PURE DATA SPACES....DRAFT.

SAUL YOUSSEF
Department of Physics
Boston University

August 6, 2025

Abstract

All about spaces.

1 Overview

....WRITE INTRODUCTION...

- Conceptual overview: small(est?) axiomatic system analogous to ZFC, where proof and computation are the same thing.
- Point of the paper is to investigate the concept of a "Space" introduced in ref. 1. This is analogous to "Set" or "Category."
- Theme is to "grow the simplest spaces organically" and compare the simplest results with classical mathematics.
- Minimal container...associative data...semiring...analogous to sets or categories.

2 Foundation

In [1], we argue that Mathematics and Computing as a whole can be captured within a small axiomatic framework, based on **finite sequence** as the foundational concept. Assuming that finite sequences are understood, *data* and *coda* are defined by

Definition 1. **Data** is a finite sequence of **codas**, where each **coda** is a pair of **data**.

The two foundational operations defined on data are 1) concatenation of data A and data B, written 'A B', and 2) pairing of data A and data B as a coda, written with a colon '$A : B$'. By Definition 1, the empty sequence, written '()', qualifies as data and, therefore, () paired with itself is a coda, written '(():())' or '(:)'. Any finite sequence of codas is data, so, for example, (:) (:(:)) ((:):(:(:))) is data consisting of a sequence of three codas. We can think of this as *pure data* since it is "data made of nothing." By convention, the colon binds from the right first and binds less strongly than concatenation, so that $A : B : C$ is defined to be $(A : (B : C))$ and $A : B\ C$ is defined to be $(A : (B\ C))$. Data is typically written with upper case, and codas are typically written in lower case. To indicate the left and right data of a coda, we sometimes use L/R superscripts so, for any coda c, $c = (c^L : c^R)$.

All meaning within the system is determined by a chosen partial function from coda to data called a *context*. Contexts are partially ordered by inclusion, so if δ and δ' are contexts, $\delta \leq \delta'$ if δ and δ' are equal on the domain of δ. Given a context δ, equality of data is the equivalence generated by $c \sim \delta(c)$ for any coda c, and by compatibility with concatenation and colon in the sense that if $A \sim B$, then $(A\ X) \sim (B\ X)$, $(X\ A) \sim (X\ B)$, $(A : X) \sim (B : X)$ and $(X : A) \sim (X : B)$ for any data X. This relation $A \sim B$ is denoted $A \overset{\delta}{=} B$, or simply $A = B$ when the context is unambiguous. It follows that if $\delta \leq \delta'$, then $A \overset{\delta}{=} B$ implies $A \overset{\delta'}{=} B$. We say that if A and B are equal then they are "always equal," thinking of moving from δ to δ' as the passage of "time." The fixed points of a context δ are *atoms*. Note that if c is an atom in context δ, and if $\delta \leq \delta'$, then c is still an atom a context in δ'. Thus, atoms are "permanent" and if c is an atom, c is "always" an atom.

Definition 2. Within a given context δ, coda c is an **atom** if $\delta : c \mapsto c$. If data A contains an atom in its sequence, A is **atomic** data. Data A is **invariant** if every coda a in the sequence of A is an atom and if a^L and a^R are both **invariant**.

It follows from the definition that empty data is invariant and atoms a and b are equal if and only if $a^L = b^L$ and $a^R = b^R$. If a and b are atoms, A and B are data, then $(a\ A) = (b\ B)$ and $(A\ a) = (B\ b)$, if and only if $a = b$ and $A = B$. If $A = B$ and A and B are invariant, then A and B are identical as pure data [proof].

3 Genesis

In the proposed framework, all mathematical objects are pure data and all meaning is defined by a chosen context δ and its corresponding equivalence. A **proof** in δ, for example, is just a sequence $A_1 \overset{\delta}{=} A_2 \overset{\delta}{=} \ldots \overset{\delta}{=} A_n$ which can be viewed either as proving A_n given A_1 or as a computation $A_1 \mapsto A_n$. The immediate issue is how to choose δ.

In the beginning, there are no definitions, and the corresponding context is the empty partial function δ_0. Since the domain of δ_0 is empty, it has no fixed points and, therefore no atoms. Thus, the empty sequence is the only invariant, $X \overset{\delta}{=} X$ is the only valid proof, and the only valid computations do nothing $X \mapsto X$ for any data X. To define a non-empty context $\delta_0 \leq \delta$, we need, at least, an invariant specification of the domain of δ as a partial function. Since the empty sequence is the only invariant, the only way to specify if a coda (A:B) is to be in the domain of δ is to require either A or B or both to be the empty sequence. In each of these cases, the coda (:) is within the domain of δ, and so we must, in any case, decide on what (:) maps to. There are three possibilities;

choice 1: $\delta : (:) \mapsto ()$,

choice 2: $\delta : (:) \mapsto (:)$, or

choice 3: $\delta : (:) \mapsto$ *anything other than* () *or* (:).

Since pure data is "made of (:)", choice 1 trivially causes all data to be equal to the empty sequence. On the other hand, choice 3 would mean that for any non-empty data A, the number of (:) atoms in A grows without limit. This would mean, for instance, that nocomputation could produce a final result. Thus, we are constrained to choice 2, where (:) is an invariant atom in δ, and, therefore an atom in any context $\delta \leq \delta'$ as well. It is convenient to generalize choice 2 and to let δ:$(:X) \mapsto (:X)$ for all data X, since this provides a supply of invariant atoms (:), (:(:)), (:(:) (:)),... which can be used to expand the domain of δ.

We adopt a convention for expanding the domain of δ while preserving context partial ordering. A context of the form

$$\delta_a : (a\ A : B) \mapsto \delta_a(A, B) \tag{1}$$

for some invariant atom a is called a **definition**. We conventionally restrict ourselves to contexts $\delta \cup \delta_a \cup \delta_b \cup \ldots$ where a is an invariant atom in δ, b is an invariant atom in $\delta \cup \delta_a$, c is an invariant atom in $\delta \cup \delta_a \cup \delta_b$ and so forth. By requiring a, b,... to be disjoint, we maintain the context partial ordering.

4 Definitions

Before proceeding to mathematical exploration, we need to add enough expressive power to the original context δ. To do this, we introduce minimal definitions using the mechanism explained in the previous section.

1. Define "bits", "bytes" and "byte sequences" so that words are single atoms, so that a can be a word in future definitions δ_a.
2. Define all of the naturally occurring combinatoric operations given two finite sequences A and B. For example δ_{ap} maps (ap A : $\text{b}_1\ \text{b}_2\ \ldots\ \text{b}_n$) to ($A$:$\text{b}_1$) ($A$:$\text{b}_2$)... ($A$:$\text{b}_n$), so that ap "applies A to each atom of input."
3. Define a minimal internal language $\delta_{\{\}}$ so that a coda ({`language expression`} A : B) is mapped to data as described by the expression and possibly using data A and data B.
4. Include definitions giving direct access to context-level values. This include a Boolean definition δ_{bool} to determine if data is empty, $\delta_{=}$ to test if two data are equal in δ, and δ_{def} to add definitions via Equation 1.

These are meant to be a minimal collection of definitions, including only the minimal inevitable finite sequence concepts and, more generally, the combinatorics of Equation X. There are intentionally no definitions which presume underlying arithmetical operations. A natural number, '123', for instance is a single atom byte sequence and there is no definition which implies interpreting this as a number in the usual way. The idea is to allow natural numbers and whatever else to emerge "organically." There are approximately fifty definitions needed to create a basic system system. We define some here for orientation. More details can be found in the Glossary.

4.1 Bits, Bytes and Byte sequences

If 'a' is an invariant atom, a new atom with domain (a A:B) is defined by $\delta_a : (a\ A : B) \mapsto (a\ A : B)$, so that δ_a is a fixed point on its domain. We may refer to these as "a-atoms." For readability convenience, we define (:)-atoms, ((:):)-atoms and ((:):(:))-atoms to hold bits, bytes and words respectively so that text strings are invariant atoms within the context $\delta \cup \delta_{(:)} \cup \delta_{((:):)} \cup \delta_{((:):(:))}$.

4.2 Combinatorics

Given that text strings are invariant atoms, we can add definitions with readable names. Define the identity 'pass' so that (pass : X=X) for all data X.

- δ_{pass} : (pass A:B) $\mapsto$ B and define ‘null’ so that (null : X=()) for all X.

- δ_{null} : (null A:B) $\mapsto$ ()

It is convenient to define “getting the A and B components of a coda” like so

- δ_{left} : (left A:B) $\mapsto$ A
- δ_{right} : (right A:B) $\mapsto$ B

We proceed to define the minimal combinatorics involving of A and B as finite sequences. These don’t have commonly understood names, so we are forced to invent new names. For example the name ‘ap’ is meant to suggest the concept “apply A to each atom of B” is expressed by the definition.

- δ_{ap} : (ap A : b B) $\mapsto$ (A:b) (ap A : B)
- δ_{ap} : (ap A : ()) $\mapsto$ ()

The domain of δ_{ap} is defined with the assumption that b is an atom, so the domain of the first branch of δ_{ap} is exactly codas (ap A, B) where B starts with an atom. By definition, if we list multiple “branches” as in this case, the first branch in the domain applies.

4.3 Language

Axiomatic systems like ZFC[ref] or dependent type theories[ref] are formal languages. There is an alphabet, special symbols, and syntax rules for defining valid expressions. Our approach avoids this by defining a language as just one more definition like any other. The basic idea is to define minimalist language, mainly giving access to the two foundational operations via text blank space (for concatentation) and text colon (for pairing to data as a coda). So, the idea is

- $\delta_{\{\}}$: ({x y} A : B) $\mapsto$ ({x} A:B) ({y} A:B)
- $\delta_{\{\}}$: ({x:y} A : B) $\mapsto$ ({x} A:B) : ({y} A:B)

where x and y are byte sequences as defined above. As written, this is ambiguous since x and y may contain space and colon characters, but these ambiguities can be resolved just by choosing and order, thus forcing the language into one definition $\delta_{\{\}}$. This definition includes

- $\delta_{\{\}}$: ({A} A : B) $\mapsto$ A
- $\delta_{\{\}}$: ({B} A : B) $\mapsto$ B

so that A and B have special meaning in the language. As a result, for example, ({B}:1 2 3)=1 2 3, ({B B}:1 2 3)=1 2 3 1 2 3, and ({A B} a b : 1 2)=a b 1 2.

Given a language expression in the form of byte sequence data L, the corresponding data is, simply, (L:). Every sequence of bytes is a valid language expression, so there is actually no need to define or check for syntax errors. Similarly, an *evaluation* of (L:) is merely some sequence (L:)=D_1=D_2=...=D_n producing an “answer” D_n. Typically, this is done with a simple strategy based on applying δ whenever possible or up to time or space limits. This also means that there is also no such thing as a “run time error.”

4.4 System

The most basic question to ask about data is whether it is empty or not. This is captured by the definition δ_{bool}, defined so that (bool:B) is () if B is empty ("true") and (bool:B)=(:) if B is atomic ("false").

- δ_{bool} : (bool A : B) $\mapsto$ () if B is empty, (:) if B is atomic.
- δ_{not} : (not A : B) $\mapsto$ (:) if B is empty, () if B is atomic.

The equality defined by the context δ is available within δ via the following definition:

- $\delta_{=}$: (= A : ()) $\mapsto$ A
- $\delta_{=}$: (= (): A) $\mapsto$ A

and if a and b are atoms,

- $\delta_{=}$: (= a A : b B) $\mapsto$ (= a:b) (=A:B)
- $\delta_{=}$: (= A a : B b) $\mapsto$ (=A:B) (= a:b)

so, if $(A : B)$ is empty, A and B are "always" equal and if $(A : B)$ is atomic, A and B are "never" equal. The full language has a little bit of syntactic sugar, so one can write (A=B) instead of (= A:B) [ref].

New definitions can also be added to context via

- δ_{def} : (def `name` A : B $\mapsto$ ())

which is defined to be in domain as a partial function if `name` is not already in the current context, and, in that case, it adds the following definition to context.

- $\delta_{\texttt{name}}$:(`name` A':B') $\mapsto$ (B A':B')

so, for instance (def first2 : first 2 : B), means that (first2 : a b c d) is interpreted as ({first 2:B} : a b c d) which is equal to the data (a b).

We take this as the "organic" base of naturally occurring definitions plus the language will be a starting point in searching for the "spaces" defined in section 5. Further definitions used in the text can be found in the Glossary. Examples, tutorials, a complete definition of the language, and software can be found in reference [x].

5 Global Structure

Pure data has a global structure in the sense that the foundational operations $(A\ B)$ and $(A : B)$ are defined for any data A, B. This suggests defining a corresponding associative product $(A \cdot B)$ and associative sum $(A + B)$ by

$$(A \cdot B) : X = A : B : X \tag{2}$$

$$(A + B) : X = (A : X)\ (B : X) \tag{3}$$

for all data X. This product distributes over the sum, from the right only, so we have

$$(A + B) \cdot C = (A \cdot C) + (B \cdot C) \tag{4}$$

for any data A, B, and C. Since $(\text{pass} \cdot A)$=$(A \cdot \text{pass})$=A and $(A + \text{null})$=$(\text{null} + A)$=A for any A, (pass) and (null) are the units of data composition and data addition respectively. Following standard terminology, we say that A is *idempotent* if $A \cdot A$=A, is an *involution* if $A \cdot A$=1, and A has an *inverse* if $A \cdot A'$=$A' \cdot A$=1 for some data A'. In the case of a *product* A=$A_n \cdot \cdots \cdot A_1$, we say that A *starts with* A_1 and *ends with* A_n.

Data A is called *algebraic* or *commutative* if $A : X\ Y$=$A : Y\ X$ for all X, Y. Data A is *distributive* if $A : X\ Y$=$(A : X)\ (A : Y)$ for all X and for all Y. Data A defines a binary operator on data via $(X \overset{A}{*} Y) = (A : X\ Y)$. Since $\overset{A}{*}$ is associative if and only if $(A{:}X\ Y)$=$(A{:}(A{:}X)\ Y)$=$(A{:}X\ (A{:}Y))$ for all X,Y we say that data A is *associative* if it has this property.

Algebraic and distributive data foreshadow a theme where these two properties make data mathematically interesting for complementary reasons. Algebraic data has, in a sense, "transcended finite sequence," the foundational concept of the system and has a platonic existence in that sense. Distributive data, on the other hand, is maximally sequence dependent, but it may also be mathematically interesting since such data is a "morphism of finite sequence."

6 Spaces

A basic requirement for useful mathematics is to have a way to define and refer to collections. Within the framework of pure data, however, there is only one "substance" to work with. All mathematical objects are merely data and all collections of mathematical objects are also merely data. Thus, we are lead to ask, given some data S, how could S represent a collection of other data? There are a couple of plausible ways to do this.

- The collection corresponding to S would be the collection of data $(S{:}X)$ for any data X;
- The collection corresponding to S would be the collection of all of the fixed points of S.

These ideas coincide if we require S to be idempotent, so that $(S{:}X)$ is automatically a fixed point. It is natural to also expect compatibility with sequences in the sense that if $(S{:}X)$ is in the collection and $(S{:}Y)$ is in the collection, then $(S{:}X)\ (S{:}Y)$ is also in the collection. This is guaranteed if we require

$$(S : X\ Y) = S : (S : X)\ (S : Y) \tag{5}$$

for all data X, Y. Note that data S is idempotent and satisfies Equation 4 if and only if S is associative, and so we are lead to a simple definition.

Definition 3. Data S is a **space** if S is associative.

If S is a space, we say that any data $(S{:}X)$ is *in* S. The data $(S{:})$ is called the *neutral data* of S.

Examples of spaces can be found in the basic definitions, including 'pass', 'null' and 'bool'. More examples come from noting that any data which is both idempotent and distributive is a space. Thus, if J is idempotent, then (ap J) is idempotent and distributive, and is, therefore, a space. If spaces S and T commute, then $S \cdot T$ is a space.

6.1 Morphisms

A product F that starts with space S and ends with space T is a *morphism* from S to T and can be written $S \xrightarrow{F} T$. Since spaces are idempotent, F=$(F \cdot S)$=$(T \cdot F)$. Since $T \cdot S$ is a morphism from S to T, morphisms always exist. Since the product is associative, morphisms can be composed, so that if $S \xrightarrow{F} T$ and $T \xrightarrow{G} U$, then $S \xrightarrow{G \cdot F} U$. If F and G are morphisms from S to T, we can also define a *sum of morphisms* by $F \oplus G = T \cdot (F + G)$, so that $S \xrightarrow{F \oplus G} T$. A morphism from a space to itself is called an *endomorphism*.

Since $T \cdot X \cdot S$ is a morphism from S to T for any data X, a morphism can define any function. Special morphisms which "preserve the structure" of S and T can be defined as follows. A morphism $S \xrightarrow{H} T$ is a *homomorphism* if

$$H \cdot (f \oplus_S g) = (H \cdot f) \oplus_T (H \cdot g) \tag{6}$$

for all endomorphisms f and g of S. Compare with morphisms that happen to also be a space. Morphism $S \xrightarrow{U} T$ is a space if and only if

$$U \cdot (f \oplus_S g) = U \cdot ((U \cdot f) \oplus_T (U \cdot g)) \tag{7}$$

for all endomorphisms f and g of S. Thus, any idempotent homomorphism is a space. Note that homomorphisms are closed under composition, but the composition of two spaces is a space only if the two spaces commute.

6.2 Spaces are Semirings

Consider the endomorphisms of a fixed space S. Since S itself is a product starting at S and ending at S, S qualifies as an endomorphism of itself. Since $(S \cdot f) = (f \cdot S) = f$ for any endomorphism f, S is "its own identity morphism." The endomorphisms of S are closed under both composition and addition, with addition defined by $f \oplus g = S \cdot (f + g) \cdot S = S \cdot (f + g)$. Thus, the endomorphisms of S are a 'semiring' with composition and addition and with $1 = (S \cdot \text{pass} \cdot S) = S$ and with $0 = (S \cdot \text{null} \cdot S) = (S \cdot \text{null})$.

Definition 4. A **semiring** is a set with associative addition and multiplication (denoted $f \oplus g$ and $f \cdot g$), with additive and multiplicative identities $\mathbf{0}$ and $\mathbf{1}$ satisfying $f \oplus 0 = f = 0 \oplus f$ and $f \cdot 1 = f = 1 \cdot f$ for all f in the semiring, and where $(f \oplus g) \cdot h = (f \cdot h) \oplus (g \cdot h)$ for all f, g, and h in the semiring. A subset of a semiring S which contains 0 and 1 and is closed under addition and multiplication is a **subsemiring** of S.

Note that the semiring of space S has 1=0 if and only if S is a constant space. These spaces are (const K) for any chosen data K including (const ())=null. It is easy to verify that the semiring of the space (pass) is exactly the the global algebra defined in Section X. Since isomorphic spaces have isomorphic semirings, understanding the semiring of a space is mainly what we aim for in the examples which follow. Before going to examples, however, it is helpful to identify semiring features of interest.

Consider the semiring of a fixed space S.

- **Subspaces.** If an endomorphism s of S happens to also be a space, we say that s is a *subspace* of S. The name "subspace" is justified because every data contained in s is also contained in S, and every endomorphism $s{\cdot}\mathrm{X}{\cdot}s$ of s is also an endomorphism of S. Subspaces partially distribute from the left as with $s \cdot (f \oplus g) = s \cdot ((s \cdot f) \oplus (s \cdot g))$. Every space has subspaces 1 (the whole space), and 0, the constant neutral space. If subspace s is not equal to S, s is a *proper* subspace.

- **Constants.** An endomorphism k of S satisfying $k \cdot 0 = k$ is a **constant** subspace of S. There is one such constant for each data in S. Since $k \cdot f$ and $k \oplus l$ are constants for constant k and l and for any endomorphism f, the constants of S are a left-ideal of the semiring of S.

- **Homomorphisms.** By Equation 5, the homomorphisms of S are exactly the endomorphisms which distribute over addition from both the left and from the right. The endomorphisms 0 and 1 are homomorphisms. If f and g are homomorphisms of S, then $f \cdot g$ is a homomorphism. If S is algebraic, the homomorphisms are a subsemiring of the semiring of S.

- **Group of units**. The collection of endomorphisms with multiplicative inverses is called the *group of units* of the space S.

- **Central endomorphisms**. Endomorphisms which commute with the group of units are *central endomorphisms*. The endomorphisms 0 and 1 are central. If f and g are central endomorphisms, then $f \cdot g$ is a central endomorphism. If f and g are *central homomorphisms*, then $f \oplus g$ is also a central homomorphism. Thus, the central homomorphisms of S are a subsemiring of the semiring of S.

- **Neutral spaces and sets**. If addition in S allows left and right cancellation ($f \oplus g = f \oplus h$ implies $g = h$ and $g \oplus f = h \oplus f$ implies $g = h$), S is a *neutral* space. The neutrality of S guarantees $h \cdot 0{=}0$ for homomorphism h, and guarantees that if $f \oplus f{=}f$, then $f = 0$. At the other extreme, if $f \oplus f{=}f$ is true for all endomorphisms and if S is algebraic, then S is a *set*.

- **Positive spaces**. If endomorphisms cannot cancel in the sense that $f \oplus g{=}0$ implies $f{=}g{=}0$, S is a *positive* space. If non-zero endomorphisms f and g imply that $f \cdot g$ is also non-zero, then S is a *non-zero preserving* space.

- **Idempotents**. Define a relation $\overset{i}{\leq}$ on the endomorphisms of S by $f \overset{i}{\leq} g$ if $(f \cdot g){=}f$. The equivalence classes of the corresponding symmetric relation $f \overset{i}{\sim} g$ contain all the idempotent endomorphisms, since $f \overset{i}{\sim} g$ implies that both f and g are idempotent. In this *idempotent order*, 1 is the maximum endomorphism and the constant endomorphisms are minimal.

In the case where S is distributive, $(f \oplus g) = (f + g)$ for f, g endomorphisms of S. This means that the homomorphisms of S are the distributive endomorphism. If the group of units of S includes permutations, then the central homomorphisms are algebraic. This illustrates the generalization of the earlier remark about algebraic and distributive data being mathematically interesting for complementary reasons. Central endomorphisms are the generalization of algebraic data and homomorphisms are the generalization of distributive data. Naturally, one suspects that the central homomorphisms of a space will be particularly interesting.

6.3 Isomorphic Spaces have Isomorphic Semirings

The fact that homomorphisms compose suggests adapting definitions from Category Theory. We say, for instance, that spaces S and T are *isomorphic* if the diagram of Figure X commutes for homomorphisms h and h'. where we have used S and T instead of respective "identity morphisms." Given an isomorphism as in Figure X, the bijection $f \mapsto h' \cdot f \cdot h$ is a semiring isomorphism, preserving product, sum, 1 and 0, so that isomorphic spaces have the same features defined in the previous section, since these are all defined with semiring operations. A homomorphism h where $h \cdot \alpha = h \cdot \beta$ implies $\alpha = \beta$ is a *monomorphism* and and if $\alpha \cdot h = \beta \cdot h$ implies $\alpha = \beta$, h is an *epimorphism*.

6.4 Spaces can be Semialgebras

The semiring of a space S is close to the structure of an "algebra" in standard mathematics. There is a binary operation $f \oplus g$, so constants of the space can be added. Homomorphisms have

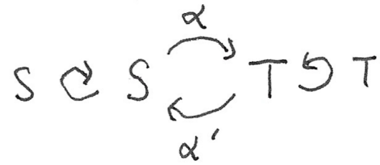

Figure 1: *In a definition adopted from Category Theory, spaces S and T are isomorphic if the diagram commutes for some homomorphisms* α *and* α'*.*

the property that they map constants to constants and are distributive with $h \cdot (k \oplus l)$ equal to $(h \cdot k) \oplus (h \cdot l)$ so homomorphisms of S are like a module over the constants of S. This would be an algebra in the ordinary sense except that homomorphisms might not be constants. This suggests a definition

Definition 5. A space S is a **semialgebra** if the homomorphisms of S are bijectively equivalent to the constants of S. A space S is a **central semialgebra** if the central homomorphisms of S are bijectively equivalent to the constants of S.

Since pure data is meant to include all mathematical objects, we might expect that important algebras of standard mathematics could appear semialgebras. Whether this is true or not will be part of our investigation.

6.5 Subspaces are Substructures

Suppose that space S has subspace s and space T has subspace t, and suppose that $F{=}t \cdot X \cdot s$ is a morphism from s to t. Since $(t \cdot X \cdot s) = (T \cdot t \cdot X \cdot s \cdot S)$, F is also a morphism from S to T, with the extra property that F respects the structure of s and t in the sense that $t \cdot F = F \cdot s$. If a space has multiple subspaces, each commuting pair of subspaces is a new subspace with compatible structure.

If s is a subspace of S, it is easy to verify that if h is a homomorphism of S and h is also an endomorphism of s, then h is a homomorphism of s. Similarly, if u is a unit of S and u is an endomorphism of s, then u is also a unit of s. Subspaces, however, can have homomorphisms and units which are not inherited in this way.

For any morphism $S \xrightarrow{h} T$, there is a guaranteed factorization

$$h = m \cdot e \tag{8}$$

where m is a monomorphism and e is an idempotent of S. If h is a homomorphism, we can substitute this into the definition of a homomorphism, use $h \cdot e = h$, and, cancelling m on the left, find

$$e \cdot (f \oplus g) = e \cdot (e \cdot f \oplus e \cdot g), \tag{9}$$

and conclude that e is a subspace of S, which we call a *quotient* of S, and denote e by S/h. Note that $S/h = S$ if and only if h is a unit as well as a homomorphism.

6.6 Fields

Since spaces have subspaces, which may themselves have subspaces, a descending sequence of subspaces can only terminate with a constant space, since constant spaces are the only spaces with no proper subspaces. From the semiring perspective, constant spaces have no variety. All constants are isomorphic and they all have 0=1. On the other hand, there is a wide and interesting variety of spaces one step above constants. These are the 'fields'.

Definition 6. A space where all proper subspaces are constants is a **field**.

In classical ring theory, a field is guaranteed to have multiplicative inverses. An analogous result nicely combines homomorphisms, constants, units and subspaces.

Theorem 6.1. A space S is a field if and only if every non-constant homomorphism of S is a unit.

Proof. If S is a field, every non-constant homomorphism h has $S/h{=}S$ and, thus, is a unit. Conversely, suppose that S is not a field. Then S contains a proper non-constant subspace s and $\sigma = S \cdot \text{ap } s \cdot S$ is, therefore, a non-constant homomorphism of S. Since $\sigma : X$ is in s for any data X, σ is not a unit. ☐

Corollary 6.2. Every field is a semialgebra.

Proof. ap const K trick... ☐

The obvious examples of fields are null, bool and any constant space. In the examples, we will see familiar fields such as the natural number modulo a prime or the field of rational numbers. In these examples, zero will be the only constant homomorphism and since semialgebra multiplication is application by a homomorphism, the guarantee of an inverse for non-zero elements comes from a guarantee that non-constant homomorphisms are units.

6.7 Themes

Although the motivation for the concept of a space is merely to define a general collection of data, and a space is merely any associative data, we find that spaces and their endomorphisms have a rich internal structure including subspaces, a group of "symmetries," a class of endomorphisms which qualify as homomorphisms, and central endomorphisms which commute with the symmetries. This is a common structure shared by all collections ranging from a space containing only empty data (null) to a space containing one atom (const (:)) to a space containing all mathematical objects (pass). The scope of these results means that pure data spaces can be thought of as a general point of view about mathematical objects, analogous to sets with structure or Category Theory. Before exploring simple examples, lets highlight the differences.

- A set is defined by its contents, but a space is not. Spaces (ap {a}) and (is a), for instance, contain the same data, are idempotence equivalent (ap {a})$\overset{i}{\sim}$(is a), are isomorphic as spaces (and therefore have the same semirings), but they are not the same space. For example, ap {a}:(:) = a, but (is a):(:) = (). Since every space contains its neutral data at least, it is not possible for a space to be empty.

- Spaces have morphisms between them which compose associatively, but, unlike Category Theory, morphisms and spaces are "made of the same substance." They are both merely data. For instance, every data A of pass is also a morphism pass$\cdot A\cdot$pass from pass to pass. Unlike Category Theory, a morphism $S \xrightarrow{T \cdot S} T$ exists between any two spaces S and T, so there are no closed categories where morphisms do not cross category boundaries. Unlike Category Theory, a morphism can be morphisms between multiple spaces at once. For instance, if S and T are commuting spaces, any fixed morphism $(S \cdot T) \xrightarrow{F} U$ means that F is a morphism from $S \cdot T$ to U, from S to U and is also a morphism from T to U.

- Category theory is a theory of structures with corresponding morphisms. In the theory of spaces, each space comes with both structure preserving "homomorphisms" and not-necessarily-structure-preserving morphisms, packaged together in a coherent semiring.

- In classical mathematics, one expects sets to have subsets, groups to have subgroups, rings to have ideals, and, in general, structures to have similar substructures. Thus, it is not surprising that spaces can have subspaces, but unlike the classical situation, subspaces of a space can have completely different mathematical structures from the ambient space. For example, we shall see that the space (is a b c d) has various free monoids, natural numbers, integers, the algebra of 4x4 natural number matrices and Gaussian integers as different subspaces. The extreme version of this is the space pass which contains *every* space as a subspace. One can think of an algebraic space S as having "horizontal" features coming from the subsemiring of homomorphisms and "vertical" features coming from general endomorphisms and subspaces. Classical fields are limited in the sense that this is a theory of Commutative Rings. Interestingly, fields as defined in the previous section appears to be a generalization which appears wherever spaces appears.

Since this is a rich but unfamiliar viewpoint, we proceed with the simplest, most "organic" spaces first, starting with foundational definitions which happen to also be spaces. The pure data view of spaces gives us a criteria for what is "most organic" and gives a way to systematically search for all spaces up to a specified data width and depth. Although we mainly use the pure data perspective to choose spaces of interest, what we say about spaces will be put in semiring language whenever possible. Aside from the organic and semiring themes, we will take special notice of algebraic data following the intuition that algebraic spaces are most likely to be mathematically interesting since they have, in a sense, earned a platonic meaning by transcending the foundational finite sequence structure of the system. In semiring terms, we will thus focus on the central homomorphisms of a space as the most likely features of interest.

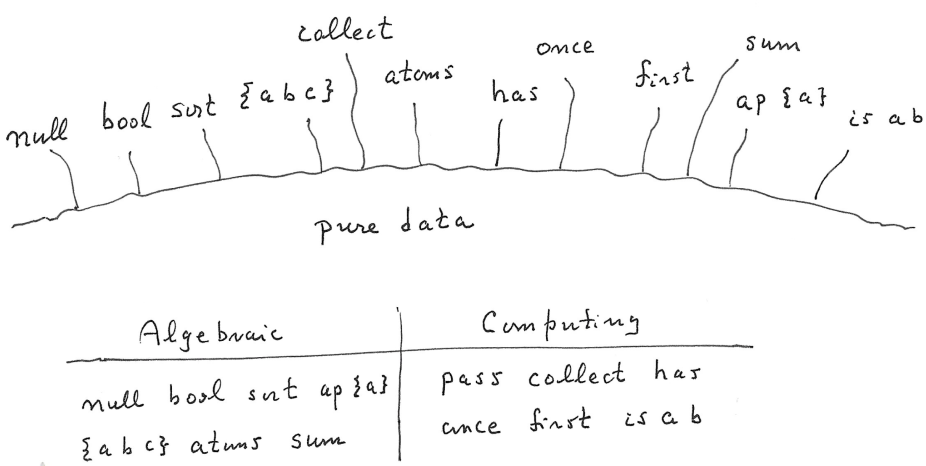

Figure 2: *The space of all pure data contains all mathematical objects. Our strategy for exploring this space is to first introduce the minimal combinatorics implicit from the foundational finite sequence concept, and then examine the simplest spaces first as they appear "organically." Since algebraic spaces are commutative, they, in a sense, transcend the foundational sequence concept and are more "Platonic."*

7 The Space of all data

As a first example and for orientation, let's consider the space of all pure data. Since (pass:X)=X for all data X, the distributive space pass contains all pure data. Since spaces are fixed points on their contents, pass is the unique space containing all data. Every data X is both in pass and is also an endomorphism of pass, since X is equal to pass$\cdot X\cdot$pass. Thus, the semiring of pass is also the collection of all data, with addition $(X \oplus Y)$=$(X+Y)$, and multiplication $(X \cdot Y)$. The units of the semiring are 1=pass·pass·pass=pass, and 0=pass·null·pass=null respectively.

Consider the properties defined in section 6.

- **Subspaces.** Since every data is a morphism of pass, every space is a subspace of pass. The constants of pass are the data K such that $K \cdot 0$=K, in other words, the constant spaces (const K) are the constants.

- **Homomorphisms.** Since pass is distributive, the homomorphisms of pass are the distributive data in pass, including spaces, such as (ap {a}) and non-spaces such as (ap {B B}).

- **Group of units.** The units of pass are permutations such as (swap 1 2) and (rev). The central constants of pass are the data which are invariant under all permutations. These are the sequences of identical atoms such as (:) (:) (:) or (a a a a a). These can be interpreted as "organic natural numbers."

- **Central endomorphisms** The maximally platonic, maximally significant content of pass are the central homomorphisms, which include subspaces such as (ap const a), (atoms), (is b). These are isomorphic spaces, each representing natural numbers with natural number addition. This and the direct connection of bool with the underlying pure data context means that both the natural numbers and bool are good starting points for exploration.

- **Semialgebra** In pass, there is a bijection between the homomorphisms of pass and the constants of pass given by $X \leftrightarrow$ (ap X). Thus, pass is a semialgebra as well as a semiring with $X \star (Y+Z)$ equal to $(X \star Y) + (X \star Z)$ where $X \star Y$ is defined to be (ap X)$\cdot Y$.

- **Semilattice** The ideal semilattice includes the spaces (bool), (once) and (const K) for any data K. Although not a homomorphism, bool is a central endomorphism as well as a semilattice and is the unique space with exactly two fixed points.

- **Idempotents** The idempotent equivalence classes of pass includes all idempotent data where (pass) is the unique maximum idempotent and where all constants are in the minimal equivalence class.

In the idempotence order, pass is the unique maximum $X \overset{i}{\leq}$pass since pass uniquely satisfies $X \cdot \text{pass} = X$ for all X. Pass is its own image and the kernel of pass is the maximal idempotent e such that $e \cdot \text{pass} = \text{null} \cdot \text{pass}$, so the space (null) is the kernel of pass.

In classical mathematics, the concept of a Set intentionally gives no information about its contents. Thus, the "Set of all Sets" is vast and, in a sense, contains everything, but we don't expect any insight from this. The space (pass) is also vast. It contains all mathematical objects of any kind and all possible computations within the framework of pure data. Nevertheless, pass does have a specific structure with specific features that indicate what is most important mathematically: the central homomorphisms (the natural numbers) and the central semilattice (bool). After describing (pass) and (null), these will be our starting points for investigation.

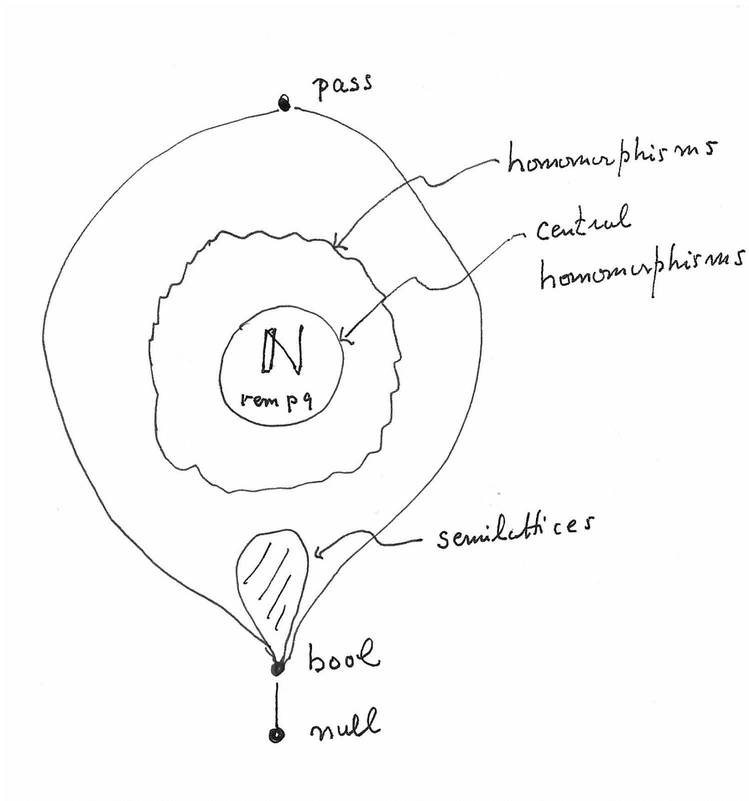

Figure 3: *As a collection, pass contains all pure data; that is, all objects with mathematical meaning as well as all computations. As a space, it has a specific structure which is typical of other examples in the text. In the idempotent partial order, (pass) is the unique maximum and (null) is minimal. Each data in pass is also an endomorphism of pass. The homomorphisms of pass are the distributive data. The central homomorphisms are a single isomorphism class equivalent to the "organic" natural numbers. Spaces that are fields includes bool, null, and the rem(p,p) subspaces of* $\mathbf{N}$*, equal to* $\mathbf{F}_p$ *for prime p. Bool and null are in the ideal of semilattice spaces as well.*

8 Null

Null is the distributive algebraic space defined by (null:X)=() for all X. Since null$\cdot X\cdot$null=null, null is the only endomorphism of null, and so 1=0 in the semiring, as is true for all other constant spaces. Since $0 \cdot (0 \oplus 0) = (0 \cdot 0) \oplus (0 \cdot 0) = 0$, the one endomorphism is a homomorphism as well as a subspace. Null is minimal in the idempotence order, as is all of the other constant spaces. The kernel of null is pass.

9 Organic Numbers

We have seen that "organic natural numbers" appear as sequences of identical atoms where we identify (:) (:) (:) or (a a a) with "3," just depending on a conventional choice of atom to represent "1." The spaces where these numbers are data are the maximally platonic central homomorphisms of pass. For instance, the spaces (atoms), (ap const (:)), (ap const a), (ap {#}), (is a) all are central homomorphisms and each represents natural numbers with natural number addition. Since these spaces are isomorphic and, therefore, have the same semirings, we can choose any one for investigation. For convenience we will choose (is a) as the particular space to start with. For convenience, denote n concatenations of data A by A^n so so that “5” represented by (a a a a a) and can be written as a^5. The space (is a) has natural extensions (is a b), (is a b c),.... Lets agree to refer to these spaces and their subspaces as the “organic numbers.”

9.1 Organic Natural Numbers

The first thing to notice is that N contains one data a^n for each natural number n, and addition in N is natural number addition, since (N:X Y) is the natural number sum of (N:X) and (N:Y). Since homomorphisms are distributive, a homomorphism h of N is determined by (h:a), so that h is natural number multiplication by some natural number, for instance, (h = ap const a a a) is multiplication by 3. There is one such homomorphism for each data in N, and so N is a semialgebra and is identical to the standard semiring of natural numbers with multiplication distributing over addition.

Subspaces of N are indicated by subspace endomorphisms. As always, the subspace 1 is the whole of N and the subspace 0 consists of just the neutral data of N, which, since N is distributive, is the empty sequence. It is easy to guess some subspaces of N.

1. **Saturation subspaces** An endomorphism (min a a) is a subspace which effectively computes the minimum of 2 and its argument. Thus (min a a) contains the data (), (a), and (a a).
2. **Modular arithmetic subspaces** An endomorphism which removes three (a) atoms while possible, such as (while remove a a a) is a subspace which computes the sum modulo 3. Note that (while remove a a a) contains the same data as (min a a), but they have a completely different algebraic structure.

These unsurprising subspaces have a slightly more surprising generalization. For $p, q \geq 1$, let rem(p,q) be the endomorphism defined by

$$\text{while } n \geq p, q : \text{remove p from n} \tag{10}$$

The rem(p,q) are also subspaces and modular sum and saturation cases since rem(p,p) is $n \mapsto n$ mod p and rem(1,q) is $n \mapsto \min(n, q-1)$. The rem subspaces are also closed under composition via

$$\text{rem}(p, q) \cdot \text{rem}(p', q') = \text{rem}(\text{LCM}(p, p'), \max(q, q')) \tag{11}$$

where LCM is the least common multiple. This makes the rem a closed commutative semilattice of subspaces of N.

This completes the analysis of N. The homomorphisms of N are multiplication by natural numbers, making N a semialgebra identical to the standard semiring $\mathbb{N}$. The only subspaces of N are 1, the constants, and the rem(p,q) subspaces. All subspaces are mutually commuting and, thus, are closed under composition.

9.2 is a b

The second simplest “organic number” is (is a b), which we can refer to as N_2 for brevity. Since N_2 is distributive, the homomorphisms of N_2 are the distributive endomorphisms. Consider subspaces N_2.

- The subspaces 1 and 0 give the whole of N_2 and just the empty sequence as subspaces, respectively. Projections N_2·(is a)·N_2 and N_2·(is b)·N_2 produce two N_1-isomorphic subspaces.

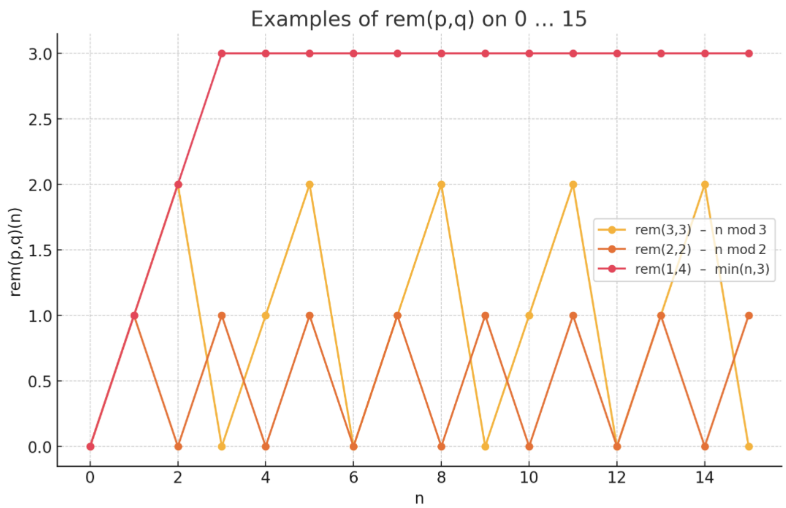

Figure 4: *Subspaces of (is a) or N are N itself, constants and rem(p,q) subspaces. These specialize to natural numbers mod p and ceiling functions min(q).*

- Let 'reduce' be the endomorphism of N_2 which removes (a b) and (b a) subsequences until saturation. Thus, every data in reduce is either a^n or b^n for some natural number n. A homomorphism h satisfies $h : X\ Y =$ reduce : $(h : X)\ (h : Y)$, so h is determined by (h:a) and (h:b). If h is a central homomorphism, (h:b) is (h:a) followed by an a$\leftrightarrow$b swap. Thus, if we interpret a^n as the integer n and b^n as the integer $-n$, then applying h is standard distributive integer multiplication by the integer (h:a). Since there is one central homomorphism for each data in the subspace reduce, reduce is a central semialgebra isomorphic to the standard ring $\mathbf{Z}$ of integers.

- If 'sort' is the endomorphism which does lexical sorting of N_2 data, then sort is also a subspace of N_2 where the data of sort can be written $a^m b^n$ for $m, n \geq 0$. As in the previous case, a homomorphism M of sort is defined by

  $M : a \mapsto a^{m_{11}} b^{m_{21}}$, and
  $M : b \mapsto a^{m_{12}} b^{m_{22}}$

  for some choice of $\begin{pmatrix} m_{11} & m_{21} \\ m_{12} & m_{22} \end{pmatrix}$ so the action of M is standard matrix multiplication. Thus, the data of sort are pairs of natural numbers and the homomorphisms are 2x2 matrices with natural number entries. The central homomorphisms of sort commute with the involution $\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ and so the central homomorphisms are matrices $\begin{pmatrix} m & n \\ n & m \end{pmatrix}$ for natural numbers m and n. Thus, there is exactly one central homomorphism for each data in sort, and sort is, therefore, a central semialgebra, equivalent to the standard matrix semiring $\mathrm{Mat}_{2\times2}(\mathbf{N})$.

- Lexical sort followed by reducing (b b) to (b) results in a subspace equivalent to $\mathbf{N} \times \{0, 1\}$ with addition defined by $(n, \alpha) + (m, \beta) = (n + m, \alpha \vee \beta)$.

- Lexical sort followed by reducing $a^n b^m$ to $a^{(n/g)} b^{(m/g)}$ where g=GCD(n, m) is the greatest common divisor. This is the space of positive rational numbers, but with non-standard 'mediant' addition[ref].

9.3 is a b c d...

It is clear that in going to higher organic numbers, some of the features of N_1 and N_2 will repeat. N_n sorted will always result in a space with homomorphisms $\mathrm{Mat}_{n\times n}(\mathbf{N})$. Sorting followed by cancelling atoms in pairs will result in $\mathrm{Mat}_{(n/2)\times(n/2)}(\mathbf{Z})$. For example, in (is a b c d) with subspace (a b)=(b a)=(c d)=(d c)=(), homomorphisms are $\mathrm{Mat}_{2\times 2}(\mathbf{Z})$. Central homomorphisms must commute with $J = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$ and are thus matrices of the form $\begin{pmatrix} x & -y \\ y & x \end{pmatrix}$ and we have the central semialgebra equivalent of the Gaussian Integers $\mathbf{Z}[i]$. Figure 5 shows similar results for some of the more notable Organic Number subspaces. Notice that this is a success from the organic gardening perspective. Important mathematical structures are emerging quite naturally. Note that even though both semiring operations are associative, a non-associative algebra such as the integer Octonions appears in the same way as the other spaces with homomorphisms implementing the non-associative octonion multiplication.

Organic Numbers

Space	Subspace	Data Elements	Central Homomorphisms	Central Semialgebra
is a	—	$\mathbb{N}$	Multiply-by-k	$\mathbb{N}$
is a b	$ab = ba = ()$	$\mathbb{Z}$	Multiply-by-k	$\mathbb{Z}$
is a b	sort	$\mathbb{N}^2$	Symmetric 2×2 matrices	$\mathrm{Mat}_{2\times 2}^{sym}(\mathbb{N})$
is a b	sort, $bb = ()$	$\mathbb{N} \times \mathbb{Z}_2$	Block-diagonal maps	$\mathbb{N} \times \mathbb{Z}_2$
is a b	sort, gcd-reduced	Coprime (m, n)	Identity/constants only	None
is a b c d	$ab = ba = (), cd = dc = ()$	$\mathbb{Z}^2$	Complex scalars	$\mathbb{Z}[i]$
is $a_1 \ldots a_8$	*pair reductions*	$\mathbb{Z}^4$	Quaternion scalars	$\mathbb{H}_{\mathbb{Z}}$
is $a_1 \ldots a_{16}$	*pair reductions*	$\mathbb{Z}^8$	Octonion scalars	$\mathbb{O}_{\mathbb{Z}}$
is $a_1 \ldots a_{32}$	*pair reductions*	$\mathbb{Z}^{16}$	Sedenion scalars	$\mathbb{S}_{\mathbb{Z}}$

Figure 5: *Organic Numbers.*

Although many interesting spaces appear as "Organic Numbers", some simple spaces appear to be missing. It's noticeable, for instance, that only non-standard rationals appear. To understand why, notice that any non-constant subspace of an Organic Number space (is a b c...) will have proper non-constant subspaces by projecting onto a subset of a b c... Thus, the only non-trivial fields that appear as Organic Number subspaces are the F_p subspaces of (is a). Thus, the rationals are not Organic Numbers.

10 Rationals

Although the rationals are not organic numbers, they appear quite naturally as sequences of atoms (Q a a a : a a a a) with some fixed chosen atom Q....(Q m:n) (Q m':n) – (Q m m':n)....(Q 1 m:n) = (Q 1:n) (Q m:n)...+homomorphism product =¿ field =¿ semialgebra.

11 Number Sequences

Given the Organic Naturals N=(is a), let ℕ be

$$\text{ap (put n)} \cdot N \cdot \text{(get n)} \tag{12}$$

where 'n' is some chosen atom. Since (put n) $\cdot$ N $\cdot$ (get n) is idempotent, ℕ is a distributive space containing sequences of n-atoms containing N-data, such as

$$\text{T = (n : a a a) (n :) (n : a a) (n : a) (n :)} \tag{13}$$

Unlike the case of N, the action of ℕ is merely to concatenate sequences. It's easy, however, to identify natural number sum again as one of several subspaces

- sum:T = (n:a a a a a a)
- sort:T = (n:) (n:) (n:a) (n:a a) (n:a a a)
- min:T = (n:)
- first:T = (n:a a a)

All but the last are algebraic subspaces and, predictably, have established names. Some endomorphisms of ℕ are "inherited" from N as follows. Define inner:F to be

$$\mathbb{N} \cdot \text{(put n)} \cdot \text{F} \cdot \text{(get n)} \cdot \mathbb{N} \tag{14}$$

so that (inner:F):X means letting F act on the concatenated N-contents of X, returning the result in a single n-atom. The sum endomorphism above, for instance, is equal to inner:N. The construction of Equation 7 works for any space, and so we can define a "functor" *Seq* to be

$$\{\text{(put A)} \cdot \text{B} \cdot \text{(get A)}\} \tag{15}$$

Then ℕ is equal to (Seq n:N) and given any atom s and any space S, (Seq s:S) is the space of S-values stored in s-atoms. Similarly, if we define *inner* to be

$$S \cdot \{\text{(put s)} \cdot \text{B} \cdot \text{(get s)}\} \cdot S \tag{16}$$

then inner:f is the inner version of an endomorphism f of S. Other general constructions of this type are possible. For example, since the endomorphism last2 = (inner:last 2) sums the last two atoms of a data of ℕ, (series last2) generates the fibonacci sequence.

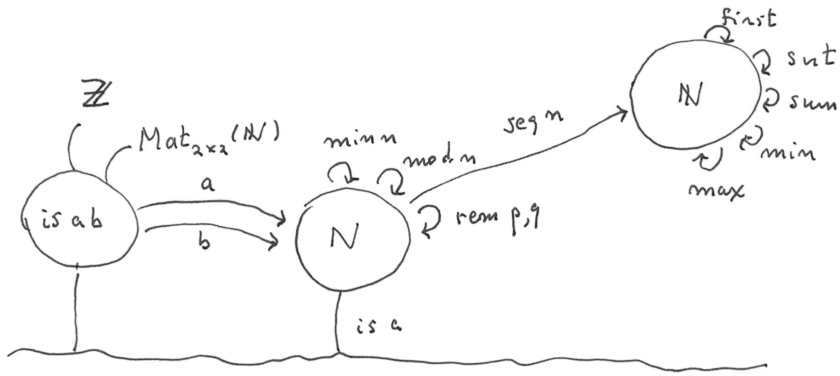

Figure 6: *Organic numbers.*

$f \cdot g$	ID	TRUE	FALSE	NOT
ID	ID	TRUE	FALSE	NOT
TRUE	TRUE	TRUE	TRUE	TRUE
FALSE	FALSE	FALSE	FALSE	FALSE
NOT	NOT	FALSE	TRUE	ID

$f \oplus g$	ID	TRUE	FALSE	NOT
ID	ID	ID	FALSE	FALSE
TRUE	ID	TRUE	FALSE	NOT
FALSE	FALSE	FALSE	FALSE	FALSE
NOT	FALSE	NOT	FALSE	NOT

Table 1: *Product and sum of the four morphisms ID, TRUE, FALSE, NOT of the space bool.* Note that ID is the unit of multiplication and TRUE is the unit of addition, and we have $f \oplus g = g \oplus f$, since bool is algebraic, and $f \oplus f = f$.

12 Boolean

As mentioned already, the space bool is likely to be mathematically interesting, both because of its direct connection to the underlying pure data context and, in abstract terms, because it is a central, algebraic space within pass. Although central, bool is not a homomorphism and so has a different character from natural numbers. Since bool contains two data: () for *true* and (:) for *false*, bool has only four endomorphisms: the identity ID=bool, two constants: TRUE=(const ()), and FALSE=(const (:)), and one involution NOT=bool·not·bool. Since there are only four, we include the complete semiring operations in Table X. Note that $\oplus$ is commutative since bool is algebraic and $f \oplus f = f$, since bool has the semilattice property. The subspaces of bool are ID, TRUE and FALSE, so bool has only trivial subspaces. The endomorphisms ID and TRUE are neutral, ID is the only positive endomorphism and all endomorphisms are algebraic since bool itself is algebraic. The group of units is ID and NOT, and ID is the only central endomorphism.

12.1 Boolean Sequences

As in the case of N, we can let $\mathbb{L}$=(Seq b:bool), be the distributive space of bool-valued data stored in b-atoms, so a typical data in $\mathbb{L}$ is

$$\text{(b :) (b :) (b : (:)) (b :) (b : (:)) (b : (:))} \tag{17}$$

Let's agree to write such sequences replacing (b:) with T, (b:(:)) with F and the empty sequence with 0, so that the above is written TTFTFF. Let's also consider the shortest subspaces of $\mathbb{L}$ first. The subspace $\mathbb{L} \cdot \text{first} \cdot \mathbb{L}$ contains the $\mathbb{L}$ sequences of length less than or equal to 1 and similarly, $\mathbb{L} \cdot \text{first } 2 \cdot \mathbb{L}$ is the subspace of sequences with length less than or equal to 2. Lets refer to these spaces as $\mathbb{L}_1$ and $\mathbb{L}_2$ repectively. Thus, $\mathbb{L}_1$ contains data {0,T,F} and $\mathbb{L}_2$ contains data {0,T,F,TT,TF,FT,FF}. For $\mathbb{L}_1$, we can specify an endomorphism f of $\mathbb{L}_1$ by listing the values of f on 0,T,F in standard order. So, for example, '0TF' denotes the identity endmorphism of $\mathbb{L}_1$. Since $\mathbb{L}_1$ has 27 morphisms, we can be completely explicit about this space. Figure 4 lists the 27 endomorphisms and indicates endomorphisms with special properties. Even with relatively small space like $\mathbb{L}_1$, all of the classes of endomorphisms appear in a nontrivial way.

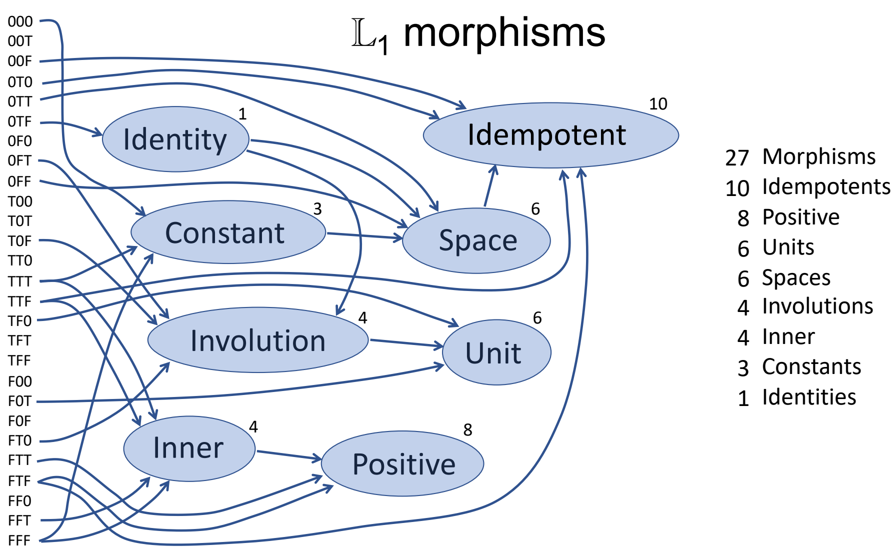

Figure 7: *The endomorphisms of* $\mathbb{L}_1$ *are maps from from {0,T,F} to itself. These are specified as the three function values in the column on the left. Even with a small space of this size, the* $3^3 = 27$ *all of the classes of endomorphism identified in the text appear in a non-trivial way.*

Moving to $\mathbb{L}_2$, there are already $7^7 = 823,543$ morphisms; far too many to be as explicit as in the case of $\mathbb{L}_1$. As a first attempt, examine just the inner endomorphisms, since there are only eight of these consisting of the endomorphisms $\mathbb{L}_2 \cdot (\text{put } b) \cdot \text{F} \cdot (\text{get } b) \cdot \mathbb{L}_2$ for some data F. These eight depend only on the total number of (:) values in the b-atoms of input, and each morphism returns exactly one b-atom. The inner morphisms of $\mathbb{L}_2$ are all central homomorphisms, so we might expect them to be mathematically interesting. Indeed, Table Y shows the eight morphisms and their values on {0,T,F,TT,TF,FT,FF}. Looking at the values for the last four entries {TT,TF,FT,FF} we recognize that these are slight generalizations of the eight standard symmetric binary boolean operators. The values of the eight morphisms on {0,T,F} suggests that the standard binary operator names ('OR' as opposed to 'any') are not particularly illuminating in this larger context.

13 Summary, Global Strategies, Questions

Summary: we have a coherent view of mathematics analogous to sets with structure or Category Theory.

- View as pure data, semiring or confluent sequence compatible rewrites within a global rewrite. Pure data is good for searching ordered by "organic". semiring is good for adapting math ideas from semiring, near-ring, theory etc. Rewrite may provide a way to generalize from the underlying space of all pure data.

Table 2: The eight inner morphisms of $\mathbb{L}_2$ slightly generalize the eight standard symmetric binary boolean operators.

$e_i \in \mathbb{L}_2$	0	T	F	TT	TF	FT	FF	standard	generalized	description
e_1	T	T	T	T	T	T	T	TRUE	**always**	always true
e_2	T	T	T	T	T	T	F	OR	**any**	any are true
e_3	T	T	F	T	F	F	T	XNOR	**even**	even (:)s
e_4	T	T	F	T	F	F	F	AND	**all**	all are true
e_5	F	F	T	F	T	T	T	NAND	**notall**	not all are true
e_6	F	F	T	F	T	T	F	XOR	**odd**	odd (:)s
e_7	F	F	F	F	F	F	T	NOR	**none**	none are true
e_8	F	F	F	F	F	F	F	FALSE	**never**	never true
Number of (:)	0	0	1	0	1	1	2			

- Deeper results, algebra, geometric constructions, Homotopy,...
- Use deep learning to treat evaluation like a game to be learned. Automating, searching, proofs, optimizing computations.

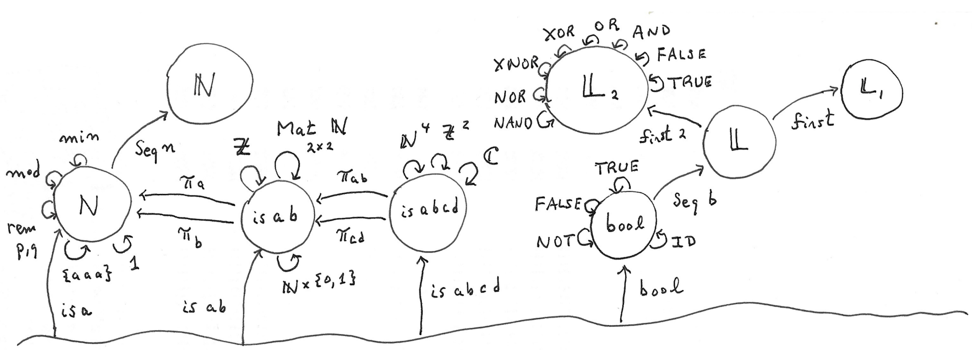

Figure 8: *Some of the spaces and morphisms discussed in the text and "grown organically" from the space of all pure data. The main starting points are natural number spaces (central homomorphism spaces) and bool (central algebraic spaces).*

14 Glossary

1. Basics
   (a) δ_{const} : (const A : B) $\mapsto$ A
   (b) δ_{put} : (put A : B) $\mapsto$ (A:B)
   (c) δ_{get} : (get : (:B)) $\mapsto$ B....FIXME
   (d) δ_{atoms} : (atoms : b B) $\mapsto$ (:) (atoms : B)
   (e) δ_{bin} : (bin A:B) $\mapsto$ (bin A:B), atom
2. Control
   (a) δ_{if}: (if ():B) $\mapsto$ B

   (b) δ_{if}: (if a A:B) $\mapsto$ ()
   (c) δ_{while}: (while A:B) $\mapsto$ B if (A:B)=B
   (d) δ_{while}: (while A:B) $\mapsto$ (while A: A : B)

3. Semiring
   (a) prod
   (b) sum

4. Definition

5. Combinatorics
   (a) map
   (b) various aps

6. Sequence
   (a) (nat:n)-¿n (nat:n+1), example of infinite data. MOVE UP TO TOP?
   (b) δ_{first} : (first : b B) $\mapsto$ b
   (c) δ_{first} : (first : ()) $\mapsto$ ()
   (d) δ_{last} : (last : B b) $\mapsto$ b
   (e) δ_{last} : (last : ()) $\mapsto$ ()
   (f) has
   (g) hasnt
   (h) is
   (i) isnt
   (j) once
   (k) δ_{rev} : (rev : B b) $\mapsto$ b (rev:B)
   (l) δ_{rev} : (rev : ()) $\mapsto$ ()
   (m) rem
   (n) sort

References

[1] *Pure Data Foundation of Mathematics and Computing*, Saul Youssef, 2023.

[2] Nicholas Griffin (2003-06-23). *The Cambridge Companion to Bertrand Russell.* Cambridge University Press. p. 63. ISBN 978-0-521-63634-6.

[3] Barry Mazur, *When is one thing equal to some other thing?*, Harvard University, 2007, https://people.math.harvard.edu/∼mazur/preprints/when_is_one.pdf.