Constructing a hyperoperation sequence – pisa hyperoperations

A P Andonov
St. Cyril and St. Methodius University of Veliko Turnovo, Faculty of Mathematics and Informatics, Department of Computer Systems and Technologies, 5003 Veliko Tarnovo, 2 T. Turnovski str., Bulgaria

e-mail D1097@sd.uni-vt.bg

Abstract. In the context of other hyperoperation sequences, a new sequence of operations is constructed. A review of its properties reveals dependences between pairs of numbers, and so the sibling numbers are established. Two theorems are proven and the connection to small base tetration is revealed. The problem of extending the sequence to real levels is considered and linked to tetration height extension. The pisa operations can be a useful tool for exploring tetration and large numbers.

1. Introduction

For decades there is a race to construct larger and larger numbers and these numbers usually represent nothing in the physical world. These unimaginable numbers are so far apart that they don’t relate, and the only action on them is to compare their values, which is usually trivial.

In the near part of the spectrum of extremely large numbers, the ones constructed by tetration can be found. These numbers could compare with one another or with some fast growing functions, such as factorial, hypersquare and others. In this sense tetration is more interesting than the other faster increasing operations.

Tetration is well defined when its second argument (height) is integer. It is still an open question how to expand to rational, real and even complex heights. Some progress is achieved as in [1] and [2] and now calculations for some bases (first argument of tetration) are implemented with real heights.

Operation sequences are defined by their recursive rules plus their initial operations.

The recursive rule of multiple application of an operation # is:

\[ r = b \circ @ h = b \# b \# \ldots \ldots \# b \]

\[ \} \text{h operands} \]
\[ \} \text{h-1 operators} \]

The results differ when operations are performed left-to-right and right-to-left and # operation is not associative.

- The hyperoperation sequence.

The term hyperoperation can be used to denote operations that follow the classical three direct operations: addition, multiplication and exponentiation. Or hyperoperations can be all operations in a sequence. But there is one sequence that is called the hyperoperation sequence, defined by initial operations – addition (level n=1), multiplication (level n=2) and exponentiation (level n=3). Sometimes a level zero operation is considered – succession (or zeration), which is in reality an unary operation.
The recursive formula is (1) and it is applied right-associatively. The next hyperoperations are called tetration (level \( n=4 \)), pentation (level \( n=5 \)), hexation (level \( n=6 \)), and so on. Formally:

\[
H_n(b, h) = \begin{cases} 
  h + 1 & \text{if } n = 0 \\
  b & \text{if } n = 1 \text{ and } h = 0 \\
  0 & \text{if } n = 2 \text{ and } h = 0 \\
  1 & \text{if } n \geq 3 \text{ and } h = 0 \\
  H_{n-1}(b, H_n(b, h - 1)) & \text{otherwise}
\end{cases}
\]

Hyperoperations with integer arguments, factorization and divisibility are examined in [3].

- Lower hyperoperations start with addition or succession. The recursive formula is (1) and it is applied left-associatively. More formally:

\[
L_1(b, h) = b + h ; \quad L_{n+1}(b, 1) = b ; \quad L_{n+1}(b, h) = L_n(L_{n+1}(b, h - 1), b)
\]

Initial operations are succession (level \( n=0 \)), addition (level \( n=1 \)), multiplication (level \( n=2 \)) and exponentiation (level \( n=3 \)). The next operation is \(<\text{et}\>\) or lower4:

\[
L_4(b, h) = b^{b^{h-1}}
\]

- Commutative operations were submitted by Albert Bennett in [4]. This sequence includes addition and multiplication.

The recursive formula is:

\[
F_{n+1}(b, h) = \exp(F_n(\ln(b), \ln(h)))
\]

All operations are commutative:

\[
F(x, y) = F(y, x) \quad \forall x, y
\]

Some of the operations are:

- level \( n=0 \): \( F_0(b, h) = \ln(e^b + e^h) \);
- level \( n=1 \): \( F_1(b, h) = b + h \);
- level \( n=2 \): \( F_2(b, h) = b \cdot h = e^{\ln(b) + \ln(h)} \);
- level \( n=3 \): \( F_3(b, h) = e^{\ln(b) \cdot \ln(h)} \);
- level \( n=4 \): \( F_4(b, h) = e^{e^{\ln(b) \cdot \ln(h)}} \);

2. Definition of pisa operations

The pisa hyperoperations are a sequence of binary arithmetic operations including multiplication, exponentiation and lower4 – the left-associative repeated exponentiation. These are all members of the lower hyperoperation sequence. The pisa operation sequence is constructed by observing and using another relation that they share.

The first operation is multiplication, which is well defined. For consistency with the other hyperoperations and the Ackermann function, the level of the multiplication operation should be set at \( n = \log_2 2 \).

The sequence will be defined by its recursion formula and its initial operation.

2.1. Representing operations

Usually operations are represented with a sign or a symbol inserted between arguments. Because there is equivalence between binary operations and functions with two arguments, the former can be represented as functions. The proper usage of operations facilitates human perception and calculations.

Operational representation:

\[
re = ba \#_n \text{he} = ba < \text{ope} \#_n > \text{he}
\]

Functional representation:

\[
re = \text{ope}_n(ba, \text{he})
\]

In formulae (6) and (7) ope could be any binary (or ternary) operation.
The operands are:

- **ba** or **b** – first argument; known as base;
- **he** or **h** – second argument, degree of applying the operation on the base; known as exponent or degree or height;
- **le** or **n** – feature of the operation, reflecting the level of recursion; known as grade or rank or level;
- **re** or **r** – value of calculation; known as result or product;

### 2.2. The recursive formula

Pisa operations with integer levels greater than two are defined by the recursive formula:

\[
P_2(b, h) = b \cdot h \quad \text{// initial operation;}
\]

\[
P_{n+1}(b, h) = \left( \frac{b^h}{b} \right)^{P_n(b, h)}
\]  

(8)

Applying the recursive rule gives us:

\[
P_3(b, h) = b^h \quad \text{// exponentiation}
\]

(9)

\[
P_4(b, h) = b^{b^{h-1}} \quad \text{// lower4 = < cet >}
\]

(10)

and so on...

![Figure 1. Pisa operation. Every big circle is an exponent to the lower big circle. The little circles inside are the base arguments. The little figure on the top is the height argument. The walls of the tower represent the level argument.](image)

### 2.3. The opposite recursion

Pisa operations with integer levels less than two are defined by the recursive formula:

\[
P_2(b, h) = b \cdot h \quad \text{// initial operation;}
\]

\[
P_{n-1}(b, h) = \log_{b^h}(P_n(b, h)) = b \cdot \log_b(P_n(b, h))
\]  

(11)

Formula (11) yields result only when \( P_n \) is positive, which puts some limitations on the domain of **he** depending on **ba**.

Applying the recursive rule gives us:

\[
P_1(b, h) = b + b \cdot \log_b h
\]

(12)

This is the pisa summation operation. Height domain: \( h > 0 \).

and so on...

### 2.4. Ternary pisa function

All pisa operations can be combined in one function with three arguments, the third being the level of the operation. This is very much like the relation between the hyperoperation sequence \( H_n \) and the three-argument Ackermann function \( \varphi(m, n, p) \).

Ternary pisa hyperoperation (pho) function representation:

\[
pisa(ba, he, le) = pho(ba, he, le) = re = P_n(b, h)
\]  

(13)

### 2.5. Some properties of pisa operations

Changing the level of pisa operations:

\[
ba < pisa\#n + 1 > he = ba \frac{ba < pisa\#n > he}{ba} \quad \text{// the recursive formula (8)}
\]
\[ ba < \text{pisa}^n > \frac{1}{ba} = ba < \text{pisa}^n + 1 > 0 \]  
(14)

Right neutral number \( n n r = 1 \) – for every level \( n \):
\[ ba < \text{pisa}^n > 1 = ba \]  
(15)

One connection with small base tetration:
\[ ba < \text{pisa}^n > \frac{1}{ba} = (ba \frac{1}{ba}) < \text{tet} > (n - 2) \]  
(16)

CONJECTURE: formulae (8) and (14) ÷ (16) will hold for pisa operations with real levels (\( le \in \mathbb{R} \)).

### 2.6. Pisa sequence members

Multiplication and exponentiation are well established and well known.

- Multiplication represented as binary operation is \(<\text{mul}>\), and as pisa function is \(\text{pisa}(ba, he, 2)\).
- Exponentiation represented as binary operation is \(<\text{deg}>\), and as pisa function is \(\text{pisa}(ba, he, 3)\).

#### 2.6.1. Operation lower4

This operation may be called \textbf{chetvartaope}, named after the Slavic word for four, similarly to tetration. This way next lower hyperoperations may be called petaope, shestaope, and so on.

- Represented as binary operation it is \(<\text{cet}>\) or \(<\text{uer}>\).
- Represented as function chetvartaope is
  \[ \text{cet}(ba, he) = \text{pisa}(ba, he, 4) = ba^{ba \cdot he - 1} \]  
(17)

Domain in reals is \( ba > 0 \).

- Auxiliary functions of chetvartaope are defined:
  - chetvartaope with zero height
    \[ \text{coh}(x) = \text{cet}(x, 0) = x^{1/x} \]  
(18)

  - its inverse could be multivalued
    \[ \text{hoc}(x) \text{ is } x = \text{hoc}(x)^{1/\text{hoc}(x)} \]  
(19)
    \[ \text{hoc}(x) \text{ is with codomain (0; 1)} \]  
(20)
    \[ \text{hocs}(x) \text{ is with codomain (1; eta)} \]  
(21)
    \[ \text{hocl}(x) \text{ is with codomain (eta; +\infty)} \]  
(22)
    \[ \text{hcon}(x) \text{ may return negative rationals} \]  
(23)

Here the constant eta is \( \text{eta} := e^{1/e} = \eta \approx 1.444 667 8 \).

- Inverse operations of chetvartaope:
  \[ ba = \text{btec}(re, he) = (\text{hroot}(re^{he-1}))^{1/he-1} \]  
(24)
  \[ he = \text{h tec}(re, ba) = 1 + \log_{ba} \log_{re} \]  
(25)

- Changing the base:
  \[ ba < \text{cet} > he = x < \text{cet} > (1 + (he - 1) \log_x ba + \log_x \log_x ba) \]  
(26)

- Derivative:
  \[ (b < \text{cet} > h)' = b^{h-1} \cdot b^{h-2} [b'.(1 + (h - 1).\ln b) + h'.b.(\ln b)^2] \]  
(27)

#### 2.6.2. Operation pisa summation

This is the pisa operation with level one.

- Represented as binary operation it is \(<\text{sps}>\).
- Represented as function pisa summation is
  \[ \text{sps}(ba, he) = \text{pisa}(ba, he, 1) = ba.\log_{ba} (ba. he) \]  
(28)

- Domain in reals is \( ba > 0; ba \neq 1; he > 0; \).

Some properties:
\[ b < \text{sps} > 1 = b \]  
(29)
\[ b < \text{sps} > b = b + b = 2.b \]  
(30)
\[ b < \text{sps} > (1/b) = 0 \]  
(31)
\[ b < \text{sps} > (\frac{1-b}{b}) = 1 \]  
(32)
b < sps > (b^{1/b}) = b + 1 \tag{33}

b < sps > (b^x) = b \cdot (x + 1) \tag{34}

(h^x) < sps > h = \frac{x+1}{x} \cdot h^x \tag{35}

(b < sps > x) + (b < sps > (1/x)) = b + b = 2 \cdot b \tag{36}

(b < sps > x) + (b < sps > y) = b < sps > (b \cdot x \cdot y) \tag{37}

b < sps > (x \cdot y) = (b < sps > x) + (b < sps > y) - b \tag{38}

b < sps > ((b^{1/b})^{r-b}) = r \tag{39}

Inverse operations:

\[ ba = bsps(re, he) = basip(re, he, 1) \tag{40} \]

\[ he = hsp (re, ba) = ba \frac{re-ba}{ba} \tag{41} \]

3. Domains

Pisa operations with positive bases and positive integer levels greater than one are defined for any height. When lowering the level, the domain of the height shrinks, and depends on the base. For level le=1, the height domain is positive numbers. For level le=0, the height domain is he>1 when ba>1 and 0<he<1 when 0<ba<1.

Some negative numbers can be bases of some pisa operations. If \( b^{1/b} = coh(b) \) exists and is positive, all pisa operations with integer levels greater or equal to two are calculable. Negative rationals with odd numerators (in canonical form) have a value of \( b^{1/b} \). If the denominator is odd, the value is negative. If the denominator is even, the value is positive.

For integer levels of pisa, complex bases and heights are defined.

In this paper the focus will be on the first argument being real bases greater than one.

3.1. Extending domains

The recursive formula (8) defines pisa operations with integer levels le ≥ 2. Recursive formula (11) extends pisa for integer levels le < 2, with the cost of decreasing the domain of he. Bases and heights can be extended to complex numbers.

On positive bases pisa function is continuous over base and height arguments.

CONJECTURE: for positive bases and levels there should be some kind of continuity of pisa function on all three arguments.

Extending the levels of pisa to real numbers is interconnected with the problem of extending the heights of tetration to real values. This is somewhat similar to the problem of extending the factorial to Gamma function. Solving the tetration problem will grant us pisa functions with real levels (and vice versa – real levels of pisa will define real heights of tetration), as per formulae (16) and (58).

There is some progress in solving the tetration problem – a unique analytic solution is considered in [1]. Calculations are being performed with real heights for some bases. The tetration calculator [5], upheld by prof. William Paulsen of the Arkansas State University, calculates results for 2; e; 10; i; 1+i and other bases and any real or complex height.

4. Inverse operations

The inverse operations return as results the arguments of the examined direct operation. These are often called hyperroot and hyperlogarithm. When the level of operation is three, these are the normal root() and log() functions.

The notation used here is that the first argument of inverse operations is always the result of the direct operation.

Direct operation:

\[ re = pisa(ba, he, le) \tag{42} \]

The first inverse operation (the hyperroot) returns the first argument of the direct operation. The first inverse of the hyperroot operation is the direct one. The hyperroot function is usually with two
arguments because the level is known from the context, and it is omitted. Hyperroot is used for the hyperoperation sequence, so here the pisa analog is defined:

\[ ba = \text{basip}(re, he, le) \]  \hspace{1cm} (43)

The second inverse operation (the hyperlog) returns the second argument of the direct operation. The hyperlog function is usually with two arguments because the level is known from the context, and it is omitted. Hyperlog is used for the hyperoperation sequence, so here the pisa analog is defined:

\[ he = \text{hasip}(re, ba, le) \]  \hspace{1cm} (44)

The third inverse operation should return the third argument of the direct operation – that is the recursion level.

\[ le = \text{lasip}(re, ba, he) \]  \hspace{1cm} (45)

Some calculated values of inverse functions are collected in Table 1.

| Function | basip() | hasip() | lasip() |
|----------|---------|---------|---------|
| f(8; 3; 6.240) | 1.0 | | |
| f(8; 3; 5) | 1.788 | 1.417 | 1 < le < 2 |
| f(8; 3; 4) | 1.844 | 1.581 | 1 < le < 2 |
| f(8; 3; 3) | 2.0 | 1.893 | 1 < le < 2 |
| f(8; 3; 2.666) | | 2.0 | |
| f(8; 3; 2) | 2.666 | 2.666 | 2 < le < 3 |
| f(8; 3; 1.893) | | 3.0 | |
| f(8; 3; 1.581) | 1.218 | 6.240 | No value |
| f(8; 3; 1) | | | |

5. **Sibling numbers**

In the real domains ds \((1; e)\) and dl \((e; \infty)\) there are pairs of numbers: \(ts\) in ds and \(tl\) in dl, such that:

\[ Tt = ts^{1/ts} = tl^{1/tl} \]  \hspace{1cm} (46)

The only pair of integer siblings is \(\{ts=2; tl=4\}\).

When \(ts\) approaches one, \(tl\) approaches infinity.

When \(ts\) approaches \(e\), \(tl\) also approaches \(e\) – the natural base.

5.1. **Generating sibling numbers**

For a real number \(a\) (called generating number) greater than one, the pair of sibling numbers are calculated by these sibling functions:

\[ ts = ts(a) = a^{1/a-1} \]  \hspace{1cm} (47)

\[ tl = tl(a) = a^{a/a-1} \]  \hspace{1cm} (48)

For a generating number \(a\) approaching one, both sibling numbers approach \(e\) – the natural base.

A generating number \(d=1/a\) from the unit interval will generate the same sibling numbers in reverse order.

\[ d^{1/a-1} = tl \geq ts = d^{d/a-1} \]  \hspace{1cm} (49)

A complex generating number will generate a complex pair of siblings.

Some of the negative rational numbers can generate real numbers, and they yield a pair of one negative algebraic number and one positive algebraic number in the unit interval.
5.2. Properties of sibling numbers
For a pair of sibling numbers ts and tl with generating number a and for #=<pisa> any pisa operation (with any integer level with arguments within domains) it is fulfilled:

\[ tl = ts#a \] (50)

That is: \( tl = ts \cdot a = ts^a = ts^{ts^{a-1}} = ts + ts \cdot \log_{ts} a = \ldots = pisa(ts, a, \forall le) \)

\[ ts = tl\left(\frac{2}{a}\right) \] (51)

\[ ts = basip(tl, a) \] (52)

\[ a = hastip(tl, ts) \] (53)

\[ ts\#(a, x) = tl\#x \] (54)

Sibling ts and tl are a commutative pair for any pisa operation:

\[ re(a) = tl\#ts = ts\#tl \] (55)

Formulae (50) ÷ (55) are easy to prove by induction.

CONJECTURE: formulae (50) ÷ (55) will hold for pisa operations with real levels (le ∈ ℜ).

Variables a, ts, and tl have a relation that is fixed and does not depend on the level of the operation. These are pisa fixed points of a kind – they don’t depend on the iterations (the level of recursion).

6. Decomposition into two pisa operations
THEOREM 1: Every pisa operation can be broken into two (or more) nested pisa operations. The sum of the levels of the result operations exceeds the level of the original operation by two. The second operation is divided by the first argument (ba), and nested as a second argument of the first operation.

That is for every base, height and level in domains:

THEOREM 1:

\[ ba < pisa#A + B - 2 > x = ba < pisa#A > \left(\frac{ba < pisa#B > x}{ba}\right) \] (56)

In functional form:

\[ pisa(ba, x, A + B - 2) = pisa(ba, \frac{pis(a(ba, x, B))}{ba}, A) \] (57)

Note that the A and B values can be interchanged, or substituted with any values that sum up to \( A + B - 2 \). If we have pisa values for a given base and all heights with level \( le = B + \delta \), where \( B \) is integer and \( \delta \) is a fraction, then with Theorem 1 we can determine pisa values for levels \( le = B + 2\delta \), \( le = B + 3\delta \), \( le = B + 4\delta \), and so on.

CONJECTURE: Theorem 1 will hold for pisa operations with real levels (\( A, B \in ℜ \)).

This shift by 2 of the sum of levels is due to setting the level of multiplication: level(<mul>) = 2. With another setting of the level, another shift will be obtained.

6.1. Proving theorem 1 for integer levels
Considering mathematical induction for increasing A and for every B:

Step 1 – base case for \( A = 2 \)

\[ LHS = ba < pisa#B > x \]

\[ RHS = ba < pisa#B > x \]

\[ LHS = RHS \] for every B, so theorem (56) holds.

Step 2 – assumption
For \( A = k \) and for every B let \( LHS = RHS = D \).

\[ LHS = ba < pisa#B + k - 2 > x = D \]

\[ RHS = ba < pisa#k > \left(\frac{ba < pisa#B > x}{ba}\right) = D \]

Step 3 – induction step
For \( A = k + 1 \) and for every B

\[ LHS = ba < pisa#B + k - 1 > x \]

Using formula (8)
\[
LHS = (ba^{1/ba})^{ba < \text{pisa#B} + k - 2 > x} = (ba^{1/ba})^D \\
RHS = ba < \text{pisa#B} + k > \left(\frac{ba < \text{pisa#B} > x}{ba}\right)
\]

Using formula (8)
\[
RHS = (ba^{1/ba})^{ba < \text{pisa#B} > x} = (ba^{1/ba})^D \\
LHS = RHS = (ba^{1/ba})^D
\]

Concluding from this that \(LHS = RHS\) for every \(B\) and for natural \(A \geq 2\).

Considering mathematical induction for decreasing \(A\) and for every \(B\):

Step 1 – base case for \(A=2\)
\[
LHS = ba < \text{pisa#B} > x \\
RHS = ba < \text{pisa#B} > x \\
\]

\(LHS = RHS\) for every \(B\), so theorem (56) holds.

Step 2 – assumption
For \(A=k\) and for every \(B\) let \(LHS = RHS = D\).
\[
LHS = ba < \text{pisa#B} + k - 2 > x = D \\
RHS = ba < \text{pisa#B} + k > \left(\frac{ba < \text{pisa#B} > x}{ba}\right) = D
\]

Step 3 – induction step
For \(A=k-1\) and for every \(B\), and when \(LHS\) and \(RHS\) are defined
\[
LHS = ba < \text{pisa#B} + k - 3 > x \\
\]

Using formula (11)
\[
LHS = \log_{ba^{1/ba}} (ba < \text{pisa#B} + k - 2 > x) = \log_{ba^{1/ba}} (D) \\
RHS = ba < \text{pisa#B} + k - 1 > \left(\frac{ba < \text{pisa#B} > x}{ba}\right)
\]

Using formula (11)
\[
RHS = \log_{ba^{1/ba}} \left( ba < \text{pisa#B} + k > \left(\frac{ba < \text{pisa#B} > x}{ba}\right) \right) = \log_{ba^{1/ba}} (D) \\
LHS = RHS = \log_{ba^{1/ba}} (D)
\]

Concluding from this that \(LHS = RHS\) for \(ba^{1/ba}\) real, for every \(B\) and for integer \(A \leq 2\).

Altogether theorem 1 (formula (56) ) holds for every \(B\) and for all integer \(A\), when all arguments are within domains.

7. Connecting to tetration
Small base tetration is tetration with base in the real interval \((1; \eta)\), with eta: = \(e^{1/e}\). For real \(x>1\), \(coh(x) = x^{1/x}\) returns values only in that interval.

THEOREM 2: Every small base tetration operation can be broken into nested pisa and tetration. The sum of the levels of the result operations exceeds the level of the original operation by two. The tetration is divided by the first argument \((ba)\), and nested as a second argument of the pisa operation.

That is for every base, height and level in domains:

THEOREM 2:
\[
ba^{1/ba} < \text{tet} > (A + B - 2) = ba < \text{pisa#A} > \left(\frac{ba^{1/ba} < \text{tet} > B}{ba}\right)
\]

In functional form:
\[
tet(ba^{1/ba}, A + B - 2) = pisa(ba, \frac{1}{ba^{1/ba}} < \text{tet} > B, A)
\]
Note that the $A$ and $B$ values can be interchanged, or substituted with any values that sum up to $A+B-2$. The values should be within the domains though, as tetration is defined for heights $h > -2$, so $B > -2$ and $A + B > 0$ should hold.

CONJECTURE: Theorem 2 will hold for operations (pisa and tetration) with real levels ($A, B \in \mathbb{R}$).

CONJECTURE: Theorem 2 will hold for tetrations with bigger bases ($ba^{1/ba} \eta = e^{1/e}, ba \in \mathbb{C}$) and for complex bases.

7.1. Proving theorem 2 for integer levels

Considering mathematical induction for integer $A$ and for every $B$:

Step 1 – base case for $A=2$

$LHS = ba^{1/ba} < \text{tet} > (A + B - 2) = ba^{1/ba} < \text{tet} > B$

$RHS = ba. \frac{ba^{1/ba} < \text{tet} > B}{ba} = ba^{1/ba} < \text{tet} > B$

$LHS = RHS$ for every $B$, so theorem (58) holds.

Step 2 – assumption

For $A=k$ and for every $B$ let $LHS = RHS = D$.

$LHS = ba^{1/ba} < \text{tet} > (B + k - 2) = D$

$RHS = ba < \text{pisa}#k > \left( \frac{ba^{1/ba} < \text{tet} > B}{ba} \right) = D$

Step 3 – induction step for increasing $A$

For $A=k+1$ and for every $B$

$LHS = ba^{1/ba} < \text{tet} > (B + k - 1)$

Using tetration property

$LHS = (ba^{1/ba})^{ba^{1/ba} < \text{tet} > (B + k - 2)} = (ba^{1/ba})^D$

$RHS = ba < \text{pisa}#k + 1 > \left( \frac{ba^{1/ba} < \text{tet} > B}{ba} \right)$

Using formula (8)

$RHS = (ba^{1/ba})^{ba < \text{pisa}#k} \left( \frac{ba^{1/ba} < \text{tet} > B}{ba} \right) = (ba^{1/ba})^D$

$LHS = RHS = (ba^{1/ba})^D$

Step 4 – induction step for decreasing $A$

For $A=k-1$ and for every $B$, and when $LHS$ and $RHS$ are defined

$LHS = ba^{1/ba} < \text{tet} > (B + k - 3)$

Using tetration property

$LHS = \log_{ba^{1/ba}} \left( ba^{1/ba} < \text{tet} > (B + k - 2) \right) = \log_{ba^{1/ba}} \left( D \right)$

$RHS = ba < \text{pisa}#k - 1 > \left( \frac{ba^{1/ba} < \text{tet} > B}{ba} \right)$

Using formula (11)

$RHS = \log_{ba^{1/ba}} \left( ba < \text{pisa}#k \left( \frac{ba^{1/ba} < \text{tet} > B}{ba} \right) \right) = \log_{ba^{1/ba}} \left( D \right)$

$LHS = RHS = \log_{ba^{1/ba}} \left( D \right)$

Concluding from this that theorem 2 (formula (58)) holds for $ba^{1/ba}$ real, for every $B$, for integer $A$ and for arguments within domains.
8. Tetrational heights

Tetrational heights of pisa are such values of the second argument that can transform a pisa operation with integer level into a tetration with integer height. The base of tetration is $ba^{1/ba}$. Tetrational heights can be regarded as a sequence or as a set of numbers (or functions on $ba$).

$$pisa(ba, th_i, n) = tet \left( \frac{ba^{n + i - 2}}{b}, n + i - 2 \right)$$

(60)

The zeroth tetrational height $th_0$ transforms multiplication ($n=2$) into zero height tetration. Its value can be determined by applying formula (16).

$$th_0 = 1/ba$$

(61)

Tetrational heights are recursively defined:

$$th_{i+1} = ba^{th_{i-1}}$$

(62)

By applying formula (62) it could be found:

$$th_{-1} = 0$$

(63)

Tetrational heights (of tetration) are limited by -2, and so are pisa tetrational heights.

$$ba > 1: \quad th_{-2} \rightarrow -\infty \quad (64)$$

$$0 < ba < 1: \quad th_{-2} \rightarrow +\infty \quad (65)$$

The pisa tetrational heights set assumes the form:

$$Th\{ \pm \infty; 0; \frac{1}{b^b}, b^{\frac{1-b}{b^{1-b}}}; th_3(b); th_4(b); \ldots \}$$

(66)

An example is the set of tetrational heights for base $e$:

$$Th(e)\{-\infty; 0; 0.368; 0.531; 0.626; 0.688; 0.732; 0.765; 0.790; \ldots \}$$

with $th_\infty(e) \rightarrow 1^{-i}$.

If the value of a height of a pisa operation is not in the set $Th\{\}$, the result is not a tetration with integer height and base $ba^{1/ba}$. Thus tetrations are pisa operations with special heights ($he_{pisa} \in Th\{\}$) and bases ($ba_{pisa} = hoc(ba_{tet})$; $ba_{tet} = coh(ba_{pisa}) = b^{1/b}$).

Every tetration can be represented as a pisa operation. But not every pisa can be a tetration, even when using fractional heights.

9. Using theorems in calculations

Using tetration results from [5], some non-integer level pisa calculations can be implemented.

To find pisa result for real height and non-integer level

$$z^{1/2} = coh(z) = 2$$

$$z = hoc(2)$$

$$z_1 = 0.1379 + 0.2264. i = 0.2652. exp (1.0237. i)$$

$$z_2 = 0.8247 + 1.5674. i = 1.7711. exp (1.0864. i)$$

And more solutions exist. Calculating further with $= z_2$.

$$1/z = 0.2629 - 0.4997. i = 0.5646. exp (-1.0864. i)$$

$$\ln z = 0.5716 + 1.0864. i = 1.2267. exp (1.0864. i)$$

$$z^{1/2} = \exp(\ln z /z) = 2.000 + 0.000. i$$

Using continuity of tet(), tetration result multiple of z (with the same angle) can be found.

$$B = 1.791 + 3.700. i = 4.1107. exp (1.1200. i)$$

$$s = tet(2; B) = a. z = 1.0447. z$$

Using (59) for any $A$

$$pisa(z; s/z; A) = pisa(z; a; A) = tet(2; A + B - 2)$$

Results can be obtained by giving values to $A$.

$A= 2.209$

$$pisa(z; a; A) = pisa(0.824 + 1.567i; 1.044; 2.209) = tet(2; 2 + 3.7i)$$

$$= 0.8456 + 1.6478i = 1.8521. exp (1.0967. i)$$

$A= 2.209 - 3.700. i$
\[ pisa(z; a; A) = pisa(0.824 + 1.567i; 1.044; 2.209 - 3.700i) = \text{tet}(2; 2) = 4.000 \]

\[ A = 2.000 \]

\[ pisa(z; a; A) = a.z \]

\[ = 0.8615 + 1.6375i = 1.8503 \exp (1.0864i) = \text{tet}(2; A + B - 2) = \text{tet}(2; B) = s \]

\[ = pisa(z; 1/z; B + 2) \]

Two different pisas with the same bases produce same results:

\[ pisa(z; 1.044; 2) = pisa(z; 0.263 - 0.500i; 3.791 + 3.700i) \]

10. Increasing pisa level

Examining pisa functions in the real interval \([1; \infty)\) for domain \((ba)\) and codomain \((re)\), \(ba\) and \(he\) are fixed, \(le\) is increasing.

Let \(ts\) and \(tl\) be the siblings of \(ba\) \((ba=ts\) or \(ba=tl\)).

\[ Tt = \frac{ba^{1/ba}}{ts^{1/ts}} = \frac{tl^{1/tl}}{ts^{1/ts}} \]

There are 5 fixed points in the examined interval – shown on figure 2.

\[ re = ba = 1; \quad \text{repelling fixed point} \]
\[ re = ba = ts; \quad \text{attracting fixed point} \]
\[ re = ba = e; \quad \text{repelling fixed point} \]
\[ re = ba = tl; \quad \text{attracting fixed point} \]
\[ re = ba \rightarrow +\infty; \quad \text{repelling fixed point} \]

![Figure 2. Pisa operation plotted on the graph of \(b^{1/b}\).](image)

Determining current and next results:

\[ re = pisa(ba; he; le) \]
\[ re_n = pisa(ba; he; le + 1) = (ba^{1/ba})^{re} \]

If the value of \(re\) falls in \((1; ts)\) subinterval:

\[ 1 < re < ts \]
\[ re_n = (ba^{1/ba})^{re} \Rightarrow re_n^{1/re} = ba^{1/ba} \]
\[ re < ts \Leftarrow re^{1/re} < ts^{1/ts} = ba^{1/ba} \]
\[ \Rightarrow re < re_n \]
\[ r_{e_n} = (ba^{1/ba})^{re} = (ts^{1/ts})^{re} < (ts^{1/ts})^{ts} = ts \]

\[ 1 < re < re_n < ts \]

\[ 1 < re(Le = 1) < re(2) < re(3) < \cdots < re(\infty) = ts \]

When the level increases, the value of the pisa operation increases and stays within the subinterval. When the level approaches infinity, the value approaches ts and the attracting fixed point.

The same way the other subintervals can be examined, outcomes are displayed on figure 2 with arrows.

11. Comparing values of different operations

Given two binary operations – \(<\text{op}_F>\) and \(<\text{op}_S>\), with codomains in \(\mathbb{R}\), the less than relation of results is defined.

The area of domination of an operation is that part of the domains, where its result is bigger than the one of the other operation. If the base argument is plotted on the ordinate axis, and the height argument on the abscissa, the areas of domination of each operation can be determined. Example is shown in figure 3. The limits of the areas are found by the condition:

\[ ba < \text{op}_F \left( he \right) \]
\[ = ba < \text{op}_S \left( he \right) \]

\[ (67) \]

11.1. Comparing \(<\text{mul}>\) and \(<\text{deg}>\)

Both operations are defined when \(ba > 0\) and \(he \in \mathbb{R}\).

Considering (67) for multiplication and exponentiation

\[ ba^{he} = ba \cdot he \]

\[ (68) \]

the limits of the areas are

\[ | \]
\[ ba = he^{he^{-1}} \]
\[ he = 1 \]

\[ (69) \]

and when limit cases are included

\[ | \]
\[ ba = he^{he^{-1}} \]
\[ he = 1 \]
\[ ba = 0 \}
\[ he > 0 ; \]
\[ ba > 1 \}
\[ he \to +\infty ; \]
\[ ba \to +\infty \}
\[ he > 0 ; \]

\[ (70) \]

\[ Figure 3. Areas of domination of operations \(<\text{mul}>\) and \(<\text{deg}>\). \]
11.2. Comparing any pair of different pisa operations

Examining domination between any other pair of pisa operations, curiously, the same areas are found.

CONJECTURE: The limits of the areas of domination of any pisa operations with different levels are defined by (69) and (70).

For integer levels this conjecture is provable by induction. For fractional levels and bases $ba > 1$, considerations of increasing $pisa(ba, he, le)$ when level increases could be applied. Using formulae (47), (50) and (15), the same limits of domination areas can be proven.

Thus these areas of domination are invariant for pisa operations and are their feature.

12. Conclusion

Pisa operations are a new sequence. It is intersected with lower hyperoperations in three points – multiplication, exponentiation and cet(lower4). The relation with one operation of the hyperoperation sequence (namely tetration) has been revealed. Every tetration can be represented by a pisa operation, thus pisas are an extension of tetrations.

The sibling numbers are connected with the structure of pisa operations. They provide fixed points and could ease calculations. They are the appropriate tool to peek into real value levels of pisa and real value heights of tetration, which is the hot problem of the moment.

When the level arguments of pisa operations are integer, the results are well defined. Non-integer levels are a challenge. By extending pisa operations to real levels we acquire not just a sequence of operations, but a line of operations. Levels can be extended further to complex numbers.

Pisa operations are closely related to tetration and combined with sibling numbers could be a useful tool in that context. Pisa function can facilitate solving the tetration problem, and can give some meaning to tetrations with real heights, as a multitude of new operations will appear between addition, multiplication and exponentiation.

References

[1] Trappmann H and Kouznetsov D 2011 Uniqueness of holomorphic Abel function at a complex fixed point pair Aequationes Mathematicae 81:1 pp 65-76
[2] Paulsen W and Cowgill S 2010 Solving $F(z+1) = b^{F(z)}$ in the complex plane Mathematics of Computation 78(267) (doi:10.1090/S0025-5718-09-02188-7)
[3] Barrett R 2019 The Fundamental Theorems of Hyper-Operations (Preprint doi:10.20944/preprints201902.0176.v1)
[4] Bennett A 1915 Note on an Operation of the Third Grade Annals of Mathematics. Second Series 17 (2) pp 74–75
[5] Tetration calculator: http://myweb.astate.edu/wpaulsen/tetcalc/tetcalc.html