Some Properties of Inclusions of Multisets and Contractive Boolean Operators

Pierre Hyvernat[1]

Université de Savoie,
Laboratoire de Mathématiques,
73376 Le Bourget-du-Lac Cedex,
France

Abstract

Consider the following curious puzzle: call an n-tuple $\overline{X} = (X_1, \dots, X_n)$ of sets smaller than another n-tuple $\overline{Y}$ if it has fewer *unordered sections*. We show that equivalence classes for this preorder are very easy to describe and characterize the preorder in terms of the simpler pointwise inclusion and the existence of a special increasing Boolean operator $f : \mathbf{B}^n \to \mathbf{B}^n$. We also show that contrary to increasing Boolean operators, the relevant operators are not finitely generated, which might explain why this preorder is not easy to describe concretely.

Keywords: multiset, system of representative, Boolean operators
2000 MSC: 06A06, 06E30, 94C10

Introduction: a puzzle

Let N be a (fixed) set and n be a (fixed) natural number. We can consider the following partial order on $\mathcal{P}_*(N)^n$, the collection of n-tuples of *nonempty* subsets of N:

$$\overline{X} \subseteq \overline{Y} \quad \stackrel{\text{def}}{=} \quad \prod_{1\leq i\leq n} X_i \subseteq \prod_{1\leq i\leq n} Y_i$$

where $\prod_i X_i$ is the usual cartesian product. Because we restrict to nonempty subsets, this preorder coincides with pointwise inclusion:

$$(X_1, \dots, X_n) \subseteq (Y_1, \dots, Y_n) \quad \iff \quad \forall 1 \leq i \leq n,\ X_i \subseteq Y_i\ .$$

We now consider a commutative version of the cartesian product where instead of the usual ordered n-tuples, we take "unordered n-tuples".

Definition 1. If $\overline{X} = (X_1, \dots, X_n)$ is an n-tuple of nonempty subsets of N, define $\mathcal{S}(\overline{X})$, the set of *unordered sections* of $\overline{X}$, as

$$\mathcal{S}(\overline{X}) \quad \stackrel{\text{def}}{=} \quad \left(\prod_{1\leq i\leq n} X_i\right) \Big/ \mathsf{S}_n\ , \tag{1}$$

where $\_/\mathsf{S}_n$ denotes quotienting by the action of the symmetric group S_n.

Email address: Pierre.Hyvernat@univ-savoie.fr (Pierre Hyvernat)
URL: http://lama.univ-savoie.fr/~hyvernat/ (Pierre Hyvernat)
[1]This work was partially funded by the French ANR project récré ANR-11-BS02-0010.

Strictly speaking, S_n does not really act on $\prod_i X_i$ but on N^n. The notion is well-defined because the orbit of an element of $\prod_i X_i$ exists even if not all its elements are themselves in $\prod_i X_i$. From now on, we will drop the adjective "unordered" and refer to an element of $\mathcal{S}(\overline{X})$ simply as a section of $\overline{X}$. We now define the preorder $\sqsubseteq$ on $\mathcal{P}_*(N)^n$:

Definition 2. If $\overline{X}$ and $\overline{Y}$ are n-tuples of nonempty subsets of N, we define $\overline{X} \sqsubseteq \overline{Y}$ to mean $\mathcal{S}(\overline{X}) \subseteq \mathcal{S}(\overline{Y})$. We write $\overline{X} \approx \overline{Y}$ for "$\overline{X} \sqsubseteq \overline{Y}$ and $\overline{Y} \sqsubseteq \overline{X}$", that is, for $\mathcal{S}(\overline{X}) = \mathcal{S}(\overline{Y})$.

The relation $\sqsubseteq$ is only a *preorder* because it is not antisymmetric: $(X_{\sigma(1)}, \dots, X_{\sigma(n)}) \approx (X_1, \dots, X_n)$ for any permutation σ.

The aim of this note is to answer the following questions:

1. When do we have $\overline{X} \approx \overline{Y}$?
2. What is the relation between $\overline{X} \sqsubseteq \overline{Y}$ and $\overline{X} \subseteq \overline{Y}$?

The problem is subtler than it appears and the first question makes for an interesting puzzle: while elementary, the proof is more complex than what most people initially think. Readers are thus encouraged to spend a couple of minutes playing with the problem before reading on.

Related notions. The notion of *system of representatives* was introduced by P. Hall in 1935 [3]. A system of representatives for the n-tuple of sets $\overline{X}$ is simply an n-tuple $\overline{x}$ such that there is a permutation σ satisfying $x_i \in X_{\sigma(i)}$ for each $i \in \{1, \dots, n\}$. Equivalence classes of those under permutations are exactly the unordered sections of $\overline{X}$ of Definition 1. A lot of attention has been devoted to systems of *distinct* representatives, also called *transversals*, where the components of $\overline{x}$ are pairwise distinct [8]. Rather than looking at them individually, we look here at the collection of all possible systems of representatives. This shift of focus seems to be new in itself, as is the notion of contractive increasing Boolean operator that appears later. Relating the two will give a concise answer to the second question.

This work can also be seen as a first step toward a *factor theory* for commutative regular algebra [2]. In his book on regular languages, John Conway develops a fascinating theory of factorization: if R is a regular set, a subfactorization is tuple of sets $\overline{X}$ of words satisfying $X_1 \cdot \ldots \cdot X_n \subseteq R$, where $\_ \cdot \_$ denotes concatenation of regular sets. A subfactorization is a *factorization* if each X_i is maximal and a *factor* of R is any such X_i. Conway shows in particular that a regular set has only finitely many factors, and that they are all regular.

Conway devotes a chapter to commutative regular algebra, i.e. the theory arising from regular algebra when word concatenation is made commutative. Factor theory isn't part of this chapter, probably because *"Commutative regular algebra is notable for the number of results whose proofs one would expect to be trivial, but which turn out to be very subtle."* ([2], page 95). Commutative factor theory certainly looks very subtle and this work only gives a very partial answer: given the regular set $Y_1 \cdot \ldots \cdot Y_n$ where each Y_i is a set of *symbols*, we characterize it factorizations consisting of exactly n factors.

The initial motivation for this work comes from a very different area: denotational models of linear logic. In [4], the relation $\mathcal{S}(\overline{X}) \subseteq T$ played an important role, where the set T was an arbitrary collection of n-multisets. Understanding this relation was necessary to compute small examples, and the preorder "$\sqsubseteq$" naturally appeared in this way. (Note that the results of this paper are not to make those computations any easier than they were...)

Notation. To make formulas less verbose, we will abuse the vector notation by lifting "$\in$" pointwise: just as $\overline{X} \subseteq \overline{Y}$ means "$\forall 1 \leq i \leq n,\ X_i \subseteq Y_i$", the notation $\overline{a} \in \overline{X}$ is a synonym for "$a_i \in X_i$ for all $1 \leq i \leq n$". The (left) action of S_n on n-tuples is written with a dot and is defined as $\sigma \cdot \overline{a} \stackrel{\text{def}}{=} (a_{\sigma^{-1}(1)}, \dots, a_{\sigma^{-1}(n)})$. When talking about n-multisets (orbits for the action of S_n), we identify an n-tuple with its orbit. In particular, $\overline{a} \in \mathcal{S}(\overline{X})$ means that $\sigma \cdot \overline{a} \in \overline{X}$ for some permutation σ, i.e., that $a_i \in X_{\sigma(i)}$ for all $1 \leq i \leq n$.

1. The equivalence relation

The first question has a simple answer: equivalence is just equality up to a permutation of the sets. In other words, the failure of antisymmetry is captured by the remark coming after Definition 2.

Proposition 1. *Given any* $\overline{X}$ *and* $\overline{Y}$ *in* $\mathcal{P}_*(N)^n$*, we have*

$$\overline{X} \approx \overline{Y} \quad \Longleftrightarrow \quad \exists \sigma \in \mathsf{S}_n,\ \sigma \cdot \overline{X} = \overline{Y}\ . \tag{2}$$

This proposition is slightly surprising because the left side is definitionally equal to

$$\forall \overline{a} \in \overline{X},\ \exists \sigma \in \mathsf{S}_n,\ \sigma \cdot \overline{a} \in \overline{Y} \quad \text{and} \quad \forall \overline{b} \in \overline{Y},\ \exists \sigma' \in \mathsf{S}_n,\ \sigma' \cdot \overline{b} \in \overline{X}\ , \tag{3}$$

while the right side is definitionally equal to

$$\exists \sigma \in \mathsf{S}_n, \quad \Big(\forall \overline{a} \in \overline{X},\ \sigma \cdot \overline{a} \in \overline{Y} \text{ and } \forall \overline{b} \in \overline{Y},\ \sigma^{-1} \cdot \overline{b} \in \overline{X}\Big)\ .$$

That the latter implies the former is trivial. Proposition 1 asserts the converse: in (3), we can choose the permutation uniformly for all the $\overline{a} \in \overline{X}$ and $\overline{b} \in \overline{Y}$!

Lemma 1. *We have*

1. *If* $\overline{X} \sqsubseteq \overline{Y}$ *then, for all* $1 \leq j \leq n$*, there is some* $1 \leq i \leq n$ *s.t.* $X_i \subseteq Y_j$*.*
2. *If* $\overline{X} \approx \overline{Y}$ *then* $X_{i_0} = Y_{j_0}$ *for some pair* i_0, j_0*.*

Proof. For the first point, suppose that there is some j_0 satisfying $X_i \not\subseteq Y_{j_0}$ for all $1 \leq i \leq n$. This means that there is an $\overline{a} \in \overline{X}$ s.t. $a_i \notin Y_{j_0}$ for all i. This $\overline{a}$ cannot be a section of $\overline{Y}$. Contradiction!

The second point follows easily: starting from Y_1 and repeatedly using the first point, we can construct an infinite sequence $i_1, j_2, i_3, j_4, \ldots$ satisfying:

$$\cdots \quad \subseteq X_{i_{2k+1}} \subseteq Y_{i_{2k}} \subseteq \cdots \subseteq X_{i_3} \subseteq Y_{j_2} \subseteq X_{i_1} \subseteq Y_1$$

Because there are only finitely many possible indices, there are k and k', with $k < k'$ and $i_{2k} = i_{2k'}$. This implies that the sets $X_{i_{2k}}$ and $Y_{j_{2k+1}}$ are equal. □

Thus, if $\overline{X} \approx \overline{Y}$, one of the sets appears on both sides and we can start the construction of σ in (2). To finish the proof of Proposition 1 by induction on n, we need to show the following implication:

$$(Z, X_2, \ldots, X_n) \approx (Z, Y_2, \ldots, Y_n) \ \Rightarrow\ (X_2, \ldots, X_n) \approx (Y_2, \ldots, Y_n)\ . \tag{4}$$

If the collection of unordered sections is seen as a "commutative cartesian product", the next definition would be the corresponding "division".

Definition 3. Let T be a collection of n-multisets and $Z \in \mathcal{P}_*(N)$ we put

$$T \div Z \quad \stackrel{\text{def}}{=} \quad \Big\{(a_2, \ldots, a_n) \mid \forall a \in Z,\ (a, a_2, \ldots, a_n) \in T\Big\}\ .$$

When $Z = \{a\}$, it is a commutative version of Brozowski's *derivative* [1], and in the general case, it corresponds to the commutative notion of *factor* of T with respect to Z [7].

Implication (4) above follows from the following lemma:

Lemma 2. *We have:*

$$\mathcal{S}(X_1, X_2, \ldots, X_n) \div X_1 \quad = \quad \mathcal{S}(X_2, \ldots, X_n)\ . \tag{5}$$

Proof. The "$\supseteq$" inclusion follows from the definition.

For the "$\subseteq$" inclusion, suppose $(a_2, \ldots, a_n) \in \mathcal{S}(\overline{X}) \div X_1$ and choose $b \in X_1$. By hypothesis, we necessarily have $(b, a_2, \ldots, a_n) \in \mathcal{S}(\overline{X})$, i.e., there is a permutation τ s.t. $(b, a_2, \ldots, a_n) \in (X_{\tau(1)}, \ldots, X_{\tau(n)})$.

If $\tau(1) = 1$, then τ defines a permutation on $\{2, \ldots, n\}$ and we have $(a_2, \ldots, a_n) \in (X_{\tau(2)}, \ldots, X_{\tau(n)})$. We can conclude directly.

If $\tau(1) \neq 1$, up to permuting the sets X_2, …, X_n and choosing an appropriate element in the orbit of $(a_2, \ldots a_n)$, we can assume that $\tau(1) = 2$, $\tau(2) = 1$ and $\tau(i) = i$ when $2 < i \leq n$, or in other words, that $b \in X_2$, $a_2 \in X_1$ and $a_i \in X_i$ whenever $2 < i \leq n$.
Put $a_1 \stackrel{\mathsf{def}}{=} a_2$. Because $a_1 = a_2 \in X_1$, we have $(a_1, a_2, \ldots, a_n) \in \mathcal{S}(\overline{X})$ by hypothesis, that is, $\sigma \cdot \overline{a} \in \overline{X}$ for some permutation σ. Note that since $a_1 = a_2$, we can interchange the values $\sigma(1)$ and $\sigma(2)$ and still have $\sigma \cdot \overline{a} \in \overline{X}$.
Let $k \stackrel{\mathsf{def}}{=} \min\{i \mid i > 0, \sigma^i(1) \in \{1, 2\}\}$. Up to changing the values of $\sigma(1)$ and $\sigma(2)$, we can assume that $\sigma^k(1) = 1$ and that the set $I \stackrel{\mathsf{def}}{=} \{1, \sigma(1), \ldots, \sigma^{k-1}(1)\}$ is the cycle containing 1. We define the set I^c as $\{1, \ldots, n\} \setminus I$.
Rearrange the columns of

$$\begin{array}{ccccccc} a_1 & a_2 & & a_i & & a_n \\ \in & \in & & \in & & \in \\ X_{\sigma(1)} & X_{\sigma(2)} & \cdots & X_{\sigma(i)} & \cdots & X_{\sigma(n)} \end{array}$$

into two parts:

$$\underbrace{\begin{array}{ccccc} a_1 & a_{\sigma(1)} & & a_{\sigma^{k-1}(1)} \\ \in & \in & & \in \\ X_{\sigma(1)} & X_{\sigma^2(1)} & \cdots & X_{\sigma^k(1)} = X_1 \end{array}}_{I} \qquad \underbrace{\begin{array}{ccccc} a_2 & & a_i & & \\ \in & & \in & & \\ X_{\sigma(2)} & \cdots & X_{\sigma(i)} & \cdots \end{array}}_{I^c} .$$

The indices of $\overline{X}$ on the left are exactly those in I, and so are the indices of $\overline{a}$. Thus, the indices of $\overline{X}$ and $\overline{a}$ on the right are exactly those in I^c. This shows that $(a_i)_{i \in I^c}$ is a section of $(X_i)_{i \in I^c}$. Also, because each of $\sigma(1)$, …, $\sigma^{k-1}(1)$ is strictly more than 2 (by the definition of k), we have $a_{\sigma^i(1)} \in X_{\sigma^i(1)}$ for all $1 \leq i \leq k-1$ by a previous hypothesis. This shows that the permutation

$$\rho : \{2, \ldots, n\} \to \{2, \ldots, n\}, \quad \rho(i) \stackrel{\mathsf{def}}{=} \begin{cases} i & \text{if } i \in \{\sigma(1), \ldots, \sigma^{k-1}(1)\} \\ \sigma(i) & \text{otherwise} \end{cases}$$

satisfies $\rho \cdot (a_2, \ldots, a_n) \in (X_2, \ldots, X_n)$. This finishes the proof that $(a_2, \ldots, a_n)$ is indeed a section of $(X_2, \ldots, X_n)$. □

2. The preorder

The initial question was not very formal and read as: "*What is the relation between $\overline{X} \sqsubseteq \overline{Y}$ and $\overline{X} \subseteq \overline{Y}$?*" It is obvious that $\overline{X} \subseteq \overline{Y}$ implies $\overline{X} \sqsubseteq \overline{Y}$, but unfortunately, the converse does not hold, even if we consider n-tuples of sets up to permutations. For example, we have

$$\overline{X} := \big(\{3\}, \{1, 2, 3\}\big) \quad \sqsubseteq \quad \overline{Y} := \big(\{1, 3\}, \{2, 3\}\big)$$

because the sections of $\overline{X}$ are all sections of $\overline{Y}$:

$$\mathcal{S}(\overline{X}) = \big\{[3, 1], [3, 2], [3, 3]\big\} \quad \subset \quad \mathcal{S}(\overline{Y}) = \big\{[1, 2], [1, 3], [3, 2], [3, 3]\big\} .$$

Lemma 1 asserts that each set on the right is a superset of some set on the left. This is indeed the case as both $\{1, 3\}$ and $\{2, 3\}$ are supersets of the same set $\{3\}$. However, one set on the left side is strictly bigger than all the sets on the right side: $\{1, 2, 3\} \supset \{2, 3\}$ and $\{1, 2, 3\} \supset \{1, 3\}$!

More generally, $\big(Y_1 \cap Y_2, Y_1 \cup Y_2\big) \sqsubseteq \big(Y_1, Y_2\big)$ and any operator F on n-tuples of sets obtained by composing functions $(Y_i, Y_j) \mapsto (Y_i \cap Y_j, Y_i \cup Y_j)$ on any pairs of coordinates,[2] will satisfy $F\big(\overline{Y}\big) \sqsubseteq \overline{Y}$. For example,

$$\begin{array}{lll} F(Y_1, Y_2, Y_3) & \stackrel{\text{def}}{=} & \Big(Y_1 \cap Y_3, Y_2 \cap (Y_1 \cup Y_3), Y_2 \cup (Y_1 \cup Y_3)\Big) \\ & \sqsubseteq & \big(Y_1 \cap Y_3, Y_2, Y_1 \cup Y_3\big) \\ & \sqsubseteq & \big(Y_1, Y_2, Y_3\big)\ . \end{array}$$

We will characterize (Proposition 3) which operators F on $\mathcal{P}_*(N)^n$ satisfy $F(\overline{Y}) \sqsubseteq \overline{Y}$ by looking at functions acting on tuples of Boolean values, i.e., *Boolean operators*.

Definition 4. Let $\mathbf{B} \stackrel{\text{def}}{=} \{0, 1\}$ equipped with the order $0 \leq 1$. This is a complete lattice with operations written $\vee$ and $\wedge$. The lattice structure is lifted pointwise to $\mathbf{B}^n$.
If $u \in \mathbf{B}^n$, the *weight* of u is the number of 1s in u. It is written $|u|$.

We write elements of $\mathbf{B}^n$ as words: for example, "$011101 \in \mathbf{B}^6$".

Definition 5. If $a \in N$ and $\overline{X} \in \mathcal{P}_*(N)^n$, the *characteristic word of a along $\overline{X}$* is an element of $\mathbf{B}^n$. It is written $\chi_{\overline{X}}(a)$ and is defined by $\big(\chi_{\overline{X}}(a)\big)_i = 1$ iff $a \in X_i$.

Thus, $\chi_{\overline{X}}(a)$ describes in which components of $\overline{X}$ the element a appears, and $|\chi_{\overline{X}}(a)|$ is the number of the components of $\overline{X}$ which contain a. There is a necessary condition for $\overline{X} \sqsubseteq \overline{Y}$: "for all $a \in N$, if a appears in k components of $\overline{Y}$, then it appears in at most k components of $\overline{X}$". Concisely, this condition can be written as "$|\chi_{\overline{X}}(a)| \leq |\chi_{\overline{Y}}(a)|$ for all $a \in N$". This condition is reminiscent of the condition appearing in Hall's celebrated "marriage theorem" [3]. Like in the marriage theorem, this condition is also sufficient in the appropriate setting:

Proposition 2. *Given $\overline{X}$ and $\overline{Y}$ two n-tuples of non-empty subsets of N that satisfy*

1. *the function $a \mapsto \chi_{\overline{Y}}(a)$ is bijective from N to $\mathbf{B}^n \setminus \{0 \cdots 0\}$,*
2. *the function $f : \chi_{\overline{Y}}(a) \mapsto \chi_{\overline{X}}(a)$, with domain $\mathbf{B}^n \setminus \{0 \cdots 0\}$ is increasing;*

we have $\overline{X} \sqsubseteq \overline{Y}$ if and only if $\big|\chi_{\overline{X}}(a)\big| \leq \big|\chi_{\overline{Y}}(a)\big|$ for all $a \in N$.

Looking at the example $\big(\{3\}, \{1, 2, 3\}\big) \sqsubseteq \big(\{2, 3\}, \{1, 3\}\big)$ might help to understand the conditions of the proposition. The first condition means that there is exactly one element that belongs only to Y_1 ("2"), exactly one element that belongs only to Y_2 ("1") and exactly one element that belongs to both ("3"). When the first condition is satisfied, the second condition amounts to "when a appears in more sets than b on the right side, then a appears in more sets than b on the left side". The graph of the resulting function f can be read below

a :	1	2	3
$\chi_{\overline{Y}}(a)$:	01	10	11
$\chi_{\overline{X}}(a)$:	01	01	11

.

We can extend this graph with a harmless $0 \cdots 0 \mapsto 0 \cdots 0$ to obtain the function $(b_1, b_2) \mapsto (b_1 \wedge b_2, b_1 \vee b_2)$. Proposition 2 follows from a more general lemma:

Lemma 3. *We have $\overline{X} \sqsubseteq \overline{Y}$ iff $f(u) \stackrel{\text{def}}{=} \bigvee_{\chi_{\overline{Y}}(a) \leq u} \chi_{\overline{X}}(a)$ satisfies $|f(u)| \leq |u|$ for all u.*

The function f is the least increasing function (for the extensional order) satisfying $f\big(\chi_{\overline{Y}}(a)\big) \geq \chi_{\overline{X}}(a)$ for any a.

[2] provided the intersection isn't empty to agree with Definition 1

PROOF. For the "$\Leftarrow$" implication, suppose that $\overline{a} \in \overline{X}$. We want to show that $\sigma \cdot \overline{a} \in \overline{Y}$ for some permutation σ. If we look at the bipartite graph $G_{\overline{a},\overline{Y}}$

$$G_{\overline{a},\overline{Y}} \quad \stackrel{\text{def}}{=} \quad \begin{array}{cccccc} & a_1 & & a_2 & \cdots & a_n \\ & \bullet & & \bullet & & \bullet \\ & & & & & \\ & \bullet & & \bullet & & \bullet \\ Y_1 & & Y_2 & & \cdots & Y_n \end{array} ,$$

with an edge between a_i and Y_j when $a_i \in Y_j$, finding a σ s.t. $\sigma \cdot \overline{a} \in \overline{Y}$ is equivalent to finding a perfect matching in $G_{\overline{a},\overline{Y}}$. By Hall's marriage theorem, this is equivalent to "every subset of $\{a_1, \ldots, a_n\}$ of cardinality p has at least p neighbors".

Take some subset $U \subseteq \{a_1, \ldots, a_n\}$ of cardinality p. Because $\overline{a}$ is a section of $\overline{X}$, this set has at least p neighbors in the corresponding $G_{\overline{a},\overline{X}}$ graph. Let $u \stackrel{\text{def}}{=} \bigvee_{a \in U} \chi_{\overline{Y}}(a)$, if $a \in U$, then $\chi_{\overline{Y}}(a) \leq u$ by definition of u, so that $\chi_{\overline{X}}(a) \leq f(u)$ by definition of f. We thus have $\bigvee_{a \in U} \chi_{\overline{X}}(a) \leq f(u)$. We get

$$p \quad \leq \quad \underbrace{\left| \bigvee_{a \in U} \chi_{\overline{X}}(a) \right|}_{\substack{\text{\# of neighboors} \\ \text{of } U \text{ in } G_{\overline{a},\overline{X}}.}} \quad \leq \quad |f(u)| \quad \leq \quad |u| \quad \stackrel{\text{def}}{=} \quad \underbrace{\left| \bigvee_{a \in U} \chi_{\overline{Y}}(a) \right|}_{\substack{\text{\# of neighboors} \\ \text{of } U \text{ in } G_{\overline{a},\overline{Y}}.}} ,$$

which concludes the "$\Leftarrow$" implication.

For the "$\Rightarrow$" implication, let $\overline{X} \sqsubseteq \overline{Y}$ and $p = |u| < |f(u)|$. By definition of f, we can find a set $\{a_1, \ldots, a_k\} \subseteq N$ which satisfies $\chi_{\overline{Y}}(a_i) \leq u$ for $i = 1, \ldots, k$ and $\left| \bigvee_{i \leq k} \chi_{\overline{X}}(a_i) \right| > p$. In particular, we have

$$\left| \bigvee_{1 \leq i \leq k} \chi_{\overline{Y}}(a_i) \right| \quad \leq \quad p \quad < \quad \left| \bigvee_{1 \leq i \leq k} \chi_{\overline{X}}(a_i) \right| .$$

For each 1 in $\bigvee_{i \leq k} \chi_{\overline{X}}(a_i)$ take one element of $\{a_1, \ldots, a_k\}$ that accounts for this 1. Call the resulting tuple $\overline{a}$. Note that this might not be an n-tuple, but its length is strictly greater than p (and may contain repetitions). It is only a partial section of $\overline{X}$: to complete it into a section of the whole $\overline{X}$, simply add one element from each of the remaining (non-empty) sets. The result is also a section of $\overline{Y}$ and in particular, each element of $\overline{a}$ needs to fit in one component of $\overline{Y}$. This is impossible because there are at most p sets Y_j that can contain the elements of the tuple $\overline{a}$. Contradiction!

□

Lemma 3 does characterize the $\sqsubseteq$ preorder but still looks a little ad-hoc. We now give a more concise characterization that relates $\sqsubseteq$ with $\subseteq$, thus answering our initial question. First note that we can lift any $f : \mathbf{B}^n \to \mathbf{B}^m$ to a function $\mathcal{P}(N)^n \to \mathcal{P}(N)^m$:

Definition 6. Suppose $f : \mathbf{B}^n \to \mathbf{B}^m$, define $\widehat{f} : \mathcal{P}(N)^n \to \mathcal{P}(N)^m$ as

$$\widehat{f}\left(\overline{Y}\right) \stackrel{\text{def}}{=} \overline{X} \quad \text{with } a \in X_i \text{ iff } f\left(\chi_{\overline{Y}}(a)\right) \text{ has a 1 at coordinate } i \ .$$

This transformation is, in a precise categorical sense, *natural*. It amounts to lifting the Boolean operations $\wedge$ and $\vee$ to their set theoretic versions $\cap$ and $\cup$ in a way that is compatible with function composition. For example, with the "and/or" function $(b_1, b_2) \mapsto (b_1 \wedge b_2, b_1 \vee b_2)$ we obtain $\left(Y_1, Y_2\right) \mapsto \left(Y_1 \cap Y_2, Y_1 \cup Y_2\right)$.

Definition 7. Call an increasing function $f : \mathbf{B}^n \to \mathbf{B}^n$ *contractive* if it satisfies $|f(u)| \leq |u|$ for all $u \in \mathbf{B}^n$.

A corollary to Lemma 3 is:

Proposition 3. *For any $\overline{X}$ and $\overline{Y}$, we have*

$$\overline{X} \sqsubseteq \overline{Y} \quad \Longleftrightarrow \quad \overline{X} \subseteq \widehat{f}\left(\overline{Y}\right) \quad \textit{for some increasing, contractive } f : \mathbf{B}^n \to \mathbf{B}^n .$$

PROOF. We know that $\overline{X} \sqsubseteq \overline{Y}$ is equivalent to having $|f(u)| \leq |u|$ for all u in $\mathbf{B}^n$, where f is defined as in Lemma 3. This function f satisfies $\overline{X} \subseteq \widehat{f}(\overline{Y})$: use $u \stackrel{\text{def}}{=} \chi_{\overline{Y}}(a)$ to check that $a \in X_i$ is an element of the i-th set of $\widehat{f}(\overline{Y})$.

For the converse, suppose f is contractive increasing with $\overline{X} \subseteq \widehat{f}(\overline{Y})$ and let $\chi_{\overline{Y}}(a) \leq u$. Suppose that $\chi_{\overline{X}}(a)$ contains a 1 in position i. This means that $a \in X_i$ and thus a is in the i-th set of $\widehat{f}(\overline{Y})$. We can conclude that $f\big(\chi_{\overline{Y}}(a)\big)$ contains a 1 in position i. This implies that $f(u)$ also contains a 1 in position i. □

3. Contractive functions are not finitely generated

If one had a simple representation of contractive increasing Boolean operators from $\mathbf{B}^n$ to itself, then Proposition 3 would give a simple representation of the $\sqsubseteq$ preorder. It is well known that all Boolean operators $\mathbf{B}^n \to \mathbf{B}^m$ with n inputs and m outputs can be represented by a Boolean circuit using only "and", "or" together with "not" cells. Strictly speaking, we also need constant values and need a way to forget or duplicate inputs. The complete set of cells is depicted in Figure 1, where the last cells are:

- constants 1 and 0 (zero input, one output),
- drop (one input, zero output),
- duplicate (one input, two outputs),
- crossing (two inputs, two outputs).

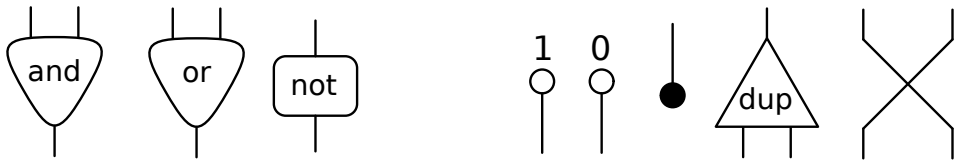

Figure 1: Boolean cells

These cells, together with a finite set of relations expressing properties of the operations (associativity, etc.), give a finite presentation of the monoidal category of Boolean operators.[3] We can generate the subcategory of increasing operators by removing the "not" cell from the generators. Unfortunately, no such thing is possible for *contractive increasing* Boolean operators.

Proposition 4. *Contractive increasing Boolean operators are not finitely generated.*

In other words, any finite set of cells will either miss some contractive Boolean operator, or generate some non contractive Boolean operator.

First, a preliminary lemma:

Lemma 4. *Let $f : \mathbf{B}^n \to \mathbf{B}^n$ be an increasing contractive Boolean operator; the following are equivalent:*

1. *f is the action of a permutation $u \mapsto \sigma \cdot u$ for some $\sigma \in \mathsf{S}_n$,*
2. *f is bijective,*
3. *f is injective on words of weight 1.*

[3] All of this has a precise algebraic meaning, see [6] for details.

Proof. Trivially, 1 implies 2 and 2 implies 3. Suppose now that f is increasing and contractive on $\mathbf{B}^n$. Suppose moreover that f is injective on words of weight 1. We can define a permutation σ on $\{1,\dots,n\}$ by putting

$$\tau(i) = j \quad \text{iff} \quad f(e_i) = e_j$$

where e_i represents the word with a single 1 in position i. If u contains a 1 in position i, then, because f is increasing, $f(u)$ must contain a 1 in position $\tau(i)$. Because f is contractive, $f(u)$ cannot contain more 1s than there are in u. Thus, the 1s of $f(u)$ correspond exactly to the images of the 1s of u along τ: f is indeed the action of a permutation. □

Proof (Proposition 4). Suppose, by contradiction, that there is a finite set of cells that generates all increasing contractive Boolean operators, and write m for the maximal arity of the cells in this set.
Any non-invertible function has a representation as in Figure 2 where

- the topmost rectangle contains only crossings (and invertible cells which are, by Lemma 4, equivalent to crossings),
- the cell C is not invertible and has arity $c \leq m$,
- and the lowermost rectangle contains the rest of the circuit.

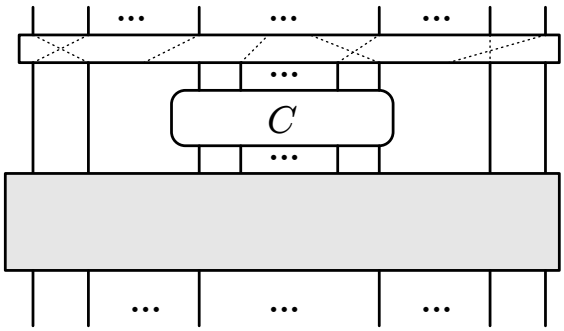

Figure 2: A Boolean circuit

By Lemma 4, we know that the cell C is not injective on inputs of weight 1. It means there are two input wires i_1 and i_2 s.t. C gives the same value on the two elements of $\mathbf{B}^n$ consisting of 0s and a single 1 in position i_1 or i_2. Because this is independent of the inputs $\overline{v}$ on the $n-m$ remaining wires, we obtain:

Claim. *Supposing contractive increasing Boolean operators were finitely generated with a cell of arity less than m, then for any non-invertible contractive increasing Boolean operator $f : \mathbf{B}^n \to \mathbf{B}^n$, with $n \geq m$, we have:*

$$\exists \sigma \in \mathsf{S}_n \quad \forall \overline{v} \in \mathbf{B}^{n-m} \quad f\big(\sigma(01.\overline{0}.\overline{v})\big) = f\big(\sigma(10.\overline{0}.\overline{v})\big) \ .$$

The permutation σ is used to simplify the notation: it reorders the wires to put i_1 and i_2 in positions 1 and 2, and the remaining input wires for C in positions $3,\dots,c$.

For any maximal arity m, we will construct a (large) n together with a function $f : \mathbf{B}^n \to \mathbf{B}^n$ that contradicts this fact: whenever we choose input wires i_1 and i_2 and put any $m-2$ other input wires to 0, we can complete the remaining input wires in such a way that putting $i_1 \overset{\text{def}}{=} 0$ and $i_2 \overset{\text{def}}{=} 1$, or putting $i_2 \overset{\text{def}}{=} 0$ and $i_1 \overset{\text{def}}{=} 1$ makes a difference in the output of the function. Thus, this function will not be representable using the given set of cells.

Given a (large) n, define $f : \mathbf{B}^n \to \mathbf{B}^n$ as:

$$f(u) \quad \stackrel{\text{def}}{=} \quad \begin{cases} 0^n & \text{if } |u| = 0 & (1) \\ 1\ 0^{n-1} & \text{if } |u| = 1 & (2) \\ 1^k\ 0^{n-k} & \text{if } |u| = k \text{ is even} & (3) \\ 1101\ 0^{n-4} & \text{if } u = 0\cdots0\ 110^l1\ 0\cdots0, \text{ with } l > 0 & (4) \\ 1110\ 0^{n-4} & \text{if } |u| = 3 \text{ but } u \text{ not of the previous shape} & (5) \\ 1^{2^k}01\ 0^{n-2^k-2} & \text{if } u = 0\cdots0\ 1^{2^k}0^{2^k}1\ 0\cdots0, \text{ with } k > 1 & (6) \\ 1^{2^k}01\ 0^{n-2^k-2} & \text{if } u = 0\cdots0\ 10^{2^k}1^{2^k}\ 0\cdots0, \text{ with } k > 1 & (7) \\ 1^{2k}10\ 0^{n-2k-2} & \text{in all the remaining cases.} & (8) \end{cases}$$

This function is contractive because we have $|f(u)| = |u|$. Moreover, it is increasing because whenever v is a successor[4] of u, we have $f(v) > f(u)$:

- $f(u) = 1^{2k}\ 0\cdots$ when $|u| = 2k$
- $f(u) = 1^{2k}10\ 0\cdots$ or $f(u) = 1^{2k}01\ 0\cdots$ when $|u| = 2k + 1$.

Suppose input wires $k_1, \ldots, k_{m-2}$ are fixed to 0 and we want to differentiate between input wires i_1 and i_2, with $i_1 < i_2$. By putting some 1s in the appropriate remaining wires, we can make f give different results when "$i_1 \stackrel{\text{def}}{=} 0$, $i_2 \stackrel{\text{def}}{=} 1$" and "$i_1 \stackrel{\text{def}}{=} 1$, $i_2 \stackrel{\text{def}}{=} 0$".

- If there are two consecutive wires between i_1 and i_2 (but not touching i_2) which are not among k_1, $\ldots$, k_{m-2}, we put those two wires to 1 and all the other wires to 0. By lines (4) and (5) from the definition of f, we will get two different results.
- If not, the wires i_1 and i_2 cannot be too far apart. (There are at most $2m - 2$ wires between them...) If we can find a sequence of 2^k consecutive wires at distance 2^k to the left of i_1, or a sequence of 2^k consecutive wires at distance 2^k to the right of i_2, we can put those wires to 1 and the rest to 0. By lines (6) and (8) or (7) and (8) of the definition of f, we will also get different results.

For this to work, we have to make sure n is big enough. At worst, the wires $k_1,\ldots, k_{m-2}$ can prevent us from finding an appropriate sequence $m - 2$ times. In particular, if i_1 is big enough (bigger than 2^{m+1}), such a sequence is bound to happen. The same is true when i_2 is small enough compared to n. In the end, choosing n bigger than, say, 2^{2m+2} plus an additional ε will guarantee that we can differentiate any i_1 and i_2 among any set of m wires. A more careful analysis shows that it is in fact enough to take $n \stackrel{\text{def}}{=} 2^{m+1} + 4$. This concludes the proof. □

References

[1] Brzozowski, J. A., Oct. 1964. Derivatives of regular expressions. J. ACM 11 (4), 481–494.

[2] Conway, J. H., 1971. Regular Algebra and Finite Machines. Chapman and Hall.

[3] Hall, P., 1935. On representatives of subsets. Journal of the London Mathematical Society 10, 26–30.

[4] Hyvernat, P., September 2004. Predicate transformers and linear logic: yet another denotational model. In: Marcinkowski, J., Tarlecki, A. (Eds.), 18th International Workshop CSL 2004. Vol. 3210 of Lecture Notes in Computer Science. Springer-Verlag, pp. 115–129.

[5] Knuth, D. E., Jan. 2011. Combinatorial Algorithms. Vol. 4A of The Art of Computer Programming. Addison-Wesley Professional.

[6] Lafont, Y., 2003. Towards an algebraic theory of boolean circuits. Journal of Pure and Applied Algebra 184 (23), 257 – 310.

[7] Marin, M., Kutsia, T., 2010. On the computation of quotients and factors of regular languages. Frontiers of Computer Science in China 4 (2), 173–184.

[8] Mirsky, L., 1971. Transversal theory; an account of some aspects of combinatorial mathematics. Mathematics in Science and Engineering. Elsevier Science.

[4] v is a *successor* of u if $v > u$ and $|v| = |u| + 1$.

Appendix A. An algorithm

Proposition 3 is more elegant but Lemma 3 has an interesting byproduct: it gives a concrete algorithm to check if $\overline{X} \sqsubseteq \overline{Y}$. For that, construct the function f from Lemma 3 and check that it satisfies the condition. Just as a proof of concept, here is the main part of the algorithm, in the Python programming language. Minor alterations have been made to make it more readable. The most difficult (and fun) part was to write the function `combinations` that generates all the vectors of length `n` and weight `w` using one of the subtle algorithms from [5]![5]

```
def check(N,n,X,Y):
    # N is a set, n is an integer, X / Y are tuples of sets.
    def combinations(w):
        # generates all vectors of weight w
        # omitted (see Knuth, or use you favorite method)
    def sup(u,v):            # complexity: O(n)
        # computes the pointwise "or" on n-tuples
        # omitted (simple)
    def weight(u):           # complexity: O(n)
        # computes the weight of an n-tuple
        # omitted (simple)
    def chi(a,Z):            # complexity: O(n log(z)) (z is cardinality of Z)
        for i in range(n):   # we use Python builtin "set" type
            if a in Z[i]:    # so that "a in Z[i]" mean "a belongs to Z[i]"
                u[i] = 1
        return u

    F = {}                   # F is a finite map with at most 2^n elements,
                             # access is logarithmic: O(log(2^n)) = O(n)
    for a in N:                          # complexity: c *
      chiX = chi(a,X)                    #                   n log(x)
      chiY = chi(a,Y)                    #                 + n log(y)
      F[chiY] = sup(F[chiY] , chiX)      #                 + 2n
    for w in range(n+1):                 # generating all tuples
        for u in combinations(w):        # complexity : about 2^n *
            v = F[u]                                    #           n
            for i in range(n):                          #         + n *
                if u[i] == 1:                           #
                    u[i] = 0                            #
                    v = sup(v,F[u])                     #                 n^2
                    u[i] = 1                            #
            F[u] = v                                    #
            if weight(v) > w:                           #         + n
                return False
    return True      # if we reached this far, the condition is satisfied
```

If N has cardinality c and the components of $\overline{X}$ and $\overline{Y}$ have cardinalities at most x and y; and if we suppose that the standard operations on sets and finite functions have logarithmic complexity, the hints in the comments give a total complexity of roughly $O\big(cn\big(\log(x)+\log(y)\big)+2^n n^3\big)$. Because both $x = O(c)$ and $y = O(c)$, we get a complexity of $O\big(nc\log(c)+2^n n^3\big)$. If c is fixed, this is $O(n^3 2^n)$; if n is fixed, this is $O\big(c\log(c)\big)$. In almost all cases, this is better (and much easier to write) than the naive approach that checks if each $\overline{a} \in \overline{X}$ is a section of $\overline{Y}$, even if we are allowed to use an oracle to guess the permutations.

[5] The complete file is available from http://lama.univ-savoie.fr/~hyvernat/research.php