert r_k\rvert$ |
|---|---|---|
| $\begin{bmatrix}1\\1\\1\end{bmatrix}$ | $0.224\alpha + 3.168 - 0.168\eta$ | - |
| $\begin{bmatrix}0.829 + 1.604\eta - 12.139\eta^2\\-0.531 + 2.561\eta - 10.287\eta^2\\0.175 + 0.168\eta + 0.134\eta^2\end{bmatrix}$ | $\alpha + 5 - 13.157\eta$ | $0.776\alpha + 1.832 - 12.991\eta$ |
| $\begin{bmatrix}0.829 + 1.604\eta^2\\-0.531 + 2.561\eta^2\\0.175 + 0.167\eta^2\end{bmatrix}$ | $\alpha + 5 - 4\eta$ | $9.159\eta$ |
| $\begin{bmatrix}0.829\\-0.531\\0.175\end{bmatrix}$ | $\alpha + 5 - 4\eta$ | $0$ |

$$A = \begin{bmatrix} 0.687\alpha + 3.719 - 2.719\eta & -0.440\alpha - 1.753 + 1.753\eta & 0.145\alpha + 0.754 - 0.754\eta \\ -0.440\alpha - 1.753 + 1.753\eta & 0.282\alpha + 2.126 - 1.126\eta & -0.093\alpha - 0.417 + 0.417\eta \\ 0.145\alpha + 0.754 - 0.754\eta & -0.093\alpha - 0.417 + 0.417\eta & 0.031\alpha + 0.156 + 0.844\eta \end{bmatrix}, \tag{6}$$

$$\tilde{A}^{-1} = \begin{bmatrix} -1.3e16 - 5e33\eta - 2e51\eta^2 & 3.3e15 + 1.3e33\eta + 5.2e50\eta^2 & 6.9e16 + 2.8e34\eta + 1.1e52\eta^2 \\ 3.3e15 + 1.3e33\eta + 5.2e50\eta^2 & -8.7e14 - 3.5e32\eta - 1.4e50\eta^2 & -1.8e16 - 7.3e33\eta - 2.9e51\eta^2 \\ 6.9e16 + 2.7e34\eta + 1.1e52\eta^2 & -1.8e16 - 7.3e33\eta - 2.9e51\eta^2 & -3.8e17 - 1.5e35\eta - 6.1e52\eta^2 \end{bmatrix}. \tag{7}$$

$$A^{-1} = \begin{bmatrix} 0.031\alpha + 0.281 + 0.687\eta & -0.008\alpha + 0.449 - 0.440\eta & -0.174\alpha + 0.029 + 0.145\eta \\ -0.008\alpha + 0.449 - 0.440\eta & 0.002\alpha + 0.716 + 0.282\eta & 0.046\alpha + 0.0474 - 0.092\eta \\ -0.174\alpha + 0.029 + 0.145\eta & 0.046\alpha + 0.047 - 0.093\eta & 0.966\alpha + 0.003 + 0.031\eta \end{bmatrix}, \tag{8}$$

suspiciously big with respect to the ones in Eqn. (6), but also the magnitude of inverse entries, which differ in scale by a factor of $\alpha$. For completeness, we computed the highest-absolute-value eigenvalue also for (8), which outputs the correct value, i.e., $1/\eta = \alpha$. The same would not have been true if we had launched the algorithm on (7). Likely, one may look for the closest eigenvalue to a given constant $\mu$. In the literature, the algorithm which accomplishes the this task is the called inverse iteration method (Pohlhausen, 1921). Its non-standard extension, which again differs from its standard counterpart just by the stopping criterion, is reported in Algorithm 3.

As a practical case of study, we performed an experiment with a $8 \times 8$ random input matrix (omitted for space reasons) whose eigenvalues were

$$\begin{aligned} &\lambda_1 = 100\eta - 3\eta^2 + 20\eta^3, \quad &&\lambda_2 = -4\eta + 5\eta^2 + 7\eta^3, \\ &\lambda_3 = \eta + 2\eta^2 + 9\eta^3, &&\lambda_4 = -2\eta + 5\eta^2 - 100\eta^3, \\ &\lambda_5 = 13\eta - 3\eta^2 + 15\eta^3, &&\lambda_6 = \ \ 30\eta - 19\eta^2 - \eta^3, \\ &\lambda_7 = -25\eta - \eta^2 + 8\eta^3, &&\lambda_8 = -42\eta + 2\eta^2 - 2\eta^3. \end{aligned}$$

Notice that all the chosen eigenvalues have the same magnitude because of the reasons discussed about the condition number and the numerical stability of the algorithm. We set the reference eigenvalue $\mu$ to $\eta$; thus the inverse iteration method should output $\eta + 2\eta^2 + 9\eta^3$. The algorithm iterations are reported in Table 4.

Among the possible applications of the search for eigenvalues, we count game theory (Thompson and Weil, 1972; 1969; Weil, 1968) (recent studies in non-Archimedean game theory can be found in the works of Cococcioni *et al.* (2021), or Fiaschi and Cococcioni (2018; 2020)), or the search for the stationary distribution of an ergodic Markov chain (as shown in next section), to mention a few.

**5.2. Markov chains and ANs to model quasi-unreachable states, quasi-absorbing states and quasi-bipartite chains.** In this subsection we discuss how ANs can be of any help when dealing with discrete-time finite Markov chains (MCs in brief) (Gagniuc, 2017). Recalling some basic knowledge, an MC is fully described by the square matrix $P$ of the transition kernels, also

Table 3. Non-standard power iteration method applied step-by-step to Eqn. (8).

| $\mathbf{v}_k$ | $\lambda_k$ | $\lvert r_k\rvert$ |
|---|---|---|
| $\begin{bmatrix}1\\1\\1\end{bmatrix}$ | $0.727\alpha + 2.049 + 0.224\eta$ | - |
| $\begin{bmatrix}-0.177 + 0.890\eta + 0.710\eta^2\\ 0.047 + 1.421\eta - 0.361\eta^2\\ 0.983 + 0.093\eta - 1.289\eta^2\end{bmatrix}$ | $\alpha - 2.819\eta$ | $0.273\alpha - 2.049 - 3.043\eta$ |
| $\begin{bmatrix}-0.177 + 0.890\eta^2\\ 0.047 + 1.421\eta^2\\ 0.983 + 0.093\eta^2\end{bmatrix}$ | $\alpha$ | $2.819\eta$ |
| $\begin{bmatrix}-0.177\\ 0.047\\ 0.983\end{bmatrix}$ | $\alpha$ | $0$ |

Table 4. Non-standard inverse iteration method with $\mu = \eta$.

| $\lambda_k$ | $\lvert r_k\rvert$ |
|---|---|
| $-0.745\eta + 2.005\eta^2 - 38.466\eta^3$ | – |
| $\eta + 2.0\eta^2 + 8.846\eta^3$ | $1.745\eta - 0.005\eta^2 + 47.312\eta^3$ |
| $\eta + 2.0\eta^2 + 6.434\eta^3$ | $-2.413\eta^3$ |
| $\eta + 2.0\eta^2 + 3.619\eta^3$ | $2.815\eta^3$ |
| $\eta + 2.0\eta^2 + 10.053\eta^3$ | $6.434\eta^3$ |
| $\eta + 2.0\eta^2 + 11.259\eta^3$ | $1.206\eta^3$ |
| $\eta + 2.0\eta^2 + 6.836\eta^3$ | $4.423\eta^3$ |
| $\eta + 2.0\eta^2 + 10.455\eta^3$ | $3.619\eta^3$ |
| $\eta + 2.0\eta^2 + 5.63\eta^3$ | $4.825\eta^3$ |
| $\eta + 2.0\eta^2 + 6.434\eta^3$ | $0.804\eta^3$ |
| $\eta + 2.0\eta^2 + 5.227\eta^3$ | $1.206\eta^3$ |
| $\eta + 2.0\eta^2 + 8.846\eta^3$ | $3.619\eta^3$ |
| $\eta + 2.0\eta^2 + 2.815\eta^3$ | $6.032\eta^3$ |
| $\eta + 2.0\eta^2 + 11.661\eta^3$ | $8.846\eta^3$ |
| $\eta + 2.0\eta^2 + 5.63\eta^3$ | $6.032\eta^3$ |
| $\eta + 2.0\eta^2 + 8.444\eta^3$ | $2.815\eta^3$ |
| $\eta + 2.0\eta^2 + 8.444\eta^3$ | $0$ |

known as the transition matrix, such that

$$P_{i,j} \geq 0, \quad \forall\, i, j = 1, \ldots, n$$

and

$$\sum_{i=1}^{n} P_{i,j} = 1, \quad \forall\, j = 1, \ldots, n,$$

where $n$ is the matrix dimension.

A distribution $\pi$ over states is said stationary if it is invariant with respect to $P$, i.e., $P\pi = \pi$. It can be found by solving the (redundant) linear system in (9) or computing the eigenvector associated with the unit eigenvalue by means of Algorithm 3, as already shown in the previous section,

$$\begin{cases} P\pi = \pi, \\ ||\pi||_1 = 1, \end{cases} \tag{9}$$

The existence and uniqueness of such a distribution are guaranteed by the ergodic theorem. Finally, a chain satisfying the hypotheses of the ergodic theorem is referred to as ergodic.

**5.2.1. Modelling quasi-unreachable states.** A quasi-unreachable state is a state of an MC which is theoretically reachable from elsewhere, but in practice the probability that this happens is so small that it results to be out of scale when compared with the other quantities in play. Sometimes one cannot get rid of such states because they are essential in the description of the system under study. A possible way to model them is to construct transition kernels assigning a very small probability to hit these states. This approach can work but lays itself open to numerical instabilities. On the other hand, here we propose to model such situations by means of non-standard kernels which assign infinitesimal probabilities (Benci *et al.*, 2018), rather than very small but finite ones. In this way, the numerical instabilities are forestalled without affecting the quality of the model.

As a practical example, consider the chain in Fig. 1 along with its matrix representation

$$P = \begin{bmatrix} \frac{1}{3} & 0 & 0 & \frac{1}{4} & 0 \\ \frac{1}{2} & 0 & \frac{1}{3} & 0 & 0 \\ \frac{1}{6} & 0 & 0 & \frac{3}{4} & 1 \\ 0 & 1-\eta & \frac{2}{3} & 0 & 0 \\ 0 & \eta & 0 & 0 & 0 \end{bmatrix}. \tag{10}$$

Here, the state 5 is a quasi-unreachable state, since it can be hit just from 2 and with the infinitesimal probability $\eta$.

It can be demonstrated that the chain in (10) is ergodic, and it admits one unique stationary distribution. Leveraging Algorithm 3, one can numerically approximate it obtaining the distribution

$$\pi_n = \begin{bmatrix} 0.142 - 0.057\eta + 0.004\eta^2 \\ 0.173 - 0.012\eta + 0.001\eta^2 \\ 0.307 - 0.049\eta + 0.003\eta^2 \\ 0.378 - 0.153\eta + 0.011\eta^2 \\ \boldsymbol{0.173\eta + 0.012\eta^2} \end{bmatrix}. \tag{11}$$

As expected, the probability to asymptotically be in state 5 is infinitesimal, since it is extremely difficult to reach, but it is surprisingly easy to leave. This fact reflects exactly what a quasi-unreachable state is and how it should behave.

Since $P$ is low dimensional, an exact computation of the stationary distribution can be attained with the help of symbolic tools such as Mathematica®. The symbolic stationary distribution $\pi_s$, computed with Mathematica, is

$$\pi_s = \left[ \frac{18-6\eta}{127+9\eta}, \quad \frac{22}{127+9\eta}, \quad \frac{39+9\eta}{127+9\eta}, \quad \frac{48-16\eta}{127+9\eta}, \quad \frac{\boldsymbol{22\eta}}{\boldsymbol{127+9\eta}} \right].$$

Thus $\pi_n$ is only a numerical approximation the exact value, $\pi_s$. However, computing the exact solution using Mathematica® is not a viable solution for practical problems, as already said in Section 3. Indeed, symbolic tools do not scale with problem dimension and start struggling even with simple $8 \times 8$ linear systems (Cococcioni *et al.*, 2021).

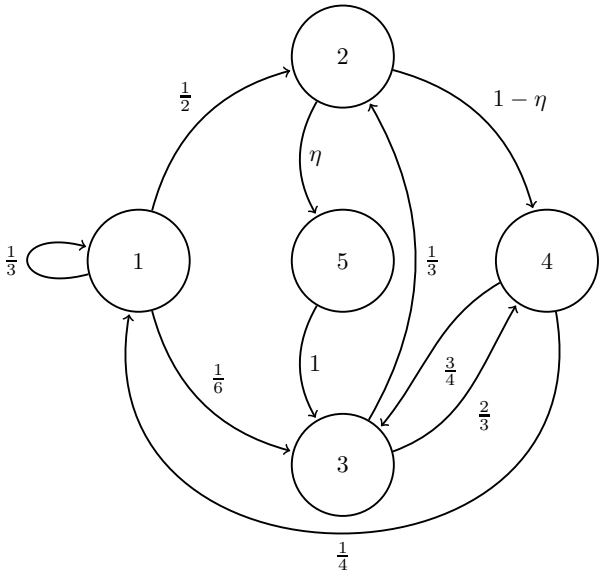



Fig. 1. Markov chain where state 5 is a quasi-unreachable state.

**Algorithm 3.** Non-standard inverse iteration method.

```
/* A: square matrix
   μ: reference eigenvalue
   ε: real valued tolerance */
Procedure NS_Inverse_Iteration(A, μ, ε)
B = (A − μI)^{-1}
v = 1
λ_r = 1 / (v^T A^{-1} v)
v = Bv / ||Bv||
λ = 1 / (v^T Bv) + μ
while not all_component(|λ − λ_r|) < ε do
    λ_r = λ
    v = Bv / ||Bv||
    λ = 1 / (v^T Bv) + μ
end while
return λ
```

**5.2.2. Modelling quasi-absorbing states.** As opposed to quasi-unreachable states, a quasi-absorbing state is a state which is very difficult to abandon when compared with the probability to hit it. Similarly, however, one can model such a kind of states with an infinitesimal outgoing

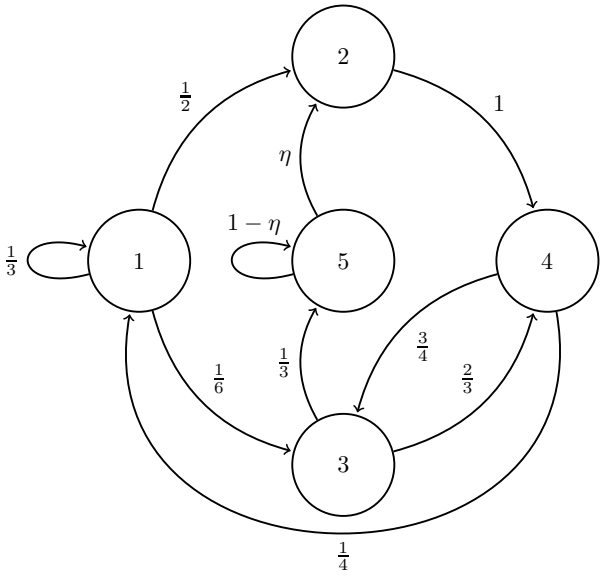



Fig. 2. Markov chain where state 5 is a quasi-absorbing state.

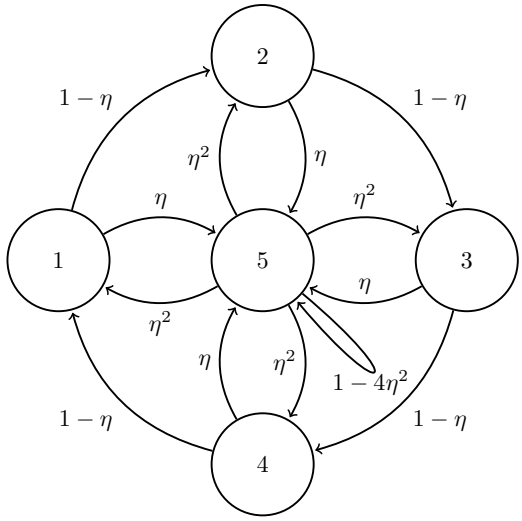



Fig. 3. Markov chain where state 5 is more quasi-absorbing than quasi-unreachable.

probability. A first example is reported in

$$P = \begin{bmatrix} \frac{1}{3} & 0 & 0 & \frac{1}{4} & 0 \\ \frac{1}{2} & 0 & 0 & 0 & \eta \\ \frac{1}{6} & 0 & 0 & \frac{3}{4} & 0 \\ 0 & 1 & \frac{2}{3} & 0 & 0 \\ 0 & 0 & \frac{1}{3} & 0 & 1-\eta \end{bmatrix} \tag{12}$$

and depicted in Fig. 2. For the sake of brevity, we anticipate that all the experiments here on will involve ergodic chains.

From the stationary distribution

$$\pi_n = \begin{bmatrix} 1.385\eta - 13.527\eta^2 \\ 1.692\eta - 16.533\eta^2 \\ 3\eta - 29.308\eta^2 \\ 3.692\eta - 36.071\eta^2 \\ \mathbf{1 - 9.769\eta + 95.438\eta^2} \end{bmatrix},$$

$$\pi_s = \left[ \frac{18\eta}{13+127\eta}, \quad \frac{22\eta}{13+127\eta}, \right.$$
$$\left. \frac{39\eta}{13+127\eta}, \quad \frac{48\eta}{13+127\eta}, \quad \frac{\mathbf{13}}{\mathbf{13+127\eta}} \right]^T, \tag{13}$$

we can deduce that state 5 has an asymptotic probability very close to 1, as expected by a quasi-absorbing state.

Even if this result seems to be trivial, it is not. First of all, it confirms the good behaviour of the numerical technology; secondly, which is more important, it sheds light on the probability distribution *outside* the quasi-absorbing state. Indeed, not considering the improbable states, one cannot assert that state 1 is about 2.67 times less frequent than state 4, or that the latter is 2.18 times more frequent than state 2. Even in case $\eta$ was represented by a very small value, we would not be able to say something similar with absolute certainty. Indeed, numerical inaccuracies introduced by such a rough approximation may badly affect the result, especially in non-trivial cases. On the contrary, a numerical non-Archimedean approach does not suffer from this potential lack of reliability and provides a result coherent with the expectations: the probability of finding the system in state 5 is incommensurable with respect to the other states.

Now let us consider two even more interesting examples where the quasi-absorbing state is, *at the same time*, a quasi-unreachable one. These case studies are useful to get some insight into scenarios involving MCs with infinitesimals.

The first one is about a chain where the incriminated state, number 5, has an outgoing probability which is infinitely smaller than the ingoing one, even if the latter is already infinitesimal. The chain is reported in Fig. 3 and

$$P = \begin{bmatrix} 0 & 0 & 0 & 1-\eta & \eta^2 \\ 1-\eta & 0 & 0 & 0 & \eta^2 \\ 0 & 1-\eta & 0 & 0 & \eta^2 \\ 0 & 0 & 1-\eta & 0 & \eta^2 \\ \eta & \eta & \eta & \eta & 1-4\eta^2 \end{bmatrix}; \tag{14}$$

its stationary distribution is

$$\pi_n = \begin{bmatrix} \eta - 4\eta^2 \\ \eta - 4\eta^2 \\ \eta - 4\eta^2 \\ \eta - 4\eta^2 \\ \mathbf{1 - 4\eta + 16\eta^2} \end{bmatrix}, \tag{15}$$

$$\pi_s = \left[\tfrac{\eta}{1+4\eta}, \quad \tfrac{\eta}{1+4\eta}, \quad \tfrac{\eta}{1+4\eta}, \quad \tfrac{\eta}{1+4\eta}, \quad \tfrac{\mathbf{1}}{\mathbf{1+4\eta}}\right]^T.$$

Even if reasonable, it is interesting to note that 5 has a finite asymptotic probability while all the other states infinitesimal. This holds true even if 5 is extremely difficult to reach (a probability to be hit of just $\eta$) and even if starting from a node different from 5. The reason stems from the fact that the flow of the chain inexorably passes through 5 accumulating there, since it still inexorably leaves the state but with infinitely greater difficulty. Thus, 5 acts like a sink in the previous experiment, even if it is quasi-unreachable. This means that what counts to define the behaviour of a node in a chain is mainly the magnitude of the ratio among its input and output links. A possible system which can be modelled by this chain may be a very efficient device (nodes 1-2-3-4) which slowly loses energy (node 5). In addition, the loss is very marginal with respect to the device frequency of activity, and a small part of the lost energy can be renewed and put back in the system.

The second experiment involves again a slight variation in the chain (10), where this time we assume that also the transition from 5 to 3 has probability $\eta$. The chain is resumed in Fig. 4 and

$$P = \begin{bmatrix} \frac{1}{3} & 0 & 0 & \frac{1}{4} & 0 \\ \frac{1}{2} & 0 & \frac{1}{3} & 0 & 0 \\ \frac{1}{6} & 0 & 0 & \frac{3}{4} & \eta \\ 0 & 1-\eta & \frac{2}{3} & 0 & 0 \\ 0 & \eta & 0 & 0 & 1-\eta \end{bmatrix}. \tag{16}$$

The stationary distribution

$$\pi_n = \begin{bmatrix} \mathbf{0.121} - 0.030\eta - 0.003\eta^2 \\ \mathbf{0.148} + 0.013\eta + 0.001\eta^2 \\ \mathbf{0.262} + 0.083\eta + 0.007\eta^2 \\ \mathbf{0.322} - 0.079\eta - 0.007\eta^2 \\ \mathbf{0.148} + 0.013\eta - 0.001\eta^2 \end{bmatrix}, \tag{17}$$

$$\pi_s = \left[\tfrac{18-6\eta}{149-13\eta}, \quad \tfrac{22}{149-13\eta}, \quad \tfrac{39+9\eta}{149-13\eta}, \quad \tfrac{48-16\eta}{149-13\eta}, \quad \tfrac{22}{149-13\eta}\right]^T$$

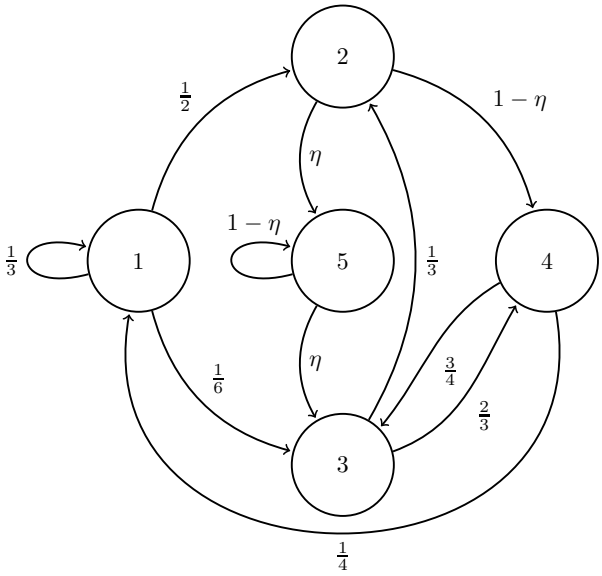



Fig. 4. Markov chain where state 5 is both a quasi-unreachable state and a quasi-absorbing one.

shows how, even if there exists a quasi-unreachable and a quasi-absorbing state, all the states can still have an asymptotic finite probability to be visited, provided that the quasi-absorbing state is also the quasi-unreachable one. We think this was a not so obvious *a-priori* outcome, and thus this makes this result rather interesting.

As a future work, we shall use non-Archimedean MCs to model quasi-unreachable and quasi-absorbing states, to model deterioration of financial assets in quantitative finance applications or vicious circles of recession in economics.

**5.3. Non-Archimedean Cholesky matrix factorization.** The Cholesky factorization is the decomposition of a positive definite square matrix $A$ into the product of a lower triangular matrix $L$ and its conjugate transpose, i.e., $LL^* = A$ (Golub and Van Loan, 2013). Among its many applications we count linear and non-linear optimization (Arora, 2004), matrix inversion (Krishnamoorthy and Menon, 2013), Monte Carlo simulations (see the Matlab implementation of the `randn` function), and Kalman filters (Bierman, 2006). The algorithm for computing the Cholesky factorization require the computation of the square root, which makes this example challenging, since it does not require just the algebraic operations mentioned above.

In the following, we show an example of Cholesky factorization of a non-Archimedean square positive definite symmetric random matrix represented using the BAN3 encoding. The pipeline involves the following steps:

- Generate a non-standard random matrix $G$ having

size $m \times n$, with $m \geq n$. Its entries are generated uniformly sampling each component in $[0, 1)$, admitting negative signs for non-leading monosemia.

- Compute $A$ as $G^T G$. $A$ is square, $n \times n$, and symmetric by construction. Shifting its spectrum by any positive constant, say 1, guarantees it to be positive-definite. To do this, it is enough to add the identity matrix $I$ to $A$.

- Finally, compute the matrix $L$, using the usual Cholesky algorithm implemented in a way which is BANs compliant.

The numerical results are reported as Eqns. (18)–(21), we set $m$ and $n$ equal to 5 and we skipped the addition of the identity matrix because $A$ was, by chance, already positive definite. In particular, Eqn. (20) shows the obtained Cholesky matrix $L$, while Eqn. (21) shows $LL^*$, i.e., what is expected to be close to A, up to numerical errors. For all the matrices we reported ANs in BAN2 format for space reasons but, as has been said, for a better precision we realized all the computations by means of BAN3 encoding. We evaluated the discrepancy between $A$ and $LL^*$ in order to measure the precision of the computations. It turned out that $||A - LL^*||_2$ amounts to $0.005 + 0.001\eta^2$, i.e., the approximation is quite accurate even on the infinitesimal monosemia since $||A||_2 = 6.24 + 3.88\eta$. Similar computations showed the same behaviour, suggesting the possibility to find analytical bounding errors for such non-Archimedean Cholesky factorization; this aspect will be investigated in a future work.

While in the work of Cococcioni *et al*. (2021) for the first time the numerical inverse of a non-Archimedean matrix was used operationally within an algorithm (the simplex one), this is the first time that a numerical Cholesky factorization of a non-Archimedean matrix is obtained. This approach would scale for large matrices, contrary to a pure symbolic approach like Mathematica®. As a final consideration concerning the Cholesky factorization obtained above, it is important to highlight that this was possible since, in this specific case, all the involved BANs in matrix $A$ had an even order (equal to zero, to be precise).

In general, the factorization might fail, since the Cholesky factorization requires the computation of the squared root of a BAN and this root cannot always be represented as a BAN, e.g., $\sqrt{\eta}$. To overcome this difficulty, two solutions are possible: (i) make use of general-purpose ANs instead of BANs; or, (ii) resort to the $LDL$ decomposition, which is a variant of the Cholesky one which does not require the computation of the square root. As a future work, we plan to implement the non-Archimedean $LDL$ factorization to solve lexicographic multi-objective quadratic programming problems using the non-Archimedean interior point method (Fiaschi and Cococcioni, 2021). Another interesting thing to do is to enrich our software library with a non-Archimedean function able to compute the non-Archimedean LU factorization of a matrix, as LU factorization is another key pillar of numerical computing.

## 6. Related works

Non-Archimedean scientific computing is an emerging research topic still in its infancy, with a huge amount of investigations waiting to be done. The biggest contribution to numerical non-Archimedean computing has been mainly due to Y.D. Sergeyev, who has introduced the grossone methodology (Sergeyev, 2017). Since its appearance in 2003, a number of applications have emerged in extremely disparate fields: optimization (De Leone *et al.*, 2020b; Lai *et al.*, 2021a; 2021b; De Leone, 2018; Cococcioni and Fiaschi, 2020; Cococcioni *et al.*, 2020) ordinary differential equations (Sergeyev *et al*., 2016; Amodio *et al*., 2017; Iavernaro *et al*., 2020) and machine learning (De Leone *et al.*, 2020a; Astorino and Fuduli, 2020), among others. In addition, an implementation of the grossone methodology within Simulink® has been recently introduced (Falcone *et al.*, 2020a; 2020b).

## 7. Conclusions

In this paper we have introduced a novel, simpler axiomatization of Alpha-Theory, to facilitate the adoption of non-standard methods among practitioners. This theory is easier than Robinson's non-standard analysis, since it is axiomatic and it does not require the knowledge of model theory and advanced concepts mathematical logic. Then, we have recalled the recently introduced algorithmic numbers. Finally we have shown three new applications (NA eigenvalue computation, NA Markov chains, and NA Cholesky factorization) of Alpha-Theory, never addressed before in a non-Archimedean setting. We believe that the time is now mature to start non-Archimedean scientific computing, which promises the possibility to model and solve new interesting problems, which cannot be modelled/solved using standard (i.e., Archimedean) numerical algorithms. In addition, our opinion is that fixed-precision non-Archimedean scientific computing has the chance to efficiently solve real-life problems, where infinite-precision or purely symbolic approaches might fail, due to their lower hardware-friendly nature.

## Acknowledgment

This work was partly supported by the Italian Ministry of Education and Research (MIUR) in the framework of the

$$G = \begin{bmatrix} 0.628 - 0.629\eta & 0.781 - 0.343\eta & 0.457 + 0.908\eta & 0.367 + 0.733\eta & 0.685 - 0.481\eta \\ 0.419 + 0.383\eta & 0.815 + 0.765\eta & 0.145 + 0.614\eta & 0.457 + 0.127\eta & 0.040 + 0.783\eta \\ 0.966 - 0.733\eta & 0.191 + 0.810\eta & 0.360 - 0.751\eta & 0.224 - 0.167\eta & 0.749 - 0.035\eta \\ 0.311 + 0.213\eta & 0.621 + 0.372\eta & 0.784 + 0.888\eta & 0.113 + 0.584\eta & 0.270 + 0.733\eta \\ 0.505 - 0.242\eta & 0.808 + 0.984\eta & 0.724 - 0.438\eta & 0.472 - 0.053\eta & 0.046 - 0.211\eta \end{bmatrix}, \tag{18}$$

$$A = \begin{bmatrix} 1.819 - 0.617\eta & 1.162 + 1.606\eta & 1.516 - 0.798\eta & 1.266 + 1.803\eta & 1.484 + 0.639\eta \\ 1.162 + 1.606\eta & 1.072 + 1.925\eta & 0.746 + 1.517\eta & 0.813 + 2.119\eta & 1.193 + 1.956\eta \\ 1.516 - 0.798\eta & 0.746 + 1.517\eta & 1.710 - 1.773\eta & 0.929 + 0.934\eta & 1.043 - 0.713\eta \\ 1.266 + 1.803\eta & 0.813 + 2.119\eta & 0.929 + 0.934\eta & 1.183 + 2.514\eta & 1.292 + 1.489\eta \\ 1.484 + 0.639\eta & 1.193 + 1.956\eta & 1.043 - 0.713\eta & 1.292 + 1.489\eta & 1.657 + 0.642\eta \end{bmatrix}, \tag{19}$$

$$\boldsymbol{L} = \begin{bmatrix} \mathbf{1.349 - 0.229\eta} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0.862 + 1.337\eta} & \mathbf{0.574 - 0.330\eta} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{1.125 - 0.401\eta} & \mathbf{-0.389 + 0.402\eta} & \mathbf{0.542 - 0.514\eta} & \mathbf{0} & \mathbf{0} \\ \mathbf{0.938 + 1.496\eta} & \mathbf{0.008 - 0.736\eta} & \mathbf{-0.227 - 1.436\eta} & \mathbf{0.500 - 0.934\eta} & \mathbf{0} \\ \mathbf{1.101 + 0.660\eta} & \mathbf{0.427 + 0.098\eta} & \mathbf{-0.053 - 2.166\eta} & \mathbf{0.488 - 1.151\eta} & \mathbf{0.152 - 0.001\eta} \end{bmatrix}, \tag{20}$$

$$A \simeq LL^* = \begin{bmatrix} 1.820 - 0.618\eta & 1.160 + 1.610\eta & 1.520 - 0.799\eta & 1.260 + 1.800\eta & 1.480 + 0.638\eta \\ 1.160 + 1.610\eta & 1.070 + 1.930\eta & 0.750 + 1.520\eta & 0.810 + 2.120\eta & 1.190 + 1.960\eta \\ 1.520 - 0.799\eta & 0.750 + 1.520\eta & 1.710 - 1.770\eta & 0.930 + 0.935\eta & 1.040 - 0.712\eta \\ 1.260 + 1.800\eta & 0.810 + 2.120\eta & 0.930 + 0.935\eta & 1.180 + 2.510\eta & 1.290 + 1.490\eta \\ 1.480 + 0.638\eta & 1.190 + 1.960\eta & 1.040 - 0.712\eta & 1.290 + 1.490\eta & 1.660 + 0.643\eta \end{bmatrix}. \tag{21}$$

CrossLab project (Departments of Excellence), granted to the Department of Information Engineering of the University of Pisa. We wish to thank the three anonymous reviewers for their helpful comments.

## References

Amodio, P., Iavernaro, F., Mazzia, F., Mukhametzhanov, M. and Sergeyev, Y. (2017). A generalized Taylor method of order three for the solution of initial value problems in standard and infinity floating-point arithmetic, *Mathematics and Computers in Simulation* **141**: 24–39, DOI: 10.1016/j.matcom.2016.03.007.

Arora, J.S. (2004). *Introduction to Optimum Design*, Elsevier, San Diego.

Astorino, A. and Fuduli, A. (2020). Spherical separation with infinitely far center, *Soft Computing* **24**(23): 17751–17759.

Benci, V. and Cococcioni, M. (2020). The algorithmic numbers in non-Archimedean numerical computing environments, *Discrete and Continuous Dynamical Systems S* **14**(5): 1673–1692, DOI: 10.3934/dcdss.2020449.

Benci, V. and Di Nasso, M. (2003). Numerosities of labelled sets: A new way of counting, *Advances in Mathematics* **173**(1): 50–67.

Benci, V. and Di Nasso, M. (2018). *How to Measure the Infinite: Mathematics with Infinite and Infinitesimal Numbers*, World Scientific, Singapore.

Benci, V., Di Nasso, M. and Forti, M. (2006). The eightfold path to nonstandard analysis, *in* N. J. Cutland *et al.* (Eds), *Nonstandard Methods and Applications in Mathematics*, AK Peters, Wellesley, pp. 3–44.

Benci, V., Horsten, L. and Wenmackers, S. (2018). Infinitesimal probabilities, *British Journal for the Philosophy of Science* **69**(2): 509–552.

Bierman, G.J. (2006). *Factorization Methods for Discrete Sequential Estimation*, Courier Corporation, North Chelmsford.

Cococcioni, M., Cudazzo, A., Pappalardo, M. and Sergeyev, Y. (2020). Solving the lexicographic multi-objective mixed-integer linear programming problem using branch-and-bound and grossone methodology, *Communications in Nonlinear Science and Numerical Simulation* **84**: 105177, DOI: 10.1016/j.cnsns.2020.105177.

Cococcioni, M. and Fiaschi, L. (2020). The Big-M method with the numerical infinite M, *Optimization Letters* **15**: 2455–2468, DOI: 10.1007/s11590-020-01644-6.

Cococcioni, M., Fiaschi, L. and Lambertini, L. (2021). Non-Achimedean zero-sum games, *Journal of Applied and Computational Mathematics* **113483**: 1–17, DOI: 10.1016/j.cam.2021.113483.

Conway, J.H. (2000). *On Numbers and Games*, CRC Press, New York.

De Leone, R. (2018). Nonlinear programming and grossone: Quadratic programming and the role of constraint qualifications, *Applied Mathematics and Computation* **318**: 290–297, DOI: 10.1016/j.amc.2017.03.029.

De Leone, R., Egidi, N. and Fatone, L. (2020a). The use of grossone in elastic net regularization and sparse support vector machines, *Soft Computing* **24**(23): 17669–17677.

De Leone, R., Fasano, G., Roma, M. and Sergeyev, Y.D. (2020b). Iterative grossone-based computation of negative

curvature directions in large-scale optimization, *Journal of Optimization Theory and Applications* **186**(2): 554–589.

Dehn, M. (1900). Die Legendre'schen Sätze über die Winkelsumme im Dreieck, *Mathematische Annalen* **53**(1900): 404–439, DOI: 10.1007/BF01448980.

Deveau, M. and Teismann, H. (2014). 72+ 42: Characterizations of the completeness and Archimedean properties of ordered fields, *Real Analysis Exchange* **39**(2): 261–304.

Falcone, A., Garro, A., Mukhametzhanov, M.S. and Sergeyev, Y.D. (2020a). Representation of grossone-based arithmetic in simulink for scientific computing, *Soft Computing* **24**(23): 17525–17539.

Falcone, A., Garro, A., Mukhametzhanov, M.S. and Sergeyev, Y.D. (2020b). A Simulink-based software solution using the infinity computer methodology for higher order differentiation, *Applied Mathematics and Computation* **409**: 125606, DOI: 10.1016/j.amc.2020.125606.

Fiaschi, L. and Cococcioni, M. (2018). Numerical asymptotic results in game theory using Sergeyev's infinity computing, *International Journal of Unconventional Computing* **14**: 1–25.

Fiaschi, L. and Cococcioni, M. (2020). Non-Archimedean game theory: A numerical approach, *Applied Mathematics and Computation* **409**: 125356, DOI: 10.1016/j.amc.2020.125356.

Fiaschi, L. and Cococcioni, M. (2021). A non-Archimedean interior point method for solving lexicographic multi-objective quadratic programming problems, *EURO Journal on Computational Optimization*, (submitted).

Gagniuc, P.A. (2017). *Markov Chains: From Theory to Implementation and Experimentation*, Wiley, Hoboken.

Golub, G.H. and Van Loan, C.F. (2013). *Matrix Computations*, JHU Press, Baltimore.

Iavernaro, F., Mazzia, F., Mukhametzhanov, M. and Sergeyev, Y. (2020). Conjugate-symplecticity properties of Euler–Maclaurin methods and their implementation on the infinity computer, *Applied Numerical Mathematics* **155**: 58–72, DOI: 10.1016/j.apnum.2019.06.011.

Keisler, H.J. (1976). *Foundations of Infinitesimal Calculus*, Prindle, Weber & Schmidt, Boston.

Krishnamoorthy, A. and Menon, D. (2013). Matrix inversion using Cholesky decomposition, *2013 IEEE Conference on Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA'13), Poznan, Poland*, pp. 70–72.

Lai, L., Fiaschi, L., Cococcioni, M. and Deb, K. (2021a). Handling priority levels in mixed Pareto-lexicographic many-objective optimization problems, *Evolutionary Multi-Criterion Optimization, Shenzhen, China,* pp. 362–374, DOI: 10.1007/978-3-030-72062-9_29.

Lai, L., Fiaschi, L., Cococcioni, M. and Deb, K. (2021b). Solving mixed pareto-lexicographic multi-objective optimization problems: The case of priority levels, *IEEE Transactions on Evolutionary Computation* **25**(5): 971–985, DOI: 10.1109/TEVC.2021.3068816.

Levi-Civita, T. (1892). Sugli infiniti ed infinitesimi attuali quali elementi analitici, *Atti del R. Istituto Veneto di Scienze Lettere ed Arti* **Series 7**: 1892–1893.

Mises, R. and Pollaczek-Geiringer, H. (1929). Praktische verfahren der gleichungsauflösung, *Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik* **9**(1): 58–77.

Pohlhausen, E. (1921). Berechnung der eigenschwingungen statisch-bestimmter fachwerke, *Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik* **1**(1): 28–42.

Robinson, A. (1996). *Non-Standard Analysis*, 2nd Edn, Princeton University Press, Princeton.

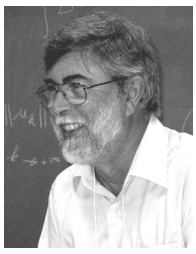
Sergeyev, Y. (2017). Numerical infinities and infinitesimals: Methodology, applications, and repercussions on two Hilbert problems, *EMS Surveys in Mathematical Sciences* **4**(2): 219—320.

Sergeyev, Y.D., Mukhametzhanov, M., Mazzia, F., Iavernaro, F. and Amodio, P. (2016). Numerical methods for solving initial value problems on the infinity computer, *Journal of Unconventional Computing* **12**(1): 3–23.

Thompson, G.L. and Weil, Jr, R.L. (1969). Further relations between game theory and eigensystems, *SIAM Review* **11**(4): 597–602.

Thompson, G.L. and Weil, R.L. (1972). The roots of matrix pencils ($Ay = \lambda By$): Existence, calculations, and relations to game theory, *Linear Algebra and Its Applications* **5**(3): 207–226.

Weil, Jr, R.L. (1968). Game theory and eigensystems, *SIAM Review* **10**(3): 360–367.



**Vieri Benci** has mainly worked in the fields of equations of mathematical physics and questions of logic related to infinitesimal and infinite numbers and appears in the *ISI Highly Cited* list. Among his most recent books are *Variational Methods in Nonlinear Field Equations* (with D. Fortunato) (Springer, 2014), *How to Measure the Infinite* (with M. Di Nasso) (World Scientific, 2018), *La matematica e l'infinito* (with P. Freguglia) (Carocci, 2019). During his career he has also taught courses in many foreign countries such as the USA, China, Brazil, Belgium and Saudi Arabia.

**Marco Cococcioni** received his *Laurea* degree in 2000 and his PhD degree in computer engineering at the University of Pisa in 2000 and 2004, respectively. In 2010–2011 he was a senior visiting scientist at the NATO Undersea Research Centre (now CMRE) in La Spezia, Italy. Since 2016 he has been an associate professor at the Department of Information Engineering of the University of Pisa. He is a member of three IEEE task forces: Genetic Fuzzy Systems, Computational Intelligence in Security and Defense, and Intelligent System Application. Professor Cococcioni has co-authored more than 100 contributions to international journals and conferences.

Table 5. Non-Archimedean power iteration method applied step-by-step to Eqn. (6) with BANs, this time expressed in their normal form.

| $\mathbf{v}_k$ | $\lambda_k$ | $\lvert r_k\rvert$ |
|---|---|---|
| $\begin{bmatrix}\alpha^0(1+0\eta+0\eta^2)\\ \alpha^0(1+0\eta+0\eta^2)\\ \alpha^0(1+0\eta+0\eta^2)\end{bmatrix}$ | $\alpha^1(0.224+3.168\eta-0.168\eta^2)$ | — |
| $\begin{bmatrix}\alpha^0(0.829+1.604\eta-12.139\eta^2\\ \alpha^0(-0.531+2.561\eta-10.287\eta^2)\\ \alpha^0(0.175+0.168\eta+0.134\eta^2)\end{bmatrix}$ | $\alpha^1(1+5\eta-13.157\eta^2)$ | $\alpha^1(0.776+1.832\eta-12.991\eta^2)$ |
| $\begin{bmatrix}\alpha^0(0.829+0\eta+1.604\eta^2)\\ \alpha^0(-0.531+0\eta+2.561\eta^2)\\ \alpha^0(0.175+0\eta+0.167\eta^2)\end{bmatrix}$ | $\alpha^1(1+5\eta-4\eta^2)$ | $\alpha^{-1}(9.159+0\eta+0\eta^2)$ |
| $\begin{bmatrix}\alpha^0(0.829+0\eta+0\eta^2)\\ \alpha^0(-0.531+0\eta+0\eta^2)\\ \alpha^0(0.175+0\eta+0\eta^2)\end{bmatrix}$ | $\alpha^1(1+5\eta-4\eta^2)$ | $\alpha^0(0+0\eta+0\eta^2)$ |

**Lorenzo Fiaschi** was born in 1995. He received his BSc degree in computer engineering from the University of Pisa, Italy, in 2018. He received his MSc degree in computer science and artificial intelligence from the University of Genoa, Italy, in 2019. He is working towards his PhD degree in information engineering at the University of Pisa. His current research interests include artificial and evolutionary intelligence, applied non-Archimedean models and game theory.

## Appendix

## Example of BANs expressed in their normal form

In Table 5 we report the result already shown in Table 2, but this time presenting BANs in their normal form, i.e., how they are internally represented in our software library. Even if this version is less readable and elegant than the one of Table 2, it helps to better understand how BANs can appear in numerical computations performed by the computer.

Received: 12 February 2021
Revised: 7 July 2021
Accepted: 21 September 2021