The Constituents of Sets, Numbers, and Other Mathematical Objects

Ruadhan O'Flanagan*

April 2019

The sets used to construct other mathematical objects are pure sets, which means that all of their elements are sets, which are themselves pure. One set may therefore be within another, not as an element, but as an element of an element, or even deeper, inside several layers of sets within sets.

The introduction of the term *constituent* to describe a set which is within a given set, however deep, induces an apparently novel partial order on sets, and assigns to any given set a diagram which specifies a directed graph, or category, herein dubbed its constituent structure, indicating which sets within it are constituents of which others.

Sets with different numbers of elements can have exactly the same constituent structure. Consequently, constituent structure isomorphisms between sets need not preserve the number of elements, although they are still injective, surjective, and invertible. We consider in detail an example of an isomorphism between a one-element set and a five-element set, which is a surjective mapping despite the mismatch in cardinalities.

The constituent structure of a set determines the mathematical objects for which the set is a suitable representation. Different schemes for constructing the natural numbers, such as those of von Neumann and Zermelo, generate sets with the same constituent structures. Objects share the constituent structures, not the elements, of the sets used to construct or represent them.

The requirement that an object's properties be faithfully encoded within a set's constituent structure and not its non-constituent characteristics such as its cardinality, when made explicit, dictates a specific and novel way of representing ordered pairs and tuples of sets as sets, providing simple formulae for addressing and extracting sets located deep within nested tuples.

The same requirement dictates extensions of the representations of natural numbers as finite sets to integers, arithmetic expressions, and rational numbers. Each of these has a specific constituent structure which is replicated by various finite sets, of which one unique set is the simplest. Aligning the constituent structures of sets with those of the numbers they represent produces a corresponding alignment of arithmetic operations with set operations, which implement the evaluation of arithmetic expressions using sets.

The set which most naturally encodes a rational number expresses it as a novel order-preserving form of continued fraction, which provides a new efficient algorithm for finding rational approximations to irrational numbers. It also uniquely extends the Stern-Brocot tree to all rationals, including zero and negative numbers, introducing new non-trivial symmetries.

Introduction

The subset relation, $\subset$, has the properties that make life easy and pleasant for those who wish to study sets. It is transitive, $a \subset b \;\&\; b \subset c \implies a \subset c$, which provides plenty of opportunities to draw deductions from given information, and it is antisymmetric, $a \subset b \;\&\; b \subset a \implies a = b$, which makes it possible for us to figure out what a set that we are interested in is equal to, which is generally at least a minor success.

The membership relation, $\in$, however, is not so pleasant to work with. It is not transitive, the sets it relates are never equal, and given a list of sets, $A, B, C, \cdots$, and membership relations

*rof@ruadhan.net

between them, $B \in C, A \in C, \cdots$, there is no way of combining the given statements that permits us to draw a conclusion.

A careful comparison of the two basic relations between sets reveals that we don't like membership very much and would like it to go away. The standard ridding procedure is to contend that membership doesn't express any information that isn't accessible through the subset relation, which makes membership a superfluous and irrelevant part of the formalism. We can define membership using the subset relation: $A \in B \iff \{A\} \subset B$, so subsets give us everything we need, we tell ourselves.

Unfortunately, when we do this, we lose the ability to reference the structure deep inside sets.

When we construct mathematical objects of interest, it typically involves gathering certain previously defined objects, $x_1, x_2, \cdots$, into a set, X. Then we equip that set with some structure, S, such as a function or a relation, by putting X and S together into an ordered pair, $P = (X, S)$, which is a specific type of set. We might combine two objects of that type, P_1 and P_2, with another structure, such as an isomorphism, M, to get an object, K, that we want to talk about, $K = (P_1, P_2, M)$.

The result is a set, K, which, deep inside, has sets whose internal structures replicate the properties of the objects $x_1, x_2, \cdots$, and larger sets which specify additional structure.

If somebody else constructs that set, K, and then gives it to us, and we try to understand what it is, using all the information that can be expressed in terms of the subset relation, so that we can list, as distinct symbols, all of the subsets of K, and say which of those are subsets of which others, we will be able to deduce that there are three things inside K and nothing further.

This disappointing achievement is formalized on a grand scale in category theory, in which sets are regarded as equivalent if there is an invertible mapping, or isomorphism, between them. These mappings are assumed to be functions, which assign to each element of one set a corresponding element of the other. A one-to-one correspondence of elements, which establishes that two sets have the same number of elements, renders the sets equivalent, and, within the category of sets, indistinguishable.

Despite this drastic loss of information about the contents of a set, category theory appears to be the right way to represent structure, effectively using directed graphs instead of sets as the most elementary objects, thereby putting relations on the same footing as the objects they relate, instead of building an object to encode a relation and placing that relation object into a set along with the objects it relates.

A category is a neatly self-contained chunk of structure which can be isolated from other structures and studied in isolation. The size, or cardinality, of finite sets provides an obvious example of information that can be studied on its own while everything else about those sets is forgotten. This is the information which is captured and probed by the category of sets.

This leads to the questions: What category contains the other information about the set, apart from its cardinality? What structure is left when a set's cardinality is taken from it?

We can access information deep inside a set using the membership relation, $\in$, but we can't use that relation to specify which dots in a graph depicting a category should have arrows connecting them, because a relation in a category must be reflexive, which means that every dot in the graph must have an arrow from itself to itself, and relations depicted by arrows must be transitive. $\in$ is neither reflexive nor transitive.

We also know that the $\in$ relation specifies all of the information inside a pure set. If we know its elements, and their elements and so on, all the way down to the empty set, then we know everything about the pure set. But part of this information, namely cardinality, is isolated by the category of sets, and when that's removed, the remainder must be less informative than $\in$.

So we seek another relation, which is transitive and reflexive, and discards some information.

That search leads to the constituency relation, and to the category of constituent structures.

Outline

The sets studied in this paper are all finite pure sets. A theory of the constituents of infinite sets requires a separate investigation to be undertaken after the finite case has been fully understood with clarity and certainty. For this reason, the construction of numbers given here ends with the rationals, since the construction of the real numbers requires the use of infinity.

Because every set considered is both pure and finite, it can be constructed in a finite number of steps, starting from the empty set and successively gathering previously generated sets together in a new set. There is consequently no need or opportunity to worry about whether the sets exist or are compatible with any specific system of axioms, other than those of pure finite sets.

In the first section, immediately following this, the constituents of a set and its constituent structure, along with the corresponding diagram, are defined and described.

Section 2 uses the different constructions of the natural numbers introduced by von Neumann and Zermelo as examples of constituent structures which are shared by apparently very different sets. The isomorphism between Zermelo's number 5 and von Neumann's number 5 is expounded as an example of an invertible mapping between sets with different numbers of elements, and a surjection from a one-element set to a five-element set.

Some operations on sets relevant to constituent structure are introduced and given notation in the section after that, and then section 4 shows how set operations relate to natural numbers in specific representations as sets. Subsequent reflections reveal the guiding principle for the construction of mathematical objects using sets, namely that the properties of objects should be encoded only within the constituent structure of the underlying sets.

The next section examines the representation within constituent structures of ordered pairs and tuples of sets. Formulae are developed for placing a set at a position within another, asserting that a set has something at a position or that one set is at a position within another, and extracting a set from a position arbitrarily deep inside tuples of tuples.

Attention then turns to the natural representations as sets of integers and expressions involving addition and subtraction, and how these operations on integers relate to corresponding set operations. Finally, we consider arithmetic expressions involving multiplication and division, and the natural representation of rational numbers as finite sets.

Contents

I. The Constituents of a Set

I. Definition and Notation

The foundational concept is most easily understood when the definition is given in words:

Definition: The constituents of a set are the set itself and the constituents of its elements.

To compile a list of a set's constituents, we first add the set itself to the list. Then we add the constituents of each of the set's elements. Those include, for each element, the element itself, and the constituents of its elements. So the set, its elements, their elements, and so on, are all included.

The choice of word is appropriate because this formal definition coincides with the existing usage of the verb, "to constitute": A thing constitutes itself, and, at the same time, its parts constitute it.

In symbols, the definition has the form:

$$x \lhd y \ \equiv \ x = y \text{ OR } x \lhd z \in y \tag{1}$$

where $\lhd$ is the symbol for "is a constituent of" and $\equiv$ indicates that the left hand side is defined by the right hand side. The condition $x \lhd z \in y$ means that *some* element, z, of y, has x as a constituent.

The symbolic definition can be read in words as: "'x is a constituent of y' is defined to mean that x is equal to y or x is a constituent of some element, z, of y".

The symbol $\lhd$ was chosen so that the expression $x \lhd y$ can be interpreted to graphically depict x as something small and pointlike at the leftmost tip of the triangle, which is included within the larger object y depicted by the side of the triangle opposite to x.

II. Properties

Constituency has the useful properties which the membership relation lacked.

• Transitivity

Each set in a membership chain such as $A \in B \in C \in D$ is an element of the next set, but is a constituent of every set farther down the chain. In this example, the sets satisfy $A \lhd B \lhd C \lhd D$ automatically, and from these relations we can deduce others such as $A \lhd C$ and so on. So we can draw conclusions from multiple instances of the $\in$ relation after all, but only if we have the $\lhd$ symbol to express them.

• Reflexivity

The fact that a set is a constituent of itself distinguishes constituency from a previous way of constructing a form of "transitive membership". That way involved the concept of a transitive set, which is a set that contains the elements of its elements. Obviously, the set itself could not be included among those.

The reflexive property, $x \lhd x$, is central to the concept of constituency. The constituency relation with reflexivity provides sets with the structure of a category, which essentially means that something has clicked into place. A large body of mathematical knowledge now automatically applies to collections of sets and their constituency relations.

• Antisymmetry

Our attention is restricted here to pure finite sets. Two such sets, x and y, which satisfy $x \lhd y \lhd x$, must clearly be equal to one another, because x cannot be inside a smaller part of itself.

However, it is worth briefly observing that without the restriction to pure finite sets, a proof that $x \lhd y \lhd x \iff x = y$ would require the axiom of regularity, which states that every non-empty set contains an element that is disjoint from itself. The truth of that axiom is not obvious even for finite sets, but the possibility of membership loops among sets can be eliminated with the more obvious statement that no set is a constituent of any of its elements, which is equivalent to the statement that constituency is antisymmetric.

• Partial Order

The three properties of transitivity, reflexivity and antisymmetry make constituency a partial order on sets. In that context, there is a least element - the empty set, $\varnothing \equiv \{\}$, which satisfies $\varnothing \lhd x$ for every set, x.

When we consider only the constituents of a specific set, y, there is also a greatest element, namely the set y itself. The ordering of the set's constituents can be shown in a diagram, with y at the top and the empty set at the bottom, revealing the internal structure of the set.

III. Constituent Structure

Formally, the constituent structure of a set is the category whose objects are the set's constituents and whose morphisms are constituency relations.

Informally, the constituent structure of a set is the specification of which of its constituents are contained as constituents in which others.

We can display this structure for a set, S, in a diagram consisting of distinct horizontal levels, constructed using the following procedure:

1. Assign unique symbols, $A, B, \cdots$, to each set which is a constituent of S. Sets which occur more than once get only one symbol - symbols are assigned to sets, not occurrences of sets.

2. Add the symbol S to the diagram. This is the top horizontal level.

3. For each symbol, X, at the lowest horizontal level, add the symbols of that set's elements one level below X.

4. For any symbol which occurs more than once in the diagram, remove all occurrences apart from a single instance at the lowest level.

5. Repeat steps 3 and 4 until every constituent has been added to the diagram exactly once. The bottom horizontal level will contain just one symbol indicating the empty set.

6. Draw one edge for each membership relation, connecting the symbol of a set to the symbol of each set which contains it as an element.

7. Remove any edges connecting symbols which are connected by an upward path of two or more edges. These are the edges showing membership, $A \in C$, for which a membership chain, $A \in B \in C$, exists.

Technically, this is the Hasse diagram of the set's constituents ordered by constituency.

The purpose of the diagram is to show constituency relations rather than membership. The final step of the procedure above removes membership information which isn't implied by the constituency relations.

A is a constituent of B if B can be reached from A in the diagram by following a path of zero or more edges upward.

Every set contains the empty set as a constituent, so every set's symbol in the graph will be reachable from the empty set at the bottom via an upward path of edges.

Similarly, every set is a constituent of the full set, S, and is connected to S by an upward path of edges.

As a simple example, we can consider the set $S = \{A, B\}$, with $B = \{A\}$ and $A = \varnothing = \{\}$. This set contains two elements, A and B, and three constituents, A, B, and S itself. S is at the top of the diagram and A, the empty set, is at the bottom, with B in the middle.

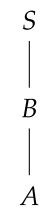


Figure 1

The constituent structure of the set $\{\varnothing, \{\varnothing\}\}$.

Despite the fact that A is inside S twice, once as an element of S and once as an element of B, it appears in the diagram only once. Also, A is an element of S, but there is no edge connecting A directly to S. There is no way to tell, from the diagram, whether A is an element of S or not. The diagram only shows that A is a constituent of S.

The constituent structure diagram doesn't show the full structure of S, including which constituents multiple levels below a set are also elements of it. That information has been removed because it is not part of the constituent structure[1].

Since we will often be interested in the constituent structures themselves rather than any specific sets with those structures, we may simply place a dot, $\bullet$, instead of a letter, at each position in the diagram.

In cases where we have a constituent structure, but no corresponding set, we can construct the simplest set with that structure by starting with the empty set, identified with the dot at the bottom of the diagram, and assigning to each dot at any horizontal level the set containing as elements the sets denoted by the dots below it to which it is connected by a single edge.

Figure 2

The membership graph of $\{\{\varnothing\}\}$ is the same as its constituent structure graph.

That is, the unique set whose membership graph is the same as a given constituent structure graph is the simplest set with that constituent structure.

II. Natural Numbers

I. Von Neumann's Construction

Different ways of constructing the natural numbers from pure sets have been proposed. Von Neumann[1] introduced the following system:

[1] The information which is missing from the constituent structure shown here is to be found in the category of sets, which knows the cardinality of each set shown, but nothing apart from that. The full information about the set, expressible using $\in$, has been separated into cardinality information, #, and constituent information, $\triangleleft$.

$$
\begin{aligned}
0 &\equiv \{\} \\
1 &\equiv \{0\} \\
2 &\equiv \{0,1\} \\
3 &\equiv \{0,1,2\} \\
4 &\equiv \{0,1,2,3\} \\
&\vdots
\end{aligned} \tag{2}
$$

When we draw the constituent structure graph for the set identified with the number 3, the fact that it has more than one element is not visible in the graph. The sets corresponding to 2, 1 and 0 are all constituents of the set for 3, and, because of this, the fact that they are also elements of that set provides no constituent information.

Figure 3

The constituent structure of von Neumann's construction of the natural number 3.

The information about the set identified with 3 that can be expressed in terms of the constituency relation is exhaustively specified by $0 \lhd 1 \lhd 2 \lhd 3$. It reveals that the set for 3 has at least one element, namely the set for 2, and at most three elements, since it has only three constituents apart from itself which could possibly be among its elements.

Somebody who knew the cardinality of each of these sets but didn't know their constituent structure would associate each of them with the number they are identified with; 3 has 3 elements and so on. From that point of view, this system of constructing the natural numbers successfully encodes the information in both the constituent structures of the sets and in their cardinalities.

II. Zermelo's Construction

Zermelo[2] used a different scheme for constructing the natural numbers:

$$
\begin{aligned}
0 &\equiv \{\} \\
1 &\equiv \{0\} \\
2 &\equiv \{1\} \\
3 &\equiv \{2\} \\
4 &\equiv \{3\} \\
&\vdots
\end{aligned} \tag{3}
$$

Apart from 0, each of these has only a single element, so somebody who only has access to cardinality information about a set would be unable to associate any of these sets with their corresponding number, apart from the empty set.

Considered in terms of constituent structure, however, each set is the simplest set with that structure. The edges in the constituent structure graph completely specify the membership relations between the sets. Zermelo's construction of the natural numbers is, in a sense, minimal, while von Neumann's is maximal, with each set containing as much as it possibly can.

From these observations, it is clear that, for each natural number, von Neumann's construction has the same constituent structure as Zermelo's construction of that number, despite the fact that the sets have different numbers of elements. The graph shared by both sets, for a natural number, N, always consists of a simple chain of $N+1$ constituents.

$$
\begin{tikzcd}
3 \arrow[d, dash] \\
2 \arrow[d, dash] \\
1 \arrow[d, dash] \\
0
\end{tikzcd}
$$

Figure 4

The constituent structure of Zermelo's construction of the natural number 3.

III. Constituent Structure Isomorphisms

When sets have the same number of elements, there is an invertible function which maps the elements of one set to the elements of the other set, establishing a one-to-one mapping between the sets. Those sets are "the same" in terms of cardinality.

In this case, Zermelo's sets and von Neumann's sets are "the same", but in a completely different way; they have the same constituent structures, but different cardinalities. There should therefore be an invertible map from one set to the other which preserves this structure in the same way that invertible functions between sets preserve the number of elements. It would not be a function between sets which puts their elements into one-to-one correspondence; instead it would map one constituent structure, depicted by the vertices and edges in the corresponding diagram, onto the other, sending vertices to vertices and edges to edges, in an invertible way.

This mapping, considered to send one graph to the other, is a directed graph isomorphism. When the constituent structure is considered as a category, it's an invertible functor between the categories for the two sets. In the category of constituent structures, it's an isomorphism between objects.

Like a function, such a mapping sends elements of sets to elements of sets, but in this case, they are not all elements of the same set. It's the constituents of one set which are mapped to the constituents of the other set in a one-to-one way.

Since every element of a set is a constituent, and the isomorphism specifies an invertible mapping between the constituents of the source and the destination set, every element in the destination set must have something mapped to it from the source set, and every element of the source set must be mapped to something in the destination set.

Figure 5 shows how the constituent structure of Zermelo's number 5 is mapped by an isomorphism, M, to the constituent structure of von Neumann's number 5, using subscripts of Z and V to denote Zermelo's construction and von Neumann's construction of each natural number.

$$
\begin{tikzcd}
5_Z \arrow[r] \arrow[d, dash] & 5_V \arrow[d, dash] \\
4_Z \arrow[r] \arrow[d, dash] \arrow[u, phantom, "\longrightarrow" near start, shift left=6] & 4_V \arrow[d, dash] \\
3_Z \arrow[r] \arrow[d, dash] \arrow[u, phantom, "\longrightarrow" near start, shift left=6] & 3_V \arrow[d, dash] \\
2_Z \arrow[r] \arrow[d, dash] \arrow[u, phantom, "\longrightarrow" near start, shift left=6] & 2_V \arrow[d, dash] \\
1_Z \arrow[r] \arrow[d, dash] \arrow[u, phantom, "\longrightarrow" near start, shift left=6] & 1_V \arrow[d, dash] \\
0_Z \arrow[r] \arrow[u, phantom, "\longrightarrow" near start, shift left=6] & 0_V
\end{tikzcd}
$$

Figure 5

An isomorphic mapping, M, from the constituent structure of Zermelo's construction of the number 5 to that of von Neumann.

The mapping is surjective, injective, and invertible, and also satisfies the condition:

$$a \lhd b \iff M(a) \lhd M(b). \tag{4}$$

Because 5_V has a larger cardinality than 5_Z, it has more *instances* of its constituents within it, including more elements. Multiple instances of a set are, however, literally the same thing. If $M(2_Z) = 2_V$, then every instance of 2_V within 5_V is the image of 2_Z, despite the fact that 5_Z contains only one instance of 2_Z.

This does not break the rule which says that a mapping cannot send one thing to more than one thing. That rule exists to ensure that nothing is ever mapped to two different things. The multiple instances of 2_V are not different things.

It would be a violation of that rule if two instances of the same set were to be mapped to two different destination sets. For example, in the inverse mapping, M^{-1}: $5_V \to 5_Z$, a set such as 1_V must be mapped to a single set, $M^{-1}(1_V)$. If two instances of 1_V, such as those which are elements of 2_V and of 3_V respectively, were to be mapped to different constituents of 5_Z, then it would truly be a case of one thing being mapped to two different things.

The mapping M^{-1}: $5_V \to 5_Z$ is therefore an injective mapping from a five-element set to a one-element set, just as M: $5_Z \to 5_V$ is a surjective mapping from a one-element set to a five-element set, which maps all of the constituents of 5_Z to all of the constituents of 5_V in a one-to-one way, despite there being 32 instances of the latter and only 6 instances of the former.

IV. A Constituent Structure Which is Not a Number

The set produced by von Neumann's construction of a natural number such as 5 is very different from the corresponding set in Zermelo's scheme, but they have the same constituent structure.

It appears that the information which is lost when the constituent structure is extracted from each of those sets is the specification of which scheme was used for the construction, while the information which is kept by the constituent structure is the specification of the object constructed.

It is reasonable to enquire whether the different constructions of a single number will appear to be the same thing or different things within the constituent structure of a set that contains both of them.

The simplest set containing two distinct representations of the same number is $\{2_Z, 2_V\} = \{\{1\}, \{0,1\}\}$. The constituent structure of this set is shown in figure 6. Neither of the sets, 2_Z and 2_V, are constituents of each other, but $1 = \{0\}$ and $0 = \{\}$ are constituents of both.

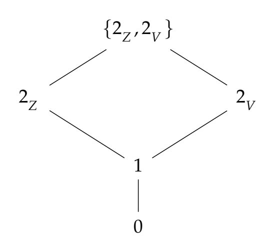


Figure 6
A set containing conflicting representations of the same natural number has a constituent structure which is not that of any natural number.

The fact that the set contains the two incompatible representations of the natural number 2 is clearly visible in the diagram of its constituent structure. The resulting graph is not isomorphic to the graph of any natural number.

This shows that the procedure of extracting the constituent structure from a set discards the information about which scheme was used to construct the natural numbers, as long as all occurrences of a natural number are encoded using the same set. Mixing one representation of a natural number with a distinct representation of the same number within a single set proves that something other than a consistent representation of the natural numbers was involved in the generation of that set.

The set $\{\{1\}, \{0,1\}\}$ is the smallest, simplest set with a structure distinct from any natural number. It can therefore be expected to play a role in the construction of other mathematical objects which are themselves distinct from natural numbers.

We will use the symbol $\diamond$ to refer to this set, and call it the diamond set, in reference to its structure which distinguishes it from natural numbers, although in contexts where it plays an important specific role, we may refer to the same set in a different way to clearly state the role that it plays.

III. Operations on Sets

When we consider a set's constituents, the obvious ways in which two or more sets can be used in combination to specify another set are not the familiar operations of union and intersection, which are defined in terms of elements.

I. Constituent Replacement

The primary operation in which constituents play the central role is constituent replacement. We use the notation:

$$x(y \to z)$$

to denote the set that results from replacing every occurrence of y in x with z.

This operation can be performed for any three pure finite sets, x, y and z. Any pure finite set can be expressed uniquely as a sequence of curly brackets, or braces, and commas[2], and this operation is a simple substring substitution in that representation.

It has properties which are expressible using the $\lhd$ relation:

$$y \lhd x \implies z \lhd x(y \to z) \tag{5}$$

$$\neg(y \lhd x) \implies x(y \to z) = x \tag{6}$$

$$\neg(y \lhd z) \implies \neg\left(y \lhd x(y \to z)\right) \tag{7}$$

where $\neg$ is logical negation, indicating that the statement following it is false.

Every finite pure set contains the empty set, $\varnothing = \{\}$, as a constituent. In the special case in which the set being replaced is the empty set, we use the notation:

$$x(y) \;\equiv\; x(\{\} \to y). \tag{8}$$

It has the properties:

$$x(\{\}) = x \tag{9}$$

$$\{\}(x) = x \tag{10}$$

$$y \lhd x(y) \tag{11}$$

$$x(y)(y \to \{\}) = x \tag{12}$$

$$x(y)(z) = x(y(z)). \tag{13}$$

The properties above show that this binary operation is associative, invertible, has an identity set, $\{\}$, and that it constructs sets related by $\lhd$. $x(y)$ can be thought of as "x on top of y", since its constituent structure diagram is x's diagram on top of y's diagram.

It's also appropriate and helpful to think of $x(y)$ as "x after y", since the procedure for constructing $x(y)$ starting from the empty set and successively enclosing sets within sets necessarily involves first constructing y, and then repeating the procedure for constructing x, but using y as the starting point instead of the empty set.

II. The Constituent Algebra

The set $x(y)$ actually satisfies a stronger condition than $y \lhd x(y)$. There are no occurrences of the empty set within $x(y)$ other than those inside an occurrence of y. We can denote this by $x(y) \vdash y$.

[2] Equivalent representations of this type can be sorted and the first among them chosen as the unique one.

This notation can be thought of as graphically depicting y as the horizontal line segment, x as the vertical segment, and $x(y)$ as the entire $\vdash$ symbol, which is x on top of y, displayed in the customary horizontal arrangement of symbols instead of vertically.

$\vdash$ can be defined as:

$$b \vdash a \;\equiv\; b(a \to \{\})(a) = b \tag{14}$$

which means that every occurrence of the empty set within $b(a \to \{\})$ corresponds to an occurrence of a within b.

When this condition, $b \vdash a$, is satisfied, there is some set, c, such that $b = c(a)$, which can be extracted from b with the operation $b(a \to \{\}) = c$, which "removes a from the bottom of b".

Similarly, when $b = c(a)$, the set a can be obtained from b and c by "removing c from the top of b". Consistency of notation suggests that we denote this as $(\{\} \leftarrow c)b = a$, and define it as:

$$(\{\} \leftarrow c)b \;\equiv\; \begin{cases} a & \exists a : b = c(a) \\ b & \text{otherwise.} \end{cases} \tag{15}$$

This symmetry within the notation further suggests that we describe the relation between x and $x(y)$ using the $\dashv$ symbol: $x \dashv x(y)$, which graphically depicts x as the horizontal line segment, y as the vertical one, and $x(y)$ as the entire $\dashv$ symbol, which consists of x on top of y when it is appropriately rotated to convert our left-to-right order of symbols into a top-to-bottom arrangement of constituent structures in a diagram, so:

$$c \dashv b \;\equiv\; \exists a : b = c(a) \tag{16}$$

provides the definition of $\dashv$.

Note that the reversed symbol, $\dashv$, does not denote the same relation as $\vdash$ with the symbols in reverse order: $b \vdash a$ does not mean $a \dashv b$. The horizontal line segment in both cases points at a symbol whose constituent structure diagram is at the bottom, $\vdash$, or the top, $\dashv$, of that of the other symbol. So $b \vdash a$ can be expressed in words as "a is the bottom of b", and $a \dashv b$ can be expressed as "a is the top of b".

The relations $\vdash$ and $\dashv$, like constituency, are partial orders on sets. They satisfy $\{\} \dashv x \vdash \{\}$ for all sets, x. The empty set is at the bottom of every set, x, because $x(\{\}) = x$, so $x \vdash \{\}$, and at the same time, it's at the top of x because adding it to the top of x leaves x unchanged: $\{\}(x) = x$, so $\{\} \dashv x$.

The notations for asserting that one set is at the top or bottom of another, and for removing one set from the top or bottom of another, lead to intelligible results, especially when we observe that associativity, $x(y(z)) = x(y)(z)$, allows us to unambiguously use the expression xyz:

$$xyz \vdash z \tag{17}$$

$$x \dashv xyz \tag{18}$$

$$xyz(z \to \{\}) = xy \tag{19}$$

$$(\{\} \leftarrow x)xyz = yz \tag{20}$$

$$(\{\} \leftarrow x)xyz(z \to \{\}) = y. \tag{21}$$

The construction of a set from many others usually involves the addition of new sets to an expression such as xyz on the left, to get a new set, $wxyz$, in which w is added to the expression after x is added. $wxyz$ can be read or thought of as w after x after y after z, and the sequence of steps involved in the set's construction is encoded in the expression from right to left.

Reading the expression $wxyz$ from right to left shows the order of its construction but conflicts with our conventions for reading and writing arithmetic expressions: $1 + 2 + 3$ is thought of as starting with 1 and then adding 2 and then 3. It is consistent with existing conventions for composition of functions, though: $f(g(h(x)))$ indicates that the functions should be evaluated starting with h followed by g and then f.

For finite pure sets in general or any other specific class of sets, we will refer to the expressions, operations, relations, and other sets that can be referenced, using the notation introduced here, as the constituent algebra of those sets.

III. Extracting Constituents

We can use the fact that $\lhd$ is a partial order to define an operation which selects the biggest constituents of a set.

For a set, S, we can say that a set, m, is maximal in S if $S \neq m \lhd S$ and $m \lhd y \lhd S \Rightarrow y = m$ or y = S.

That is, a set is maximal in S if there are exactly two distinct constituents of S which contain it as a constituent, namely S and itself.

We can write $\max_{\lhd}^{\{\}}(S)$ to denote the set which contains as elements all the maximal constituents of S.

In cases where there is only one maximal constituent we can call it the unique maximum and denote it by $\max_{\lhd}$. For example:

$$\max_{\lhd}(\{S\}) = S. \tag{22}$$

We don't need to resort to cardinality information and $\in$ to detect and extract the single maximal constituent from the set containing it. The set containing y is $\{\varnothing\}(y) = \{\varnothing\}(\varnothing \to y) = \{y\}$, so we can extract the element y from the set $\{y\}$ using $(\{\} \leftarrow \{\varnothing\})\{y\} = y$. We can detect whether there is only one element in a set, x, by seeing if $\{\varnothing\} \dashv x$.

In place of the intersection of two sets, the natural operation in this case is the selection of the Largest Common Constituents of two sets:

$$\mathrm{LCC}^{\{\}}(\mathrm{a},\mathrm{b}) \equiv \max_{\lhd}^{\{\}}\{\mathrm{x} \lhd \mathrm{a} : \mathrm{x} \lhd \mathrm{b}\}. \tag{23}$$

When there is only one of these, we can call it the Largest Common Constituent:

$$\mathrm{LCC}(\mathrm{a},\mathrm{b}) \equiv \max_{\lhd}\{\mathrm{x} \lhd \mathrm{a} : \mathrm{x} \lhd \mathrm{b}\}. \tag{24}$$

Finally, one can extract from a given set, a, the maximal constituents with another set, b, at the bottom:

$$a_{\vdash b}^{\{\}} \equiv \max_{\lhd}^{\{\}}\{c \lhd a : c \vdash b\} \tag{25}$$

and when there is only one, we can denote it by:

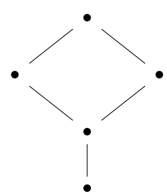
$$a_{\vdash b} \equiv \max_{\lhd}\{c \lhd a : c \vdash b\} \tag{26}$$

and the constituents with b at the top can be extracted in a similar way, but without using $\max_{\lhd}^{\{\}}$:

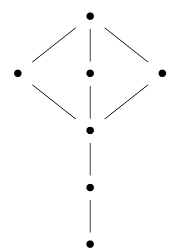
$$a_{b\dashv}^{\{\}} \equiv \{c \lhd a : b \dashv c\} \tag{27}$$

with the corresponding notation introduced to specify the unique constituent with b at the top when there is only one: $a_{b\dashv} \equiv (\{\} \leftarrow \{\varnothing\})a_{b\dashv}^{\{\}}$.

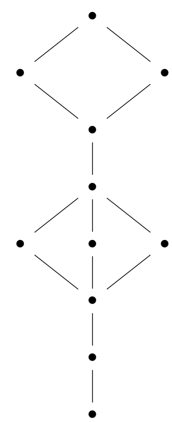
Table 1
Set Operations Used to Join Constituent Structures Together



Constituent structure of a



Constituent structure of b



Constituent structure of $a(b)$ as well as $a(* \to * \cup b)$

IV. Natural Number Arithmetic and the Guiding Principle

The sets, n_Z, that Zermelo identifies with natural numbers satisfy the following relation:

$$n_Z(m_Z) = n_Z + m_Z \tag{28}$$

meaning that the set representing the sum of two natural numbers can be obtained by inserting the set representing one number into the set representing the other, replacing the empty set.

The simplest non-trivial example is given by $1_Z(1_Z)$. The set representing 1 is $1_Z = \{\varnothing\}$, which contains just the empty set, $\varnothing$, which represents zero. Replacing $\varnothing$ with 1_Z yields $1_Z(1_Z) = \{1_Z\} = 2_Z$. The set substitution operation implements addition of natural numbers.

This establishes a simple correspondence between an elementary operation on sets and an elementary arithmetic operation on natural numbers.

The same relation is not true for von Neumann's construction. In that construction, each number is represented by a set with that number of elements. 2_V has two elements, so regardless of which set replaces the single empty set inside the representation $1_V = \{\varnothing\}$, the result of that replacement will always be a set with a single element, which can never be 2_V.

There is a different operation on set constituents which results in addition of natural numbers in von Neumann's representation. Replacing every constituent, a, of n_V with $a \cup m_V$ produces the set $(n+m)_V$. This can be expressed using the notation $n_V(* \to * \cup m_V) = (n+m)_V$.

Although the set operations which implement addition are different in the two cases, they produce the same effect on the constituent structures. For any two sets, a and b, the diagram showing the constituent structure of $a(b)$ consists of the diagram for a on top of the diagram for b, with the dot or symbol for the null set at the bottom of a's diagram identified with the dot or symbol for b at the top of b's diagram. The diagram for $a(* \to * \cup b)$ is exactly the same.

This is shown in table 1 for two hypothetical sets, a and b, with distinct constituent structures.

It is worth observing that both the sets and the set operations are considerably more complicated in the case of von Neumann's construction of the natural numbers, due to the double burden of making the number visible in both the set's cardinality and its constituent structure.

The sets, n_Z, used by Zermelo, on the other hand, are the simplest sets with the structure needed to represent the natural numbers. Their membership graphs, which specify everything about them, are identical with their constituent structure graphs, so they contain no information other than the structure of the natural number they represent.

The fact that addition of natural numbers is so simply implemented by a basic set operation, $n_Z(m_Z) = n_Z + m_Z$, when constituent structure alone is used to replicate the structure of natural numbers, is a hint that this is the right way in general to construct and represent mathematical objects using sets[3].

This hint prompts us to observe that natural numbers are the only structures which it is possible to encode in the cardinalities of finite sets. Anything more complicated, such as a negative integer or an ordered pair, will need to be encoded in a set's constituent structure, since the cardinality is always a natural number.

In fact, mathematical objects in general have structures, while sets have constituent structures, elements and cardinality. When we construct an object from sets and subsequently abstract from the underlying set to get the object as an object rather than as a set, the resulting object has structure but no elements or cardinality. The constituent structure of a set is the structure which survives within the constructed object when that abstraction occurs.

This gives us a guiding principle when constructing mathematical objects in general from sets:

- The structure of the object must be encoded solely in the set's constituent structure.

Everything else will be lost when we forget that the underlying object is a set.

V. Ordered Pairs and Tuples

I. The Standard Construction of an Ordered Pair

First introduced by Kuratowski[3], the definition of an ordered pair of sets as:

$$(a, b) = \{\{a\}, \{a, b\}\} \tag{29}$$

is universally accepted today for good reason.

It is extremely simple, intuitive, and it specifies the order of a and b successfully for all sets. Its structure clearly encodes the appropriate concept: First a, then b.

We bring new requirements, though:

- The set (a, b) should have a constituent structure which contains within it the constituent structures of a and b, in the correct order and separately retrievable.

- Given the set (a, b), it should be clear from its constituent structure diagram that its structure is that of an ordered pair of two sets.

[3] In the category whose objects are constituent structures and whose morphisms are constituent structure homomorphisms, this addition procedure corresponds to the sum of the two objects defined in terms of universal properties, while the product $x \star y$ is the structure obtained by replacing every edge in x with a copy of y, identifying the top and bottom vertices in y's diagram with the vertices at the top and the bottom of the edge in x. This operation implements multiplication of natural numbers, $n \star m = n \times m$, and results in a set only when a procedure for constructing a set from the resulting constituent structure is specified. The simplest set with that structure is the natural choice, which makes sums and products of Zermelo's natural numbers coincide with sums and products of constituent structures of sets in general.

So we are considering the case when we can't see the elements of (a, b), only the constituents. We have access to the information regarding (a, b) expressible in terms of $\lhd$, which we can display in a diagram, and from that information, we need to be able to reconstruct the diagrams for a and b and in addition determine which of the two is first and which is second in the pair.

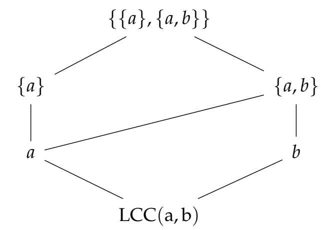


Figure 7
When neither a nor b are constituents of each other, their order and structures can be retrieved from the Kuratowski pair's structure.

One indication that Kuratowski's definition might not be sufficient for us can be seen from the fact that, in order to tell which of the two sets in $\{\{a\}, \{a, b\}\}$ contains the first set in the pair, we use the cardinalities of the sets, choosing the set with one element. Cardinality information is exactly what we need to avoid using.

The shape of the constituent structure diagram for $\{\{a\}, \{a, b\}\}$ will depend on the details of a and b. In the best case, neither entry in the pair will be a constituent of the other, and the resulting constituent structure is shown in figure 7. The unknown internal structures of a and b are omitted from the diagram, and it has been assumed for simplicity that they have a unique largest common constituent, whose structure is also omitted.

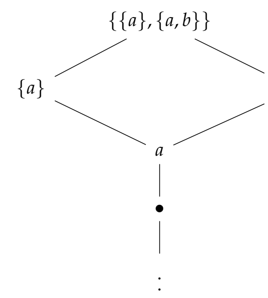


Figure 8
When $b \lhd a$, a's structure can be recovered, but b could be any constituent of a other than a.

In this case, it's clear from the fact that a is within both elements of the pair's set, while b is in just one, that a is the first entry and b is the second.

It's also possible to recover the constituent structure of a from the graph; it includes a and everything which can be reached from a along a downward path of edges. b's structure can be recovered in the same way.

Next, consider the case when $b \lhd a$. a can be identified as the set contained in both elements of the set representing the ordered pair, but b is among the constituents of a, with no way to tell from the diagram which of those constituents of a corresponds to b.

There is only one case in which the constituent of a which is equal to b can be deduced, specifically if a has only one constituent apart from itself, which would have to be the empty set. So the Kuratowski ordered pair has a unique pair of sets, namely $b = \varnothing$ and $a = \{\varnothing\}$, which it is able to represent as arranged in a specific order, $(\{\varnothing\}, \varnothing)$.

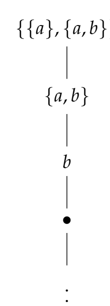


Figure 9
When $\{a\} \lhd b$, only b's structure can be recovered, and the diagram's structure does not indicate that it encodes an ordered pair.

This uniquely representable ordering of the two simplest possible sets, $0 = \{\}$ and $1 = \{0\}$, is encoded in the set $\{\{1\}, \{1, 0\}\} = (1, 0)$, which is the $\diamond$ set encountered previously, the simplest set which does not have the structure of a natural number.

The Kuratowski pair also fails to encode both entries in the pair when $\{a\} \lhd b$. b is then recoverable from the constituent structure since it is the vertex in the third horizontal layer, but a is not specified by the diagram, and it is also not evident from the diagram that it encodes an ordered pair.

II. Construction of an Ordered Pair Which Respects Constituent Structure

From these examples it is clear that a different way of encoding ordered pairs in a set is necessary in order to encode arbitrary constituent structures in a specific order in another constituent structure in a way that allows them to be retrieved.

One difficult case which indicates the general form that the solution must have is the case when one of the sets is several layers deep and the other is the empty set. The empty set is always at the very bottom of the graph, with a single upward connection to $\{\varnothing\}$.

The only way for the diagram itself to point at a specific vertex is by connecting an edge to it, but we need to be able to see that edge and distinguish it from the edges which form part of the structures of the entries in the pair.

There is no way for the shape of the diagram in higher layers to connect such a distinguished edge to the empty set, since all paths to the empty set go through $\{\varnothing\}$.

From this, we can conclude that a vertex higher up in the graph will need to act as a proxy for the empty set. If the first entry in the pair, a, is the empty set, $\varnothing$, then there would need to be a proxy vertex, $v_\varnothing$, which is not a constituent of the second entry, b, so that an edge from the highest levels of the graph can connect to $v_\varnothing$ without travelling through any part of b, to indicate that $v_\varnothing$ is the proxy for the first entry in the ordered pair.

These considerations lead to an inevitable conclusion: a must be placed on top of something, v_a, resulting in the set $a(v_a)$, and b must be placed on top of something else, v_b, resulting in $b(v_b)$, in a way that makes it impossible for $a(v_a)$ to contain v_b or for $b(v_b)$ to contain v_a.

This requires the use of two sets, v_a and v_b, whose constituent structures are incompatible with the possibility of either one being a constituent of a set with the other at the bottom.

We have an example of a set, denoted by $\diamond$, with a structure that can never be isomorphic to a natural number. So if a is placed on top of that set, and b is placed on top of a natural number, such as 3_Z, then neither of the resulting sets, $a(\diamond)$ and $b(3_Z)$, can contain the other.

This scheme only works for ordered pairs: With an ordered triple, (a, b, c), c can't be placed on top of a natural number, since it could then contain or be contained in $b(3_Z)$, and it can't be placed on top of a diamond, since it could then contain or be contained in $a(\diamond)$.

The general solution, which, in the case of an ordered tuple, $(x_0, x_1, \cdots, x_k)$, generates corresponding sets which are guaranteed not to contain each other, is to place each set on top of $\diamond$ on top of a distinct natural number, generating the set $x_n(\diamond(n))$ for the n^{th} entry.

In this context, $\diamond$ takes on the role of a position indicator, and its role is more clearly comprehensible when it is denoted by the word Position.

With this notation, the ordered pair with a in the first position and b in the second position is successfully encoded in the constituent structure of the set:

$$(a, b) = \{a(\text{Position}(0)), b(\text{Position}(1))\}. \tag{30}$$

The constituent structures of the two sets used here are shown in figure 10.

III. Extracting and Addressing Sets Inside Tuples of Tuples

A set, T_0, containing several distinct sets, $s_0, s_1, \cdots, s_k$, in their respective positions in that order, is:

$$T_0 = \{s_0(\text{Position}(0)), s_1(\text{Position}(1)), \cdots, s_k(\text{Position}(k))\} \tag{31}$$

which successfully encodes the constituent structures of those sets within its own, along with their order, which provides a construction of the (k+1)-tuple $(s_0, s_1, \cdots, s_k)$ as a set.

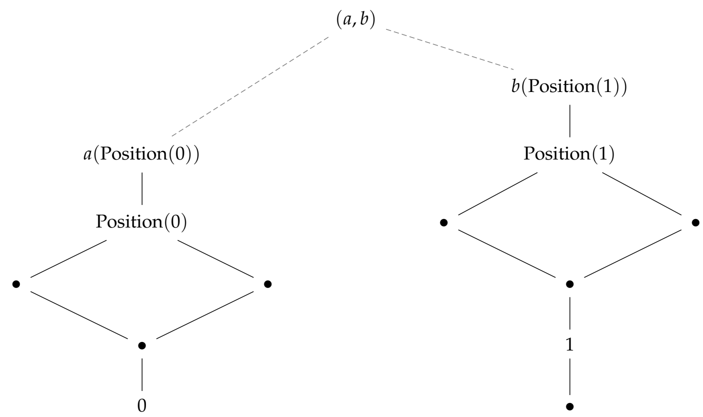


Figure 10
The sets $a(\text{Position}(0))$ *and* $b(\text{Position}(1))$ *encode the constituent structures of a and b in their specified positions with no possibility that one could be a constituent of the other. The constituent structure diagrams of the two sets are shown side-by-side for clarity, rather than as constituents of a single set containing them both. Grey dashed lines indicate edges that would connect the two sets to a set that did contain them as elements. The constituent structure diagram for such a set would fuse the bottom three vertices of the two diagrams shown here together.*

That tuple may then be included within another tuple, T_1. For example, it might be put at the second position of an ordered pair after a different set, U:

$$T_1 = \{U(\text{Position}(0)), T_0(\text{Position}(1))\}. \tag{32}$$

The set $T_1(\text{Position}(1))$ is T_1 with all of the empty sets inside it replaced by Position(1). Those empty sets inside T_0 are deep within its elements, such as $s_0(\text{Position}(0))$. Replacing the empty sets inside $s_0(\text{Position}(0))$ results in $s_0(\text{Position}(0)\text{Position}(1))$, which is therefore an element of $T_0(\text{Position}(1))$ as well as a constituent of T_1.

If the tuple, T_1, is subsequently positioned within another tuple, which in turn is positioned within another, and so on, the resulting $(n+1)$-dimensional tuple, T, will contain the constituent:

$$s_0(\text{Position}(p_0)\text{Position}(p_1)\cdots\text{Position}(p_n))$$

where p_j is the position in the j^{th} tuple of the previous tuple for $j > 0$.

If we introduce the notation $\text{Position}(p_0, p_1, \cdots, p_n)$ to refer to the multi-dimensional position set $\text{Position}(p_0)\cdots\text{Position}(p_n)$, then we can assert that s_0 is at that $(n+1)$-dimensional position within the $(n+1)$-dimensional tuple, T, with the expression:

$$s_0(\text{Position}(p_0, p_1, \cdots, p_n)) \lhd T. \tag{33}$$

Every constituent of s_0 satisfies the same condition, so this actually asserts that s_0 is a constituent of the object at that position. This means that we can assert that T contains *something* at that position with:

$$\text{Position}(p_0, p_1, \cdots, p_n) \lhd T. \tag{34}$$

To extract the full set at that position, we can take the maximal constituent of T with that position at the bottom:

$$T_{\vdash \text{Position}(p_0, p_1, \cdots, p_n)} = s_0(\text{Position}(p_0, p_1, \cdots, p_n)). \tag{35}$$

Equation 35 also allows us to assert that s_0 is the full set at that position and not just a constituent of that set.

This gives us s_0 on top of its multi-position set. To retrieve s_0 in its original form, we can remove the position set from the bottom of the constituent of T:

$$s_0 = T_{\vdash \text{Position}(p_0, p_1, \cdots, p_n)}(\text{Position}(p_0, p_1, \cdots, p_n) \to \{\}). \tag{36}$$

IV. Observations

The construction of ordered pairs and tuples given here has a number of desirable properties, one of which is that it is essentially the only solution. Minor adjustments, such as starting with position 1 instead of 0, or using a more complicated not-a-number object in place of $\diamond$, are possible, but any solution which works for all sets will involve placing the entries in the tuple on top of sets with non-isomorphic structures, of which the ones given here are the simplest. This is dictated by the requirement that the order and constituent structures of the entries should be retrievable from the constituent structure of the tuple.

The result of adherence to that requirement is a positional representation which closely matches our existing concept of an ordered pair: It's a structure containing two things, one of which is in the first position and the second of which is in the second position[4].

The other benefit is that the expressions needed to construct the ordered pairs and even multi-dimensional tuples, to address the positions within them and the objects at those positions, and to extract the objects from them, are extremely simple and also closely match our own conceptual understanding of what is involved in each case.

Like the case of addition for natural numbers, when the structure of each object is replicated solely by the constituent structure of the underlying set, the operations and relations natural to the objects represented coincide with simple relations and operations on sets.

It is worth bearing in mind that we were constrained to this form of solution because it was necessary to be able to encode and retrieve arbitrary sets. For sets of a specific type, such as those which encode natural numbers in Zermelo's encoding, easier methods of encoding ordered pairs of those sets may exist.

In fact, the construction of multi-dimensional tuples given here specifies a way to encode finite sequences of natural numbers. The multi-dimensional position set, $\text{Position}(p_0, p_1, \cdots, p_n)$, is essentially a finite sequence of natural numbers with a non-number object, $\diamond$, situated between each and the next: $\text{Position}(p_0, p_1, \cdots, p_n) = \diamond p_0 \diamond p_1 \cdots \diamond p_n$.

In this context, the $\diamond$ set plays the role of a comma, delimiting the numbers which specify a sequence of coordinates.

That set itself is a unique representation of two specific objects in a specific order. It's the Kuratowski ordering of 1 followed by 0, and these are the only sets already ordered by constituency which can unambiguously be given a Kuratowski ordering within a constituent structure, and it can only be this order: 1, then 0.

VI. Conceptual Introduction to the Encoding of Arithmetic Within Sets

The following sections apply the concept of constituent structure, and the necessity of encoding mathematical objects within it, beyond the natural numbers, to integers, arithmetic expressions, and rational numbers.

[4] One possible variation of this method is to use the entries themselves instead of numbers to specify the order. For example, when a is not empty, $\{\{a\}, b\{\varnothing, a\}\}$ places b after a rather than in position 1, and b can be retrieved by querying for the constituent with $\{\varnothing, a\}$ at the bottom, giving this the structure of a linked list rather than an ordered pair.

To avoid confusion regarding what exactly is being done and why, it is worth explaining in advance how the current project differs from what has been done before, in previous instances in which sets were used to define products, sums and other constructions involving numbers and operations on them.

One primary difference is that in previous cases, the purpose of defining numbers, and their operations, using sets, was to use set theory to provide an unambiguous construction of the numbers that mathematicians used in practice, so that no dispute or confusion can ever arise regarding what a number actually is, or whether the type of existence that a number has is sufficient to justify its use in a given circumstance without introducing possible contradictions.

In those cases, it didn't matter precisely what sets were chosen to represent specific numbers. As long as some collection of sets could be found which had all the needed properties, arithmetic, along with the rest of mathematics, was on solid ground.

When it came to defining the sum, product, or difference of two numbers, when each number had already been assigned a set to represent it, the task at hand was to specify a procedure which a mathematician could follow, starting from the two sets representing the two numbers, to identify a third set, and that third set would have to be the set which represents the number reached by applying the arithmetic operation to the two numbers.

When this has been achieved, the product, sum or difference of any two numbers can be referenced with confidence and without any ambiguity, because it refers to a specific set and can no longer be interpreted in a way that might cause doubt to arise. $^1/_7$ has a decimal expansion which never terminates, and so does $^1/_{13}$. Adding these numbers by using the digits in their decimal expansions is a process which never finishes, so doubt can arise if we don't have a clear definition of the sum of two numbers, and one of the great achievements of twentieth century mathematics was to banish such doubt and confusion forever by grounding all of mathematics on the solid foundation of set theory.

With the current project, we start with the observation that we have a new way to understand the structure within a set, which allows us to draw a diagram that reveals what type of mathematical objects the set can represent. It discards the details of the set that belong only to sets - elements, subsets and cardinality, and displays the structure that remains when we remember what objects went into its construction, and what objects were used to construct those objects, but forget which sets we chose to represent those objects.

We also have a principle which tells us whether some way of encoding other mathematical objects within sets will lead to simple formulae and conceptual clarity later on, or will lead to complexity and confusion. For a number or other mathematical object that we wish to construct or represent using sets, we can choose the simplest set whose constituent structure matches that of the object, and the resulting set will have properties which correspond exactly to those of the object.

Along with these new instruments, we have a new algebra of sets, which was made visible by the concept of a constituent, and which operates specifically on constituent structure. We can construct a new set by replacing specific constituents within a set with different sets. Successively replacing the empty set, which every set contains, with other sets, allows us to construct or specify a complicated structure using a string of symbols such as xyz denoting simpler sets used in the construction. The operation is associative and invertible, and there is a set, the empty set, which has no effect when added to the left or the right of such a string of sets.

A promising sign of the usefulness of this new algebra is given by the simplest sets with the constituent structure of the natural numbers, for which this basic set operation coincides with addition: $n(m) = m(n) = n + m$. We regard this as an indication that those sets are not just good enough to serve as representatives of the natural numbers, but that the natural numbers are *naturally* encoded by those sets. The arithmetic algebra of the natural numbers is the same as the constituent algebra of those sets, not by happy coincidence, but because each of those sets has exactly the same structure as the corresponding natural number. This brings the set algebra and the arithmetic algebra into alignment.

As we saw in the case of ordered pairs and tuples, other structures, more complicated than simple counts, can be constructed using the constituent substitution operation, and result in simple formulae which are conceptually clear.

So we now direct our attention to numbers with more complicated structure than a simple count, namely the integers, and, later on, the rational numbers. Unlike previous projects in which sets were found to represent the integers, we are not trying to find any collection of sets which can be put into perfect correspondence with integers and in terms of which the arithmetic of integers can be defined.

Instead, we are trying to find the sets which have exactly the same structure as the integers. We conjecture that, like the natural numbers, each integer has a constituent structure that specifies all of its properties and nothing else, and that when the simplest set with the same constituent structure is identified for each integer, the constituent algebra of those sets will coincide with the arithmetic of integers.

It is certainly true that addition of natural numbers expressed in unary is the only arithmetic operation for which the result of performing the operation is always identical with the expression for it: adding 111 to 11 gives 11111 when 1 is the only digit, and concatenation is then identical with addition. This is why $n(m) = n + m$. In the case of operations such as multiplication, division, and subtraction, there will necessarily be a distinction between an expression for an operation and the result of the operation.

So unlike previous cases where products of numbers were defined using sets, in which there was a set for each of the numbers to be multiplied and a third set for the number that resulted from the multiplication, we will have four sets, with the additional set representing the arithmetic expression which designates the multiplication, and a procedure for evaluating the expression to reach the result.

We also depart from previous practice by imposing on ourselves the requirement that no choice of ours may be introduced into the doctrine. Our intention is to find the structure of arithmetic within the structure of sets, not to put it there. We are not building a copy of the numbers, or setting a new convention, whereby this or that set is selected to represent something because it serves the purpose as well as any other. We intend to identify, and make visible for the first time, in constituent structure diagrams, the structure of numbers themselves, by studying the simplest sets and their constituent algebra.

The claim made in this paper is that this project has succeeded, and that the correspondence between the sets identified here and the integers and rational numbers is not a choice that we make or a convention that we introduce.

In support of this claim, we will later see that the sets which naturally represent the rational numbers, identified here for the first time, specify an internal structure for each rational number consisting of a finite sequence of integers with remarkable properties.

The sequence of integers is similar to the representation of the rational number as a continued fraction, but the sequence of integers is significantly more compact than that given by the standard continued fraction representation, and the algorithm which converts between the numerator/denominator form of the rational number and the sequence of integers is significantly more efficient, particularly in computationally complex cases. It provides a new and faster way to find rational approximations to irrational numbers, by achieving the same precision as the existing continued fraction algorithm in fewer steps.

This new representation of rational numbers is related to a known way of arranging the positive rational numbers in a binary tree known as the Stern-Brocot tree, and it provides new information about the structure of the tree, including an annotation of the edges, specifying the sets which each edge, when traversed, adds to the expression that specifies the set representing the rational number reached.

It also uniquely extends the tree to include all rational numbers including zero and negative numbers, preserving existing symmetries and introducing new symmetries which overlap with one another and together form a group, which has the same algebra as the group of symmetries of an equilateral triangle.

The arrangement of the negative rationals in the tree is quite non-trivial, but there is no choice of convention or new definition required to introduce them or determine their locations. The same rules which specify where each positive rational number appears in the tree, based on the sequence of integers in its natural representation, apply to negative numbers automatically.

The rational numbers and their natural representations as sequences of integers share the same order. At any given level in the tree, the rational numbers appear in increasing order from left to right and the sequences of integers which specify the internal structure of each rational number also appear in increasing order from left to right, ordered lexicographically. The novel form of continued fraction is an order-preserving isomorphism between those sequences and the rational numbers.

No considerations of algorithmic efficiency or order isomorphisms are involved in the process of identifying the sets which naturally encode the rational numbers. It is the study of the structure of the simplest sets, specifically the sets $\{0, 1\}$ and $\{\{1\}, \{0, 1\}\}$, which leads to the resulting representation of the rationals as finite sequences of integers.

Sets are nowhere to be seen in the novel type of continued fraction or the extended Stern-Brocot tree without annotations, but the consideration of sets and constituent structure is what leads to the improvements in efficiency and in the understanding of the relationship between rational numbers and integers. This is presented here as evidence, if not proof, that the natural encoding of integers, arithmetic, and rational numbers within finite sets has been correctly identified.

VII. Integers and Arithmetic Expressions

The standard construction of the integers uses an ordered pair of natural numbers. In a previous section, the simplest encoding of a finite sequence of natural numbers as a set emerged naturally as the set which specifies coordinates within tuples of tuples.

That experience suggests that an ordered pair of natural numbers, (n, m), can be represented by the set $n\diamond m$, where n and m refer to Zermelo's construction of those natural numbers.

When the corresponding set is constructed for the ordered pair $(2, 3)$, its structure diagram consists of 4 vertices arranged vertically at the bottom, corresponding to the sets representing Zermelo's natural numbers 0, 1, 2 and 3, followed by the diagram for $\diamond$, followed by sets which have the proposed structure of ordered pairs, reaching $(2, 3)$ at the top.

When constructing the integers using ordered pairs, we attend to the difference between the two natural numbers in the pair. In the standard construction, an integer is identified with the infinite set of ordered pairs of natural numbers with a given difference, and whether the first or the second natural number in the pair is the negative, or subtracted, number, is a matter of convention.

In our case, it's not a matter of convention. We have a correspondence between set operations and natural number addition, given by $n(m) = n + m$, and we need to understand how the set $\diamond$ interacts with the set operations and the arithmetic operations, in order to say whether the number whose constituent structure is above the diamond plays the role of a negative number, or the one whose structure is below.

So rather than considering $\diamond$ to be useful merely as something which is not a number, it is worth considering what its structure suggests about numbers before and after it.

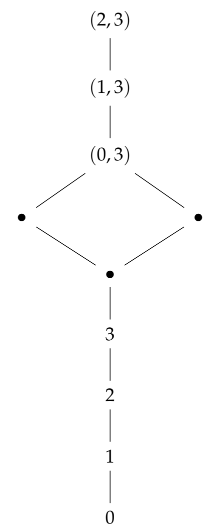


Figure 11
When we use the set $n\diamond m$ to represent the ordered pair of natural numbers, (n, m), the set representing the pair $(2, 3)$ has the structure shown above.

The set $\diamond$ is $\{\{1\}, \{1, 0\}\}$, which is the Kuratowski ordered pair $(1, 0)$, encapsulating the concept of 1 followed by 0. If the empty sets within this are replaced by a natural number, n, resulting in the set $\diamond n$, then the result will be the Kuratowski ordered pair $(n + 1, n)$.

This ordered pair has a constituent structure which, among ordered pairs of arbitrary sets, doesn't unambiguously specify n as the second entry in the pair. When representing integers, we need to ensure that no two integers are encoded in the same constituent structure, requiring us to look to other information to distinguish them. Among sets constructed using only $\diamond$ and the sets representing natural numbers, however, $(n + 1, n)$ is the only set with that structure, making it unambiguous and an acceptable representation.

The question then becomes how to interpret $1\diamond n$, if $\diamond n$ is $(n + 1, n)$.

The set $1(x) = \{\varnothing\}(x) = \{x\}$, which adds a single vertex to the top of x's constituent structure, clearly corresponds to proceeding to the next number when x is a natural number, but when x is not a natural number but rather a pair of consecutive numbers in reverse order, such as $(n + 1, n)$, a different sequence must be conceived of in order to say what comes next.

When we consider what mathematical object could be represented by the set $1\diamond n$, by asking which thing naturally comes after $(n + 1, n)$, the most obvious candidates are $(n + 2, n + 1)$, $(n, n - 1)$, and $(n + 1, n, n - 1)$. The first two, however, can be ruled out, because $(n + 2, n + 1) = \diamond(n + 1)$ and $(n, n - 1) = \diamond(n - 1)$, which are distinct sets.

That leaves $(n + 1, n, n - 1)$ as the natural interpretation of $1\diamond n$. Continuing to the m^{th} step of the sequence gives:

$$m\diamond n = (n + 1, n, n - 1, \cdots, n - m). \tag{37}$$

This reveals that, when successive addition of a vertex is interpreted as proceeding to the next step of a sequence, $\diamond n$ has a natural interpretation as the beginning of a sequence which counts downwards from the natural number n.

I. Expressions and Evaluation

$m\diamond n$ is obviously not the same set as the natural number $n - m$, presuming $n \geq m$, but its structure designates that number as the current step of a downward count. A downward count starting from that number is also designated by $\diamond(n - m)$, which is a different set, one that contains only the last two entries, $(n - m + 1, n - m)$, of the finite sequence specified by $m\diamond n$.

The set $\diamond(n - m)$ has a structure consisting of that of a single natural number, $n - m$, topped by $\diamond$, and can be considered to be the result of evaluating $m\diamond n$, which has the structure of an arithmetic expression consisting of unary representations of the natural numbers n and m connected by a non-numeric structure designating subtraction.

The fact that the structure of a set, x, appears to specify a procedure for reaching another set, y, implies that there is a natural encoding of certain operations on sets within the structure of sets which represent the same type of object as x.

To make the recognition of this encoding of set operations within set structure explicit, and to clarify the requirement that the encoding should be natural and consistent, we define:

Definition: Given a specification of sets including the empty set, called expressions, an encoding of an operation on those expressions is a rule which assigns to every expression, x, an expression, $[x]$, called the evaluation of x, which satisfies:

$$[ab] = [[a]b] = [a[b]] \tag{38}$$

for any two expressions, a and b, where $ab \equiv a(b)$.

This condition requires evaluation to respect associativity, $a(b(c)) = a(b)(c) = abc$, and also ensures that $[[a]] = [a]$, since b can be the empty set.

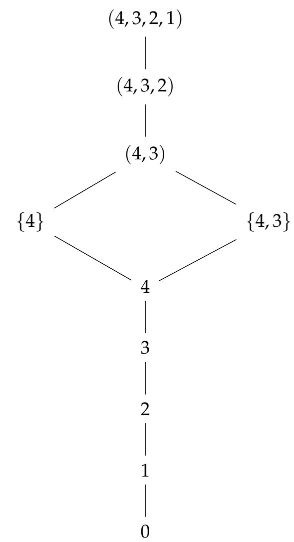


Figure 12

The set $\diamond 3$ is the Kuratowski pair $(4,3)_K$. If the following sets, $1\diamond 3 = \{\diamond 3\}$ and so on, are interpreted as continuations of a sequence started by the pair $(4,3)$, that sequence counts down.

II. The Operation Encoded in $\diamond$

We can use the requirement that evaluation must respect associativity to specify the evaluation of every expression, starting from a number of set replacement operations which define the encoding of an operation.

In the case of $\diamond$, the defining replacement is[5]:

$$[1\diamond 1] = \diamond \tag{39}$$

which allows us to conclude that:

$$[11\diamond 11] = [1[[1\diamond 1]1]] = [1\diamond 1] = \diamond \tag{40}$$

which shows that $[(n+m)\diamond n] = m\diamond$ and $[n\diamond(n+m)] = \diamond m$ for natural numbers, n and m.

With this understanding of the relation between natural numbers, subtraction, and the diamond set, we can specify the construction of the positive and negative integers, $+n$ and $-n$, corresponding to the natural number n as:

$$+n \equiv \diamond n \tag{41}$$
$$-n \equiv n\diamond. \tag{42}$$

Given an integer, x, and a natural number, m, we can specify the construction as a set of the unary expression for the addition, $x+m$, as:

$$\text{“}x+m\text{”} \equiv x(m) \tag{43}$$

and the unary expression for the subtraction, $x-m$, as:

$$\text{“}x-m\text{”} \equiv m(x) \tag{44}$$

[5] This replacement can be specified in the language of set operations as $(1\diamond 1 \to \diamond)$, and can be encoded as a set using the ordered pair construction given in section V, if the need arises.

where the quotation marks indicate that we are referring to the expressions, not the results of evaluating them, which are[6]:

$$[x(m)] = x + m \tag{45}$$
$$[m(x)] = x - m \tag{46}$$

although the unary expression will be identical with the result of evaluating it if m is added to a positive integer, $\diamond n(m)$, or subtracted from a negative integer, $m(n)\diamond$.

To specify constructions for expressions involving addition to integers and subtraction from integers of other integers, it is necessary to consider structures containing multiple diamond sets, such as $a\diamond b\diamond c$, where a, b, and c are natural numbers.

When a sequence element, s_n, is used to replace the empty sets within $\diamond$, the result is an ordered pair containing that sequence element occurring after its successor in that sequence: $\diamond(s_n) = (s_{n+1}, s_n)$. Adding $\diamond$ to a descending sequence will therefore result in an ascending sequence.

The structure $a\diamond b\diamond c$ will evaluate to an increasing sequence starting from $c - b + a$. An expression, e, which evaluates to an integer such as $m\diamond$ or $\diamond m$ must contain an odd number of diamonds in its structure, so that $1e$ indicates $e - 1$ and $e1$ indicates $e + 1$.

To get an integer from an expression with three $\diamond$ structures, it is necessary to include the additional rule of replacement which states that reversing direction twice evaluates to nothing:

$$[\diamond\diamond] = \{\}. \tag{47}$$

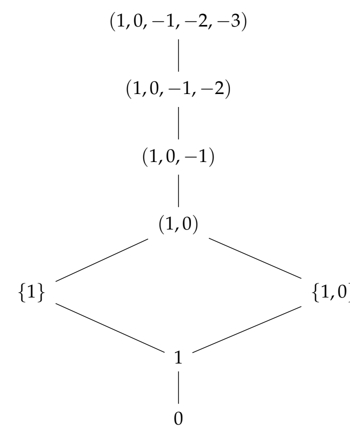


Figure 13

The set $3\diamond$, interpreted as the continuation of the sequence specified by $\diamond$.

The expressions for the sum and difference of two integers will therefore need to include a $\diamond$ alongside the integers themselves. If x and y are integers of the form $\diamond n$ and $\diamond m$, which have the same sign, then $xy = x(y) = \diamond n\diamond m$ will perform a subtraction using n and m. So the set which represents the expression for the subtraction, $x - y$, of one integer, y, from another, x, can be specified as:

$$\text{“}x - y\text{”} \equiv \diamond y(x) = \diamond yx \tag{48}$$

and the expression for the addition, $x + y$, can be specified as:

$$\text{“}x + y\text{”} \equiv y(\diamond x) = y\diamond x. \tag{49}$$

These expressions yield the corresponding results when evaluated:

$$[\diamond yx] = x - y \tag{50}$$
$$[y\diamond x] = x + y \tag{51}$$

which shows that the integers, arithmetic expressions involving addition and subtraction, and the evaluation of those expressions to yield integers, all have natural representations as finite sets,

[6] Gratuitously abusing notation, we use x and m to refer to both the numbers and the sets representing those numbers. When x appears beside a sign denoting an arithmetic operation such as $+$ or $-$, it denotes the number, and when it appears in a set operation such as $x(y) = xy$, it denotes the set. Arithmetic multiplication will always be explicitly denoted using $\times$ to prevent ambiguity.

given by the defining equations:

$$
\begin{aligned}
+n &\equiv \diamond n \\
-n &\equiv n\diamond \\
\text{“}x+m\text{”} &\equiv x(m) \\
\text{“}x-m\text{”} &\equiv m(x) \\
\text{“}x-y\text{”} &\equiv \diamond y(x) \\
\text{“}x+y\text{”} &\equiv y(\diamond x) \\
[1\diamond 1] &= \diamond \\
[\diamond\diamond] &= \{\}
\end{aligned} \tag{52}
$$

where n and m are natural numbers, x and y are integers, quotation marks denote an expression, square brackets denote evaluation, and $\diamond = (1,0) = \{\{1\},\{1,0\}\}$.

VIII. Multiplication

I. Arithmetic Expressions and Rules of Evaluation for Products of Integers

The natural representation of the integers in the form of sets reveals that the Kuratowski ordered pair, $\diamond = (1,0)$, naturally starts a sequence that counts down, with the succeeding terms in the sequence, $1\diamond = \{\diamond\}, 2\diamond = \{\{\diamond\}\}$, and so on, continuing the sequence, by successively making a step of the same magnitude in the same direction.

In the same way, an ordered pair of distinct integers naturally encodes a step from the first integer in the pair to the second, and a natural interpretation of the successor of that pair is another step with the same magnitude and direction, continuing the sequence started by the pair.

The Kuratowski pair $(x,y) = \{\{x\},\{x,y\}\}$ can encode all pairs of distinct positive integers unambiguously within its constituent structure, because $\diamond n$ is not a constituent of $\diamond(n+m)$ unless $m = 0$.

In the case when both integers are negative, $(x,y) = (n\diamond, m\diamond)$, or one integer is negative and the other is zero, the set representing one integer will be a constituent of that of the other, and the constituent structure of the Kuratowski pair will not unambiguously specify the order and structure of the elements it contains.

This ambiguity also occurs with Kuratowski ordered pairs of natural numbers, (n,m), since that also causes the set representing one entry in the pair to be a constituent of the set representing the other.

A small modification to the encoding of a pair permits pairs containing negative integers, zero, and also natural numbers, to be unambiguously encoded within the constituent structure of sets:

$$(x,y) = (\diamond x, \diamond y)_K = \{\{\diamond x\},\{\diamond x, \diamond y\}\} \tag{53}$$

where $()_K$ refers to the Kuratowski encoding of the pair.

The only case when this pair fails is when the entries in the pair are equal, in which case $\{\{a\},\{a,b\}\}$ reduces to $\{\{a\}\} = 2a$, which is the set two steps after a in whatever sequence a itself naturally specifies. This is conceivably a more natural way to interpret (a,a) than as the specification of the start of a sequence which goes nowhere.

We will take the hint and abstain from multiplying by zero. When we consider a and b to be distinct integers, the result, after evaluation, of proceeding one step further along the sequence started by the pair (a, b) is the pair $(b, b + (b - a))$, which is the pair representing the step from b that proceeds the same distance in the same direction as the previous step.

The evaluation rule for stepping forward one term in a sequence started by the pair of integers (a, b), is then given by:

$$[1(a, b)] = [(a, b)ab]. \tag{54}$$

$\diamond ab$ is the expression for the integer $b - a$, which, as an integer, subtracts numbers supplied to it on the left, while ab has the same numeric value as that integer, but adds numbers supplied to it on the left, since it contains two instances of $\diamond$, one inside a and one inside b. The set $(a, b)ab$ is equal to (aab, bab). The entries in this pair are expressions for $a - a + b$ and $b - a + b$, which provide the result $(b, b + (b - a))$.

For a natural number, n, the expression $n(a, b)$ will step the sequence forward n times from the starting point, b, to $b + n(b - a)$.

Multiplication of integers by natural numbers greater than zero can then be achieved by setting a equal to the integer 0, which is represented by the set $\diamond$:

$$[1(\diamond, b)] = [(b, b\diamond b)] = [(\diamond, b)\diamond b]. \tag{55}$$

The 1 on the left of the equation above steps the sequence forward from $(0, b)$ to $(b, b + b) = (b, 2 \times b)$, with the n^{th} step after $(0, b)$ reaching the numeric value $n \times b + b = (n + 1) \times b$, encoded by the expression $n(\diamond, b)$.

We can use trailing dots, $\cdots$, when we want to make it explicit which part of an expression will be updated by taking that expression's successor:

$$\text{“}(1 + \cdots) \times b\text{”} \equiv (\diamond, b) \tag{56}$$

and we can call it an incomplete arithmetic expression to express the fact that it does not subtract numbers supplied to it on the left like integers do:

$$\text{“}x - \cdots\text{”} \equiv x. \tag{57}$$

To evaluate an incomplete arithmetic expression and get an integer as a result, it is necessary to start a new sequence counting downwards in steps of 1 from latest number of a given sequence.

The following rule of evaluation allows sequence step sizes to be changed in the necessary way:

$$[((a, b), (c, d))] = [((a, b), d)] = [(b, (c, d))] = [(b, d)]. \tag{58}$$

That is, only the last number reached by a sequence matters when switching to a new sequence; the steps leading to it, if there are any, are forgotten during evaluation when a new sequence is started.

This matches the structure of integers and their multiplicative sequences, since each positive integer is a pair of natural numbers, $\diamond n = (1, 0)n = (n + 1, n)$, and a pair of integers starts a sequence, $((n + 1, n), (m + 1, m)) \rightarrow (n, m)$ with the step size and position determined by the second entries in the original ordered pairs.

Given an expression, e, which has an integer numeric value but is part of a sequence with a step size other than -1, an integer with that value can be obtained evaluating the expression $(e(1), e)$, which starts a downward sequence at that numeric value with a step size of 1. We can call the result of this a complete arithmetic expression, and explicitly denote it by showing the trailing dots subtracted from it: “$(n \times m) - \cdots$”.

This converts the sequence generator (n, m) into $(m + 1, m) = (1, 0)m = \diamond m$, allowing an integer result to be obtained after a multiplication.

This sequence-switching rule also automatically handles multiplication by negative integers:

$$[\diamond(a,b)] = [(1,0)(a,b)] = [(1(a,b),(a,b))] = [(b,bab),(a,b)] = [(bab,b)] \tag{59}$$

where (bab,b) refers to $(b+(b-a),b)$, which leads to $\diamond$ naturally reversing of the direction of an existing sequence. It changes the step size from $(b-a)$ to $-(b-a)$, while keeping the current position of the sequence at b.

Using this rule, the still-multiplying expression for the product of integers $x+1$ and y:

$$\text{“}(x+1-\cdots) \times y\text{”} \equiv x(\diamond,y) \tag{60}$$

can be turned into an expression for the integer that results from the multiplication:

$$\text{“}((x+1)\times y) - \cdots\text{”} \equiv (x(\diamond,y)1,\ x(\diamond,y))\,. \tag{61}$$

II. Multiplication With Ordered Pairs of Natural Numbers

When the ordered pair contains natural numbers, (n,m), the next term in the sequence is the pair $(m, m+(m-n))$, whose elements exceed those of the original pair by the amount $m-n$.

The evaluation rule for the corresponding sets is:

$$[1(n,m)] = [(n,m)\diamond n\diamond m]. \tag{62}$$

In the special case when $n=1$ and $m=0$, the pair is $(1,0)=\diamond$ and this gives the result:

$$[1\diamond] = [1(1,0)] = [(1,0)\diamond 1\diamond 0] = [\diamond\diamond 1\diamond] = 1\diamond \tag{63}$$

which shows that negative integers, $n\diamond$, do not evaluate to anything simpler.

The evaluation of the expression for multiplication of natural numbers, $1(n,m) = (n,m)\diamond n\diamond m$, appears more complex than the result for integers, $1(a,b) = (a,b)ab$, but it coincides with the evaluation of the expression obtained by replacing n and m with the equivalent positive integers, $\diamond n$ and $\diamond m$:

$$[1(\diamond n,\diamond m)] = [(\diamond n,\diamond m)\diamond n\diamond m]. \tag{64}$$

This is because:

$$[(\diamond n,\diamond m)] = [((n+1,n),(m+1,m))] = [(n,m)] \tag{65}$$

which shows that encoding an ordered pair, (x,y) as $(\diamond x,\diamond y)_K = \{\{\diamond x\},\{\diamond x,\diamond y\}\}$ leads to exactly the same final evaluation of expressions as that obtained by using the pair $(x,y)_K = \{\{x\},\{x,y\}\}$.

We can therefore use ordered pairs of natural numbers such as $(2,0)$ without worrying about the encoding of their information within constituent structure, since the evaluation rule ensures that the result of evaluating expressions involving them will be the same as the result for the corresponding positive integers, whose order and constituent structures are unambiguously encoded in their ordered pairs.

One simple expression involving natural numbers occurs when $n=0$ and the pair is $(0,m)$:

$$[1(0,m)] \equiv (0,m)\diamond\diamond m = (0,m)m \tag{66}$$

leading to a multiplication formula for natural numbers without $\diamond$:

$$[n(0,m)] = (0,m)(n\times m). \tag{67}$$

III. The Minimal General Evaluation Rule for Multiplication

The evaluation rule $[1(a,b)] = (a,b)ab$ is valid for integers because they subtract numbers supplied to them on the left.

The set $(e1,e)$, has this property for any expression, e, while having the same numeric value as e, so the most general form of the multiplication rule, which is valid for any two expressions, e_1 and e_2, is:

$$[1(e_1,e_2)] = [(e_1,e_2)(e_1 1,e_1)(e_2 1,e_2)]. \tag{68}$$

It suffices to specify the evaluation rule for the case when one of the entries in the ordered pair is zero, since the fact that $(x,y)z = (xz,yz)$ with $z = (1x,x)$, for example, allows any pair, (x,y), to be expressed as a pair with one zero entry followed by an expression which subtracts on the left.

This allows multiplication of expressions in general to be evaluated with the rule:

$$[1(0,e)] = [(0,e)\diamond(e1,e)]. \tag{69}$$

IX. Division and the Rational Numbers

I. Reciprocals

Starting from the set, $2_V = \{0,1\}$, and proceeding n steps forward, to get the set $n2_V = n(2_V)$, results in a set which, on its own, has a constituent structure indistinguishable from $2+n$, since the set 2_V is von Neumann's construction of the number 2. The distinction becomes apparent in sets which contain successors of 2_V and also natural numbers in Zermelo's representation.

When the natural numbers and 2_V and its successors are seen within the same structure, it becomes clear that, alongside the sequence, $0,1,2,3,\cdots$, there is another sequence, $0,1,2_V,1(2_V),\cdots$, which mirrors that of the natural numbers.

From the point of view of constituent structure alone, either sequence could be taken to be the natural numbers; all that the structure shows is that they are distinct from the number 2 onwards, with 2 and each succeeding natural number having a counterpart in the other sequence.

The number $1 = \{\varnothing\}$ is the origin of both sequences. There are exactly two sets whose constituents are $0 = \{\}$ and $1 = \{0\}$. One of these is $\{1\}$, the set which, in a sense, loses sight of 0 as it steps forward from 1, and which we, following Zermelo, identify with the number 2.

The other set, $2_V = \{0,1\}$, "straddles" 1 and 0. In Conway's construction of surreal numbers[4], the number $\{0|1\}$ is identified with ½, and it appears to be natural to interpret $2_V = \{0,1\}$ as ½ in our case, since the sequence of reciprocals, 1, ½, ⅓, $\cdots$, does have the relationship to the sequence $1,2,3,\cdots$ that is visible in figure 14.

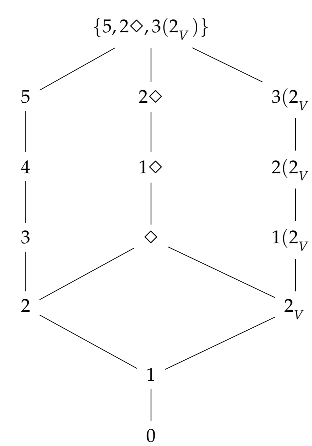


Figure 14
$2_V = \{0,1\}$ starts a sequence of sets which are distinguished from natural numbers and integers within structures that contain them.

There are several other indications within the structure of sets that this is the right set to start the sequence of reciprocals. It is the simplest set available: $\{0,1\}$ is the 4$^{\text{th}}$ simplest set, and is the simplest set which hasn't previously been assigned any numeric value.

The integer $0 = \diamond$, which is the simplest set that cannot be a natural number because of its constituent structure, introduces subtraction, negative numbers, and the integers, extremely naturally, by introducing the sequence which starts with $(1, 0)$. This set, $\diamond$, is also unique among all orderings of sets because it is the only structure of its kind that determines an unambiguous ordering of sets already ordered by constituency.

So $\diamond$ is not an arbitrarily chosen set; it's a structure within sets which genuinely encodes properties of integers, and $\diamond$ contains $2_V = \{0, 1\}$ and $2 = \{1\}$ as its only elements. So 2_V is already within the system of numbers, but the number it represents has not previously been articulated.

2_V contains within its structure the assertion that only natural numbers which are greater than or equal to 2 have reciprocals, which is quite intelligible and natural. Finally, 2_V is the only choice of set which produces the result that each number has a constituent structure which is isomorphic to that of its reciprocal. This is the property which specifies that a number and its reciprocal are a match for each other: It seems natural that only numbers with matching structures should be able to cancel each other arithmetically, and no other set can achieve this.

With this interpretation, for any natural number n, we can specify the reciprocal number $1/(2+n)$ using the expression:

$$\frac{1}{2+n} \equiv n2_V \tag{70}$$

where $2_V = \{0, 1\}$.

However, we need to remember that the next set in the sequence, $1n2_V$, is the reciprocal of $n+3$. The expression $n2_V$ is still in the process of accepting additions to the number whose reciprocal is to be constructed:

$$\text{“}\frac{1}{2+n+\cdots}\text{”} = n2_V \tag{71}$$

so the expression is incomplete[7].

It can be made into a complete arithmetic expression using the operation $e \to (e1, e)$:

$$\text{“}\frac{1}{2+n} - \cdots\text{”} = (n2_V 1, n2_V) \tag{72}$$

which can then be added to and subtracted from integers using the existing rules:

$$x + (1/_{2+n}) = (1/_{2+n}) \diamond x \tag{73}$$

$$x - (1/_{2+n}) = \diamond (1/_{2+n}) x. \tag{74}$$

II. Fractions

The set $2_V = \{0, 1\}$ and its successors, $n(2_V)$, provide sets which represent fractions of the form $\frac{1}{2+n}$. The next task is to determine which sets provide the simplest and most natural representations of fractions with numerators other than 1.

A way to represent $\frac{m}{n}$ for any two natural numbers, n and m, is actually automatically specified once reciprocals have been assigned a representation. Any rational number can be expressed in

[7] Although only rational numbers are constructed here, it may be of interest to note that the expression $2_V 2_V 2_V \cdots = 1/(2+1/(2+1/(2+\cdots)))$ converges to $\sqrt{2}-1$. The real number $\sqrt{2}$ can therefore be designated, although not evaluated, using finite sets, by encoding a substitution such as $2_V \to 2_V 2_V$.

the form of a continued fraction:

$$a_0 + \cfrac{1}{a_1 + \cfrac{1}{a_2 + \cfrac{1}{\ddots + \cfrac{1}{a_k}}}}$$

where a_0 is an integer and a_i is a positive number for $i > 0$.

All rational numbers can be expressed as a continued fraction with a finite number of terms in the sequence, $[a_0; a_1, \cdots, a_k]$. The first term, a_0, specifies the integer part of a rational number, which can be positive, negative or zero, and the remaining terms, $a_1, \cdots, a_k$, specify a fraction between 0 and 1.

When $m_1 < n_1$, the fraction m_1/n_1 has a reciprocal, n_1/m_1, which is greater than one, and which therefore has an integer part, a_1, which provides the next term in the sequence. Subtracting a_1 from n_1/m_1 produces another fraction, m_2/n_2, which is less than one and has a reciprocal with another integer part.

The numbers n_i and m_i get smaller each time because of the subtractions, and the sequence eventually terminates.

So we already have an expression for a fraction with a denominator other than 1, given by:

$$2_V 2_V = \frac{1}{2 + \frac{1}{2}} = \frac{1}{\frac{5}{2}} = \frac{2}{5} \tag{75}$$

where we have used the fact that:

$$2_V = \frac{1}{2 + \cdots}$$

and added another reciprocal rather than a natural number to the denominator.

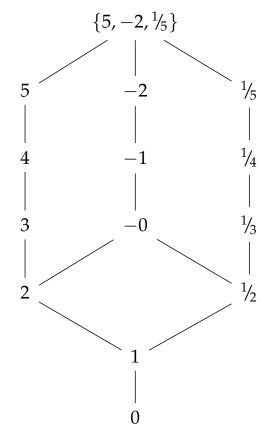


Figure 15

The constituent structure of a set containing natural numbers, negative numbers, and reciprocals.

Unfortunately, but interestingly, we cannot use the continued fraction representation to construct all rational numbers using 2_V to represent reciprocals. The terms, $a_1, a_2, \cdots$, in the continued fraction representation of a rational number must be positive, but they can be equal to 1, while 2_V can only construct reciprocals of natural numbers greater than or equal to 2.

We have no way to construct a fraction such as:

$$\cfrac{1}{1 + \cfrac{1}{1 + \frac{1}{2}}}$$

using only natural numbers and 2_V in an expression of the form:

$$2_V m 2_V n 2_V = \cfrac{1}{2 + n + \cfrac{1}{2 + m + \frac{1}{2}}}. \tag{76}$$

Our ability to represent reciprocals as sets starts with the number 2 for a good reason: 2 is the first natural number that has a reciprocal distinct from any natural number or integer. The reciprocal of 1, if it can be called that, is 1, and the next number after 1 is $\{1\} = 2$, so the set which represents the number 1, namely $\{\varnothing\}$, is an expression for $1 + \cdots$, not:

$$\frac{1}{1+\cdots}.$$

The reciprocal of 2 is distinct from 2, and is a completely different type of number, represented by a different type of set, $\{0, 1\}$, which is suitable for starting the sequence of reciprocals and specifying the sets that encode them. So the structures of reciprocals and sets agree with each other that reciprocals start at 2, not 1. This raises the intriguing possibility that the natural representation of rational numbers within sets, when we find it, will tell us something we didn't already know about continued fractions.

III. Representing Fractions with Positive Integers

Since some fractions between 0 and 1 can't be constructed with any combination of natural numbers and reciprocals generated using 2_V, it will be necessary to use the $\Diamond$ set and introduce subtraction, which doesn't appear in standard continued fractions.

We can start by considering just positive, or, rather, non-negative integers, $\Diamond m$, which are expressions of the form:

$$\Diamond m = m - \cdots$$

and in fact doing this immediately produces a way to construct every fraction between 0 and 1:

$$\Diamond m_3 2_V \Diamond m_2 2_V \Diamond m_1 2_V = \cfrac{1}{2 + m_1 - \cfrac{1}{2 + m_2 - \frac{1}{2 + m_3 - \cdots}}}. \tag{77}$$

Since each reciprocal is between 0 and 1, subtracting it from the number $2 + m$ results in a rational number which is between $1 + m$ and $2 + m$, where $m \geq 0$.

So each denominator in this variant of a continued fraction has an integer part which can be 1 or any other positive whole number, and a fractional part between 0 and 1. These are exactly the denominators which can be specified in standard continued fractions.

Positive integers, $\Diamond m$, and reciprocals generated by 2_V, are therefore able to represent all fractions between 0 and 1 in the form of a continued fraction which subtracts reciprocals instead of adding them.

We can specify the procedure for generating the sequence of numbers, $m_1, m_2, \cdots, m_k$, which are added to 2 in each denominator in the continued fraction expression for a fraction $\frac{n}{d}$ whose numerator, n, and denominator, d, are positive natural numbers with $n < d$:

1. Let $i = 1$ and $n_1 = n$ and $d_1 = d$.

2. If $n_i = 1$, compute $m_i = d_i - 2$ and stop.

3. Compute $m_i = \mathrm{floor}(\mathrm{d_i}/\mathrm{n_i}) - 1$.

4. Compute $d_{i+1} = n_i$ and $n_{i+1} = (2 + m_i) \times n_i - d_i$.

5. Increment i and go to step 2.

Table 2
Representations of Rational Numbers as Continued Fractions

	Standard Continued Fraction	Non-Negative Integers and 2_V
1/2	2	0
1/3	3	1
1/4	4	2
1/5	5	3
2/3	1, 2	0, 0
2/5	2, 2	1, 0
3/4	1, 3	0, 0, 0
3/5	1, 1, 2	0, 1
4/5	1, 4	0, 0, 0, 0

This procedure repeatedly finds the smallest multiple of the numerator which exceeds the denominator, and uses the amount by which it exceeds the denominator as the next numerator, terminating when the numerator reaches 1, indicating that a reciprocal of a natural number has been reached, which is then the final reciprocal in the continued fraction.

The procedure for reconstructing the numerator and denominator of a rational number from the sequence, $m_1, \cdots, m_k$, is simply to define a function, $f(m_1, m_2, \cdots)$, as:

$$f(m_1, m_2, \cdots, m_k) = \frac{1}{2 + m_1 - f(m_2, \cdots, m_k)} \tag{78}$$

with f having a value of zero when it has zero arguments.

The sequence of numbers that this type of continued fraction assigns to any specific fraction will differ from the sequence which appears in its standard continued fraction, because in this case the numbers $n_1, n_2, \cdots$, can be zero.

Table 2 shows the sequences which this continued fraction expression and the standard one generate for fractions with numerators and denominators below 6.

Although this novel kind of continued fraction generates sequences with smaller numbers than the standard version, it is evidently very inefficient at encoding fractions of the form $m/(m+1)$, for which it uses a sequence of m zeros, which will become awkward for large values of m. The standard version is able to encode those fractions using just two numbers rather than a long sequence.

We have yet to consider how the set 2_V interacts with negative integers, of the form $n\Diamond$, which might provide a more efficient way to encode fractions which are awkward to represent using only non-negative integers.

IV. Reciprocals and Negative Integers

When we consider the expression:

$$n\Diamond 2_V \tag{79}$$

where $n\Diamond$ is a negative integer, it seems natural to insert $n\Diamond$ into the denominator of the reciprocal in the place where n appears in its corresponding expression:

$$n2_V = \frac{1}{2 + n + \cdots} \quad \longrightarrow \quad n\Diamond 2_V = \frac{1}{2 - n - \cdots}. \tag{80}$$

There are a few indications that this might not be right, though. Since 2 is positive, $2-n$ is not just a negative version of $2+n$. The negative integer whose reciprocal generated by 2_V is equal in magnitude and opposite in sign that of n would be $-(n+4)$, which is somewhat bizarre and breaks symmetries for no apparent reason.

It would also be the case that $4\diamond 2_V = 2(2\diamond 2_V)$ would be the number which is obtained by continuing two steps past the reciprocal of zero, which requires us to assume that the occurrence within a numerical sequence of a term which has no definable numerical value does not disrupt the previous pattern.

The appearance and irremovability of the set $2\diamond 2_V$ representing the non-numeric expression $1/0$ would require special rules of evaluation for expressions in which it appears, and there would be no clear way for us to determine what the result of adding such a number to another, or multiplying by it, would be.

For these reasons, we must consider the possibility that $n\diamond 2_V$ is not the reciprocal of $2-n-\cdots$, and that some considerations will lead us to another value for it.

We can list a few desiderata:

- Associativity requires that $\diamond 2_V$ can be evaluated, and that it yields an expression, $[\diamond 2_V]$, with the same numeric value as 2_V, namely ½.
- Associativity also requires $[[\diamond\diamond]2_V] = 2_V$, so the expression for $[\diamond 2_V]$ would need to satisfy $[\diamond[\diamond 2_V]] = 2_V$.
- $1[\diamond 2_V]$ should be larger than ½, and $n[\diamond 2_V]$ should increase as n increases, because it goes in the opposite direction to $n2_V$, which decreases with n.
- The numeric value of $1[\diamond 2_V]$ should be determined by a rule which applies to sets in general and isn't just made up for this one case.

One candidate for a rule which could specify a value for $1[\diamond 2_V]$ is the rule which steps a sequence forward given an ordered pair of numeric values to start it.

The set $\diamond 2_V$ is an ordered pair, $(1,0)2_V = (1(2_V), 2_V)$, and the entries in the pair have numeric values, $(\frac{1}{2+1}, \frac{1}{2}) = (\frac{1}{3}, \frac{1}{2})$.

This identifies the numeric value of the next term in the sequence started by that pair as:

$$1\diamond 2_V = \frac{1}{2} + (\frac{1}{2} - \frac{1}{3}) = \frac{1}{2} + \frac{1}{6} = \frac{2}{3}. \tag{81}$$

This value can be obtained from $\frac{1}{2}$ by:

$$\frac{2}{3} = \frac{1+1}{2+1} \tag{82}$$

which suggests that:

$$[\diamond 2_V] = \frac{1+\cdots}{2+\cdots} = 1 - \frac{1}{2+\cdots} = 2_V\diamond 1. \tag{83}$$

This evaluation rule satisfies all of the desiderata above: $1-\frac{1}{2} = \frac{1}{2}$ so the numerical value is correct, $1-(1-x) = x$ so $[\diamond[\diamond 2_V]] = 2_V$, two thirds is larger than one half, and the rule that specified this value is simple sequence progression. It also creates a symmetry around the number $\frac{1}{2}$.

Equipped with this understanding of how negative integers interact with 2_V, we can consider the question of how they appear in the natural representations of rational numbers.

V. An Efficient Continued Fraction Representation

The evaluation rule for reciprocals involving negative integers:

$$[\diamond 2_V] = 2_V \diamond 1 \implies [n \diamond 2_V] = n 2_V \diamond 1 \tag{84}$$

immediately tells us how to handle negative integers which appear in a sequence of numbers representing a continued fraction constructed using 2_V.

Specifically, if the sequence of numbers, which may be positive, negative, or zero, is $s_1, s_2, \cdots, s_k$, then the function:

$$f(s_1, s_2, \cdots, s_k) = \begin{cases} 1 - f(-s_1, -s_2, \cdots, -s_k) & s_1 < 0 \\ \dfrac{1}{2 + s_1 - f(s_2, \cdots, s_k)} & s_1 \geq 0 \\ 0 & k = 0 \end{cases} \tag{85}$$

computes the value of the fraction represented by that sequence.

The first case above implements the evaluation rule $\diamond 2_V = 1 - \frac{1}{2+\cdots}$. It reverses the sign of every subsequent integer in the sequence and not just the first one, because if the first term is positive, $s_1 = \diamond n_1$, then:

$$s_2 2_V s_1 2_V = s_2 2_V \diamond n_1 2_V$$

while if s_1 is negative, $s_1 = n_1 \diamond$:

$$s_2 2_V s_1 2_V = s_2 2_V n_1 \diamond 2_V = s_2 2_V n_1 2_V \diamond 1$$

which shows that both instances of 2_V in the above expressions differ in sign. When the $\diamond$ is moved to the right of the rightmost 2_V, the negative integer, s_1 between the first and the second 2_V changes into a natural number, n_1, not into a positive integer, $\diamond n_1$, and so it affects later reciprocals in the continued fraction.

The second case, when $s_1 \geq 0$, recursively evaluates the reciprocal to yield a fraction specified by a numerator and a denominator. The third case, $k = 0$, specifies what happens when the number of arguments supplied to f is zero. When that occurs, the recursion terminates: $f() = 0$, so $f(x) = \frac{1}{2+x-0}$ when x is positive.

The following procedure constructs the sequence, $s_1, s_2, \cdots, s_k$, starting from a fraction, $\frac{n}{d}$, with a numerator and denominator which are natural numbers satisfying $0 < n < d$:

1. Let $sign = 1$, $i = 1$, $n_1 = n$ and $d_1 = d$.
2. If $n_i = 1$, compute $s_i = sign \times (d_i - 2)$ and stop.
3. Compute $s_i = sign \times (\text{floor}(d_i / n_i) - 1)$.
4. If $s_i = 0$, set $sign = -sign$ and $n_i = d_i - n_i$ and go to step 2.
5. Compute $d_{i+1} = n_i$ and $n_{i+1} = (2 + |s_i|) \times n_i - d_i$.
6. Increment i and go to step 2.

This procedure is similar to the one which was specified for the case when all the integers were non-negative, but step 4 checks to see if the numerator is larger than half of the denominator, and if it is, it replaces the numerator with the difference between it and the denominator. This difference will be less than half as large as the denominator and will therefore produce a reciprocal which exceeds two. Step 4 also records the occurrence of this replacement in the sign of the resulting sequence elements, so that it can be correctly decoded later.

This gives us a natural encoding of all the rational numbers between 0 and 1 within finite sets.

Any rational number which is not itself an integer is equal to an integer, s_0, minus a fraction between 0 and 1, so every rational number has a natural encoding in a set given by:

$$s_k 2_V \cdots s_2 2_V s_1 2_V s_0 \tag{86}$$

where $s_1, \cdots, s_k$ are the integers in the continued fraction sequence defined above, and s_0 is the integer from which that fraction is subtracted to obtain the full rational number. The fraction is subtracted rather than added because integers such as s_0 subtract numbers on their left and a more complicated, less symmetric expression involving a $\diamond$ would be necessary to add the fraction rather than subtract it.

The sequences for standard continued fractions are usually specified using the notation, $[a_0; a_1, a_2, \cdots, a_n]$, where the first entry in the sequence is separated from the rest by a semicolon to indicate its role as the integer part of the rational number. We can use the same notation for sequences specifying natural representations of rational numbers whose first term is an integer when the expression in equation 86 would be awkward.

As the symmetry in this expression for a general rational number suggests, the integer part of a rational number can be included along with the fractional part in a procedure which handles them both in a consistent way.

The encoding, $s(f) = [s_0; s_1, \cdots, s_k]$, of an arbitrary rational number, f, is given by:

$$s(f) = \begin{cases} (f), & f - \lfloor f \rfloor = 0 \\ (\lceil f \rceil) \ominus s(\frac{1}{f - \lfloor f \rfloor} - 2), & f - \lfloor f \rfloor < \frac{1}{2} \\ (\lceil f \rceil) \oplus s(\frac{1}{1-(f - \lfloor f \rfloor)} - 2), & f - \lfloor f \rfloor \geq \frac{1}{2} \end{cases} \tag{87}$$

with $\oplus$ defined as concatenation of tuples:

$$(a) \oplus (b, c, d, \cdots) \quad \equiv \quad (a, b, c, d, \cdots) \tag{88}$$

and $\ominus$ denoting concatenation with the signs of the second sequence's terms flipped:

$$(a) \ominus (b, c, d, \cdots) \quad \equiv \quad (a, -b, -c, -d, \cdots). \tag{89}$$

$\lfloor f \rfloor$ refers to the largest integer that doesn't exceed f and $\lceil f \rceil$ is the smallest integer that is not less than f. This encoding chooses whether to take the reciprocal of the fractional part of f or one minus that fractional part depending on which of those will yield a number greater than or equal to 2. Subtracting 2 from that number is then guaranteed to give a non-negative result.

Decoding is accomplished by the function:

$$f(s_0, s_1, s_2, \cdots, s_k) = \begin{cases} s_0 & k = 0 \\ s_0 - \frac{1}{2 + f(s_1, s_2, \cdots, s_k)} & s_1 \geq 0 \\ s_0 - 1 + \frac{1}{2 + f(-s_1, -s_2, \cdots, -s_k)} & s_1 < 0 \end{cases} \tag{90}$$

which uses the sign of each term in the sequence, $[s_0; s_1, \cdots, s_k]$, apart from the first term, to determine whether that term specifies the reciprocal of a fractional part of a number, $f - \lfloor f \rfloor$, or the reciprocal of $1 - (f - \lfloor f \rfloor)$.

Two implementations of these functions are provided in appendix A. The Python version is intended to be readable with no concern for efficiency, and works for positive and negative

	Standard Continued Fraction	Natural Representation
1/2	[0; 2]	[1; 0]
1/3	[0; 3]	[1; -1]
1/4	[0; 4]	[1; -2]
1/5	[0; 5]	[1; -3]
2/3	[0; 1, 2]	[1; 1]
3/2	[1; 2]	[2; 0]
2/5	[0; 2, 2]	[1; -1, 0]
5/2	[2; 2]	[3; 0]
3/4	[0; 1, 3]	[1; 2]
4/3	[1; 3]	[2; -1]
3/5	[0; 1, 1, 2]	[1; 1, 0]
5/3	[1; 1, 2]	[2; 1]
4/5	[0; 1, 4]	[1; 3]
21/29	[0; 1, 2, 1, 1, 1, 2]	[1; 2, 1, 1]
89/144	[0; 1, 1, 1, 1, 1, 1, 1, 1, 1, 2]	[1; 1, 1, 1, 1, 1]

Table 3
Rational numbers expressed as sequences of integers given by their continued fraction expansions and their natural representations.

rational numbers, while the C version is intended to be fast and uses unsigned numerators and denominators.

The sequences which encode various rational numbers are shown in table 3, with the sequences generated by the standard continued fraction shown alongside for comparison.

The table shows that the sequences generated by the natural encoding of the fractions tend to be shorter and contain numbers which are closer to zero.

The final entry, 89/144, in table 3 is a ratio of consecutive terms in the Fibonacci sequence, which provides an approximation to the fractional part of the golden ratio, $\varphi = (1+\sqrt{5})/2$. The golden ratio, $1.618\cdots$, is the irrational number which is most difficult to approximate using rational numbers. The standard continued fraction representations of the rational approximations to it contain long strings of consecutive 1's, which provide the smallest possible update to the number with each additional term in the sequence.

For each such approximation, the natural representation encodes the same number in a sequence which is approximately half as long. This compact encoding is accompanied by an improvement in the computational efficiency.

Table 4 shows the time taken by both the standard and the natural continued fraction algorithms to encode and decode ratios of successive Fibonacci numbers, using the implementations written in C given in the appendix, using `gcc`'s compiler optimizations and run on a machine with an AMD Ryzen 7 1700x processor.

As the table shows, the improvement in performance achieved by using the natural representation instead of the standard continued fraction is quite significant and becomes more pronounced as the computational complexity increases. Because of this, the natural representation provides a faster algorithm for generating rational approximations to certain irrational numbers.

For example, the standard continued fraction expansion of $\sqrt{3}$ is $[1;1,2,1,2,1,2,\cdots]$, while the natural version is $[2;2,2,2,\cdots]$. The same approximation to $\sqrt{3}$ can be achieved by evaluating a natural representation's continued fraction with a finite sequence of 2's, or by evaluating a standard continued fraction using a sequence which is twice as long and takes about 50% more

Ratio	Continued Fraction Encoding	Continued Fraction Decoding	Natural Representation Encoding	Natural Representation Decoding
f_5/f_6	156	25	114	22
f_{10}/f_{11}	341	57	226	56
f_{20}/f_{21}	791	124	414	111
f_{30}/f_{31}	1682	281	605	179
f_{40}/f_{41}	2140	353	783	241
f_{50}/f_{51}	2667	454	970	305
f_{60}/f_{61}	3201	560	1231	379
f_{70}/f_{71}	3712	667	1352	503
f_{80}/f_{81}	4268	772	1596	581

Table 4

Time Taken to Encode and Decode Ratios of Fibonacci Numbers, $f_1 = f_2 = 1, f_{n+1} = f_n + f_{n-1}$. Each entry in the table shows the time taken in milliseconds for 10,000,000 successive repetitions of the encoding or decoding operation.

time to compute.

So the project of identifying the natural encoding of rational numbers within sets unexpectedly leads to a more compact representation of those numbers and a faster algorithm for computing them.

VI. Evaluation Rules for Division

A rational number is naturally encoded by a set of the form $s_k 2_V \cdots s_2 2_V s_1 2_V s_0$ so there must be rules of evaluation which convert a product of an integer and a reciprocal into this form.

For simplicity, we can express the rules for multiplication using ordered pairs in which the second entry is zero:

$$\text{“} \cdots \times x\text{”} = (x\diamond, 0). \tag{91}$$

This represents the start of a sequence which is initially at zero and has a step size of x. The requirement that evaluation must respect associativity ensures that other expressions involving multiplication which can be transformed into an expression containing this one will evaluate to the correct result.

If x is an integer, then:

$$[1(x\diamond, 0)] = [1(x, 0)\diamond] = [(x\diamond, 0)\diamond x] \tag{92}$$

which says:

$$\text{“}(1 + \cdots) \times x\text{”} = \text{“}(-(1 + \cdots) \times -x)\text{”} = \text{“} \cdots \times x + x\text{”}. \tag{93}$$

If we consider the product of a natural number greater than 1 with a reciprocal of a natural number:

$$\text{“}(\cdots + 2 + n) \times \frac{1}{2+m}\text{”} = n2(m2_V\diamond, 0) \tag{94}$$

then the integer part of the fraction $(2+n)/(2+m)$ is generated by the evaluation rule:

$$[m2(m2_V\diamond, 0)] = [(m2_V\diamond, 0)1] \tag{95}$$

which is equivalent to:

$$[m2(0, m2_V)] = [(0, m2_V)1]. \tag{96}$$

After that rule has been applied to any expression which multiplies a natural number by a reciprocal, the resulting expression will contain a natural number, a, plus a fractional part with a denominator equal to $2 + m$.

This is equivalent to $a + 1$ minus a different fraction with the same denominator, which generates the integer part, $s_0 = \diamond 1a$, of the natural encoding of the rational number $s_k 2_V \cdots s_2 2_V s_1 2_V s_0$.

If the numerator of the subtracted fraction is zero, the rational number that results from the division has no fractional part and the process of division is finished. If it is 1, then the rational number is of the form $s_1 2_V s_0$, where $s_1 = \diamond m$, and the division is finished.

So we can assume that the remaining numerator is at least 2, and we can also assume that it is less than half of the denominator, since the rule $\diamond 2_V = 2_V \diamond 1$ can be used to replace n with $m - n$ if it is greater than half of the denominator, with the consequence that the integer s_1 that results will be negative.

The resulting fraction can be expressed as:

$$\frac{n_1 + 2}{m_1 + 2} = \frac{1}{\frac{m_1+2}{n_1+2}} = \frac{1}{2 + \frac{m_1+2}{n_1+2} - \frac{2n_1+4}{n_1+2}} = \frac{1}{2 + \frac{m_1-2n_1-2}{n_1+2}} \tag{97}$$

where $m_1 - 2n_1 - 2$ is non-negative. The set which encodes this expression is:

$$m_1 \diamond n_1 n_1 2 \diamond (n_1 2_V \diamond, 0) 2_V \tag{98}$$

and $m_1 \diamond n_1 n_1 2 \diamond$ will evaluate to a non-negative natural number. We can shorten this expression by using the non-positive number $m_1 \diamond n_1 n_1 2$:

$$m_1 \diamond n_1 n_1 2 (n_1 2_V, 0) 2_V \tag{99}$$

which is equivalent under the rules of evaluation to:

$$m_1 \diamond n_1 (n_1 2_V, 0) \diamond 1 \diamond 2_V = m_1 \diamond n_1 (n_1 2_V \diamond, 0) 1 \diamond 2_V. \tag{100}$$

If the numerator in this resulting fraction is greater than $2 + n_1$, then equation 92 will apply, and will eventually result in an expression of the form:

$$h(n_1 2_V \diamond, 0) s_1 2_V \tag{101}$$

where $h < 2 + n_1$ and s_1 is an integer.

If $h = 0$ then the process has finished. If $h = 1$, then it has reached the final reciprocal. If $h > 1$, then the expression can be written as:

$$n_2 2 (m_2 2_V \diamond, 0) s_1 2_V \tag{102}$$

where $m_2 = n_1$ and n_2 is a natural number satisfying $n_2 < m_2$. The above process for n_1 and m_1 can then be repeated with n_2 and m_2, after which the expression for the rational number will have $s_2 2_V s_1 2_V s_0$ on the right of the ordered pair.

The numerators and denominators get smaller in each iteration of this procedure until the process terminates, yielding an expression of the form $(m_k 2_V \diamond, 0) s_k 2_V \cdots s_2 2_V s_1 2_V s_0$ where the final multiplicative operator $(m_k 2_V \diamond, 0)$ has nothing left to multiply and the signs of the integers s_0, s_1 and so on are chosen to ensure that the numerator is less than half of the denominator before each reciprocal is taken.

The evaluation rule which generates the natural representation of the rational number that results from division is therefore:

$$[n2(m 2_V \diamond, 0)] = [m \diamond n (n 2_V \diamond, 0) 1 \diamond 2_V]. \tag{103}$$

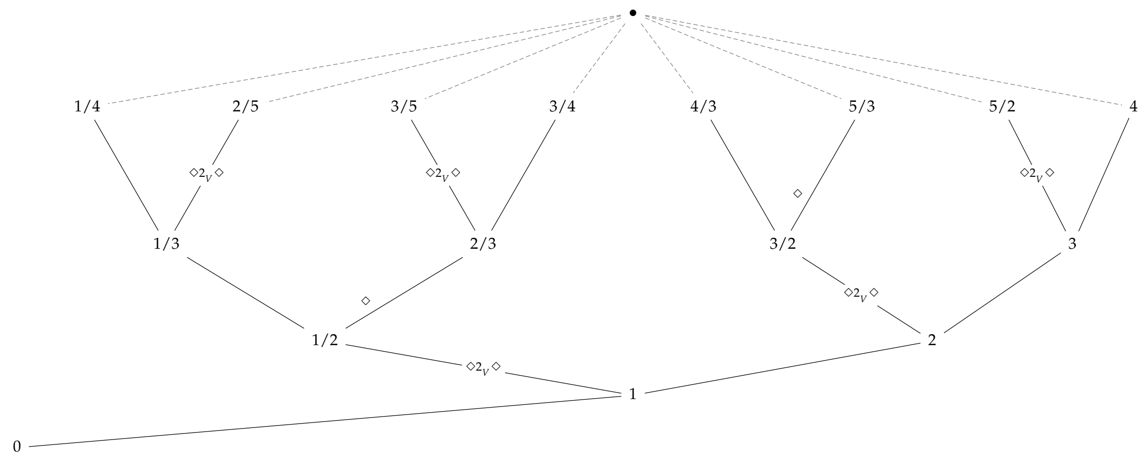


Figure 16
The constituent structure of a set containing several rational numbers. The path connecting 0 to a number, s, specifies the set, $s_k 2_V \cdots s_2 2_V s_1 2_V s_0$, which encodes that number, and the sequence of integers, $[s_0; s_1, \cdots, s_k]$. Each edge traversed along the path adds symbols to the left of the expression for the constructed set. Unbroken edges add 1; edges containing $\diamond 2_V \diamond$ add $\diamond 2_V \diamond$; edges marked with $\diamond$ add the $\diamond$ to the expression, possibly cancelling an existing $\diamond$, before the 1 specified by the edge is added. The set for $5/3$, for example, is $1 \diamond \diamond 2_V \diamond 2 = 1(2_V \diamond 2)$, indicating the numeric value $2 - \frac{1}{3}$ and the natural representation $[2; 1]$.

VII. What Does the Natural Encoding of a Rational Number Mean?

For natural numbers, the set which encodes a number has a very clear relation to that number, and the constituent structure of the set clearly mirrors the structure of the number. With the integers, once we become comfortable with the concept that the structure of the $\diamond$ set naturally reverses the direction of a sequence, the constituent structure of the set that represents an integer can be understood as an accurate representation of the integer itself.

In this case, the set, $s_k 2_V \cdots 2_V s_1 2_V s_0$, representing a rational number, s, doesn't match our concept of a fraction as a numerator and a denominator. Its constituent structure is just a restatement of its expression: several integers with instances of 2_V between each integer and the next.

We would like to have a clear understanding of what the integers mean, not as an input to an algorithm which will compute the rational number, but as a part of the rational number's structure.

The natural numbers and their reciprocals taught us that distinct numbers represented by distinct sets can have identical constituent structures, and we can see how they relate to one another in the constituent structure diagram of a set that contains both of those distinct sets.

So we will consider a set which contains multiple rational numbers and natural numbers as constituents. Figure 16 shows the constituent structure of the set $\{1/4, 2/5, 3/5, 3/4, 4/3, 5/3, 5/2, 4\}$.

Each node in the diagram corresponds to a rational number. The structures of the $\diamond$ and 2_V sets situated on top of numbers have been omitted for clarity. Symbols along the edges connecting one number to another indicate the sets which must be added to the set for one number to reach the set for the other, after evaluating the resulting expression.

The edge connecting 1 to $1/2$ contains $\diamond 2_V \diamond$, which means that evaluating the expression $\diamond 2_V \diamond 1$

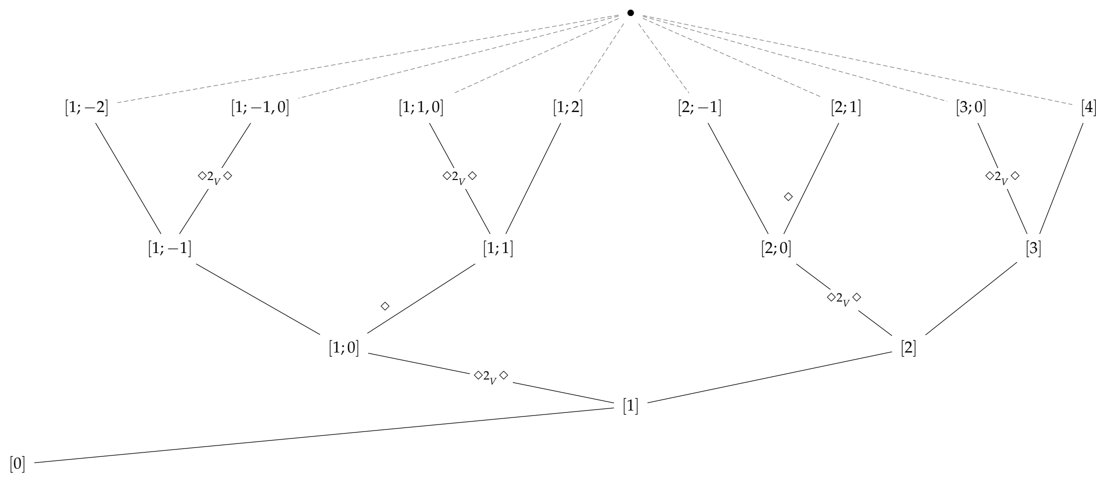


Figure 17
The natural representations of the rational numbers specify their locations within the constituent structure of the sets which encode and contain them.

produces $1/2 = 2_V$:

$$[\Diamond 2_V \Diamond 1] = [\Diamond [2_V \Diamond 1]] = [\Diamond \Diamond 2_V] = 2_V.$$

The diagram provides insight into the meaning of the integers, $[s_0; s_1, \cdots, s_k]$, which appear in the continued fraction expression for a rational number encoded using 2_V. The integers specify a route within this binary tree structure that connects 0 to the specific rational number encoded.

A positive integer, $\Diamond n$, or a negative one, $n\Diamond$, specifies the number, n, of consecutive steps along the same direction that should be followed before changing direction or reaching the end of the route.

Successive integers in the set representing a rational number, $s_k 2_V \cdots s_2 2_V s_1 2_V s_0$, are separated by an occurrence of 2_V, which changes the direction of the route through the binary tree by traversing one of the edges labelled with $\Diamond 2_V \Diamond$ in figure 16.

This brings the route into a different sequence of numbers by introducing the reciprocal:

$$m + \cdots \longrightarrow m - \frac{1}{2 + \cdots}$$

The sign of the integer which follows the 2_V determines which of the two edges that proceed away from the current number will be followed in the next step. Although it may seem counterintuitive, a negative integer, $n\Diamond$, specifies that the edge without the $\Diamond$ symbol should be traversed. The reason for this is that integers subtract numbers that are supplied to them on the left, so in the expression for the set, $s_k 2_V \cdots s_2 2_V s_1 2_V s_0$, each instance of 2_V is subtracted from the integer on its right which precedes it in the construction of the expression for the entire set, leading to a downward sequence which a subsequent negative integer can reverse.

In figure 16, each step from an integer across an edge labelled with $\Diamond 2_V \Diamond$ effectively subtracts $1/2$ from the integer, and further steps in that direction lead to a decreasing sequence of numbers.

The natural representations of the rational numbers are shown in figure 17 within this structure. The fact that the integers which represent a rational number encode the route to it within the tree is visible from their pattern: All of the nodes which can be reached by traversing the $\Diamond 2_V \Diamond$ edge from 2 have 2 as their first entry, and the same is true of the other integers.

The route to the number with the sequence $[1;1,0]$ goes through the number with the sequence $[1;1]$, and the route to that number goes through $[1]$.

The representation in figure 17 makes it evident that which numbers are constituents of which others is indicated in their natural representation, with 2 being a constituent of $[2;-1]$ and so on. In contrast, figure 16 shows that $1/5$ is not a constituent of $3/5$, and 5 and 3 are also not constituents of it, which reveals that the numerator and denominator of a rational number don't specify its constituents.

It is also worth noting that the standard continued fraction representation of a rational number also doesn't reveal its constituents. For example, $1/3$ is a constituent of $2/5$, but the standard continued fraction representation of $1/3$ is $[0;3]$, while that of $2/5$ is $[0;2,2]$.

Only the natural representation clearly expresses the constituency relation, with $2/5$ having the representation $[1;-1,0]$ which contains the representation of $1/3$, which is $[1;-1]$.

The binary tree structure shown in figures 16 and 17 is already known to mathematicians as the Stern-Brocot tree[5]. Many of its properties, such as the occurrence of each rational number exactly once in its simplest form, the ordering of numbers from left to right in accordance with their numerical order, and its relation to standard continued fractions, have been known for more than a century and a half[6, 7], as has its usefulness for identifying integers whose ratio best approximates a given number.

What has not previously been made clear is that the tree displays the structure of the rational numbers themselves. Without the concept of constituent structure, the relationship between the numbers in the tree has always seemed to be arithmetical and algorithmic. Unlike the standard continued fraction representation, each rational number's natural representation is unique, and every finite sequence of integers in which all of them, apart from the first and the last, are required to be non-zero, identifies a unique rational number which can be reached by following that route in the tree.

This includes the negative rational numbers, which are naturally included in the tree as shown in figures 18 and 19. The only pattern which is broken by their inclusion is that there are three edges connecting upward from the number 0 which appears at the root of the tree. The numbers 1, -1 and $-1/2$ can all be reached from 0 by traversing a single edge.

Zero is the only number which can have three edges from it because $[0;0]$, which indicates the number $-1/2$, is the only natural representation that contains two consecutive zeros. The sign of an integer appearing in a natural representation encodes the information about whether the $f \leftrightarrow 1-f$ operation needs to be applied before taking the corresponding reciprocal.

There are only two positions in the list for which this information isn't necessary. Specifically, it isn't needed for the first integer in the list, which is the integer part of the rational number and is not involved in a reciprocal, and it isn't needed for the last integer in a list with two or more integers in it, if that integer is zero, indicating $f = 1/2 = 1-f$.

So $[0]$ is the only case when a zero can be added after a list of integers which already ends in zero. Every other node in the tree falls into one of two categories:

- The final integer in the list is negative or positive, and the two available moves are "go forward" to the next positive or negative integer and "branch to a new reciprocal", which adds a 0 to the end of the list.

- The final integer is zero, and the two available moves are "go in the positive direction" and "go in the negative direction", which change the final 0 to 1 and -1 respectively.

There are new, overlapping symmetries present and visible in figure 18 in the full tree which includes negative numbers, which aren't present in the purely positive part of the tree. This specific way of connecting the negative numbers is dictated by the natural representations of the rational numbers. The systematic arrangement of the natural representations is visible in figure 19, which makes it clear that the same rules apply to positive and negative rational numbers.

This structure can be contrasted with the tree shown in figure 20, which shows how one might expect an extension of the Stern-Brocot tree to negative numbers to appear. While it appears at first to be quite symmetric, each symmetry within it is localized to the specific binary tree with the number around which the symmetry manifests at the root. Different symmetries don't overlap, except in the trivial case when one tree is entirely within another, because they don't extend outside of their local binary trees.

Symmetries that don't extend outside of their own binary tree, and don't overlap with other symmetries, are, in a sense, trivial, because the constraints they impose can always be satisfied, regardless of the objects that are arranged within the tree and how the symmetries relate them.

For example, if we are building a tree with objects at each vertex, whose root is x, and we require any objects, a and b, positioned symmetrically around x to satisfy $a = f(b)$ for some function, f, then we can build the tree on the left side of x first, and then use f to determine what object should be at each position in the tree on the right side of x. If we choose x_l to be the object at the vertex reached by moving left from x, then the binary tree starting from x_l is not constrained at all by the symmetry around x. We can therefore impose any symmetry we want on that tree, choosing a new, arbitrary function, g, which determines the objects on the right side given the objects on the left, without any consideration of the symmetries already imposed.

However, if a tree has multiple overlapping symmetries, each of which extends beyond the binary tree whose root is the line of reflection for that symmetry, then the functions which implement the symmetries, f, g and so on, cannot be arbitrary. They must obey an algebra which makes them compatible with one another, and if the functions are simple, natural operations on the objects at the vertices of the tree, then the algebra they satisfy reveals information about the structures of those objects.

In our case, with the symmetries $f(x) = -1 - x$ and $g(x) = {}^1\!/_x$, where f reflects a number around $-{}^1\!/_2$ and g reflects positive numbers around 1 and negative numbers around -1, the functions need to satisfy $fgf = gfg$, or, equivalently, $fgfgfg = 1$, in order for the two numbers symmetric around $-{}^1\!/_2$ that are obtained from two positive reciprocals, x and ${}^1\!/_x$, by reflecting them first around $-{}^1\!/_2$ and then around -1, to satisfy the symmetry around $-{}^1\!/_2$.

So the symmetries of the tree specified by the constituent structures and natural representations of rational numbers specify an algebra, $f^2 = g^2 = (fg)^3 = 1$, which encodes certain properties of numbers and corresponding operations on them. This is the algebra of the group, D_3, of symmetries of an equilateral triangle, with f, g, and $fgf = gfg$ corresponding to reflections through the lines which bisect the angles of the triangle, and fg corresponding to rotation through 120 degrees.

One might observe that the tree dictated by constituent structure is not a binary tree, but has a symmetry which binary trees, as commonly defined, lack: Every vertex in it, including the root at zero, has exactly three edges connected to it. If we ignore constituent structure and arrange these three edges at zero in the same way as every other vertex's edges are arranged, we get the tree shown in figure 21, which is an infinite binary tree with no root.

This tree has a more uniform structure than the tree shown in figure 18, but the arrangement of the numbers within it has fewer symmetries, none of which overlap.

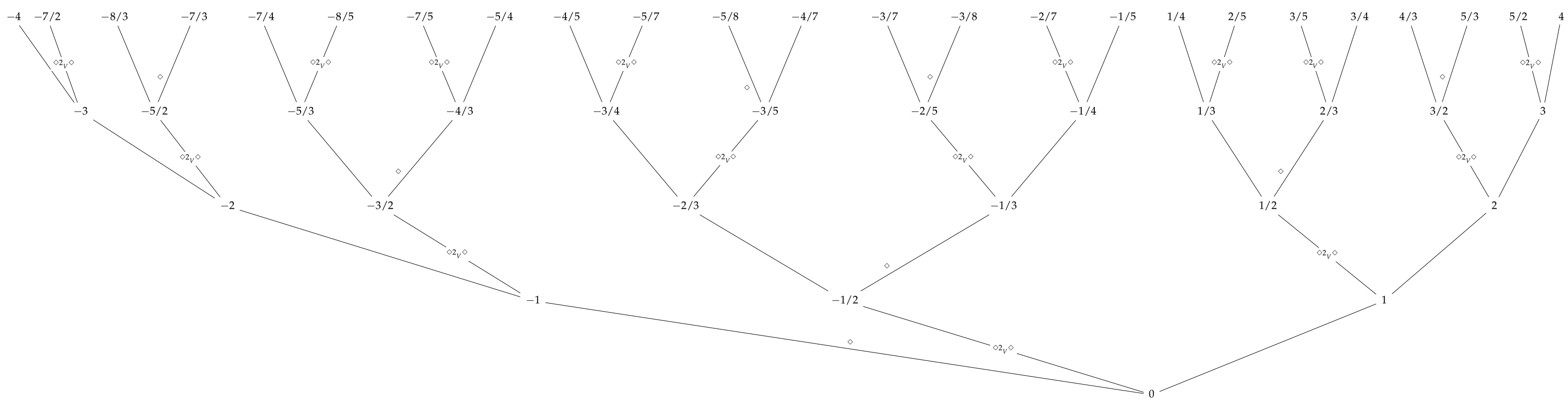


Figure 18
The Stern-Brocot tree extended to include all rational numbers, including negative numbers, as specified by their constituency relations and natural representations. Various symmetries are evident: Apart from integers, numbers symmetrically located around $-\frac{1}{2}$ *sum to* -1*; numbers symmetric around* 1 *or* -1 *are reciprocals; numbers a and b which are symmetric around* 0 *satisfy* $a = {}^{-b}/_{(1+b)}$*. Symmetries extend outside the binary tree of the number around which the symmetry appears:* $\frac{5}{2}$ *and* $-\frac{7}{2}$ *sum to -1 despite being outside the binary tree that starts from* $-\frac{1}{2}$*. The symmetries overlap: The reciprocals symmetric around* -1 *extend from* $-\frac{7}{2}$ *to* $-\frac{2}{7}$*, covering almost two thirds of the width of the tree, while the* ${}^{-b}/_{(1+b)}$ *symmetry around* 0 *extends from* $-\frac{4}{5}$ *to* 4*, covering two thirds of the tree and overlapping with the former symmetry in the middle third.*

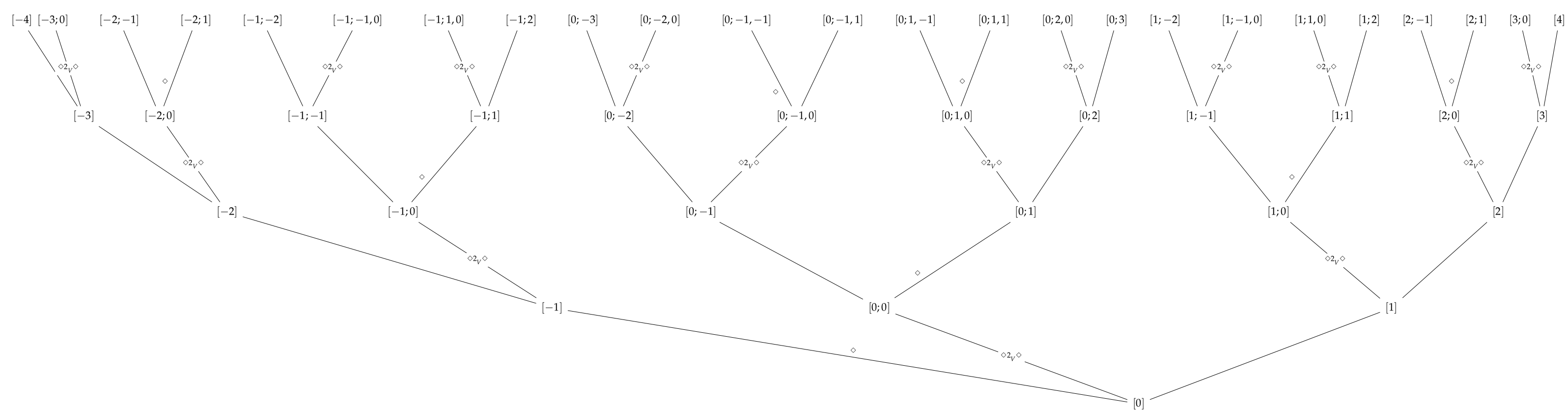


Figure 19

The natural representations of positive and negative rational numbers. Note that, like the rational numbers themselves, the lists of integers appear in increasing order from left to right, when the earliest integers in each list have the highest priority in determining their order. $[-4]$ *is to left of* $[-3;0]$ *because* $-4 < -3$ *and so on. The tree is symmetric around* $[0;0] = -1/2$ *from that level upwards. Sequences symmetrically located around* $[0;0]$ *have opposite signs. The height of a sequence within the tree is equal to the sum of the absolute values of its entries plus its length.*

Within the tree shown in figure 19, the sequences of integers can be seen to exhibit a consistent order in their horizontal and vertical arrangement.

Considering the bottom of the tree containing [0] to have a vertical position or height of 1, the vertical position of a sequence of integers is equal to the length of the sequence plus the sum of the absolute values of its entries. So $-4/7 = [0; -1, 1]$ appears at the 5^{th} level in the list, because $5 = 3 + |0| + |-1| + |1|$.

The width of the tree at that height, $h = 5$, is given by $3 \times 2^{h-2} = 24$, since the total tree consists of three binary trees originating from level 2. The natural representations of each of those rational numbers can be determined from the rule that the length of each natural representation plus the sum of the absolute values of the integers in it is equal to h.

As figure 19 shows, the $3 \times 2^{h-2}$ natural representations at a given height in the tree appear in lexicographical order from left to right. So $[-4] < [-3; 0]$ because $-4 < -3$, and $[0; 2, 0] < [0; 3]$ because $2 < 3$.

This allows us to specify that, for a given sequence, its horizontal position in the tree is equal to its position within the sorted list of sequences at that height.

Equation 104 below gives a formal specification of the order of sequences which define natural representations of rational numbers. Sequences at the same height in the tree will always differ at some position, but the order extends to the entire tree, and to all natural representations, when it is also specified that sequences of two or more integers which end in negative numbers are less than all sequences that continue them, and all other sequences are greater than sequences that continue them:

$$\begin{aligned}
b_1 > c_1 &\implies & [a_0; a_1, \cdots, a_n, b_1, \cdots, b_m] &> [a_0; a_1, \cdots, a_n, c_1, \cdots, c_k] \\
a_n > 0 \text{ OR } n = 0 &\implies & [a_0; a_1, \cdots, a_n] &> [a_0; a_1, \cdots, a_n, a_{n+1}, \cdots, a_m] \\
a_n < 0 \text{ AND } n > 0 &\implies & [a_0; a_1, \cdots, a_n] &< [a_0; a_1, \cdots, a_n, a_{n+1}, \cdots, a_m]
\end{aligned} \tag{104}$$

So natural representations have the same order, as sequences of integers, as the rational numbers they represent. If $s(f)$ is the sequence of integers for a given fraction, f, then

$$s(f_1) \leq s(f_2) \iff f_1 \leq f_2$$

when sequences are ordered according to equation 104.

This is not true of the standard continued fraction representation of rational numbers, for which $3/8$ and $7/8$ have the sequences $[0; 2, 1, 2]$ and $[0; 1, 7]$, and these sequences are on the same side of $[0; 1, 1, 1, 2]$, which represents $5/8$, in lexicographic order or any simple ordering of sequences based on their entries, although it is possible to define an ordering on standard continued fraction representations which coincides with the order of the rational numbers they represent, by flipping the sign of every second entry and then taking the lexicographic order.

The standard continued fraction representation is also not unique, but can be made unique by requiring the final denominator to be greater than or equal to 2. Like the sign-flipping procedure which makes the ordering of continued fraction representations match the order of the rationals, and which suggests subtraction rather than addition of the next reciprocal, requiring the final denominator to be greater than or equal to 2 brings the standard continued fraction representation closer to the natural representation.

These are hints from the mathematics of continued fractions that a more fundamental representation of a rational number exists, but it was the constituent structure of the simplest sets which led us to it.

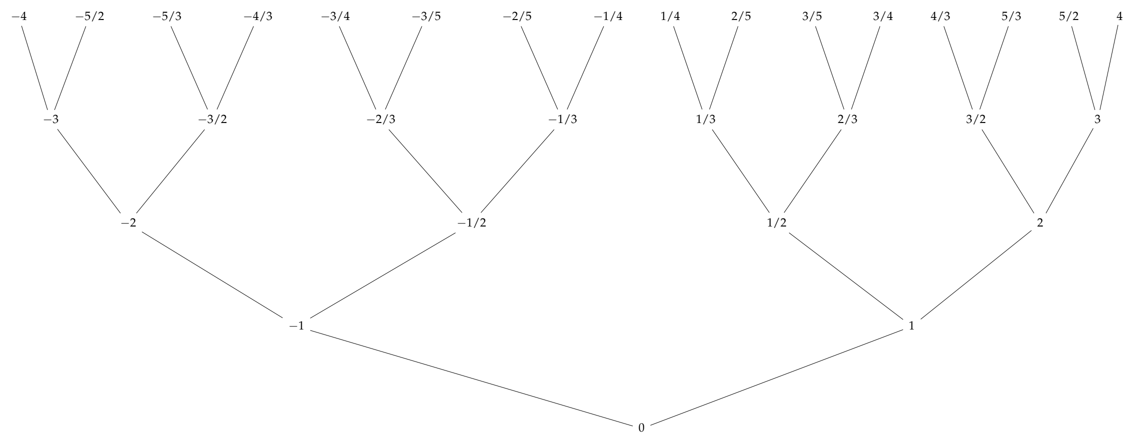


Figure 20

The most intuitively obvious way to extend the Stern-Brocot tree to negative rationals is to take the mirror image of the positive tree and connect the vertices at 1 *and* -1 *to a single root at* 0*. The resulting tree contains all rationals in the right order, but is not generated in its entirety by any single procedure, has ad-hoc rules for the vertex at zero, which differ from the rules for other vertices, and has no overlapping symmetries. Only the numbers within the local binary tree originating from a given number exhibit a symmetry around that number: Numbers symmetrically positioned around* 1 *are inverses; and the same is true for* -1*, but there are no numbers which participate in both symmetries. Numbers symmetric around* $-\frac{1}{2}$ *sum to* -1*, but only if they are within the binary tree originating from* $-\frac{1}{2}$*.*

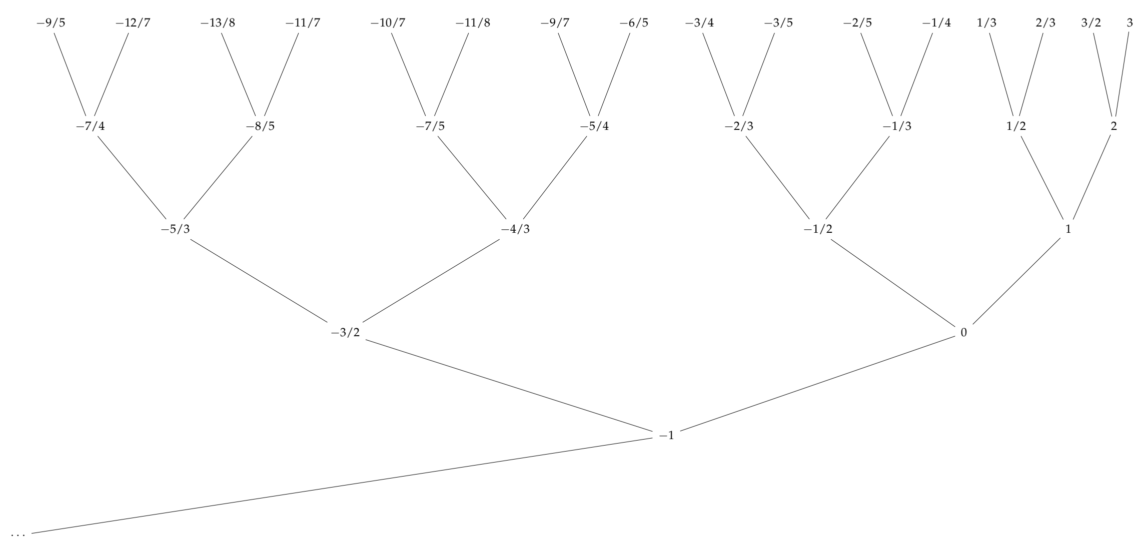


Figure 21

Another way to include negative rationals is to repeat the operation which yields the binary tree which starts at 1 *from the tree which starts at* 2 *by subtracting* 1 *from each number. Subtracting* 1 *from every number in the tree of positive rationals generates a tree whose root is zero and which includes the negative numbers greater than* -1*. Repeating this indefinitely generates an infinite binary tree with no root which includes every rational number. This tree has fewer symmetries than the tree in figure 18. Numbers symmetric around* $-\frac{1}{2}$ *sum to* -1*, but only within its own binary tree. Numbers symmetric around* -1 *are not reciprocals. Symmetries are restricted to the binary tree whose root is the number around which the symmetry manifests, so they don't overlap with other symmetries.*

Table 5
The Encoding of Arithmetic Within Finite Sets

	Natural Numbers	Integers	Rational Numbers
Numbers	$0 = \{\}$ $n = \{n-1\}$	$\Diamond = \{2, 2_V\}$ $+n = \Diamond n$ $-n = n\Diamond$	$2_V = \{0, 1\}$ $\frac{1}{2+n} = n2_V$
Expressions	$“n + \cdots” = n$ $“n + m + \cdots” = mn$	$“x - \cdots” = x$ $“x - y - \cdots” = \Diamond yx$ $“x + y - \cdots” = y\Diamond x$ $“(1 + \cdots) \times y” = (0, y)$ $“(e) - \cdots” = (e1, e)$	$“\frac{1}{2+n+\cdots}” = n2_V$ $“\frac{1+\cdots}{2+n}” = (0, n2_V)$ $“\frac{1}{2+n} - \cdots” = (n2_V1, n2_V)$
Evaluation		$1\Diamond 1 \to \Diamond$ $\Diamond\Diamond \to \{\}$ $1(0, e) \to (0, e)\Diamond(e1, e)$ $(b, (c, d)) \to (b, d)$ $((a, b), d) \to (b, d)$	$\Diamond 2_V \to 2_V \Diamond 1$ $m2(0, m2_V) \to (0, m2_V)1$ $n2(m2_V\Diamond, 0) \to m\Diamond n(n2_V\Diamond, 0)1\Diamond 2_V$
Notation	$ab \equiv a(\{\} \to b)$	$(a, b) \equiv \{\{\Diamond a\}, \{\Diamond a, \Diamond b\}\}$	

X. Summary

All of the objects we deal with in mathematics have constituents, while sets also have elements and cardinality. When we construct mathematical objects from sets, we should therefore encode the constituent structures of the objects in the constituent structures of the sets, not partially there and partially in the specific details of their elements and cardinality.

If we need to refer to the cardinality or elements of the underlying set which represents an object in order to know something about that object, then we can never let go of the arbitrary choices made during the construction of the object. The properties of the object will be inseparable from the properties of the underlying set, and our representation of the object will be unnecessarily confusing and complicated.

We successfully construct an object from sets if we articulate the object's structure within a set, and can then discard all information about, and end all talk of, elements and cardinality, retaining the object's structure and abstracting from the details of its construction.

Functions between sets, which map elements to elements, are blind to most of the structure within a set, which resides deep within layers of nested sets. Bijective functions, which prove that two sets have the same cardinality, provide no information about whether the sets are suitable representations of the same kind of object.

Whether two distinct sets constructed by two distinguished mathematicians represent the same object in the same branch of mathematics, or, by chance, objects with the same structure in different branches, is determined by the existence or non-existence of a constituent structure isomorphism between the two sets, which may have very different cardinalities. One set may have a single element, and be isomorphic to another set with very many elements.

In the case of the arithmetic of reciprocals, it is the existence of an isomorphism between the constituent structure of a number and its reciprocal which proves that they match and can cancel each other. Isomorphisms between the underlying sets of distinct objects with identical structures, in general, like in this case, will map some set to another set with a different number of elements, while still being injective, surjective and invertible. Functions will never be sufficient to detect the matching structures.

One of the principles of set theory is that a set contains only one instance of each of its elements. A set with two identical elements has one element. The concept of a constituent extends this removal of redundancy beyond a set's top layer. A set can have many copies of another set inside it, but from the point of view of functions, subsets and elements, no repetition is detectable. The constituent structure of the set is inaccessible when there's no way to express the fact that those repeated instances are all the same thing.

When we require our constructions of mathematical objects as sets to encode the object solely in the set's constituent structure, the resulting encoding is simple, natural, usually matches our conceptual understanding of the object, and contains a natural representation of the object's operations and relations as operations and relations on sets. As an example, the encoding of arithmetic within the structure of sets is simple enough to display in table 5.

Considering the constituent structures of the sets which naturally encode the rational numbers leads to several new insights, including a natural order-preserving representation of rational numbers as finite sequences of integers and an efficient algorithm for finding rational approximations, as well as a unique extension of the Stern-Brocot tree to include all rational numbers within a single tree, which exhibits new overlapping symmetries whose algebra is the same as the group of symmetries of an equilateral triangle.

These unexpected results suggest that the category whose objects are sets and whose morphisms are constituent structure homomorphisms is likely to be a more interesting and fruitful object of study than the category of sets and functions.

A. Natural Representation and Continued Fraction Implementations

This code is available on github at:
`https://github.com/roflanagan/natural-representation`

I. Implementation in Python

```python
from fractions import Fraction
from math import floor
from numpy import array

def natural_representation(f):
    """
    Compute the sequence of integers in the natural representation of the fraction f
    """
    integer_part = int(floor(f))
    if f == integer_part:
        return [integer_part]
    fractional_part = f - integer_part
    if fractional_part >= 0.5:
        return [integer_part + 1] + natural_representation(1 / (1-fractional_part) - 2)
    else:
        rest = array(natural_representation(1 / fractional_part - 2))
        return [integer_part + 1] + list(-rest)

def evaluate_natural_representation(sequence):
    """
    Compute the fraction f from the sequence of integers in its natural representation
    """
    rest_of_sequence = array(sequence[1:])
    if len(rest_of_sequence) == 0:
        return sequence[0]
    if rest_of_sequence[0] >= 0:
        rest = evaluate_natural_representation(rest_of_sequence)
        return sequence[0] - Fraction(1, 2 + rest)
    else:
        rest = evaluate_natural_representation(-rest_of_sequence)
        return sequence[0] - 1 + Fraction(1, 2 + rest)

def show_examples(max_numerator, max_denominator):
    """
    Encode and decode all positive and negative rational numbers whose numerators
    and denominators don't exceed the specified limits
    """
    from fractions import gcd as greatest_common_divisor
    for n in range(-max_numerator, max_numerator + 1):
        for d in range(1, max_denominator + 1):
            if greatest_common_divisor(n, d) == 1:
                fraction = Fraction(n, d)
                sequence = natural_representation(fraction)
                recovered_fraction = evaluate_natural_representation(sequence)
                print fraction, "\t-->", str(sequence).ljust(12), "-->", recovered_fraction


if __name__ == "__main__":
    show_examples(9, 9)
```

II. Implementation in C

```c
#include <stdio.h>
#include <math.h>
#include <string.h>
#include <sys/time.h>
#include <stdlib.h>

void continued_fraction(unsigned long numerator, unsigned long denominator, int *sequence, int *i)
{
    unsigned long new_numerator;
    *i = 0;

    while (denominator > 1)
    {
        sequence[*i] = numerator / denominator;
        new_numerator = numerator - sequence[*i] * denominator;
        numerator = denominator;
        denominator = new_numerator;
        *i = *i + 1;
    }
    sequence[*i] = numerator;
    *i = *i + denominator;
}

void evaluate_continued_fraction(unsigned long *numerator, unsigned long *denominator,
                                 int *sequence, int *sequence_size)
{
    unsigned long old_numerator;
    int i = *sequence_size - 1;
    *numerator = sequence[i];
    *denominator = 1;

    while (i > 0)
    {
        i -= 1;
        old_numerator = *numerator;
        *numerator = *denominator + sequence[i] * *numerator;
        *denominator = old_numerator;
    }
}

void natural_representation(unsigned long numerator, unsigned long denominator, int *sequence, int *i)
{
    int sign = 1;
    unsigned long integer_part, fractional_part, old_denominator;
    *i = 0;

    while (1)
    {
        integer_part = numerator / denominator;
        fractional_part = numerator - integer_part * denominator;
        sequence[*i] = sign * (integer_part + (fractional_part > 0));
        *i = *i + 1;
        if (2 * fractional_part >= denominator)
        {
            denominator = denominator - fractional_part;
            numerator = fractional_part - denominator;
        }
        else
        {
            if (fractional_part == 0)
                return;
            old_denominator = denominator;
            denominator = fractional_part;
            numerator = old_denominator - 2 * fractional_part;
            sign *= -1;
        }

    }
}
```

```
void evaluate_natural_representation(unsigned long *numerator, unsigned long *denominator,
                                     int *sequence, int *sequence_size)
{
    unsigned long new_denominator, old_numerator;
    int i = *sequence_size - 1, previous_sign = (sequence[i] >= 0);
    *numerator = abs(sequence[i]);
    *denominator = 1;

    while (i > 0)
    {
        i -= 1;
        old_numerator = *numerator;
        new_denominator = old_numerator + 2 * *denominator;
        *numerator = abs(sequence[i]) * new_denominator - *denominator;

        if ((sequence[i] < 0) == previous_sign)
        {
            *numerator -= old_numerator;
            previous_sign = 1 - previous_sign;
        }
        *denominator = new_denominator;
    }
}

unsigned long current_time()
{
    struct timeval tv;
    gettimeofday(&tv,NULL);
    return 1000000 * tv.tv_sec + tv.tv_usec;
}

void repeatedly_construct_continued_fraction(unsigned long numerator,
                                             unsigned long denominator,
                                             int *sequence, int *sequence_size, int iterations)
{
    long start, i;
    printf("continued fraction:\n");

    start = current_time();
    for (i = 0; i < iterations; i++)
        continued_fraction(numerator, denominator, sequence, sequence_size);

    printf("completed in %ld microseconds\n", current_time()-start);

    for (i=0 ; i < *sequence_size; i++)
        printf("%d, ", sequence[i]);
    printf("\n");
}

void repeatedly_evaluate_continued_fraction(unsigned long numerator,
                                            unsigned long denominator,
                                            int *sequence, int *sequence_size, int iterations)
{
    long start, i;
    printf("\n\nevaluation of continued fraction:\n");

    start = current_time();
    for (i = 0; i < iterations; i++)
        evaluate_continued_fraction(&numerator, &denominator, sequence, sequence_size);

    printf("completed in %ld microseconds\n", current_time()-start);
    printf("numerator: %lu   denominator: %lu\n\n", numerator, denominator);
}

void repeatedly_construct_natural_representation(unsigned long numerator,
                                                 unsigned long denominator,
                                                 int *sequence, int *sequence_size,
                                                 int iterations)
{
    long start, i;
```

```c
    printf("\n\n\nnatural representation:\n");

    start = current_time();
    for (i = 0; i < iterations; i++)
        natural_representation(numerator, denominator, sequence, sequence_size);

    printf("completed in %ld microseconds\n", current_time()-start);

    for (i=0 ; i < *sequence_size; i++)
        printf("%d, ", sequence[i]);
    printf("\n\n");
}

void repeatedly_evaluate_natural_representation(unsigned long numerator,
                                                unsigned long denominator,
                                                int *sequence, int *sequence_size,
                                                int iterations)
{
    long start, i;

    printf("\n\nevaluation of natural representation:\n");

    start = current_time();
    for (i = 0; i < iterations; i++)
        evaluate_natural_representation(&numerator, &denominator, sequence, sequence_size);

    printf("completed in %ld microseconds\n", current_time()-start);
    printf("numerator: %lu   denominator: %lu\n\n", numerator, denominator);
}

int main(int argc, char **argv)
{
    unsigned long numerator, denominator;
    int sequence[BUFSIZ], sequence_size, iterations = 10000000;

    if (argc < 3)
    {
        printf("Usage: %s numerator denominator\n", argv[0]);
        return 0;
    }
    sscanf(argv[1], "%lu", &numerator);
    sscanf(argv[2], "%lu", &denominator);

    printf("numerator: %lu   denominator: %lu\n\n", numerator, denominator);

    repeatedly_construct_continued_fraction(numerator, denominator, sequence, &sequence_size, iterations);
    repeatedly_evaluate_continued_fraction(numerator, denominator, sequence, &sequence_size, iterations);

    repeatedly_construct_natural_representation(numerator, denominator, sequence, &sequence_size, iterations);
    repeatedly_evaluate_natural_representation(numerator, denominator, sequence, &sequence_size, iterations);
}
```

Acknowledgement

The author is deeply grateful to Venkata Rayudu Posina, whose patient explanations of category theory made the need for a category of sets modulo cardinalities visible, and also to David Marin Roma for useful discussions and suggestions which improved the readability of the article. VRP also articulated the significance for the category of constituent structures of the product and sum constructions in terms of universal properties.

References

[1] Von Neumann, John (1923)
Zur Einführung der transfiniten Zahlen
Acta litterarum ac scientiarum Ragiae Universitatis Hungaricae Francisco-Josephinae, Sectio scientiarum mathematicarum, 1: 199-208

[2] Levy A. (1979)
Basic set theory
Springer
ISBN 0486420795

[3] Kuratowski, Casimir (1921)
Sur la notion de l'ordre dans la Théorie des Ensembles
Fundamenta Mathematicae. 2 (1): 161-171
`http://matwbn.icm.edu.pl/ksiazki/fm/fm2/fm2122.pdf`

[4] Conway, John (1976)
On Numbers and Games
CRC Press
ISBN 9781568811277

[5] Austin, David (2008)
Trees, Teeth, and Time: The mathematics of clock making
American Mathematical Society, Feature Column, December 2008
`http://www.ams.org/publicoutreach/feature-column/fcarc-stern-brocot`

[6] Stern, Moritz A. (1858)
Ueber eine zahlentheoretische Funktion
Journal für die reine und angewandte Mathematik, 55: 193-220
`http://www.digizeitschriften.de/resolveppn/GDZPPN002150301`

[7] Brocot, Achille (1861)
Calcul des rouages par approximation, nouvelle méthode
Revue Chronométrique, 3: 186-194
`https://gallica.bnf.fr/ark:/12148/bpt6k1661912/f92.image` [English version]